Probability flow solution of the Fokker–Planck equation

IOP Publishing Mach. Learn.: Sci. Technol. 4 (2023) 035012 https://doi.org/10.1088/2632-2153/ace2aa

MACHINE LEARNING
Science and Technology

OPEN ACCESS

RECEIVED
18 February 2023

REVISED
14 June 2023

ACCEPTED FOR PUBLICATION
28 June 2023

PUBLISHED
25 July 2023

PAPER

Probability flow solution of the Fokker–Planck equation

Nicholas M Boffi* and **Eric Vanden-Eijnden**

Courant Institute of Mathematical Sciences, New York University, New York, NY 10012, United States of America
* Author to whom any correspondence should be addressed.

E-mail: boffi@cims.nyu.edu

Keywords: Fokker–Planck equation, statistical physics, stochastic dynamics, high-dimensional scientific computing, diffusion models

Abstract

The method of choice for integrating the time-dependent Fokker–Planck equation (FPE) in high-dimension is to generate samples from the solution via integration of the associated stochastic differential equation (SDE). Here, we study an alternative scheme based on integrating an ordinary differential equation that describes the flow of probability. Acting as a transport map, this equation deterministically pushes samples from the initial density onto samples from the solution at any later time. Unlike integration of the stochastic dynamics, the method has the advantage of giving direct access to quantities that are challenging to estimate from trajectories alone, such as the probability current, the density itself, and its entropy. The probability flow equation depends on the gradient of the logarithm of the solution (its 'score'), and so is *a-priori* unknown. To resolve this dependence, we model the score with a deep neural network that is learned on-the-fly by propagating a set of samples according to the instantaneous probability current. We show theoretically that the proposed approach controls the Kullback–Leibler (KL) divergence from the learned solution to the target, while learning on external samples from the SDE does not control either direction of the KL divergence. Empirically, we consider several high-dimensional FPEs from the physics of interacting particle systems. We find that the method accurately matches analytical solutions when they are available as well as moments computed via Monte-Carlo when they are not. Moreover, the method offers compelling predictions for the global entropy production rate that out-perform those obtained from learning on stochastic trajectories, and can effectively capture non-equilibrium steady-state probability currents over long time intervals.

1. Introduction

The time evolution of many dynamical processes occurring in the natural sciences, engineering, economics, and statistics are naturally described in the language of stochastic differential equations (SDE) [12, 14, 40]. Typically, one is interested in the probability density function (PDF) of these processes, which describes the probability that the system will occupy a given state at a given time. The density can be obtained as the solution to a Fokker–Planck equation (FPE), which can generically be written as [1, 45]

$$\partial_t \rho_t^*(x) = -\nabla \cdot (b_t(x)\rho_t^*(x) - D_t(x)\nabla \rho_t^*(x)), \qquad x \in \Omega \subseteq \mathbb{R}^d, \tag{1}$$

where $\rho_t^*(x) \in \mathbb{R}_{\geqslant 0}$ denotes the value of the density at time t, $b_t(x) \in \mathbb{R}^d$ is a vector field known as the drift, and $D_t(x) \in \mathbb{R}^{d\times d}$ is a positive-semidefinite tensor known as the diffusion matrix. (1) must be solved for $t \geqslant 0$ from some initial condition $\rho_{t=0}^*(x) = \rho_0(x)$, but in all but the simplest cases, the solution is not available analytically and can only be approximated via numerical integration.

High-dimensionality. For many systems of interest—such as interacting particle systems in statistical physics [4, 56], stochastic control systems [26], and models in mathematical finance [40]—the dimensionality d can be very large. This renders standard numerical methods for partial differential equations inapplicable, which become infeasible for d as small as five or six due to an exponential scaling of

the computational complexity with d. The standard solution to this problem is a Monte-Carlo approach, whereby the SDE associated with (1)

$$\mathrm{d}x_t = b_t(x_t)\mathrm{d}t + \nabla \cdot D_t(x_t)\mathrm{d}t + \sqrt{2}\sigma_t(x_t)\mathrm{d}W_t, \qquad x_0 \sim \rho_0 \tag{2}$$

is evolved via numerical integration to obtain a large number n of trajectories [24]. In (2), $\sigma_t(x)$ satisfies $\sigma_t(x)\sigma_t^{\mathsf{T}}(x) = D_t(x)$ and W_t is a standard Brownian motion on $\mathbb{R}^d$. Assuming that we can draw samples $\{x_0^i\}_{i=1}^n$ from the initial PDF ρ_0, simulation of (2) enables the estimation of expectations via empirical averages

$$\int_\Omega \phi(x)\rho_t^*(x)\mathrm{d}x \approx \frac{1}{n}\sum_{i=1}^n \phi(x_t^i), \tag{3}$$

where $\phi : \Omega \to \mathbb{R}$ is an observable of interest. While widely used, this method only provides samples from ρ_t^*, and hence other quantities of interest like the value of ρ_t^* itself or the time-dependent differential entropy of the system $H_t = -\int_\Omega \log \rho_t^*(x)\rho_t^*(x)\mathrm{d}x$ require sophisticated interpolation methods that typically do not scale well to high-dimension.

A transport map approach. Another possibility, building on recent theoretical advances that connect transportation of measures to the FPE [22], is to recast (1) as the transport equation [49, 60]

$$\partial_t \rho_t^*(x) = -\nabla \cdot (v_t^*(x)\rho_t^*(x)), \qquad \rho_{t=0}^* = \rho_0 \tag{4}$$

where we have defined the velocity field

$$v_t^*(x) = b_t(x) - D_t(x)\nabla \log \rho_t^*(x). \tag{5}$$

This formulation reveals that ρ_t^* can be viewed as the pushforward of ρ_0 under the flow map $X_{\tau,t}^*(\cdot)$ of the ordinary differential equation

$$\frac{\mathrm{d}}{\mathrm{d}t}X_{\tau,t}^*(x) = v_t^*(X_{\tau,t}^*(x)), \qquad X_{\tau,\tau}^*(x) = x, \quad t,\tau \geqslant 0. \tag{6}$$

Equation (6) is known as the probability flow equation, and its solution has the remarkable property that if x is a sample from ρ_0, then $X_{0,t}^*(x)$ will be a sample from ρ_t^*. Viewing $X_{\tau,t}^* : \Omega \to \Omega$ as a transport map and letting $\sharp$ denote the push-forward operation, $\rho_t^* = X_{0,t}^*\sharp\rho_0$ can be evaluated at any position in Ω via the change of variables formula [49, 60]

$$\rho_t^*(x) = \rho_0(X_{t,0}^*(x)) \exp\left(-\int_0^t \nabla \cdot v_\tau^*(X_{t,\tau}^*(x))\mathrm{d}\tau\right) \tag{7}$$

where $X_{t,0}^*(x)$ is obtained by solving (6) backward from some given x. Importantly, access to the PDF as provided by (7) immediately gives the ability to compute quantities such as the probability current or the entropy; by contrast, this capability is absent when directly simulating the SDE.

Learning the flow. The simplicity of the probability flow equation (6) is somewhat deceptive, because the velocity v_t^* depends explicitly on the solution ρ_t^* to the FPE (1). Nevertheless, recent work in generative modeling via score-based diffusion [52–54] has shown that it is possible to use deep neural networks to estimate v_t^*, or equivalently the so-called score $\nabla \log \rho_t^*$ of the solution density. Here, we introduce a variant of score-based diffusion modeling in which the score is learned on-the-fly over samples generated by the probability flow equation itself. The method is self-contained and enables us to bypass simulation of the SDE entirely; moreover, we provide both empirical and theoretical evidence that the resulting self-consistent training procedure offers improved performance when compared to training via samples produced from simulation of the SDE.

1.1. Contributions

Our contributions are both theoretical and computational:

- We provide a bound on the Kullback–Leibler (KL) divergence from the estimate ρ_t produced via an approximate velocity field v_t to the target ρ_t^*. This bound motivates our approach, and shows that minimizing the discrepancy between the learned score and the score of the push-forward distribution systematically improves the accuracy of ρ_t.

- Based on this bound, we introduce two optimization problems that can be used to learn the velocity field (5) in the transport equation (4) so that its solution coincides with that of the FPE (1). Due to its similarities with score-based diffusion approaches in generative modeling (SBDM), we call the resulting method score-based transport modeling (SBTM).
- We provide specific estimators for quantities that can be computed via SBTM but are not directly available from samples alone, like point-wise evaluation of ρ_t itself, the differential entropy, and the probability current.
- We test SBTM on several examples involving interacting particles that pairwise repel but are kept close by common attraction to a moving trap. In these systems, the FPE is high-dimensional due to the large number of particles, which vary from 5 to 50 in the examples below. Problems of this type frequently appear in the molecular dynamics of externally-driven soft matter systems [13, 56]. We show that our method can be used to accurately compute the entropy production rate, a quantity of interest in the active matter community [39], as it quantifies the out-of-equilibrium nature of the system's dynamics.

1.2. Notation and assumptions

Throughout, we assume that the stochastic process (2) evolves over $\Omega = \mathbb{R}^d$, though our results can easily be extended to domains with either reflecting [30] or periodic boundary conditions. We let $|\cdot| : \mathbb{R}^d \to \mathbb{R}_{\geqslant 0}$ denote the Euclidean norm on vectors and $|\cdot|_F : \mathbb{R}^{d\times d} \to \mathbb{R}_{\geqslant 0}$ denote the Fröbenius norm on matrices. For our theory, we assume that the drift vector $b_t : \mathbb{R}^d \to \mathbb{R}^d$ and the diffusion tensor $D_t : \mathbb{R}^d \to \mathbb{R}^{d\times d}$ with $D_t(x) = \sigma_t(x)\sigma_t(x)^\mathsf{T}$ are both twice-differentiable in x for each t and satisfy, for some fixed $C > 0$, $L > 0$, and $T > 0$

$$\begin{aligned} |b_t(x)| + |\sigma_t(x)|_F &\leqslant C(1+|x|) \qquad \forall\,(x,t) \in \mathbb{R}^n \times [0,T], \\ |b(t,x) - b(t,y)| + |\sigma(t,x) - \sigma(t,y)|_F &\leqslant L|x-y| \qquad \forall\,(x,y,t) \in \mathbb{R}^n \times \mathbb{R}^n \times [0,T], \end{aligned} \tag{8}$$

so that the solution to the SDE (2) is well-defined for $t \in [0,T]$ [40]. We further assume that the initial PDF ρ_0 is three-times differentiable, positive everywhere on Ω, and such that $H_0 = -\int_\Omega \log\rho_0(x)\rho_0(x)\mathrm{d}x < \infty$; ρ_t^* then enjoys the same properties at all times $t \in [0,T]$. Finally, we assume that $\log\rho_t^*$ is K-smooth globally for $(t,x) \in [0,\infty) \times \Omega$, i.e.

$$\exists K > 0 : \quad \forall (t,x) \in [0,\infty)\times\Omega \quad |\nabla\log\rho_t^*(x) - \nabla\log\rho_t^*(y)| \leqslant K|x-y|. \tag{9}$$

This technical assumption is needed to guarantee global existence and uniqueness of the solution of the probability flow equation. Throughout, we use the shorthand notation $\dot{y}_t = \frac{\mathrm{d}}{\mathrm{d}t}y_t$ interchangeably for a time-dependent quantity y_t.

2. Related work

Score matching. Our approach builds directly on the toolbox of score matching originally developed by Hyvärinen [17–20] and more recently extended in the context of diffusion-based generative modeling [7, 10, 38, 52, 53, 55]. These approaches assume access to training samples from the target distribution (e.g. in the form of examples of natural images). Here, we bypass this need and use the probability flow equation to obtain the samples needed to learn an approximation of the score. Lu *et al* [34] recently showed that using the transport equation (10) with a velocity field learned via SBDM can lead to inaccuracies in the likelihood unless higher-order score terms are well-approximated. Proposition 1 shows that the self-consistent approach used in SBTM solves these issues and ensures a systematic approximation of the target ρ_t^*. Lai *et al* [27] recently used a similar idea to improve sample quality with score-based probability flow equations in generative modeling.

Density estimation and Bayesian inference. Our method shares commonalities with transport map-based approaches [37] for density estimation and variational inference [2, 62] such as normalizing flows [16, 25, 41, 44, 57, 58]. Moreover, because expectations are approximated over a set of samples according to (3), the method also inherits elements of classical 'particle-based' approaches for density estimation such as Markov chain Monte Carlo [46] and sequential Monte Carlo [6, 9].

Our approach is also reminiscent of a recent line of work in Bayesian inference that aims to combine the strengths of particle methods with those of variational approximations [5, 48]. In particular, the method we propose bears some similarity with Stein variational gradient descent (SVGD) [31–33] (see also [28, 35]), in that both methods approximate the target distribution via *deterministic* propagation of a set of samples. The key differences are that (i) our method learns the map used to propagate the samples, while the map in

SVGD corresponds to optimization of the kernelized Stein discrepancy, and (ii) the methods have distinct goals, as we are interested in capturing the dynamical evolution of ρ_t^* rather than sampling from an equilibrium density. Indeed, many of the examples we consider do not have an equilibrium density, i.e. $\lim_{t\to\infty}\rho_t^*$ does not exist.

Approaches for solving the FPE. Most closely connected to our paper are the works by Maoutsa *et al* [36] and Shen *et al* [50], who similarly propose to bypass the SDE through use of the probability flow equation, building on earlier work by Degond and Mustieles [8] and Russo [47]. The critical differences between Maoutsa *et al* [36] and our approach are that they perform estimation over a linear space or a reproducing kernel Hilbert space rather than over the significantly richer class of neural networks, and that they train using the original score matching loss of Hyvärinen [18], while the use of neural networks requires the introduction of regularized variants. Because of this, [36] studies systems of dimension less than or equal to five; in contrast, we study systems with dimensionality as high as 100.

Concurrently to our work, Shen *et al* [50] proposed a variational problem similar to SBTM. A key difference is that SBTM is not limited to FPEs that can be viewed as a gradient flow in the Wasserstein metric over some energy (i.e. the drift term in the SDE (2) need not be the gradient of a potential), and that it allows for spatially-dependent and rank-deficient diffusion matrices. Moreover, our theoretical results are similar, but by avoiding the use of costly Sobolev norms lead to a practical optimization problem that we show can be solved in high dimension and over long times. In a follow-up to Shen *et al* [50] and our present work, Li *et al* [29] propose an algorithm that can be seen as an expectation-maximization algorithm for the loss function in (15), which avoids calculation of G_t according to equation (13).

Neural-network solutions to PDEs. Our approach can also be viewed as an alternative to recent neural network-based methods for the solution of partial differential equations (see e.g. [3, 11, 15, 42, 51]). Unlike these existing approaches, our method is tailored to the solution of the FPE and guarantees that the solution is a valid probability density. Our approach is fundamentally Lagrangian in nature, which has the advantage that it only involves learning quantities locally at the positions of a set of evolving samples; this is naturally conducive to efficient scaling for high-dimensional systems.

3. Methodology

3.1. Score-based transport modeling

Let $s_t : \Omega \to \mathbb{R}^d$ denote an approximation to the score of the target $\nabla \log \rho_t^*$, and consider the solution $\rho_t : \Omega \to \mathbb{R}_{\geqslant 0}$ to the transport equation

$$\partial_t \rho_t(x) = -\nabla \cdot (v_t(x)\rho_t(x)) \qquad \text{with} \quad v_t(x) = b_t(x) - D_t(x)s_t(x), \tag{10}$$

subject to the initial condition $\rho_{t=0} = \rho_0$. Our goal is to develop a variational principle that may be used to adjust s_t so that ρ_t tracks ρ_t^*. Our approach is based on the following inequality, whose proof may be found in appendix B.1:

Proposition 1 (Control of the KL divergence). *Assume that the conditions listed in section 1.2 hold. Let ρ_t denote the solution to the transport equation (10), and let ρ_t^* denote the solution to the FPE (1). Assume that $\rho_{t=0}(x) = \rho_{t=0}^*(x) = \rho_0(x)$ for all $x \in \Omega$. Then*

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathsf{KL}(\rho_t \,\|\, \rho_t^*) \leqslant \frac{1}{2}\int_\Omega |s_t(x) - \nabla \log \rho_t(x)|^2_{D_t(x)}\, \rho_t(x)\mathrm{d}x, \tag{11}$$

where $|\cdot|^2_{D_t(x)} = \langle \cdot, D_t(x)\cdot\rangle$.

In particular, (11) implies that for any $T \in [0,\infty)$ we have explicit control on the KL divergence

$$\mathsf{KL}(\rho_T \,\|\, \rho_T^*) \leqslant \frac{1}{2}\int_0^T\int_\Omega |s_t(x) - \nabla \log \rho_t(x)|^2_{D_t(x)}\, \rho_t(x)\mathrm{d}x\mathrm{d}t. \tag{12}$$

Remarkably, (12) only depends on the approximate ρ_t and does not include ρ_t^*: it states that the accuracy of ρ_t as an approximation of ρ_t^* can be improved by enforcing agreement between s_t and $\nabla \log \rho_t$. This means that we can optimize (12) without making use of external data from ρ_t^*, which offers a self-consistent objective function to learn the score s_t using (10) alone.

The primary difficulty with this approach is that ρ_t must be considered as a functional of s_t, since the velocity v_t used in (10) depends on s_t. To render the resulting minimization of the right-hand side of (12)

practical, we can exploit that (10) can be solved via the method of characteristics, as summarized in appendix A. Specifically, if $\dot{X}_t(x) = v_t(X_t(x))$ is the probability flow equation associated with the velocity v_t, then $\rho_t = X_t \sharp \rho_0$. This means that the expectation of any function $\phi(x)$ over $\rho_t(x)$ can be expressed as the expectation of $\phi_t(X_t(x))$ over $\rho_0(x)$. Observing that the score of the solution to (10) along trajectories of the probability flow $\nabla \log \rho_t(X_t(x))$ solves a closed equation leads to the following proposition.

Proposition 2 (Score-based transport modeling). *Assume that the conditions listed in section 1.2 hold. Define $v_t(x) = b_t(x) - D_t(x)s_t(x)$ and consider*

$$\begin{aligned} \dot{X}_t(x) &= v_t(X_t(x)), & X_0(x) &= x, \\ \dot{G}_t(x) &= -[\nabla v_t(X_t(x))]^{\mathsf{T}} G_t(x) - \nabla \nabla \cdot v_t(X_t(x)), & G_0(x) &= \nabla \log \rho_0(x). \end{aligned} \tag{13}$$

Then $\rho_t = X_t \sharp \rho_0$ solves (10), the equality $G_t(x) = \nabla \log \rho_t(X_t(x))$ holds, and for any $T \in [0, \infty)$

$$\mathsf{KL}(X_T \sharp \rho_0 \,\|\, \rho_T^*) \leqslant \frac{1}{2} \int_0^T \int_\Omega |s_t(X_t(x)) - G_t(x)|^2_{D_t(X_t(x))} \, \rho_0(x) \mathrm{d}x \mathrm{d}t. \tag{14}$$

Moreover, if s_t^* is a minimizer of the constrained optimization problem

$$\min_s \int_0^T \int_\Omega |s_t(X_t(x)) - G_t(x)|^2_{D_t(X_t(x))} \, \rho_0(x) \mathrm{d}x \mathrm{d}t \quad \textit{subject to } (13) \tag{15}$$

then $D_t(x)s_t^*(x) = D_t(x)\nabla \log \rho_t^*(x)$ where ρ_t^* solves the FPE (1). The map X_t^* associated to any minimizer is a transport map from ρ_0 to ρ_t^*, i.e.

$$x \sim \rho_0 \qquad \textit{implies that} \qquad X_t^*(x) \sim \rho_t^*, \qquad \forall t \in [0, T]. \tag{16}$$

Proposition 2 is proven in appendix B.3. The result also holds with a standard Euclidean norm replacing the diffusion-weighted norm, in which case the minimizer is unique and is given by $s_t^*(x) = \nabla \log \rho_t^*(x)$. In the special case when the SDE is an Ornstein–Uhlenbeck (OU) process, the score and the equations for both X_t and G_t can be written explicitly; they are studied in appendix C.

In practice, the objective in (15) can be estimated empirically by generating samples from ρ_0 and solving the equations for $X_t(x)$ and $G_t(x)$ with $x \sim \rho_0$. The constrained minimization problem (15) can then in principle be solved with gradient-based techniques via the adjoint method. The corresponding equations are written in appendix B.3, but they involve fourth-order spatial derivatives that are computationally expensive to compute via automatic differentiation. Moreover, each gradient step requires solving a system of ordinary differential equations whose dimensionality is equal to the number of samples used to compute expectations times the dimension of (1). Instead, we now develop a sequential timestepping procedure that avoids these difficulties entirely, and as a byproduct can scale to arbitrarily long time windows.

3.2. Sequential score-based transport modeling

An alternative to the constrained minimization in proposition 2 is to consider an approach whereby the score s_t is obtained independently at each time to ensure that $\mathsf{KL}(\rho_t \,\|\, \rho_t^*)$ remains small. This suggests choosing s_t to minimize $\frac{\mathrm{d}}{\mathrm{d}t}\mathsf{KL}(\rho_t \,\|\, \rho_t^*)$, which admits a simple closed-form bound, as shown in proposition 1. While this explicit form can be used directly, an application of Stein's identity recovers an implicit objective analogous to Hyvärinen score-matching that is equivalent to minimizing $\frac{\mathrm{d}}{\mathrm{d}t}\mathsf{KL}(\rho_t \,\|\, \rho_t^*)$ but obviates the calculation of G_t. Expanding the square in (11) and applying $\int_\Omega s_t(x)^{\mathsf{T}} \nabla \log \rho_t(x) \, \rho_t(x) \mathrm{d}x = -\int_\Omega \nabla \cdot s_t(x) \, \rho_t(x) \mathrm{d}x$, we may write

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}t}\mathsf{KL}(\rho_t \,\|\, \rho_t^*) \leqslant \frac{1}{2} \int_\Omega \left(|s_t(X_t(x))|^2_{D_t(X_t(x))} + 2\nabla \cdot (D_t(X_t(x))s_t(X_t(x))) \right) \rho_0(x) \mathrm{d}x \\ + \frac{1}{2} \int |G_t(x)|^2 \rho_0(x) \mathrm{d}x. \end{aligned}$$

Because $\nabla \log \rho_t(X_t(x)) = G_t(x)$ is independent of s_t, we may neglect the corresponding square term during optimization. This leads to a simple and comparatively less expensive way to build the pushforward X_t^* such that $X_t^* \sharp \rho_0 = \rho_t^*$ sequentially in time, as stated in the following proposition.

Proposition 3 (Sequential SBTM). *In the same setting as proposition 2, let $X_t(x)$ solve the first equation in (13) with $v_t(x) = b_t(x) - D_t(x)s_t(x)$. Let s_t be obtained via*

$$\min_{s_t} \int_\Omega \left(|s_t(X_t(x))|^2_{D_t(X_t(x))} + 2\nabla \cdot (D_t(X_t(x))s_t(X_t(x))) \right) \rho_0(x)\mathrm{d}x. \tag{17}$$

Then, each minimizer s_t^* of (17) satisfies $D_t(x)s_t^*(x) = D_t(x)\nabla \log \rho_t^*(x)$ where ρ_t^* is the solution to (1). Moreover, the map X_t^* associated to s_t^* is a transport map from ρ_0 to ρ_t^*.

Proposition 3 is proven in appendix B.4. Critically, (17) is no longer a constrained optimization problem. Given the current value of X_t at any time t, we can obtain s_t via direct minimization of the objective in (17). Given s_t, we may compute the right-hand side of (13) and propagate X_t (and possibly G_t) forward in time. The resulting procedure, which alternates between self-consistent score estimation and sample propagation, is presented in algorithm 1 for the choice of a forward-Euler integration routine in time. The output of the method produces a feasible solution $\rho_t = X_t \sharp \rho_0$ for (15) because $\dot{X}_t$ satisfies the first constraint in (13) by construction. Moreover, because the method controls $\frac{\mathrm{d}}{\mathrm{d}t}\mathsf{KL}(\rho_t \,\|\, \rho_t^*)$ at each t, it also controls $\mathsf{KL}(\rho_t \,\|\, \rho_t^*)$ by integration; an *a-posteriori* bound can be obtained by calculating $G_t(x)$ according to the second equation in (13) and computing the loss in (15). A few remarks on algorithm 1 are now in order.

Algorithm 1. Sequential score-based transport modeling.

1: **Input**: An initial time $t_0 \in \mathbb{R}_{\geqslant 0}$. A set of n samples $\{x_i\}_{i=1}^n$ from ρ_{t_0}. A set of N_T timesteps $\{\Delta t_k\}_{k=0}^{N_T-1}$.
2: Initialize sample locations $X^i_{t_0} = x_i$ for $i = 1, \ldots, n$.
3: **for** $k = 0, \ldots, N_t - 1$ **do**
4: Optimize: $s_{t_k} = \mathrm{argmin}_s \frac{1}{n}\sum_{i=1}^n \left[|s(X^i_{t_k})|^2_{D_{t_k}(X^i_{t_k})} + 2\nabla \cdot \left(D_{t_k}(X^i_{t_k})s(X^i_{t_k})\right)\right]$.
5: Propagate samples:
$X^i_{t_{k+1}} = X^i_{t_k} + \Delta t_k \left(b_{t_k}(X^i_{t_k}) - D_{t_k}(X^i_{t_k})s_{t_k}(X^i_{t_k})\right)$.
6: Set $t_{k+1} = t_k + \Delta t_k$.
7: **Output**: A set of n samples $\{X^i_{t_k}\}_{i=1}^n$ from ρ_{t_k} and the score $\{s_{t_k}(X^i_{t_k})\}_{i=1}^n$ for all $\{t_k\}_{k=0}^{N_T}$.

Higher-order integrators. Algorithm 1 is stated for choice of forward-Euler integration for simplicity. In practice, any off-the-shelf integrator can be used, such as an adaptive Runge–Kutta method, by temporal discretization of the dynamics

$$\dot{X}_t(x) = v_t(X_t(x))$$
$$s_t = \mathrm{argmin}_s \int_\Omega \left(|s(X_t(x))|^2_{D_t(X_t(x))} + 2\nabla \cdot (D_t(X_t(x))s(X_t(x))) \right) \rho_0(x)\mathrm{d}x$$

and spatial discretization of the expectation over a set of samples propagated according to the equation for $X_t(x)$. In practice, the minimization can be performed over a parametric class of functions such as neural networks via a few steps of gradient descent.

Divergence computation. To avoid computation of the divergence—which can be costly for neural networks with high input dimension—we can use the denoising score matching loss function introduced by [61], which we discuss in appendix B.6. Empirically, we find that use of either the denoising objective or explicit derivative regularization is necessary for stable training to avoid overfitting to the training data; the level of regularization (or the noise scale in the denoising objective) can be decreased as the size of the dataset increases.

Time-dependence. When optimizing over a parametric class of functions, the score can be taken to be explicitly time-dependent, or the time-dependence can originate only through the parameters. In either case, all required outputs can be computed on-the-fly to avoid saving the entire history of parameters, which could be memory-intensive for large neural networks. If a time-dependent architecture is used, the method is amenable to online learning by randomly re-drawing initial conditions and optimizing over the resulting trajectory. In the numerical experiments below, we consider time-independent models with time-dependent parameters, because we found them to be sufficient.

SBTM vs. Sequential SBTM. Given the simplicity of the optimization problem (17), one may wonder if (15) is useful in practice, or if it is simply a stepping stone to arrive at (17). The primary difference is that (15) offers global control on the discrepancy between s_t and $\nabla \log \rho_t$ over $t \in [0, T]$, in the sense that it directly minimizes the time-integrated error, while (17) controls a local truncation error that could lead to the accumulation of learning and time-discretization errors. In the numerical examples below, we took the

timestep Δt sufficiently small, and the number of samples n sufficiently large, that we did not observe any accumulation of error. Nevertheless, (15) may allow for more accurate approximation, because the loss is exactly minimized at zero. Moreover, the higher-order derivatives contained in $\nabla\nabla\cdot(D_t s_t)$ must remain well-behaved when using (15) because this term appears in the definition of $\dot{G}_t$, while (17) only contains $\nabla\cdot(D_t s_t)$.

3.3. Learning on external data

An alternative to the sequential procedure outlined here would be to generate samples from the target ρ_t^* via simulation of the associated SDE (2), and then to approximate the score $\nabla\log\rho_t^*$ via minimization of the loss

$$\int_0^T\int_\Omega (|s_t(x)-\nabla\log\rho_t^*(x)|^2_{D_t(x)}\rho_t^*(x)\mathrm{d}x\mathrm{d}t, \tag{18}$$

similar to SBDM. ρ_t can be computed as in SBTM or sequential SBTM by simulation of the probability flow with the learned s_t. As we now show, neither $\mathsf{KL}(\rho_t\,\|\,\rho_t^*)$ nor $\mathsf{KL}(\rho_t^*\,\|\,\rho_t)$ are controlled when using this procedure.

Proposition 4 (Learning on external data). *Let $\rho_t:\Omega\to\mathbb{R}_{>0}$ solve (10), and let $\rho_t^*:\Omega\to\mathbb{R}_{>0}$ solve (1). Then, the following equality holds*

$$\begin{aligned}\mathsf{KL}(\rho_T^*\,\|\,\rho_T) = &\int_0^T\int_\Omega |s_t(x)-\nabla\log\rho_t^*(x)|^2_{D_t(x)}\rho_t^*(x)\mathrm{d}x\mathrm{d}t\\ &+\int_0^T\int_\Omega (\nabla\log\rho_t(x)-s_t(x))^{\mathsf{T}}D_t(x)\,(s_t(x)-\nabla\log\rho_t^*(x))\,\rho_t^*(x)\mathrm{d}x\mathrm{d}t.\end{aligned} \tag{19}$$

Proposition 4 shows that minimizing the error between s_t and $\nabla\log\rho_t^*$ on samples of ρ_t^* leaves a remainder term, because in general $\nabla\log\rho_t\neq s_t$. Young's inequality gives the simple upper bound

$$\begin{aligned}\mathsf{KL}(\rho_T^*\,\|\,\rho_T) \leqslant &\frac{3}{2}\int_0^T\int_\Omega |s_t(x)-\nabla\log\rho_t^*(x)|^2_{D_t(x)}\rho_t^*(x)\mathrm{d}x\mathrm{d}t\\ &+\frac{1}{2}\int_0^T\int_\Omega |s_t(x)-\nabla\log\rho_t(x)|^2_{D_t(x)}\rho_t^*(x)\mathrm{d}x\mathrm{d}t.\end{aligned} \tag{20}$$

However, controlling the above quantity requires enforcing agreement between s_t and $\nabla\log\rho_t$ in addition to s_t and $\nabla\log\rho_t^*$, which is precisely the idea of SBTM. Empirically, we find in our numerical experiments that training on external data alone is significantly less stable than sequential SBTM. In particular, and importantly for the applications we consider, we could not stably estimate the trajectory of the entropy production rate using a score model learned from the SDE with the same number of samples as used for sequential SBTM.

4. Numerical experiments

In the following, we study two high-dimensional examples from the physics of interacting particle systems, where the spatial variable of the FPE (1) can be written as $x=\left(x^{(1)},x^{(2)},\ldots,x^{(N)}\right)^{\mathsf{T}}$ with each $x^{(i)}\in\mathbb{R}^{\bar{d}}$. Here, $\bar{d}$ describes a lower-dimensional ambient space, e.g. $\bar{d}=2$, so that the dimensionality of the FPE $d=N\bar{d}$ will be high if the number of particles N is even moderate[1]. The still figures shown in this section do not fully depict the complexity of the interacting particle dynamics, and we encourage the reader to view the movies available here. With a timestep $\Delta t=10^{-3}$, a horizon $T=10$, and a fixed $nN\bar{d}=10^5$, we find that the sequential SBTM procedure takes around two hours for each simulation on a single NVIDIA RTX8000 GPU. In addition, study a low-dimensional example from the physics of active matter, which highlights the ability of sequential SBTM to remain stable over long times and to capture non-equilibrium probability currents. Our second and third examples go beyond the conditions required for existence and uniqueness assumed in section 1.2; nevertheless, due to the presence of a confining potential, solutions to the SDE (2), FPE (1), and probability flow (6) exist, as our numerical results show.

[1] We would like to emphasize at this stage the difference between the number of physical particles N, which is a parameter for the system under study and sets the dimensionality of the resulting FPE, and the number of algorithmic samples n, which is a hyper-parameter that can be chosen at will to improve the accuracy of the learning.

4.1. Harmonically interacting particles in a harmonic trap

Setup. Here we study a problem that admits a tractable analytical solution for direct comparison. We consider N two-dimensional particles ($\bar{d} = 2$) that repel according to a harmonic interaction but experience harmonic attraction towards a moving trap $\beta_t \in \mathbb{R}^2$. The motion of the physical particles is governed by the stochastic dynamics

$$\mathrm{d}x_t^{(i)} = (\beta_t - x_t^{(i)})\mathrm{d}t + \alpha\Big(x_t^{(i)} - \frac{1}{N}\sum_{j=1}^{N} x_t^{(j)}\Big)\mathrm{d}t + \sqrt{2D}\,\mathrm{d}W_t^{(i)}, \quad i = 1,\dots,N \tag{21}$$

where $\alpha \in (0,1)$ is a fixed coefficient that sets the magnitude of the repulsion and each $x_0^{(i)} \sim \rho_0$. The dynamics (21) is an OU process in the extended variable $x \in \mathbb{R}^{\bar{d}N}$ with block components $x^{(i)}$. Assuming a Gaussian initial condition, the solution to the FPE associated with (21) is a Gaussian for all time and hence can be characterized entirely by its mean m_t and covariance C_t. These can be obtained analytically (appendices C and D), which facilitates a quantitative comparison to the learned model. The differential entropy S_t is given by

$$H_t = \tfrac{1}{2}\bar{d}N(\log(2\pi) + 1) + \tfrac{1}{2}\log\det C_t. \tag{22}$$

In the experiments, we take $\beta_t = a(\cos\pi\omega t, \sin\pi\omega t)^\mathsf{T}$ with $a = 2$, $\omega = 1$, $D = 0.25$, $\alpha = 0.5$, and $N = 50$, giving rise to a 100-dimensional FPE. The particles are initialized from an isotropic Gaussian with mean β_0 (the initial trap position) and variance $\sigma_0^2 = 0.25$.

Network architecture. We take $s_t(x) = -\nabla U_{\theta_t}(x)$, where the potential $U_{\theta_t}(\cdot)$ is given as a sum of one- and two-particle terms

$$U_{\theta_t}\big(x^{(1)},\dots,x^{(N)}\big) = \sum_{i=1}^{N} U_{\theta_t,1}\big(x^{(i)}\big) + \frac{1}{N}\sum_{\substack{i,j=1\\ i\neq j}}^{N} U_{\theta_t,2}\big(x^{(i)},x^{(j)}\big), \tag{23}$$

which ensures permutation symmetry amongst the physical particles by direct summation over all pairs. Modeling at the level of the potential introduces an additional gradient into the loss function, but makes it simple to enforce permutation symmetry; moreover, by writing the potential as a sum of one- and two-particle terms, the dimensionality of the function estimation problem is reduced. As motivation for this choice of architecture, we show in appendix D.1 that the class of scores representable by (23) contains the analytical score for the harmonic problem considered in this section. To obtain the parameters $\theta_{t_k+\Delta t_k}$, we perform a warm start and initialize from θ_{t_k}, which reduces the number of optimization steps that need to be performed at each iteration. All networks are taken to be multi-layer perceptrons with the `swish` activation function [43]; further details on the architectures used can be found in appendix D.

Quantitative comparison. For a quantitative comparison between the learned model and the exact solution, we study the empirical covariance Σ over the samples and the entropy production rate $\frac{\mathrm{d}H_t}{\mathrm{d}t}$. Because an analytical solution is available for this system, we may also compute the target $\nabla\log\rho_t(x) = -C_t^{-1}(x - m_t)$ and measure the goodness of fit via the relative Fisher divergence

$$\frac{\int_\Omega |s_t(x) - \nabla\log\rho_t(x)|^2\bar{\rho}(x)\mathrm{d}x}{\int_\Omega |\nabla\log\rho_t(x)|^2\bar{\rho}(x)\mathrm{d}x}. \tag{24}$$

In equation (24), $\bar{\rho}$ can be taken to be equal to the current empirical estimate of ρ_t (the training data), or estimated using samples from the SDE(the SDE data).

Results. The representation of the dynamics (21) in terms of the flow of probability leads to an intuitive deterministic motion that accurately captures the statistics of the underlying stochastic process. Snapshots of particle trajectories from the learned probability flow (6), the SDE (21), and the noise-free equation obtained by setting $D = 0$ in (21) are shown in figure 1(A).

Results for this quantitative comparison are shown in figure 1(B). The learned model accurately predicts the entropy production rate of the system and minimizes the relative metric (24) to the order of 10^{-2}. The noise-free system incorrectly predicts a constant and negative entropy production rate, while the SDE cannot

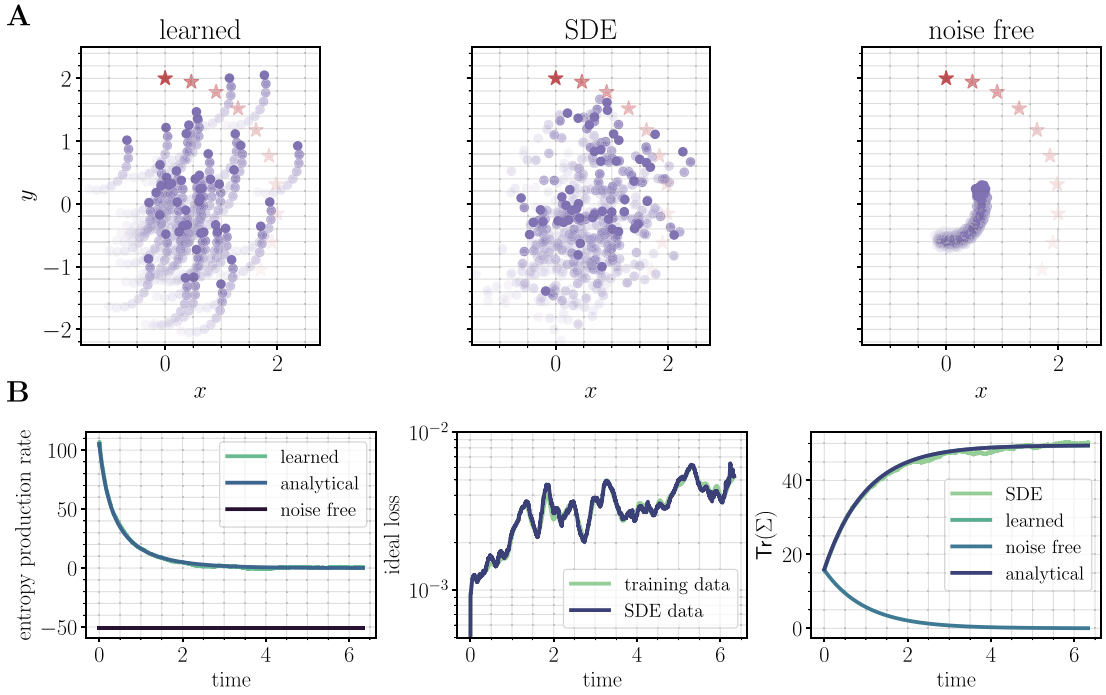

Figure 1. A system of $N = 50$ particles in a harmonic trap with a harmonic interaction: (A) A single sample trajectory. The mean of the trap β_t is shown with a red star, while past positions of the particles are indicated by a fading trajectory. The noise-free system (right) is too concentrated, and fails to capture the variance of the stochastic dynamics (center). The learned system (left) accurately captures the variance, and in addition generates physically interpretable trajectories for the particles. (B) Quantitative comparison to the analytical solution. The learned solution matches the entropy production rate, score, and covariance well. A movie of the particle motion can be found here.

make a prediction for the entropy production rate without an additional learning component; we study this possibility in the next example. In addition, the learned model accurately predicts the high-dimensional covariance Σ of the system (curves lie directly on top of the analytical result, trace shown for simplicity). The SDE also captures the covariance, but exhibits more fluctuations in the estimate; the noise-free system incorrectly estimates all covariance components as decaying to zero.

4.2. Soft spheres in an anharmonic trap

Setup. Here, we consider a system of $N = 5$ physical particles in an anharmonic trap in dimension $\bar{d} = 2$ that exhibit soft-sphere repulsion. This system gives rise to a 10-dimensional (1), which is significantly too high for standard PDE solvers. The stochastic dynamics is given by

$$\begin{aligned} \mathrm{d}x_t^{(i)} &= 4B\big(\beta_t - x_t^{(i)}\big)|x_t^{(i)} - \beta_t|^2\mathrm{d}t \\ &\quad + \frac{A}{Nr^2}\sum_{j=1}^{N}\big(x_t^{(i)} - x_t^{(j)}\big)\exp\left(-\frac{|x_t^{(i)} - x_t^{(j)}|^2}{2r^2}\right)\mathrm{d}t + \sqrt{2D}\,\mathrm{d}W_t, \quad i = 1,\dots,N, \end{aligned} \tag{25}$$

where β_t again represents a moving trap, $A > 0$ sets the strength of the repulsion between the spheres, r sets their size, $B > 0$ sets the strength of the trap, and each $x_0^{(i)} \sim \rho_0$. We set $\beta(t) = a(\cos\pi\omega t, \sin\pi\omega t)^\mathsf{T}$ or $\beta(t) = a(\cos\pi\omega t, 0)^\mathsf{T}$ with $a = 2, \omega = 1, D = 0.25, A = 10$, and $r = 0.5$. We fix $B = D/R^2$ with $R = \sqrt{\gamma N}r$ and $\gamma = 5.0$. This ensures that the trap scales with the number of particles and that they have sufficient room in the trap to generate a complex dynamics. The circular case converges to a distribution $\rho_t^* = \rho^* \circ Q_t$ that can be described as a fixed distribution ρ^* composed with a time-dependent rotation Q_t, and hence the entropy production rate converges to zero by change of variables. The linear case does not exhibit this kind of convergence, and the entropy production rate should oscillate around zero as the particles are repeatedly pushed and pulled by the trap. We make use of the same network architecture as in section 4.1.

Results. Similar to section 4.1, an example trajectory from the learned system, the SDE (25), and the noise-free system obtained by setting $D = 0$ are shown in figure 2(A) in the circular case. The learned particle

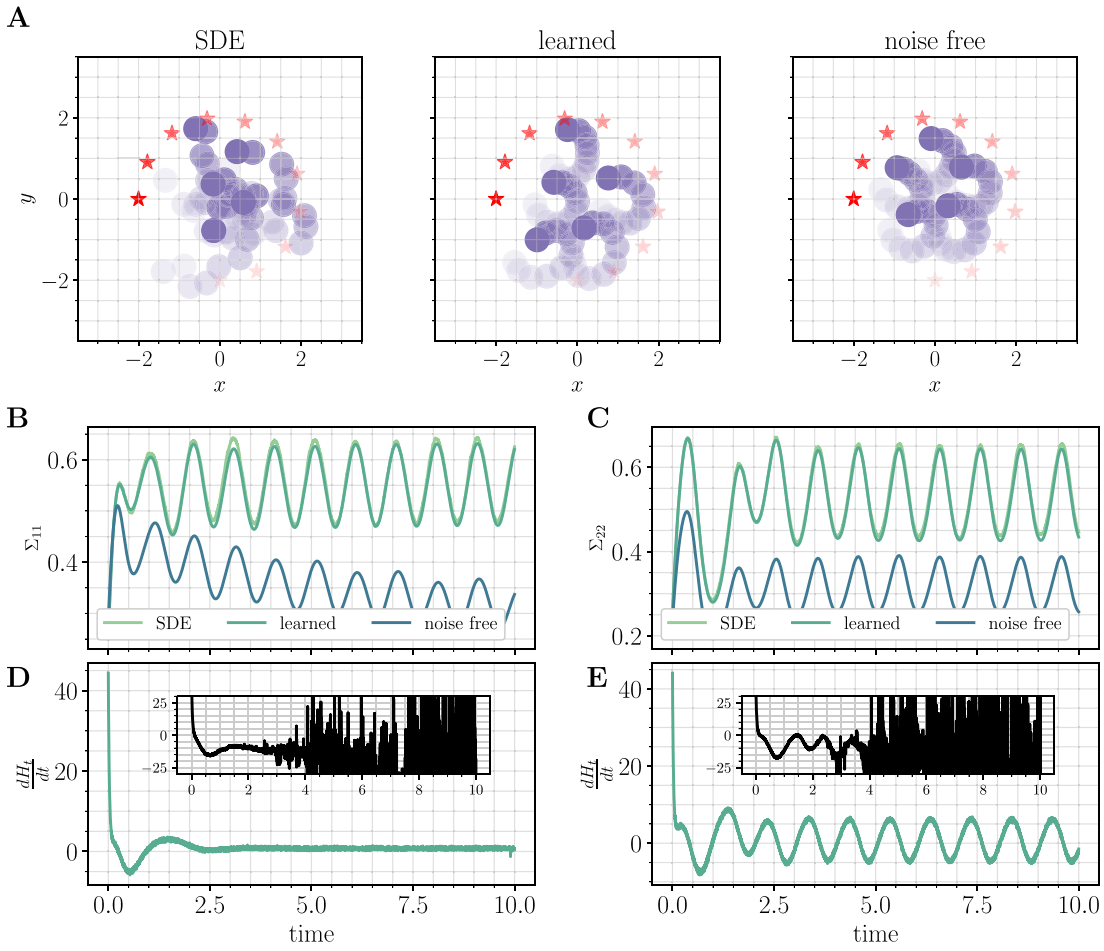

Figure 2. A system of $N = 5$ soft-spheres in an anharmonic trap: (A) Example particle trajectories in the case of a rotating trap. Trap position shown with a red star. Movies of the circular and linear motion can be viewed here and here, respectively. ((B)/(C)) A single component of the covariance of the samples, in the case of a rotating trap in (B) and a linearly oscillating trap in (C). The learned system agrees well with the SDE, while the noise-free system under-predicts the moments. ((D)/(E)) Prediction of the entropy production rate for a rotating trap in (D) and linearly oscillating trap in (E). Main figure depicts the prediction obtained from sequential SBTM, while the inset depicts the prediction obtained when learning on samples from the SDE. Sequential SBTM captures the temporal evolution of the entropy production rate, while learning on the SDE is initially offset and later divergent.

trajectories exhibit an intuitive circular motion when compared to the SDE trajectory. When compared to the noise-free system, the learned trajectories exhibit a greater amount of spread, which enables the deterministic dynamics to accurately capture the statistics of the stochastic dynamics. Numerical estimates of a single component of the covariance and of the entropy production rate are shown in figures 2(B)/(C), with all moments shown in appendix D.2. The learned and SDE systems accurately capture the covariance, while the noise-free system underestimates the covariance in both the linear and the circular case. The prediction of the entropy production rate via algorithm 1 is reasonable in both cases, exhibiting the expected convergence to and oscillation around zero in the circular and linear cases, respectively. In the inset, we show the prediction of the entropy production rate when learning on samples from the SDE; the prediction is initially offset, and later becomes divergent. We found that this behavior was generic when training on the SDE, but never observed it when training on self-consistent samples.

4.3. An active swimmer

Setup. We now consider a model from the physics of active matter, which describes the motion of a single motile swimmer in an anharmonic trap. The swimmer can be thought of as a run-and-tumble bacterium [59]; it travels in a fixed direction for a fluctuating duration before picking a new direction at random in which to swim. The system is two-dimensional, and is given by the SDE for the position x and velocity v

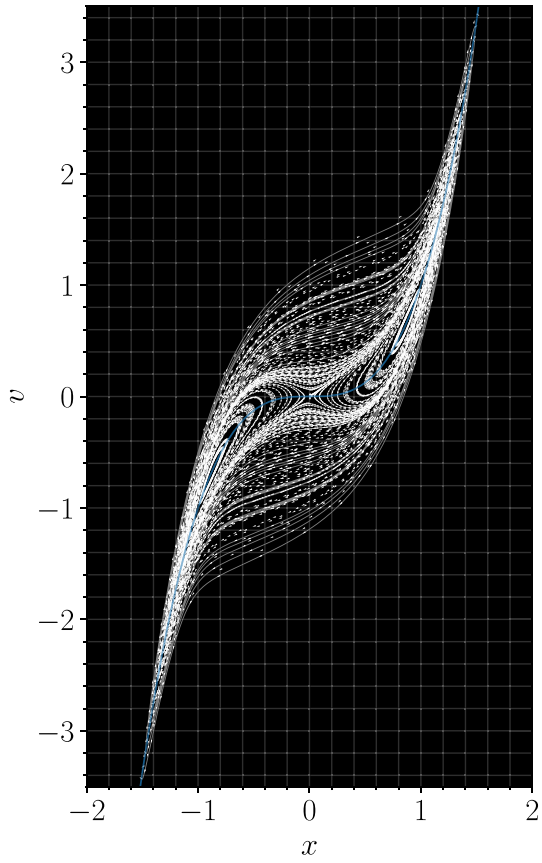

Figure 3. An active swimmer: probability flow phase portrait. Phase portrait of the probability flow, computed with parameters frozen at the fixed time $t = 10/\gamma$. Low-opacity curves depict closed limit cycles, while arrows indicate the direction of the probability flow. The phase portrait reveals non-equilibrium steady-state currents, both within and between the two modes. The nullcline $v = x^3$ passes through the two modes (shown in blue), with an unstable equilibrium at the origin.

$$\begin{aligned} \mathrm{d}x &= \left(-x^3 + v\right)\mathrm{d}t, \\ \mathrm{d}v &= -\gamma v \mathrm{d}t + \sqrt{2\gamma D}\mathrm{d}W_t. \end{aligned} \tag{26}$$

While low-dimensional, (26) exhibits convergence to a non-equilibrium statistical steady state in which the probability current $j_t(x) = v_t(x)\rho_t(x)$ is non-zero. Here, we show that sequential SBTM is capable of accurately capturing such currents, which is necessary to resolve the dynamics of the FPE: if our goal were solely to sample at equilibrium, it would be sufficient to freeze the samples after an initial transient. Moreover, we show that the method preserves the stationary distribution over long times relative to the persistence time $1/\gamma$ of the swimmer, and does not display appreciable accumulation of error.

We set $\gamma = 0.1$ and $D = 1.0$. Because noise only enters the system through the velocity variable v in (26), the score can be taken to be one-dimensional, which is equivalent to learning the score only in the range of the rank-deficient diffusion matrix. Further details on the architecture can be found in appendix D.3.

Results. A phase portrait for the learned probability flow dynamics is shown in figure 3, computed by rolling out an additional set of 50 trajectories for time $5/\gamma$ with a fixed set of parameters (after learning for time $10/\gamma$). The phase portrait depicts closed limit cycles between and centered within the modes reminiscent of the classical phase portrait for the pendulum. Here, the closed limit cycles correspond to non-equilibrium currents that preserve the steady-state density.

A kernel density estimate for the distribution of samples produced by the learned system, the stochastic system, and the noise-free systems are shown in figure 4, which demonstrate that the distribution of the learned samples qualitatively matches the distribution of the SDE samples. Comparatively, the noise-free system grows overly concentrated with time, ultimately converging to a singular dirac measure at the origin. A movie of the motion of the samples $(x_i(t), v_i(t))_{t\geqslant 0}$ over a duration $10/\gamma$ in phase space can be seen at this link. The movie highlights convergence of the learned solution to one with a non-zero steady-state probability current that qualitatively matches that of the SDE, but which enjoys more interpretable sample trajectories.

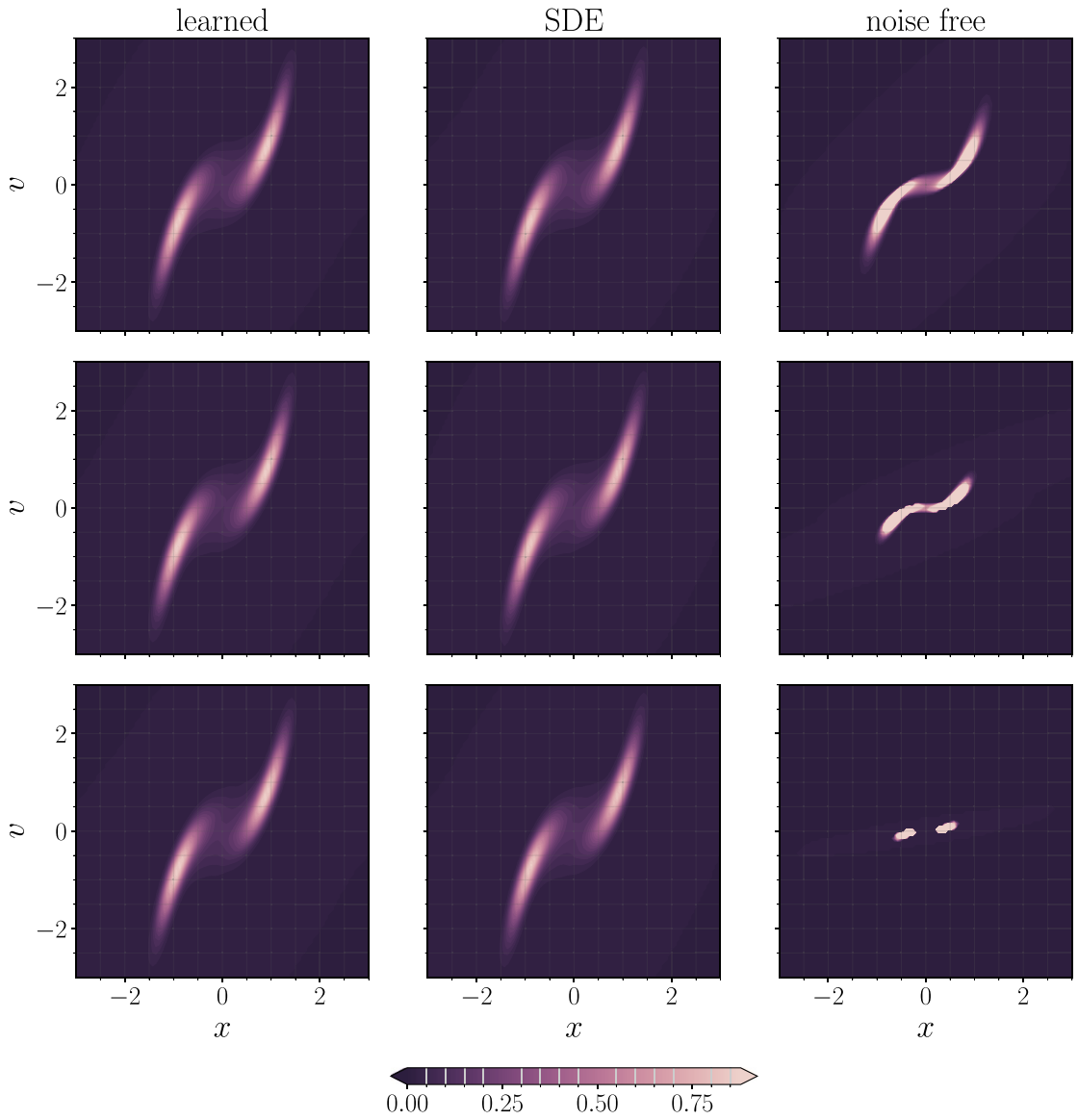

Figure 4. An active swimmer: kernel density estimates. PDFs computed via kernel density estimation in the xv plane. Columns denote solution type and rows denote snapshots in time ($t = 0.5/\gamma, 1.5/\gamma$, and $3.0/\gamma$, respectively). The KDE reveals bimodality in the probability density brought about by the activity of the particle. The noise free system becomes too concentrated around the nullcline $v = x^3$, and does not accurately capture the shape of the SDE and learned solutions, while the SDE and learned solutions are nearly identical.

5. Outlook and conclusions

Building on the toolbox of score-based diffusion recently developed for generative modeling, we introduced a related approach—SBTM – that gives an alternative to simulating the corresponding SDE to solve the FPE. While SBTM is more costly than integration of the SDE because it involves a learning component, it gives access to quantities that are not directly accessible from the samples given by integrating the SDE, such as pointwise evaluation of the PDF, the probability current, or the entropy. Our numerical examples indicate that SBTM is scalable to systems in high dimension where standard numerical techniques for partial differential equations are inapplicable. The method can be viewed as a deterministic Lagrangian integration method for the FPE, and our results show that its trajectories are more easily interpretable than the corresponding trajectories of the SDE.

Data availability statement

All data that support the findings of this study are included within the article (and any supplementary files).

Acknowledgments

We thank Michael Albergo, Joan Bruna, Jonathan Niles-Weed, Stephen Tu, and Jean-Jacques Slotine for helpful discussions and feedback on the manuscript. N M B is partially supported by the Research Training Group in Modeling and Simulation funded by the National Science Foundation via Grant RTG/DMS—1646339. E V E is supported by the National Science Foundation under Awards DMR-1420073, DMS-2012510, and DMS-2134216, by the Simons Collaboration on Wave Turbulence, Grant No. 617006, and by a Vannevar Bush Faculty Fellowship.

Appendix A. Some basic formulas

Here, we derive some results linking the solution of the transport equation (10) with that of the probability flow equation (6).

A.1. Probability density and probability current

We begin with a lemma.

Lemma 5. *Let $\rho_t : \Omega \to \mathbb{R}_{\geqslant 0}$ satisfy the transport equation*

$$\partial_t \rho_t(x) = -\nabla \cdot (v_t(x)\rho_t(x)). \tag{A.1}$$

Assume that $v_t(x)$ is C^2 in both t and x for $t \geqslant 0$ and globally Lipschitz in x. Then, given any $t, t' \geqslant 0$, the solution of (A.1) satisfies

$$\rho_t(x) = \rho_{t'}(X_{t,t'}(x)) \exp\left(-\int_{t'}^{t} \nabla \cdot v_\tau(X_{t,\tau}(x))\mathrm{d}\tau\right) \tag{A.2}$$

where $X_{\tau,t}$ is the probability flow solution to (6). In addition, given any test function $\phi : \Omega \to \mathbb{R}$, we have

$$\int_\Omega \phi(x)\rho_t(x)\mathrm{d}x = \int_\Omega \phi(X_{t',t}(x))\rho_{t'}(x)\mathrm{d}x. \tag{A.3}$$

In words, lemma 5 states that an evaluation of the PDF ρ_t at a given point x may be obtained by evolving the probability flow equation (6) backwards to some earlier time t' to find the point x' that evolves to x at time t, assuming that $\rho_{t'}(x')$ is available. In particular, for $t' = 0$, we obtain

$$\rho_t(x) = \rho_0(X_{t,0}(x)) \exp\left(-\int_0^t \nabla \cdot v_\tau(X_{t,\tau}(x))\mathrm{d}\tau\right), \tag{A.4}$$

and

$$\int_\Omega \phi(x)\rho_t(x)\mathrm{d}x = \int_\Omega \phi(X_{0,t}(x))\rho_0(x)\mathrm{d}x. \tag{A.5}$$

Since the probability current is by definition $v_t(x)\rho_t(x)$, using (A.4) to express $\rho_t(x)$ also gives the following equation for the current:

$$v_t(x)\rho_t(x) = v_t(x)\rho_0(X_{t,0}(x)) \exp\left(-\int_0^t \nabla \cdot v_\tau(X_{\tau,t}(x))\mathrm{d}\tau\right). \tag{A.6}$$

Proof. The assumed C^2 and globally Lipschitz conditions on v_t guarantee global existence (on $t \geqslant 0$) and uniqueness of the solution to (6). Differentiating $\rho_t(X_{t',t}(x))$ with respect to t and using (6) and (A.1) we deduce

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\rho_t(X_{t',t}(x)) &= \partial_t \rho_t(X_{t',t}(x)) + \frac{\mathrm{d}}{\mathrm{d}t}X_{t',t}(x) \cdot \nabla \rho_t(X_{t',t}(x)) \\ &= \partial_t \rho_t(X_{t',t}(x)) + v_t(X_{t',t}(x)) \cdot \nabla \rho_t(X_{t',t}(x)) \\ &= -\nabla \cdot v_t(X_{t',t}(x))\, \rho_t(X_{t',t}(x)).\end{aligned} \tag{A.7}$$

Integrating this equation in t from $t = t'$ to $t = t$ gives

$$\rho_t(X_{t',t}(x)) = \rho_{t'}(x) \exp\left(-\int_{t'}^t \nabla \cdot v_\tau(X_{t',\tau}(x))\mathrm{d}\tau\right). \tag{A.8}$$

Evaluating this expression at $x = X_{t,t'}(x)$ and using the group properties (i) $X_{t',t}(X_{t,t'}(x)) = x$ and (ii) $X_{t',\tau}(X_{t,t'}(x)) = X_{t,\tau}(x)$ gives (A.2). Equation (A.3) can be derived by using (A.2) to express $\rho_t(x)$ in the integral at the left hand-side, changing integration variable $x \to X_{t',t}(x)$ and noting that the factor $\exp(-\int_{t'}^{t} \nabla \cdot v_\tau(X_{t',\tau}(x)))$ is precisely the Jacobian of this change of variable. To see this, note that by definition the flow map satisfies

$$\frac{\mathrm{d}}{\mathrm{d}t} X_{t',t}(x) = v_t(X_{t',t}(x)), \qquad X_{\tau,\tau}(x) = x \ \forall \tau \geqslant 0.$$

Hence,

$$\frac{\mathrm{d}}{\mathrm{d}t} \nabla X_{t',t}(x) = \nabla v_t(X_{t',t}(x)) \nabla X_{t',t}(x), \qquad \nabla X_{\tau,\tau}(x) = I \ \forall \tau \geqslant 0.$$

By Jacobi's formula for determinants,

$$\frac{\mathrm{d}}{\mathrm{d}t} \det(\nabla X_{t',t}(x)) = \mathrm{Tr}(\nabla v_t(X_{t',t}(x))) \det(\nabla X_{t',t}(x)), \qquad \det(\nabla X_{\tau,\tau}(x)) = 1 \ \forall \tau \geqslant 0.$$

Integrating this equation with respect to t and using that $\mathrm{Tr}(\nabla v_t(\cdot)) = \nabla \cdot v_t(\cdot)$, we obtain

$$\det(\nabla X_{t',t}(x)) = \exp\left(\int_{t'}^{t} \nabla \cdot v_\tau(X_{t',\tau}(x)) \mathrm{d}\tau \right).$$

The result is the integral at the right hand-side of (A.3). □

Lemma 5 also holds locally in time for any $v_t(x)$ that is C^2 in both t and x. In particular, it holds locally if we set $s_t(x) = \nabla \log \rho_t(x)$ and if we assume that $\rho_0(x)$ is (i) positive everywhere on Ω and (ii) C^3 in x. In this case, (A.1) is the FPEs (1) and (A.2) holds for the solution to that equation.

A.2. Calculation of the differential entropy

We now consider computation of the differential entropy, and state a similar result.

Lemma 6. *Assume that $\rho_0 : \Omega \to \mathbb{R}_{\geqslant 0}$ is positive everywhere on Ω and C^3 in its argument. Let $\rho_t : \Omega \to \mathbb{R}_{\geqslant 0}$ denote the solution to the FPE (1) (or equivalently, to the transport equation (A.1) with $s_t(x) = \nabla \log \rho_t(x)$ in the definition of $v_t(x)$). Then the differential entropy $H_t = -\int_\Omega \log \rho_t(x)\, \rho_t(x) \mathrm{d}x$ can expressed as*

$$H_t = -\int_\Omega \log \rho_t(X_{0,t}(x))\, \rho_0(x) \mathrm{d}x = H_0 + \int_0^t \int_\Omega \nabla \cdot v_\tau(X_{0,\tau}(x)) \rho_0(x) \mathrm{d}x \mathrm{d}\tau \tag{A.9}$$

or

$$H_t = H_0 - \int_0^t \int_\Omega s_\tau(X_{0,\tau}(x)) \cdot v_\tau(X_{0,\tau}(x)) \rho_0(x) \mathrm{d}x \mathrm{d}\tau. \tag{A.10}$$

Proof. We first derive (A.9). Observe that applying (A.5) with $\phi = \log \rho_t$ leads to the first equality. The second can then be deduced from (A.4). To derive (A.10), notice that from (A.1),

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}t} H_t &= \int_\Omega \log \rho_t(x) \nabla \cdot (v_t(x) \rho_t(x)) \, \mathrm{d}x, \\ &= -\int_\Omega \nabla \log \rho_t(x) \cdot v_t(x) \rho_t(x) \mathrm{d}x, \\ &= -\int_\Omega s_t(x) \cdot v_t(x) \rho_t(x) \mathrm{d}x. \end{aligned} \tag{A.11}$$

Above, we used integration by parts to obtain the second equality and $s_t = \nabla \log \rho_t$ to get the third. Now, using (A.5) with $\phi = s_t \cdot v_t$ integrating the result gives (A.10). □

A.3. Resampling of ρ_t at any time t

If the score $s_t \approx \nabla \log \rho_t$ is known to sufficient accuracy, ρ_t can be resampled at any time t using the dynamics

$$\mathrm{d}X_\tau = s_t(X_\tau) \mathrm{d}\tau + dW_\tau. \tag{A.12}$$

In (A.12), τ is an artificial time used for sampling that is distinct from the physical time in (2). For $s_t = \nabla \log \rho_t$, the equilibrium distribution of (A.12) is exactly ρ_t. In practice, s_t will be imperfect and will have an error that increases away from the samples used to learn it; as a result, (A.12) should be used near samples for a fixed amount of time to avoid the introduction of additional errors.

Appendix B. Further details on score-based transport modeling

B.1. Bounding the KL divergence

Let us restate proposition 1 for convenience:

Proposition 1 (Control of the KL divergence). *Assume that the conditions listed in section 1.2 hold. Let ρ_t denote the solution to the transport equation (10), and let ρ_t^* denote the solution to the FPE (1). Assume that $\rho_{t=0}(x) = \rho^*_{t=0}(x) = \rho_0(x)$ for all $x \in \Omega$. Then*

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathsf{KL}(\rho_t \,\|\, \rho_t^*) \leqslant \frac{1}{2}\int_\Omega |s_t(x) - \nabla \log \rho_t(x)|^2_{D_t(x)}\, \rho_t(x)\mathrm{d}x, \tag{11}$$

where $|\cdot|^2_{D_t(x)} = \langle \cdot, D_t(x)\cdot\rangle$.

Proof. By assumption, ρ_t solves (10) and ρ_t^* solves (1). Denote by $v_t(x) = b_t(x) - D_t(x)s_t(x)$ and $v_t^*(x) = b_t(x) - D_t(x)s_t^*(x)$ with $s_t^*(x) = \nabla \log \rho_t^*(x)$. Then, we have

$$\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}t}\mathsf{KL}(\rho_t \,\|\, \rho_t^*) &= \frac{\mathrm{d}}{\mathrm{d}t}\int_\Omega \log\left(\frac{\rho_t(x)}{\rho_t^*(x)}\right)\rho_t(x)\mathrm{d}x, \\
&= -\int_\Omega \frac{\rho_t(x)}{\rho_t^*(x)}\partial_t \rho_t^*(x)\mathrm{d}x + \int_\Omega \log\left(\frac{\rho_t(x)}{\rho_t^*(x)}\right)\partial_t\rho_t(x)\mathrm{d}x, \\
&= -\int_\Omega v_t^*(x)\cdot\nabla\left(\frac{\rho_t(x)}{\rho_t^*(x)}\right)\rho_t^*(x)\mathrm{d}x + \int_\Omega v_t(x)\cdot\nabla\log\left(\frac{\rho_t(x)}{\rho_t^*(x)}\right)\rho_t(x)\mathrm{d}x, \\
&= -\int_\Omega (v_t^*(x) - v_t(x))\cdot(\nabla\log\rho_t(x) - \nabla\log\rho_t^*(x))\,\rho_t(x)\mathrm{d}x, \\
&= \int_\Omega (s_t^*(x) - s_t(x))\cdot D_t(x)\,(\nabla\log\rho_t(x) - s_t^*(x))\,\rho_t(x)\mathrm{d}x.
\end{aligned}$$

Above, we used integration by parts to obtain the third equality. Now, dropping function arguments for simplicity of notation, we have that

$$\begin{aligned}
|\nabla\log\rho_t - s_t|^2_{D_t} &= |\nabla\log\rho_t - s_t^* + s_t^* - s_t|^2_{D_t}, \\
&= |\nabla\log\rho_t - s_t^*|^2_{D_t} + |s_t^* - s_t|^2_{D_t} + 2(\nabla\log\rho_t - s_t^*)\cdot D_t(s_t^* - s_t), \\
&\geqslant 2(\nabla\log\rho_t - s_t^*)\cdot D_t(s_t^* - s_t).
\end{aligned}$$

Hence, we deduce that

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathsf{KL}(\rho_t \,\|\, \rho_t^*) \leqslant \frac{1}{2}\int_\Omega |s_t(x) - \nabla\log\rho_t(x)|^2_{D_t(x)}\rho_t(x)\mathrm{d}x. \tag{B.1}$$

□

B.2. SBTM in the Eulerian frame

The Eulerian equivalent of proposition 2 can be stated as:

Proposition 7 (SBTM in the Eulerian frame). *Assume that the conditions listed in section 1.2 hold. Fix $T \in (0, \infty]$ and consider the optimization problem*

$$\begin{aligned}
&\min_{\{s_t : t\in[0,T)\}} \int_0^T \int_\Omega |s_t(x) - \nabla\log\rho_t(x)|^2_{D_t(x)}\,\rho_t(x)\mathrm{d}x\mathrm{d}t \\
&\text{subject to:} \quad \partial_t\rho_t(x) = -\nabla\cdot(v_t(x)\rho_t(x)), \quad x\in\Omega
\end{aligned} \tag{B.2}$$

with $v_t(x) = b_t(x) - D_t(x)s_t(x)$. Then every minimizer of (B.2) satisfies $D_t(x)s_t^*(x) = D_t(x)\nabla\log\rho_t^*(x)$ where $\rho_t^* : \Omega \to \mathbb{R}_{>0}$ solves (1).

In words, this proposition states that solving the constrained optimization problem (B.2) is equivalent to solving the FPE (1).

Proof. The constrained minimization problem (B.2) can be handled by considering the extended objective

$$\int_0^T\int_\Omega \left(|s_t(x) - \nabla\log\rho_t(x)|^2_{D_t(x)}\,\rho_t(x) + \mu_t(x)\,(\partial_t\rho_t(x) + \nabla\cdot(v_t(x)\rho_t(x)))\right)\mathrm{d}x\mathrm{d}t \tag{B.3}$$

where $v_t(x) = b_t(x) - D_t(x)s_t(x)$ and $\mu_t : \mathbb{R}^d \to \mathbb{R}_{\geqslant 0}$ is a Lagrange multiplier. The Euler–Lagrange equations associated with (B.3) read

$$\begin{aligned}
\partial_t \rho_t(x) &= -\nabla \cdot (v_t(x)\rho_t(x)) \\
\partial_t \mu_t(x) &= -v_t(x) \cdot \nabla \mu_t(x) + |s_t(x)|^2_{D_t(x)} - |\nabla \log \rho_t|^2_{D_t(x)} \\
&\quad + 2\nabla \cdot \left[D_t(x) \left(s_t(x) - \nabla \log \rho_t(x) \right) \right], \\
0 &= \mu_T(x), \\
0 &= 2D_t(x) \left(s_t(x) - \nabla \log \rho_t(x) \right) \rho_t(x) - D_t(x) \nabla \mu_t(x) \rho_t(x).
\end{aligned} \tag{B.4}$$

Clearly, these equations will be satisfied if $s_t^*(x) = \nabla \log \rho_t^*(x)$ for all $x \in \Omega$, $\mu_t^*(x) = 0$ for all x, and ρ_t^* solves (1). This solution is also a global minimizer, because it zeroes the value of the objective. Moreover, all global minimizers must satisfy $D_t(x)s_t^*(x) = D_t(x)\nabla \log \rho_t^*(x)$ (ρ_t-almost everywhere), as this is the only way to zero the objective.

It is also easy to see that there are no other local minimizers. To check this, we can use the fourth equation to write

$$D_t(x)(s_t(x) - \nabla \log \rho_t(x)) = \tfrac{1}{2} D_t(x) \nabla \mu_t(x).$$

Then,

$$|s_t(x)|^2_{D_t(x)} - |\nabla \log \rho_t(x)|^2_{D_t(x)} = \tfrac{1}{2} \left(s_t(x) + \nabla \log \rho_t(x) \right)^{\mathsf{T}} D_t(x) \nabla \mu_t(x).$$

This reduces the first three equations to

$$\begin{aligned}
\partial_t \rho_t(x) &= -\nabla \cdot \left(b_t(x)\rho_t(x) - D_t(x)\nabla \rho_t(x) - \tfrac{1}{2}\rho_t D_t(x) \nabla \mu_t(x) \right) \\
\partial_t \mu_t &= \left(b_t(x) - D_t(x)\nabla \log \rho_t(x) - \tfrac{1}{2} D_t(x) \nabla \mu_t(x) \right)^{\mathsf{T}} \nabla \mu_t(x) \\
&\quad + \nabla \cdot (D_t(x) \nabla \mu_t(x)) + \tfrac{1}{2} \left(s_t(x) + \nabla \log \rho_t(x) \right)^{\mathsf{T}} D_t(x) \nabla \mu_t(x). \\
\mu_T(x) &= 0.
\end{aligned} \tag{B.5}$$

Since the equation for μ_t is homogeneous in μ_t and $\mu_T = 0$, we must have $\mu_t = 0$ for all $t \in [0, T)$, and the equation for ρ_t reduces to (1). □

B.3. SBTM in the Lagrangian frame

As stated, proposition 7 is not practical, because it is phrased in terms of the density ρ_t. The following result demonstrates that the transport map identity (7) can be used to re-express proposition 7 entirely in terms of known quantities.

Proposition 2 (Score-based transport modeling). *Assume that the conditions listed in section 1.2 hold. Define $v_t(x) = b_t(x) - D_t(x)s_t(x)$ and consider*

$$\begin{aligned}
\dot{X}_t(x) &= v_t(X_t(x)), & X_0(x) &= x, \\
\dot{G}_t(x) &= -[\nabla v_t(X_t(x))]^{\mathsf{T}} G_t(x) - \nabla \nabla \cdot v_t(X_t(x)), & G_0(x) &= \nabla \log \rho_0(x).
\end{aligned} \tag{13}$$

Then $\rho_t = X_t \sharp \rho_0$ solves (10), the equality $G_t(x) = \nabla \log \rho_t(X_t(x))$ holds, and for any $T \in [0, \infty)$

$$\mathsf{KL}(X_T \sharp \rho_0 \,\|\, \rho_T^*) \leqslant \frac{1}{2} \int_0^T \int_\Omega |s_t(X_t(x)) - G_t(x)|^2_{D_t(X_t(x))} \, \rho_0(x) \mathrm{d}x \mathrm{d}t. \tag{14}$$

Moreover, if s_t^* is a minimizer of the constrained optimization problem

$$\min_s \int_0^T \int_\Omega |s_t(X_t(x)) - G_t(x)|^2_{D_t(X_t(x))} \, \rho_0(x) \mathrm{d}x \mathrm{d}t \quad \textit{subject to (13)} \tag{15}$$

then $D_t(x)s_t^*(x) = D_t(x)\nabla \log \rho_t^*(x)$ where ρ_t^* solves the FPE (1). The map X_t^* associated to any minimizer is a transport map from ρ_0 to ρ_t^*, i.e.

$$x \sim \rho_0 \qquad \textit{implies that} \qquad X_t^*(x) \sim \rho_t^*, \qquad \forall t \in [0, T]. \tag{16}$$

Proof. Let us first show that $G_t(x) = \nabla \log \rho_t(X_t(x))$ satisfies (13) if $\rho_t = X_t \sharp \rho_0$, i.e. if ρ_t satisfies the transport equation (10). Since (10) implies that

$$\partial_t \log \rho_t(x) + v_t(x) \cdot \nabla \log \rho_t(x) = -\nabla \cdot v_t(x), \tag{B.6}$$

taking the gradient gives

$$\partial_t \nabla \log \rho_t(x) + [\nabla v_t(x)]^\mathsf{T} \nabla \log \rho_t(x) + \nabla \nabla \log \rho_t(x) \cdot v_t(x) = -\nabla \nabla \cdot v_t(x). \tag{B.7}$$

Therefore $G_t(x) = \nabla \log \rho_t(X_t(x))$ solves

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}t} G_t(x) &= \partial_t \nabla \log \rho_t(X_t(x)) + \nabla \nabla \log \rho_t(X_t(x)) \cdot \frac{\mathrm{d}}{\mathrm{d}t} X_t(x), \\ &= \partial_t \nabla \log \rho_t(X_t(x)) + \nabla \nabla \log \rho_t(X_t(x)) \cdot v_t(x), \\ &= -\nabla \nabla \cdot v_t(X_t(x)) - [\nabla v_t(X_t(x))]^\mathsf{T} \nabla \log \rho_t(X_t(x)), \end{aligned} \tag{B.8}$$

which recovers the equation for $G_t(x)$ in (13). Hence, the objective in (15) can also be written as

$$\begin{aligned} &\int_0^T \int_\Omega |s_t(X_t(x)) - \nabla \log \rho_t(X_t(x))|^2_{D_t(X_t(x))} \rho_0(x) \mathrm{d}x \mathrm{d}t \\ &\quad = \int_0^T \int_\Omega |s_t(x) - \nabla \log \rho_t(x)|^2_{D_t(x)} \rho_t(x) \mathrm{d}x \mathrm{d}t \end{aligned} \tag{B.9}$$

where the second equality follows from (A.5) if $\rho_t(x)$ satisfies (A.1). Hence, (15) is equivalent to (B.2). The bound on $\mathsf{KL}(X_T \sharp \rho_0 \,\|\, \rho_T^*)$ follows from (12). □

Adjoint equations. In terms of a practical implementation, the objective in (B.2) can be evaluated by generating samples $\{x_i\}_{i=1}^n$ from ρ_0 and solving the equations for X_t and G_t using the initial conditions $X_0(x_i) = x_i$ and $G_0(x_i) = \nabla \log \rho_0(x_i)$. Note that evaluating this second initial condition only requires one to know ρ_0 up to a normalization factor. To evaluate the gradient of the objective, we can introduce equations adjoint to those for X_t and G_t. To do so, we can consider the extended objective

$$\begin{aligned} L[s_t] = &\int_0^T \int_\Omega |s_t(X_t(x)) - G_t(x)|^2_{D_t(X_t(x))} \rho_0(x) \mathrm{d}x \mathrm{d}t \\ &+ \int_0^T \int_\Omega \theta_t(x) \left(\dot{X}_t(x) - v_t(X_t(x)) \right) \rho_0(x) \mathrm{d}x \mathrm{d}t \\ &+ \int_0^T \int_\Omega \eta_t(x) \left(\dot{G}_t(x) + \nabla v_t(X_t(x))^\mathsf{T} G_t(X_t(x)) + \nabla \nabla \cdot v_t(X_t(x)) \right) \rho_0(x) \mathrm{d}x \mathrm{d}t. \end{aligned} \tag{B.10}$$

Taking the first variation with respect to $G_t(x)$ and $X_t(x)$, respectively, gives the equations

$$\begin{aligned} \partial_t \eta_t(x) &= \nabla v_t(X_t(x)) \eta_t(x) + 2 D_t(X_t(x)) \left(G_t(x) - s_t(X_t(x)) \right) \\ \eta_T(x) &= 0 \\ \partial_t \theta_t(x) + \nabla v_t(X_t(x))^\mathsf{T} \theta_t(x) &= 2 \nabla s_t(X_t(x))^\mathsf{T} D_t(X_t(x)) \left(s_t(X_t(x)) - G_t(x) \right) \\ &\quad + \left(s_t(X_t(x)) - G_t(x) \right) \cdot \nabla D_t(X_t(x)) \left(s_t(X_t(x)) - G_t(x) \right) \\ &\quad + \eta_t(x) \cdot \left(\nabla \nabla v_t(X_t(x))^\mathsf{T} G_t(x) \right) + \eta_t \cdot \nabla \nabla \nabla \cdot v_t(X_t(x)) \\ \theta_T(x) &= 0. \end{aligned} \tag{B.11}$$

B.4. Sequential SBTM

Let us restate proposition 3 for convenience:

Proposition 3 (Sequential SBTM). *In the same setting as proposition 2, let $X_t(x)$ solve the first equation in (13) with $v_t(x) = b_t(x) - D_t(x) s_t(x)$. Let s_t be obtained via*

$$\min_{s_t} \int_\Omega \left(|s_t(X_t(x))|^2_{D_t(X_t(x))} + 2 \nabla \cdot (D_t(X_t(x)) s_t(X_t(x))) \right) \rho_0(x) \mathrm{d}x. \tag{17}$$

Then, each minimizer s_t^* of (17) satisfies $D_t(x) s_t^*(x) = D_t(x) \nabla \log \rho_t^*(x)$ where ρ_t^* is the solution to (1). Moreover, the map X_t^* associated to s_t^* is a transport map from ρ_0 to ρ_t^*.

Proof. If $X_t \sharp \rho_0 = \rho_t$, then by definition we have the identity

$$\int_\Omega \left(|s_t(X_t(x))|^2_{D_t(X_t(x))} + 2\nabla \cdot (D_t(X_t(x))s_t(X_t(x))) \right) \rho_0(x)\mathrm{d}x = \int_\Omega \left(|s_t(x)|^2_{D_t(x)} + 2\nabla \cdot (D_t(x)s_t(x)) \right) \rho_t(x)\mathrm{d}x. \tag{B.12}$$

This means that the optimization problem in (17) is equivalent to

$$\min_{s_t} \int_\Omega \left(|s_t(x)|^2_{D_t(x)} + 2\nabla \cdot (D_t(x)s_t(x)) \right) \rho_t(x)\mathrm{d}x.$$

All minimizers s_t^* of this optimization problem satisfy $D_t(x)s_t^*(x) = D_t(x)\nabla \log \rho_t(x)$. Hence, by (10),

$$\partial_t \rho_t(x) = -\nabla \cdot (b_t(x)\rho_t(x) - D_t(x)\nabla \rho_t(x)) \tag{B.13}$$

which recovers (1), so that $\rho_t(x) = \rho_t^*(x)$ solves (1). □

B.5. Learning from the SDE

In this section, we show that learning from the SDE alone—i.e. avoiding the use of self-consistent samples—does not provide a guarantee on the accuracy of ρ_t. We have already seen in (12) that it is sufficient to control $\int_0^T \int_\Omega |s_t(x) - \nabla \log \rho_t(x)|^2_{D_t(x)} \rho_t(x)\mathrm{d}x\mathrm{d}t$ to control $\mathsf{KL}(\rho_T \,\|\, \rho_T^*)$. The proof of proposition 1 shows that control on

$$\int_0^T \int_\Omega |s_t(x) - \nabla \log \rho_t^*(x)|^2_{D_t(x)} \rho_t^*(x)\mathrm{d}x\mathrm{d}t, \tag{B.14}$$

as would be provided by training on samples from the SDE, does not ensure control on $\mathsf{KL}(\rho_T \,\|\, \rho_T^*)$. The following proposition shows that control on (B.14) does not guarantee control on $\mathsf{KL}(\rho_T^* \,\|\, \rho_T)$ either. An analogous result appeared in [34] in the context of SBDM for generative modeling; here, we provide a self-contained treatment to motivate the use of the sequential SBTM procedure discussed in the main text.

Proposition 4 (Learning on external data). *Let $\rho_t : \Omega \to \mathbb{R}_{>0}$ solve (10), and let $\rho_t^* : \Omega \to \mathbb{R}_{>0}$ solve (1). Then, the following equality holds*

$$\mathsf{KL}(\rho_T^* \,\|\, \rho_T) = \int_0^T \int_\Omega |s_t(x) - \nabla \log \rho_t^*(x)|^2_{D_t(x)} \rho_t^*(x)\mathrm{d}x\mathrm{d}t + \int_0^T \int_\Omega (\nabla \log \rho_t(x) - s_t(x))^\mathsf{T} D_t(x) \left(s_t(x) - \nabla \log \rho_t^*(x)\right) \rho_t^*(x)\mathrm{d}x\mathrm{d}t. \tag{19}$$

Proof. By an analogous argument as in the proof of proposition 1, we find

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathsf{KL}(\rho_t^* \,\|\, \rho_t) = \int (\nabla \log \rho_t(x) - \nabla \log \rho_t^*(x))^\mathsf{T} D_t(x) \left(s_t(x) - \nabla \log \rho_t^*(x)\right) \rho_t^*(x)\mathrm{d}x.$$

Adding and subtracting $s_t(x)$ to the first term in the inner product and expanding gives

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathsf{KL}(\rho_t^* \,\|\, \rho_t) = \int_\Omega |s_t(x) - \nabla \log \rho_t^*(x)|^2_{D_t(x)} \rho_t^*(x) \mathrm{d}x + \int_\Omega (\nabla \log \rho_t(x) - s_t(x))^\mathsf{T} D_t(x) \left(s_t(x) - \nabla \log \rho_t^*(x)\right) \rho_t^*(x) \mathrm{d}x. \tag{B.15}$$

Integrating from 0 to T completes the proof. □

B.6. Denoising loss

The following standard trick can be used to avoid computing the divergence of $s_t(x)$:

Lemma 8. *Given $\xi = N(0, I)$, we have*

$$\begin{aligned} &\lim_{\alpha \downarrow 0} \alpha^{-1} \mathbb{E}\left(s_t(x + \alpha\xi) \cdot \xi\right) = \nabla \cdot s_t(x), \\ &\lim_{\alpha \downarrow 0} \alpha^{-1} \mathbb{E}\left(s_t(x + \alpha\sigma_t(x)\xi) \cdot \sigma_t(x)\xi\right) = \mathrm{tr}\left(D_t(x) \nabla s_t(x)\right). \end{aligned} \tag{B.16}$$

Proof. We have

$$\alpha^{-1} s_t(x + \alpha\xi) \cdot \xi = \alpha^{-1} s_t(x) \cdot \xi + (\nabla s_t(x)\xi) \cdot \xi + o(\alpha). \tag{B.17}$$

The expectation of the first term on the right-hand side of this equation is zero; the expectation of the second gives the result in (B.16). Hence, taking the expectation of (B.17) and evaluating the result in the limit as $\alpha \downarrow 0$ gives the first equation in (B.16). The second equation in (B.16) can be proven similarly using $\sigma_t(x)\sigma_t(x)^\mathsf{T} = D_t(x)$. □

Replacing $\nabla \cdot s_t(x)$ in (17) with the first expression in (B.17) for a fixed $\alpha > 0$ gives the loss

$$\mathcal{L}[s_t] = \mathbb{E}_\xi \left[\int_\Omega \left(|s_t(X_t(x))|^2 + \frac{2}{\alpha} s_t(X_t(x) + \alpha\xi) \cdot \xi \right) \rho_0(x) \mathrm{d}x \right]. \tag{B.18}$$

Evaluating the square term at a perturbed data point recovers the denoising loss of Vincent [61]

$$\mathcal{L}[s_t] = \mathbb{E}_\xi \left[\int_\Omega \left| s_t(X_t(x) + \alpha\xi) + \frac{\xi}{\alpha} \right|^2 \rho_0(x) \mathrm{d}x \right]. \tag{B.19}$$

We can improve the accuracy of the approximation with a 'doubling trick' that applies two draws of the noise of opposite sign to reduce the variance. This amounts to replacing the expectations in (B.16) with

$$\begin{aligned} &\tfrac{1}{2}\alpha^{-1} \mathbb{E}\left[s_t(x + \alpha\xi) \cdot \xi - s_t(x - \alpha\xi) \cdot \xi\right], \\ &\tfrac{1}{2}\alpha^{-1} \mathbb{E}\left[s_t(x + \alpha\sigma_t(x)\xi) \cdot \sigma_t(x)\xi - s_t(x - \alpha\sigma_t(x)\xi) \cdot \sigma_t(x)\xi\right], \end{aligned} \tag{B.20}$$

whose limits as $\alpha \to 0$ are $\nabla \cdot s_t(x)$ and $\mathrm{tr}\left(D_t(x)\nabla s_t(x)\right)$, respectively. In practice, we observe that this approach always helps stabilize training. Moreover, we observe that use of the denoising loss also stabilizes training, so that it is preferable to full computation of $\nabla \cdot s_t(x)$ even when the latter is feasible.

Appendix C. Gaussian case

Here, we consider the case of an OU process where the score can be written analytically, thereby providing a benchmark for our approach. The example treated in section 4.1 with details in appendix D.1 is a special case of such an OU process with additional symmetry arising from permutations of the particles. The SDE reads

$$\mathrm{d}X_t = -\Gamma_t(X_t - b_t)\mathrm{d}t + \sqrt{2}\sigma_t \mathrm{d}W_t \tag{C.1}$$

where $X_t \in \mathbb{R}^d$, $\Gamma_t \in \mathbb{R}^{d\times d}$ is a time-dependent positive-definite tensor (not necessarily symmetric), $b_t \in \mathbb{R}^d$ is a time-dependent vector, and $\sigma_t \in \mathbb{R}^{d\times d}$ is a time-dependent tensor. The FPE associated with (C.1) is

$$\partial_t \rho_t^*(x) = -\nabla \cdot \left((\Gamma_t x - b_t)\rho_t^*(x) - D_t \nabla \rho_t^*(x)\right) \tag{C.2}$$

where $D_t = \sigma_t\sigma_t^\intercal$. Assuming that the initial condition is Gaussian, $\rho_0 = \mathsf{N}(m_0, C_0)$ with $C_0 = C_0^\intercal \in \mathbb{R}^{d\times d}$ positive-definite, the solution is Gaussian at all times $t \geqslant 0$, $\rho_t^* = \mathsf{N}(m_t, C_t)$ with m_t and $C_t = C_t^\intercal$ solutions to

$$\begin{aligned}\dot{m}_t &= -\Gamma_t(m_t - b_t)\\ \dot{C}_t &= -\Gamma_t C_t - C_t\Gamma_t^\intercal + 2D_t.\end{aligned} \tag{C.3}$$

This implies in particular that

$$\nabla \log \rho_t^*(x) = -C_t^{-1}(x - m_t), \tag{C.4}$$

so that the probability flow equation for X_t and the equation for G_t written in (13) read

$$\begin{aligned}\dot{X}_t(x) &= (D_tC_t^{-1} - \Gamma_t)X_t(x) + \Gamma_t b_t - D_tC_t^{-1}m_t,\\ \dot{G}_t(x) &= (\Gamma_t^\intercal - C_t^{-1}D_t)G_t(x),\end{aligned} \tag{C.5}$$

with initial condition $X_0(x) = x$ and $G_0(x) = \nabla \log \rho_0(x) = -C_0^{-1}(x - m_0)$. It is easy to see that with $x \sim \rho_0 = \mathsf{N}(m_0, C_0)$ we have $X_t(x) \sim \rho_t^* = \mathsf{N}(m_t, C_t)$ since, from the first equation in (C.5), the mean and variance of X_t satisfy (C.3). Similarly, when $x \sim \rho_0 = \mathsf{N}(m_0, C_0)$, $G_0(x) \sim N(0, C_0^{-1})$, so that $G_t(x) \sim \mathsf{N}(0, C_t^{-1})$ because the second equation in (C.5) is linear and hence preserves Gaussianity. Moreover, $\mathbb{E}_0 G_t(x) = 0$ and $B_t = B_t^\intercal = \mathbb{E}_0[G_t(x)G_t^\intercal(x)]$ satisfies

$$\frac{\mathrm{d}}{\mathrm{d}t}B_t = (\Gamma_t^\intercal - C_t^{-1}D_t)B_t + B_t(\Gamma_t - D_tC_t^{-1}). \tag{C.6}$$

The solution to this equation is $B_t = C_t^{-1}$ since substituting this ansatz into (C.6) gives the equation for C_t^{-1} that we can deduce from (C.3)

$$\frac{\mathrm{d}}{\mathrm{d}t}C_t^{-1} = C_t^{-1}\dot{C}_tC_t^{-1} = -C_t^{-1}\Gamma_t - \Gamma_t^\intercal C_t^{-1} + 2C_t^{-1}D_tC_t^{-1}. \tag{C.7}$$

Note that if $\Gamma_t = \Gamma$, $b_t = b$, and $D_t = D$ are all time-independent, then $\lim_{t\to\infty}\rho_t = N(m_\infty, C_\infty)$ with $m_\infty = b$ and C_∞ the solution to the Lyapunov matrix equation

$$\Gamma C_\infty + C_\infty\Gamma^\intercal = 2D. \tag{C.8}$$

This means that at long times the coefficients at the right-hand sides of (C.5) also settle on constant values. However, X_t and G_t do not necessarily stop evolving; one situation where they too converge is when the OU process is in detailed balance, i.e. when $\Gamma = DA$ for some $A = A^\intercal \in \mathbb{R}^{d\times d}$ positive-definite. In that case, the solution to (C.8) is $C_\infty = A^{-1}$ and it is easy to see that at long times the right-hand sides of (C.5) tend to zero.

Remark 9. This last conclusion is actually more generic than for a simple OU process. For any SDE in detailed balance, i.e. that can be written as

$$\mathrm{d}X_t = -D(X_t)\nabla U(X_t)\mathrm{d}t + \nabla\cdot D(X_t)\mathrm{d}t + \sqrt{2}\sigma_t(X_t)\mathrm{d}W_t \tag{C.9}$$

where $U:\mathbb{R}^d \to \mathbb{R}_{>0}$ is a C^2-potential such that $Z = \int_{\mathbb{R}^d}\mathrm{e}^{-U(x)}\mathrm{d}x < \infty$, we have that $\lim_{t\to\infty}\rho_t(x) = Z^{-1}\mathrm{e}^{-U(x)}$, and the corresponding flows X_t and G_t eventually stop as $t\to\infty$. In this case, ρ_t follows gradient descent in W_2 over the energy

$$E[\rho] = \int_{\mathbb{R}^d}(U(x) + \log\rho(x))\rho(x)\mathrm{d}x. \tag{C.10}$$

The unique PDF minimizing this energy is $Z^{-1}\mathrm{e}^{-U(x)}$, and as $t\to\infty$ X_t converges towards a transport map between the initial ρ_0 and $Z^{-1}\mathrm{e}^{-U(x)}$.

Appendix D. Experimental details and additional examples

All numerical experiments were performed in `jax` using the `dm-haiku` package to implement the networks and the `optax` package for optimization.

D.1. Harmonically interacting particles in a harmonic trap

Network architecture. Both the single-particle energy $U_{\theta_t,1}:\mathbb{R}^d\to\mathbb{R}$ and two-particle interaction energy $U_{\theta_t,2}:\mathbb{R}^d\times\mathbb{R}^d\to\mathbb{R}$ are parameterized as single hidden-layer neural networks with the `swish` activation function [43] and `n_hidden` $=100$ hidden neurons. The hidden layer biases are initialized to zero while the hidden layer weights are initialized from a truncated normal distribution with variance $1/$`fan_in`, following the guidelines recommended in [21].

Optimization. The Adam [23] optimizer is used with an initial learning rate of $\eta=10^{-4}$ and otherwise default settings. At time $t=0$, the analytical relative loss

$$L[s_0]=\frac{\int|s_0(x)-\nabla\log\rho_0(x)|^2\rho_0(x)\mathrm{d}x}{\int|\nabla\log\rho_0(x)|^2\rho_0(x)\mathrm{d}x} \tag{D.1}$$

is minimized to a value less than 10^{-4} using knowledge of the initial condition $\rho_0=\mathsf{N}\left(\beta_0,\sigma_0^2 I\right)$ with $\sigma_0=0.25$. In (D.1), the expectation with respect to ρ_0 is approximated by an initial set of samples $x_j=(x_j^{(1)},x_j^{(2)},\ldots,x_j^{(N)})^\mathsf{T}$ with $j=1,\ldots,n$ drawn from ρ_0. In the experiments, we set $n=100$, which we found to be sufficient to obtain a few digits of relative accuracy on various quantities of interest. We set the physical timestep $\Delta t=10^{-3}$ and take `n_opt_steps` $=25$ steps of Adam on the standard sequential SBTM loss function (17) until the norm of the gradient is below `gtol` $=0.1$.

Analytical moments. First define the mean, second moment, and covariance according to

$$\begin{aligned}m_t^{(i)}&=\mathbb{E}\left[X_t^{(i)}\right],\\ M_t^{(ij)}&=\mathbb{E}\left[X_t^{(i)}\left(X_t^{(j)}\right)^\mathsf{T}\right],\\ C_t^{(ij)}&=M^{(ij)}-m^{(i)}\left(m^{(j)}\right)^\mathsf{T}.\end{aligned}$$

It is straightforward to show that the mean and covariance obey the dynamics

$$\dot m_t^{(i)}=-(m_t^{(i)}-\beta_t)+\frac{\alpha}{N}\sum_{k=1}^N\left(m_t^{(i)}-m_t^{(k)}\right), \tag{D.2}$$

$$\dot C_t^{(ij)}=-2(1-\alpha)C_t^{(ij)}+2DI\delta_{ij}-\frac{\alpha}{N}\sum_{k=1}^N\left(C_t^{(kj)}+C_t^{(ik)}\right). \tag{D.3}$$

Because the particles are indistinguishable so long as they are initialized from a distribution that is symmetric with respect to permutations of their labeling, the moments will satisfy the ansatz

$$m_t^{(i)}=\bar m(t),\quad i=1,\ldots,N \tag{D.4}$$

$$C_t^{(ij)}=C_d(t)\delta_{ij}+C_o(t)(1-\delta_{ij}),\quad i,j=1,\ldots,N. \tag{D.5}$$

The dynamics for the vector $\bar m:\mathbb{R}_{\geqslant 0}\to\mathbb{R}^{\bar d}$, as well as the matrices $C_d:\mathbb{R}_{\geqslant 0}\to\mathbb{R}^{\bar d\times\bar d}$ and $C_o:\mathbb{R}_{\geqslant 0}\to\mathbb{R}^{\bar d\times\bar d}$ can then be obtained from (D.2) and (D.3) as

$$\begin{aligned}\dot{\bar m}&=\beta_t-\bar m,\\ \dot C_d&=2(\alpha-1)C_d-2\frac{\alpha}{N}(C_d+(n-1)C_o)+2DI,\\ \dot C_o&=2(\alpha-1)C_o-2\frac{\alpha}{N}(C_d+(n-1)C_o).\end{aligned}$$

For a given $\beta:\mathbb{R}\to\mathbb{R}^{\bar d}$, these equations can be solved analytically in Mathematica as a function of time, giving the mean $m_t=\bar m(t)\otimes 1_N\in\mathbb{R}^{N\bar d}$ and covariance $C_t=(C_d(t)-C_o(t))\otimes I_{N\times N}+C_o(t)\otimes\left(1_N1_N^\mathsf{T}\right)\in\mathbb{R}^{N\bar d\times N\bar d}$. Because the solution is Gaussian for all t, we then obtain the analytical solution to the FPE $\rho_t^*=\mathsf{N}(m_t,C_t)$ and the corresponding analytical score $-\nabla\log\rho_t^*(x)=C_t^{-1}(x-m_t)$.

Potential structure. Here, we show that the potential for this example lies in the class of potentials described by (23). From equation (D.5), we have a characterization of the structure of the covariance matrix C_t for the analytical potential $U_t(x)=\frac{1}{2}(x-m_t)^\mathsf{T}C_t^{-1}(x-m_t)$. In particular, C_t is block circulant, and hence is block diagonalized by the roots of unity (the block discrete Fourier transform). That is, we may take a

'block eigenvector' of the form $\omega_k = \left(I_{\bar{d}\times\bar{d}}\rho^k, I_{\bar{d}\times\bar{d}}\rho^{2k}, \ldots, I_{\bar{d}\times\bar{d}}\rho^{(N-1)k}\right)^{\mathsf{T}}$ with $\rho = \exp(-2\pi i/N)$ for $k = 0, \ldots N-1$. By direct calculation, this block diagonalization leads to two distinct block eigenmatrices,

$$C_t = V \begin{pmatrix} C_d(t) + (N-1)C_o(t) & 0 & 0 & \ldots & 0 \\ 0 & C_d(t) - C_o(t) & 0 & \ldots & 0 \\ & & \ddots & & \\ 0 & 0 & \ddots & \ldots & 0 \\ 0 & 0 & 0 & \ldots & C_d(t) - C_o(t) \end{pmatrix} V^{-1}$$

where $V \in \mathbb{R}^{N\bar{d}\times N\bar{d}}$ denotes the matrix with block columns ω_k. The inverse matrix C_t^{-1} then must similarly have only two distinct block eigenmatrices given by $(C_d(t) + (N-1)C_o(t))^{-1}$ and $(C_d(t) - C_o(t))^{-1}$. By inversion of the block Fourier transform, we then find that

$$\left(C_t^{-1}\right)^{(ij)} = \bar{C}_d\delta_{ij} + \bar{C}_o(1-\delta_{ij})$$

for some matrices $\bar{C}_d, \bar{C}_o$. Hence, by direct calculation

$$\begin{aligned}(x-m_t)^{\mathsf{T}} C_t^{-1} (x-m_t) &= \sum_{i,j}^{N} \left(x^{(i)} - m_t^{(i)}\right)^{\mathsf{T}} \left(C_t^{-1}\right)^{(ij)} \left(x^{(j)} - m_t^{(j)}\right) \\ &= \sum_{i,j}^{N} \left(x^{(i)} - \bar{m}(t)\right)^{\mathsf{T}} \left(\bar{C}_d\delta_{ij} + \bar{C}_o(1-\delta_{ij})\right) \left(x^{(j)} - \bar{m}(t)\right) \\ &= \sum_{i}^{N} \left(x^{(i)} - \bar{m}(t)\right)^{\mathsf{T}} \bar{C}_d \left(x^{(i)} - \bar{m}(t)\right)^{\mathsf{T}} \\ &\quad + \sum_{i\neq j}^{N} \left(x^{(i)} - \bar{m}(t)\right)^{\mathsf{T}} \bar{C}_o \left(x^{(j)} - \bar{m}(t)\right).\end{aligned}$$

Above, we may identify the first term in the last line as $\sum_{i=1}^{N} U_1(x^{(i)})$ and the second term in the last line as $\frac{1}{N}\sum_{i\neq j}^{N} U_2(x^{(i)}, x^{(j)})$. Moreover, $U_2(\cdot,\cdot)$ is symmetric with respect to its arguments.

Analytical entropy. For this example, the entropy can be computed analytically and compared directly to the learned numerical estimate. By definition,

$$\begin{aligned} s_t &= -\int_{\mathbb{R}^{N\bar{d}}} \log\rho_t(x)\rho_t(x)\mathrm{d}x, \\ &= -\int_{\mathbb{R}^{N\bar{d}}} \left(-\frac{N\bar{d}}{2}\log(2\pi) - \frac{1}{2}\log\det C_t - \frac{1}{2}(x-m_t)^{\mathsf{T}}C_t^{-1}(x-m_t)\right)\rho_t(x)\mathrm{d}x, \\ &= \frac{N\bar{d}}{2}\left(\log(2\pi)+1\right) + \frac{1}{2}\log\det C_t. \end{aligned}$$

Additional figures. Images of the learned velocity field and potential in comparison to the corresponding analytical solutions can be found in figures D1 and D2, respectively. Further detail can be found in the corresponding captions. We stress that the two-dimensional images represent single-particle slices of the high-dimensional functions.

D.2. Soft spheres in an anharmonic trap

Network architecture. Both potential terms $U_{\theta_t,1}$ and $U_{\theta_t,2}$ are modeled as four hidden-layer deep fully connected networks with `n_hidden` $= 32$ neurons in each layer. The initialization is identical to appendix D.2.

Optimization and initialization. The Adam optimizer is used with an initial learning rate of $\eta = 5\times10^{-3}$ and otherwise default settings. At time $t = 0$, the loss (D.1) is minimized to a value less than 10^{-6} over n samples $X_0^{(i)} \sim \otimes_{j=1}^{N}\mathsf{N}(\beta_0, \sigma_0^2 I)$, $i = 1, \ldots, n$ with $\sigma_0 = 0.5$ and $n = 10^4$. Past this initial stage, the denoising loss is used with a noise scale $\sigma = 0.1$; we found that a higher noise scale regularized the problem and led to a smoother prediction for the entropy, at the expense of a slight bias in the moments. By increasing the number of samples n, the noise scale can be reduced while maintaining an accurate prediction for the entropy. The loss is minimized by taking `n_opt_steps` $= 25$ steps of Adam at each timestep. The physical timestep is set to $\Delta t = 10^{-3}$.

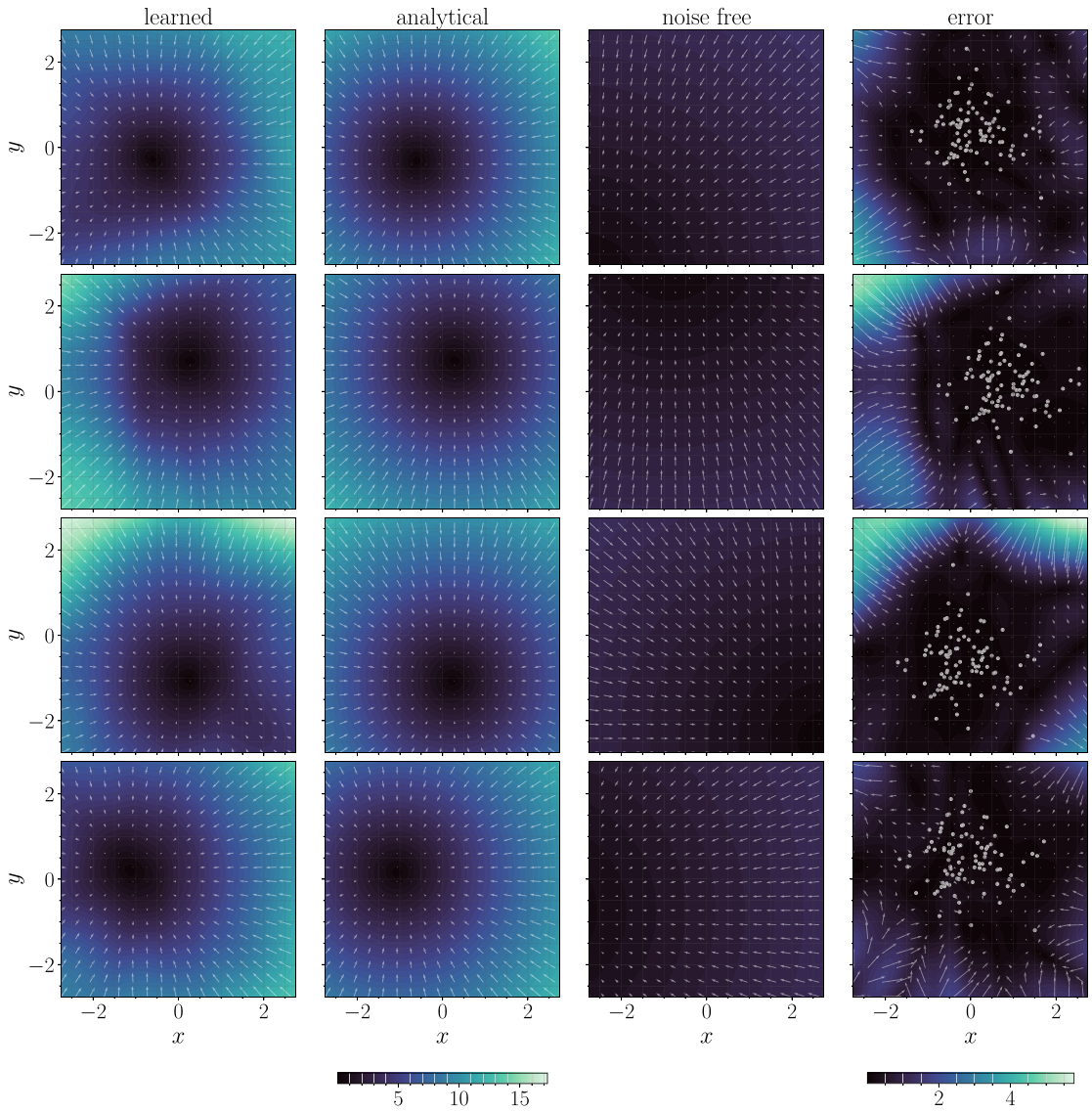

Figure D1. A system of $N = 50$ harmonically interacting particles in a harmonic trap: slices of the high-dimensional velocity field. Cross sections of the velocity field for $N = 50$ harmonically interacting particles in a moving harmonic trap. Columns depict the learned, analytical, noise-free, and error between the learned and analytical velocity fields, respectively. Rows indicate different time points, corresponding to $t = 1.25, 2.5, 3.75$, and 5.0, respectively. Each velocity field is plotted as a function of a single particle's coordinate (denoted as x and y); all other particle coordinates are fixed to be at the location of a sample. Color depicts the magnitude of the velocity field while arrows indicate the direction. Learned, analytical, and noise-free share a colorbar for direct comparison; the error occurs on a different scale and is plotted with its own colorbar. White circles in the error plot indicate samples projected onto the xy plane; locations of low error correlate well with the presence of samples.

Additional figures. Figures D3 and D4 show the full grid of covariance components for the SDE, learned, and noise free systems. The noise free underestimates the moments, while the learned and SDE agree well.

D.3. An active swimmer

Setup. We parameterize the score directly $s_t : \mathbb{R}^2 \to \mathbb{R}$ using a three hidden layer neural network with `n_hidden` $= 32$ neurons per hidden layer. Because the dynamics is anti-symmetric, we impose that $s(x, v) = -s(-x, -v)$.

Optimization and initialization. The network initialization is identical to the previous two experiments. The physical timestep is set to $\Delta t = 10^{-3}$. The Adam optimizer is used with an initial learning rate of $\eta = 10^{-4}$. At time $t = 0$ the loss (D.1) is minimized to a tolerance of 10^{-4} over $n = 10^4$ samples drawn from an initial distribution $\mathsf{N}(0, \sigma_0^2 I)$ with $\sigma_0 = 1$. The denoising loss is used with a noise scale $\sigma = 0.05$, using `n_opt_steps` $= 25$ steps of Adam until the norm of the gradient is below `gtol` $= 0.5$.

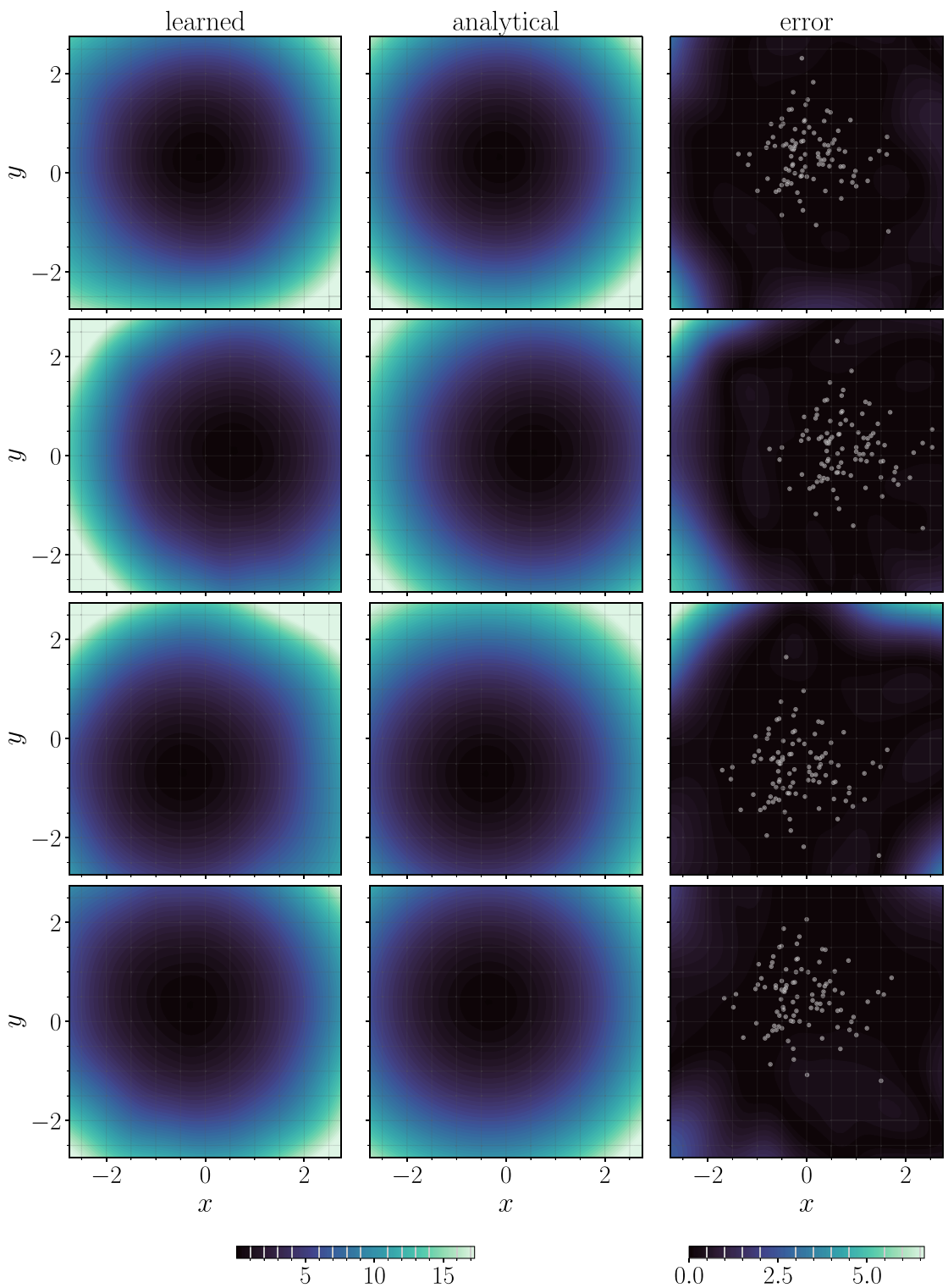

Figure D2. A system of $N = 50$ harmonically interacting particles in a harmonic trap: slices of the high-dimensional potential. Cross sections of the potential field $U_{\theta_t}(x)$ computed via (23). Columns depict the learned, analytical, and error between the learned and analytical, respectively. Rows indicate distinct time points, corresponding to $t = 1.25, 2.5, 3.75$, and 5.0, respectively. As in figure D1, each potential field is plotted as a function of a single particle's coordinate (denoted as x and y) with other particle coordinates fixed on a sample. All potentials are normalized via an overall shift so that the minimum value is zero. White circles in the error plot indicate samples from the learned system projected onto the xy plane.

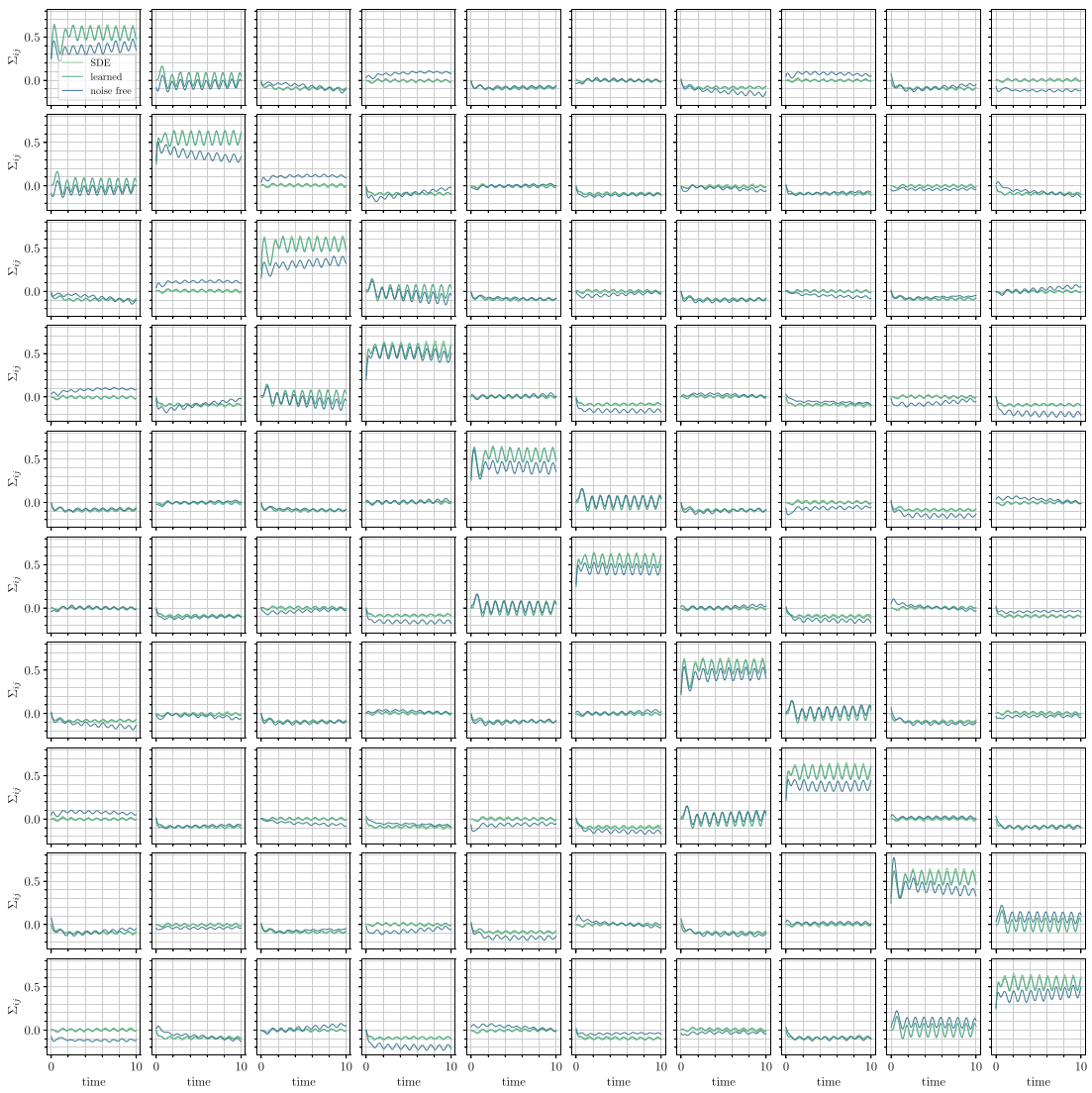

Figure D3. A system of $N = 5$ soft-sphere particles in an anharmonic trap: moments. All components of the covariance matrix over time for the circular trap motion. The learned system and the stochastic system agree well, while the noise free system underestimates the moments.

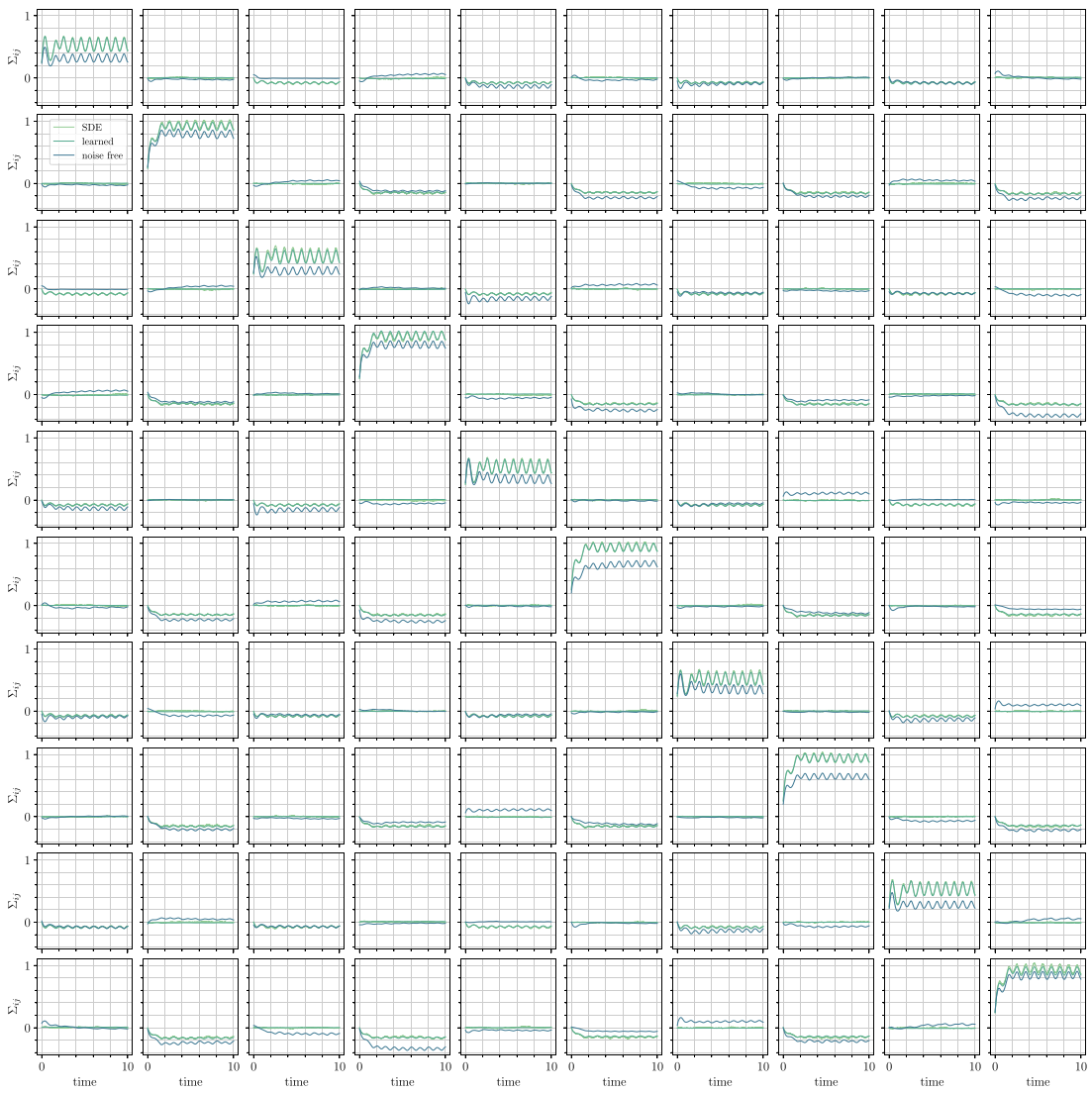

Figure D4. A system of $N = 5$ soft-sphere particles in an anharmonic trap: moments. All components of the covariance matrix over time for the linear trap motion. The learned system and the stochastic system agree well, while the noise free system underestimates the moments.

ORCID iD

Nicholas M Boffi https://orcid.org/0000-0003-1336-7568

References

[1] Bass R F 2011 *Stochastic Processes* vol 33 (Cambridge University Press)
[2] Blei D M, Kucukelbir A and McAuliffe J D 2017 Variational inference: a review for statisticians *J. Am. Stat. Assoc.* **112** 859–77
[3] Bruna J, Peherstorfer B and Vanden-Eijnden E 2022 Neural Galerkin scheme with active learning for high-dimensional evolution equations (arXiv:2203.01360)
[4] Chandler D 1987 *Introduction to Modern Statistical Mechanics* vol 5 (Oxford University Press)
[5] Dai B, He N, Dai H and Song L 2016 Provable Bayesian inference via particle mirror descent (arXiv:1506.03101)
[6] Dai C, Heng J, Jacob P E and Whiteley N 2020 An invitation to sequential Monte Carlo samplers (arXiv:2007.11936)
[7] De Bortoli V, Thornton J, Heng J and Doucet A 2021 Diffusion Schrödinger bridge with applications to score-based generative modeling (arXiv:2106.01357)
[8] Degond P and Mustieles F-J 1990 A deterministic approximation of diffusion equations using particles *SIAM J. Sci. Stat. Comput.* **11** 293–310
[9] Del Moral P, Doucet A and Jasra A 2006 Sequential Monte Carlo samplers *J. R. Stat. Soc.* B **68** 411–36
[10] Dockhorn T, Vahdat A and Kreis K 2022 Score-based generative modeling with critically-damped Langevin diffusion (arXiv:2112.07068)
[11] E W and Yu B 2017 The Deep Ritz method: a deep learning-based numerical algorithm for solving variational problems (arXiv:1710.00211)
[12] Evans L C 2012 *An Introduction to Stochastic Differential Equations* vol 82 (American Mathematical Society)

[13] Frenkel D and Smit B 2001 *Understanding Molecular Simulation: From Algorithms to Applications* vol 1 (Elsevier)
[14] Gardiner C 2009 *Stochastic Methods* 4th edn (Springer)
[15] Han J, Jentzen A and E W 2018 Solving high-dimensional partial differential equations using deep learning *Proc. Natl Acad. Sci.* **115** 8505–10
[16] Huang C-W, Chen R T Q, Tsirigotis C and Courville A 2021 Convex potential flows: universal probability distributions with optimal transport and convex optimization (arXiv:2012.05942)
[17] Hyvarinen A 2007 Connections between score matching, contrastive divergence and pseudolikelihood for continuous-valued variables *IEEE Trans. Neural Netw.* **18** 1529–31
[18] Hyvärinen A 2005 Estimation of non-normalized statistical models by score matching *J. Mach. Learn. Res.* **6** 695–709 (available at: http://jmlr.org/papers/v6/hyvarinen05a.html)
[19] Hyvärinen A 2007 Some extensions of score matching *Comput. Stat. Data Anal.* **51** 2499–512
[20] Hyvärinen A 2008 Optimal approximation of signal priors *Neural Comput.* **20** 3087–110
[21] Ioffe S and Szegedy C 2015 Batch normalization: accelerating deep network training by reducing internal covariate shift (arXiv:1502.03167)
[22] Jordan R, Kinderlehrer D and Otto F 1998 The variational formulation of the Fokker–Planck equation *SIAM J. Math. Anal.* **29** 1–17
[23] Kingma D P and Ba J 2017 Adam: a method for stochastic optimization (arXiv:1412.6980)
[24] Kloeden P E and Platen E 1992 Stochastic differential equations *Numerical Solution of Stochastic Differential Equations* (Springer) pp 103–60
[25] Kobyzev I, Prince S J D and Brubaker M A 2021 Normalizing flows: an introduction and review of current methods *IEEE Trans. Pattern Anal. Mach. Intell.* **43** 3964–79
[26] Kushner H J, Dupuis P G and Dupuis P 2001 *Numerical Methods for Stochastic Control Problems in Continuous Time* vol 24 (Springer)
[27] Lai C-H, Takida Y, Murata N, Uesaka T, Mitsufuji Y and Ermon S 2023 Improving score-based diffusion models by enforcing the underlying score Fokker-Planck equation (arXiv:2210.04296)
[28] Li L, Li Y, Liu J-G, Liu Z and Lu J 2020 A stochastic version of Stein variational gradient descent for efficient sampling *Commun. Appl. Math. Comput. Sci.* **15** 37–63
[29] Li L, Hurault S and Solomon J 2023 Self-consistent velocity matching of probability flows (arXiv:2301.13737)
[30] Lions P L and Sznitman A S 1984 Stochastic differential equations with reflecting boundary conditions *Commun. Pure Appl. Math.* **37** 511–37
[31] Liu Q 2017 Stein variational gradient descent as gradient flow (arXiv:1704.07520)
[32] Liu Q and Wang D 2018 Stein variational gradient descent as moment matching (arXiv:1810.11693)
[33] Liu Q and Wang D 2019 Stein variational gradient descent: a general purpose Bayesian inference algorithm (arXiv:1608.04471)
[34] Lu C, Zheng K, Bao F, Chen J, Li C and Zhu J 2022 Maximum likelihood training for score-based diffusion ODEs by high-order denoising score matching (arXiv:2206.08265)
[35] Lu J, Lu Y and Nolen J 2018 Scaling limit of the Stein variational gradient descent: the mean field regime (arXiv:1805.04035)
[36] Maoutsa D, Reich S and Opper M 2020 Interacting particle solutions of Fokker-Planck equations through gradient-log-density estimation *Entropy* **22** 802
[37] Marzouk Y, Moselhy T, Parno M and Spantini A 2016 An introduction to sampling via measure transport *Handbook of Uncertainty Quantification* ed R Ghanem, D Higdon and H Owhadi (Springer) pp 1–41
[38] Mittal G, Engel J, Hawthorne C and Simon I 2021 Symbolic music generation with diffusion models (arXiv:2103.16091)
[39] Nardini C, Fodor É, Tjhung E, van Wijland F'eric, Tailleur J and Cates M E 2017 Entropy production in field theories without time-reversal symmetry: quantifying the non-equilibrium character of active matter *Phys. Rev.* X **7** 021007
[40] Oksendal B 2003 *Stochastic Differential Equations* 6th edn (Springer)
[41] Papamakarios G, Nalisnick E, Rezende D J, Mohamed S and Lakshminarayanan B 2021 Normalizing flows for probabilistic modeling and inference *J. Mach. Learn. Res.* **22** 1–64
[42] Raissi M, Perdikaris P and Karniadakis G E 2019 Physics-informed neural networks: a deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations *J. Comput. Phys.* **378** 686–707
[43] Ramachandran P, Zoph B and Le Q V 2017 Searching for activation functions (arXiv:1710.05941)
[44] Rezende D J and Mohamed S 2016 Variational inference with normalizing flows (arXiv:1505.05770)
[45] Risken H 1996 Fokker-Planck equation *The Fokker-Planck Equation* (Springer) pp 63–95
[46] Robert C P and Casella G 2004 *Monte Carlo Statistical Methods* (Springer)
[47] Russo G 1990 Deterministic diffusion of particles *Commun. Pure Appl. Math.* **43** 697–733
[48] Saeedi A, Kulkarni T D, Mansinghka V K and Gershman S J 2017 Variational particle approximations *J. Mach. Learn. Res.* **18** 1–29
[49] Santambrogio F 2015 *Optimal Transport for Applied Mathematicians* vol 55 (Birkäuser) p 94
[50] Shen Z, Wang Z, Kale S, Ribeiro A, Karbasi A and Hassani H 2022 Self-consistency of the Fokker-Planck equation (arXiv:2206.00860)
[51] Sirignano J and Spiliopoulos K 2018 DGM: a deep learning algorithm for solving partial differential equations *J. Comput. Phys.* **375** 1339–64
[52] Song Y and Ermon S 2020 Generative modeling by estimating gradients of the data distribution (arXiv:1907.05600)
[53] Song Y and Ermon S 2020 Improved techniques for training score-based generative models (arXiv:2006.09011)
[54] Song Y and Kingma D P 2021 How to train your energy-based models (arXiv:2101.03288)
[55] Song Y, Sohl-Dickstein J, Kingma D P, Kumar A, Ermon S and Poole B 2021 Score-based generative modeling through stochastic differential equations (arXiv:2011.13456)
[56] Spohn H 2012 *Large Scale Dynamics of Interacting Particles* (Springer)
[57] Tabak E G and Turner C V 2013 A family of nonparametric density estimation algorithms *Commun. Pure Appl. Math.* **66** 145–64
[58] Tabak E G and Vanden-Eijnden E 2010 Density estimation by dual ascent of the log-likelihood *Commun. Math. Sci.* **8** 217–33
[59] Tailleur J and Cates M E 2008 Statistical mechanics of interacting run-and-tumble bacteria *Phys. Rev. Lett.* **100** 218103
[60] Villani C 2009 *Optimal Transport: Old and New* vol 338 (Springer)
[61] Vincent P 2011 A connection between score matching and denoising autoencoders *Neural Comput.* **23** 1661–74
[62] Zhang C, Bütepage J, Kjellström H and Mandt S 2019 Advances in variational inference *IEEE Trans. Pattern Anal. Mach. Intell.* **41** 2008–26