What's the score?
Automated Denoising Score Matching for Nonlinear Diffusions

Raghav Singhal [*1] **Mark Goldstein** [*1] **Rajesh Ranganath** [1 2]

Abstract

Reversing a diffusion process by learning its score forms the heart of diffusion-based generative modeling and for estimating properties of scientific systems. The diffusion processes that are tractable center on linear processes with a Gaussian stationary distribution, limiting the kinds of models that can be built to those that target a Gaussian prior or more generally limits the kinds of problems that can be generically solved to those that have conditionally linear score functions. In this work, we introduce a family of tractable denoising score matching objectives, called local-DSM, built using local increments of the diffusion process. We show how local-DSM melded with Taylor expansions enables automated training and score estimation with nonlinear diffusion processes. To demonstrate these ideas, we use automated-DSM to train generative models using non-Gaussian priors on challenging low dimensional distributions and the CIFAR10 image dataset. Additionally, we use the automated-DSM to learn the scores for nonlinear processes studied in statistical physics.

1. Introduction

Modeling with diffusion processes has led to advances in generative models (Dhariwal & Nichol, 2021; Nichol & Dhariwal, 2021; Nichol et al., 2021; Sasaki et al., 2021) and in the computation of properties of scientific systems through the estimation of the score of a diffusion (Boffi & Vanden-Eijnden, 2023a;b; Huang & Wang, 2024).

Score models can be trained for a generic diffusion process, that may be nonlinear, using the the implicit score matching (ISM) objective (Huang et al., 2021; Song & Ermon, 2020; Boffi & Vanden-Eijnden, 2023b). However, estimating the ISM objective requires computing the divergence of the score model. Computing the divergence directly is memory intensive, therefore, the stochastic Hutchinson trace estimator (Hutchinson, 1989; Grathwohl et al., 2018) is used for computational efficiency. However, the use of the stochastic trace estimator leads to noisy gradients and requires differentiation during the forward pass.

An alternative to the ISM objective is the denoising score matching (DSM) objective (Vincent, 2011; Song et al., 2020a;b). The DSM objective has powered many of the improvements in diffusion-based generative models (DBGMs) (Song et al., 2020b; Dockhorn et al., 2021; Singhal et al., 2023). However, training with DSM requires the score of the transition kernel $q(\mathbf{y}_t \mid \mathbf{y}_0)$, which is typically not available for nonlinear processes. Neither ISM or DSM provide a good option for training score models with generic, nonlinear noise or inference processes.

A natural question one can ask is why study nonlinear inference processes? At a high level more generic, easy-to-use computation has a history of unlocking other techniques (Baydin et al., 2018; Ranganath et al., 2014; 2016; Kucukelbir et al., 2017). Recent work introduces new choices of inference processes for generative modeling, but the processes introduced are limited to linear ones with Gaussian stationary distributions (Dockhorn et al., 2021; Singhal et al., 2023; Pandey & Mandt, 2023; Du et al., 2023). Automated training for nonlinear inference processes would allow for rapid prototyping of non-Gaussian priors using nonlinear Langevin processes (Pavliotis, 2016) and, more generally, nonlinear drifts in the inference process.

Next, in several applications the inference process is given to us. For many systems of interest in statistical physics (Chandler, 1987; Spohn, 2012; Otsubo et al., 2022), finance (Kusuoka & Ninomiya, 2004), biology (Fleming, 1975), the evolution of the system is governed by high-dimensional nonlinear diffusion processes. Several properties of these systems, such as the entropy production rate (Otsubo et al., 2022), require access to the density and are challenging to estimate from samples alone. Typical approaches for estimating the density, such as solving the Fokker-Planck equation (Pavliotis, 2016) are infeasible in high dimensions. Therefore, Boffi & Vanden-Eijnden

[*]Equal contribution [1]Courant Institute of Mathematical Sciences, New York University [2]Center for Data Science, New York University. Correspondence to: Raghav Singhal <rsinghal@nyu.edu>, Mark Goldstein <goldstein@nyu.edu>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024.

(2023a;b) use techniques for learning the score developed in DBGMs to study quantities such as the density, the probability current, and the entropy production of physical systems. *Given the utility of nonlinear inference processes and the lack of efficient estimation with them, we need new objectives for training with nonlinear inference processes.*

In this work, we introduce a training algorithm, *automated* DSM, that expands the applicability of DSM to a broad class of nonlinear inference processes. Automated DSM relies on a few methodological innovations:

1. Derive a local-DSM objective built from local increments of the transition kernel. For image-generation experiments, we also develop a perceptually weighted local-DSM objective.
2. Create tractable approximations to the score of the transition kernel $q(\mathbf{y}_t \mid \mathbf{y}_s)$ using local linearization
3. Design time pairs s, t to control the error in approximating the local transition kernel $q(\mathbf{y}_t \mid \mathbf{y}_s)$.

To test these automations, we train DBGMs with inference processes with non-Gaussian stationary distribution and score models for nonlinear inference processes studied in the physical sciences. In our experiments:

1. We show that training DBGMs with the local-DSM objective is faster than the ISM objective, on low-dimensional synthetic datasets, physical systems, and CIFAR10.
2. We demonstrate the flexibility of automated DSM by training DBGMs with non-Gaussian priors, such as a mixture of Gaussians and the Logistic distribution, and estimating scores for nonlinear inference processes in the sciences *without requiring manual derivations*.

These findings highlights that local DSM objectives and the automations provided in this work enable fast and derivation free training for nonlinear inference processes.

1.1. Related Work

Huang et al. (2022); Boffi & Vanden-Eijnden (2023a;b) train diffusion models using nonlinear inference processes with the ISM objective. In section 4, we show that even for 2d problems, using the local-DSM objective leads to faster convergence and better sample quality compared to using the ISM objective.

Doucet et al. (2022) apply techniques from score-based generative modeling to annealed importance sampling (Neal, 2001). For a given unnormalized target density π, they specify discrete-time Markov transition kernels $q(\mathbf{y}_{k+1} \mid \mathbf{y}_k)$ using the Euler-Maruyama (Särkkä & Solin, 2019) updates of a Langevin process with π as the stationary distribution, and then learn the reverse transition kernels $p_\theta(\mathbf{z}_k \mid \mathbf{z}_{k+1})$. They derive a discrete-time denoising score matching objective based on Kullback-Leibler (KL) divergence, similar to Sohl-Dickstein et al. (2015); Ho et al. (2020). In this work, we derive a continuous-time variational lower bound (ELBO) on the model likelihood $\log p_\theta(x)$ as well as considering arbitrary nonlinear inference processes. Training in continuous-time is known to lead to tighter likelihood bounds (Kingma et al., 2021).

Implicit nonlinear Diffusions. Kim et al. (2022) introduce a variational lower bound for *implicit* nonlinear inference processes by using a normalizing flow to map the data to a latent space and then learning a DBGM in the latent space with linear inference proceesses. Similarly, Vahdat et al. (2021); Rombach et al. (2022) train DBGMs in the latent space of variational autoencoders. However, the set of processes considered in the latent space are still linear. In this work, we consider a complementary approach: diffusion processes that are *explicitly* nonlinear, without the use of a latent space.

Stochastic Interpolants. Albergo & Vanden-Eijnden (2022); Albergo et al. (2023) introduce an interpolant process that is defined via independent samples $\mathbf{y}_0 \sim q_{\text{data}}$ and $\mathbf{y}_1 \sim \pi_\theta$. The interpolant is defined as $\mathbf{y}_t = I(t, \mathbf{y}_0, \mathbf{y}_1)$, and the idea is to define noisy states as an interpolation between samples from two endpoint distributions, as opposed to the approach of picking a stationary distribution in DBGMs. However, when the interest is not generative modeling, but to study physical, biological, or financial systems that are explicitly known to follow a certain nonlinear stochastic differential equation (SDE), it may be challenging to find the endpoint distribution $\mathbf{y}_1$ and interpolant I such that $\mathbf{y}_t$ is distributed according to solutions of the given SDE under the given initial conditions $\mathbf{y}_0$.

Bartosh et al. (2024) introduce neural flow diffusion models. They define an inference process $\mathbf{y}_t$ using a learnable transformation $\mathbf{y}_t = F_\phi(\varepsilon, t, x)$, where $\varepsilon \sim \mathcal{N}(0, I_d)$ and the transformation F_ϕ is invertible with respect to ε; these transformations are shown to improve likelihoods on image modeling tasks. However, if the object of interest is the score of a *given* SDE, finding the corresponding invertible transformation F_ϕ is challenging in general.

2. Background and Setup

Training generative models with diffusions or score estimation starts with defining an *inference process* $\mathbf{y}_t$, which is of the form:

$$d\mathbf{y}_t = f(\mathbf{y}_t, t)dt + g(t)d\mathbf{w}_t, \qquad t \in [0, T] \tag{1}$$

where $\mathbf{y}_0 \sim q_{\text{data}}$ and f, g are chosen such that $q(\mathbf{y}_T) \approx \pi_\theta$ where π_θ is the model prior. We then define a generative

process $\mathbf{z}_t$ with the model drift and diffusion co-efficient tied to the inference process:

$$d\mathbf{z}_t = \left[gg^\top s_\theta - f\right](\mathbf{z}_t, T-t)dt + g(T-t)d\mathbf{w}_t, \quad (2)$$

where $s_\theta : \mathbf{R}^d \to \mathbf{R}^d$ is the score network and integration is in the forward direction (Huang et al., 2021; Singhal et al., 2023).

Training the score network s_θ with maximum likelihood estimation is computationally expensive as it requires estimating the model likelihood $\log p_\theta(\mathbf{z}_T = x)$, which would require solving a high-dimensional partial differential equation. Song & Ermon (2020); Huang et al. (2021); Kingma et al. (2021) instead derive a variational lower bound, called the ISM ELBO:

$$\log p_\theta(x) \geq \mathop{\mathbb{E}}_{q(\mathbf{y}_T \mid x)}[\log \pi_\theta(\mathbf{y}_T)] \quad + \quad (3)$$
$$\int_0^T \mathop{\mathbb{E}}_{q(\mathbf{y}_t \mid x)}\Big[-\frac{1}{2}\left\|s_\theta\right\|^2_{gg^\top} - \nabla_{\mathbf{y}_t} \cdot (gg^\top s_\theta - f)\Big]dt$$

where $\|\mathbf{x}\|_{\mathbf{A}} = \mathbf{x}^\top \mathbf{A}\mathbf{x}$ for a positive semi-definite matrix $\mathbf{A}$. Estimating the ISM ELBO requires computing the divergence of the score network s_θ, an memory intensive computation. For computational feasibility, the Hutchinson trace estimator Hutchinson (1989) is used to estimate the divergence $\nabla \cdot s_\theta$, leading to noisy gradients and expensive forward and backward passes.

Denoising Score Matching. In practice, the ISM ELBO is not used for training, instead the DSM ELBO (Vincent, 2011; Song & Ermon, 2020; Huang et al., 2021) is used:

$$\log p_\theta(x) \geq \mathop{\mathbb{E}}_{q(\mathbf{y}_T \mid x)}[\log \pi_\theta(\mathbf{y}_T)] \quad + \quad (4)$$
$$\int_0^T \mathop{\mathbb{E}}_{q(\mathbf{y}_t \mid x)}\Big[\nabla_{\mathbf{y}_t} \cdot f - \frac{1}{2}\left\|s_\theta - s_q\right\|^2_{gg^\top} + \frac{1}{2}\left\|s_q\right\|^2_{gg^\top}\Big]dt$$

where s_q is the score of the transition kernel of the inference process, $s_q(t, \mathbf{y}_t) = \nabla_{\mathbf{y}_t} \log q(\mathbf{y}_t \mid x)$. To train a diffusion model with the DSM objective requires the following:

(**D1**) Samples from the transition kernel $q(\mathbf{y}_t \mid x)$

(**D2**) The score of the transition kernel, $\nabla_{\mathbf{y}_t} \log q(\mathbf{y}_t \mid x)$

In Singhal et al. (2023), the authors automate derivations for both **D1** and **D2** for linear processes, including for processes with auxiliary variables, such that the user is only required to specify the linear functions $f(\mathbf{y}, t), g(t)$.

However, no such automations exist for DSM training with nonlinear inference processes, as estimating the transition score for nonlinear processes requires solving high-dimensional partial differential equation (a version of the Fokker-Planck equation, see Lai et al. (2023)) for every forward pass, infeasible in high-dimensions.

Assumptions. We assume that the diffusion coefficient g is a function of t only, which can be either integrated on intervals $[s, t]$ analytically or numerically. We also assume that the drift f, the diffusion coefficient g and the initial condition q_{data} satisfy smoothness and integrability assumptions in appendix D, these assumptions guarantee that $q(\mathbf{y}_t), q(\mathbf{y}_t \mid \mathbf{y}_s)$ exist and are smooth and unique.

3. Automated DSM training for nonlinear diffusions

The approach we will take to make DSM tractable for nonlinear processes is to first derive a version of DSM that makes use of transitions $q(\mathbf{y}_t \mid \mathbf{y}_s)$, with s close to t, instead of transitions $q(\mathbf{y}_t \mid \mathbf{y}_0)$, and then showing how these transitions can be approximated fairly generally.

Local DSM. Suppose we are given a nonlinear diffusion process of the form eq. (1)

$$d\mathbf{y}_t = f(\mathbf{y}_t, t)dt + g(t)d\mathbf{w}_t$$

where the drift f is a function of $\mathbf{y}_t$ and t. Both the ISM and the DSM ELBOs are integrals of score matching terms:

$$\mathcal{L}_{\text{ISM}}(x,t) = \mathop{\mathbb{E}}_{q(\mathbf{y}_t \mid x)}\left[\frac{1}{2}\left\|s_\theta\right\|^2_{gg^\top} + \nabla_{\mathbf{y}_t} \cdot gg^\top s_\theta(\mathbf{y}_t, t)\right]$$
$$\mathcal{L}_{\text{DSM}}(x,t) = \mathop{\mathbb{E}}_{q(\mathbf{y}_t \mid x)}\Big[\frac{1}{2}\left\|s_\theta - \nabla_{\mathbf{y}_t}\log q(\mathbf{y}_t \mid x)\right\|^2_{gg^\top} - \frac{1}{2}\left\|\nabla_{\mathbf{y}_t}\log q(\mathbf{y}_t \mid x)\right\|^2_{gg^\top}\Big]$$

where $\mathcal{L}_{\text{DSM}}(x,t) = \mathcal{L}_{\text{ISM}}(x,t)$ (Huang et al., 2021; Song & Ermon, 2020). Now, as computing $q(\mathbf{y}_t \mid x)$ is computionally infeasible for arbitrary nonlinear inference processes, we show that we can use local transition kernels $q(\mathbf{y}_t \mid \mathbf{y}_s)$, where $0 < s < t$ instead of $q(\mathbf{y}_t \mid \mathbf{y}_0 = x)$, to define the local-DSM objective,

$$\mathcal{L}_{\text{L-DSM}}(x,t) = \mathop{\mathbb{E}}_{q(\mathbf{y}_t, \mathbf{y}_s \mid x)}\Big[\frac{1}{2}\left\|s_\theta - \nabla_{\mathbf{y}_t}\log q(\mathbf{y}_t \mid \mathbf{y}_s)\right\|^2_{gg^\top} - \frac{1}{2}\left\|\nabla_{\mathbf{y}_t}\log q(\mathbf{y}_t \mid \mathbf{y}_s)\right\|^2_{gg^\top}\Big].$$

In lemma 1, we show that $\mathcal{L}_{\text{L-DSM}}(x,t) = \mathcal{L}_{\text{ISM}}(x,t)$.

Lemma 1. *Let $q(\mathbf{y}_s \mid x), q(\mathbf{y}_t \mid \mathbf{y}_s)$ be the transition kernels of the process defined in eq. (1). For any $0 \leq s < t < T$, we have:*

$$\mathop{\mathbb{E}}_{q(\mathbf{y}_t \mid x)}\left[\frac{1}{2}\left\|s_\theta\right\|^2_{gg^\top} + \nabla_{\mathbf{y}_t} \cdot gg^\top s_\theta(\mathbf{y}_t, t)\right]$$
$$= \mathop{\mathbb{E}}_{q(\mathbf{y}_t, \mathbf{y}_s \mid x)}\left[\frac{1}{2}\left\|s_\theta\right\|^2_{gg^\top} + \nabla_{\mathbf{y}_t} \cdot gg^\top s_\theta(\mathbf{y}_t, t)\right]$$
$$= \mathop{\mathbb{E}}_{q(\mathbf{y}_t, \mathbf{y}_s \mid x)}\left[\frac{1}{2}\left\|s_\theta - \nabla_{\mathbf{y}_t}\log q(\mathbf{y}_t \mid \mathbf{y}_s)\right\|^2_{gg^\top}\right.$$

$$- \frac{1}{2} \left\| \nabla_{\mathbf{y}_t} \log q(\mathbf{y}_t \mid \mathbf{y}_s) \right\|^2_{gg^\top} \Big]. \tag{5}$$

where $q(\mathbf{y}_t, \mathbf{y}_s \mid x) = q(\mathbf{y}_t \mid \mathbf{y}_s) q(\mathbf{y}_s \mid x)$.

For a proof, see appendix A. Note that in eq. (5), while we still require samples $\mathbf{y}_s \sim q(\mathbf{y}_t \mid x)$, we only require the score of the transition kernel $q(\mathbf{y}_t \mid \mathbf{y}_s)$, where the choice of s is up to the user.

For a given time t, we define a *schedule* $s(t)$ as a function which satisfies $0 \leq s(t) < t$ for all $t \in (0, T]$. Using the schedule $s(t)$ and lemma 1 allows us to write the ELBO using local increments $q(\mathbf{y}_t \mid \mathbf{y}_s)$, instead of using the score of the transition kernel $q(\mathbf{y}_t \mid \mathbf{y}_0)$.

Theorem 1. *Let $q(\mathbf{y}_t \mid \mathbf{y}_s)$ be the transition kernel of the process in eq. (1) and $s(t)$ be a schedule, which satisfies $0 \leq s(t) < t$ for all $t \in (0, T]$. Then for a model process $\mathbf{z}_t$ defined in eq. (2), we can lower bound the model log-likelihood as follows:*

$$\begin{aligned} \log p_\theta(x) \geq & \mathop{\mathbb{E}}_{q(\mathbf{y}_T \mid x)} \left[\log \pi_\theta(\mathbf{y}_T) \right] \\ & + \int_0^T \mathop{\mathbb{E}}_{q(\mathbf{y}_t, \mathbf{y}_s \mid x)} \Big[\nabla_{\mathbf{y}_t} \cdot f(\mathbf{y}_t, t) \\ & \quad - \frac{1}{2} \left\| s_\theta - \nabla_{\mathbf{y}_t} \log q(\mathbf{y}_t \mid \mathbf{y}_s) \right\|^2_{gg^\top} \\ & \quad + \frac{1}{2} \left\| \nabla_{\mathbf{y}_t} \log q(\mathbf{y}_t \mid \mathbf{y}_s) \right\|^2_{gg^\top} dt \Big] \end{aligned} \tag{6}$$

where $s = s(t)$ and $q(\mathbf{y}_t, \mathbf{y}_s \mid x) = q(\mathbf{y}_t \mid \mathbf{y}_s) q(\mathbf{y}_s \mid x)$ due to the Markov property.

For a proof, see appendix A. Although, the local-DSM ELBO holds for arbitrary pairs t, s, estimating the score of the transition kernel $q(\mathbf{y}_t \mid \mathbf{y}_s)$ where $s > 0$ is still not feasible for nonlinear drifts.

In the next section, we show how the transition kernel $q(\mathbf{y}_t \mid \mathbf{y}_s)$ is well approximated using local linearization techniques.

Local Linearization. The idea is to define a *locally linear* diffusion process on the interval $(s, T]$ with a linearized drift f, using an operator $\mathcal{T}_s$ such that the function $\mathcal{T}_s f$ is a linear in $\mathbf{y}_t, t$. Since the process is linear, the transition kernel $\widehat{q}(\hat{\mathbf{y}}_t \mid \mathbf{y}_s)$ is Gaussian with mean and covariance characterized by solutions to ordinary differential equations (ODEs) (Särkkä & Solin, 2019).

Suppose we are given a sample $\mathbf{y}_s$ at time s, then for $t > s$ we define a locally linear diffusion process

$$d\mathbf{y}_t = (\mathcal{T}_s f)(\mathbf{y}_t, t) dt + g(t) d\mathbf{w}_t, t \in (s, T]. \tag{7}$$

We have several choices for the operator $\mathcal{T}_s$ (Ozaki, 1993; 1992), see section 9.3 in Särkkä & Solin (2019) for examples. In this work, we study two examples of the operator $\mathcal{T}_s$, first $\mathcal{T}_{\mathbf{y}_s, s}$ which is a first-order Taylor expansion of the drift drift $f(\mathbf{y}_t, t)$ around $(\mathbf{y}_s, s)$ and second $\mathcal{T}_{\mathbf{y}_s, t}$ a first-order Taylor expansion around $(\mathbf{y}_s, t)$. For ease of exposition, we discuss the first operator:

$$\begin{aligned} (\mathcal{T}_{\mathbf{y}_s, s} f)(\mathbf{y}_t, t) &= f(\mathbf{y}_s, s) + \nabla_s f(\mathbf{y}_s, s)(t - s) \\ &\quad + \nabla_{\mathbf{y}_s} f(\mathbf{y}_s, s)(\mathbf{y}_t - \mathbf{y}_s) \\ &= \Big(f(\mathbf{y}_s, s) + \nabla_s f(\mathbf{y}_s, s)(t - s) + \nabla_{\mathbf{y}_s} f(\mathbf{y}_s, s) \mathbf{y}_s \Big) \\ &\quad + \nabla_{\mathbf{y}_s} f(\mathbf{y}_s, s) \mathbf{y}_t \end{aligned} \tag{8}$$

$$:= \mathbf{c}_t + \mathbf{A}_t \mathbf{y}_t \tag{9}$$

The main idea is that the drift of the locally linear process in eq. (7) can be expressed as an affine function $(\mathcal{T}_s f)(\mathbf{y}_t, t) = \mathbf{c}_t + \mathbf{A}_t \mathbf{y}_t$, where $\mathbf{c}_t \in \mathbf{R}^d$ and $\mathbf{A}_t \in \mathbf{R}^{d \times d}$. For processes with affine drifts and spatially invariant diffusion coefficient ($g(t, \mathbf{y}) = g(t)$), the transition kernel $q(\mathbf{y}_t \mid \mathbf{y}_s)$ is Gaussian (see section 6.1 in Särkkä & Solin (2019)), therefore we only need to compute the mean and covariance of the locally linear process.

Next, we present how to compute the mean and covariance and then show how we can apply these ideas to the locally-linear approximations of nonlinear drifts. We provide all derivations in appendix C including those for the second Taylor expansion around $(\mathbf{y}_s, t)$. In this expansion, the matrix $\mathbf{A}$ is a function of time t.

Mean and Covariance Equations. For linear processes with drift $f(\mathbf{y}_t, t) = \mathbf{c}_t + \mathbf{A}_t \mathbf{y}_t$ and diffusion co-efficient $g(t)$, the mean and covariance are solutions to the following ODEs:

$$\frac{d}{dt} \mathbf{m}_{t|s} = \mathbf{c}_t + \mathbf{A}_t \mathbf{m}_{t|s} \tag{10}$$

$$\frac{d}{dt} \mathbf{P}_{t|s} = \mathbf{A}_t \mathbf{P}_{t|s} + \mathbf{P}_{t|s} \mathbf{A}_t^\top + g g^\top(t) \tag{11}$$

where $\mathbf{m}_{s|s} = \mathbf{y}_s$ and $\mathbf{P}_{s|s} = 0$. The solutions to eqs. (10) and (11) can be expressed as integrals:

$$\mathbf{m}_{t|s} = \exp\left[\int_s^t \mathbf{A}_\tau d\tau \right] \mathbf{y}_s + \int_s^t \exp[\mathbf{A}_{t-\tau}] \mathbf{c}_\tau d\tau \tag{12}$$

$$\mathbf{P}_{t|s} = \int_s^t \exp[\mathbf{A}_{t-\tau}] g g^\top(\tau) \exp[\mathbf{A}_{t-\tau}^\top] d\tau \tag{13}$$

See appendix C for derivations. Both the mean and covariance ODE solutions require integrating matrix exponentials, which are not amenable to easy manipulation and require specific derivations for each inference process, for instance see pages 50-54 in Dockhorn et al. (2021).

In the next section, for any choice of the drift f and diffusion coefficient g, we derive a solution to the mean ODE in eq. (10) and the covariance ODE, using matrix exponentials, for the Taylor expansion operator around $(\mathbf{y}_s, s)$ that only involves integrating the diffusion co-efficient g.

Algorithm 1 Sampling and score estimation

Input: Inference process q, time t, scheduler $s(t)$, and data x
Output: Samples $q(\mathbf{y}_t, \mathbf{y}_s \mid x)$ and score estimate $\nabla_{\mathbf{y}_t} \log q(\mathbf{y}_t \mid \mathbf{y}_s)$
Sample $\mathbf{y}_s$ by numerically integrating eq. (1)
Compute $\mathbf{m}_{t|s}, \sigma_{t|s}$, solutions to eqs. (10) and (11) respectively.
Sample $\varepsilon \sim \mathcal{N}(0, I_d)$ and then let:

$$\mathbf{y}_t = \mathbf{m}_{t \mid s} + \sigma(t|s)\varepsilon$$
$$\nabla_{\mathbf{y}_t} \log \widehat{q}(\mathbf{y}_t \mid \mathbf{y}_s) = -\sigma_{t|s}^{-1}\varepsilon$$

Return: $\mathbf{y}_t, \mathbf{y}_s$ and score estimate $\nabla_{\mathbf{y}_t} \log \widehat{q}(\mathbf{y}_t \mid \mathbf{y}_s)$

Mean and Covariance Estimation. Singhal et al. (2023) use a matrix factorization technique (see section 6.2 in Särkkä & Solin (2019)) to automate solving differential equations like in eqs. (10) and (11) using matrix exponentials.

The idea is that equations of the form eq. (11) can be solved using the matrix factorization $\mathbf{P}_{t|s} = \mathbf{C}_t \mathbf{H}_t^{-1}$, where $\mathbf{C}_t, \mathbf{H}_t$ evolve as follows:

$$\begin{pmatrix} \frac{d}{dt}\mathbf{C}_t \\ \frac{d}{dt}\mathbf{H}_t \end{pmatrix} = \begin{pmatrix} \mathbf{A}_t & gg^\top(t) \\ \mathbf{0} & -\mathbf{A}^\top(t) \end{pmatrix} \begin{pmatrix} \mathbf{C}_t \\ \mathbf{H}_t \end{pmatrix} \tag{14}$$

which can be solved by matrix factorization and scalar integration of $\mathbf{A}_\tau$ and $gg_\tau^\top$ on the interval $[s, t]$:

$$\begin{pmatrix} \mathbf{C}_t \\ \mathbf{H}_t \end{pmatrix} = \exp \begin{pmatrix} [\mathbf{A}_\tau]_s^t & [gg^\top(\tau)]_s^t \\ \mathbf{0} & -[\mathbf{A}_\tau^\top]_s^t \end{pmatrix} \begin{pmatrix} \mathbf{0} \\ \mathbf{I} \end{pmatrix} \tag{15}$$

where $[\mathbf{A}_\tau]_s^t := \int_s^t \mathbf{A}_\tau d\tau$. Since $\mathbf{A}_t$ is defined to be homogeneous, we do not have to integrate $\mathbf{A}$, while g can be time in-homogeneous.

We can solve the mean ODE in eq. (10) for the Taylor expansion around $(\mathbf{y}_s, s)$. The matrix $\mathbf{A}$ is time-homogeneous and the function $\mathbf{c}$ can be separated into a time-varying and time-homogeneous part, $\mathbf{c}_t = \mathbf{c}_1 + \mathbf{c}_2 t$. We can solve this affine ODE exactly:

$$\mathbf{m}_{t|s} = \exp\left[\int_s^t \mathbf{A}_\tau d\tau\right] \mathbf{y}_s + \int_s^t \exp[\mathbf{A}_{t-\tau}]\mathbf{c}_\tau d\tau$$
$$\begin{aligned} \mathbf{m}_{t \mid s} = &\exp((t-s)\mathbf{A}) + (\exp((t-s)\mathbf{A}) - I)\mathbf{A}^{-1}\mathbf{c}_1 \\ &+ \exp((t-s)\mathbf{A})\left[s\mathbf{A}^{-1} + \mathbf{A}^{-2}\right]\mathbf{c}_2 \\ &- \left[t\mathbf{A}^{-1} + \mathbf{A}^{-2}\right] c_2 \end{aligned}$$

For complete derivations, see appendix C.1.

Now, given a sample $\mathbf{y}_s$ at time s, we can sample from the locally linear process $q(\mathbf{y}_t \mid \mathbf{y}_s)$ as follows:

$$\mathbf{y}_t = \mathbf{m}_{t \mid s} + \sigma_{t \mid s}\varepsilon \tag{16}$$

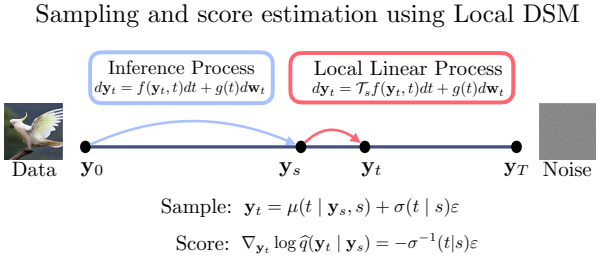

Figure 1: **Training with Automated DSM**: Given a nonlinear inference process q and a time t with sample $\mathbf{y}_0 = x$, we use a numerical sampler till time $s(t)$ and then use the locally linear process for sampling $\mathbf{y}_t \mid \mathbf{y}_s$ and estimating the transition score.

where $\varepsilon \sim \mathcal{N}(0, \mathbf{I}_d)$ and $\sigma_{t|s}$ is the matrix square root of $\mathbf{P}_{t|s}$ and $\sigma_{t|s}^{-1}$ is the inverse of the matrix square root, similar to the transition score computation defined for multivariate diffusion model (MDM) processes in Singhal et al. (2023). We can estimate the score of the transition kernel $q(\mathbf{y}_t \mid \mathbf{y}_s)$ at a sample from Equation (16) as

$$\nabla_{\mathbf{y}_t} \log \widehat{q}(\mathbf{y}_t \mid \mathbf{y}_s) = -\sigma_{t|s}^{-1}\varepsilon. \tag{17}$$

Algorithms. Making use of the local linearization and the automated mean and covariance derivations, we provide algorithms for automated training with nonlinear inference processes called automated DSM. In Algorithm 1 we show how to sample from $\widehat{q}(\mathbf{y}_t \mid \mathbf{y}_s)$ and computing its transition score. Finally, in algorithm 2, we present the automated DSM algorithm, where for a given score network s_θ and sample x, we return an estimate of the local-DSM ELBO. See fig. 1 for an overview of the local DSM training pipeline.

Now, despite having access to a tractable score approximation, we note that a first-order Taylor approximation introduces errors in the estimate of the score, specifically when the gap between s, t is large. In the next section, we discuss methods to control the approximation error, particularly by tailoring a schedule to control the Taylor approximation error.

Controlling the Taylor Error with Scheduled Pairs. Suppose $\mathbf{y}_t$ is the variance-preserving stochastic differential equation (VPSDE) process (Song et al., 2020b):

$$d\mathbf{y}_t = -\frac{1}{2}\beta_t \mathbf{y}_t + \sqrt{\beta_t} d\mathbf{w}_t \tag{18}$$

Then the mean and covariance are:

$$\mathbf{m}_{t|s} = \exp\left(-\frac{1}{2}[\beta_\tau]_s^t\right)\mathbf{y}_s, \quad \mathbf{P}_{t|s} = 1 - \exp\left(-[\beta_\tau]_s^t\right),$$

Algorithm 2 Automated DSM: estimating local-DSM ELBO

Input: Inference process q, model prior π_θ, score network architecture $s_\theta(\mathbf{y}_t, t)$, scheduler $s(t)$, and data x
Return: Differentiable Local DSM ELBO estimate
Sample $t \sim \text{Uniform}[0, T]$
Use algorithm 1 to get samples $q(\mathbf{y}_t, \mathbf{y}_s \mid x)$ and score estimate $\nabla_{\mathbf{y}} \log \widehat{q}(\mathbf{y}_t \mid \mathbf{y}_s)$
Compute

$$\mathcal{L}(x, \theta) = \frac{1}{2} \left\| s_\theta - \nabla_{\mathbf{y}} \log \widehat{q}(\mathbf{y}_t \mid \mathbf{y}_s) \right\|^2_{gg^\top} - \frac{1}{2} \left\| \nabla_{\mathbf{y}_t} \log \widehat{q}(\mathbf{y}_t \mid \mathbf{y}_s) \right\|^2_{gg^\top} - \nabla \cdot f(\mathbf{y}_t, t)$$

Sample $\mathbf{y}_T$ by numerical integration.
Output: $-T\mathcal{L} + \log \pi_\theta(\mathbf{y}_T)$

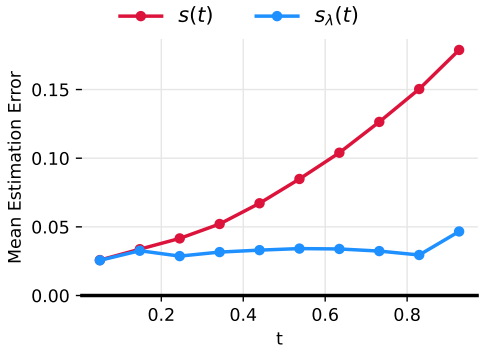

Figure 2: **Local mean $\mathbf{m}_{t|s}$ Estimation Error**: we compare the estimation error when using the schedule $s_\lambda(t)$ with versus $s(t) = t - 0.05$. We note that using $s(t)$ instead of $s_\lambda(t)$ leads to higher error.

where $[\beta_\tau]_s^t = \int_s^t \beta_\tau d\tau$. The difference between the distributions $q(\mathbf{y}_t), q(\mathbf{y}_s)$ is therefore controlled by the integral $[\beta_\tau]_s^t$. The gap can be made large or small depending on the values taken by β_t in $[s, t]$ not on the length, of the interval. For instance, if $\beta_t = 0.1 + 10t$, then the gap between $q(\mathbf{y}_t)$ and $q(\mathbf{y}_{t-\ell})$ is larger for larger t values. Therefore, to control the change between $q(\mathbf{y}_t)$ and $q(\mathbf{y}_s)$, we propose the following heuristic: choose pairs (s, t) based on the integrals of the form $\int_s^t gg^\top(\tau)d\tau$ rather than a fixed gap $s(t) = t - \ell$ in time for a constant value ℓ.

To control the error introduced by local linearization, we define *scheduled pairs* (s, t) so that for all $\forall t > t_{\text{min}} > 0$, for a given $g(t)$ we define $s_\lambda(t)$ such that the integral $\int_s^t g^2(\tau)d\tau$ is equal to a constant λ and for $0 < t \leq t_{\text{min}}$, we set $s_\lambda(t) = 0$. We provide a derivation for $s_\lambda(t)$ for commonly used g functions in appendix F. In case, g cannot be expressed as $gg_t^\top = g^2(t)\mathbf{I}_d$ where $g^2(t)$ is a scalar, we can select s_λ such that $\max_{i,j} \int_s^t [gg^\top]_{i,j}(\tau)d\tau = \lambda$.

In fig. 2, we estimate the mean of the local transition kernel for the diffusion process:

$$d\mathbf{y}_t = \beta_t \nabla_{\mathbf{y}} \log \pi_\theta(\mathbf{y}_t)dt + \sqrt{2\beta_t} d\mathbf{w}_t,$$

with $\beta_t = 0.1 + 9.9t$ and model prior $\pi_\theta = \frac{1}{2}\mathcal{N}(-1, \frac{1}{2}) + \frac{1}{2}\mathcal{N}(1, \frac{1}{2})$. We observe that the error in estimating $\mathbf{m}_{t|s}, \sigma^2_{t|s}$ is constant for the scheduler $s_\lambda(t)$ with $\lambda = 0.05$ versus exploding for $s(t) = t - 0.05$. Here we use x sampled from the two-dimensional checkerboard distribution, see fig. 3.

Bounds on the error from Taylor expansion. As noted in the previous section, Taylor expansions of the drift can introduce error. In lemma 2 in appendix E, we show that the approximation error between the true marginal density $q(\mathbf{y}_t)$ and the locally linear approximation $\widehat{q}(\mathbf{y}_t) = \mathbb{E}_{q(\mathbf{y}_s)}[\widehat{q}(\mathbf{y}_t \mid \mathbf{y}_s)]$ can be controlled by the difference of the drifts f and the Taylor approximation $\mathcal{T}_s f$ on the interval $[s(t), t]$ by upper bounding the KL-divergence. This lemma controls the error between distributions of the exact and approximate process in terms of the error from the Taylor approximation.

3.1. Extensions

In this section, we present extensions of the local DSM ELBO. First, we present a perceptually weighted version of the local DSM ELBO, typically used for image-modeling. Next, we present a version of the local DSM ELBO for use in score modeling (Boffi & Vanden-Eijnden, 2023b;a; Lu et al., 2023) in the sciences, where the object of interest is the score of a nonlinear diffusion process and not maximizing the likelihood of a data distribution. The score of the diffusion process is used to study properties of the process such as the entropy, entropy production rate and the density itself (Otsubo et al., 2022; Boffi & Vanden-Eijnden, 2023b).

Perceptual Weighting. In practice, the DSM loss is often re-weighted to give uniform weight to each t (Song & Ermon, 2020; Ho et al., 2020). To apply this idea in our case, we can observe that $\nabla \log q(\mathbf{y}_t \mid \mathbf{y}_s) = -\sigma^{-1}_{t|s}\epsilon$, parameterize the model as $s_\theta(\mathbf{y}_t, t) = \gamma^{-1}(t, s)\epsilon_\theta(\mathbf{y}_t, t)$ and multiply the integrand in eq. (6) by $\sigma^2_{t|s}$:

$$\sigma^2_{t|s} \left\| s_\theta - \nabla \log \widehat{q}(\mathbf{y}_t \mid \mathbf{y}_s) \right\|^2_{gg^\top} = \left\| \frac{\sigma_{t|s}}{\gamma(t, s)} \epsilon_\theta(y_t, t) - \epsilon \right\|^2_{gg^\top} \tag{19}$$

where we choose γ so that $\sigma_{t|s}/\gamma(t, s) \approx 1$. In our generative modeling experiments, we choose $\gamma^2(t, s) =$

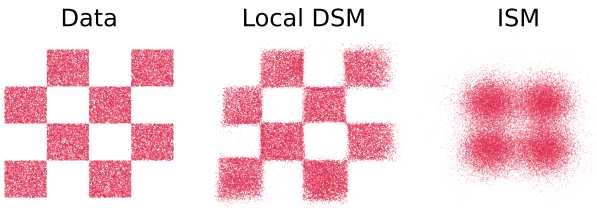

Figure 3: **ISM v local-DSM:** Samples from a local-DSM trained model in the middle panel, and samples from an ISM trained model on the right panel. Both models were trained for 20k gradient steps, however the local-DSM trained model has better sample quality.

$1 - \exp(-2\int_s^t \beta_\tau d\tau)$ for inference processes where the drift takes the form $f(\mathbf{y}, t) = \beta_t h(\mathbf{y})$.

Score Modeling. For processes studied in statistical physics, biology, etc, learning the score model is of primary interest. In such instances, we can optimize the denoising score matching term in local-DSM:

$$\int_0^T \mathop{\mathbb{E}}_{\widehat{q}(\mathbf{y}_t, \mathbf{y}_s \mid x)} \|s_\theta - \nabla_{\mathbf{y}_t} \log q(\mathbf{y}_t \mid \mathbf{y}_s)\|^2_{gg^\top} \, dt \qquad (20)$$

using the automated derivations in this work.

4. Experiments

We test the local-DSM objective for training DBGMs on a challenging low-dimensional example, CIFAR10 and learning the score for coupled equilibrium and non-equilibrium diffusion processes studied in (Boffi & Vanden-Eijnden, 2023b).

For all experiments, we chose the scheduler $s_\lambda(t)$ with $\lambda = 10^{-2}$, unless otherwise stated.

The integrand in the ELBO defined in eq. (6) is unbounded at $t = 0$ and is numerically unstable for small values of t. Therefore, we estimate the integral on an interval $(\delta, T]$ where $\delta = 10^{-3}$. Truncating the ELBO biases the estimate. Sohl-Dickstein et al. (2015); Song & Ermon (2019) use a variational lower bound to derive a valid ELBO. We derive a valid ELBO with truncation in appendix B and report bits-per-dims (BPDs) using the valid ELBO.

For sampling from the forward process, we use an adaptive solver (Lamba, 2003) in all experiments. For the generative modeling experiments we use the Taylor operator that expands around $(\mathbf{y}_s, t)$, while for the score modeling for non-equilibrium stochastic dynamics we use the Taylor expansion around $(\mathbf{y}_s, s)$.

For the generative modeling experiments, we use a Langevin diffusion process with the model prior as its stationary distribution:

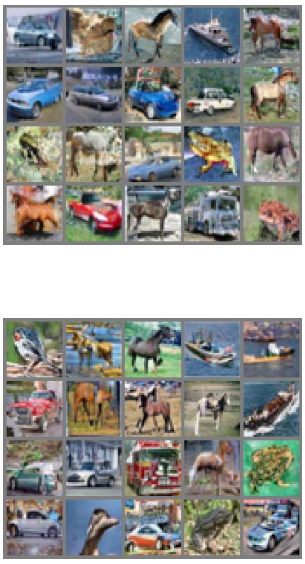

Figure 4: **CIFAR10 samples from DBGMs trained using nonlinear inference processes.** Sample from the MoG (top) and Logistic prior (bottom) DBGMs.

$$d\mathbf{y}_t = \beta(t)\nabla_{\mathbf{y}} \log \pi_\theta(\mathbf{y}_t)dt + \sqrt{2\beta(t)}d\mathbf{w}_t, \qquad (21)$$

with $\beta(t) = \beta_0 + t(\beta_1 - \beta_0)$ and $\beta_0 = 0.1$ and $\beta_1 = 10$ and the approximation $\mathcal{T}_{\mathbf{y}_s, t}$. We parameterize the score model as $s_\theta(t, \mathbf{y}_t) = -\gamma_{t|s}\varepsilon(t, \mathbf{y}_t)$, where $\gamma^2_{t|s} = 1 - \exp(-2\int_s^t \beta(\tau)d\tau)$. For the science experiments, we parameterize s_θ as feedforward neural networks, see the experiments for a description.

Local DSM vs ISM. In this experiment, we show that using the local-DSM objective leads to faster convergence compared to using the ISM objective on synthetic 2d.

As a low-dimensional example, we train the models on the two-dimensional checkerboard density. We use a three layer feed-forward network with width 256 and with the ReLU activation (Nair & Hinton, 2010) as the ε_θ model. We train two models using the local-DSM and ISM ELBOs with a Logistic distribution as π_θ in eq. (21).

We train both models with a batch size of 1024 for $20,000$ gradient steps using the AdamW optimizer (Loshchilov & Hutter, 2017). Figure 3 shows that using the local-DSM ELBO leads to significantly faster convergence even on a low-dimensional synthetic dataset.

Image Modeling with Non-Gaussian Priors. Next, we train diffusion models on the CIFAR10 dataset, with a

Langevin inference process using a non-Gaussian prior as defined in eq. (21).

Prior π_θ	Objective	ISM BPD
Logistic	local-DSM ELBO	$\leq 3.568 \pm 0.07$
Logistic	local-DSM (PW)	$\leq 3.561 \pm 0.09$
Logistic	ISM ELBO	$\leq 3.741 \pm 0.09$
MoG	local-DSM ELBO	$\leq 3.496 \pm 0.11$
MoG	local-DSM (PW)	$\leq 3.503 \pm 0.151$
MoG	ISM ELBO	$\leq 3.637 \pm 0.14$

Table 1: **BPDs on CIFAR-10**: We compare models trained using nonlinear inference processes via the ISM and the local-DSM objectives, both the ELBO and the perceptually-weighted (PW) versions. For the same amount of compute, the local-DSM trained models achieve significantly better BPDs. *A lower* BPD *is better.*

For the model prior, we choose (a) a mixture of Gaussians (MoG) $\pi_\theta(\mathbf{y}) = \frac{1}{2}\mathcal{N}(-\frac{1}{2}, \frac{1}{2}) + \frac{1}{2}\mathcal{N}(\frac{1}{2}, \frac{1}{2})$ and (b) a Logisitic distribution $\pi_\theta = \frac{\exp(-x)}{(1+\exp(-x))^2}$. Similar to the previous experiment, the score network is a U-Net from Ho et al. (2020). We train using the perceptual weighted objective defined in eq. (19), the local-DSM and the ISM ELBOs. For all models we use the noise parameterization for the score model.

In table 1 we compare the bits-per-dim (BPDs, Van Den Oord et al. (2016); Song et al. (2020b); Huang et al. (2021)) of models trained using the local-DSM ELBO, perceptual loss and the ISM ELBO. Table 1 shows that given the same amount of compute, the local-DSM trained models get better BPD upper-bounds. In fig. 4, we show samples generated using models trained with the perceptually-weighted loss introduced in eq. (19) for the tailored scheduler $s_\lambda(t)$.

Prior π_θ	ISM BPD	λ	local-DSM BPD
Logistic	$\leq 3.568 \pm 0.07$	0.01	$\leq 3.566 \pm 0.097$
Logistic		0.02	$\leq 3.530 \pm 0.084$
Logistic		0.05	$\leq 3.422 \pm 0.096$
MoG	$\leq 3.496 \pm 0.11$	0.01	$\leq 3.465 \pm 0.1242$
MoG		0.02	$\leq 3.434 \pm 0.1419$
MoG		0.05	$\leq 3.354 \pm 0.1879$

Table 2: **Increasing λ in the scheduled pair** $s_\lambda(t)$. Using the scheduler $s_\lambda(t)$ with varying values of λ, we see increasing the gap between $\mathbf{y}_t$ and $\mathbf{y}_s$ leads to a growing gap between the unbiased ISM objective and the local-DSM objective.

Do the ISM and Local DSM ELBOs match? The local-DSM objective makes use of two approximations, the local transition score and numerical sampling, while the ISM objective only requires numerical sampling. In table 2, we show that using the constant scheduler s_λ for training and parameterization leads to models where the unbiased ISM and local-DSM BPDs have similar estimates for smaller values of λ, and the approximation error increases as λ increases.

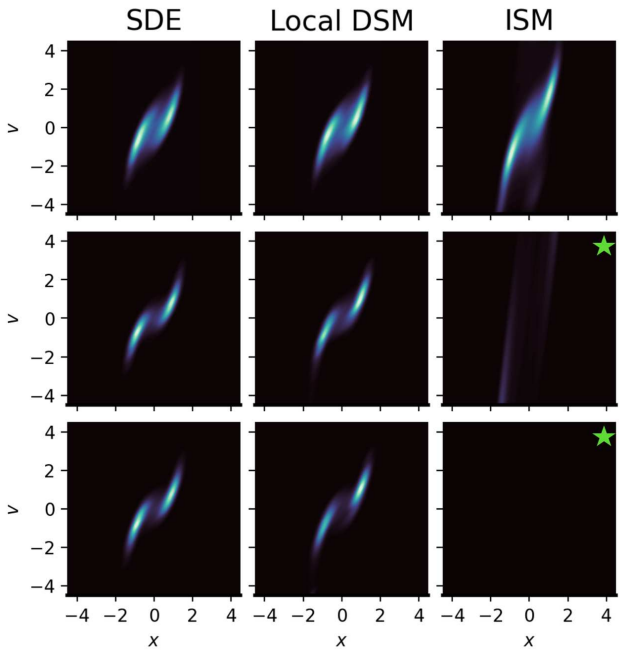

Figure 5: **Samples at** $t \in \{1, 3, 5\}$. Here we compare samples from the process defined in eq. (22) on the left panel, and local-DSM and ISM trained model samples in the middle and right panels. The inference process and local-DSM trained model samples are near identical. ⋆ We note that ISM trained model samples quality did not match the inference process' samples and diverged, see fig. 7 for ISM model samples.

Score Modeling for Non-Equilibrium Stochastic Dynamics. In this experiment, we study a nonlinear system $\mathbf{y} = (x, v)^\top$, described in Tailleur & Cates (2008); Boffi & Vanden-Eijnden (2023b) as

$$dx = (-x^3 + v)dt, \quad dv = -\gamma v dt + \sqrt{2\gamma D} d\mathbf{w}_t \tag{22}$$

for $t \in [0, T]$ and where $\gamma = 0.1, D = 1.0$ and $T = 5.0$ with initial conditions $x_0, v_0 \sim \mathcal{N}(0, 1)$. The system of equations described in eq. (22) does not have a stationary distribution but does exhibit a non-equilibrium statistical steady state (Boffi & Vanden-Eijnden, 2023b).

Figure 5 shows samples from the *probability flow* ODE (ODE) (Song et al., 2020b):

$$\frac{d}{dt}\mathbf{y}_t = f(\mathbf{y}_t, t) - \frac{1}{2} g g^\top(t) s_\theta(\mathbf{y}_t, t), \tag{23}$$

at different times $t \in \{1, 3, 5\}$. The PF-ODE defined in eq. (23) simulates the inference process in forward time, such that $q_{\text{ode}}(\mathbf{y}_t) = q_{\text{SDE}}(\mathbf{y}_t)$ when the score model s_θ matches the actual score of the inference SDE: $\nabla_{\mathbf{y}} \log q_{\text{SDE}}(\mathbf{y}_t)$.

We parameterize the score model s_θ as 3 layer feed-forward network with width 256. Following Boffi & Vanden-Eijnden (2023b), we enforce that the score model is anti-symmetric $s_\theta(t, x, v) = s_\theta(t, -x, -v)$ since the drift f is anti-symmetric. We train both the local-DSM and ISM models for 200,000 gradient steps with a batch size of 1024.

Figure 5 compares samples from a local-DSM trained model versus samples from the ISM trained model against samples from the inference process defined in eq. (22). The samples produced by the local-DSM trained model and the inference process distribution are near identical, the ISM trained model samples diverge, see fig. 7 for the ISM samples. For a quantitative comparison, in fig. 9 in appendix G.1, we compare the maximum mean discrepancy (MMD) distance (Smola et al., 2006) between the model generated samples and the inference process' samples. We observe that the ISM model's sample quality deteriorates very rapidly compared to the sample quality of local DSM trained models.

Score Modeling for Interacting Particle Systems. In this experiment, following Boffi & Vanden-Eijnden (2023b), we consider a system of $N = 5$ particles $\mathbf{y}_t^{(i)} \in \mathbf{R}^2$ for $t \in [0, 10]$, which evolve as :

$$d\mathbf{y}_t^{(i)} = 4B(\beta_t - \mathbf{y}_t^{(i)}) \left\| \mathbf{y}_t^{(i)} - \beta_t \right\|_2^2 dt + \frac{A}{Nr^2} \sum_{j=1}^{N} (\mathbf{y}_t^{(i)} - \mathbf{y}_t^{(j)}) \exp\left(-\frac{2}{2r^2} \left\| \mathbf{y}_t^{(i)} - \mathbf{y}_t^{(j)} \right\|_2^2 \right) dt + \sqrt{2D} d\mathbf{w}_t^{(i)} \tag{24}$$

where $A = 10, r = 0.5, a = 2, \omega = 1, D = 0.25, B = D/R^2, \gamma = 5, R = \sqrt{\gamma N} r, \beta(t) = a(\cos \pi\omega t, \sin \pi\omega t)$ and $\mathbf{y}_0^{(i)} \sim \mathcal{N}(0, \sigma_0^2 I_d)$ with $\sigma_0 = 0.5$. We train with the local DSM and ISM objectives. We train both models with a batch size of 1024 for $10,000$ gradient steps using AdamW. We use a three-layer feedforward network with a hidden size of 256.

In fig. 6, we plot the variance of the components of the first particle $\mathbf{y}_t^{(1)}$ for $t \in [0, 10]$. We plot the variance of the samples generated using the process in eq. (24) as well as samples from the PF-ODE for local DSM and ISM trained models. In fig. 8 in appendix G.2, we plot the MMD (Smola et al., 2006) of the local DSM and ISM samples compared to the diffusion process samples. Both comparison show that the local DSM trained model samples are more faithful to the diffusion process compared to the ISM trained model.

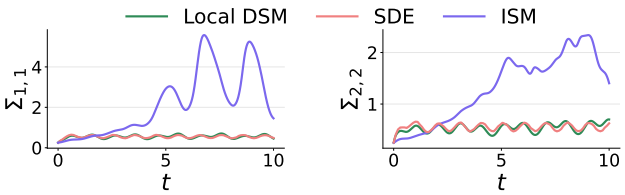

Figure 6: **Sample variance at** $t \in [0, 10]$. Here we plot the variance of the individual components of the first particle $\mathbf{y}_t^{(1)}$ simulated using the diffusion process defined in eq. (24) (SDE) and the local DSM and ISM PF-ODE. We observe that the local DSM trained model is more faithful to the ground truth compared to the ISM trained model.

5. Discussion

This work presents algorithms for training diffusion-based generative modeling with nonlinear inference processes. First, we introduce the local-DSM variational lower bound that is amenable to approximations where computation can be automated. We show how to build approximations using locally linear processes and derive automated approaches to compute the transition score function needed in the local-DSM objective. To control the error introduced in the locally linear approximation, we design pairs $(s(t), t)$ such that the estimation error remains well-behaved for larger values of t. The experiments show that using the local-DSM objective leads to faster training and has better sample quality compared to ISM, for generative modeling as well as score estimation for physical systems. This work advances the computational frontier for working with nonlinear inference processes.

Impact Statement

Diffusion models can be used to generate high-resolution realistic images, we along with other researchers in the field take seriously that we should monitor the data used to train these models along with what they are used for.

Acknowledgements

This work was partly supported by the NIH/NHLBI Award R01HL148248, NSF Award 1922658 NRT-HDR: FUTURE Foundations, Translation, and Responsibility for Data Science, NSF CAREER Award 2145542, ONR N00014-23-1-2634, and Apple.

References

Albergo, M. S. and Vanden-Eijnden, E. Building normalizing flows with stochastic interpolants. *arXiv preprint arXiv:2209.15571*, 2022.

Albergo, M. S., Boffi, N. M., and Vanden-Eijnden, E.

Stochastic interpolants: A unifying framework for flows and diffusions. *arXiv preprint arXiv:2303.08797*, 2023.

Bartosh, G., Vetrov, D., and Naesseth, C. A. Neural flow diffusion models: Learnable forward process for improved diffusion modelling. *arXiv preprint arXiv:2404.12940*, 2024.

Baydin, A. G., Pearlmutter, B. A., Radul, A. A., and Siskind, J. M. Automatic differentiation in machine learning: a survey. *Journal of Marchine Learning Research*, 18:1–43, 2018.

Boffi, N. M. and Vanden-Eijnden, E. Deep learning probability flows and entropy production rates in active matter. *arXiv preprint arXiv:2309.12991*, 2023a.

Boffi, N. M. and Vanden-Eijnden, E. Probability flow solution of the fokker–planck equation. *Machine Learning: Science and Technology*, 4(3):035012, 2023b.

Chandler, D. Introduction to modern statistical. *Mechanics. Oxford University Press, Oxford, UK*, 5:449, 1987.

Chen, T. On the importance of noise scheduling for diffusion models. *arXiv preprint arXiv:2301.10972*, 2023.

Dhariwal, P. and Nichol, A. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34, 2021.

Dockhorn, T., Vahdat, A., and Kreis, K. Score-based generative modeling with critically-damped langevin diffusion. *arXiv preprint arXiv:2112.07068*, 2021.

Doucet, A., Grathwohl, W., Matthews, A. G., and Strathmann, H. Score-based diffusion meets annealed importance sampling. *Advances in Neural Information Processing Systems*, 35:21482–21494, 2022.

Du, W., Zhang, H., Yang, T., and Du, Y. A flexible diffusion model. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 8678–8696. PMLR, 2023. URL https://proceedings.mlr.press/v202/du23g.html.

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

Fleming, W. H. Diffusion processes in population biology. *Advances in Applied Probability*, 7:100–105, 1975.

Grathwohl, W., Chen, R. T., Bettencourt, J., Sutskever, I., and Duvenaud, D. Ffjord: Free-form continuous dynamics for scalable reversible generative models. *arXiv preprint arXiv:1810.01367*, 2018.

Haussmann, U. G. and Pardoux, E. Time reversal of diffusions. *The Annals of Probability*, pp. 1188–1205, 1986.

Ho, J., Jain, A., and Abbeel, P. Denoising diffusion probabilistic models. *arXiv preprint arXiv:2006.11239*, 2020.

Huang, C.-W., Lim, J. H., and Courville, A. C. A variational perspective on diffusion-based generative models and score matching. *Advances in Neural Information Processing Systems*, 34, 2021.

Huang, C.-W., Aghajohari, M., Bose, J., Panangaden, P., and Courville, A. C. Riemannian diffusion models. *Advances in Neural Information Processing Systems*, 35: 2750–2761, 2022.

Huang, Y. and Wang, L. A score-based particle method for homogeneous landau equation. *arXiv preprint arXiv:2405.05187*, 2024.

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

Kim, D., Na, B., Kwon, S. J., Lee, D., Kang, W., and Moon, I.-c. Maximum likelihood training of implicit nonlinear diffusion model. *Advances in Neural Information Processing Systems*, 35:32270–32284, 2022.

Kingma, D. P., Salimans, T., Poole, B., and Ho, J. Variational diffusion models. *arXiv preprint arXiv:2107.00630*, 2021.

Kucukelbir, A., Tran, D., Ranganath, R., Gelman, A., and Blei, D. M. Automatic differentiation variational inference. *Journal of machine learning research*, 18(14):1–45, 2017.

Kusuoka, S. and Ninomiya, S. Diffusion processes applied to finance. In *Stochastic Processes and Applications to Mathematical Finance: Proceedings of the Ritsumeikan International Symposium, Kusatsu, Shiga, Japan, 5-9 March 2003*, pp. 233. World Scientific, 2004.

Lai, C.-H., Takida, Y., Murata, N., Uesaka, T., Mitsufuji, Y., and Ermon, S. Fp-diffusion: Improving score-based diffusion models by enforcing the underlying score fokker-planck equation. In *International Conference on Machine Learning*, pp. 18365–18398. PMLR, 2023.

Lamba, H. An adaptive timestepping algorithm for stochastic differential equations. *Journal of computational and applied mathematics*, 161(2):417–430, 2003.

Loshchilov, I. and Hutter, F. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Lu, J., Wu, Y., and Xiang, Y. Score-based transport modeling for mean-field fokker-planck equations. *arXiv preprint arXiv:2305.03729*, 2023.

Nair, V. and Hinton, G. E. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, pp. 807–814, 2010.

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

Nichol, A. and Dhariwal, P. Improved denoising diffusion probabilistic models. *arXiv preprint arXiv:2102.09672*, 2021.

Nichol, A., Dhariwal, P., Ramesh, A., Shyam, P., Mishkin, P., McGrew, B., Sutskever, I., and Chen, M. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741*, 2021.

Otsubo, S., Manikandan, S. K., Sagawa, T., and Krishnamurthy, S. Estimating time-dependent entropy production from non-equilibrium trajectories. *Communications Physics*, 5(1):11, 2022.

Ozaki, T. A bridge between nonlinear time series models and nonlinear stochastic dynamical systems: a local linearization approach. *Statistica Sinica*, pp. 113–135, 1992.

Ozaki, T. A local linearization approach to nonlinear filtering. *International Journal of Control*, 57(1):75–96, 1993.

Pandey, K. and Mandt, S. Generative diffusions in augmented spaces: A complete recipe. *arXiv preprint arXiv:2303.01748*, 2023.

Pavliotis, G. A. *Stochastic processes and applications*. Springer, 2016.

Ranganath, R., Gerrish, S., and Blei, D. Black box variational inference. In *Artificial intelligence and statistics*, pp. 814–822. PMLR, 2014.

Ranganath, R., Tran, D., and Blei, D. Hierarchical variational models. In *International conference on machine learning*, pp. 324–333. PMLR, 2016.

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10684–10695, 2022.

Särkkä, S. and Solin, A. *Applied stochastic differential equations*, volume 10. Cambridge University Press, 2019.

Sasaki, H., Willcocks, C. G., and Breckon, T. P. Unit-ddpm: Unpaired image translation with denoising diffusion probabilistic models. *arXiv preprint arXiv:2104.05358*, 2021.

Singhal, R., Goldstein, M., and Ranganath, R. Where to diffuse, how to diffuse, and how to get back: Automated learning for multivariate diffusions. *arXiv preprint arXiv:2302.07261*, 2023.

Smola, A. J., Gretton, A., and Borgwardt, K. Maximum mean discrepancy. In *13th international conference, ICONIP*, pp. 3–6, 2006.

Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N., and Ganguli, S. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, pp. 2256–2265. PMLR, 2015.

Song, J., Meng, C., and Ermon, S. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020a.

Song, Y. and Ermon, S. Generative modeling by estimating gradients of the data distribution. *arXiv preprint arXiv:1907.05600*, 2019.

Song, Y. and Ermon, S. Improved techniques for training score-based generative models. *arXiv preprint arXiv:2006.09011*, 2020.

Song, Y., Sohl-Dickstein, J., Kingma, D. P., Kumar, A., Ermon, S., and Poole, B. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020b.

Spohn, H. *Large scale dynamics of interacting particles*. Springer Science & Business Media, 2012.

Tailleur, J. and Cates, M. Statistical mechanics of interacting run-and-tumble bacteria. *Physical review letters*, 100 (21):218103, 2008.

Vahdat, A., Kreis, K., and Kautz, J. Score-based generative modeling in latent space. *Advances in Neural Information Processing Systems*, 34:11287–11302, 2021.

Van Den Oord, A., Kalchbrenner, N., and Kavukcuoglu, K. Pixel recurrent neural networks. In *International conference on machine learning*, pp. 1747–1756. PMLR, 2016.

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

Wu, Y. Lecture notes on information-theoretic methods for high-dimensional statistics. *Lecture Notes for ECE598YW (UIUC)*, 16:15, 2017.

A. Local DSM

Suppose we have an inference diffusion process of the form:

$$d\mathbf{y}_t = f(\mathbf{y}_t)dt + g(t)d\mathbf{w}_t \tag{25}$$

where $\mathbf{y}_0 \sim q_{\text{data}}$ and the model process is defined as:

$$d\mathbf{z}_t = [gg^\top(T-t)s_\theta(\mathbf{z}_t, T-t) - f(\mathbf{z}_t, T-t)]dt + g(T-t)d\mathbf{w}_t \tag{26}$$

where $\mathbf{z}_0 \sim \pi_\theta$. Huang et al. (2021); Song & Ermon (2020) derive a variational lower bound on the model log-likelihood $\log p_\theta(x)$:

$$\log p_\theta(x) \geq \mathop{\mathbb{E}}_{q(\mathbf{y}_T \mid x)} \log \pi_\theta(\mathbf{y}_T) + \int_0^T \mathop{\mathbb{E}}_{q(\mathbf{y} \mid x)} \Big[-\frac{1}{2} \left\| s_\theta(\mathbf{y}_t, t) \right\|^2_{gg^\top(t)} - \nabla_{\mathbf{y}_t} \cdot (gg^\top(t) s_\theta(\mathbf{y}_t, t) - f(\mathbf{y}_t, t)) dt \Big] \tag{27}$$

Next, we prove lemma 1, restated here for convenience. Lemma 1 converts the ISM ELBO into the DSM ELBO using transition kernels $q(\mathbf{y}_t \mid \mathbf{y}_s)$.

Lemma. *Let $q(\mathbf{y}_s \mid x), q(\mathbf{y}_t \mid \mathbf{y}_s)$ be the transition kernels of the process defined in eq. (1). For any $0 \leq s < t < T$, we have:*

$$\begin{aligned}
&\mathop{\mathbb{E}}_{q(\mathbf{y}_t \mid x)} \frac{1}{2} \left[\left\| s_\theta(\mathbf{y}_t, t) \right\|^2_{gg^\top} + \nabla_{\mathbf{y}} \cdot gg^\top s_\theta(\mathbf{y}_t, t) \right] \\
&\quad = \mathop{\mathbb{E}}_{q(\mathbf{y}_t, \mathbf{y}_s \mid x)} \left[\frac{1}{2} \left\| s_\theta(\mathbf{y}_t, t) - \nabla_{\mathbf{y}} \log q(\mathbf{y}_t \mid \mathbf{y}_s) \right\|^2_{gg^\top} - \frac{1}{2} \left\| \nabla_{\mathbf{y}} \log q(\mathbf{y}_t \mid \mathbf{y}_s) \right\|^2_{gg^\top} \right].
\end{aligned}$$

where $q(\mathbf{y}_t, \mathbf{y}_s \mid x) = q(\mathbf{y}_t \mid \mathbf{y}_s) q(\mathbf{y}_s \mid x)$.

Proof. Let $F(x, t)$ be defined as:

$$F(x,t) = \mathop{\mathbb{E}}_{q(\mathbf{y}_t \mid x)} \left[\frac{1}{2} \left\| s_\theta \right\|^2_{gg^\top} + \nabla_{\mathbf{y}} \cdot gg^\top s_\theta(\mathbf{y}_t, t) \right] \tag{28}$$

Then note that we can use the Markov property ($q(\mathbf{y}_t, \mathbf{y}_s \mid x) = q(\mathbf{y}_t \mid \mathbf{y}_s) q(\mathbf{y}_s \mid x)$) as follows:

$$\mathop{\mathbb{E}}_{q(\mathbf{y}_t \mid x)} \left[\frac{1}{2} \left\| s_\theta \right\|^2_{gg^\top} + \nabla_{\mathbf{y}} \cdot gg^\top s_\theta(\mathbf{y}_t, t) \right] = \mathop{\mathbb{E}}_{q(\mathbf{y}_t, \mathbf{y}_s \mid x)} \left[\frac{1}{2} \left\| s_\theta \right\|^2_{gg^\top} + \nabla_{\mathbf{y}} \cdot gg^\top s_\theta(\mathbf{y}_t, t) \right] \tag{29}$$

$$= \mathop{\mathbb{E}}_{q(\mathbf{y}_s \mid x)} \left[\mathop{\mathbb{E}}_{q(\mathbf{y}_t \mid \mathbf{y}_s)} \left[\frac{1}{2} \left\| s_\theta \right\|^2_{gg^\top} + \nabla_{\mathbf{y}} \cdot gg^\top s_\theta(\mathbf{y}_t, t) \right] \right] \tag{30}$$

Next, we convert the ISM objective to the DSM objective as follows:

$$\mathop{\mathbb{E}}_{q(\mathbf{y}_t \mid \mathbf{y}_s)} \left[\left\| s_\theta \right\|^2_{gg^\top} \right] \tag{31}$$

$$= \mathop{\mathbb{E}}_{q(\mathbf{y}_t \mid \mathbf{y}_s)} \left[\left\| s_\theta - \nabla_{\mathbf{y}} \log q(\mathbf{y}_t \mid \mathbf{y}_s) \right\|^2_{gg^\top} - \left\| \nabla_{\mathbf{y}} \log q(\mathbf{y}_t \mid \mathbf{y}_s) \right\|^2_{gg^\top} + 2 \left(gg^\top s_\theta \right)^\top \nabla_{\mathbf{y}} \log q(\mathbf{y}_t \mid \mathbf{y}_s) \right] \tag{32}$$

The last term $\mathbb{E}_{q(\mathbf{y}_t \mid \mathbf{y}_s)}[\left(gg^\top s_\theta \right)^\top \nabla_{\mathbf{y}} \log q(\mathbf{y}_t \mid \mathbf{y}_s)]$ is equal to $\mathbb{E}_{q(\mathbf{y}_t \mid \mathbf{y}_s)}[-\nabla_{\mathbf{y}} \cdot gg^\top s_\theta]$ using integration by parts, such that we get:

$$\mathop{\mathbb{E}}_{q(\mathbf{y}_t \mid \mathbf{y}_s)} \left[\left\| s_\theta \right\|^2_{gg^\top} \right] \tag{33}$$

$$= \mathop{\mathbb{E}}_{q(\mathbf{y}_t \mid \mathbf{y}_s)} \left[\left\| s_\theta - \nabla_{\mathbf{y}} \log q(\mathbf{y}_t \mid \mathbf{y}_s) \right\|^2_{gg^\top} - \left\| \nabla_{\mathbf{y}} \log q(\mathbf{y}_t \mid \mathbf{y}_s) \right\|^2_{gg^\top} - 2 \nabla_{\mathbf{y}} \cdot gg^\top s_\theta \right] \tag{34}$$

Combining the last equation with eq. (30), the divergence term gets cancelled out:

$$\mathop{\mathbb{E}}_{q(\mathbf{y}_t \mid \mathbf{y}_s)} \left[\frac{1}{2} \left\| s_\theta \right\|^2_{gg^\top} + \nabla_{\mathbf{y}} \cdot gg^\top s_\theta(\mathbf{y}_t, t) \right] \tag{35}$$

$$= \mathop{\mathbb{E}}_{q(\mathbf{y}_t \mid \mathbf{y}_s)} \left[\frac{1}{2} \left\| s_\theta - \nabla_{\mathbf{y}} \log q(\mathbf{y}_t \mid \mathbf{y}_s) \right\|^2_{gg^\top} - \frac{1}{2} \left\| \nabla_{\mathbf{y}} \log q(\mathbf{y}_t \mid \mathbf{y}_s) \right\|^2_{gg^\top} - \nabla_{\mathbf{y}} \cdot gg^\top s_\theta + \nabla_{\mathbf{y}} \cdot gg^\top s_\theta \right] \tag{36}$$

$$= \mathop{\mathbb{E}}_{q(\mathbf{y}_t \mid \mathbf{y}_s)} \left[\frac{1}{2} \left\| s_\theta - \nabla_{\mathbf{y}} \log q(\mathbf{y}_t \mid \mathbf{y}_s) \right\|^2_{gg^\top} - \frac{1}{2} \left\| \nabla_{\mathbf{y}} \log q(\mathbf{y}_t \mid \mathbf{y}_s) \right\|^2_{gg^\top} \right] \tag{37}$$

Finally, we get:

$$= \mathop{\mathbb{E}}_{q(\mathbf{y}_t \mid x)} \left[\frac{1}{2} \left\| s_\theta \right\|^2_{gg^\top} + \nabla_{\mathbf{y}} \cdot \left(gg^\top s_\theta(\mathbf{y}_t, t) \right) \right]$$

$$= \mathop{\mathbb{E}}_{q(\mathbf{y}_s \mid x)} \left[\mathop{\mathbb{E}}_{q(\mathbf{y}_t \mid \mathbf{y}_s)} \left[\frac{1}{2} \left\| s_\theta - \nabla_{\mathbf{y}} \log q(\mathbf{y}_t \mid \mathbf{y}_s) \right\|^2_{gg^\top} - \left\| \nabla_{\mathbf{y}} \log q(\mathbf{y}_t \mid \mathbf{y}_s) \right\|^2_{gg^\top} \right] \right] \tag{38}$$

$$= \mathop{\mathbb{E}}_{q(\mathbf{y}_t, \mathbf{y}_s \mid x)} \left[\left\| s_\theta - \nabla_{\mathbf{y}} \log q(\mathbf{y}_t \mid \mathbf{y}_s) \right\|^2_{gg^\top} - \frac{1}{2} \left\| \nabla_{\mathbf{y}} \log q(\mathbf{y}_t \mid \mathbf{y}_s) \right\|^2_{gg^\top} \right] \tag{39}$$

□

Now, using lemma 1, we derive the local DSM ELBO for a schedule $s(t)$ which satisfies $0 \leq s(t) < t$ for all $t \in (0, T)$.

Theorem. *Let $q(\mathbf{y}_t \mid \mathbf{y}_s)$ be the transition kernel of the process in eq. (1) and $s(t)$ be a schedule, satisfying $0 \leq s(t) < t$ for all $t \in (0, T]$. Then for a model process $\mathbf{z}_t$ defined in eq. (2), we can lower bound the model log-likelihood as follows:*

$$\log p_\theta(x) \geq \mathop{\mathbb{E}}_{q(\mathbf{y}_T \mid x)} \left[\log \pi_\theta(\mathbf{y}_T) \right] + \int_0^T \mathop{\mathbb{E}}_{q(\mathbf{y}_t, \mathbf{y}_s \mid x)} \left[-\frac{1}{2} \left\| s_\theta - \nabla_{\mathbf{y}} \log q(\mathbf{y}_t \mid \mathbf{y}_s) \right\|^2_{gg^\top} + \frac{1}{2} \left\| \nabla_{\mathbf{y}} \log q(\mathbf{y}_t \mid \mathbf{y}_s) \right\|^2_{gg^\top} + \nabla_{\mathbf{y}} \cdot f dt \right] \tag{40}$$

where $s = s(t)$ and $q(\mathbf{y}_t, \mathbf{y}_s \mid x) = q(\mathbf{y}_t \mid \mathbf{y}_s) q(\mathbf{y}_s \mid x)$ due to the Markov property.

Proof. Using the Markov property, the integrand in the ISM ELBO can be written as:

$$\int_0^T \mathop{\mathbb{E}}_{q(\mathbf{y}_t \mid x)} \left[-\frac{1}{2} \left\| s_\theta \right\|^2_{gg^\top} - \nabla_{\mathbf{y}} \cdot (gg^\top s_\theta - f) dt \right] = \int_0^T \mathop{\mathbb{E}}_{q(\mathbf{y}_t, \mathbf{y}_s \mid x)} \left[-\frac{1}{2} \left\| s_\theta \right\|^2_{gg^\top} - \nabla_{\mathbf{y}} \cdot (gg^\top s_\theta - f) dt \right]$$

$$= \int_0^T \mathop{\mathbb{E}}_{q(\mathbf{y}_s \mid x)} \left[\mathop{\mathbb{E}}_{q(\mathbf{y}_t \mid \mathbf{y}_s)} \left[-\frac{1}{2} \left\| s_\theta \right\|^2_{gg^\top} - \nabla_{\mathbf{y}} \cdot (gg^\top s_\theta - f) \right] dt \right]$$

Using lemma 1, which shows that the ISM integrand is equal to the local DSM integrand in eq. (39), we can convert the ISM ELBO as follows::

$$\mathop{\mathbb{E}}_{q(\mathbf{y}_T \mid x)} \left[\log \pi_\theta(\mathbf{y}_T) \right] + \int_0^T \mathop{\mathbb{E}}_{q(\mathbf{y}_t \mid x)} \left[-\frac{1}{2} \left\| s_\theta \right\|^2_{gg^\top} - \nabla_{\mathbf{y}} \cdot (gg^\top s_\theta - f) dt \right] \tag{41}$$

$$= \mathop{\mathbb{E}}_{q(\mathbf{y}_T \mid x)} \left[\log \pi_\theta(\mathbf{y}_T) \right] + \int_0^T \mathop{\mathbb{E}}_{q(\mathbf{y}_t, \mathbf{y}_s \mid x)} \left[\frac{1}{2} \left\| s_\theta - \nabla_{\mathbf{y}} \log q(\mathbf{y}_t \mid \mathbf{y}_s) \right\|^2_{gg^\top} - \frac{1}{2} \left\| \nabla_{\mathbf{y}} \log q(\mathbf{y}_t \mid \mathbf{y}_s) \right\|^2_{gg^\top} + \nabla_{\mathbf{y}} \cdot f \right]$$

Therefore, we get:

$$\log p_\theta(x) \geq \mathop{\mathbb{E}}_{q(\mathbf{y}_T \mid x)} \left[\log \pi_\theta(\mathbf{y}_T) \right] + \int_0^T \mathop{\mathbb{E}}_{q(\mathbf{y}_t, \mathbf{y}_s \mid x)} \left[-\frac{1}{2} \left\| s_\theta - \nabla_{\mathbf{y}} \log q(\mathbf{y}_t \mid \mathbf{y}_s) \right\|^2_{gg^\top} + \frac{1}{2} \left\| \nabla_{\mathbf{y}} \log q(\mathbf{y}_t \mid \mathbf{y}_s) \right\|^2_{gg^\top} + \nabla_{\mathbf{y}} \cdot f dt \right] \tag{42}$$

□

B. Valid ELBO with Truncation

For numerical stability, the integral term in the ELBO is truncated below by $t_{\text{min}} = \delta$. This leads to a biased estimate. Sohl-Dickstein et al. (2015); Song & Ermon (2020) provide a valid ELBO by using a variational lower bound:

$$\log p_\theta(\mathbf{y}_0) \geq \mathop{\mathbb{E}}_{q(\mathbf{y}_\delta \mid \mathbf{y}_0)} \left[\log \frac{p_\theta(\mathbf{y}_0 \mid \mathbf{y}_\delta)}{q(\mathbf{y}_\delta \mid \mathbf{y}_0)} + \log p_\theta(\mathbf{y}_\delta) \right], \tag{43}$$

where the choice of the likelihood $\log q(\mathbf{y}_\delta \mid \mathbf{y}_0)$ is up to the user. The term $\mathbb{E}_{q(\mathbf{y}_\delta \mid \mathbf{y}_0)} \log p_\theta(\mathbf{y}_\delta)$ can be lower bounded similar to Song & Ermon (2020):

$$\mathop{\mathbb{E}}_{q(\mathbf{y}_\delta \mid \mathbf{y}_0)} [\log p_\theta(\mathbf{y}_\delta)] \geq \mathop{\mathbb{E}}_{q(\mathbf{y}_T \mid x)} \log \pi_\theta(\mathbf{y}_T) + \int_\delta^T \mathop{\mathbb{E}}_{q(\mathbf{y} \mid x)} \Big[-\frac{1}{2} \|s_\theta\|^2_{gg^\top} - \nabla_{\mathbf{y}} \cdot (gg^\top s_\theta - f) dt \Big] \tag{44}$$

see theorem 6 in Song & Ermon (2020). Then using lemma 1, we note that:

$$\begin{aligned}
&\mathop{\mathbb{E}}_{q(\mathbf{y}_T \mid x)} [\log \pi_\theta(\mathbf{y}_T)] + \int_\delta^T \mathop{\mathbb{E}}_{q(\mathbf{y} \mid x)} \Big[-\frac{1}{2} \|s_\theta\|^2_{gg^\top} - \nabla_{\mathbf{y}} \cdot (gg^\top s_\theta - f) dt \Big] \\
&\quad = \mathop{\mathbb{E}}_{q(\mathbf{y}_T \mid x)} [\log \pi_\theta(\mathbf{y}_T)] \qquad (45) \\
&\qquad + \int_\delta^T \mathop{\mathbb{E}}_{q(\mathbf{y}_t, \mathbf{y}_s \mid x)} \Big[-\frac{1}{2} \|s_\theta - \nabla_{\mathbf{y}} \log q(\mathbf{y}_t \mid \mathbf{y}_s)\|^2_{gg^\top} + \frac{1}{2} \|\nabla_{\mathbf{y}} \log q(\mathbf{y}_t \mid \mathbf{y}_s)\|^2_{gg^\top} + \nabla_{\mathbf{y}} \cdot f dt \Big] \qquad (46)
\end{aligned}$$

Now, we choose $q(\mathbf{y}_\delta \mid \mathbf{y}_0) = \mathcal{N}(\mathbf{y}_\delta \mid \mathbf{A}\mathbf{y}_0 + \mathbf{c}, \Sigma(\delta|0))$ and covariance of the locally linear process on the interval $[0, \delta]$. Next, similar to Song & Ermon (2020) we choose $p_\theta(\mathbf{y}_0 \mid \mathbf{y}_\delta)$ to be Gaussian with mean and covariance derived using Tweedie's formula (Efron, 2011). $\mu_\theta = \mathbb{E}[\mathbf{y}_0 \mid \mathbf{y}_\delta]$and $\Sigma_\theta = \text{Var}[\mathbf{y}_0 \mid \mathbf{y}_\delta]$, which are derived below.

First, we derive the conditional variance:

$$\mathbf{y}_\delta = A\mathbf{y}_0 + \mathbf{c} + \Sigma^{-1/2}(\delta \mid 0)\mathbf{z}, \quad \text{where } \mathbf{z} \sim \mathcal{N}(0, I_d) \tag{47}$$

$$\mathbf{y}_0 = \mathbf{A}^{-1} \Big(\mathbf{y}_\delta - c - \Sigma^{-1/2}(\delta \mid 0)\mathbf{z}) \Big) \tag{48}$$

$$\text{Var}(\mathbf{y}_0 \mid \mathbf{y}_\delta) = \mathbf{A}^{-1} P(\delta \mid 0) \mathbf{A}^{-\top} \tag{49}$$

then note that the conditional mean can be derived using Tweedie's formula as follows: let η be the natural parameter of the Gaussian distribution $\mathcal{N}(\mathbf{y}_\delta \mid \mathbf{A}\mathbf{y}_0 + \mathbf{c}, \Sigma(\delta|0))$, then we the fact that (Efron, 2011)

$$\mathbb{E}[\eta \mid \mathbf{y}_\delta] = s_\theta(\mathbf{y}_\delta, \delta) + \Sigma^{-1}(\delta \mid 0)\mathbf{y}_\delta \tag{50}$$

and the definition of the natural parameter η, $\eta = P(\delta \mid 0)^{-1}(\mathbf{A}\mathbf{y}_0 + \mathbf{c})$ to get

$$\begin{aligned}
\mathbb{E}[\mathbf{y}_0 \mid \mathbf{y}_\delta] &= \mathbf{A}^{-1}(\Sigma(\delta|0)\, \mathbb{E}[\eta \mid \mathbf{y}_\delta] - c) && (51) \\
&= \mathbf{A}^{-1}(P(\delta \mid 0) s_\theta(\mathbf{y}_\delta, \delta) + \mathbf{y}_\delta - c) && (52) \\
&= \mathbf{A}^{-1}(\mathbf{y}_\delta - c) + \mathbf{A}^{-1} P(\delta \mid 0) s_\theta(\mathbf{y}_\delta, \delta) && (53)
\end{aligned}$$

See page 26-27 in Singhal et al. (2023) for a full derivation.

C. Local Linearization

Suppose we have diffusions of the form

$$d\mathbf{y}_t = f(\mathbf{y}_t, t)dt + g(t)d\mathbf{w}_t$$

where f is a non-linear function of y. For every any s, we linearize f around the sample $\mathbf{y}_s$ such that

$$d\hat{\mathbf{y}}_t = \mathcal{T}_s f(\hat{\mathbf{y}}_t, t)dt + g(t)d\mathbf{w}_t, \qquad t \in [s, T] \tag{54}$$

where $\mathcal{T}_s$ is a operator that produces a linear approximation of f, such that $\mathcal{T}_{\hat{\mathbf{y}}_s} f = c(t) + A(t)\hat{\mathbf{y}}_t$. Since the drift is affine, eq. (82) is a linear diffusion process with a Gaussian transition kernel (see eq 6.5 in Särkkä & Solin (2019)).

To derive the mean and covariance of the process $\hat{\mathbf{y}}_t$ we use Ito's lemma (see theorem 4.2 in (Särkkä & Solin, 2019)): for a scalar function $F(t, \mathbf{y}_t)$, we have

$$dF = \partial_t F dt + d\hat{\mathbf{y}}^\top \nabla_{\hat{\mathbf{y}}} F + \sum_{i,j=1}^{d} \frac{1}{2} \partial_{\hat{\mathbf{y}}_i} \partial_{\hat{\mathbf{y}}_j} F d\hat{\mathbf{y}}_i d\hat{\mathbf{y}}_j \tag{55}$$

$$= \partial_t F dt + f^\top \nabla_{\hat{\mathbf{y}}} F dt + g^\top \nabla_{\hat{\mathbf{y}}} F d\mathbf{w}_t + \sum_{i,j=1}^{d} \frac{1}{2} \partial_{\hat{\mathbf{y}}_i} \partial_{\hat{\mathbf{y}}_j} F (gg^\top)_{i,j} dt \tag{56}$$

$$= \left[\partial_t F + f^\top \nabla_{\hat{\mathbf{y}}} F + \sum_{i,j=1}^{d} \frac{1}{2} \partial_{\hat{\mathbf{y}}_i} \partial_{\hat{\mathbf{y}}_j} F (gg^\top)_{i,j} \right] dt + + g^\top \nabla_{\hat{\mathbf{y}}} F d\mathbf{w}_t \tag{57}$$

where we use the fact that $dt \times dt = 0, dt \times d\mathbf{w}_t = 0$. Next, we take the expectation

$$d\,\mathbb{E}[F(\hat{\mathbf{y}}_t) \mid \mathbf{y}_s] = \mathbb{E}\left(\partial_t F + f^\top \nabla_{\mathbf{y}} F + \sum_{i,j=1}^{d} \frac{gg^\top(t)_{i,j}}{2} \partial_{\mathbf{y}_i} \partial_{\mathbf{y}_j} F \mid \mathbf{y}_s \right) dt + \mathbb{E}\left[g(t)^\top \partial_{\mathbf{y}} F d\mathbf{w}_t \mid \mathbf{y}_s \right] \tag{58}$$

$$\frac{d}{dt}\mathbb{E}[F(\hat{\mathbf{y}}_t) \mid \mathbf{y}_s] = \mathbb{E}\left(\partial_t F + f^\top \nabla_{\mathbf{y}} F + \sum_{i,j=1}^{d} \frac{gg^\top(t)_{i,j}}{2} \partial_{\mathbf{y}_i} \partial_{\mathbf{y}_j} F \mid \mathbf{y}_s \right) \tag{59}$$

now, for computing the mean and covariance we use Ito's lemma on the functions: $F(t, \hat{\mathbf{y}}) = [\hat{\mathbf{y}}_t]_i$ for the mean and $F_{i,j}(t, \hat{\mathbf{y}}) = [\mathbf{y}_t]_i [\mathbf{y}_t]_j - \mathbb{E}[\hat{\mathbf{y}}_t \mid \mathbf{y}_s]_i \,\mathbb{E}[\hat{\mathbf{y}}_t \mid \mathbf{y}_s]_j$ for all $1 \le i, j \le d$ and therefore we can get the evolution of the mean and covariance ODEs.

The mean $m(t|s) = \mathbb{E}[\hat{\mathbf{y}}_t \mid \hat{\mathbf{y}}_s]$ and covariance $P(t|s) = \mathbb{E}[(\hat{\mathbf{y}}_t - m(t|s))(\hat{\mathbf{y}}_t - m(t|s))^\top \mid \hat{\mathbf{y}}_s]$ obey the following ODEs (see eq 5.50-5.51 in Särkkä & Solin (2019)):

$$\begin{aligned} \frac{d}{dt} m(t \mid s) &= \mathbb{E}[\mathcal{T} f(\hat{\mathbf{y}}_t, t) \mid \hat{\mathbf{y}}_s] \\ &= c(t) + A(t)\,\mathbb{E}[\hat{\mathbf{y}}_t \mid \hat{\mathbf{y}}_s] \\ &= c(t) + A(t) m(t \mid s) \end{aligned} \tag{60}$$

Now, to derive the covariance, we first note that for $F_{i,j}(t, \hat{\mathbf{y}}) = [\mathbf{y}_t]_i [\mathbf{y}_t]_j - \mathbb{E}[\hat{\mathbf{y}}_t \mid \mathbf{y}_s]_i \,\mathbb{E}[\hat{\mathbf{y}}_t \mid \mathbf{y}_s]_j$, we have:

$$\begin{aligned} \frac{d}{dt}\,\mathbb{E}\left[F_{i,j}(t, \mathbf{y}_t) \mid \mathbf{y}_s\right] &= \frac{d}{dt}\,\mathbb{E}\left[[\mathbf{y}_t]_i [\mathbf{y}_t]_j - \mathbb{E}[\hat{\mathbf{y}}_t \mid \mathbf{y}_s]_i \,\mathbb{E}[\hat{\mathbf{y}}_t \mid \mathbf{y}_s]_j \Big| \mathbf{y}_s \right] \\ &= \mathbb{E}\left[\partial_t F_{i,j} + d\mathbf{y}_t^\top \nabla F_{i,j}(t, \mathbf{y}_t) \mid \mathbf{y}_s \right] + \mathbb{E}\left[\sum_{i,j=1}^{d} \frac{gg^\top(t)_{i,j}}{2} \partial_{\mathbf{y}_i} \partial_{\mathbf{y}_j} F_{i,j} \mid \mathbf{y}_s \right] \\ &= \mathbb{E}\left[\partial_t F_{i,j} + d\mathbf{y}_t^\top \nabla F_{i,j}(t, \mathbf{y}_t) \mid \mathbf{y}_s \right] + gg^\top(t)_{i,j} \\ &= -m_i(t \mid s) \partial_t m_j(t \mid s) - m_j(t \mid s) \partial_t m_i(t \mid s) \\ &\quad + \mathbb{E}[[\mathcal{T} f]_i [\mathbf{y}_t]_j \mid \mathbf{y}_s] + \mathbb{E}[[\mathcal{T} f]_j [\mathbf{y}_t]_i \mid \mathbf{y}_s]] + gg^\top(t)_{i,j} \\ &= -m_i(t \mid s)\,\mathbb{E}[[\mathcal{T} f]_j \mid \mathbf{y}_s] - m_j(t \mid s)\,\mathbb{E}[[\mathcal{T} f]_i \mid \mathbf{y}_s] \\ &\quad + \mathbb{E}[[\mathcal{T} f]_i [\mathbf{y}_t]_j \mid \mathbf{y}_s] + \mathbb{E}[[\mathcal{T} f]_j [\mathbf{y}_t]_i \mid \mathbf{y}_s] + gg^\top(t)_{i,j} \\ &= \mathbb{E}[[\mathcal{T} f]_i (\mathbf{y}_t - m(t|s))_j \mid \mathbf{y}_s] + \mathbb{E}\left[[\mathcal{T} f]_j (\mathbf{y}_t - m(t|s))_i \mid \mathbf{y}_s \right] + gg^\top(t)_{i,j} \end{aligned}$$

therefore, we get

$$\frac{d}{dt} P(t \mid s) = \mathbb{E}[\mathcal{T} f(\hat{\mathbf{y}}_t, t)(\hat{\mathbf{y}}_t - m(t|s))^\top \mid \hat{\mathbf{y}}_s] + \mathbb{E}[(\hat{\mathbf{y}}_t - m(t|s))\mathcal{T} f(\hat{\mathbf{y}}_t, t)^\top \mid \hat{\mathbf{y}}_s] + \mathbb{E}[gg^\top(t) \mid \hat{\mathbf{y}}_s] \tag{61}$$

Now, using the fact that $\mathbb{E}[\hat{\mathbf{y}}_t - m(t|s) \mid \hat{\mathbf{y}}_s] = 0$, we get:

$$\begin{aligned}
\mathbb{E}[\mathcal{T}f(\hat{\mathbf{y}}_t, t)(\hat{\mathbf{y}}_t - m(t|s))^\top \mid \hat{\mathbf{y}}_s] &= \mathbb{E}\left[(c(t) + A(t)\hat{\mathbf{y}}_t)\,(\hat{\mathbf{y}}_t - m(t|s))^\top \mid \hat{\mathbf{y}}_s\right] \\
&= \mathbb{E}[c(t)(\hat{\mathbf{y}}_t - \mathbb{E}[\hat{\mathbf{y}}_t \mid \hat{\mathbf{y}}_s])^\top + A(t)\,\mathbb{E}\left[\hat{\mathbf{y}}_t(\hat{\mathbf{y}}_t - m(t|s))^\top \mid \hat{\mathbf{y}}_s\right] \\
&= 0 + A(t)\,\mathbb{E}\left[\hat{\mathbf{y}}_t(\hat{\mathbf{y}}_t - m(t|s))^\top \mid \hat{\mathbf{y}}_s\right], \quad \text{using } \mathbb{E}[\hat{\mathbf{y}}_t - m(t|s) \mid \hat{\mathbf{y}}_s] = 0 \\
&= A(t)\,\mathbb{E}[(\hat{\mathbf{y}}_t - m(t|s))(\hat{\mathbf{y}}_t - m(t|s))^\top \mid \hat{\mathbf{y}}_s] + A(t)m(t|s)\,\mathbb{E}[\hat{\mathbf{y}}_t - m(t|s) \mid \hat{\mathbf{y}}_s]^\top \\
&= A(t)\,\mathbb{E}[(\hat{\mathbf{y}}_t - m(t|s))(\hat{\mathbf{y}}_t - m(t|s))^\top \mid \hat{\mathbf{y}}_s] \\
&= A(t)P(t \mid s),
\end{aligned}$$

and similarly, we get $\mathbb{E}\left[(\hat{\mathbf{y}}_t - m(t|s))\,(c(t) + A(t)\hat{\mathbf{y}}_t)^\top \mid \mathbf{y}_s\right] = P(t \mid s)A(t)^\top$. Therefore, eq. (61) becomes:

$$\frac{d}{dt}P(t|s) = gg^\top(t) + A(t)P(t|s) + P(t|s)A(t)^\top \tag{62}$$

To get the mean and the covariance for the process conditioned on $\hat{\mathbf{y}}_s$ with $s = s$, we solve:

$$\frac{d}{dt}m(t|s) = c(t) + A(t)m(t|s) \tag{63}$$

$$\frac{d}{dt}P(t|s) = gg^\top(t) + A(t)P(t|s) + P(t|s)A(t)^\top \tag{64}$$

where $m(s|s) = \hat{\mathbf{y}}_s$ and $P(s|s) = 0$.

In the next section, we derive the solutions to the ODEs:

- In section appendix C.1, we consider the first-order Taylor expansion $\mathcal{T}_{\mathbf{y}_s,s}f(t,\mathbf{y}_t)$ as the operator $\mathcal{T}$. Here the matrix A is not a function of time t.
- In section appendix C.2, we consider the first-order Taylor expansion $\mathcal{T}_{\mathbf{y}_s,t}f(t,\mathbf{y}_t)$, which provides a more accurate approximation. Here we assume that the drift takes the forms:

  $$f(\mathbf{y},t) = (f_1(\mathbf{y}_1,t),\ldots,f_d(\mathbf{y}_d,t)) \in \mathbf{R}^d \tag{65}$$

  this causes the inference process $\mathbf{y}_t$'s coordinates to be independent given $\mathbf{y}_0$, that is $(\mathbf{y}_t)_i \perp (\mathbf{y}_t)_j \mid \mathbf{y}_s$ for all $i \neq j$.

C.1. Mean And Covariance for Taylor expansion around $f(\mathbf{y}_s, s)$

Suppose we have diffusions of the form

$$d\mathbf{y}_t = f(\mathbf{y}_t,t)dt + g(t)d\mathbf{w}_t$$

where f is a non-linear function of y. Then note that we can simulate the density for any interval $[s,T]$, we linearize f around $\mathbf{y}_s, s$ such that

$$d\hat{\mathbf{y}}_t = \mathcal{T}_{\hat{\mathbf{y}}_s,s}f(\hat{\mathbf{y}}_t,t)dt + g(t)d\mathbf{w}_t, \qquad t \in [s,T] \tag{66}$$

where $\mathcal{T}_{\mathbf{y}_s}$ is a operator that produces a linear approximation of f, for instance, we can use the a first-order Taylor approximation as follows:

$$\begin{aligned}
\mathcal{T}_{\mathbf{y}_s,s}f(\hat{\mathbf{y}}_t,t) &= f(\mathbf{y}_s,s) + \nabla_y f(\mathbf{y}_s,t)(\hat{\mathbf{y}}_t - \mathbf{y}_s) + \partial_t f(\mathbf{y}_s,s)(t-s) \\
&= (f(\mathbf{y}_s,s) - \nabla_y f(\mathbf{y}_s,s)\mathbf{y}_s + \partial_t f(\mathbf{y}_s,s)(t-s)) + \nabla_y f(\mathbf{y}_s,s)\hat{\mathbf{y}}_t \\
&=: (c_1 + c_2 t) + A\hat{\mathbf{y}}_t \\
&=: c(t) + A\hat{\mathbf{y}}_t
\end{aligned}$$

where

- $c_1 = f(\mathbf{y}_s,s) - \nabla_{\mathbf{y}} f(\mathbf{y}_s,s) - \partial_t f(\mathbf{y}_s,s)s$

- $c_2 = \partial_t f(\mathbf{y}_s, s)$
- $A = \nabla_y f(\mathbf{y}_s, s)$

and since, eq. (67) is an affine process, the transition kernel is Gaussian (see eq 6.5 in Särkkä & Solin (2019)). To sample and compute the score of a Gaussian transition kernel requires solving the mean and covariance ODEs.

For the mean, we can solve:

$$\frac{d}{dt} m(t|s) = c(t) + A m(t|s), \qquad m(s|s) = \mathbf{y}_s$$

which we using the following facts:

$$\begin{aligned}
\exp(tA) \int_s^t \exp(-\tau A) c_1 d\tau &= \exp(tA)[-\exp(-\tau A)A^{-1}]_s^t c_1 && (67)\\
&= \exp(tA)[\exp(-sA)A^{-1} - \exp(-tA)A^{-1}]c_1 && (68)\\
&= [\exp((t-s)A)A^{-1} - \exp((t-t)A)A^{-1}]c_1 && (69)\\
&= [\exp((t-s)A)A^{-1} - A^{-1}]c_1 && (70)\\
&= [\exp((t-s)A) - I]A^{-1}c_1 && (71)
\end{aligned}$$

and using integration by parts and the above integral we get:

$$\begin{aligned}
\int \exp(-\tau A)\tau d\tau &= [\tau \int \exp(-\tau A) - \int \frac{d}{d\tau}\tau \int \exp(-\tau A)] && (72)\\
&= \left[-\tau \exp(-\tau A)A^{-1} - \int -\exp(-\tau A)A^{-1}\right] && (73)\\
&= \left[-\tau \exp(-\tau A)A^{-1} - \exp(-\tau A)A^{-2}\right] && (74)
\end{aligned}$$

Now, using these identities and the general solution to affine linear ODEs (see eq 2.31 in Särkkä & Solin (2019)) we get that $m(t|s)$ evolves as

$$\begin{aligned}
m(t|s) &= \exp((t-s)A)\mathbf{y}_s + \int_s^t \exp\left((t-\tau)A\right) c(\tau) d\tau && (75)\\
&= \exp((t-s)A)\mathbf{y}_s + \exp(tA) \int_s^t \exp\left(-\tau A\right) c(\tau) d\tau && (76)\\
&= \exp((t-s)A)\mathbf{y}_s + \exp(tA) \int_s^t \exp\left(-\tau A\right) (c_1 + c_2\tau) d\tau && (77)\\
&= \exp((t-s)A)\mathbf{y}_s + \exp(tA) \int_s^t \exp\left(-\tau A\right) c_1 d\tau + \exp(tA) \int_s^t \exp\left(-\tau A\right) c_2 \tau d\tau && (78)\\
&= \exp((t-s)A)\mathbf{y}_s + (\exp((t-s)A) - I)A^{-1}c_1 + \exp(tA) \int_s^t \exp\left(-\tau A\right) c_2 \tau d\tau && (79)
\end{aligned}$$

Now, to integrate the last term, we note that using integration by parts we get on the integrand $\exp(-\tau A)\tau$ we get:

$$\begin{aligned}
\int_s^t \exp\left(-\tau A\right) \tau d\tau &= \left[s \exp(-sA)A^{-1} + \exp(-sA)A^{-2}\right] - \left[t \exp(-tA)A^{-1} + \exp(-tA)A^{-2}\right]\\
\exp(tA) \int_s^t \exp\left(-\tau A\right) \tau d\tau &= \left[s \exp((t-s)A)A^{-1} + \exp((t-s)A)A^{-2}\right] - \left[tA^{-1} + A^{-2}\right]\\
&= \exp((t-s)A)\left[sA^{-1} + A^{-2}\right] - \left[tA^{-1} + A^{-2}\right]
\end{aligned}$$

Finally, the mean:

$$m(t \mid s) = \exp((t-s)A)\mathbf{y}_s + (\exp((t-s)A) - I)A^{-1}c_1$$

$$+ \exp((t-s)A)\left[sA^{-1} + A^{-2}\right]c_2 - \left[tA^{-1} + A^{-2}\right]c_2$$

Following Särkkä & Solin (2019); Singhal et al. (2023), we can solve the covariance using the matrix factorization trick. Let $P(t|s) = C_{t|s}H_{t|s}^{-1}$, then $C_{t|s}, H_{t|s}$ evolve as follows:

$$\begin{pmatrix} C_{t|s} \\ H_{t|s} \end{pmatrix} = \exp\begin{pmatrix} \int_s^t A(\tau)d\tau & \int_s^t gg^\top(\tau)d\tau \\ 0 & -\int_s^t A^\top(\tau)d\tau \end{pmatrix}\begin{pmatrix} C_0 \\ H_0 \end{pmatrix} \tag{80}$$

where $C_0 = 0$ and $H_0 = I$.

C.2. Mean And Covariance for Taylor expansion around $f(\mathbf{y}_s, t)$

Suppose we have a diffusion process of the form

$$d\mathbf{y}_t = f(\mathbf{y}_t, t)dt + g(t)d\mathbf{w}_t$$

where f is a non-linear function of y.

Here we also assume that $\nabla_{\mathbf{y}_j} f_i(\mathbf{y}, t) = 0$ for all i, j where $f = (f_1, \ldots, f_d) \in \mathbf{R}^d$, which implies that conditional on $\mathbf{y}_s$ for $s \in [0, t)$ the inference process obeys

$$q(\mathbf{y}_t \mid \mathbf{y}_s) = \prod_{i=1}^{d} q([\mathbf{y}_t]_i \mid [\mathbf{y}_s]_i) \tag{81}$$

And since the inference process coordinates $[\mathbf{y}_t]_i$ and $[\mathbf{y}_t]_j$ for all $i \neq j$ are independent conditional on $\mathbf{y}_s$, we treat m, P as ***scalar values***. We also note that the matrix A is a function of t, unlike the previous section.

Then, similar to appendix C.1, to simulate the density for any interval $[s, T]$, we linearize f around $\mathbf{y}_s, s$ by defining a linear process:

$$d\hat{\mathbf{y}}_t = \mathcal{T}_{\mathbf{y}_s, s} f(\hat{\mathbf{y}}_t, t)dt + g(t)d\mathbf{w}_t, \qquad t \in [s, T] \tag{82}$$

where $\mathcal{T}_{\mathbf{y}_s}$ is a operator that produces a linear approximation of f, for instance, we can use the a first-order Taylor approximation as follows:

$$\begin{aligned} \mathcal{T}_{\mathbf{y}_s, t} f(\hat{\mathbf{y}}_t, t) &= f(\mathbf{y}_s, t) + \nabla_y f(\mathbf{y}_s, t)(\hat{\mathbf{y}}_t - \mathbf{y}_s) \\ &= (f(\mathbf{y}_s, t) - \nabla_y f(\mathbf{y}_s, t)\mathbf{y}_s) + \nabla_y f(\mathbf{y}_s, s)\hat{\mathbf{y}}_t \\ &=: c(t) + A(t)\hat{\mathbf{y}}_t \end{aligned}$$

here both c, A are a function of t. As shown earlier, the transition kernel is Gaussian (see eq 6.5 in Särkkä & Solin (2019)). To sample and compute the score of a Gaussian transition kernel requires solving the mean and covariance ODEs.

To solve for $P(t|s) \in \mathbf{R}, m(t|s) \in \mathbf{R}$, we make use of the matrix exponential technique from eq 6.36-39 in Särkkä & Solin (2019); Singhal et al. (2023). For solving the covariance matrix ODE, we let $P(t \mid s) = C_{t|s}H_{t|s}^{-1}$ where C, H evolve as follows:

$$\frac{d}{dt}\begin{pmatrix} C_{t|s} \\ H_{t|s} \end{pmatrix} = \begin{pmatrix} A(t) & gg^\top \\ 0 & -A^\top(t) \end{pmatrix}\begin{pmatrix} C_s \\ H_s \end{pmatrix} \tag{83}$$

where $C_0 = 0$ and $H_0 = I$. Now, since C, H evolve linearly, we can solve them using matrix exponentials.

$$\begin{pmatrix} C_{t|s} \\ H_{t|s} \end{pmatrix} = \exp\begin{pmatrix} \int_s^t A(t) & \int_s^t gg^\top(t) \\ 0 & -\int_s^t A^\top(t) \end{pmatrix}\begin{pmatrix} C_0 \\ H_0 \end{pmatrix} \tag{84}$$

$$\frac{d}{dt}\begin{pmatrix} C_{t|s} \\ H_{t|s} \end{pmatrix} = \begin{pmatrix} A(t) & gg^\top \\ 0 & -A^\top(t) \end{pmatrix}\begin{pmatrix} C_s \\ H_s \end{pmatrix} \tag{85}$$

$$P(t|s) = C_{t|s}H_{t|s}^{-1} \tag{86}$$

Mean ODE solution. To solve the mean ODE:

$$\frac{d}{dt}m(t|s) = c(t) + A(t)m(t|s)$$

and since $A(t)$ is a scalar, $A = A^\top$, therefore we can use the same matrix exponential technique as we used for the covariance matrix. Let $m(t|s) = D_{t|s}R_{t|s}^{-1}$, where D, R evolve as:

$$\frac{d}{dt}\begin{pmatrix} D_{t|s} \\ R_{t|s} \end{pmatrix} = \begin{pmatrix} \frac{1}{2}A(t) & c(t) \\ 0 & -\frac{1}{2}A^\top(t) \end{pmatrix}\begin{pmatrix} D_{t|s} \\ R_{t|s} \end{pmatrix}. \tag{87}$$

Here, the factorization $m(t|s) = D_{t|s}R_{t|s}^{-1}$ holds as:

$$\begin{aligned}
\frac{d}{dt}D_{t|s}R_{t|s}^{-1} &= R_{t|s}^{-1}\frac{d}{dt}D_{t|s} + D_{t|s}\frac{d}{dt}R_{t|s}^{-1} && (88)\\
&= R_{t|s}^{-1}\left(\frac{1}{2}A(t)D_{t|s} + c(t)R_{t|s}\right) + D_{t|s}\frac{-1}{R_{t|s}^2}\frac{d}{dt}R_{t|s} && (89)\\
&= R_{t|s}^{-1}\left(\frac{1}{2}A(t)D_{t|s} + c(t)R_{t|s}\right) + D_{t|s}\frac{-1}{R_{t|s}^2}\frac{-1}{2}A(t)R_{t|s} && (90)\\
&= \left(\frac{1}{2}A(t)D_{t|s}R_{t|s}^{-1} + c(t)\right) + D_{t|s}\frac{1}{R_{t|s}}\frac{1}{2}A(t) && (91)\\
&= \left(\frac{1}{2}A(t)D_{t|s}R_{t|s}^{-1} + c(t)\right) + \frac{1}{2}D_{t|s}R_{t|s}^{-1}A(t) && (92)\\
&= \frac{1}{2}A(t)D_{t|s}R_{t|s}^{-1} + c(t) + \frac{1}{2}D_{t|s}R_{t|s}^{-1}A(t) && (93)\\
&= A(t)D_{t|s}R_{t|s}^{-1} + c(t) && (94)\\
&= A(t)m(t|s) + c(t) && (95)
\end{aligned}$$

Now, we can solve for R_s, P_s in closed-form as

$$\begin{pmatrix} D_{t|s} \\ R_{t|s} \end{pmatrix} = \exp\begin{pmatrix} \frac{1}{2}\int_s^t A(t) & \int_s^t c(t) \\ 0 & -\frac{1}{2}\int_s^t A^\top(t) \end{pmatrix}\begin{pmatrix} D_s \\ R_s \end{pmatrix} \tag{96}$$

where $D_s = \mathbf{y}_s$ and $R_s = I$.

D. Regularity assumptions

In this section, we list a set of assumptions on f, g and q_{data} which we assume throughout the paper:

(A1) $q_{\text{data}}(\mathbf{y}_t)$ is twice differentiable for all t, $q_{\text{data}} \in C^2(\mathbf{R}^d)$.

(A2) The drift $f(t, \mathbf{y})$ and diffusion coefficient $g(t)$ satisfy:

- $f \in C^2(\mathbf{R}^d, \mathbf{R}_+)$, and f is Lipschitz in the $\mathbf{y}$ argument
- f, g are integrable with respect to q_{data}

Both A1-A2 imply that $q(\mathbf{y}_t)$ exists and $q(\mathbf{y}_t)$ is twice differentiable, see Haussmann & Pardoux (1986).

E. Error Estimate

In this section we prove that for any $t \in (0, T]$, the gap between the true marginal $q(\mathbf{y}_t)$ and the locally linear approximation $\widehat{q}(\mathbf{y}_t) = \mathbb{E}_{q(\mathbf{y}_s)}[\widehat{q}(\mathbf{y}_t \mid \mathbf{y}_s)]$ is upper bounded by the difference of the drifts between the interval $(s(t), t)$.

Lemma 2. *For $t \in (0, T]$, we assume that $gg^T(t) = g^2(t)\mathbf{I}_d$, where $g^2(t)$ is a scalar, and f, g, q_{data} satisfy smoothness assumptions in appendix D. For any $t \in (0, T]$, we have:*

$$\text{KL}\left(q(\mathbf{y}_t) \mid \widehat{q}(\mathbf{y}_t)\right) \leq \int_{s(t)}^t \mathop{\mathbb{E}}_{q(\mathbf{y}_\tau)}\left[\frac{1}{2g^2}\left\|f(\tau, \mathbf{y}_\tau) - \mathcal{T}_s f(t, \mathbf{y}_\tau)\right\|_2^2\right] d\tau$$

where $\mathcal{T}_s$ is the linearization operator and $q(\mathbf{y}_t \mid \mathbf{y}_s)$ is the exact transition kernel.

The main idea behind the proof is the following:

- Due to Jensen's inequality and convexity of f-divergences (see theorem 4.1 in Wu (2017)), we have:

$$\mathrm{KL}\left(q(\mathbf{y}_t), \widehat{q}(\mathbf{y}_t)\right) \leq \mathop{\mathbb{E}}_{q(\mathbf{y}_s)} \mathrm{KL}\left(q(\mathbf{y}_t \mid \mathbf{y}_s), \widehat{q}(\mathbf{y}_t \mid \mathbf{y}_s)\right) \tag{97}$$

- Next, we upper bound $\mathrm{KL}\left(q(\mathbf{y}_t \mid \mathbf{y}_s), \widehat{q}(\mathbf{y}_t \mid \mathbf{y}_s)\right)$ using proposition 1.

Proposition 1 (Lemma 2.21 in Albergo et al. (2023))**.** *Suppose $q, \widehat{q}$ evolve as follows:*

$$\partial_t q + \nabla \cdot (Fq) = 0 \tag{98}$$

$$\partial_t \widehat{q} + \nabla \cdot (\widehat{F}\widehat{q}) = 0, \tag{99}$$

$$\tag{100}$$

where $F = f - \frac{1}{2}g^2 \nabla \log q$, then the KL *divergence between $q, \widehat{q}$ can be expressed as:*

$$\mathrm{KL}\left(q(\mathbf{y}_t \mid \mathbf{y}_s), \widehat{q}(\mathbf{y}_t \mid \mathbf{y}_s)\right) = \int_s^t \int_{\mathbf{R}^d} \left(s_q - s_{\widehat{q}}\right)^\top \left(F - \widehat{F}\right) q(\mathbf{y}_t \mid \mathbf{y}_s) d\mathbf{y}_t dt \tag{101}$$

which implies

$$\mathrm{KL}\left(q(\mathbf{y}_t \mid \mathbf{y}_s), \widehat{q}(\mathbf{y}_t \mid \mathbf{y}_s)\right) \leq \int_s^t \mathop{\mathbb{E}}_{q(\mathbf{y}_\tau \mid \mathbf{y}_s)} \left[\frac{1}{2g^2} \left\| f - \widehat{f} \right\|_2^2\right] d\tau \tag{102}$$

Proof. KL divergence evolves as:

$$\begin{aligned}
\frac{d}{dt}\mathrm{KL}\left(q(\mathbf{y}_t \mid \mathbf{y}_s), \widehat{q}(\mathbf{y}_t \mid \mathbf{y}_s)\right) &= \frac{d}{dt} \int \log \frac{q}{\widehat{q}} q d\mathbf{y} \\
&= -\frac{d}{dt} \int q \log \widehat{q} d\mathbf{y} + \frac{d}{dt} \int q \log q d\mathbf{y} \\
&= -\int \partial_t (q \log \widehat{q}) d\mathbf{y} + \int \partial_t (q \log q) d\mathbf{y} \\
&= -\int \left(q \partial_t \log \widehat{q} + \log \widehat{q} \partial_t q\right) d\mathbf{y} + \int \left(q \partial_t \log q + \log q \partial_t q\right) d\mathbf{y} \\
&= -\int \left(\frac{q}{\widehat{q}} \partial_t \widehat{q} + \log \widehat{q} \partial_t q\right) d\mathbf{y} + \int \left(\partial_t q + \log q \partial_t q\right) d\mathbf{y} \\
&= -\int \left(\frac{q}{\widehat{q}} \partial_t \widehat{q} + \log \frac{\widehat{q}}{q} \partial_t q\right) d\mathbf{y} + \int \left(\partial_t q\right) d\mathbf{y} \\
&= -\int \frac{q}{\widehat{q}} \partial_t \widehat{q} d\mathbf{y} + \int \log \frac{q}{\widehat{q}} \partial_t q d\mathbf{y}, \qquad \text{since } \partial_t \int q d\mathbf{y} = 0 \\
&= -\int \frac{q}{\widehat{q}} \nabla \cdot (-\widehat{F}\widehat{q}) d\mathbf{y} + \int \log \frac{q}{\widehat{q}} \nabla \cdot (-Fq) d\mathbf{y} \\
&= \int \nabla \left(\frac{q}{\widehat{q}}\right)^\top (\widehat{F}\widehat{q}) d\mathbf{y} - \int \left(\nabla \log q - \nabla \log \widehat{q}\right)^\top (Fq) d\mathbf{y} \\
&= \int \left(\frac{q}{\widehat{q}}\right) \frac{1}{\left(\frac{q}{\widehat{q}}\right)} \nabla \left(\frac{q}{\widehat{q}}\right)^\top (\widehat{F}\widehat{q}) d\mathbf{y} - \int \left(\nabla \log q - \nabla \log \widehat{q}\right)^\top (Fq) d\mathbf{y} \\
&= \int \left(\frac{q}{\widehat{q}}\right) \nabla \log \left(\frac{q}{\widehat{q}}\right)^\top (\widehat{F}\widehat{q}) d\mathbf{y} - \int \left(\nabla \log q - \nabla \log \widehat{q}\right)^\top (F) d\mathbf{y} \\
&= \int \left(\nabla \log q - \nabla \log \widehat{q}\right)^\top \widehat{F} q d\mathbf{y} - \int \left(\nabla \log q - \nabla \log \widehat{q}\right)^\top (Fq) d\mathbf{y}
\end{aligned}$$

$$
\begin{aligned}
&= \int (\nabla \log q - \nabla \log \widehat{q})^\top \left(F - \widehat{F}\right) q d\mathbf{y} \\
&= \int (s_q - s_{\widehat{q}})^\top \left(F - \widehat{F}\right) q d\mathbf{y} \qquad (103)
\end{aligned}
$$

Then we bound the KL divergence between $q, \widehat{q}$ using eq. (103) we get:

$$
\begin{aligned}
\mathrm{KL}\left(q(\mathbf{y}_t \mid \mathbf{y}_s), \widehat{q}(\mathbf{y}_t \mid \mathbf{y}_s)\right) &= \int_s^t \int_{\mathbf{R}^d} \left(s_q - s_{\widehat{q}}\right)^\top \left([f - \frac{g^2}{2} s_q] - [\widehat{f} - \frac{g^2}{2} s_{\widehat{q}}]\right) q(\mathbf{y}_t \mid \mathbf{y}_s) d\mathbf{y}_t dt \\
&= \int_s^t \int_{\mathbf{R}^d} \left(s_q - s_{\widehat{q}}\right)^\top \left(f - \widehat{f}\right) q(\mathbf{y}_t \mid \mathbf{y}_s) d\mathbf{y}_t dt \\
&\quad - \int_s^t \int_{\mathbf{R}^d} \left(s_q - s_{\widehat{q}}\right)^\top \frac{g^2}{2} \left(s_q - s_{\widehat{q}}\right) q(\mathbf{y}_t \mid \mathbf{y}_s) d\mathbf{y}_t dt \\
&= \int_s^t \int_{\mathbf{R}^d} \left(s_q - s_{\widehat{q}}\right)^\top \left(f - \widehat{f}\right) q(\mathbf{y}_t \mid \mathbf{y}_s) d\mathbf{y}_t dt \qquad (104) \\
&\quad - \int_s^t \int_{\mathbf{R}^d} \frac{g^2}{2} \left\| s_q - s_{\widehat{q}} \right\|_2^2 q(\mathbf{y}_t \mid \mathbf{y}_s) d\mathbf{y}_t dt
\end{aligned}
$$

Now, to upper bound the first integral, we use the fact that for any vectors $\mathbf{a}, \mathbf{b} \in \mathbf{R}^d$

$$
\begin{aligned}
\|\mathbf{a} - \mathbf{b}\|_2^2 &\geq 0 \\
\|\mathbf{a}\|_2^2 + \|\mathbf{b}\|_2^2 - 2\mathbf{a}^\top \mathbf{b} &\geq 0 \\
\mathbf{a}^\top \mathbf{b} &\leq \frac{1}{2}\left(\|\mathbf{a}\|_2^2 + \|\mathbf{b}\|_2^2\right)
\end{aligned}
$$

which implies that for $\eta > 0$ and $\mathbf{a} = \frac{1}{\sqrt{\eta}}(f - \widehat{f})$ and $\mathbf{b} = \sqrt{\eta}(s_q - s_{\widehat{q}})$, we get:

$$
\int_s^t \int_{\mathbf{R}^d} \left(s_q - s_{\widehat{q}}\right)^\top \left(f - \widehat{f}\right) q(\mathbf{y}_t \mid \mathbf{y}_s) d\mathbf{y}_t dt \leq \int_s^t \int_{\mathbf{R}^d} \left(\frac{\eta}{2} \left\| s_q - s_{\widehat{q}} \right\|_2^2 + \frac{1}{2\eta} \left\| f - \widehat{f} \right\|_2^2\right) q(\mathbf{y}_t \mid \mathbf{y}_s) d\mathbf{y}_t dt \qquad (105)
$$

now, using eq. (105) and setting $\eta = g^2$ in eq. (104), we get:

$$
\begin{aligned}
\mathrm{KL}\left(q(\mathbf{y}_t \mid \mathbf{y}_s), \widehat{q}(\mathbf{y}_t \mid \mathbf{y}_s)\right) &\leq \int_s^t \int_{\mathbf{R}^d} \frac{1}{2g^2} \left\| f - \widehat{f} \right\|_2^2 q(\mathbf{y}_t \mid \mathbf{y}_s) d\mathbf{y}_t dt \\
&= \int_s^t \mathop{\mathbb{E}}_{q(\mathbf{y}_t \mid \mathbf{y}_s)} \left[\frac{1}{2g^2} \left\| f - \widehat{f} \right\|_2^2\right] dt \qquad (106)
\end{aligned}
$$

□

We note that due to Jensen's inequality (see theorem 4.1 in Wu (2017)):

$$
\mathrm{KL}\left(q(\mathbf{y}_t), \widehat{q}(\mathbf{y}_t)\right) \leq \mathop{\mathbb{E}}_{q(\mathbf{y}_s)} \mathrm{KL}\left(q(\mathbf{y}_t \mid \mathbf{y}_s), \widehat{q}(\mathbf{y}_t \mid \mathbf{y}_s)\right) \qquad (107)
$$

which combined with eq. (106) gets:

$$
\begin{aligned}
\mathrm{KL}\left(q(\mathbf{y}_t), \widehat{q}(\mathbf{y}_t)\right) &\leq \mathop{\mathbb{E}}_{q(\mathbf{y}_s)} \int_s^t \int_{\mathbf{R}^d} \frac{1}{2g^2} \left\| f - \widehat{f} \right\|_2^2 q(\mathbf{y}_\tau \mid \mathbf{y}_s) d\mathbf{y}_\tau dt \qquad (108) \\
&= \int_s^t \mathop{\mathbb{E}}_{q(\mathbf{y}_s)} \mathop{\mathbb{E}}_{q(\mathbf{y}_\tau \mid \mathbf{y}_s)} \left[\frac{1}{2g^2} \left\| f - \widehat{f} \right\|_2^2\right] d\tau \qquad (109) \\
&= \int_s^t \mathop{\mathbb{E}}_{q(\mathbf{y}_\tau)} \left[\frac{1}{2g^2} \left\| f - \widehat{f} \right\|_2^2\right] d\tau \qquad (110)
\end{aligned}
$$

F. Taylor Series tricks.

F.1. Time-dependent $s(t)$

Now, we find a function $s(t)$ such that for $t > t_{\text{min}}$, we get the following:

$$\int_{s(t)}^{t} \beta(\tau) d\tau \tag{111}$$

We first derive it for a linear $\beta(t)$ followed by $\beta(t)$ used to derive the linear and cosine noise schedules (Chen, 2023).

Linear $\beta(t)$. For a linear $\beta(t)$ function, we let $s(t) = t - \epsilon(t)$

$$\int_{t-\epsilon(t)}^{t} \beta(s) ds = [\beta_{min} t + \beta_{max} \frac{t^2}{2}]_{t-\epsilon(t)}^{t} \tag{112}$$

$$= \beta_{min}(\epsilon(t)) + \beta_{max} \frac{1}{2}(t^2 - (t - \epsilon(t))^2) \tag{113}$$

$$= \beta_{min}\epsilon(t) + \frac{\beta_{max}}{2}\epsilon(t)(2t - \epsilon(t)) \tag{114}$$

$$= \beta_{min}\epsilon(t) + \frac{\beta_{max}}{2}(2t\epsilon(t) - \epsilon(t)^2) \tag{115}$$

$$\tag{116}$$

Suppose if we choose $\epsilon(t)$ such that $\int_{t-\epsilon(t)}^{t} \beta(s) ds = \lambda$, then note that

$$\lambda = \int_{t-\epsilon(t)}^{t} \beta(s) ds \tag{117}$$

$$= \beta_{min}\epsilon(t) + \frac{\beta_{max}}{2}(2t\epsilon(t) - \epsilon(t)^2) \tag{118}$$

now, to find $\epsilon(t)$ we define a polynomial:

$$P(x) = \beta_{min} x + \frac{\beta_{max}}{2}(2tx - x^2) - \lambda \tag{119}$$

$$= -\frac{\beta_{max}}{2}x^2 + (\beta_{min} + \beta_{max} t)x - \lambda \tag{120}$$

then $\epsilon(t)$ is a zero of the polynomial $P(x)$. We can find the zeros of $P(x)$:

$$x^* = \frac{-(\beta_{min} + \beta_{max} t) \pm \sqrt{(\beta_{min} + \beta_{max} t)^2 - 4\lambda \frac{\beta_{max}}{2}}}{-\beta_{max}} \tag{121}$$

$$= \frac{-(\beta_{min} + \beta_{max} t) \pm \sqrt{(\beta_{min} + \beta_{max} t)^2 - 2\lambda\beta_{max}}}{-\beta_{max}} \tag{122}$$

$$= \frac{(\beta_{min} + \beta_{max} t) \pm \sqrt{(\beta_{min} + \beta_{max} t)^2 - 2\lambda\beta_{max}}}{\beta_{max}} \tag{123}$$

$$= t + \frac{\beta_{min}}{\beta_{max}} \pm \frac{\sqrt{(\beta_{min} + \beta_{max} t)^2 - 2\lambda\beta_{max}}}{\beta_{max}} \tag{124}$$

the constraint $0 < t - \epsilon(t) < t$, implies that:

$$\beta_{min} \pm \sqrt{(\beta_{min} + \beta_{max} t)^2 - 2\lambda\beta_{max}} < 0 \tag{125}$$

$$\beta_{min} \pm \sqrt{\beta(t)^2 - 2\lambda\beta_{max}} < 0 \tag{126}$$

$$\beta(t)^2 - 2\lambda\beta_{max} < \beta_{min}^2 \tag{127}$$

$$\beta(t) < \sqrt{\beta_{min}^2 + 2\lambda\beta_{max}} \tag{128}$$

and we require that $\beta(t)^2 - 2\lambda\beta_{max} > 0$ such that $\epsilon(t)$ is not complex-valued:

$$\beta(t)^2 - 2\lambda\beta_{max} > 0 \tag{129}$$

$$\beta(t) > \sqrt{2\lambda\beta_{max}} \tag{130}$$

which implies that

$$\epsilon(t) = t + \frac{\beta_{min} - \sqrt{\beta(t)^2 - 2\lambda\beta_{max}}}{\beta_{max}} \tag{131}$$

for t such that

$$\sqrt{2\lambda\beta_{max}} < \beta(t) < \sqrt{\beta_{min}^2 + 2\lambda\beta_{max}} \tag{132}$$

Commonly used $\beta(t)$ functions. Chen (2023) studies the effect of different noise schedules $\gamma(t)$:

$$\mathbf{y}_t = \sqrt{\gamma(t)}x + \sqrt{1-\gamma(t)}\epsilon \tag{133}$$

with the following choices for $\gamma(t)$:

$$\text{cosine} : \gamma(t) = \cos\left(\frac{\pi}{2}t\right) \tag{134}$$

$$\text{linear} : \gamma(t) = 1 - t \tag{135}$$

Now, note that for the VPSDE process, we have $\mathbf{y}_t = m(t)x + \sigma(t)\epsilon$, where

$$m(t) = \exp\left(-\int_0^t \frac{1}{2}\beta(s)ds\right) = \sqrt{\exp\left(-\int_0^t \beta(s)ds\right)} \tag{136}$$

$$\sigma(t) = \sqrt{1 - \exp\left(-\int_0^t \beta(s)ds\right)} \tag{137}$$

which implies that

$$\frac{d}{dt}\log m(t) = -\frac{1}{2}\left(\beta(t) - \beta(0)\right) \tag{138}$$

$$\frac{d}{dt}\log\sqrt{\gamma(t)} = -\frac{1}{2}(\beta(t) - \beta(0)) \tag{139}$$

$$\frac{1}{2}\frac{d}{dt}\log\gamma(t) = -\frac{1}{2}(\beta(t) - \beta(0)) \tag{140}$$

$$\frac{d}{dt}\log\gamma(t) = -(\beta(t) - \beta(0)) \tag{141}$$

$$\beta(t) = \beta(0) - \frac{d}{dt}\log\gamma(t) \tag{142}$$

For the commonly used noise schedules, we can derive the $\beta(t)$ function:

$$\text{cosine} : \beta(t) = \beta(0) - \frac{-\sin\left(\frac{\pi}{2}t\right)}{\cos\left(\frac{\pi}{2}t\right)} = \beta(0) + \tan(\frac{\pi}{2}t) \tag{143}$$

$$\text{linear} : \beta(t) = \beta(0) - \frac{-1}{1-t} = \beta(0) + \frac{1}{1-t} \tag{144}$$

Now, note that we can find $s(t)$ such that $\int_{s(t)}^t \beta(\tau)d\tau = \lambda$ for a user-specified λ and linear $\beta(t)$, as follows:

$$\lambda = \int_{s(t)}^t \beta(\tau)d\tau \tag{145}$$

$$= \int_{s(t)}^{t} \beta(0) + \frac{1}{1-\tau} d\tau \tag{146}$$

$$= \left[-\log(1-\tau) \right]_{s(t)}^{t}, \qquad \text{assuming } \beta(0) = 0 \tag{147}$$

$$\exp(-\lambda) = \frac{1-t}{1-s(t)} \tag{148}$$

$$1 - s(t) = \frac{1-t}{\exp(-\lambda)} \tag{149}$$

$$s(t) = 1 - \frac{1-t}{\exp(-\lambda)} \tag{150}$$

Similarly for a cosine $\beta(t)$, we note that

$$\lambda = \int_{s(t)}^{t} \beta(\tau) d\tau \tag{151}$$

$$= \int_{s(t)}^{t} \beta(0) + \frac{1}{1-\tau} d\tau, \qquad \text{assume } \beta(0) = 0 \tag{152}$$

$$= \left[-\frac{2}{\pi} \log \cos(\frac{\pi}{2}\tau) \right]_{s(t)}^{t} \tag{153}$$

$$= -\frac{2}{\pi} \log \frac{\cos(\frac{\pi}{2}t)}{\cos(\frac{\pi}{2}s(t))} \tag{154}$$

$$\exp(-\frac{\pi}{2}\lambda) = \frac{\cos(\frac{\pi}{2}t)}{\cos(\frac{\pi}{2}s(t))} \tag{155}$$

$$\cos(\frac{\pi}{2}s(t)) = \frac{1}{\exp(-\frac{\pi}{2}\lambda)} \cos(\frac{\pi}{2}t) \tag{156}$$

$$\frac{\pi}{2}s(t) = \cos^{-1}\left(\frac{1}{\exp(-\frac{\pi}{2}\lambda)} \cos(\frac{\pi}{2}t) \right) \tag{157}$$

$$s(t) = \frac{2}{\pi} \cos^{-1}\left(\frac{1}{\exp(-\frac{\pi}{2}\lambda)} \cos(\frac{\pi}{2}t) \right) \tag{158}$$

G. Active Matter Experiments

G.1. Active Swimmer

In this section we plot the samples from the ISM trained model versus the inference process samples in fig. 7, and in fig. 9 we compare the MMD between the model samples and the inference process samples at various times $t \in [0, T]$. The inference process is defined as

$$dx = (-x^3 + v)dt \tag{159}$$

$$dv = -\gamma v dt + \sqrt{2\gamma D} d\mathbf{w}_t, \qquad t \in [0, T] \tag{160}$$

where $\gamma = 0.1, D = 1.0$ and $T = 5.0$ with initial conditions $x_0, v_0 \sim \mathcal{N}(0, 1)$. We generate samples from the score trained by the local DSM and ISM objectives using the probability-flow ODE:

$$\frac{d}{dt}\mathbf{y}_t = f(\mathbf{y}_t, t) - \frac{1}{2} g g^\top s_\theta(\mathbf{y}_t, t) \tag{161}$$

where $\mathbf{y} = (x, v)^\top$. Note that when $s_\theta = \nabla_{\mathbf{y}} \log q(\mathbf{y}_t)$, then $q_{\text{ODE}} = q_{\text{SDE}}$, that is the distribution of the inference process and the PF-ODE match at any time $t \in [0, T]$.

G.2. Interacting Particle System

In this section we plot the MMD between PF-ODE samples from the local DSM and ISM trained model and the diffusion process, defined in eq. (24), samples between $t \in [0, 10]$. We note that for all $t \in [0, 10]$, the local DSM trained models has a lower MMD.

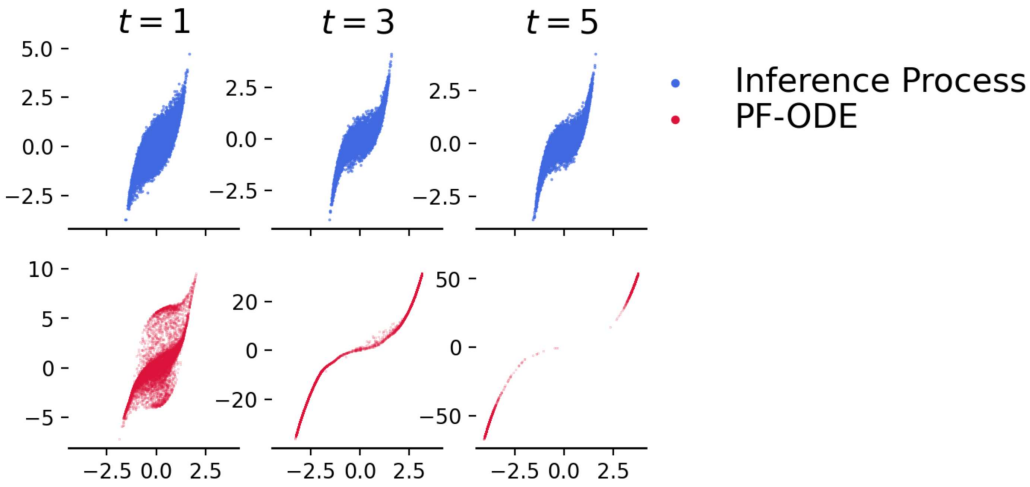

Figure 7: **ISM Samples at** $t \in \{1, 3, 5\}$. Here we compare samples from the process defined in eq. (160) on the top panel and samples from ISM trained model on the bottom panel. The samples from the PF-ODE start diverging and do not match the inference process' distribution.

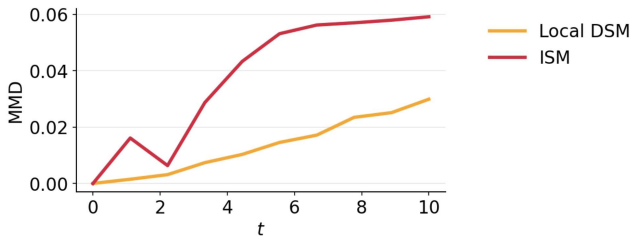

Figure 8: **MMD for** $t \in [0, 10]$

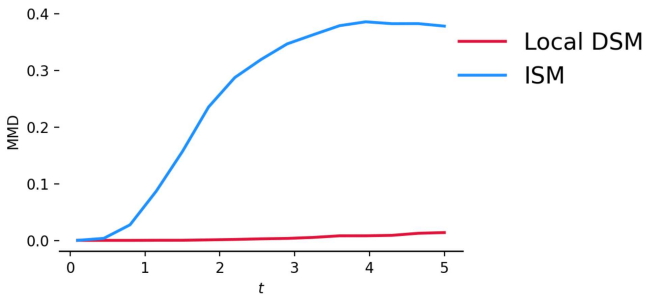

Figure 9: Here we compare the MMD metric between model generated samples and the inference process samples at various time slices. We observe that both models have an increasing trend but the ISM model sample quality deteriorates rapidly compared to the local DSM trained model.