Score-Based Diffusion meets Annealed Importance Sampling

Arnaud Doucet, Will Grathwohl, Alexander G. D. G. Matthews & Heiko Strathmann *
DeepMind
{arnauddoucet,wgrathwohl,alexmatthews,strathmann}@google.com

Abstract

More than twenty years after its introduction, Annealed Importance Sampling (AIS) remains one of the most effective methods for marginal likelihood estimation. It relies on a sequence of distributions interpolating between a tractable initial distribution and the target distribution of interest which we simulate from approximately using a non-homogeneous Markov chain. To obtain an importance sampling estimate of the marginal likelihood, AIS introduces an extended target distribution to reweight the Markov chain proposal. While much effort has been devoted to improving the proposal distribution used by AIS, an underappreciated issue is that AIS uses a convenient but suboptimal extended target distribution. We here leverage recent progress in score-based generative modeling (SGM) to approximate the optimal extended target distribution minimizing the variance of the marginal likelihood estimate for AIS proposals corresponding to the discretization of Langevin and Hamiltonian dynamics. We demonstrate these novel, differentiable, AIS procedures on a number of synthetic benchmark distributions and variational auto-encoders.

1 Introduction

Evaluating the marginal likelihood, also known as evidence, is of key interest in Bayesian statistics as it allows not only model comparison but is also often used to select hyperparameters. A large variety of Monte Carlo methods have been proposed to address this problem, including path sampling [19], AIS [37] and related Sequential Monte Carlo methods [13]. An appealing feature of AIS is that it provides an unbiased estimate of the marginal likelihood and can thus be used to define an evidence lower bound (ELBO) or mutual information bounds; see e.g. [53, 51, 7].

AIS builds a proposal distribution using a Markov chain $(x_k)_{k=0}^K$ initialized at an easy-to-sample distribution followed by a sequence of Markov chain Monte Carlo (MCMC) transitions targeting typically annealed versions of the posterior. By proceeding this way, we obtain a proposal x_K whose distribution is expected to be a reasonable approximation to the target posterior. However, this distribution is intractable as it requires integrating the joint proposal distribution over previous states $(x_k)_{k=0}^{K-1}$. AIS bypasses this issue by instead using Importance Sampling (IS) on the whole path $(x_k)_{k=0}^K$ through the introduction of an artificial extended target distribution whose marginal at time K coincides with the posterior.

There has been much work devoted to improving AIS in machine learning and statistics but also in physics where it was introduced independently in [29, 9]. A standard approach to improve AIS is to modify the intermediate distributions [45, 21, 34] and corresponding transition kernels of the proposal [10, 53, 18, 51, 54]. We here address a distinct problem. For a given proposal, it was shown in [13] that the extended target distribution minimizing the variance of the evidence estimate is not

*alphabetical order, equal contribution.

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

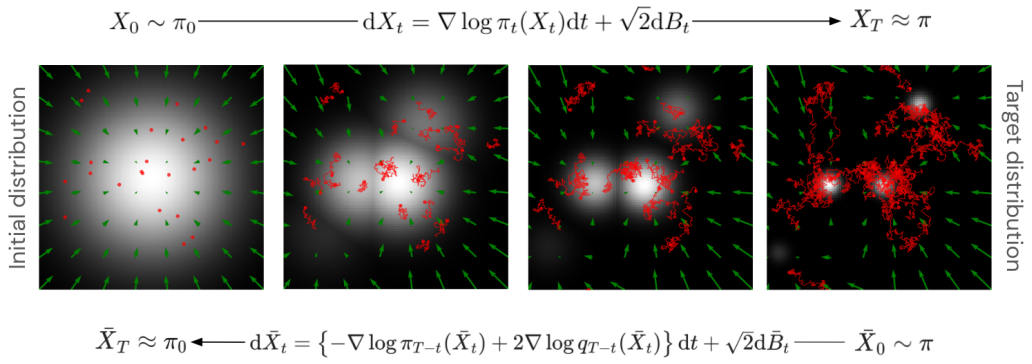


Figure 1: **Top**: Samples X_t from an AIS proposal (red) obtained by sampling initially from a Gaussian at $t = 0$ and diffusing through Langevin dynamics on intermediate targets π_t (white). The intermediate marginals of the proposal, q_t, approximated by the samples are such that $q_T \approx \pi$ for a reasonably fast mixing diffusion. **Bottom**: Computing importance weights. The optimal extended target used to compute the weights is the distribution obtained by initializing $\bar{X}_0$ exactly from π and then following the reverse-time dynamics of the forward AIS proposal. This requires access to score vectors of the marginals q_t.

the one used by AIS but is instead defined through the time-reversal of the proposal. However, this result is difficult to exploit algorithmically as the time-reversal is intractable for useful proposals.

In this paper, we show how one can combine this result with recent advances in SGM to obtain improved, lower variance, AIS estimates. We concentrate on scenarios where we use unadjusted overdamped Langevin [23, 53, 51] and unadjusted Hamiltonian proposals with partial momentum refreshment (i.e. underdamped Langevin) [10, 53, 18, 54, 28] which correspond to time-discretized diffusion processes. The first benefit of using such proposals is that, by omitting Metropolis–Hastings steps, one obtains differentiable versions of the Evidence Lower Bound (ELBO) amenable to the reparameterization trick. The second benefit of these proposals is that their time-reversal can be approximated by adapting techniques developed for SGM [24, 48, 15] to our setup. We derive a principled parameterization for an approximation of their time-reversal which we learn by maximizing the ELBO. As for SGM, this ELBO coincides with a denoising score matching loss [27, 52, 24, 48]. This provides novel, optimized and differentiable, AIS estimators which we refer to as Monte Carlo Diffusion (MCD). We demonstrate the benefits of this approach on synthetic benchmark distributions and variational auto-encoders (VAEs) [31]. All proofs can be found in the Appendix. A preliminary version of this work appeared in [16].

2 Annealed Importance Sampling

2.1 Setup and algorithm

Consider a probability density π on $\mathbb{R}^d$ of the form

$$\pi(x) = \frac{\gamma(x)}{Z}, \qquad Z = \int_{\mathbb{R}^d} \gamma(x)\mathrm{d}x, \tag{1}$$

where $\gamma(x)$ can be evaluated pointwise. We want to approximate the intractable normalizing constant Z. In a Bayesian framework, $\gamma(x) = p(x)p(\mathcal{D}|x)$ is the joint density of parameter x and data $\mathcal{D}$, $\pi(x) = p(x|\mathcal{D})$ the corresponding posterior and $Z = p(\mathcal{D})$ the evidence.

To estimate Z, AIS introduces the intermediate distributions $(\pi_k)_{k=1}^K$ bridging smoothly from a tractable distribution π_0 to the target distribution $\pi_K = \pi$ of interest. One typically uses $\pi_k(x) \propto \gamma_k(x)$ with $\gamma_k(x) = \pi_0(x)^{1-\beta_k}\gamma(x)^{\beta_k}$ for $0 = \beta_0 < \beta_1 < \cdots < \beta_K = 1$ but other choices are possible [21]. The IS proposal used by AIS is then obtained by running a Markov chain $(x_k)_{k=0}^K$ such that $x_0 \sim \pi_0(\cdot)$, and then $x_k \sim F_k(\cdot|x_{k-1})$ for $k \geq 1$ where F_k is a MCMC kernel invariant w.r.t. π_k. The proposal is thus given by

$$Q(x_{0:K}) = \pi_0(x_0) \prod_{k=1}^K F_k(x_k|x_{k-1}). \tag{2}$$

Denote by q_k the marginal distribution of x_k under Q satisfying $q_k(x_k) = \int q_{k-1}(x_{k-1})F_k(x_k|x_{k-1})\mathrm{d}x_{k-1}$ for $k \geq 1$ and $q_0 = \pi_0$, it is typically intractable for $k \geq 1$. As q_K cannot be evaluated in complex scenarios, the marginal IS estimate $w_{\text{mar}}(x_K) = \gamma(x_K)/q_K(x_K)$ of Z is intractable.

One can bypass this issue by introducing an extended target distribution

$$P(x_{0:K}) = \frac{\Gamma(x_{0:K})}{Z}, \qquad \Gamma(x_{0:K}) = \gamma(x_K)\prod_{k=0}^{K-1} B_k(x_k|x_{k+1}), \tag{3}$$

where $(B_k)_{k=0}^{K-1}$ are backward Markov transition kernels, i.e. $\int B_k(x_k|x_{k+1})\mathrm{d}x_k = 1$ for any x_{k+1}, so that by construction $x_K \sim \pi$ under P. For any selection of backward kernels such that the ratio Γ/Q is well-defined, we then have

$$\mathbb{E}_Q[w(x_{0:K})] = Z, \quad \text{for } w(x_{0:K}) = \frac{\Gamma(x_{0:K})}{Q(x_{0:K})}, \tag{4}$$

i.e. $w(x_{0:K})$ is an unbiased estimate of Z for $x_{0:K} \sim Q$.

The AIS estimate of the evidence is a specific instance of the estimator (4) relying on the backward kernels $B_k^{\text{ais}}(x_k|x_{k+1}) = \pi_{k+1}(x_k)F_{k+1}(x_{k+1}|x_k)/\pi_{k+1}(x_{k+1})$. This yields the following expression for $\log w(x_{0:K})$:

$$\log w_{\text{ais}}(x_{0:K}) = \sum_{k=1}^{K} \log\left(\gamma_k(x_{k-1})/\gamma_{k-1}(x_{k-1})\right). \tag{5}$$

2.2 Limitations of AIS

While designing P in (3) by using the backward Markov kernels $(B_k^{\text{ais}})_{k=0}^{K-1}$ is convenient, it is also suboptimal in terms of variance. For example, consider the ideal scenario where $F_k(x_k|x_{k-1}) = \pi_k(x_k)$. This scenario has been used many times in the literature to provide some guidelines on AIS, see e.g. [37, 21]. In this case, $\text{var}_Q[\log w_{\text{ais}}(x_{0:K})] = \sum_{k=1}^{K} \text{var}_{\pi_{k-1}}[\log(\gamma_k(x_{k-1})/\gamma_{k-1}(x_{k-1}))] > 0$ while $\text{var}_{q_K}[w_{\text{mar}}(x_K)] = \text{var}_\pi[w_{\text{mar}}(x_K)] = 0$.

Another illustration of the suboptimality of AIS is to consider a scenario where the proposal is a homogeneous MCMC chain, i.e. $x_0 \sim \pi_0$ and $x_k \sim F(\cdot|x_{k-1})$ for F a π-invariant MCMC kernel; i.e. use $F_k = F$ and $\pi_k = \pi$ for $k = 1, ..., K$. If F is reasonably well-mixing, then $q_K \approx \pi$ for K large enough and the evidence estimate $w_{\text{mar}}(x_K) = \gamma(x_K)/q_K(x_K)$ should have small variance. However, it is easy to check that we have $w_{\text{ais}}(x_{0:K}) = \gamma(x_0)/\pi_0(x_0)$ for the exact same proposal; i.e. the AIS estimate does not depend on the MCMC samples $x_{1:K}$ and boils down to the IS estimate of Z using the proposal π_0.

These two examples illustrate that it would be preferable to use $w_{\text{mar}}(x_K)$ rather than $w_{\text{ais}}(x_{0:K})$. In Appendix A, we provide a detailed comparison of both estimates in a scenario where their variance can be computed analytically. We propose in the next section an unbiased estimate of the evidence (MCD) approximating $w_{\text{mar}}(x_K)$ based on a different choice of backward kernels. As illustrated in Figure 2, significant gains can be achieved.

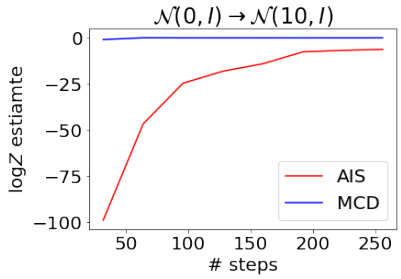


Figure 2: Comparing $\log Z$ estimates as a function of K using AIS and MCD. Both estimates use the same forward kernels but reweight samples in a different way using distinct backward kernels. Initial distribution π_0 is 20-dimensional $\mathcal{N}(0, I)$ and progressively shifts to the target $\pi = \pi_K = \mathcal{N}(10, I)$. The MCD estimate is much closer to the ground truth ($\log Z = 0$) than AIS.

3 Optimized Annealed Importance Sampling

We show here that the optimal extended target distribution P minimizing the variance of the evidence estimate (4) is defined through the time-reversal of the proposal Q. By exploiting a connection to SGM, we can approximate this reversal using score matching when the proposal is obtained through an unadjusted overdamped or underdamped Langevin algorithm.

3.1 Optimal Extended Target Distribution via Time Reversal

We summarize here Proposition 1 of [13]; see also [46, 5].

Proposition 1. *For a proposal Q of the form (2), the extended target P of the form (3) minimizing both the Kullback–Leibler divergence $D_{\mathrm{KL}}(Q||P)$ and the variance of the evidence estimate $w(x_{0:K}) = \Gamma(x_{0:K})/Q(x_{0:K})$ for $x_{0:K} \sim Q$ is given by $P^{\mathrm{opt}}(x_{0:K}) = \Gamma^{\mathrm{opt}}(x_{0:K})/Z$ where*

$$\Gamma^{\mathrm{opt}}(x_{0:K}) = \gamma(x_K) \prod_{k=0}^{K-1} B_k^{\mathrm{opt}}(x_k|x_{k+1}), \qquad B_k^{\mathrm{opt}}(x_k|x_{k+1}) = \frac{q_k(x_k)F_{k+1}(x_{k+1}|x_k)}{q_{k+1}(x_{k+1})}. \tag{6}$$

In particular, one has

$$w_{\mathrm{mar}}(x_K) = \frac{\gamma(x_K)}{q_K(x_K)} = \frac{\Gamma^{\mathrm{opt}}(x_{0:K})}{Q(x_{0:K})}, \quad \textit{and} \quad D_{\mathrm{KL}}(Q||P^{\mathrm{opt}}) = D_{\mathrm{KL}}(q_K||\pi). \tag{7}$$

This result follows simply from the chain rule and the law of total variance which yield

$$D_{\mathrm{KL}}(Q||P) = D_{\mathrm{KL}}(q_K||\pi) + \mathbb{E}_{q_K}\Big[D_{\mathrm{KL}}(Q(\cdot|x_K)||P(\cdot|x_K))\Big], \tag{8}$$

$$\mathrm{var}_Q[w(x_{0:K})] = \mathrm{var}_{q_K}[w_{\mathrm{mar}}(x_K)] + \mathbb{E}_{q_K}[\mathrm{var}_{Q(\cdot|x_K)}[w(x_{0:K})]]. \tag{9}$$

Both quantities are clearly minimized by selecting $P(x_{0:K-1}|x_K) = Q(x_{0:K-1}|x_K)$.

We emphasize that Proposition 1 applies to any forward kernels $(F_k)_{k=1}^K$ including MCMC kernels, unadjusted Langevin kernels or even deterministic maps[2]. It shows that P^{opt} is the distribution of a backward process initialized at π which then follows the time-reversed dynamics of the forward process Q. If we had $q_K = \pi$, then we would have $P^{\mathrm{opt}} = Q$ as then P^{opt} would correspond to the backward decomposition of Q.

3.2 Time reversal, Score matching and ELBO for unadjusted overdamped Langevin

We concentrate here on the case where $(F_k)_{k=1}^K$ correspond to a time-inhomogeneous unadjusted (overdamped) Langevin algorithm (ULA) as used in [23, 53, 51]; that is we consider $F_k(x_k|x_{k-1}) = \mathcal{N}(x_k; x_{k-1} + \delta\nabla \log \pi_k(x_{k-1}), 2\delta I)$ where $\delta > 0$ is a stepsize. Let $\delta := T/K$ then, as $K \to \infty$, the proposal Q converges to the path measure $\mathcal{Q}$ of the following inhomogeneous Langevin diffusion $(x_t)_{t\in[0,T]}$ defined by the stochastic differential equation (SDE)

$$\mathrm{d}x_t = \nabla \log \pi_t(x_t)\mathrm{d}t + \sqrt{2}\mathrm{d}B_t, \qquad x_0 \sim \pi_0, \tag{10}$$

where $(B_t)_{t\in[0,T]}$ is standard multivariate Brownian motion and we are slightly abusing notation from now on as π_t for $t = t_k = k\delta$ corresponds to π_k in discrete-time. Many quantitative results measuring the discrepancy between the law of x_T and $\pi_T = \pi$ for such annealed diffusions have been obtained; see e.g. [17, 50]. From [22] , it is known that the time-reversed process $(\bar{x}_t) = (x_{T-t})_{t\in[0,T]}$ is also a diffusion given by

$$\mathrm{d}\bar{x}_t = \big\{ -\nabla \log \pi_{T-t}(\bar{x}_t) + 2\nabla \log q_{T-t}(\bar{x}_t)\big\}\mathrm{d}t + \sqrt{2}\mathrm{d}\bar{B}_t, \qquad \bar{x}_0 \sim q_T, \tag{11}$$

where $(\bar{B}_t)_{t\in[0,T]}$ is another multivariate Brownian motion. The continuous-time version of P^{opt} is the path measure $\mathcal{P}^{\mathrm{opt}}$ defined by the diffusion (11) but initialized at $\bar{x}_0 \sim \pi$ rather than q_T as noted in [5]; see Figure 1 for an illustration. This shows that approximating $(B_k^{\mathrm{opt}})_{k=0}^{K-1}$ requires approximating the so-called scores $(\nabla \log q_t(x))_{t\in[0,T]}$. This can be derived heuristically through the fact that a Taylor expansion yields the following approximation of the optimal backward kernels, $B_k^{\mathrm{opt}}(x_k|x_{k+1}) \approx \mathcal{N}(x_k; x_{k+1} - \delta\nabla \log \pi_{k+1}(x_{k+1}) + 2\delta\nabla \log q_{k+1}(x_{k+1}), 2\delta I)$, which indeed corresponds to a Euler discretization of (11); see e.g. [12, Section 2.1].

In SGM [48], one gradually adds noise to data using an Ornstein–Ulhenbeck diffusion to transform the complex data distribution into a Gaussian distribution and the generative model is obtained by approximating the time-reversal of this diffusion initialized by Gaussian noise. Practically, the time-reversal approximation is obtained by estimating the scores of the noising diffusion using denoising

[2]The case of deterministic maps corresponds to normalizing flow components, where the inverse flow is the optimal and only valid reversal, see e.g. [2] which includes a detailed literature review.

score matching [52]. While in our setup, the diffusion (10) instead goes from a simple distribution to a complex one (see Appendix D for a discussion), we can still use score matching ideas. We define a path measure $\mathcal{P}_\theta$ approximating $\mathcal{P}^{\text{opt}}$ using a neural network $s_\theta(T-t, \bar{x}_t)$ in place of $\nabla \log q_{T-t}(\bar{x}_t)$ in (11), i.e. we consider

$$\mathrm{d}\bar{x}_t = \big\{ -\nabla \log \pi_{T-t}(\bar{x}_t) + 2s_\theta(T-t, \bar{x}_t)\big\}\mathrm{d}t + \sqrt{2}\mathrm{d}\bar{B}_t, \qquad \bar{x}_0 \sim \pi. \tag{12}$$

We would like to learn θ by minimizing $D_{\mathrm{KL}}(\mathcal{Q}||\mathcal{P}_\theta)$ over θ, i.e. equivalently we maximize a continuous-time ELBO. Note that it is neither easily feasible to minimize $D_{\mathrm{KL}}(\mathcal{P}^{\text{opt}}||\mathcal{P}_\theta)$ (as one cannot sample from π) nor it is desirable as the evidence estimate is computed using samples from $\mathcal{Q}$. Hence we want the scores to be well-approximated in regions of high-probability mass under $\mathcal{Q}$.

In practice, the diffusions corresponding to $\mathcal{Q}$ and $\mathcal{P}_\theta$ have to be discretized, so a more direct route adopted here is to simply take inspiration from (11) and to consider the parameterized backward kernels $B_k^\theta(x_k|x_{k+1}) = \mathcal{N}(x_k; x_{k+1} - \delta\nabla \log \pi_{k+1}(x_{k+1}) + 2\delta s_\theta(k+1, x_{k+1}), 2\delta I)$ to obtain a parameterized extended target P_θ and corresponding unnormalized target Γ_θ. We then learn θ by minimizing $D_{\mathrm{KL}}(Q||P_\theta)$ where

$$Q(x_{0:K}) = \pi_0(x_0)\prod_{k=0}^{K-1} F_{k+1}(x_{k+1}|x_k), \quad P_\theta(x_{0:K}) = \pi(x_K)\prod_{k=0}^{K-1} B_k^\theta(x_k|x_{k+1}).$$

This is obviously equivalent to maximizing the ELBO $\mathbb{E}_Q[\log w_\theta(x_{0:K})]$ where $w_\theta(x_{0:K}) = \Gamma_\theta(x_{0:K})/Q(x_{0:K})$. We note that it has previously been proposed to learn parameterized backward kernels for general AIS proposals [44, 26]. However, the parameterization adopted therein, $B_k^\theta(x_k|x_{k+1}) = \mathcal{N}(x_k; \mu_\theta(x_{k+1}), \Sigma_\theta(x_{k+1}))$, does not leverage the structure of the true reversal and performs poorly experimentally [51, Section 4.2].

As established in the next proposition, the continuous and discrete time approaches coincide for $\delta \ll 1$. Once θ is learned, we then obtain an unbiased estimate of Z through $w_\theta(x_{0:K})$ for $x_{0:K} \sim Q$.

Proposition 2. *Under regularity conditions, we have*

$$D_{\mathrm{KL}}(\mathcal{Q}||\mathcal{P}_\theta) = \mathbb{E}_{\mathcal{Q}}\Big[\int_0^T ||s_\theta(t, x_t) - \nabla \log q_t(x_t)||^2 \mathrm{d}t\Big] + C_1 \tag{13}$$

$$= \sum_{k=1}^{K}\int_{t_{k-1}}^{t_k} \mathbb{E}_{\mathcal{Q}}\left[||s_\theta(t, x_t) - \nabla \log q_{t|t_{k-1}}(x_t|x_{t_{k-1}})||^2\right] \mathrm{d}t + C_2, \tag{14}$$

where $t_k = k\delta$, $K = T/\delta$, $q_{t|s}(x'|x)$ is the density of $x_t = x'$ given $x_s = x$ under $\mathcal{Q}$ and C_1, C_2 constants independent of θ. Let $\mathcal{L}(\theta) = \delta\sum_{k=1}^{K}\mathbb{E}_Q\left[||s_\theta(k, x_k) - \nabla \log F_k(x_k|x_{k-1})||^2\right]$ denote a discrete-time approximation of this loss. We have $\nabla D_{\mathrm{KL}}(Q||P_\theta) = \nabla\mathcal{L}(\theta) + \epsilon(\theta)$ for some function ϵ satisfying $\lim_{K\to\infty}\epsilon(\theta) = 0$.

Equation (13) shows that $D_{\mathrm{KL}}(\mathcal{Q}||\mathcal{P}_\theta)$ corresponds to a score matching loss as for SGM [47]. It is possible to rewrite this loss as (14) so as to replace the intractable score term $\nabla \log q_t(x_t)$ by the easy to approximate gradients of the log-transitions $\nabla \log q_t(x_t|x_{t_{k-1}})$ [52]. In practice, as mentioned above, we simply learn θ by minimizing the discrete-time KL discrepancy $D_{\mathrm{KL}}(Q||P_\theta)$. This formulation is also very convenient as we can additionally learn potential parameters ϕ of a Q_ϕ using the same criterion. Note from equation (8) that the KL divergence decomposes as firstly a term penalizing the difference in the approximating measure and the fixed target at the final time and secondly another term which can be reduced by optimization of both Q_ϕ and P_θ conditioned on x_T; see e.g. [1].

Pseudo-code for our approach (with a comparison to the AIS algorithm proposed in [23, 53, 51]) can be found in Algorithm 1.

3.3 Incorporating Hamiltonian dynamics via the underdamped Langevin equation

We now consider a proposal defined on an extended space which arises from the time-discretization of a time-inhomogeneous underdamped Langevin dynamics; see e.g. [33, Chapter 6]. In this scenario, we first focus on continuous-time as the development of suitable numerical integrators is much more involved than for overdamped Langevin diffusions. We consider the diffusion $(x_t, p_t)_{t\in[0,T]}$ where $p_t \in \mathbb{R}^d$ is a momentum variable

$$\mathrm{d}x_t = M^{-1}p_t\mathrm{d}t, \qquad \mathrm{d}p_t = \nabla \log \pi_t(x_t)\mathrm{d}t - \zeta p_t\mathrm{d}t + \sqrt{2\zeta}M^{1/2}\mathrm{d}B_t \tag{15}$$

Algorithm 1 Unadjusted Langevin AIS/MCD – red instructions for AIS and blue for MCD

Require: Unnormalized target $\gamma(x)$, initial state proposal $\pi_0(x)$, number steps K, stepsize δ, annealing schedule $\{\beta_k\}_{k=0}^K$, score model $s_\theta(k, x)$
Sample $x_0 \sim \pi_0(x_0)$
Set $\log w = -\log \pi_0(x_0)$
for $k = 1$ to K **do**
  Define $\log \gamma_k(\cdot) = \beta_k \log \gamma(\cdot) + (1 - \beta_k) \log \pi_0(\cdot)$
  Define $F_k(x_k|x_{k-1}) = \mathcal{N}(x_k; x_{k-1} + \delta \nabla \log \gamma_k(x_{k-1}), 2\delta I)$
  Sample $x_k \sim F_k(\cdot|x_{k-1})$
  Define $B_{k-1}(x_{k-1}|x_k) = F_k(x_{k-1}|x_k)$ ▷ AIS
  Define $B_{k-1}(x_{k-1}|x_k) = \mathcal{N}(x_{k-1}; x_k - \delta \nabla \log \gamma_k(x_k) + 2\delta s_\theta(k, x_k), 2\delta I)$ ▷ MCD
  Set $\log w = \log w + \log B_{k-1}(x_{k-1}|x_k) - \log F_k(x_k|x_{k-1})$
end for
Set $\log w = \log w + \log \gamma(x_K)$

initialized at $x_0 \sim \pi_0, p_0 \sim \mathcal{N}(0, M)$ defining the path measure $\mathcal{Q}$. Here M is a positive definite mass matrix, $\zeta > 0$ a friction coefficient and $(B_t)_{t\in[0,T]}$ a multivariate Brownian motion. If π_t was not time-varying, e.g. $\pi_t = \pi$, the invariant distribution of this diffusion would be given by $\bar{\pi}(x, p) = \pi(x)\mathcal{N}(p; 0, M)$. Intuitively, in the time varying case the SDE will have enough time to approximate each intermediate $\bar{\pi}_t(x, p) = \pi_t(x)\mathcal{N}(p; 0, M)$ if we change the target sufficiently slowly. We can think of underdamped Langevin as a continuous-time version of Hamiltonian dynamics with continuous stochastic partial momentum refreshment [25].

From [22], the time-reversal of the diffusion (15) is also a diffusion process $(\bar{x}_t, \bar{p}_t)_{t\in[0,T]} = (x_{T-t}, p_{T-t})_{t\in[0,T]}$ given by $(\bar{x}_0, \bar{p}_0) \sim \eta_T$ and

$$\begin{aligned} \mathrm{d}\bar{x}_t &= -M^{-1}\bar{p}_t \mathrm{d}t, \\ \mathrm{d}\bar{p}_t &= -\nabla \log \pi_{T-t}(\bar{x}_t)\mathrm{d}t + \zeta \bar{p}_t \mathrm{d}t + 2\zeta M \nabla_{\bar{p}_t} \log \eta_{T-t}(\bar{x}_t, \bar{p}_t)\mathrm{d}t + \sqrt{2\zeta} M^{1/2} \mathrm{d}\bar{B}_t, \end{aligned} \tag{16}$$

where η_t denotes the density (x_t, p_t) under (15). In this case, the continuous-time version of P_{opt} is the path measure $\mathcal{P}_{\text{opt}}$ defined by the diffusion (16) but initialized at $\bar{x}_0 \sim \pi, \bar{p}_0 \sim \mathcal{N}(0, M)$ rather than η_T. We will approximate it by the path measure $\mathcal{P}_\theta$ using a neural network $s_\theta(T - t, \bar{x}_t, \bar{p}_t)$ in place of $\nabla \log \eta_{T-t}(\bar{x}_t, \bar{p}_t)$ in (16), i.e. we consider

$$\begin{aligned} \mathrm{d}\bar{x}_t &= -M^{-1}\bar{p}_t \mathrm{d}t, \\ \mathrm{d}\bar{p}_t &= -\nabla \log \pi_{T-t}(\bar{x}_t)\mathrm{d}t + \zeta \bar{p}_t \mathrm{d}t + 2\zeta M s_\theta(T - t, \bar{x}_t, \bar{p}_t)\mathrm{d}t + \sqrt{2\zeta} M^{1/2} \mathrm{d}\bar{B}_t. \end{aligned} \tag{17}$$

As for overdamped Langevin, we could also learn θ by minimizing $D_{\text{KL}}(\mathcal{Q}||\mathcal{P}_\theta)$ over θ. This again corresponds to minimizing a score matching loss albeit of a form slightly different from (13).

Proposition 3. *Under regularity conditions, we have*

$$\begin{aligned} D_{\text{KL}}(\mathcal{Q}||\mathcal{P}_\theta) &= \zeta \mathbb{E}_{\mathcal{Q}}\Big[\int_0^T ||s_\theta(t, x_t, p_t) - \nabla_{p_t} \log \eta_t(x_t, p_t)||^2 \mathrm{d}t\Big] + C_1 \\ &= \zeta \sum_{k=1}^K \int_{t_{k-1}}^{t_k} \mathbb{E}_{\mathcal{Q}}\left[||s_\theta(t, x_t, p_t) - \nabla_{p_t} \log \eta_{t|t_{k-1}}(x_t, p_t|x_{t_{k-1}}, p_{t_{k-1}})||_M^2\right] \mathrm{d}t + C_2, \end{aligned} \tag{18}$$

where $||x||_M := u^{\mathrm{T}} M u$, $t_k = k\delta$, $K = T/\delta$, C_1, C_2 *are constants independent of* θ *and* $\eta_{t|s}(x', p'|x, p)$ *is the density of* $(x_t, p_t) = (x', p')$ *given* $(x_s, p_s) = (x, p)$ *under* $\mathcal{Q}$.

While the continuous-time perspective shed light on how to parameterize an approximation to the time-reversal, this does not lead directly to an implementable discrete-time algorithm for underdamped Langevin. Contrary to overdamped Langevin, we cannot indeed simply use an Euler discretization of (15) defining $\mathcal{Q}$ and (17) defining $\mathcal{P}_\theta$ to obtain some discrete-time forward and backward kernels and then compute $w_\theta(x_{0:K}, p_{0:K}) = \Gamma_\theta(x_{0:K}, p_{0:K})/Q(x_{0:K}, p_{0:K})$. This is because this ratio is not well-defined due to the lack of noise on the position component in both (15) and (17).

The integrator we use for the forward equation (15), consists in alternating partial momentum refreshments and deterministic leapfrog steps (see e.g [33, 38]) giving

$$\tilde{p}_{t+\delta} \sim \mathcal{N}(hp_t, (1 - h^2)M), \qquad (x_{t+\delta}, p_{t+\delta}) = \Phi_t(x_t, \tilde{p}_{t+\delta}), \tag{19}$$

Algorithm 2 Unadjusted Hamiltonian AIS/MCD – red instructions for AIS and blue for MCD

Require: Unnormalized target $\gamma(x)$, initial state proposal $\pi_0(x)$, number steps K, stepsize η, annealing schedule $\{\beta_k\}_{k=0}^K$, damping coefficient h, mass matrix M, score model $s_\theta(k, x, p)$

Sample $x_0 \sim \pi_0(x_0)$ and $p_0 \sim \mathcal{N}(p_0; 0, M)$

Set $\log w = -\log \pi_0(x_0) - \log \mathcal{N}(p_0; 0, M)$

for $k = 1$ to K **do**

  Define $\log \gamma_k(\cdot) = \beta_k \log \gamma(\cdot) + (1 - \beta_k) \log \pi_0(\cdot)$

  Sample $\tilde{p}_k \sim \mathcal{N}(hp_{k-1}, (1 - h^2)M)$

  Set $\mu_q = p_{k-1}$

  Set $\mu_p = \tilde{p}_k$ ▷ UHA reversal mean

  Set $\mu_p = \tilde{p}_k - 2\log(h)[Ms_\theta(k, x_{k-1}, \tilde{p}_k) + \tilde{p}_k]$ ▷ MCD reversal mean

  Set $\log w = \log w + \log \mathcal{N}(p_{k-1}; h\mu_p, (1 - h)^2 M) - \log \mathcal{N}(\tilde{p}_k; h\mu_q, (1 - h)^2 M)$

  Run leapfrog integrator on γ_k and set $(x_k, p_k) = \Phi(x_{k-1}, \tilde{p}_k)$

end for

Set $\log w = \log w + \log \gamma(x_K) + \log \mathcal{N}(p_K; 0, M)$

with $h = \exp\{-\zeta\delta\}$ and Φ_t is the leapfrog integrator for π_t. The resulting forward sampler is similar to the one proposed by [18], except we do not flip the momentum after the leapfrog step[3]. This integrator may be interpreted as a splitting method for equation (15); see e.g [33, Chapter 7].

We need the integrator for the reversal to fulfill two criteria. First, by definition, as the time step $\delta \to 0$ it must recover the SDE (17). Second, the importance weight of the forward sampler to the reversal must be well defined. Since the leapfrog integrator is a diffeomorphism (or flow) the only possible way to get a well defined reversal for these steps is to take the inverse Φ_t^{-1}. As the transformation is also volume preserving, the contribution from the deterministic forward and reverse terms will then exactly cancel in the importance weight. The required form of the reverse integrator is

$$(x_t, \tilde{p}_{t+\delta}) = \Phi_t^{-1}(x_{t+\delta}, p_{t+\delta}), \qquad p_t \sim \mathcal{N}(hf_\theta(t + \delta, x_t, \tilde{p}_{t+\delta}), (1 - h^2)M), \tag{20}$$

where $f_\theta(t + \delta, x_t, \tilde{p}_{t+\delta}) := \widetilde{p}_{t+\delta} + \delta 2\zeta[Ms_\theta(t, x_t, \widetilde{p}_{t+\delta}) + \widetilde{p}_{t+\delta}]$. In Appendix C we show that as $\delta \to 0$ this can indeed be interpreted as a valid split integrator for the reverse SDE (17). The crucial point of algorithmic difference from [10, 18, 54] arises from our necessary form for the mean of the reverse momentum refreshment. These works use $h\tilde{p}_{t+\delta}$ as the mean instead of $hf_\theta(t + \delta, x_t, \tilde{p}_{t+\delta})$. We again transition to discretized notation with $\delta := T/K$, and $k = 0, ..., K$. In this case, the log importance weight, corresponding to the log evidence estimate, satisfies

$$\log w_\theta(\mathbf{x}, \mathbf{p}) = \log \frac{\gamma(x_K)\mathcal{N}(p_K; 0, M)}{\pi_0(x_0)\mathcal{N}(p_0; 0, M)} + \sum_{k=1}^{K} \log \frac{\mathcal{N}(p_{k-1}; hf_\theta(k, x_{k-1}, \tilde{p}_k), (1 - h^2)M)}{\mathcal{N}(\tilde{p}_k; hp_{k-1}, (1 - h^2)M)}, \tag{21}$$

where $(\mathbf{x}, \mathbf{p})$ denote all the variables introduced by our integration scheme. We can show informally that minimizing $D_{\mathrm{KL}}(Q||P_\theta)$, i.e. maximizing the ELBO given $\mathbb{E}_Q[\log w_\theta(\mathbf{x}, \mathbf{p})]$, again corresponds to minimizing a score matching type loss (18) when $\delta \ll 1$; see Appendix C. Pseudo-code for our approach can be found in Algorithm 2.

4 Experiments

We run a number of experiments estimating normalizing constants to validate our approach, MCD and compare to differentiable AIS with ULA [53, 51] and Unadjusted Hamiltonian Annealing (UHA) [18, 54]. We first investigate the performance of these approaches on static target distributions using the same, fixed initial distribution and annealing schedule. Finally, we explore the performance of the methods for VAEs. Here, being the most expensive of our experiments, we include runtime comparisons of our method compared to baselines. Additional results on a Normalizing Flow target can be found in Appendix F.2. Full experimental details, chosen hyper-parameters, and model architectures can be found in Appendix E.

Our score model is parameterized by an MLP with residual connections that is conditioned on integration time t, and on the momentum term for the Hamiltonian case (see Algorithm 2). For an ablation on various network architectures we refer the reader to Appendix F.3.

[3] [18] were not attempting to discretize an underdamped Langevin dynamics.

4.1 Static Targets

We estimate the normalizing constants of five simple distributions with known normalizing constants equal to $\log Z = 0$; $\mathcal{N}(10, I)$, $\mathcal{N}(0, 0.1I)$, a Gaussian mixture with 8 components whose means are drawn from $\mathcal{N}(3, I)$ where each component has variance 1, a standard Laplace distribution and a Student's T distribution with 3 degrees of freedom. We use $\mathcal{N}(0, I)$ as our initial distribution for all targets except for the Gaussian mixture and $\mathcal{N}(0, 0.1I)$ which both use $\mathcal{N}(0, 3^2 I)$. We run each method using $K \in \{64, 256\}$ steps and use a fixed, linear annealing schedule. For all methods, sampling step-sizes per-timestep are tuned to via gradient descent to maximize the ELBO and the diagonal mass matrix is learned for the Hamiltonian samplers. Gaussian Mixture and Student-T results can be found in Tables 1 and 2, respectively. Additional results can be found in Appendix F.1.

Sampler	ULA		UHA		ULA-MCD		UHA-MCD	
# steps	64	256	64	256	64	256	64	256
Dim-20	-0.47 ± 0.22	-0.13 ± 0.07	-0.03 ± 0.18	0.015 ± 0.03	0.01 ± 0.02	0.01 ± 0.01	0.01 ± 0.02	-0.01 ± 0.01
Dim-200	-85.62 ± 2.01	-21.98 ± 1.35	-8.20 ± 1.84	-1.26 ± 0.35	-0.25 ± 0.03	0.08 ± 0.122	0.20 ± 0.49	-0.05 ± 0.04
Dim-500	-304.06 ± 5.48	-83.50 ± 7.88	-44.45 ± 3.24	-8.60 ± 1.69	-2.61 ± 1.26	1.01 ± 0.99	-1.74 ± 1.25	-1.02 ± 0.03

Table 1: $\log Z$ estimates for a Gaussian mixture target. Averages and standard errors over 3 seeds.

On average and as expected, UHA outperforms ULA on most targets. Further our approximation to the optimal backward kernels can yield considerable improvements – enabling the ULA sampler to produce better results than UHA (using the standard AIS backward kernels). Additionally, we see that our approximation to the optimal backward kernels improves the performance of UHA as well.

We further emphasize that often ULA-MCD and UHA-MCD with 64 steps outperform or is on-par with ULA and UHA with 256 steps, a difference of factor 4 in terms of target gradient evaluations. As we show below in Table 3, the additional computational costs of fitting the score model of our method are only roughly twice that of the baselines.

Sampler	ULA		UHA		ULA-MCD		UHA-MCD	
# steps	64	256	64	256	64	256	64	256
Dim-20	-0.09 ± 0.02	-0.02 ± 0.01	-0.04 ± 0.02	-0.02 ± 0.01	-0.06 ± 0.02	-0.00 ± 0.01	-0.03 ± 0.04	-0.01 ± 0.01
Dim-200	-1.63 ± 0.23	-0.88 ± 0.15	-0.36 ± 0.34	-0.10 ± 0.10	-0.82 ± 0.28	-0.30 ± 0.38	-0.48 ± 0.20	-0.10 ± 0.07
Dim-500	-5.43 ± 0.78	-3.10 ± 0.07	-2.86 ± 0.36	-1.03 ± 0.13	-4.00 ± 0.51	-2.23 ± 0.18	-2.03 ± 0.18	0.06 ± 0.30

Table 2: $\log Z$ estimates for a Student-T target. Averages and standard errors over 3 seeds.

4.2 Application to Amortized Inference

Next, we explore the application of our method to amortized inference, in the context of VAEs [31, 41]. These models are trained to infer latent representations using an inference neural network that consumes an input and produces parameters of an approximation to the true posterior distribution of the underlying generative model. In particular, this posterior distribution is different for each input. When applying AIS to VAE inference [51, 18, 54], the output of the inference network parameterizes the initial distribution of the annealing sequence to the true posterior. By training this end-to-end, we effectively learn the initial distribution for the diffusion process. Consequently, the diffusion marginals and their score vectors $\nabla \log q_t$ are different for every input, and we need to condition our score model s_θ on the inputs to reflect that. We achieve this by projecting the last hidden layer of the inference network into a summary vector that is concatenated to the other conditioning inputs of s_θ.

We train a VAE on the binarized MNIST dataset [43], re-using architectures proposed in [18, 8] (two-layer MLP encoder/decoder, Bernoulli likelihood). All generative models use the same architecture

and hyper-parameters. We compare standard amortized variational inference with annealed ULA and UHA with standard AIS backward transition kernels, as well as ULA and UHA with our MCD transition kernels. We match the number of sampler steps between ULA/ULA-MCD and UHA/UHA-MCD to 64 and 32 respectively. ELBO and log-likelihood values on the test set are presented in Table 3.

Sampler	VI	ULA	UHA	ULA-MCD	UHA-MCD
ELBO	-96.32 $\pm$ 0.40	-90.41 $\pm$ 0.17	-88.58 $\pm$ 0.51	-90.10 $\pm$ 0.10	-88.08 $\pm$ 0.07
Log-lik.	-69.35 $\pm$ 0.36	-62.43 $\pm$ 0.25	-61.05 $\pm$ 1.84	-61.83 $\pm$ 0.42	-58.58 $\pm$ 0.34
Iteration time	0.024s	0.055s	0.050s	0.101s	0.098s
Total time	3263.40s	4694.52s	5072.19s	11304.36s	11345.59s

Table 3: Test set performance for MNIST VAE. Averages and standard errors over 5 seeds. We additionally report runtimes for a single training iteration, and total experiment time.

We see that, as reported in prior works [18, 51, 54], Monte Carlo based inference methods provide a significant benefit over standard amortized variational inference. In addition, our learned backward kernels lead to improved performance of both ULA and UHA, with the difference being more distinct for the latter. We also note that UHA has significantly larger error bars than all other methods, but UHA-MCD does not appear to inherit this. We further provide a runtime comparison in Table 3, both for a single training iteration, and total experiment time (including evaluation). We find that fitting the score models of our method results in roughly twice as much runtime (recall we fixed the number of sampler steps).

5 Limitations

While we have demonstrated that our optimized backward kernels can lead to large improvements over standard AIS backward kernels, our proposed approach has a number of limitations. First, since our method relies upon unadjusted Langevin or Hamiltonian sampling, we inherit many of the issues with these approaches. Using unadjusted samplers enables a fully-differentiable ELBO estimate which can in theory be used to tune the many parameters of the sampler. However, these samplers require repeated gradient steps as an inner-loop which can lead to divergence dynamics and numerical instability, making optimization difficult. Next, our proposed training procedure has a notable increase in memory consumption as we store the entire sampling trajectory in memory to train our neural network. We further note that it is possible to compute our importance weights online using $\mathcal{O}(1)$ memory. We could utilize this with Monte Carlo sub-sampling of the timestep (as with SGM [24]) to derive a constant memory training procedure for our approximate time-reversal at the cost of increased variance but we leave exploring this to future work. Finally, we observed that, without mitigation, parameterized forward samplers Q_ϕ sometimes dropped modes during training. Since this is a known challenge of using reverse KL, this might be improved by using alternative optimization objectives [35, 36].

6 Discussion

In this work we have explored AIS using unadjusted Langevin and Hamiltonian dynamics. We have demonstrated that the backward transition kernels typically used are suboptimal and we have presented the form of the optimal variance-minimizing backward kernels. We have further shown how an approximation to these kernels can be learned using score matching and that this objective corresponds to maximizing the ELBO in the limit of infinitesimally small time-discretization step-sizes. We have illustrated the benefit of using our proposed optimized backward kernels on a number of inference problems including fixed targets and amortized tasks with model learning.

References

[1] Agakov, F. V. and Barber, D. (2004). An auxiliary variational method. In *Advances in Neural Information Processing Systems*.

[2] Arbel, M., Matthews, A. G., and Doucet, A. (2021). Annealed flow transport Monte Carlo. In *International Conference on Machine Learning*.

[3] Ba, J. L., Kiros, J. R., and Hinton, G. E. (2016). Layer normalization. *arXiv preprint arXiv:1607.06450*.

[4] Babuschkin, I., Baumli, K., Bell, A., Bhupatiraju, S., Bruce, J., Buchlovsky, P., Budden, D., Cai, T., Clark, A., Danihelka, I., Fantacci, C., Godwin, J., Jones, C., Hennigan, T., Hessel, M., Kapturowski, S., Keck, T., Kemaev, I., King, M., Martens, L., Mikulik, V., Norman, T., Quan, J., Papamakarios, G., Ring, R., Ruiz, F., Sanchez, A., Schneider, R., Sezener, E., Spencer, S., Srinivasan, S., Stokowiec, W., and Viola, F. (2020). The DeepMind JAX Ecosystem.

[5] Bernton, E., Heng, J., Doucet, A., and Jacob, P. E. (2019). Schrödinger bridge samplers. *arXiv preprint arXiv:1912.13170*.

[6] Bradbury, J., Frostig, R., Hawkins, P., Johnson, M. J., Leary, C., Maclaurin, D., Necula, G., Paszke, A., VanderPlas, J., Wanderman-Milne, S., and Zhang, Q. (2018). JAX: composable transformations of Python+NumPy programs.

[7] Brekelmans, R., Huang, S., Ghassemi, M., Ver Steeg, G., Grosse, R. B., and Makhzani, A. (2022). Improving mutual information estimation with annealed and energy-based bounds. In *International Conference on Learning Representations*.

[8] Burda, Y., Grosse, R. B., and Salakhutdinov, R. (2016). Importance weighted autoencoders. In *International Conference on Learning Representations*.

[9] Crooks, G. E. (1998). Nonequilibrium measurements of free energy differences for microscopically reversible Markovian systems. *Journal of Statistical Physics*, 90(5-6):1481–1487.

[10] Dai, C., Heng, J., Jacob, P. E., and Whiteley, N. (2020). An invitation to sequential Monte Carlo samplers. *arXiv preprint arXiv:2007.11936*.

[11] De Bortoli, V., Doucet, A., Heng, J., and Thornton, J. (2021a). Simulating diffusion bridges with score matching. *arXiv preprint arXiv:2111.07243*.

[12] De Bortoli, V., Thornton, J., Heng, J., and Doucet, A. (2021b). Diffusion Schrödinger bridge with applications to score-based generative modeling. In *Advances in Neural Information Processing Systems*.

[13] Del Moral, P., Doucet, A., and Jasra, A. (2006). Sequential Monte Carlo samplers. *Journal of the Royal Statistical Society: Series* B, 68(3):411–436.

[14] Dinh, L., Krueger, D., and Bengio, Y. (2014). Nice: Non-linear independent components estimation. *arXiv preprint arXiv:1410.8516*.

[15] Dockhorn, T., Vahdat, A., and Kreis, K. (2022). Score-based generative modeling with critically-damped Langevin diffusion. In *International Conference on Learning Representations*.

[16] Doucet, A., Grathwohl, W. S., Matthews, A. G. D. G., and Strathmann, H. (2022). Annealed importance sampling meets score matching. In *ICLR Workshop on Deep Generative Models for Highly Structured Data*.

[17] Fournier, N. and Tardif, C. (2021). On the simulated annealing in $\mathbb{R}^d$. *Journal of Functional Analysis*, 281(5):109086.

[18] Geffner, T. and Domke, J. (2021). MCMC variational inference via uncorrected Hamiltonian annealing. In *Advances in Neural Information Processing Systems*.

[19] Gelman, A. and Meng, X.-L. (1998). Simulating normalizing constants: From importance sampling to bridge sampling to path sampling. *Statistical Science*, 13(2):163–185.

[20] Grillo, G. (1994). Logarithmic Sobolev inequalities and Langevin algorithms in $\mathbb{R}^n$. *Stochastic Analysis and Applications*, 12(3):309–328.

[21] Grosse, R. B., Maddison, C. J., and Salakhutdinov, R. R. (2013). Annealing between distributions by averaging moments. In *Advances in Neural Information Processing Systems*.

[22] Haussmann, U. G. and Pardoux, E. (1986). Time reversal of diffusions. *The Annals of Probability*, 14(3):1188–1205.

[23] Heng, J., Bishop, A. N., Deligiannidis, G., and Doucet, A. (2020). Controlled sequential Monte Carlo. *The Annals of Statistics*, 48(5):2904–2929.

[24] Ho, J., Jain, A., and Abbeel, P. (2020). Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*.

[25] Horowitz, A. M. (1991). A generalized guided Monte Carlo algorithm. *Physics Letters B*, 268(2):247–252.

[26] Huang, C.-W., Tan, S., Lacoste, A., and Courville, A. C. (2018). Improving explorability in variational inference with annealed variational objectives. In *Advances in Neural Information Processing Systems*.

[27] Hyvärinen, A. (2005). Estimation of non-normalized statistical models by score matching. *The Journal of Machine Learning Research*, 6:695–709.

[28] Jankowiak, M. and Phan, D. (2022). Surrogate likelihoods for variational annealed importance sampling. In *International Conference on Machine Learning*.

[29] Jarzynski, C. (1997). Nonequilibrium equality for free energy differences. *Physical Review Letters*, 78(14):2690–2963.

[30] Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

[31] Kingma, D. P. and Welling, M. (2013). Auto-encoding variational Bayes. *arXiv preprint arXiv:1312.6114*.

[32] Klebaner, F. C. (2012). *Introduction to Stochastic Calculus with Applications*. Imperial College Press.

[33] Leimkuhler, B. and Matthews, C. (2016). *Molecular Dynamics with Deterministic and Stochastic Numerical Methods*. Springer.

[34] Masrani, V., Brekelmans, R., Bui, T., Nielsen, F., Galstyan, A., Ver Steeg, G., and Wood, F. (2021). q-paths: Generalizing the geometric annealing path using power means. In *Uncertainty in Artificial Intelligence*.

[35] Masrani, V., Le, T. A., and Wood, F. (2019). The thermodynamic variational objective. In *Advances in Neural Information Processing Systems*.

[36] Matthews, A. G. D. G., Arbel, M., Rezende, D. J., and Doucet, A. (2022). Continual repeated annealed flow transport Monte Carlo. In *International Conference on Machine Learning*.

[37] Neal, R. M. (2001). Annealed importance sampling. *Statistics and Computing*, 11(2):125–139.

[38] Neal, R. M. (2011). MCMC using Hamiltonian dynamics. *Handbook of Markov Chain Monte Carlo*, 2(11):2.

[39] Opper, M. (2019). Variational inference for stochastic differential equations. *Annalen der Physik*, 531:1800233.

[40] Ramachandran, P., Zoph, B., and Le, Q. V. (2017). Searching for activation functions. *arXiv preprint arXiv:1710.05941*.

[41] Rezende, D. J., Mohamed, S., and Wierstra, D. (2014). Stochastic backpropagation and approximate inference in deep generative models. In *International Conference on Machine Learning*.

[42] Roberts, G. O. and Tweedie, R. L. (1996). Exponential convergence of Langevin distributions and their discrete approximations. *Bernoulli*, pages 341–363.

[43] Salakhutdinov, R. and Murray, I. (2008). On the quantitative analysis of deep belief networks. In *International Conference on Machine Learning*.

[44] Salimans, T., Kingma, D., and Welling, M. (2015). Markov chain Monte Carlo and variational inference: Bridging the gap. In *International Conference on Machine Learning*.

[45] Schmiedl, T. and Seifert, U. (2007). Optimal finite-time processes in stochastic thermodynamics. *Physical Review Letters*, 98(10):108301.

[46] Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N., and Ganguli, S. (2015). Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*.

[47] Song, Y., Durkan, C., Murray, I., and Ermon, S. (2021a). Maximum likelihood training of score-based diffusion models. In *Advances in Neural Information Processing Systems*.

[48] Song, Y., Sohl-Dickstein, J., Kingma, D. P., Kumar, A., Ermon, S., and Poole, B. (2021b). Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*.

[49] Sottinen, T. and Särkkä, S. (2008). Application of Girsanov theorem to particle filtering of discretely observed continuous-time non-linear systems. *Bayesian Analysis*, 3(3):555–584.

[50] Tang, W. and Zhou, X. Y. (2021). Simulated annealing from continuum to discretization: a convergence analysis via the Eyring–Kramers law. *arXiv preprint arXiv:2102.02339*.

[51] Thin, A., Kotelevskii, N., Durmus, A., Panov, M., Moulines, E., and Doucet, A. (2021). Monte Carlo variational auto-encoders. In *International Conference on Machine Learning*.

[52] Vincent, P. (2011). A connection between score matching and denoising autoencoders. *Neural Computation*, 23(7):1661–1674.

[53] Wu, H., Köhler, J., and Noé, F. (2020). Stochastic normalizing flows. In *Advances in Neural Information Processing Systems*.

[54] Zhang, G., Hsu, K., Li, J., Finn, C., and Grosse, R. (2021). Differentiable annealed importance sampling and the perils of gradient noise. In *Advances in Neural Information Processing Systems*.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] See Section 5
   (c) Did you discuss any potential negative societal impacts of your work? [N/A] Paper is methodological paper on approximate inference methods with only standard datasets.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes] See proofs of main results in Appendix.
   (b) Did you include complete proofs of all theoretical results? [Yes] See Appendix.
3. If you ran experiments...

(a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] See Appendix for architectures, Algorithm 2 for pseudo-code. We used standard datasets from the literature. We did not publish code.

(b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See experimental section and Appendix.

(c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] Where appropriate and feasible.

(d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...

(a) If your work uses existing assets, did you cite the creators? [Yes]

(b) Did you mention the license of the assets? [Yes]

(c) Did you include any new assets either in the supplemental material or as a URL? [No]

(d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]

(e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]

5. If you used crowdsourcing or conducted research with human subjects...

(a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]

(b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]

(c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]

A Variance calculations

We provide here, for a simple example, the variance expressions for $\log w_{\text{ais}}(x_{0:K})$ and $\log w_{\text{mar}}(x_K)$. All the expectations in this section are w.r.t. the proposal Q.

We consider the scenario where $\pi_0(x) = \gamma_0(x) = \mathcal{N}(x; 0, \sigma_0^2)$ and for $k \geq 1$

$$\pi_k(x) = \mathcal{N}(x; 0, \sigma_k^2), \quad \gamma_k(x) = \exp\left(-\frac{x^2}{2\sigma_k^2}\right).$$

We select the sequence of variances as follows

$$\sigma_k^2 = \left(\sigma_0^2\right)^{1-\frac{k}{K}} \left(\sigma^2\right)^{\frac{k}{K}} = \left(\frac{\sigma^2}{\sigma_0^2}\right)^{1/K} \sigma_{k-1}^2 := \beta_K \sigma_{k-1}^2$$

so that $\pi(x) = \pi_K(x) = \mathcal{N}(x; 0, \sigma^2)$. We will pick $\sigma^2 < \sigma_0^2$ so $\beta_K < 1$. Finally we consider the following proposal Q. At initialization $x_0 \sim \pi_0$ and for $k \geq 1$ we have the following transitions kernels

$$F_k(x'|x) = \mathcal{N}(x'; \alpha x, (1-\alpha^2)\sigma_k^2)$$

which are π_k-invariant and α determines how fast it mixes. For $\alpha = 0$, we have exact samples from π_k and as $\alpha \to 1$ we are mixing less and less. For this setup, it is possible to provide exact calculations for the expectation and variance of the log-evidence estimate.

Proposition 4. *Under Q, we have $x_k \sim \mathcal{N}(0, \xi_k^2)$ where $\xi_0^2 = \sigma_0^2$ and*

$$\xi_k^2 = \alpha^2 \xi_{k-1}^2 + (1-\alpha^2)\sigma_k^2.$$

The expectation of the log evidence estimates satisfies

$$\mathbb{E}[\log w_{\text{mar}}(x_K)] = \frac{1}{2}\log\left(2\pi\xi_K^2\right) + \frac{1}{2}\left(\frac{1}{\xi_K^2} - \frac{1}{\sigma_K^2}\right)\xi_K^2,$$

$$\mathbb{E}[\log w_{\text{ais}}(x_{0:K})] = \frac{1}{2}\log(2\pi\sigma_0^2) + \frac{(\beta_K - 1)}{2}\sum_{k=1}^{K}\frac{\xi_{k-1}^2}{\sigma_k^2},$$

while their variance is given by

$$\text{var}[\log w_{\text{mar}}(x_K)] = \frac{1}{2}\left(\frac{1}{\xi_K^2} - \frac{1}{\sigma_K^2}\right)^2 \xi_K^4,$$

$$\text{var}[\log w_{\text{ais}}(x_{0:K})] = (\beta_K - 1)^2\left(\sum_{k=1}^{K}\frac{\xi_{k-1}^4}{2\sigma_k^4} + \sum_{K \geq l > k \geq 1}\frac{\alpha^{2(l-k)}\xi_{k-1}^4}{\sigma_k^2\sigma_l^2}\right).$$

Proof. The proposal is given by $x_0 \sim \mathcal{N}(0, \sigma_0^2)$ and

$$x_k = \alpha x_{k-1} + \sqrt{1-\alpha^2}\sigma_k \epsilon_k, \qquad \epsilon_k \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, 1).$$

So marginally, under Q, we have $x_k \sim \mathcal{N}(0, \xi_k^2)$ where $(\xi_k)_{k \geq 0}$ satisfies the recursion stated above.

We are first looking at the optimal log-estimate of Z which is given by

$$\begin{aligned}\log w_{\text{mar}}(x_K) &= \log \gamma_K(x_K) - \log q_K(x_K) \\ &= \frac{1}{2}\log\left(2\pi\xi_K^2\right) + \frac{1}{2}\left(\frac{1}{\xi_K^2} - \frac{1}{\sigma_K^2}\right)x_K^2\end{aligned}$$

so we have

$$\mathbb{E}[\log w_{\text{mar}}(x_K)] = \frac{1}{2}\log\left(2\pi\xi_K^2\right) + \frac{1}{2}\left(\frac{1}{\xi_K^2} - \frac{1}{\sigma_K^2}\right)\xi_K^2$$

and, using $\text{var}[x^2] = 2\sigma^4$ for $x \sim \mathcal{N}(0, \sigma^2)$, we obtain

$$\text{var}[\log w_{\text{mar}}(x_K)] = \frac{1}{2}\left(\frac{1}{\xi_K^2} - \frac{1}{\sigma_K^2}\right)^2 \xi_K^4.$$

We are now looking at the AIS log-estimate of Z which is given by

$$\begin{aligned}\log w_{\text{ais}}(x_{0:K}) &= \sum_{k=1}^{K} \log(\gamma_k(x_{k-1})) - \log(\gamma_{k-1}(x_{k-1})) \\ &= \frac{1}{2}\log(2\pi\sigma_0^2) + \frac{(\beta_K - 1)}{2}\sum_{k=1}^{K} \frac{x_{k-1}^2}{\sigma_k^2},\end{aligned}$$

the term $\frac{1}{2}\log(2\pi\sigma_0^2)$ coming from the fact that we consider $\gamma_0 = \pi_0$. It follows that

$$\mathbb{E}[\log w_{\text{ais}}(x_{0:K})] = \frac{1}{2}\log(2\pi\sigma_0^2) + \frac{(\beta_K - 1)}{2}\sum_{k=1}^{K} \frac{\xi_{k-1}^2}{\sigma_k^2}.$$

The variance is given by

$$\begin{aligned}\text{var}[\log w_{\text{ais}}(x_{0:K})] &= \frac{(\beta_K - 1)^2}{4}\text{var}\left[\sum_{k=1}^{K} \frac{x_{k-1}^2}{\sigma_k^2}\right] \\ &= \frac{(\beta_K - 1)^2}{4}\sum_{k=1}^{K} \frac{1}{\sigma_k^4}\text{var}\left[x_{k-1}^2\right] + 2\frac{(\beta_K - 1)^2}{4}\sum_{K\geq l>k\geq 1}^{K} \frac{1}{\sigma_k^2\sigma_l^2}\text{cov}\left[x_{k-1}^2, x_{l-1}^2\right].\end{aligned}$$

Now, using again $\text{var}[x^2] = 2\sigma^4$ for $x \sim \mathcal{N}(0, \sigma^2)$, we have $\text{var}\left[x_{k-1}^2\right] = 2\xi_{k-1}^4$. To compute $\text{cov}\left[x_{k-1}^2, x_{l-1}^2\right]$, we use the fact that one can easily check from the form of the forward transitions that (x_{k-1}, x_{l-1}) satisfies for $l > k$

$$\text{cov}(x_{k-1}, x_{l-1}) = \alpha^{l-k}\xi_{k-1}^2.$$

So we have in distribution for $z \sim \mathcal{N}(0, 1)$ independent of x_{k-1}

$$(x_{k-1}, x_{l-1}) = (x_{k-1}, \alpha^{l-k}x_{k-1} + \sqrt{\xi_{l-1}^2 - \alpha^{2(l-k)}\xi_{k-1}^2}z).$$

Thus it follows that

$$\begin{aligned}x_{k-1}^2 x_{l-1}^2 &= x_{k-1}^2\left(\alpha^{l-k}x_{k-1} + \sqrt{\xi_{l-1}^2 - \alpha^{2(l-k)}\xi_{k-1}^2}z\right)^2 \\ &= x_{k-1}^2\left(\alpha^{2(l-k)}x_{k-1}^2 + (\xi_{l-1}^2 - \alpha^{2(l-k)}\xi_{k-1}^2)z^2 + 2\alpha^{l-k}\sqrt{\xi_{l-1}^2 - \alpha^{2(l-k)}\xi_{k-1}^2}x_{k-1}z\right) \\ &= \alpha^{2(l-k)}x_{k-1}^4 + (\xi_{l-1}^2 - \alpha^{2(l-k)}\xi_{k-1}^2)z^2 x_{k-1}^2 + 2\alpha^{l-k}\sqrt{\xi_{l-1}^2 - \alpha^{2(l-k)}\xi_{k-1}^2}x_{k-1}^3 z.\end{aligned}$$

Hence, we obtain

$$\begin{aligned}\text{cov}\left[x_{k-1}^2, x_{l-1}^2\right] &= \mathbb{E}\left[x_{k-1}^2 x_{l-1}^2\right] - \mathbb{E}\left[x_{k-1}^2\right]\mathbb{E}\left[x_{l-1}^2\right] \\ &= 3\alpha^{2(l-k)}\xi_{k-1}^4 + (\xi_{l-1}^2 - \alpha^{2(l-k)}\xi_{k-1}^2)\xi_{k-1}^2 - \xi_{k-1}^2\xi_{l-1}^2 \\ &= 2\alpha^{2(l-k)}\xi_{k-1}^4.\end{aligned}$$

This finally yields

$$\begin{aligned}\text{var}[\log w_{\text{ais}}(x_{0:K})] &= \frac{(\beta_K - 1)^2}{4}\sum_{k=1}^{K} \frac{1}{\sigma_k^4}\text{var}\left[x_{k-1}^2\right] + 2\frac{(\beta_K - 1)^2}{4}\sum_{K\geq l>k\geq 1}^{K} \frac{1}{\sigma_k^2\sigma_l^2}\text{cov}\left[x_{k-1}^2, x_{l-1}^2\right] \\ &= \frac{(\beta_K - 1)^2}{2}\sum_{k=1}^{K} \frac{\xi_{k-1}^4}{\sigma_k^4} + (\beta_K - 1)^2\sum_{K\geq l>k\geq 1}^{K} \frac{\alpha^{2(l-k)}\xi_{k-1}^4}{\sigma_k^2\sigma_l^2},\end{aligned}$$

as required. □

We now illustrate these results in Figure 3 by plotting the expectation and the root mean square error of $\log w_{\text{mar}}$ and $\log w_{\text{ais}}$ for various α and K.

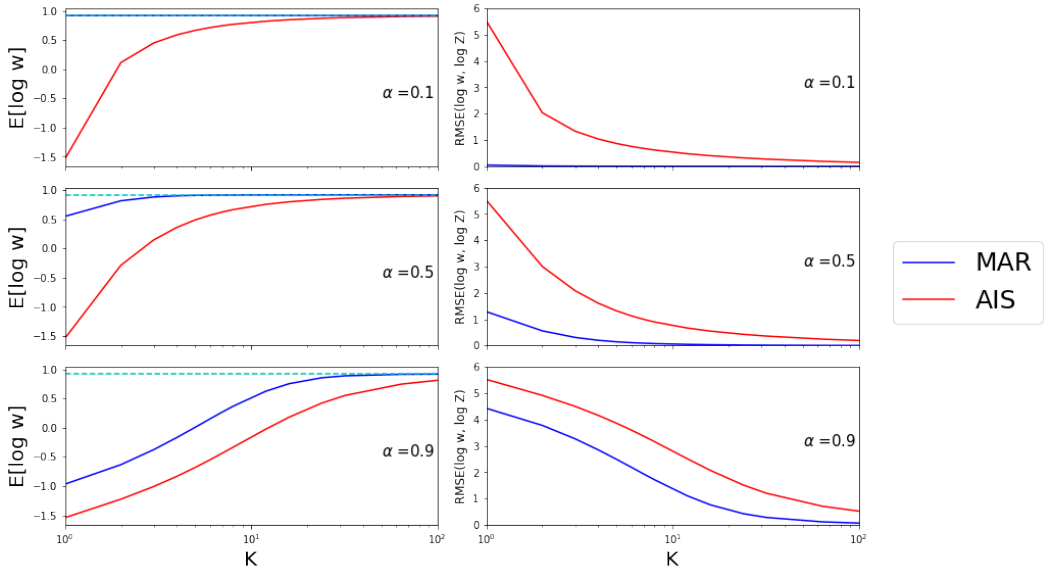


Figure 3: Plot of analytic results from Section A. The left column shows the analytic mean $\mathbb{E}[\log w]$ as a function of the number of temperature transitions K for different values of the mixing parameter $\alpha \in [0, 1]$. $\alpha = 0$ corresponds to perfect mixing while $\alpha = 1$ corresponds to no mixing. MAR denotes the optimal importance weight, where AIS shows the one from Annealed Importance Sampling. Both estimators tend to the correct value $\log Z$ (shown as a Cyan line) as K becomes large but MAR does so faster. The right column shows the same plots but for the root mean squared error (RMSE) of $\log w$ treated as an estimator of $\log Z$. It is computed as the root of the bias squared plus the variance, i.e $\sqrt{(\mathbb{E}[\log w] - \log Z)^2 + \mathrm{Var}[\log w]}$. The RMSE of both estimators tends to zero in both cases as K becomes large but MAR does so faster.

B Proof of Propositions

B.1 Proof of Proposition 1

Proof. The chain rule for the Kullback–Leibler divergence $D_{\mathrm{KL}}(Q||P)$ yields

$$D_{\mathrm{KL}}(Q||P) = D_{\mathrm{KL}}(q_K||\pi) + \mathbb{E}_{q_K}\Big[D_{\mathrm{KL}}(Q(\cdot|x_K)||P(\cdot|x_K))\Big], \tag{22}$$

where, from (2) and (3), the conditional distributions of $x_{0:K-1}$ given x_K are equal to

$$Q(x_{0:K-1}|x_K) = \prod_{k=0}^{K-1} B_k^{\mathrm{opt}}(x_k|x_{k+1}), \qquad P(x_{0:K-1}|x_K) = \prod_{k=0}^{K-1} B_k(x_k|x_{k+1}), \tag{23}$$

The expression for Q above follows directly from its time-reversed decomposition; i.e.

$$Q(x_{0:K}) = q_K(x_K) \prod_{k=0}^{K-1} \frac{q_k(x_k)F_{k+1}(x_{k+1}|x_k)}{q_{k+1}(x_{k+1})} = q_K(x_K) \prod_{k=0}^{K-1} B_k^{\mathrm{opt}}(x_k|x_{k+1}), \tag{24}$$

where we recall that $q_0(x_0) = \pi_0(x_0)$. It thus follows directly from (22) and (23) that the backward transition kernels $(B_k)_{k=0}^{K-1}$ minimizing $D_{\mathrm{KL}}(Q||P)$ are $(B_k^{\mathrm{opt}})_{k=0}^{K-1}$ as this implies $P(x_{0:K-1}|x_K) = Q(x_{0:K-1}|x_K)$.

The variance decomposition formula yields for all P

$$\begin{aligned}
\mathrm{var}_Q[w(x_{0:K})] &= \mathrm{var}_{q_K}[\mathbb{E}_{Q(\cdot|x_K)}[w(x_{0:K})]] + \mathbb{E}_{q_K}[\mathrm{var}_{Q(\cdot|x_K)}[w(x_{0:K})]] \\
&= \mathrm{var}_{q_K}[w_{\mathrm{mar}}(x_K)] + \mathbb{E}_{q_K}[\mathrm{var}_{Q(\cdot|x_K)}[w(x_{0:K})]] \\
&\geq \mathrm{var}_{q_K}[w_{\mathrm{mar}}(x_K)].
\end{aligned}$$

By direct calculations, we also have $w_{\mathrm{mar}}(x_K) = \Gamma^{\mathrm{opt}}(x_{0:K})/Q(x_{0:K})$ so P^{opt} minimizes the variance of the evidence estimate. □

B.2 Proof of Proposition 2

We establish first here Proposition 5 and Proposition 6. Both results can then be easily combined to obtain Proposition 2.

Proposition 5. *Under regularity conditions, one has*

$$D_{\mathrm{KL}}(\mathcal{Q}||\mathcal{P}_\theta) = \mathbb{E}_{\mathcal{Q}}\Big[\int_0^T ||s_\theta(t,x_t) - \nabla \log q_t(x_t)||^2 \mathrm{d}t\Big] + C_1 \tag{25}$$

$$= \sum_{k=1}^K \int_{t_{k-1}}^{t_k} \mathbb{E}_{\mathcal{Q}}\left[||s_\theta(t,x_t) - \nabla \log q_{t|t_{k-1}}(x_t|x_{t_{k-1}})||^2\right] \mathrm{d}t + C_2, \tag{26}$$

for constants C_1, C_2 independent of θ, where $t_k = k\delta$ with $K = T/\delta$ and $q_{t|s}(x'|x)$ is the density of $X_t = x'$ given $X_s = x$ under $\mathcal{Q}$.

To establish (25), we follow arguments similar to [47, Theorem 2]. The loss (26) we then consider differs from the one uses in the score-based generative modeling literature. This is because, contrary to the Ornstein–Ulhenbeck process used for SGM, the transition density $q_{t'|t}(x'|x)$ of the forward diffusion (10) is not available in closed-form and can only be approximated reliably when $t' - t$ is small. Practically, to obtain a tractable criterion, we need to first approximate the integrals in (26) by the rectangular rule. We also discretize the Langevin dynamics using an Euler–Maruyama scheme; i.e. we use an approximation Q of $\mathcal{Q}$ based on the ULA kernel $F_k(x_k|x_{k-1}) = \mathcal{N}(x_k; x_{k-1} + \delta\nabla \log \pi_k(x_{k-1}); 2\delta I)$ approximating $q_{t_k|t_{k-1}}(x'|x)$. We recall here that we slightly abuse notation by writing $\pi_{t_k} = \pi_{k\delta}$ as π_k. We thus finally obtain a loss

$$\mathcal{L}(\theta) = \delta \sum_{k=1}^K \mathbb{E}_Q\left[||s_\theta(k,x_k) - \nabla \log F_k(x_k|x_{k-1})||^2\right]. \tag{27}$$

Proof. We assume here sufficient regularity conditions ensuring that the SDEs given below admit a unique solution, their corresponding time-reversed diffusions are well-defined and Girsanov theorem applies; see e.g. [47, Appendix A].

By the chain rule for KL divergence, one has

$$D_{\mathrm{KL}}(\mathcal{Q}||\mathcal{P}_\theta) = D_{\mathrm{KL}}(q_T||\pi) + \mathbb{E}_{q_T}\Big[D_{\mathrm{KL}}(\mathcal{Q}(\cdot|x_T)||\mathcal{P}_\theta(\cdot|x_T))\Big] \tag{28}$$

where $\mathcal{Q}(\cdot|x_T)$ and $\mathcal{P}_\theta(\cdot|x_T)$ are the path measures induced by

$$\mathrm{d}\bar{x}_t = \big\{-\nabla \log \pi_{T-t}(\bar{x}_t) + 2\nabla \log q_{T-t}(\bar{x}_t)\big\}\mathrm{d}t + \sqrt{2}\mathrm{d}\bar{B}_t, \qquad \bar{x}_0 = x_T, \tag{29}$$

and

$$\mathrm{d}\bar{x}_t = \big\{-\nabla \log \pi_{T-t}(\bar{x}_t) + 2\nabla \log s_\theta(T-t,\bar{x}_t)\big\}\mathrm{d}t + \sqrt{2}\mathrm{d}\bar{B}_t, \qquad \bar{x}_0 = x_T. \tag{30}$$

We now use Girsanov theorem (see e.g. [32, Section 10.3] and [39]) to compute the Radon–Nikodym derivative $\mathrm{d}\mathcal{Q}(\cdot|x_T)/\mathrm{d}\mathcal{P}_\theta(\cdot|x_T)$ so that

$$\begin{aligned}
&\mathbb{E}_{q_T}\Big[D_{\mathrm{KL}}(\mathcal{Q}(\cdot|x_T)||\mathcal{P}_\theta(\cdot|x_T))\Big] \\
=& -\mathbb{E}_{\mathcal{Q}}\Big[\log \frac{\mathrm{d}\mathcal{P}_\theta(\cdot|x_T)}{\mathrm{d}\mathcal{Q}(\cdot|x_T)}\Big] \\
=&\mathbb{E}_{\mathcal{Q}}\Big[\sqrt{2}\int_0^T (\nabla \log q_{T-t}(\bar{x}_t) - s_\theta(T-t,\bar{x}_t))\mathrm{d}\bar{B}_t + \int_0^T ||\nabla \log q_{T-t}(\bar{x}_t) - s_\theta(T-t,\bar{x}_t)||^2 \mathrm{d}t\Big] \\
=&\mathbb{E}_{\mathcal{Q}}\Big[\int_0^T ||\nabla \log q_t(x_t) - s_\theta(t,x_t)||^2 \mathrm{d}t\Big],
\end{aligned}$$

as $\mathbb{E}_{\mathcal{Q}}\Big[\int_0^T f_t(\bar{x}_t)\mathrm{d}\bar{B}_t\Big] = 0$ for any function f_t.

As in [11] in a different context, we can write for any partition of $[0,T]$ defined by $t_0 = 0 < t_1 < \cdots < t_{K-1} < t_K = T$

$$\begin{aligned}\mathbb{E}_{\mathcal{Q}}\Big[\int_0^T ||\nabla \log q_t(x_t) - s_\theta(t,x_t)||^2 \mathrm{d}t\Big] &= \int_0^T \int ||\nabla \log q_t(x) - s_\theta(t,x)||^2 q_t(x)\mathrm{d}x\mathrm{d}t \\ &= \sum_{k=1}^K \int_{t_{k-1}}^{t_k} \int ||\nabla \log q_t(x) - s_\theta(t,x)||^2 q_t(x)\mathrm{d}x\mathrm{d}t\end{aligned}$$

where, for a constant c independent of θ, we have

$$\begin{aligned}&\int_{t_{k-1}}^{t_k} \int ||\nabla \log q_t(x) - s_\theta(t,x)||^2 q_t(x)\mathrm{d}x\mathrm{d}t \\ =&\int_{t_{k-1}}^{t_k} \int \left\{||\nabla \log q_t(x)||^2 + ||s_\theta(t,x)||^2 - 2s_\theta(t,x)^{\mathrm{T}}\nabla \log\ q_t(x)\right\} q_t(x)\mathrm{d}x\mathrm{d}t \\ =&\int_{t_{k-1}}^{t_k} \int \left\{||s_\theta(t,x)||^2 - 2s_\theta(t,x)^{\mathrm{T}}\nabla \log\ q_t(x)\right\} q_t(x)\mathrm{d}x\mathrm{d}t + c.\end{aligned}$$

Now we have

$$\int_{t_{k-1}}^{t_k} \int s_\theta(t,x)^{\mathrm{T}}\nabla \log q_t(x) q_t(x)\mathrm{d}x\mathrm{d}t = \int_{t_{k-1}}^{t_k} \int s_\theta(t,x)^{\mathrm{T}}\nabla q_t(x)\mathrm{d}x\mathrm{d}t \tag{31}$$

where, using Chapman-Kolmogorov, q_t satisfies

$$q_t(x) = \int q_{t_{k-1}}(x_{t_{k-1}}) q_{t|t_{k-1}}(x|x_{t_{k-1}})\mathrm{d}x_{t_{k-1}}. \tag{32}$$

It follows that

$$\nabla q_t(x) = \int q_{t_{k-1}}(x_{t_{k-1}}) \nabla q_{t|t_{k-1}}(x|x_{t_{k-1}})\mathrm{d}x_{t_{k-1}}. \tag{33}$$

Hence, we have

$$\begin{aligned}&\int_{t_{k-1}}^{t_k} \int s_\theta(t,x)^{\mathrm{T}}\nabla q_t(x)\mathrm{d}x\mathrm{d}t \\ =&\int_{t_{k-1}}^{t_k} \int\int s_\theta(t,x)^{\mathrm{T}}\nabla \log q_{t|t_{k-1}}(x|x_{t_{k-1}}) q_{t_{k-1}}(x_{t_{k-1}}) q_{t|t_{k-1}}(x|x_{t_{k-1}})\mathrm{d}x_{t_{k-1}}\mathrm{d}x\mathrm{d}t\end{aligned}$$

so minimizing $\mathbb{E}_{q_T}\Big[D_{\mathrm{KL}}(\mathcal{Q}(\cdot|x_T)||\mathcal{P}_\theta(\cdot|x_T))\Big]$ w.r.t. θ is equivalent to minimize

$$\begin{aligned}&\sum_{k=1}^K \int_{t_{k-1}}^{t_k} \int\int ||s_\theta(t,x)||^2 q_{t_{k-1}}(x_{t_{k-1}}) q_{t|t_{k-1}}(x|x_{t_{k-1}})\mathrm{d}x_{t_{k-1}}\mathrm{d}x\mathrm{d}t \\ -2&\sum_{k=1}^K \int_{t_{k-1}}^{t_k} \int\int s_\theta(t,x)^{\mathrm{T}}\nabla \log q_{t|t_{k-1}}(x|x_{t_{k-1}}) q_{t_{k-1}}(x_{t_{k-1}}) q_{t|t_{k-1}}(x|x_{t_{k-1}})\mathrm{d}x_{t_{k-1}}\mathrm{d}x\mathrm{d}t \\ =&\sum_{k=1}^K \int_{t_{k-1}}^{t_k} \int\int ||s_\theta(t,x) - \nabla \log q_{t|t_{k-1}}(x|x_{t_{k-1}})||^2 q_{t_{k-1}}(x_{t_{k-1}}) q_{t|t_{k-1}}(x|x_{t_{k-1}})\mathrm{d}x_{t_{k-1}}\mathrm{d}x\mathrm{d}t + C\end{aligned}$$

where C is independent of θ. Hence, this is equivalent to minimizing (26). □

We now establish results about the discrete-time Kullback–Leibler divergence $D_{\mathrm{KL}}(Q||P_\theta)$. First note that

$$\begin{aligned} D_{\mathrm{KL}}(Q||P_\theta) &= \mathbb{E}_Q\Big[\log \frac{Q(x_{0:K})}{P_\theta(x_{0:K})}\Big] \\ &= \mathbb{E}_Q\left[\log \frac{\pi_0(x_0)\prod_{k=0}^{K-1} F_{k+1}(x_{k+1}|x_k)}{\pi(x_K)\prod_{k=0}^{K-1} B_k^\theta(x_k|x_{k+1})}\right] \\ &= -\sum_{k=0}^{K-1} \mathbb{E}_Q\big[\log B_k^\theta(x_k|x_{k+1})\big] + C_1, \end{aligned} \tag{34}$$

where, as $B_k^\theta(x_k|x_{k+1}) = \mathcal{N}(x_k; x_{k+1} - \delta\nabla \log \pi_{k+1}(x_{k+1}) + 2\delta s_\theta(k+1, x_{k+1}), 2\delta I)$, one has

$$\begin{aligned} -\log B_k^\theta(x_k|x_{k+1}) &= \frac{1}{4\delta}||x_k - x_{k+1} + \delta\nabla \log \pi_{k+1}(x_{k+1}) - 2\delta s_\theta(k+1, x_{k+1})||^2 + C_2 \\ &= \delta\Big\|s_\theta(k+1, x_{k+1}) - \frac{1}{2\delta}(x_k - x_{k+1} + \delta\nabla \log \pi_{k+1}(x_{k+1}))\Big\|^2 + C_2 \quad (35) \\ &\approx \delta\Big\|s_\theta(k+1, x_{k+1}) - \frac{1}{2\delta}(x_k - x_{k+1} + \delta\nabla \log \pi_{k+1}(x_k))\Big\|^2 + C_2 \\ &= \delta\Big\|s_\theta(k+1, x_{k+1}) - \nabla \log F_{k+1}(x_{k+1}|x_k)\Big\|^2 + C_2, \quad (36) \end{aligned}$$

where we have used $\pi_{k+1}(x_{k+1}) \approx \pi_{k+1}(x_k)$ for $\delta \ll 1$. The sum over $k = 0, ..., K-1$ of the first terms on the r.h.s. of (35) are equal to the loss $\mathcal{L}(\theta)$ defined in (27). More rigorously, we can prove the following result.

Assumption 1. *There exists $L \leq \infty$ such that for all k and $x, x' \in \mathbb{R}^d$*

$$\Big\|\nabla \log \pi_k(x) - \nabla \log \pi_k(x')\Big\| \leq L\Big\|x - x'\Big\|. \tag{37}$$

Assumption 2. *There exists $C \leq \infty$ such that*

$$\limsup_K \max_{k=0,...,K-1} \mathbb{E}_{Q_K}\left[\Big\|\nabla \log \pi_{k+1}(x_k)\Big\|^2\right] \leq C \tag{38}$$

and for any θ

$$\limsup_K \max_{k=0,...,K-1} \mathbb{E}_{Q_K}\left[\Big\|\nabla_\theta s_\theta(k+1, x_{k+1})\Big\|^2\right] \leq C, \tag{39}$$

where we have emphasized here notationally that Q is a function of K.

Proposition 6. *Under Assumptions 1-2, the gradient of the Kullback–Leibler divergence $D_{\mathrm{KL}}(Q||P_\theta)$ satisfies*

$$\nabla D_{\mathrm{KL}}(Q||P_\theta) = \nabla\mathcal{L}(\theta) + \epsilon(\theta), \tag{40}$$

for $\mathcal{L}(\theta)$ defined in (27) and a function ϵ satisfying $\lim_{K\to\infty} \epsilon(\theta) = 0$.

Proof. In the rest of the proof, all the expectations are taken w.r.t. Q unless mentioned otherwise and we drop it from the notations for simplicity. However as we take gradients w.r.t. to both x and θ, this is indicated notationally to avoid confusion. We also assume that θ is a scalar in the proof, the extension to the multivariate case is straightforward.

Using (34), we have

$$\nabla_\theta D_{\mathrm{KL}}(Q||P_\theta) = -\sum_{k=0}^{K-1} \mathbb{E}\big[\nabla_\theta \log B_k^\theta(x_k|x_{k+1})\big], \tag{41}$$

where, from (35), one has

$$\begin{aligned} &-\nabla_\theta \log B_k^\theta(x_k|x_{k+1}) \quad (42) \\ &= \delta\nabla_\theta\Big\|s_\theta(k+1, x_{k+1}) - \frac{1}{2\delta}(x_k - x_{k+1} + \delta\nabla_x \log \pi_{k+1}(x_{k+1}))\Big\|^2 \\ &= 2\delta\nabla_\theta s_\theta(k+1, x_{k+1})^{\mathrm{T}}(s_\theta(k+1, x_{k+1}) - \frac{1}{2\delta}(x_k - x_{k+1} + \delta\nabla_x \log \pi_{k+1}(x_{k+1}))). \quad (43) \end{aligned}$$

We also have

$$\nabla_\theta \mathcal{L}(\theta) = \delta \sum_{k=0}^{K-1} \mathbb{E}\left[\nabla_\theta \left\| s_\theta(k, x_k) - \nabla_x \log F_k(x_k|x_{k-1}) \right\|^2\right], \tag{44}$$

where

$$\begin{aligned}
&\delta \nabla_\theta \left\| s_\theta(k, x_k) - \nabla_x \log F_k(x_k|x_{k-1}) \right\|^2 \\
=&\delta \nabla_\theta \left\| s_\theta(k, x_k) - \frac{1}{2\delta}(x_k - x_{k+1} + \delta \nabla_x \log \pi_{k+1}(x_k)) \right\|^2 \\
=&2\delta \nabla_\theta s_\theta(k+1, x_{k+1})^{\mathrm{T}} (s_\theta(k+1, x_{k+1}) - \frac{1}{2\delta}(x_k - x_{k+1} + \delta \nabla_x \log \pi_{k+1}(x_k))).
\end{aligned} \tag{45}$$

So we obtain by using (34) and (35)

$$\nabla_\theta D_{\mathrm{KL}}(Q||P_\theta) = \nabla_\theta \mathcal{L}(\theta) + \epsilon(\theta), \tag{46}$$

for

$$\epsilon(\theta) = 2\delta \mathbb{E}\left[\sum_{k=0}^{K-1} \nabla_\theta s_\theta(k+1, x_{k+1})^{\mathrm{T}} (\nabla_x \log \pi_{k+1}(x_k) - \nabla_x \log \pi_{k+1}(x_{k+1}))\right]. \tag{47}$$

Hence we have

$$\begin{aligned}
|\epsilon(\theta)| &\leq 2\delta \sum_{k=0}^{K-1} \mathbb{E}\left[|\nabla_\theta s_\theta(k+1, x_{k+1})^{\mathrm{T}} (\nabla_x \log \pi_{k+1}(x_k) - \nabla_x \log \pi_{k+1}(x_{k+1}))|\right] \\
&\leq 2\delta \sum_{k=0}^{K-1} \mathbb{E}\left[\left\| \nabla_\theta s_\theta(k+1, x_{k+1}) \right\|^2\right]^{1/2} \mathbb{E}\left[\left\| \nabla_x \log \pi_{k+1}(x_k) - \nabla_x \log \pi_{k+1}(x_{k+1}) \right\|^2\right]^{1/2}
\end{aligned} \tag{48}$$

From Assumption 1, we have

$$\begin{aligned}
\mathbb{E}\left[\left\| \nabla_x \log \pi_{k+1}(x_k) - \nabla_x \log \pi_{k+1}(x_{k+1}) \right\|^2\right] &\leq L^2 \mathbb{E}\left[\left\| x_{k+1} - x_k \right\|^2\right] \\
&\leq 2L^2 \delta \mathbb{E}\left[\delta \left\| \nabla_x \log \pi_{k+1}(x_k) \right\|^2 + 2M\right],
\end{aligned} \tag{49}$$

where $M = \mathbb{E}_{Z \sim \mathcal{N}(0,I)}[||Z||^2]$ as $x_{k+1} = x_k + \delta \nabla_x \log \pi_{k+1}(x_k) + \sqrt{2\delta} Z$ under Q. Now using Assumption 2, it follows from (48), (49), (38), (39) and $K = O(1/\delta)$ that $\epsilon(\theta) = O(\sqrt{\delta})$. The result follows.

□

B.3 Proof of Proposition 3

We also assume here sufficient regularity conditions ensuring that the SDEs given below admit a unique solution, their corresponding time-reversed diffusions are well-defined and Girsanov theorem applies. The proof is very similar to the proof of the first part of Proposition 2 (i.e. Proposition 5 in Appendix B.2).

To start with, we use again the chain rule for KL divergences, one has

$$D_{\mathrm{KL}}(\mathcal{Q}||\mathcal{P}_\theta) = D_{\mathrm{KL}}(\eta_T||\pi_T \otimes \mathcal{N}(0, M)) + \mathbb{E}_{\eta_T}\left[D_{\mathrm{KL}}(\mathcal{Q}(\cdot|x_T, p_T)||\mathcal{P}_\theta(\cdot|x_T, p_T))\right] \tag{50}$$

where $\mathcal{Q}(\cdot|x_T, p_T)$ is the path measure induced by

$$\begin{aligned}
\mathrm{d}\bar{x}_t &= -M^{-1}\bar{p}_t \mathrm{d}t, \\
\mathrm{d}\bar{p}_t &= -\nabla \log \pi_{T-t}(\bar{x}_t)\mathrm{d}t + \zeta \bar{p}_t \mathrm{d}t + 2\zeta M \nabla_{\bar{p}_t} \log \eta_{T-t}(\bar{x}_t, \bar{p}_t)\mathrm{d}t + \sqrt{2\zeta} M^{1/2} \mathrm{d}\bar{B}_t
\end{aligned} \tag{51}$$

and $\mathcal{P}_\theta(\cdot|x_T, p_T)$ the path measure induced by

$$\begin{aligned} \mathrm{d}\bar{x}_t &= -M^{-1}\bar{p}_t \mathrm{d}t, \\ \mathrm{d}\bar{p}_t &= -\nabla \log \pi_{T-t}(\bar{x}_t)\mathrm{d}t + \zeta \bar{p}_t \mathrm{d}t + 2\zeta M s_\theta(T-t, \bar{x}_t, \bar{p}_t) + \sqrt{2\zeta} M^{1/2} \mathrm{d}\bar{B}_t, \end{aligned} \tag{52}$$

these two diffusions being initialized at $(\bar{x}_0, \bar{p}_0) = (x_T, p_T)$.

By now applying a version of Girsanov's theorem that allows for some of the components of the diffusions to be noiseless [49, Theorem 4], we obtain

$$\begin{aligned} &\mathbb{E}_{\eta_T}\Big[D_{\mathrm{KL}}(\mathcal{Q}(\cdot|x_T, p_T)||\mathcal{P}_\theta(\cdot|x_T, p_T)) \Big] \\ =&\frac{1}{2}\mathbb{E}_{\mathcal{Q}}\Big[\int_0^T \int ||2\zeta M \nabla_p \log \eta_t(x_t, p_t) - 2\zeta M s_\theta(t, x_t, p_t)||^2_{(2\zeta M)^{-1}} \mathrm{d}t \Big] \\ =&\zeta \mathbb{E}_{\mathcal{Q}}\Big[\int_0^T ||\nabla_p \log \eta_t(x_t, p_t) - s_\theta(t, x_t, p_t)||^2_M \mathrm{d}t \Big]. \end{aligned} \tag{53}$$

Now equation (18) follows directly from (50) and (53). We are now going to rewrite this loss to make it more tractable. We have for any partition of $[0, T]$ defined by $t_0 = 0 < t_1 < \cdots < t_{K-1} < t_K = T$

$$\begin{aligned} &\mathbb{E}_{\mathcal{Q}}\Big[\int_0^T ||\nabla_p \log \eta_t(x_t, p_t) - s_\theta(t, x_t, p_t)||^2_M \mathrm{d}t \Big] \\ =&\int_0^T \int ||\nabla_p \log \eta_t(x, p) - s_\theta(t, x, p)||^2_M \, \eta_t(x, p) \mathrm{d}x \mathrm{d}p \mathrm{d}t \\ =&\sum_{k=1}^K \int_{t_{k-1}}^{t_k} \int ||\nabla_p \log \eta_t(x, p) - s_\theta(t, x, p)||^2_M \, \eta_t(x, p) \mathrm{d}x \mathrm{d}p \mathrm{d}t \end{aligned}$$

where

$$\begin{aligned} &\int_{t_{k-1}}^{t_k} \int ||\nabla_p \log \eta_t(x, p) - s_\theta(t, x, p)||^2_M \eta_t(x, p) \mathrm{d}x \mathrm{d}p \mathrm{d}t \\ =&\int_{t_{k-1}}^{t_k} \int \left\{ ||\nabla_p \log \eta_t(x, p)||^2 + ||s_\theta(t, x, p||^2_M - 2 s_\theta(t, x, p)^{\mathrm{T}} M \nabla_p \log \eta_t(x, p) \right\} \eta_t(x, p) \mathrm{d}x \mathrm{d}p \mathrm{d}t \\ =&\int_{t_{k-1}}^{t_k} \int \left\{ ||s_\theta(t, x, p)||^2_M - 2 s_\theta(t, x, p)^{\mathrm{T}} M \nabla_p \log \eta_t(x, p) \right\} \eta_t(x, p) \mathrm{d}x \mathrm{d}p \mathrm{d}t + c. \end{aligned}$$

Now we have

$$\int_{t_{k-1}}^{t_k} \int s_\theta(t, x, p)^{\mathrm{T}} M \nabla_p \log \eta_t(x, p) \eta_t(x, p) \mathrm{d}x \mathrm{d}p \mathrm{d}t = \int_{t_{k-1}}^{t_k} \int s_\theta(t, x, p)^{\mathrm{T}} M \nabla_p \eta_t(x, p) \mathrm{d}x \mathrm{d}p \mathrm{d}t$$

where, using Chapman-Kolmogorov, we have

$$\eta_t(x, p) = \int \eta_{t_{k-1}}(x_{t_{k-1}}, p_{t_{k-1}}) \eta_{t|t_{k-1}}(x, p|x_{t_{k-1}}, p_{t_{k-1}}) \mathrm{d}x_{t_{k-1}} \mathrm{d}p_{t_{k-1}}.$$

Here $\eta_{t|t_{k-1}}(x, p|x_{t_{k-1}}, p_{t_{k-1}})$ denote the transition density of $(x_t, p_t) = (x, p)$ given $(x_{t_{k-1}}, p_{t_{k-1}})$ under the forward dynamics (15) so that

$$\nabla_p \eta_t(x, p) = \int \eta_{t_{k-1}}(x_{t_{k-1}}, p_{t_{k-1}}) \nabla_p \eta_{t|t_{k-1}}(x, p|x_{t_{k-1}}, p_{t_{k-1}}) \mathrm{d}x_{t_{k-1}} \mathrm{d}p_{t_{k-1}}.$$

Hence, it follows that

$$\begin{aligned} &\int_{t_{k-1}}^{t_k} \int s_\theta(t, x, p)^{\mathrm{T}} M \nabla_p \eta_t(x, p) \mathrm{d}x \mathrm{d}p \mathrm{d}t \\ =&\int_{t_{k-1}}^{t_k} \int \int \eta_{t_{k-1}}(x_{t_{k-1}}, p_{t_{k-1}}) \eta_{t|t_{k-1}}(x, p|x_{t_{k-1}}, p_{t_{k-1}}) \times \\ &\qquad \times s_\theta(t, x, p)^{\mathrm{T}} M \nabla_p \log \eta_{t|t_{k-1}}(x, p|x_{t_{k-1}}, p_{t_{k-1}}) \mathrm{d}x \mathrm{d}p \mathrm{d}x_{t_{k-1}} \mathrm{d}p_{t_{k-1}} \mathrm{d}t \end{aligned}$$

so $\zeta^{-1} D_{\mathrm{KL}}(\mathcal{Q}||\mathcal{P}_\theta)$ is equal up to an additive constant c independent of θ to

$$\sum_{k=1}^{K} \int_{t_{k-1}}^{t_k} \int\int s_\theta(t,x,p)^{\mathrm{T}} M s_\theta(t,x,p) \eta_{t_{k-1}}(x_{t_{k-1}}, p_{t_{k-1}}) \eta_{t|t_{k-1}}(x,p|x_{t_{k-1}}, p_{t_{k-1}}) \mathrm{d}x \mathrm{d}p \mathrm{d}x_{t_{k-1}} \mathrm{d}p_{t_{k-1}} \mathrm{d}t$$
$$-2 \sum_{k=1}^{K} \int_{t_{k-1}}^{t_k} \int\int \eta_{t_{k-1}}(x_{t_{k-1}}, p_{t_{k-1}}) \eta_{t|t_{k-1}}(x,p|x_{t_{k-1}}, p_{t_{k-1}}) s_\theta(t,x,p)^{\mathrm{T}} M \times$$
$$\times \nabla_p \log \eta_{t|t_{k-1}}(x,p|x_{t_{k-1}}, p_{t_{k-1}}) \mathrm{d}x \mathrm{d}p \mathrm{d}x_{t_{k-1}} \mathrm{d}p_{t_{k-1}} \mathrm{d}t$$

$$= \sum_{k=1}^{K} \int_{t_{k-1}}^{t_k} \int\int ||s_\theta(t,x,p) - \nabla_p \log \eta_{t|t_{k-1}}(x,p|x_{t_{k-1}}, p_{t_{k-1}})||_M^2 \eta_{t_{k-1}}(x_{t_{k-1}}, p_{t_{k-1}}) \times$$
$$\times \eta_{t|t_{k-1}}(x,p|x_{t_{k-1}}, p_{t_{k-1}}) \mathrm{d}x \mathrm{d}p \mathrm{d}x_{t_{k-1}} \mathrm{d}p_{t_{k-1}} \mathrm{d}t + c$$

Hence, finally we obtain as required

$$D_{\mathrm{KL}}(\mathcal{Q}||\mathcal{P}_\theta) = \zeta \sum_{k=1}^{K} \int_{t_{k-1}}^{t_k} \mathbb{E}\left[||s_\theta(t,x_t,p_t) - \nabla_{p_t} \log \eta_{t|t_{k-1}}(x_t,p_t|x_{t_{k-1}}, p_{t_{k-1}})||_M^2\right] \mathrm{d}t + C_2. \tag{54}$$

C Details of the reverse integrator for the Hamiltonian variant

We take equation (17) and rewrite in the following form.

$$\mathrm{d}\bar{x}_t = -M^{-1}\bar{p}_t \mathrm{d}t, \tag{55}$$
$$\mathrm{d}\bar{p}_t = -\nabla \log \pi_{T-t}(\bar{x}_t)\mathrm{d}t + 2\zeta[M s_\theta(T-t, \bar{x}_t, \bar{p}_t) + \bar{p}_t]\mathrm{d}t + [-\zeta \bar{p}_t \mathrm{d}t + \sqrt{2\zeta} M^{1/2} \mathrm{d}\bar{B}_t].$$

As in [15], we have rewritten the SDE for $\bar{p}_t$ in three distinct components, the last one corresponding to an Ornstein–Ulhenbeck process. The integrator we propose however differs from [15] and will rely on a leapfrog component. In details, we integrate successively over the time interval $\bar{x} \in [t'-\delta, t'] \implies x \in [t+\delta, t]$, where $t = T - t'$. This allows us to directly compare random variables with the forward integrator in Section 3.3.

First, we integrate

$$\mathrm{d}\bar{x}_t = -M^{-1}\bar{p}_t \mathrm{d}t, \qquad \mathrm{d}\bar{p}_t = -\nabla \log \pi_{T-t}(\bar{x}_t)\mathrm{d}t \tag{56}$$

by using $(\tilde{x}_{t+\delta}, \tilde{p}_{t+\delta}) = \Phi_t^{-1}(x_{t+\delta}, p_{t+\delta})$ where $\Phi_t^{-1}(x,p) = \Phi_{\text{flip}} \circ \Phi_t \circ \Phi_{\text{flip}}(x,p)$ with $\Phi_{\text{flip}}(x,p) = (x,-p)$. We then integrate

$$\mathrm{d}\bar{x}_t = 0, \qquad \mathrm{d}\bar{p}_t = 2\zeta[M s_\theta(T-t, \bar{x}_t, \bar{p}_t) + \bar{p}_t]\mathrm{d}t \tag{57}$$

using $(\hat{x}_t, \hat{p}_t) = (\tilde{x}_{t+\delta}, f_\theta(t+\delta, \tilde{x}_{t+\delta}, \tilde{p}_{t+\delta}))$ for $f_\theta(t+\delta, \tilde{x}_{t+\delta}, \tilde{p}_{t+\delta}) = \tilde{p}_{t+\delta} + 2\delta\zeta[M s_\theta(t+\delta, \tilde{x}_{t+\delta}, \tilde{p}_{t+\delta}) + \tilde{p}_{t+\delta}]$ and finally

$$\mathrm{d}\bar{x}_t = 0, \qquad \mathrm{d}\bar{p}_t = -\zeta \bar{p}_t \mathrm{d}t + \sqrt{2\zeta} M^{1/2} \mathrm{d}\bar{B}_t \tag{58}$$

using $(x_t, p_t) = (\hat{x}_t, h\hat{p}_t + \sqrt{1-h^2}\epsilon)$ with $\epsilon \sim \mathcal{N}(0, M)$ for $h = \exp(-\zeta\delta)$.

Note that the x variable only changes in ones of these steps so that $x_t = \hat{x}_t = \tilde{x}_{t+\delta}$, allowing us to eliminate the redundant terms. We can also eliminate $\hat{p}_t$ by substitution. We therefore need only to work in terms of the triple $(x_t, p_t, \tilde{p}_{t+\delta})$ at each combined time step. The conditional distribution for these remaining random variables is then:

$$B_k^\theta(x_t, p_t, \tilde{p}_{t+\delta}|x_{t+\delta}, p_{t+\delta}) = \delta_{\Phi^{-1}(x_{t+\delta}, p_{t+\delta})}(x_t, \tilde{p}_{t+\delta}) \mathcal{N}(p_t; h f_\theta(t+\delta, x_t, \tilde{p}_{t+\delta}), (1-h^2)M) \tag{59}$$

where $\delta_{\Phi^{-1}(x_{t+\delta}, p_{t+\delta})}(x_t, \tilde{p}_{t+\delta})$ is a Dirac-delta centred on $\Phi^{-1}(x_{t+\delta}, p_{t+\delta})$. From the main text we note that the forward Hamiltonian integrator has the conditional distribution:

$$F_{k+1}(\tilde{p}_{t+\delta}, x_{t+\delta}, p_{t+\delta}|x_t, p_t) = \delta_{\Phi(x_t, \tilde{p}_{t+\delta})}(x_{t+\delta}, p_{t+\delta}) \mathcal{N}(\tilde{p}_{t+\delta}; h p_t, (1-h^2)M). \tag{60}$$

We again transition to discretized notation with $\delta := T/K$, and $k = 0, ..., K$. Bringing together all the remaining random variables at each time step we have:

$$Q(\mathbf{x}, \mathbf{p}) = \pi_0(x_0)\mathcal{N}(p_0; 0, M) \prod_{k=0}^{K-1} F_{k+1}(x_{k+1}, p_{k+1}, \tilde{p}_{k+1}|x_k, p_k),$$

$$\Gamma_\theta(\mathbf{x}, \mathbf{p}) = \gamma(x_K)\mathcal{N}(p_K; 0, M) \prod_{k=0}^{K-1} B_k^\theta(x_k, p_k, \tilde{p}_{k+1}|x_{k+1}, p_{k+1}).$$

We are interested in the unnormalized importance weight $w_\theta(\mathbf{x}, \mathbf{p}) = \Gamma_\theta(\mathbf{x}, \mathbf{p})/Q(\mathbf{x}, \mathbf{p})$. We note that the division of delta functions is in general ill-defined and in our case should be interpreted formally in terms of the action of the volume preserving Leapfrog integrator flows ϕ_t, as we describe in the main text. Further although the numerator and denominator in the definition of $w_\theta(\mathbf{x}, \mathbf{p})$ do not have density with respect to Lebesgue measure, $\Gamma_\theta(\mathbf{x}, \mathbf{p})$ has density with respect to $Q(\mathbf{x}, \mathbf{p})$ and therefore $w_\theta(\mathbf{x}, \mathbf{p})$ may be interpreted as the resulting Radon-Nikodym derivative. The explicit expression for $\log w_\theta(\mathbf{x}, \mathbf{p})$ in equation (21) of the main text then follows.

We now show informally how maximizing the corresponding ELBO $\mathbb{E}_Q[\log w_\theta(\mathbf{x}, \mathbf{p})]$ corresponds approximately to minimizing the score matching loss given in Proposition 3 for $\delta \ll 1$. We restrict the derivation to $M = I$ for simplicity. For pedagogical reasons, it is beneficial here to get back to the continuous time notations and recall that we use k corresponds to $t_k = k\delta$. From direct calculations, maximizing the ELBO is equivalent to minimizing

$$J(\theta) = \sum_{k=0}^{K-1} \mathbb{E}_Q \left[||p_{k\delta} - hf_\theta(k\delta, x_{k\delta}, \widetilde{p}_{(k+1)\delta})||^2\right]$$

where

$$\mathbb{E}_Q \left[||p_{k\delta} - hf_\theta(k\delta, x_{k\delta}, \widetilde{p}_{(k+1)\delta})||^2\right] = \mathbb{E}_Q \left[||p_{k\delta} - h\left(\widetilde{p}_{(k+1)\delta} + 2\delta\zeta[s_\theta(k\delta, x_{k\delta}, \widetilde{p}_{(k+1)\delta}) + \widetilde{p}_{(k+1)\delta}]\right)||^2\right]$$

Note that $h = \exp(-\delta\zeta) \approx 1 - \delta\zeta$ so we obtain

$$\begin{aligned}
&\mathbb{E}_Q \left[||p_{k\delta} - hf_\theta(k\delta, x_{k\delta}, \widetilde{p}_{(k+1)\delta})||^2\right] \\
\approx &\mathbb{E}_Q \left[||p_{k\delta} - (1-\delta\zeta)\left(\widetilde{p}_{(k+1)\delta} + 2\delta\zeta[s_\theta(k\delta, x_{k\delta}, \widetilde{p}_{(k+1)\delta}) + \widetilde{p}_{(k+1)\delta}]\right)||^2\right] \\
\approx &\mathbb{E}_Q \left[||p_{k\delta} - (1+\delta\zeta)\widetilde{p}_{(k+1)\delta} - 2\delta\zeta s_\theta(k\delta, x_{k\delta}, \widetilde{p}_{(k+1)\delta})||^2\right]
\end{aligned}$$

by neglecting terms of order δ^2. Now we try to further understand the asymptotic when $\delta \to 0$. We have that

$$(x_{k\delta}, \widetilde{p}_{(k+1)\delta}) = \Phi_{k\delta}^{-1}(x_{(k+1)\delta}, p_{(k+1)\delta}).$$

Now as we use for $\Phi_{k\delta}$ a leapfrog-type integrator, we do have

$$\Phi_{k\delta}^{-1}(x', p') = \Phi_{\text{flip}} \circ \Phi_{k\delta} \circ \Phi_{\text{flip}}(x', p')$$

where $\Phi_{\text{flip}}(x, p) = (x, -p)$ and, for $\delta \to 0$, we have

$$\Phi_{k\delta}(x, p) \approx (x + \delta p, p - \delta\nabla E_{k\delta}(x)),$$

where $\pi_{k\delta}(x) \propto \exp(-E_{k\delta}(x))$. So we have for

$$\begin{aligned}
\Phi_{k\delta}^{-1}(x', p') &= \Phi_{\text{flip}} \circ \Phi_t \circ \Phi_{\text{flip}}(x', p') \\
&= \Phi_{\text{flip}} \circ \Phi_{k\delta}(x', -p') \\
&= \Phi_{\text{flip}}(x' - \delta p', -p' - \delta\nabla E_{k\delta}(x')) \\
&= (x' - \delta p', p' + \delta\nabla E_{k\delta}(x')).
\end{aligned}$$

It follows that

$$\begin{aligned}
&\mathbb{E}_Q \left[||p_{k\delta} - (1+\delta\zeta)\widetilde{p}_{(k+1)\delta} - 2\delta\zeta s_\theta(k\delta, x_{k\delta}, \widetilde{p}_{(k+1)\delta})||^2\right] \\
\approx &\mathbb{E}_Q \left[||p_{k\delta} - (1+\delta\zeta)(p_{(k+1)\delta} + \delta\nabla E_{k\delta}(x_{(k+1)\delta})) - 2\delta\zeta s_\theta(k\delta, x_{(k+1)\delta}, p_{(k+1)\delta})||^2\right] \\
\approx &\mathbb{E}_Q \left[||p_{k\delta} - p_{(k+1)\delta} - \delta(\zeta p_{(k+1)\delta} + \nabla E_{k\delta}(x_{(k+1)\delta}) + 2\zeta s_\theta(k\delta, x_{(k+1)\delta}, p_{(k+1)\delta}))||^2\right].
\end{aligned}$$

Now we have

$$\begin{aligned}
&\frac{1}{4\zeta\delta}\mathbb{E}_Q\left[||p_{k\delta} - p_{(k+1)\delta} - \delta(\zeta p_{(k+1)\delta} + \nabla E_{k\delta}(x_{(k+1)\delta}) + 2\zeta s_\theta(k\delta, x_{(k+1)\delta}, p_{(k+1)\delta}))||^2\right] \\
=&\delta\zeta\mathbb{E}_Q\left[||s_\theta(k\delta, x_{(k+1)\delta}, p_{(k+1)\delta}) - \frac{1}{2\zeta\delta}(p_{k\delta} - p_{(k+1)\delta} - \delta(\zeta p_{(k+1)\delta} + \nabla E_{k\delta}(x_{(k+1)\delta})))||^2\right] \\
\approx&\delta\zeta\mathbb{E}_Q\left[||s_\theta(k\delta, x_{(k+1)\delta}, p_{(k+1)\delta}) - \frac{1}{2\zeta\delta}(p_{k\delta} - p_{(k+1)\delta} - \delta(\zeta p_{k\delta} + \nabla E_{k\delta}(x_{k\delta})))||^2\right] \\
=&\delta\zeta\mathbb{E}_Q\left[||s_\theta(k\delta, x_{(k+1)\delta}, p_{(k+1)\delta}) - \nabla_{p_{(k+1)\delta}} \log F(p_{(k+1)\delta}|p_{k\delta}, x_{k\delta})||^2\right], \qquad (61)
\end{aligned}$$

where $F(p_{(k+1)\delta}|p_{k\delta}, x_{k\delta}) = \mathcal{N}(p_{(k+1)\delta}; (1-\delta\zeta)p_{k\delta} - \delta\nabla E_{k\delta}(x_{k\delta}); 2\zeta\delta I)$ and the joint distribution $F(x_{(k+1)\delta}, p_{(k+1)\delta})|p_{k\delta}, x_{k\delta}) = \delta_{x_{k\delta}-\delta p_{k\delta}}(x_{(k+1)\delta})F(p_{(k+1)\delta}|p_{k\delta}, x_{k\delta})$ is an Euler approximation of the forward transition of the underdamped Langevin dynamics. Now we expect similarly $\eta_{(k+1)\delta|k\delta}(x_{(k+1)\delta}, p_{(k+1)\delta}|x_{k\delta}, p_{k\delta}) \approx \eta_{(k+1)\delta|k\delta}(p_{(k+1)\delta}|x_{k\delta}, p_{k\delta})\eta_{(k+1)\delta|k\delta}(x_{(k+1)\delta}|x_{k\delta}, p_{k\delta})$ for $\delta \ll 1$ and (61) is thus an approximation of the score matching loss.

D Diffusion processes: SGM, Langevin and AIS

We provide here a more detailed discussion between the similarities and differences between the diffusion process considered for SGM and the proposed approach. We first recall some basic elements of diffusion processes. Consider the diffusion $(x_t)_{t\in[0,T]}$ on $\mathbb{R}^d$

$$\mathrm{d}x_t = f(t, x_t)\mathrm{d}t + \sqrt{2}\mathrm{d}B_t, \qquad x_0 \sim q_0, \qquad (62)$$

where $(B_t)_{t\in[0,T]}$ is standard multivariate Brownian motion and q_0 is the initial distribution. The law q_t of x_t induced by this diffusion satisfies the Fokker–Planck–Kolmogorov equation

$$\frac{\partial q_t(x)}{\partial t} = -\nabla \cdot [f(t, x)q_t(x)] + \Delta q_t(x) \qquad (63)$$

where $\nabla \cdot [f(t, x)q_t(x)] = \sum_{i=1}^d \frac{\partial [f_i(t,x)q_t(x)]}{\partial x_i}$ and $\Delta q_t(x) = \sum_{i=1}^d \frac{\partial^2 q_t(x)}{\partial x_i^2}$ denote the divergence and Laplacian operators; see e.g. [32].

For SGM, we define the following Ornstein–Ulhenbeck process on $\mathbb{R}^d$ which corresponds to using

$$f_{\mathrm{OU}}(t, x) = -x \qquad (64)$$

where $(B_t)_{t\in[0,T]}$ is standard multivariate Brownian motion and q_0 is the data distribution. This is also known in the SGM literature as the variance preserving diffusion [48]. This process adds noise progressively to the complex data distribution and converges geometrically fast to its invariant distribution which is the standard multivariate Gaussian $\pi_{\mathrm{OU}}(x) = \mathcal{N}(x; 0, I)$ (see e.g. [12]) verifying indeed that the r.h.s. of (63) satisfies

$$-\nabla \cdot [f_{\mathrm{OU}}(t, x)\pi_{\mathrm{OU}}(x)] + \Delta\pi_{\mathrm{OU}}(x) = 0.$$

More generally, a time-homogeneous Langevin diffusion to sample from a target distribution π is of the form

$$f_{\mathrm{Lgv}}(t, x) = \nabla \log \pi(x). \qquad (65)$$

It is easily check that π is indeed an invariant distribution as the r.h.s. of (63) satisfies

$$-\nabla \cdot [f_{\mathrm{Lgv}}(t, x)\pi(x)] + \Delta\pi(x) = 0,$$

so that $q_t = \pi$ for all t if $q_0 = \pi$. Moreover, q_t converges to π whatever being π_0; see e.g. [42]. However, obtaining sharp quantitative bounds for complex π is a more difficult task than for $\pi_{\mathrm{OU}}(x) = \mathcal{N}(x; 0, I)$ in general.

In the context of this paper, the "forward" diffusion process we define is a *time-inhomogeneous* Langevin algorithm

$$f_{\mathrm{AIS}}(t, x) = \nabla \log \pi_t(x), \qquad (66)$$

with $q_0 = \pi_0$ an easy-to-sample distribution and $(\pi_t)_{t\in[0,T]}$ a non-constant curve of distributions such that $\pi_T = \pi$. So, contrary to SGM, which starts from a complex distribution and moves

towards a simple distribution, we start here from a simple distribution and moves towards a complex distribution.

In this scenario, even in $q_0 = \pi_0$ then we do not have $q_t = \pi_t$ as the diffusion always lags behind its stationary distribution at time t. However, quantitative results measuring the discrepancy between the law of x_T and π for such annealed diffusions have been obtained; see e.g. [20, 17, 50]. For this discrepancy to be small, one requires π_t to vary slowly over time.

E Experimental Details

In all experiments, we sweep over diffusion time, number of steps, step-sizes and whether to learn them, and the annealing schedule. We identify the best parameters for each sampler individually on a validation set and then re-run these methods using 5 different seeds to obtain error bars on test set performance. All experiments were executed on 8 GPUs for parallelized training and a single instance of our most expensive experiment (VAE) takes under 3 hours including evaluation. Experiments are implemented in JAX [6] using the DeepMind JAX ecosystem [4].

E.1 Sampler parameterization

For all models, the step size was learned via a function $\epsilon_\theta(t)$ which is a 2-layer neural network with 32 hidden units, followed by a scaled sigmoid function which constrains $\epsilon_\theta(t) < 0.25$. As in prior work [18] we found this alleviated some instabilities in training.

When learning the annealing schedule, we parameterize an increasing sequence of T steps using unconstrained parameters b_t (initialized to the same constant). We map these to our annealing schedule with

$$\beta_t = \frac{\sum_{t' \leq t} \sigma(b_{t'})}{\sum_{t'=1}^{T} \sigma(b_{t'})} \tag{67}$$

where we fix $\beta_0 = 0$ and σ is the sigmoid function. This ensures that $\beta_0 = 0$, $\beta_K = 1$, and $\beta_t < \beta_{t'}$ when $t < t'$.

For UHA [18], we also learn the momentum refreshment parameter $\eta \in (0, 1)$. We parameterize this with a parameter u and define $\eta = .98\sigma(u) + .01$ to keep the values in the range $(.01, .99)$ which we found alleviated sone training instabilities.

E.2 Score model parameterization

We parameterize our score model $s_\theta(t, x)$ using an MLP residual network. We first project the x to dim d_h using a linear layer and embed discrete time steps t to dim d_t using a learned embedding map. We then apply k residual blocks.

Each block begins with a layer norm [3] operation followed by a nonlinearity. We project the hidden representation to dim $2 \cdot d_h$ using a linear layer, project the embedding of t to dim $2 \cdot d_h$ using another linear map and add them together. We then apply another nonlinearity and then project the back to d_h using another linear layer. We use the swish nonlinearity [40] throughout.

To ensure our ELBO is initialized to a reasonable value we *warm start* it so that at initialization, the score model outputs the standard AIS backward kernels. For the ULA version of our approach we do this by defining a score model $\tilde{s}_\theta(t, x)$ as explained above (but set the final layer weights to 0 at initialization) and define:

$$s_\theta(t, x) = \tilde{s}_\theta(t, x) + \nabla_x \log \gamma_t(x). \tag{68}$$

For the UHA version we parameterize a score model $\tilde{s}_\theta(t, x, p)$ and define:

$$s_\theta(t, x, p) = \tilde{s}_\theta(t, x, p) - M^{-1}p. \tag{69}$$

In both cases we found this led to much faster convergence and better results overall.

E.3 Hyper-parameters

In all experiments we use $k = 3$ residual blocks in our score network. For our Gaussian experiments we set $d_h = 512$ and $d_t = 16$. All models are trained with the Adam optimizer [30] with learning rates 0.001 and 0.0001 and up to 300k iterations with a batch size of 128. For static targets, we produce an estimate of $\log Z$ using 16,384 importance samples. As these methods produce a stochastic lower-bound on $\log Z$ we report the result from the hyper-parameter setting which gives the largest $\log Z$ estimate.

The VAE experiment uses architectures described in [8], which consists of encoder and decoder MLPs with two hidden layers with 200 units each, $\tanh$ activations, and 50 latent dimensions. In contrast to [18], we found this architecture work better than the one described in [18], especially when trained for more iterations. We chose the best performing models and their hyperparameters by monitoring validation performance during training. We report performance of the best combination on the full test set, for each model respectively. The best performance were reached with a matched the number of sampler steps between ULA/ULA-MCD and UHA/UHA-MCD – 64 and 32 respectively.

F Additional Results

F.1 Static Targets

Here we present additional results on a $\mathcal{N}(10, I)$ target, a $\mathcal{N}(0, 0.1I)$ target, and a Laplace$(0, I)$ target. Results can be found in Tables 4, 5, and 6.

Sampler	ULA		UHA		ULA-MCD		UHA-MCD	
# steps	64	256	64	256	64	256	64	256
Dim-20	-46.75 ± 0.69	-6.23 ± 0.91	0.0002 ± 0.0008	0.0002 ± 0.0004	-0.017 ± 0.020	0.0034 ± 0.0055	-0.0005 ± 0.0007	0.0000 ± 0.0005
Dim-200	-752.25 ± 2.43	-160.60 ± 2.03	0.0003 ± 0.0026	-0.0005 ± 0.0007	-4.74 ± 1.20	-0.019 ± 0.047	0.0008 ± 0.0032	0.0060 ± 0.0084
Dim-500	-1999.40 ± 18.49	-455.20 ± 7.90	0.0006 ± 0.0007	-0.0008 ± 0.0012	-21.62 ± 1.64	-0.29 ± 0.16	0.0030 ± 0.0063	0.0099 ± 0.0050

Table 4: $\log Z$ estimates for a $\mathcal{N}(10, I)$ target. Averages and standard errors over 3 seeds.

Sampler	ULA		UHA		ULA-MCD		UHA-MCD	
# steps	64	256	64	256	64	256	64	256
Dim-20	-1.58 ± 0.42	-0.27 ± 0.13	-4.46 ± 0.93	-0.41 ± 0.51	0.0095 ± 0.0155	0.0057 ± 0.0052	0.0038 ± 0.0273	0.0002 ± 0.0120
Dim-200	-166.23 ± 5.15	-58.33 ± 1.99	-228.79 ± 3.64	-74.62 ± 1.60	-4.53 ± 0.96	-0.67 ± 0.33	-4.10 ± 0.98	-1.47 ± 0.57
Dim-500	-545.77 ± 5.55	-207.15 ± 5.73	-704.96 ± 7.30	-247.11 ± 4.83	-27.36 ± 2.23	-10.11 ± 1.17	-29.20 ± 3.46	-5.14 ± 1.50

Table 5: $\log Z$ estimates for a $\mathcal{N}(0, 0.1I)$ target. Averages and standard errors over 3 seeds.

Sampler	ULA		UHA		ULA-MCD		UHA-MCD	
# steps	64	256	64	256	64	256	64	256
Dim-20	0.31 ± 0.45	0.40 ± 0.63	-0.0086 ± 0.1314	-0.0077 ± 0.1340	0.092 ± 0.235	-0.23 ± 0.02	0.0003 ± 0.1573	-0.020 ± 0.141
Dim-200	-5.40 ± 0.53	-4.54 ± 0.96	-5.27 ± 0.18	-5.28 ± 0.19	-5.08 ± 0.52	-4.31 ± 0.66	-5.30 ± 0.15	-5.39 ± 0.09
Dim-500	-17.67 ± 1.91	-17.25 ± 0.85	-17.91 ± 0.78	-17.91 ± 0.78	-17.64 ± 1.98	-15.52 ± 1.26	-18.11 ± 0.70	-18.11 ± 0.72

Table 6: $\log Z$ estimates for a Student-T target. Averages and standard errors over 3 seeds.

F.2 Normalizing Flow

In this experiment we train NICE [14] flows which are fitted on downsampled variants of the MNIST dataset at resolutions 7×7, 14×14, and 28×28. All models are trained for 100K steps with a batch size of 128 and then $\log Z$ is estimated using 4096 importance samples.

Results can be seen in Table 7. In the largest setting we can see that UHA outperforms ULA, but our method outperforms both. We note that a Sequential Monte Carlo sampler or AIS with Metropolis-Hastings corrections would be capable of estimating $\log Z$ very close to the true value of 0. We have found that unadjusted samplers required for building a differentiable evidence lower-bound have difficulties sampling from this target distribution. As discussed in the Limitations section of the main paper, we hypothesize that this is due to the fact that we are limited to using a relatively small number of transitions, since backpropagating through these repeated updates can be unstable. We believe this is a limitation of this approach to inference and will impact any approach which utilizes an unadjusted forward sampler. Still, we note that our approach to learning an optimized reversal (MCD) leads to improvements over the standard AIS reversals. We believe addressing these issues to be a key area for future research to focus.

Dimension	ULA	UHA	MCD (ours)
7×7	0.14	0.17	**0.11**
14×14	13.24	15.04	**6.25**
28×28	141.29	82.16	**23.10**

Table 7: $\log Z$ estimate absolute error for Normalizing flows.

F.3 Score Network Ablation

We include additional results exploring the impact of various score-network architectures on performance. We re-run our Gaussian Mixture experiments in dimension 200 using 1) an MLP with residual connections and 2) a standard MLP, both with 1, 2, and 3 layers. We run these experiments with 64 and 128 sampling steps. Results can be seen in Figure 4. We see that more expressive architectures lead to better performance in general. Further, we find that with more sampling steps, the impact of a more expressive score network diminishes. This aligns with intuition as, when using more steps we will use a smaller step size, and the standard AIS reversal becomes a better approximation to the true reversal.

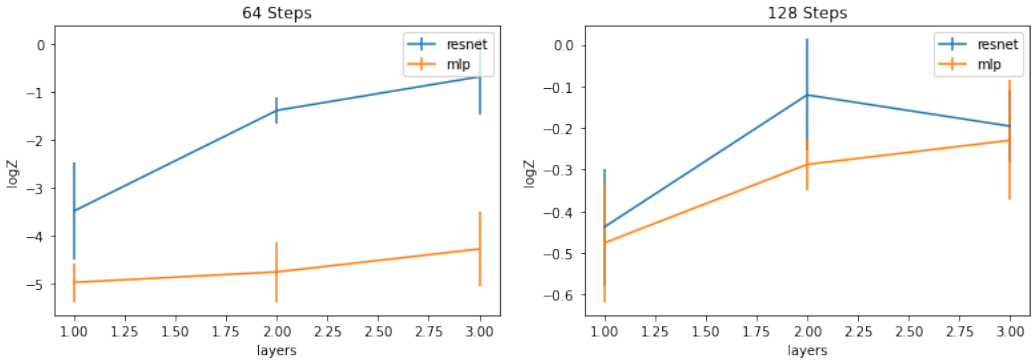


Figure 4: Score network architecture ablation on Gaussian mixture target (main results can be found in Table 1). Left: results with 64 sampling steps, right: results with 128 sampling steps.