First-Order Trajectory Matching: Fast Ensemble Predictions of Chaotic, Turbulent, Stochastic Systems

Shreya Jha
Courant Institute School of Mathematics, Computing, and Data Science
New York University
New York, NY 10012
sj4964@nyu.edu

Timo Schorlepp
Courant Institute School of Mathematics, Computing, and Data Science
New York University
New York, NY 10012
timo.schorlepp@nyu.edu

Nicholas Geissler
Courant Institute School of Mathematics, Computing, and Data Science
New York University
New York, NY 10012
nwg2009@nyu.edu

Jules Berman
Courant Institute School of Mathematics, Computing, and Data Science
New York University
New York, NY 10012
jmb1174@nyu.edu

Benjamin Peherstorfer
Courant Institute School of Mathematics, Computing, and Data Science
New York University
New York, NY 10012
pehersto@cims.nyu.edu

Abstract

We introduce First-Order Trajectory Matching (FTM), a surrogate-modeling method that learns the first-order local transport of probability mass from trajectories of stochastic systems. By matching the symmetric first-order motion of trajectories, FTM learns the probability current velocity, whose flow preserves time marginals to match ensemble averages, while also capturing current-like trajectory quantities such as fluxes, circulations, and barrier-crossing currents. FTM learns the current velocity directly from trajectories, avoiding drift, diffusion, and score estimation. Our stability analysis separates discretization error from sampling variance and shows that the one-step simulation-free FTM loss is stable when temporal resolution and sample size are properly balanced. Across stochastic dynamical systems and PDE examples, we empirically demonstrate that FTM provides trajectory-aware ensemble predictions at low, deterministic-rollout cost.

1 Introduction

Overview: Fast ensemble predictions of physical systems Many dynamical systems in scientific applications are stochastic, or effectively so, because of chaos, coarse graining, partial observation, or unresolved degrees of freedom. Numerical high-fidelity simulators for such systems are often available, but repeated traditional numerical simulation is too costly for outer-loop tasks such as design, inverse problems, control, and uncertainty quantification. This motivates fast surrogate models trained from trajectory data. In the setting of (effectively) stochastic systems, however, the surrogate should not merely predict a single trajectory; it should rapidly propagate ensembles of trajectories and estimate their statistics, which describe the range of possible outcomes [13, 30, 54].

Operator learning and best forecasting predictions that collapse to the conditional mean Neural operators and related surrogate models [40, 37] are typically trained on state-transition data with a mean-squared error loss, which means that the optimal operator-learning map is the conditional expectation of the next state given the current one. This gives a fast model, but the resulting map is a point predictor. As a consequence, intrinsic variability is averaged out so that multiple possible future

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

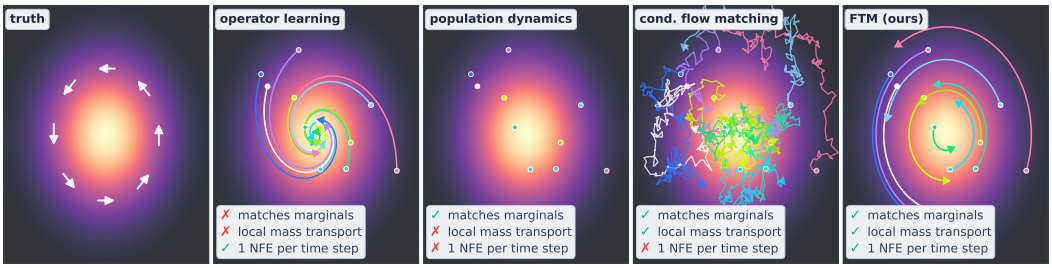


Figure 1: Statistically stationary process where probability mass circulates. A learned operator model leads to trajectories that spiral inward, collapsing to the mean. Population dynamics (marginal matching) match time marginals but with zero velocity (for minimum kinetic energy) and so miss the rotation. Conditional generative modeling such as conditional flow matching reproduces the mass transport but typically requires multiple solver steps per physical time step. FTM preserves time marginals while capturing the first-order rotational current with one neural network evaluation (NFE $= 1$) per time step. Details in Appendix A and Table 4.

states from the same observed current state are mapped to a single mean outcome; see Figure 1. Thus, such models fail to represent the transport of probability mass that determines ensemble statistics of stochastic systems [29, 16, 43].

Population dynamics match the evolution of time marginals only Population-dynamics methods infer a flow that matches the evolution of time marginals, which ensures that the right amount of probability mass is present in the right regions of the state space at each time. Examples include action matching [45, 6], the discrete inverse continuity equation method [7], and JKO/Wasserstein-gradient-flow and optimal transport-based methods [60, 14, 59]. Matching time marginals is sufficient to approximate ensemble averages but it is insufficient to determine ensemble dynamics, because it does not specify how mass flows over time. For example, a statistically stationary nonequilibrium system may have fixed time marginals while probability mass circulates persistently, which is not captured by matching time marginals alone; see Figure 1.

Conditional diffusion- and flow-based models and their inference costs An alternative is to model the full conditional transition law using conditional diffusion- and flow-based generative models [57, 38, 2, 16, 34]. Such models represent a distribution over future states rather than a single deterministic prediction, and therefore can capture stochastic variability and martingale-dominated quantities. This expressivity comes at a computational cost, typically requiring multiple neural-network evaluations or numerical solver steps to generate each next state. In autoregressive rollouts, these costs are incurred at every physical time step. Recent approaches such as consistency, mean-flow, and other distillation models aim to reduce costs; however, these methods have been developed primarily for static generative modeling and in practice often still trade off sample quality against the number of inference steps [56, 65, 26, 8, 21]. Furthermore, representing the underlying dynamic measure transport with a robust one-step map is intrinsically limited [61].

Our approach: First-Order Trajectory Matching (FTM) We propose First-Order Trajectory Matching (FTM) which learns the local probability mass transport describing how probability mass moves through the state space. As illustrated in Figure 1, FTM learns a deterministic velocity field that transports ensembles in the right way both globally and locally. Globally, its flow matches the evolution of the time marginals, so probability mass is placed in the correct regions of state space. Locally, it matches the trajectory-induced transport of mass, preserving currents such as circulations or barrier-crossing fluxes. Computationally, FTM leads to a surrogate model based on an ordinary differential equation (ODE), so ensembles can be propagated by fast deterministic rollouts. At the same time, this focus on a deterministic surrogate means that FTM is not intended to reproduce martingale-dominated path statistics such as hitting-time distributions or temporal correlations.

FTM matches local mass transport by using both directions of the observed trajectories: Looking forward asks where trajectories go next and gives a one-sided forecasting velocity; looking backward asks where they came from. Combining these incoming and outgoing views gives the symmetric first-order motion of probability mass through a space-time point. In the infinitesimal limit, this is the probability current velocity, which is the learning target of FTM.

The probability current velocity can be learned directly from stochastic trajectories, making FTM a fully trajectory-based surrogate-modeling approach. FTM uses a Stratonovich identity to turn current-velocity regression into a pathwise objective whose empirical form depends only on observed trajectory increments and evaluations of the candidate velocity field. We make this objective scalable by avoiding full-trajectory averaging, which is too expensive for long rollouts and minibatch training. We therefore introduce a chunked loss that can balance between full-trajectory averaging and one-step local training, and prove a bound that separates the effects of time discretization, chunk length, and sample size. Our analysis is developed specifically for finite-dimensional Markov diffusions with additive noise. This analysis justifies the simple one-step FTM loss used in our experiments when time resolution and sample sizes are balanced. The one-step loss is local in time, cheap to evaluate, avoids separate drift/diffusion/score estimation, and remains stable when sufficiently many trajectories are available with respect to the time resolution.

Literature review ***Probability current velocity in generative modeling*** Probability-flow ODEs and probability current velocities also appear in diffusion- and flow-based generative modeling [57, 38, 2, 1, 44], where they describe the auxiliary sampling-time transport from a reference law to a target law and often provide faster alternatives to SDE samplers [15]. In contrast, we are given physical-time trajectories of a stochastic system and seek the probability current velocity induced by those trajectories, not the velocity of an artificial path prescribed by a noising process or a stochastic interpolant. Thus, unlike standard score-matching or conditional flow-matching objectives, our objective learns the physical-time probability current directly from observed paths.

Probability current velocity in statistical physics and stochastic thermodynamics Probability currents are classical objects in stochastic mechanics and nonequilibrium diffusion theory [46, 17, 50]. Recent works in machine learning have used them in two main ways. First, in high-dimensional Fokker–Planck solvers, the probability current velocity is constructed from known drift-diffusion information together with an estimated score and then used to evolve particles along the probability flow [42, 9, 10]. Second, in stochastic thermodynamics, there are methods for learning probability current velocities or related thermodynamic force fields, primarily for entropy-production diagnostics [55, 35, 24, 11, 41, 20]. Most of this literature, which is also based on the Stratonovich identity (11), operates in the stationary regime, which excludes our setting of evolving $\rho(t)$. The work [24] gives a detailed analysis of trajectory length in this stationary regime, showing how longer averages reduce statistical error relative to the number of basis functions in linear approximations. Recent work relaxes stationarity [41, 20], but remains diagnostic, which means that the learned field is used to quantify or localize irreversibility and not integrated in time for predicting ensemble evolution. FTM instead develops a loss for learning a time-dependent velocity for predicting the evolution of a law $\rho(t)$. In particular, our analysis shows that, in this transient regime, finite-sample and discrete-time effects become central. We therefore introduce a chunk length that controls a statistical averaging scale, which in turn determines whether the empirical FTM loss provides a stable training signal for rollout.

Learning SDEs Classical SDE learning aims to estimate drift and diffusion coefficients, a long-standing problem that is difficult from finite, discretely observed trajectories and in high dimensions [47, 18, 12, 53]. Recent neural and latent SDEs parameterize these coefficients with neural networks [62, 36, 31, 19, 64, 4]. We bypass the challenge of SDE identification by learning the probability current velocity directly from trajectories, without separately estimating drift, diffusion, or score.

Contributions (a) Formulate ensemble prediction as learning local probability mass transport rather than conditional means or marginal-matching flows, identifying the probability current velocity as the deterministic surrogate for preserving marginals and first-order trajectory transport.

(b) Derive and analyze a scalable, one-step, and simulation-free loss function to learn the current velocity directly from trajectories, bypassing drift, diffusion, and score estimation.

(c) Show across stochastic and turbulent systems that FTM matches ensemble statistics and trajectory-based transport QoIs, while requiring only one neural-network evaluation per physical time step.

2 Setup, preliminaries, and problem formulation

Setup Let $(X(t))_{t\in[0,T]}$ be a stochastic process with state space $\mathbb{R}^d$, described by an SDE

$$\mathrm{d}X(t) = b(t, X(t))\mathrm{d}t + A(t)\mathrm{d}W(t)\,, \qquad X(0) \sim \rho(0)\,, \tag{1}$$

with an initial probability density $\rho(0)\colon \mathbb{R}^d \to [0,\infty)$, a sufficiently smooth drift vector field $b\colon [0,T]\times\mathbb{R}^d \to \mathbb{R}^d$, a sufficiently smooth diffusion matrix $A(t) \in \mathbb{R}^{d\times d}$ that does not depend on $X(t)$ ("additive noise"), and a standard Wiener process $(W(t))_{t\in[0,T]}$ on $\mathbb{R}^d$. For all theoretical statements, we will further assume uniform boundedness of the drift and diffusion matrix: $\|b(t,x)\|_2 \le b_{\max}$ for all $t\in[0,T], x\in\mathbb{R}^d$, and $\Sigma(t) := A(t)A(t)^\top \preceq \sigma_{\max} I_d$ for all $t\in[0,T]$. We denote the density of $X(t)$ as $\rho(t)\colon \mathbb{R}^d \to [0,\infty)$. In the following, we are given a data set

$$\mathcal{X} = \{X^{(i)}(t_k) \,|\, i = 1,\dots,N, k = 0,\dots,K\} \subset \mathbb{R}^d\,, \tag{2}$$

of $i = 1,\dots,N$ independent realizations of trajectories of (1) with K time steps $0 = t_0 < t_1 < \dots, < t_K = T$ over the time interval $[0,T]$ and time-step size $h > 0$, assumed equidistant for simplicity (all of the following extends to data that are irregularly sampled in time).

Problem formulation Given the trajectory data (2), we seek a deterministic velocity field $u : [0,T]\times\mathbb{R}^d \to \mathbb{R}^d$ whose induced ODE

$$\tfrac{\mathrm{d}}{\mathrm{d}t}\hat{x}(t) = u(t,\hat{x}(t))\,, \qquad \hat{x}(0) \sim \rho(0)\,, \tag{3}$$

can be used as a fast surrogate for ensemble prediction under the stochastic dynamics (1). The basic requirement for such a surrogate is marginal consistency with the stochastic process (1), i.e., if $\hat{x}(0) \sim \rho(0)$, then the ODE should satisfy $\hat{x}(t) \sim \rho(t)$ for $t \in [0,T]$. Marginal consistency is necessary for ensemble prediction because it ensures that the learned flow places probability mass in the correct regions of state space at each time. We seek a deterministic velocity field whose flow matches the time marginals of the stochastic process and additionally generates trajectories that are first-order consistent with sample paths of the SDE (1).

Remark 1. While our theoretical analysis is developed for finite-dimensional additive-noise diffusions (1), the FTM construction below is not inherently restricted to this setting. In particular, the Stratonovich identity (11) and symmetric-increment characterization (see Section 3.1) remain valid for state-dependent (i.e., multiplicative) noise under suitable regularity assumptions, and an extension to certain non-Markovian processes appears possible as well [63]. The discretization-bias and sampling-variance bounds derived below, however, do not directly account for state-dependent diffusions or non-Markovian dependencies, and we hence restrict ourselves to (1) here.

3 First-order trajectory matching (FTM)

With FTM, we do not only match where probability mass is globally but also how probability mass moves locally. Matching time marginals only ensures that the right amount of mass is globally present in the right regions of state space at each time. FTM additionally asks that, locally in space and time, the instantaneous transport of this mass agrees with that induced by the stochastic trajectories.

3.1 From global to local mass transport

FTM seeks a velocity that describes how probability mass, up to first order, passes through each space-time point (t,x), rather than only where the mass is located at time t.

Describing local mass transport with forward, backward, and symmetric increments A forward increment describes the first-order approximation of where trajectories currently at x move mass next,

$$\vec{b}_h(t,x) = \mathbb{E}_{\omega\sim\mathbb{P}}\left[h^{-1}\left(X_\omega(t+h) - X_\omega(t)\right)\big|X_\omega(t) = x\right]\,, \tag{4}$$

where $h > 0$ is a time-step size and $(X_\omega(t))_{t\in[0,T]}$ denotes a sample path of the SDE (1). In the limit, under suitable regularity, the expected forward increment $\vec{b}_h(t,x)$ converges point-wise to the drift of (1), $\vec{b}(t,x) := \lim_{h\downarrow 0}\vec{b}_h(t,x) = b(t,x)$ [32]. However, local mass transport through (t,x) is not only an outgoing quantity, it also depends on where the mass arriving at (t,x) came from. The backward increment captures this incoming mass transport,

$$\overleftarrow{b}_h(t,x) = \mathbb{E}_{\omega\sim\mathbb{P}}\left[h^{-1}\left(X_\omega(t) - X_\omega(t-h)\right)\big|X_\omega(t) = x\right]\,, \tag{5}$$

which converges to $\overleftarrow{b}(t,x) := \lim_{h\downarrow 0}\overleftarrow{b}_h(t,x) = b(t,x) - \Sigma(t)\nabla\log\rho(t,x)$ under suitable regularity assumptions [46, 3, 27]. With FTM, we aim to learn the velocity that describes the net flux of mass, which is obtained by combining the incoming and outgoing mass through the symmetric increment,

$$v_h(t,x) = \mathbb{E}_{\omega\sim\mathbb{P}}\left[(2h)^{-1}\left(X_\omega(t+h) - X_\omega(t-h)\right)\big|X_\omega(t) = x\right]. \tag{6}$$

Combining the limits of $\vec{b}_h$ and $\overleftarrow{b}_h$ for $h \downarrow 0$ shows the limit of the expected symmetric increment $\lim_{h\downarrow 0} v_h(t,x) = \frac{1}{2}(\vec{b}(t,x) + \overleftarrow{b}(t,x))$ [46].

Probability current velocity Using the forward and backward limits above, the limit velocity $\frac{1}{2}(\vec{b}(t,x) + \overleftarrow{b}(t,x))$ is

$$v(t,x) = b(t,x) - \tfrac{1}{2}\Sigma(t)\nabla \log \rho(t,x), \tag{7}$$

which is the probability current velocity of the SDE (1); see, e.g., [49, Chapter 4].

The importance of v is that the same velocity selected locally by the trajectories also transports the marginals globally. Indeed, using (7), the Fokker–Planck equation for (1) can be written as

$$\partial_t \rho(t,x) = -\nabla \cdot (b(t,x)\rho(t,x)) + \tfrac{1}{2}\Sigma(t) : \nabla^2 \rho(t,x) = -\nabla \cdot (\rho(t,x)v(t,x)) \,. \tag{8}$$

Equivalently, ρv is the probability current of the stochastic process (1). Thus, if $\hat{x}(0) \sim \rho(0)$ and $\hat{x}(t)$ evolves according to (3) with $u = v$, then the law of $\hat{x}(t)$ follows the same marginals as the SDE.

Therefore the probability current velocity satisfies exactly the two requirements we imposed. It matches the time marginals, so probability mass is globally in the correct places. It also arises as the symmetric first-order motion of the stochastic trajectories, so probability mass moves locally as it does under the SDE. We discuss further properties of and intuition for v in Appendices A and B, and also refer to the large body of literature that uses the probability current velocity [24, 57, 9, 41, 11]. FTM learns the probability current velocity rather than an arbitrary marginal-matching velocity.

3.2 A scalable loss function for learning the current velocity from trajectory data

At the core of FTM is the task of learning the probability current velocity v directly from trajectory data. This is not a standard regression problem, because values of $v(t,x)$ are unavailable in the data set (2). Although v can be represented as (7), estimating v through separately learning drift, diffusion, and score is typically challenging and unreliable; see Section 4. Instead, we are interested in a loss function whose minimizer is v, but whose empirical estimator uses only sample trajectories as given in the training data (2). We first reformulate the inaccessible regression objective as a trajectory-based loss using a Stratonovich identity. We then introduce a chunked estimator and analyze its bias-variance tradeoff, motivating the scalable one-step FTM loss used in our experiments.

An inaccessible regression objective Let us consider the parametrized velocity field $v_\theta : [0,T] \times \mathbb{R}^d \to \mathbb{R}^d$. If the current velocity v were known, we would have a standard regression objective

$$\min_{\theta \in \mathbb{R}^p} \mathbb{E}_{t\sim\mathcal{U}([0,T]),X(t)\sim\rho(t)} \left[\|v(t,X(t)) - v_\theta(t,X(t))\|_2^2\right] \tag{9}$$

over the parameters $\theta \in \mathbb{R}^p$, where times t are uniformly sampled $\mathcal{U}([0,T])$ and $X(t)$ is a sample of the time marginal $\rho(t)$. We expand the norm in (9) and drop the constant term to see that it is sufficient to minimize

$$J(\theta) = \mathbb{E}_{t\sim\mathcal{U}([0,T]),X(t)\sim\rho(t)} \left[\|v_\theta(t,X(t))\|_2^2 - 2\langle v(t,X(t)), v_\theta(t,X(t))\rangle\right] \,. \tag{10}$$

The quadratic term $\|v_\theta(t,X(t))\|_2^2$ can be directly estimated from data. The difficulty is the linear term $\langle v(t,X(t)), v_\theta(t,X(t))\rangle$ because it contains the current velocity v, which is unknown.

A pathwise loss via the Stratonovich identity The key observation is that we do not need to evaluate v pointwise to estimate the linear term in the objective (10). Instead, it is sufficient to be able to estimate its action on the parametrized velocity v_θ. This action can be estimated directly from SDE trajectories through the Stratonovich integral identity

$$\mathbb{E}_{\omega\sim\mathbb{P}}\left[\int_0^T \langle v_\theta(t,X_\omega(t)), v(t,X_\omega(t))\rangle \mathrm{d}t\right] = \mathbb{E}_{\omega\sim\mathbb{P}}\left[\int_0^T v_\theta(t,X_\omega(t)) \circ \mathrm{d}X_\omega(t)\right] \tag{11}$$

where $\circ$ denotes the Stratonovich dot product (see Appendix C.1 for a proof of this standard result for completeness, and, e.g., [24, 11, 41] where it is used for loss functions). The resulting objective is

$$J(\theta) = J_T(\theta) = \mathbb{E}_{\omega\sim\mathbb{P}}\left[\frac{1}{T}\int_0^T \|v_\theta(t,X_\omega(t))\|_2^2 \mathrm{d}t - \frac{2}{T}\int_0^T v_\theta(t,X_\omega(t)) \circ \mathrm{d}X_\omega(t)\right] , \tag{12}$$

which does not depend on the unknown probability current velocity v anymore. The objective has the same minimizer as the inaccessible regression loss (9) but now depends only on trajectories.

A pathwise loss with chunks The loss (12) depends on the entire trajectory over the time interval $[0, T]$, which can become intractable for long trajectories. We therefore introduce a chunk length τ with $h \leq \tau \leq T$, and train on trajectory segments or chunks of length τ. The chunked objective is

$$J_\tau(\theta) = \mathbb{E}_{t,\omega}\left[\frac{1}{\tau}\int_t^{t+\tau} \|v_\theta(s, X_\omega(s))\|_2^2 \mathrm{d}s - \frac{2}{\tau}\int_t^{t+\tau} v_\theta(s, X_\omega(s)) \circ \mathrm{d}X_\omega(s)\right], \tag{13}$$

with $t \sim \mathcal{U}([0, T-\tau])$, which still has the same minimizers as $J(\theta)$. Given trajectories (2) sampled at discrete time points of time-step size $h > 0$, we set $\tau = K_\tau h$ for an integer K_τ and approximate the Stratonovich integral, e.g., by the composite trapezoidal rule to obtain the empirical loss

$$\widehat{J}_{h,\tau}(\theta) = \frac{1}{N\tau}\sum_{i=1}^{N}\sum_{j=0}^{K_\tau-1}\Big[\|v_\theta(s_{i,j}, X^{(i)}(s_{i,j}))\|_2^2 h - \Big\langle v_\theta\left(s_{i,j+1}, X^{(i)}(s_{i,j+1})\right) + v_\theta\left(s_{i,j}, X^{(i)}(s_{i,j})\right), X^{(i)}(s_{i,j+1}) - X^{(i)}(s_{i,j})\Big\rangle\Big], \tag{14}$$

where $s_{i,j} = t_{m_i+j}$ for uniformly sampled chunk start indices $m_i \in \{0, \ldots, K - K_\tau\}$.

Scalable one-step FTM loss when time discretization and sample size are balanced The objective $\widehat{J}_{h,\tau}$ depends on two time scales: the data time step h and the chunk length τ. The step size h controls the bias from approximating the time integrals, while the chunk length τ controls how much stochastic path noise is averaged for one loss sample. The next bound makes this precise.

Proposition 1. *Suppose $v_\theta \colon [0,T] \times \mathbb{R}^d \to \mathbb{R}^d$ is uniformly bounded, such that there exists $V > 0$ with $\|v_\theta(t,x)\|_2 \leq V$ for all $t \in [0,T]$ and $x \in \mathbb{R}^d$, and uniformly Lipschitz continuous, such that $C_{div} := \sup_{t,x} \frac{1}{2}|\mathrm{Tr}\left(\Sigma(t)\nabla v_\theta(t,x)\right)| < \infty$. Then there exist constants $C, \tilde{C} > 0$ (related to the discretization error for the integrals) such that we have*

$$\mathbb{E}\left[\left(\widehat{J}_{h,\tau}(\theta) - J_\tau(\theta)\right)^2\right] \leq 2\left(V^4 + 12\left(V^2 b_{\max}^2 + \tfrac{\sigma_{\max} V^2}{\tau} + C_{div}^2\right) + \tilde{C}h\right)\tfrac{1}{N} + 4C^2V^2h^2\,. \tag{15}$$

Proof in Appendix C.2. The bound separates the effect of the discretization scale h from the effect of the averaging scale τ. The last term in the bound is the squared time-discretization bias. It decreases as the data are sampled more finely in time. The remaining terms are variance terms and scale like $1/N$ with the number of data samples N. Within this variance term, the contribution $\sigma_{\max}V^2/(N\tau)$ is the stochastic contribution from the diffusive part of the trajectories. Increasing τ averages the noise over a longer trajectory chunk; increasing N averages over more independent samples. Thus, stability (which means keeping the variance under control) is governed by both N and τ. In particular, for fixed τ, the variance remains bounded as $h \downarrow 0$. Thus, for a fixed τ, the discretization bias can be reduced by taking smaller time steps h without the variance growing unbounded. The variance-minimizing choice is $\tau = T$, but is often impractical because every training step requires processing full trajectories, which is expensive for long time horizons and limits minibatching.

We are therefore interested in $\tau \approx h$, which couples the averaging scale τ to the discretization scale h, and leads to the scalable, one-step FTM loss

$$\widehat{J}_{\mathrm{FTM}}(\theta) = \frac{1}{Nh}\sum_{i=1}^{N}\left[\|v_\theta(s_i, X^{(i)}(s_i))\|_2^2 h - \langle v_\theta(s_i, X^{(i)}(s_i)), X^{(i)}(s_i{+}h) - X^{(i)}(s_i{-}h)\rangle\right], \tag{16}$$

for uniformly sampled times $s_1, \ldots, s_N \in \{t_1, \ldots, t_{K-1}\}$. Here, we used the midpoint rule on chunks of size $\tau = 2h$ in (13), instead of the trapezoidal rule; see Appendix C.3 for further discussion and the relation of this loss to (6). The variance of the loss now behaves as $\sigma_{\max}V^2/(Nh)$. Hence, if N is fixed, the estimator becomes unstable as $h \downarrow 0$. However, the bound in (15) shows that this instability is more nuanced. The variance of the one-step FTM loss (16) remains controlled whenever Nh is sufficiently large.

The one-step FTM loss is the most scalable one and we use it in our experiments. It is local in time, simulation-free, requires only adjacent states and a single neural-network evaluation per minibatch data point, and avoids estimating drift, diffusion, or scores. Its variance scales like $1/(Nh)$, so it is appropriate when the time-step size h and the sample size N are balanced.

3.3 Time-marginal versus path-dependent inference errors

At inference time, we generate samples by solving the ODE (3) with the learned $u = v_\theta$. We denote by $(\hat{\rho}_\theta(t))_t$ the marginal laws of the $(\hat{x}_\theta(t))_t$ trajectories generated via v_θ. There are two inference

errors of interest. First, the marginal-matching error, which measures how close $\hat{\rho}_\theta(t)$ is to the true $\rho(t)$, and therefore whether ensemble averages of "time-local" quantities of interests (QoIs) are accurately estimated. Second, the error for *path-dependent QoIs*, which instead allows us to measure how accurately the learned ODE captures the path behavior of the SDE (1).

Marginal matching inference error The following proposition about the marginal-matching error is based on a standard Grönwall argument and shows that uniform approximation of the current velocity implies Wasserstein control of the generated ensemble. A similar proposition holds for learning *any* ODE (3) for some $u \neq v$ that produces the correct time marginals.

Proposition 2 (cf. e.g. [2, 5, 25])**.** *Suppose that the true probability current velocity v is uniformly Lipschitz-continuous with Lipschitz constant $L_v > 0$, and $\|v_\theta(t,x) - v(t,x)\|_2 \leq \epsilon$ for all $t \in [0,T], x \in \mathbb{R}^d$ for the learned velocity v_θ. Then $W_2^2(\hat{\rho}_\theta(t), \rho(t)) \leq \exp((2L_v+1)t)t\epsilon^2$.*

The proof is included in Appendix C.4 for completeness. Proposition 2 immediately gives bounds for time-local QoIs that depend only on the marginal distribution. If $\phi : \mathbb{R}^d \to \mathbb{R}$ is a scalar QoI, which is Lipschitz with constant L_ϕ, then $|\mathbb{E}_{X(t)\sim\rho(t)}[\phi(X(t))] - \mathbb{E}_{\hat{x}_\theta(t)\sim\hat{\rho}_\theta(t)}[\phi(\hat{x}_\theta(t))]| \leq L_\phi \exp((L_v+\frac{1}{2})t)\sqrt{t}\epsilon$ (cf. Appendix C.4). Thus, if v_θ approximates the probability current velocity v well, then the learned ODE produces accurate ensemble averages of local QoIs.

Path-dependent QoI inference error Marginal errors only measure where probability mass is located. By learning the current velocity, FTM also aims to match how probability mass moves, which means that sample trajectories generated with the learned ODE also approximate those of the SDE to first-order in the induced rate of change of probability mass; see, e.g., [10] where this is used for capturing entropy production rates. The following proposition shows an error bound for path-dependent QoIs $\Phi(X) = \Phi((X(t))_{t\in[0,T]}) := \int_0^T \phi(t, X(t)) \circ \mathrm{d}X(t)$ for $\phi\colon [0,T]\times\mathbb{R}^d \to \mathbb{R}^d$ again under uniform v_θ error control; see Appendix C.5 for a proof using standard arguments. *This* proposition is specific to FTM and the probability current velocity. In other words, targeting any other velocity field u for (3) with the same time-marginals does not permit control over the error in Φ, as the population dynamics example in Figure 1 suggests already; see also the characterization below.

Proposition 3. *Under the same assumptions as Proposition 2, suppose further that v_θ and ϕ are bounded, $\|v_\theta(t,x)\|_2 \leq B_{v_\theta}$ and $\|\phi(t,x)\|_2 \leq B_\phi$ for all $t \in [0,T]$, $x \in \mathbb{R}^d$, and Lipschitz on $\mathbb{R}^d$ with Lipschitz constants L_{v_θ} and L_ϕ, respectively. Then, with $C = (L_\phi B_{v_\theta} + L_{v_\theta} B_\phi)/(L_v + \frac{1}{2})$:*

$$|\mathbb{E}[\Phi(X)] - \mathbb{E}[\Phi(\hat{x}_\theta)]| \leq \left(\int_0^T \|\phi(t,\cdot)\|_{L^2(\rho(t))} \mathrm{d}t + C \exp\left((L_v + \tfrac{1}{2})T\right)\sqrt{T} \right) \epsilon\,. \tag{17}$$

The first term accounts for the local-in-time first-order range of change error. The second term is the marginal rollout error, which appears because the QoI is evaluated along samples from $\hat{\rho}(t)$ rather than $\rho(t)$, and is controlled by the Wasserstein bound of Proposition 2. The bound shows that models learned with FTM control both the quantities of interest under the learned ensemble and their first-order range of change, capturing fluxes and circulations that are ignored by marginal matching. The probability current velocity v is in fact the unique marginal-preserving velocity for which the expectations of all Stratonovich integral QoIs under the associated deterministic dynamics equal those under the original stochastic dynamic (Proposition 4 in Appendix B).

4 Experiments

Examples We consider the following examples; see Appendix D for details. **1. Duffing oscillator.** Samples evolve in phase space based on a double-well potential. A Brownian forcing is applied to the oscillator's acceleration equation. **2. Chaotic Rayleigh–Bénard convection.** This is a stochastically forced nine-mode model of Rayleigh–Bénard convection introduced in [51]. The system is of dimension nine and chaotic due to the thermal convection; all state variables are Brownian forced. **3. Stochastically forced Burgers.** This system serves as a simplified model of turbulence, capturing nonlinear advective transport, viscous dissipation, and the formation of sharp gradients. The stochastic forcing is given by a 10-mode Fourier expansion driven by independent Brownian motions. **4. Stochastically forced turbulence.** This system is governed by the two-dim. incompressible Navier–Stokes equation; Brownian forcing acts on the first ten Fourier modes.

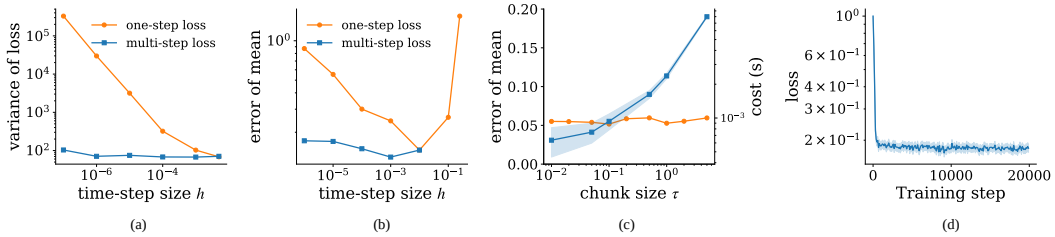


Figure 2: The one-step FTM loss is stable in the practically relevant time-resolution/sample-size balanced regime: longer chunks sizes τ are necessary only at very small time-step sizes h to reduce variance (a)–(b), but otherwise add substantial cost per training step and provide little accuracy gain at realistic resolutions (here for Duffing oscillator example) (c). FTM trains robustly (d).

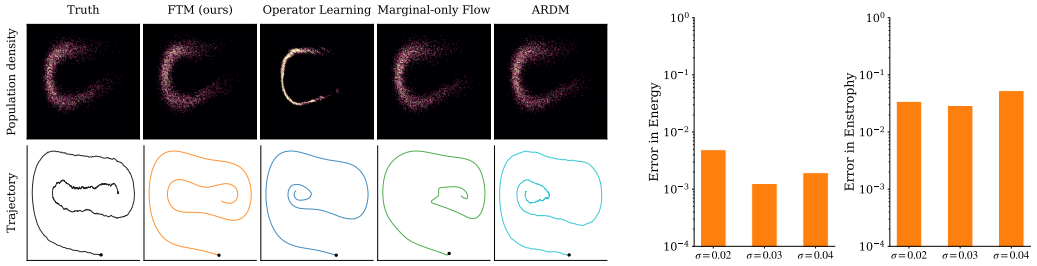


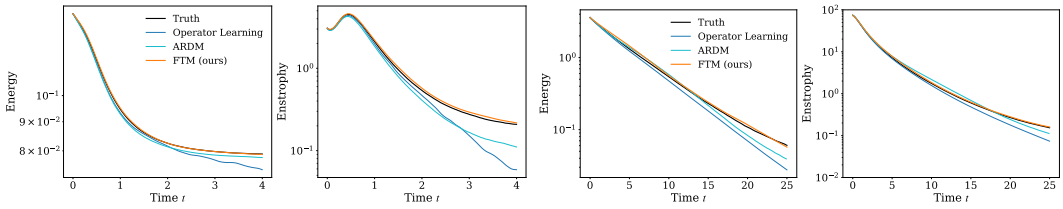
Figure 3: Duffing oscillator (left): FTM matches the evolving ensemble and generates physically meaningful trajectories. Stochastic Burgers (right): FTM is robust to increasing noise levels (σ).

The FTM loss is stable and robust to train Figure 2 shows on the example of the Duffing oscillator that the one-step FTM loss is a principled scalable estimator, in agreement with Proposition 1. Panel (a)–(b) show that for chunk size $\tau = h$, the variance/error grows only when h becomes very small; fixing the chunk size $\tau = 0.1 \gg h$ removes this small-h blow-up. Crucially, Panel (c) shows that in the time-resolution/sample-size balanced regime of our experiments (here for the Duffing oscillator), accuracy is nearly insensitive to τ; as shown in Proposition 1. Thus, longer chunks buy little accuracy in this balanced regime but incur substantially higher cost, whereas our one-step FTM loss retains a stable training signal at minimal expense. Finally, Panel (d) shows that this stability carries over to optimization. Across 100 random neural-network initializations, one-step FTM trains consistently and robustly. (The loss does not approach zero because (10) drops the constant from (9).)

FTM captures trajectory-based transport beyond ensemble statistics Figure 3 (top row) and Table 1 (distribution error W_2) show that FTM accurately matches time marginals and is competitive to marginal-only flows and explicit marginal-matching methods such as DICE [7]. FTM additionally matches local mass transport: The bottom row of Figure 3 shows that FTM follows the local motion of the stochastic trajectories, whereas marginal-only transport produces paths that do not resemble the stochastic dynamics. To quantify this, we consider trajectory-based quantities of interest (QoI):

Table 1: FTM matches the ensemble distribution and additionally gives the smallest trajectory-QoI errors, showing that FTM captures transport information missed by marginal-only methods.

	Duffing oscillator		Rayleigh–Bénard convection	
	dist. err. (W_2)	traj. QoI err.	dist. err. (W_2)	traj. QoI err.
T. stepper [48]	0.348(±2.0e−01)	1.630(±8.4e−03)	2.690(±1.6e+00)	1.650(±3.8e−03)
DICE [7]	0.340(±5.0e−01)	2.510(±7.6e−03)	0.200(±1.2e−01)	0.882(±3.8e−03)
Marginal [28]	0.081(±2.5e−02)	0.564(±8.3e−03)	0.111(±6.1e−02)	1.270(±3.8e−02)
SDE learning [23]	0.083(±2.8e−02)	0.263(±7.9e−03)	**0.054**(±2.3e−02)	0.053(±4.8e−03)
SDE matching [4]	0.549(±2.0e−01)	0.178(±7.2e−03)	0.227(±1.3e−01)	0.612(±4.4e−03)
Plug-in v (E)	0.507(±1.7e−01)	1.150(±6.1e−03)	0.374(±1.8e−01)	0.566(±5.0e−03)
FTM (ours)	**0.075**(±4.9e−02)	**0.020**(±5.7e−03)	0.060(±2.5e−02)	**0.027**(±4.6e−03)



(a) stochastic Burgers turbulence (Burgulence)

(b) stochastically forced turbulence (Navier-Stokes)

Figure 4: FTM predicts ensembles with accurate energy and enstrophy statistics of solution trajectories of the stochastic Burgers (left panel) and stochastically forced turbulence (right panel) example. (See Figure 7 in appendix for plots that include the standard deviation.)

Table 2: FTM gives the best reported errors while using only one neural-network function evaluation (NFE) per time step. Reported errors are computed as an average over physical time, with standard deviations computed across physical time steps. More extensive results are provided in the appendix in Table 5.

	Stochastic Burgers		Forced turbulence (Navier–Stokes)	
	error in energy	NFE	error in energy	NFE
Operator learning [58]	$2.21\text{e}{-2}(\pm 1.54\text{e}{-2})$	1	$2.13\text{e}{-1}(\pm 1.52\text{e}{-1})$	1
ARDM 50 [34]	$1.46\text{e}{-2}(\pm 3.39\text{e}{-3})$	50	$3.14\text{e}{-1}(\pm 1.47\text{e}{-1})$	50
ARDM 75 [34]	$1.36\text{e}{-2}(\pm 3.49\text{e}{-3})$	75	$2.75\text{e}{-1}(\pm 1.40\text{e}{-1})$	75
ARDM 100 [34]	$1.24\text{e}{-2}(\pm 3.12\text{e}{-3})$	100	$1.16\text{e}{-1}(\pm 1.09\text{e}{-1})$	100
AR-CFM [2, 38]	$2.71\text{e}{-3}(\pm 1.96\text{e}{-3})$	20	$1.39\text{e}{-1}(\pm 6.30\text{e}{-2})$	20
FTM (ours)	$\mathbf{1.95}\text{e}{-\mathbf{3}}(\pm 1.93\text{e}{-3})$	1	$\mathbf{3.43}\text{e}{-\mathbf{2}}(\pm 2.47\text{e}{-2})$	1

for Duffing, we measure the barrier-crossing current, and for Rayleigh–Bénard the rotational current (Appendix D). FTM gives the smallest trajectory-based QoI error averaged over time in Table 1.

FTM scales to stochastic PDEs with one NFE per physical time step We next demonstrate in Figures 4 and 5 and Table 2 that FTM remains effective for the stochastic Burgers and stochastically forced turbulence (Navier-Stokes) example, where each state is a high-dimensional discretized field. Operator learning is included to verify that these benchmarks contain non-negligible stochastic variability. Indeed, operator learning damps the ensemble and underestimates both the energy and enstrophy. An auto-regressive diffusion model (ARDM) [34] avoids this collapse by learning the full transition law, but rollout then requires solving an inner generative sampling problem at every physical time step, with 100 neural-network evaluations (NFEs) per time step in our experiments. FTM instead propagates ensembles with a single neural-network evaluation per physical time step while matching the mean and standard deviation of energy and enstrophy closely; see Figure 4 (and Figure 7 for standard deviation). We also report the energy spectrum decay in Figure 5, which shows that FTM captures it accurately. We further compare to a distilled auto-regressive conditional flow matching (ARCFM) model [2, 38] using mean-flow modeling [26]; see Tables 3 and 6. While closer in terms of costs to FTM, this distillation leads to at least one order of magnitude higher error than FTM.

Table 3: Stoch. Burgers: FTM vs. distilled CFM (MeanFlow [26])

	err. energy	NFE
AR-CFM [2, 38]	2.71e-3	20
MeanFlow-10 [26]	2.06e-2	10
MeanFlow-5 [26]	3.41e-2	5
MeanFlow-1[26]	8.22e-1	1
FTM (ours)	**1.95e-3**	1

5 Conclusions and limitations

By learning the probability current velocity directly from trajectories, FTM captures how probability mass moves locally, not just where it is globally. This lets the FTM deterministic surrogate model predict trajectory-based QoIs, including first-order transport structures such as fluxes and circulations, in addition to preserving ensemble statistics. As a result, FTM has the rollout cost of deterministic operator learning while avoiding collapse to the mean, and is faster than typical autoregressive generative models that require inner sampling at every physical time step. The numerical experiments demonstrate that this makes FTM a practical route to fast ensemble predictions.

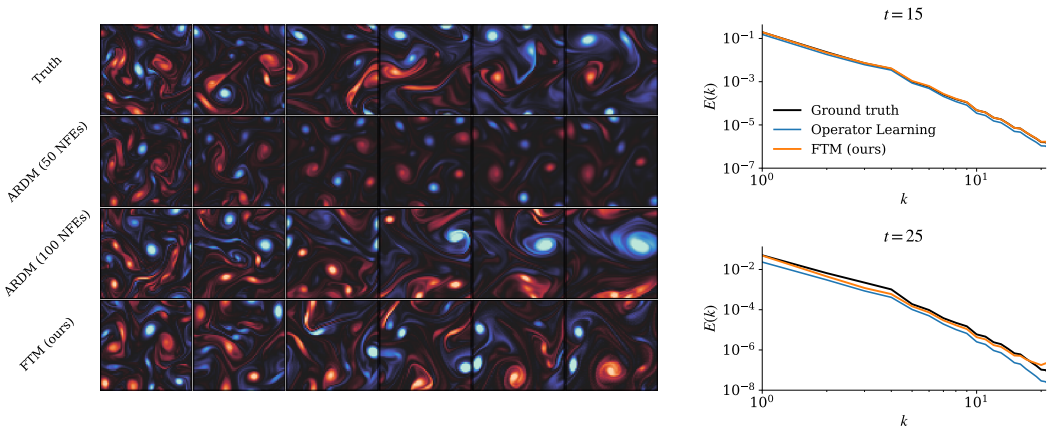


Figure 5: On stochastically forced turbulence, FTM produces accurate rollouts with one NFE per physical time step, while ARDM uses up to 100 inner sampling steps in our experiments (left). FTM also matches the energy spectrum decay well (right).

Limitations (i) FTM is designed to preserve time marginals and first-order trajectory transport, not the full stochastic path law. Thus, FTM can accurately capture current-like path-dependent QoIs, as shown in the experiments, but it should not be expected to reproduce martingale-dominated statistics such as hitting-time distributions and temporal autocorrelations. This is the key tradeoff: when the quantities of interest are ensemble statistics or current-like transport observables, FTM can provide trajectory-aware predictions at deterministic-rollout costs, instead of paying the often much higher cost of autoregressive generative modeling. *(ii)* The theory in this work establishes the estimator behavior for finite-dimensional additive-noise SDEs and clarifies the time-resolution/sample-size tradeoff of the one-step loss. These results, obtained using standard techniques, clarify the qualitative behavior of the estimator but they do not fully apply to the high-dimensional PDE/SPDE examples used in the empirical experiments. Extending the guarantees to broader PDE/SPDE settings, including establishing that bounds remain uniform under spatial discretization refinement, remains an important direction for future work.

Impact statement We are not expecting negative societal impacts that are specific to our approach.

Acknowledgements

The authors have been funded in part by the Air Force Office of Scientific Research (AFOSR), USA, award FA9550-24-1-0327. The authors were also partially supported by the US Department of Energy, Office of Scientific Computing Research, DOE Award DE-SC0024721.

References

[1] Michael Albergo, Nicholas M Boffi, and Eric Vanden-Eijnden. Stochastic interpolants: A unifying framework for flows and diffusions. *Journal of Machine Learning Research*, 26(209):1–80, 2025.

[2] Michael S Albergo and Eric Vanden-Eijnden. Building normalizing flows with stochastic interpolants. *arXiv preprint arXiv:2209.15571*, 2022.

[3] Brian DO Anderson. Reverse-time diffusion equation models. *Stochastic Processes and their Applications*, 12(3):313–326, 1982.

[4] Grigory Bartosh, Dmitry Vetrov, and Christian A. Naesseth. SDE Matching: Scalable and Simulation-Free Training of Latent Stochastic Differential Equations. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pages 3054–3070. PMLR, 2025.

[5] Joe Benton, George Deligiannidis, and Arnaud Doucet. Error bounds for flow matching methods. *arXiv preprint arXiv:2305.16860*, 2023.

[6] Jules Berman, Tobias Blickhan, and Benjamin Peherstorfer. Parametric Model Reduction of Mean-Field and Stochastic Systems via Higher-Order Action Matching. In *Advances in Neural Information Processing Systems*, 2024.

[7] Tobias Blickhan, Jules Berman, Andrew Stuart, and Benjamin Peherstorfer. DICE: Discrete inverse continuity equation for learning population dynamics, 2025.

[8] Nicholas M Boffi, Michael S Albergo, and Eric Vanden-Eijnden. How to build a consistency model: Learning flow maps via self-distillation. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

[9] Nicholas M Boffi and Eric Vanden-Eijnden. Probability flow solution of the Fokker–Planck equation. *Machine Learning: Science and Technology*, 4(3):035012, jul 2023.

[10] Nicholas M Boffi and Eric Vanden-Eijnden. Deep learning probability flows and entropy production rates in active matter. *Proceedings of the National Academy of Sciences*, 121(25):e2318106121, 2024.

[11] Nicholas M Boffi and Eric Vanden-Eijnden. Model-Free Learning of Probability Flows: Elucidating the Nonequilibrium Dynamics of Flocking. *Phys. Rev. Lett.*, 135:238301, Dec 2025.

[12] Lorenzo Boninsegna, Feliks Nüske, and Cecilia Clementi. Sparse learning of stochastic dynamical equations. *The Journal of chemical physics*, 148(24), 2018.

[13] Steven L Brunton, Bernd R Noack, and Petros Koumoutsakos. Machine Learning for Fluid Mechanics. *Annual Review of Fluid Mechanics*, 52(Volume 52, 2020):477–508, 2020.

[14] Charlotte Bunne, Laetitia Meng-Papaxanthos, Andreas Krause, and Marco Cuturi. Proximal Optimal Transport Modeling of Population Dynamics. *arXiv preprint arXiv:2106.06345*, 2022.

[15] Sitan Chen, Sinho Chewi, Holden Lee, Yuanzhi Li, Jianfeng Lu, and Adil Salim. The probability flow ODE is provably fast. *Advances in Neural Information Processing Systems*, 36:68552–68575, 2023.

[16] Yifan Chen, Mark Goldstein, Mengjian Hua, Michael S Albergo, Nicholas M Boffi, and Eric Vanden-Eijnden. Probabilistic Forecasting with Stochastic Interpolants and Föllmer Processes. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 6728–6756. PMLR, 21–27 Jul 2024.

[17] Raphaël Chetrite and Krzysztof Gawędzki. Eulerian and Lagrangian pictures of non-equilibrium diffusions. *Journal of Statistical Physics*, 137(5):890–916, 2009.

[18] Fabienne Comte and Valentine Genon-Catalot. Nonparametric drift estimation for i.i.d. paths of stochastic differential equations. *The Annals of Statistics*, 48(6):3336 – 3365, 2020.

[19] Kevin Course and Prasanth B. Nair. Amortized Reparametrization: Efficient and Scalable Variational Inference for Latent SDEs. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

[20] Biswajit Das and Sreekanth K Manikandan. Localizing entropy production along non-equilibrium trajectories. *arXiv preprint arXiv:2503.20427*, 2025.

[21] Mingyang Deng, He Li, Tianhong Li, Yilun Du, and Kaiming He. Generative Modeling via Drifting, 2026.

[22] Gideon Dresdner, Dmitrii Kochkov, Peter Norgaard, Leonardo Zepeda-Núñez, Jamie A. Smith, Michael P. Brenner, and Stephan Hoyer. Learning to correct spectral methods for simulating turbulent flows. 2022.

[23] Noura Dridi, Lucas Drumetz, and Ronan Fablet. Learning stochastic dynamical systems with neural networks mimicking the Euler–Maruyama scheme. In *2021 29th European Signal Processing Conference (EUSIPCO)*, pages 1990–1994, Dublin, Ireland, August 2021. IEEE.

[24] Anna Frishman and Pierre Ronceray. Learning Force Fields from Stochastic Trajectories. *Phys. Rev. X*, 10:021009, Apr 2020.

[25] Kenji Fukumizu, Taiji Suzuki, Noboru Isobe, Kazusato Oko, and Masanori Koyama. Flow matching achieves almost minimax optimal convergence. *arXiv preprint arXiv:2405.20879*, 2024.

[26] Zhengyang Geng, Mingyang Deng, Xingjian Bai, J Zico Kolter, and Kaiming He. Mean Flows for One-step Generative Modeling. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

[27] Ulrich G Haussmann and Etienne Pardoux. Time Reversal of Diffusions. *The Annals of Probability*, 14(4):1188–1205, Oct 1986.

[28] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, volume 33, 2020.

[29] Ruoxi Jiang, Peter Y. Lu, Elena Orlova, and Rebecca Willett. Training neural operators to preserve invariant measures of chaotic attractors. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

[30] George Em Karniadakis, Ioannis G. Kevrekidis, Lu Lu, Paris Perdikaris, Sifan Wang, and Liu Yang. Physics-informed machine learning. *Nature Reviews Physics*, 3(6):422–440, Jun 2021.

[31] Patrick Kidger, James Foster, Xuechen Li, and Terry Lyons. Efficient and Accurate Gradients for Neural SDEs. In *Advances in Neural Information Processing Systems*, volume 34, pages 18747–18761, 2021.

[32] Peter E. Kloeden and Eckhard Platen. *Numerical Solution of Stochastic Differential Equations*. Stochastic Modelling and Applied Probability. Springer, Berlin, Heidelberg, 1 edition, 1992. Springer-Verlag Berlin Heidelberg (1992); eBook ISBN 978-3-662-12616-5 (2013 reprint).

[33] Dmitrii Kochkov, Jamie A. Smith, Ayya Alieva, Qing Wang, Michael P. Brenner, and Stephan Hoyer. Machine learning–accelerated computational fluid dynamics. *Proceedings of the National Academy of Sciences*, 118(21), 2021.

[34] Georg Kohl, Li-Wei Chen, and Nils Thuerey. Benchmarking autoregressive conditional diffusion models for turbulent flow simulation. *Neural Networks*, 199:108641, 2026.

[35] Junang Li, Jordan M Horowitz, Todd R Gingrich, and Nikta Fakhri. Quantifying dissipation using fluctuating currents. *Nature communications*, 10(1):1666, 2019.

[36] Xuechen Li, Ting-Kam Leonard Wong, Ricky T. Q. Chen, and David Duvenaud. Scalable Gradients for Stochastic Differential Equations. In *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, volume 108 of *Proceedings of Machine Learning Research*, pages 3870–3882. PMLR, 2020.

[37] Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Fourier Neural Operator for Parametric Partial Differential Equations. In *International Conference on Learning Representations*, 2021.

[38] Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *The Eleventh International Conference on Learning Representations*, 2023.

[39] Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow Straight and Fast: Learning to Generate and Transfer Data with Rectified Flow. In *International Conference on Learning Representations*, 2023.

[40] Lu Lu, Pengzhan Jin, Guofei Pang, Zhongqiang Zhang, and George Em Karniadakis. Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators. *Nature Machine Intelligence*, 3(3):218–229, Mar 2021.

[41] Jinghao Lyu, Kyle J. Ray, and James P. Crutchfield. Learning Stochastic Thermodynamics Directly from Correlation and Trajectory-Fluctuation Currents. *arXiv*, 2504.19007, 2025.

[42] Dimitra Maoutsa, Sebastian Reich, and Manfred Opper. Interacting Particle Solutions of Fokker–Planck Equations Through Gradient–Log–Density Estimation. *Entropy*, 22(8), 2020.

[43] Roberto Molinaro, Samuel Lanthaler, Bogdan Raonić, Tobias Rohner, Victor Armegioiu, Stephan Simonis, Dana Grund, Yannick Ramic, Zhong Yi Wan, Fei Sha, Siddhartha Mishra, and Leonardo Zepeda-Núñez. Generative AI for fast and accurate statistical computation of fluids. 2025.

[44] Kunwoo Na, Junghyun Lee, Se-Young Yun, and Sungbin Lim. Probability-flow ODE in infinite-dimensional function spaces. *arXiv preprint arXiv:2503.10219*, 2025.

[45] Kirill Neklyudov, Rob Brekelmans, Daniel Severo, and Alireza Makhzani. Action Matching: Learning Stochastic Dynamics from Samples. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 25858–25889. PMLR, 2023.

[46] Edward Nelson. *Dynamical theories of Brownian motion*. Princeton University Press, 1967.

[47] Jan Nygaard Nielsen, Henrik Madsen, and Peter C Young. Parameter estimation in stochastic differential equations: an overview. *Annual Reviews in Control*, 24:83–94, 2000.

[48] Karl Otness, Arvi Gjoka, Joan Bruna, Daniele Panozzo, Benjamin Peherstorfer, Teseo Schneider, and Denis Zorin. An extensible benchmark suite for learning to simulate physical systems. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1)*, 2021.

[49] Grigorios A. Pavliotis. *Stochastic Processes and Applications: Diffusion Processes, the Fokker-Planck and Langevin Equations*, volume 60 of *Texts in Applied Mathematics*. Springer New York, New York, NY, 2014.

[50] Luca Peliti and Paolo Muratore-Ginanneschi. R. Fürth's 1933 paper "On certain relations between classical statistics and quantum mechanics" ["Über einige Beziehungen zwischen klassischer Statistik und Quantenmechanik", Zeitschrift für Physik, 81 143–162]. *The European Physical Journal H*, 48(1):4, May 2023.

[51] P. Reiterer, C. Lainscsek, F. Schürrer, C. Letellier, and J. Maquet. A nine-dimensional Lorenz system to study high-dimensional chaos. *Journal of Physics A: Mathematical and General*, 31(34):7121, aug 1998.

[52] Luc Rey-Bellet and Konstantinos Spiliopoulos. Irreversible Langevin samplers and variance reduction: a large deviations approach. *Nonlinearity*, 28(7):2081–2103, 2015.

[53] Tom Ryder, Andrew Golightly, A. Stephen McGough, and Dennis Prangle. Black-Box Variational Inference for Stochastic Differential Equations. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 4423–4432. PMLR, 2018.

[54] Benjamin Sanderse, Panos Stinis, Romit Maulik, and Shady E Ahmed. Scientific machine learning for closure models in multiscale problems: A review. *Foundations of Data Science*, 7(1):298–337, 2025.

[55] Udo Seifert. Stochastic thermodynamics, fluctuation theorems and molecular machines. *Reports on progress in physics*, 75(12):126001, 2012.

[56] Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. *International conference on machine learning*, 2023.

[57] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-Based Generative Modeling through Stochastic Differential Equations. In *International Conference on Learning Representations*, 2021.

[58] Kimberly Stachenfeld, Drummond B Fielding, Dmitrii Kochkov, Miles Cranmer, Tobias Pfaff, Jonathan Godwin, Can Cui, Shirley Ho, Peter Battaglia, and Alvaro Sanchez-Gonzalez. Learned Coarse Models for Efficient Turbulence Simulation. In *International Conference on Learning Representations (ICLR 2021)*, 2021.

[59] Alessandro Terpin, Nicolas Lanzetti, Marina Gadea, and Florian Dörfler. Learning Diffusion at Lightspeed. In *Advances in Neural Information Processing Systems*, 2024.

[60] Alexander Tong, Jessie Huang, Guy Wolf, David van Dijk, and Smita Krishnaswamy. TrajectoryNet: A Dynamic Optimal Transport Network for Modeling Cellular Dynamics. *arXiv preprint arXiv:2002.04461*, 2020.

[61] Panos Tsimpos, Daniel Sharp, and Youssef Marzouk. One-Shot Generative Flows: Existence and Obstructions, 2026.

[62] Belinda Tzen and Maxim Raginsky. Neural Stochastic Differential Equations: Deep Latent Gaussian Models in the Diffusion Limit. *arXiv preprint arXiv:1905.09883*, 2019.

[63] Jean-Claude Zambrini. Variational processes and stochastic versions of mechanics. *Journal of Mathematical Physics*, 27(9):2307–2330, 1986.

[64] Xi Zhang, Yuan Pu, Yuki Kawamura, Andrew Loza, Yoshua Bengio, Dennis L Shung, and Alexander Tong. Trajectory Flow Matching with Applications to Clinical Time Series Modeling. In *Advances in Neural Information Processing Systems*, volume 37, pages 107198–107224, 2024.

[65] Linqi Zhou, Stefano Ermon, and Jiaming Song. Inductive Moment Matching. *arXiv preprint arXiv:2503.07565*, 2025.

A Illustrative examples, and gauge freedom for global mass transport

We first specify the SDE toy example shown in Figure 1 in the main text via Example 1.

Example 1. We consider a statistically stationary, rotating Ornstein–Uhlenbeck process in $d = 2$ dimensions. The SDE (1) satisfied by $(X(t))_{t\in[0,T]}$ is given by

$$\begin{aligned} \mathrm{d}X(t) &= \mathrm{d}\begin{pmatrix} X_1(t) \\ X_2(t) \end{pmatrix} = b(t, X(t))\mathrm{d}t + A(t)\mathrm{d}W(t) \qquad (18) \\ &= -\left[\gamma X(t) + \Omega R X(t)\right]\mathrm{d}t + \sqrt{2D}\mathrm{d}W(t) = \begin{pmatrix} -\gamma & \Omega \\ -\Omega & -\gamma \end{pmatrix}\begin{pmatrix} X_1(t) \\ X_2(t) \end{pmatrix}\mathrm{d}t + \sqrt{2D}\begin{pmatrix} \mathrm{d}W_1(t) \\ \mathrm{d}W_2(t) \end{pmatrix}. \end{aligned}$$

Here, $\gamma > 0$ denotes the constant friction coefficient, $\Omega \in \mathbb{R}$ the constant angular velocity, and $D > 0$ the constant diffusion coefficient. We write $R = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}$ in (18). As is easy to verify from the Fokker–Planck equation (8), the stationary distribution of the process is $\rho_\infty = \mathcal{N}(0, \frac{D}{\gamma} I_2)$. We initialize $X(0) \sim \rho(0) = \rho_\infty$, which then implies $X(t) \sim \rho(t) \equiv \rho_\infty$ for all $t \in [0, T]$. Concretely, we take $\gamma = 0.35$, $\Omega = 1$, $D = 0.35$, $T = 1.5$ for Figure 1, though the numerical values do not matter for the qualitative behavior illustrated in Figure 1.

We now discuss what is shown in Figure 1 in more detail, and separately comment on the five panels of Figure 1 from left to right. Note that this illustrative figure, similar to Figure 6 below, is generated based on analytically known results, without actually training on any data. All of the trajectories that are shown start from the *same random initial conditions* $X(0) = \hat{x}(0) \sim \rho_\infty$ across panels, in order to compare the pathwise behavior of the different methods.

1. **truth:** We show the constant marginal distribution $\rho(t) \equiv \rho_\infty$ of the process on $\mathbb{R}^2$ as a heatmap, with white arrows indicating the direction of the analytically known probability current velocity $v(x) = \Omega R x$ from (7), which circulates counterclockwise for $\Omega > 0$. This probability current corresponds to sample paths of (18) circulating in the same direction, while preserving their time marginal distribution. The same heatmap of ρ_∞ is repeated in all other panels as a visual aid.
2. **operator learning:** The target for operator learning is the drift vector field b in (18) via (4). Learning only the drift, and solving the ODE $\frac{\mathrm{d}}{\mathrm{d}t}\hat{x}(t) = b(\hat{x}(t))$ at inference, leads to trajectories that spiral inwards and collapse onto the mean 0 of ρ_∞ at large t. Hence, this does not preserve the correct time marginals.
3. **population dynamics:** The main goal of population dynamics is to match the correct time marginals, which are stationary here. These methods typically assume that only *unpaired* samples from $X(t)$ at different times t are available, without trajectory information (which, as we argue in the present work, is actually available in many problems and useful to train with). Additional conditions then have to be imposed to single out a particular velocity field u for the ODE (3) among all those that yield the correct time marginals for $(\hat{x}(t))_t$. A natural condition is to demand minimum kinetic energy, resulting in $u(t, x) \equiv 0$ in the present example, which trivially satisfies $\hat{x}(t) = \hat{x}(0) \sim \rho_\infty$. However, this misses the circulation and local mass transport of the SDE (18) which is relevant for path-dependent quantities of interest. The resulting ODE $\frac{\mathrm{d}}{\mathrm{d}t}\hat{x}(t) = 0$ is trivial to integrate in this particular example, but more generally the velocity is not zero and thus time stepping with population dynamics can require more than one neural-network evaluation per time step as in our experiments.
4. **conditional flow matching:** Conditional flow matching methods are capable of reproducing the correct transition law of $X(t + h) \mid X(t)$ of the SDE (18). In the figure, we hence show sample paths of the SDE (18) generated using the Euler–Maruyama method. By construction, these will have the correct time marginals $\rho(t) \equiv \rho_\infty$, and circulate counterclockwise according to the probability current. However, sampling from (approximations of) $X(t+h) \mid X(t)$ with conditional flow matching in each time step increases the computational cost at inference.
5. **FTM:** Here, we integrate the ODE $\frac{\mathrm{d}}{\mathrm{d}t}\hat{x}(t) = v(t, \hat{x}(t))$, which preserves the time marginals, can be done at low computational cost, and the resulting pure circulation at constant radius mimics the behavior of true sample paths of (18).

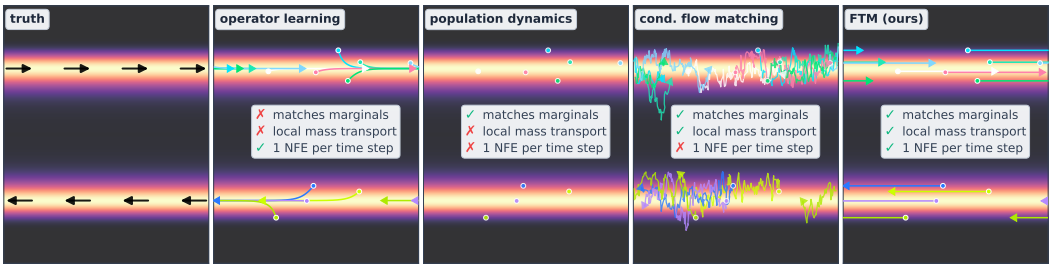


Figure 6: Statistically stationary "two-lane process" defined in Example 3.

In order to illustrate and contextualize these points further, it is useful to provide a slightly more abstract view on Example 1 using general irreversible perturbations of reversible processes that retain the Boltzmann equilibrium distribution:

Example 2 (e.g. [52]). Consider an SDE (1) with a diffusion matrix $A(t) = \sqrt{2}I_d$ and drift $b(t,x) = -\nabla U(x) + \ell(x)$ for a (quasi-) potential $U\colon \mathbb{R}^d \to [0,\infty)$ such that $Z = \int_{\mathbb{R}^d} \exp(-U(x))\mathrm{d}x < \infty$. Here, $\ell\colon \mathbb{R}^d \to \mathbb{R}^d$ denotes the "orthogonal component" of the drift, and we suppose $\langle \nabla U(x), \ell(x)\rangle = 0$ and $\nabla \cdot \ell(x) = 0$ for all $x \in \mathbb{R}^d$. The vector field ℓ represents an irreversible perturbation of the SDE. Under the stated assumptions, the stationary distribution of the process $(X(t))$ is still the Boltzmann distribution $\rho_\infty(x) = Z^{-1}\exp(-U(x))$ as can be verified from the Fokker–Planck equation (8):

$$\begin{aligned}\nabla \cdot (b(x)\rho_\infty(x) - \nabla\rho_\infty(x)) &= Z^{-1}\nabla\cdot([(-\nabla U(x) + \ell(x)) + \nabla U(x)]\exp(-U(x)))\\ &= Z^{-1}\nabla\cdot(\ell(x)\exp(-U(x))) = \rho_\infty(x)\,(\nabla\cdot\ell(x) - \langle \ell(x), \nabla U(x)\rangle) = 0\,.\end{aligned}$$

If we set $X(0) \sim \rho(0) = \rho_\infty$, then $\rho(t) = \rho_\infty$ for all $t > 0$. The probability current velocity (7) in this case is

$$v(t,x) = b(t,x) - \nabla \log \rho(t,x) = -\nabla U(x) + \ell(x) + \nabla U(x) = \ell(x)\,, \tag{19}$$

i.e., it corresponds to only the rotational or "swirling" component of the drift. Despite the time marginals $\rho(t) \equiv \rho_\infty$ being stationary, typical sample paths of the SDE (1) swirl according to ℓ, and this is correctly captured by the probability current velocity (19). The two-dimensional rotating Ornstein–Uhlenbeck process (18) as specified in Example 1 and shown in Figure 1 obviously falls exactly under the general class of examples introduced here: in this special case, we have $A(t) = \sqrt{2D}I_2$ and $b(t,x) = -\gamma x + \Omega R x$, such that $U(x) = \frac{\gamma}{2}\|x\|_2^2$ and $\ell(x) = v(x) = \Omega R x$. Evidently, $\langle \nabla U, \ell\rangle = 0$ and $\nabla\cdot\ell = 0$, such that $\rho_\infty \propto \exp(-U/D)$, and the probability current velocity corresponds to circulation with constant angular velocity Ω.

In view of the discussion above and in Section 3.1 in the main text, we remark that in general there is a "gauge freedom" in choosing a velocity field u for (3) with given time marginals. Indeed, as (8) shows, for any u that is a valid choice under the time marginal constraint, so is $u + w/\rho$ for $\nabla\cdot w = 0$. What we have argued in Section 3.1 in the main text based on the local probability mass transport of the SDE is that picking $u = v$ as the probability current velocity (7) is in some sense the canonical choice if we are given sample paths of the SDE (1).

Concretely, in the specific situation of Example 1 and more generally Example 2, picking any other velocity field $u = v + w/\rho$ with $\nabla\cdot w = 0$ for the probability flow ODE (3) that respects the time marginals would either introduce too much or too little swirl compared to the actual SDE (1). In particular, since in these examples the system is statistically stationary, $u \equiv 0$ is a valid choice for (3) with minimum kinetic energy and correct time marginals, such that $\hat{x}(t) \equiv \hat{x}(0) \sim \rho_\infty$ for all t, but this precisely misses the probability current that the actual SDE displays in its stationary state.

We provide a second illustrative example, similar to the rotating Ornstein–Uhlenbeck process (18) but on the domain $S^1 \times \mathbb{R}$ (which is not simply connected) in Example 3 and Figure 6.

Example 3. Consider the following statistically stationary "two-lane process" $(X(t))_{t\in[0,T]}$ on the infinite cylinder $S^1 \times \mathbb{R}$:

$$\mathrm{d}X(t) = \begin{pmatrix} w(X_2(t)) \\ -U'(X_2(t))\end{pmatrix}\mathrm{d}t + \begin{pmatrix}\sqrt{2D_1}\,\mathrm{d}W_1(t)\\ \sqrt{2D_2}\,\mathrm{d}W_2(t)\end{pmatrix}\,. \tag{20}$$

Table 4: Comparison of data-driven approaches for stochastic dynamics.

Method	What is learned?	What is captured?	Rollout
Operator learning	$\mathbb{E}[X_{t+h} \mid X_t]$	Point prediction; may collapse variability	Deterministic, 1 NFE
Marginal-only flow	Velocity matching $\rho(t)$	Marginals, but not currents	Deterministic, 1 NFE
Conditional generative model	$p(X_{t+h} \mid X_t)$	Marginals and stochastic transitions	Stochastic, multiple NFEs
FTM	Physical current velocity v	Marginals and currents	Deterministic, 1 NFE

Here, $U\colon \mathbb{R} \to [0,\infty)$ is a symmetric double well potential, concretely a quartic one $U(x_2) = \frac{\kappa}{4}(x_2^2 - a^2)^2$. The stationary distribution ρ_∞ of (20) is then uniform in x_1 and bimodal in x_2: $\rho_\infty(x_1, x_2) = Z\exp(-U(x_2)/D_2)$. We again initialize $X(0) \sim \rho(0) = \rho_\infty$. The function $w\colon \mathbb{R} \to \mathbb{R}$ in (20) induces a horizontal probability current according to $v(x_1, x_2) = (w(x_2), 0)^\top$. We choose w so that the current points to the right in the top well of U and to the left in the bottom well of U, concretely $w(x_2) = w_0 \tanh(x_2/l_0)$. Sample paths of the SDE (20) in the statistically stationary state then behave as two "lanes" moving in opposite directions, with rare vertical transitions depending on the barrier height. The behavior of different methods in this example is illustrated in Figure 6 and similar to Figure 1 and its discussion above. Note in particular that here, operator learning produces the correct horizontal movement but vertically collapses onto the modes of the two lanes, and that FTM does not show mode collapse, but cannot capture rare transitions of the true sample paths between the two wells. Numerical parameters in Figure 6 for concreteness: $\kappa = 2$, $a = 1$, $D_1 = 0.012$, $D_2 = 0.12$, $w_0 = 1.25$, $l_0 = 0.18$, $T = 3$, $x_1 \in [-L, L]/\sim$ for $L = 4$.

B Further properties of the probability current velocity

We first show that the probability current velocity v is uniquely characterized among marginal-preserving velocity fields by its ability to reproduce the expectations of all Stratonovich integral QoIs.

Proposition 4 (Uniqueness of the probability current velocity)**.** *Let u be a marginal-preserving velocity field for the process X, i.e., if*

$$\frac{\mathrm{d}}{\mathrm{d}t}\hat{x}(t) = u(t, \hat{x}(t)), \qquad \hat{x}(0) \sim \rho(0),$$

then $\hat{x}(t) \sim \rho(t)$ for all $t \in [0,T]$, where $X(t) \sim \rho(t)$. Under suitable regularity and integrability assumptions, the following are equivalent:

(i) For every smooth, compactly supported test vector field $\phi : [0,T] \times \mathbb{R}^d \to \mathbb{R}^d$,

$$\mathbb{E}[\Phi(X)] = \mathbb{E}[\Phi(\hat{x})], \qquad \Phi(Y) = \int_0^T \phi(t, Y(t)) \circ \mathrm{d}Y(t).$$

(ii) $u(t,\cdot) = v(t,\cdot)$ holds $\rho(t)$-almost surely for all $t \in [0,T]$.

Proof. By the Stratonovich identity (11) and the marginal consistency of u, we calculate

$$\mathbb{E}[\Phi(X)] - \mathbb{E}[\Phi(\hat{x})] = \int_0^T \int_{\mathbb{R}^d} \langle \phi(t,x), v(t,x) - u(t,x)\rangle \rho(t,x)\,\mathrm{d}x\,\mathrm{d}t.$$

If $u = v$ $\rho(t)$-almost surely for all t, the right-hand side vanishes, proving (ii)⇒(i).

Conversely, for (i)⇒(ii), proceed by contradiction and construct a suitable test function ϕ, assuming $u(t,x) - v(t,x) \neq 0$ for some $x \in \mathrm{supp}(\rho(t))$. □

The preceding proposition uniquely characterizes the probability current velocity through expected Stratonovich path functionals. A complementary characterization arises from minimum kinetic energy. Given the examples and discussion in Appendix A, one can ask in which situations the probability current velocity v corresponds to the minimum kinetic energy velocity field that generates the correct time marginals via the ODE (3). We have the following proposition:

Proposition 5. *Assume gradient drift $b(t,x) = -\nabla U(t,x)$, and that the diffusion matrix is isotropic, $\Sigma(t) = \sigma(t)^2 I_d$. Then:*

1. *the current velocity v is a gradient field,*
$$v(t,x) = -\nabla U(t,x) - \frac{\sigma(t)^2}{2}\nabla \log \rho(t,x) = -\nabla\left(U(t,x) + \frac{\sigma(t)^2}{2}\log \rho(t,x)\right).$$
2. *v is the unique minimizer of the kinetic energy functional*
$$\mathcal{K}(u) := \int_0^T \int_{\mathbb{R}^d} \rho(t,x)\,\|u(t,x)\|_2^2 \,\mathrm{d}x\,\mathrm{d}t,$$
among all velocity fields u satisfying the continuity equation
$$\partial_t \rho + \nabla\cdot(\rho u) = 0.$$

Hence, in this setting, the probability current velocity v coincides with the minimal kinetic energy velocity field associated with the prescribed marginal path $(\rho(t))_{t\in[0,T]}$.

Proof. Under the assumptions $b = -\nabla U$ and $\Sigma = \sigma^2 I_d$, we compute
$$\begin{aligned} v(t,x) &= b(t,x) - \frac{1}{2}\Sigma(t)\nabla\log\rho(t,x) = -\nabla U(t,x) - \frac{\sigma(t)^2}{2}\nabla\log\rho(t,x) \\ &= -\nabla\left(U(t,x) + \frac{\sigma(t)^2}{2}\log\rho(t,x)\right). \end{aligned}$$
Thus v is indeed a gradient field.

Next, v is admissible for the marginal-matching problem because it satisfies
$$\partial_t\rho + \nabla\cdot(\rho v) = 0.$$
We now show that among all admissible vector fields, there is at most one admissible gradient field. Suppose
$$u_1 = \nabla\phi_1, \qquad u_2 = \nabla\phi_2$$
both satisfy
$$\partial_t\rho + \nabla\cdot(\rho u_i) = 0, \qquad i = 1,2.$$
Subtracting the two equations gives
$$\nabla\cdot\big(\rho(u_1 - u_2)\big) = 0.$$
Let
$$w := u_1 - u_2 = \nabla\psi, \qquad \psi := \phi_1 - \phi_2.$$
Then
$$\nabla\cdot(\rho\nabla\psi) = 0.$$
Assuming standard decay at infinity (or, alternatively, no-flux boundary conditions on a bounded domain), we multiply by ψ and integrate:
$$0 = \int_{\mathbb{R}^d} \psi\,\nabla\cdot(\rho\nabla\psi)\,\mathrm{d}x = -\int_{\mathbb{R}^d}\rho\,\|\nabla\psi\|_2^2\,\mathrm{d}x.$$
Hence
$$\nabla\psi = 0 \qquad \text{a.e. on } \{\rho > 0\},$$
and therefore
$$u_1 = u_2 \qquad \text{a.e. with respect to } \rho(t,x)\,\mathrm{d}x\,\mathrm{d}t.$$
So there is at most one admissible gradient field.

Finally, the minimizer of the kinetic energy functional
$$\mathcal{K}(u) = \int_0^T\int_{\mathbb{R}^d}\rho\,\|u\|_2^2\,\mathrm{d}x\,\mathrm{d}t$$
subject to
$$\partial_t\rho + \nabla\cdot(\rho u) = 0$$
is known to be a gradient field. Since v is both admissible and a gradient field, and such a field is unique, it follows that v must coincide with the unique minimizer. Therefore v is exactly the minimal kinetic energy velocity field. □

Remark 2. In general, one should *not* claim that v is smoother, or has a smaller Lipschitz constant, than the minimal kinetic energy field. In the isotropic gradient-drift setting, the correct statement is stronger: $v = u_{\min}$ where $u_{\min}$ denotes the minimal kinetic energy admissible velocity field. The equality holds because v is the unique admissible gradient field, not because it enjoys any a priori superior smoothness.

Remark 3. If $\Sigma(t)$ is a general anisotropic positive definite matrix, then even if

$$b(t,x) = -\nabla U(t,x),$$

the current velocity

$$v(t,x) = b(t,x) - \frac{1}{2}\Sigma(t)\nabla \log \rho(t,x)$$

need not be a Euclidean gradient field. Thus the implication

$$\text{gradient drift} \quad \Longrightarrow \quad \text{gradient current velocity}$$

is automatic only in the isotropic case $\Sigma(t) = \sigma(t)^2 I_d$, or in special commutative situations. A natural anisotropic gradient-flow structure appears, however, when

$$b(t,x) = -\Sigma(t)\nabla U(t,x).$$

Then

$$v(t,x) = -\Sigma(t)\nabla\left(U(t,x) + \frac{1}{2}\log \rho(t,x)\right),$$

and v is the minimizer of the $\Sigma(t)^{-1}$-weighted kinetic energy

$$\int_0^T \int_{\mathbb{R}^d} \rho(t,x)\, \langle u(t,x), \Sigma(t)^{-1} u(t,x)\rangle \, \mathrm{d}x \, \mathrm{d}t$$

subject to the same continuity equation constraint.

Remark 4. Therefore, if one studies a pure overdamped Langevin diffusion with isotropic noise,

$$\mathrm{d}X(t) = -\nabla U(t, X(t)) \, \mathrm{d}t + \sigma(t) \, \mathrm{d}W(t),$$

then the current velocity field v and the minimal kinetic energy velocity field are the same object. In such an example, there is no principled distinction to visualize between the two. To obtain a visible difference, one needs a genuinely irreversible example, for instance by adding a rotational component to a gradient drift as in the examples of Appendix A.

C Proofs

C.1 Proof of Stratonovich integral identity, and loss derivation

Here, for completeness, we show that the well-known identity (11) used in the main text holds. In order to show that

$$\mathbb{E}\left[\int_0^T v_\theta(t, X(t)) \circ \mathrm{d}X(t)\right] = \mathbb{E}\left[\int_0^T \langle v_\theta(t, X(t)), v(t, X(t))\rangle \mathrm{d}t\right],$$

we first convert the Stratonovich integral on the left-hand side to an Itô integral, such that

$$\int_0^T v_\theta(t, X(t)) \circ \mathrm{d}X(t) = \int_0^T \langle v_\theta(t, X(t)), \mathrm{d}X(t)\rangle + \frac{1}{2}\int_0^T \mathrm{Tr}\left[\Sigma(t)\nabla v_\theta(t, X(t))\right] \mathrm{d}t$$
$$= \int_0^T \langle v_\theta(t, X(t)), b(t, X(t))\rangle \mathrm{d}t + \int_0^T \langle v_\theta(t, X(t)), A(t)\mathrm{d}W(t)\rangle + \frac{1}{2}\int_0^T \mathrm{Tr}\left[\Sigma(t)\nabla v_\theta(t, X(t))\right] \mathrm{d}t\,.$$

Taking the expectation over X on both sides then removes the second term on the right-hand side, since the Itô integral is non-anticipating. Finally, an integration by parts, supposing boundary terms

are 0, yields

$$\mathbb{E}\left[\int_0^T \mathrm{Tr}\left[\Sigma(t)\nabla v_\theta(t,X(t))\right]\mathrm{d}t\right] = \int_0^T \int_{\mathbb{R}^d} \mathrm{Tr}\left[\Sigma(t)\nabla v_\theta(t,x)\right]\rho(t,x)\mathrm{d}x\mathrm{d}t$$
$$= \sum_{i,j=1}^d \int_0^T \int_{\mathbb{R}^d} \Sigma_{ij}(t)\partial_i v_{\theta,j}(t,x)\rho(t,x)\mathrm{d}x\mathrm{d}t = -\sum_{i,j=1}^d \int_0^T \int_{\mathbb{R}^d} \Sigma_{ij}(t)v_{\theta,j}(t,x)\underbrace{\partial_i\rho(t,x)}_{=\rho\partial_i\log\rho}\mathrm{d}x\mathrm{d}t$$
$$= -\mathbb{E}\left[\int_0^T \langle v_\theta(t,X(t)), \Sigma(t)\nabla\log\rho(t,X(t))\rangle \mathrm{d}t\right],$$

which completes the proof.

We can hence write the objective (10) for the velocity field to be learned as

$$\begin{aligned} J_T(\theta) &= \mathbb{E}_{t\sim\mathcal{U}([0,T]),X(t)\sim\rho(t)}\left[\|v_\theta(t,X(t))\|_2^2 - 2\langle v(t,X(t)), v_\theta(t,X(t))\rangle\right] \\ &= \frac{1}{T}\left(\int_0^T \mathbb{E}_{X(t)\sim\rho(t)}\left[\|v_\theta(t,X(t))\|_2^2 - 2\langle v(t,X(t)), v_\theta(t,X(t))\rangle\right]\mathrm{d}t\right) \\ &= \frac{1}{T}\mathbb{E}_{\omega\sim\mathbb{P}}\left[\int_0^T \|v_\theta(t,X_\omega(t))\|_2^2\mathrm{d}t - 2\int_0^T v_\theta(t,X_\omega(t))\circ\mathrm{d}X_\omega(t)\right], \end{aligned}$$

and consider discretizations of the Stratonovich integral to obtain tractable estimators given trajectory data (2). For example, using the composite trapezoidal rule with step size $h > 0$ (such that $T/h = K$ and $t_j = jh$, $j = 0,\ldots,K$) for the Stratonovich integral, and a left Riemann sum for the squared velocity field, we get

$$\begin{aligned} J_{h,T}(\theta) = \frac{1}{T}\mathbb{E}_{\omega\sim\mathbb{P}}\Bigg[\sum_{j=1}^{K-1} &\left\|v_\theta\left(t_j, X_\omega(t_j)\right)\right\|_2^2 h \\ &- 2\frac{v_\theta\left(t_{j+1}, X_\omega(t_{j+1})\right) + v_\theta\left(t_j, X_\omega(t_j)\right)}{2}\left(X_\omega(t_{j+1}) - X_\omega(t_j)\right)\Bigg]. \end{aligned}$$

C.2 Proof of Proposition 1

We first want to remark here that the stated variance bound in (15) of the main text will often be overly pessimistic because of the last step in (22) below; since we deal with time averages, one could use ergodicity and mixing results to get much tighter variance bounds. We do not pursue this here since our simple, uniform-in-time bounds are enough to show that the variance of the objective $\widehat{J}_{h,\tau}(\theta)$ remains bounded as $h \downarrow 0$ at fixed N, in contrast to the symmetric difference-based objective from Proposition 6 below.

Proof of Proposition 1. We split the mean squared error into variance plus bias squared, i.e.,

$$\mathbb{E}\left[\left(\widehat{J}_{h,\tau}(\theta) - J_\tau(\theta)\right)^2\right] = \underbrace{\mathbb{E}\left[\left(\widehat{J}_{h,\tau}(\theta) - J_{h,\tau}(\theta)\right)^2\right]}_{\leq\left(2V^4+24\left(V^2 b_{\max}^2 + \frac{\sigma_{\max}V^2}{\tau} + C_{\mathrm{div}}^2\right)+2\tilde{C}h\right)\frac{1}{N}} + \underbrace{\left(J_{h,\tau}(\theta) - J_\tau(\theta)\right)^2}_{\leq 4C^2V^2h^2}, \quad (21)$$

where the indicated inequalities are what we want to show here. The bias arises from the discretization of the continuous time integrals. Under sufficient regularity of the drift b, diffusion matrix A, and velocity field v_θ, the trapezoidal approximation of the Stratonovich integral converges with weak order $O(h)$ [32]. Therefore, there exists a constant $C > 0$ such that the bias is bounded by $2CVh$ (note that the factor of 2 is introduced here by convention, to align with Proposition 6 below), and squaring this yields the quoted upper bound. The variance is

$$\mathbb{E}\left[\left(\widehat{J}_{h,\tau}(\theta) - J_{h,\tau}(\theta)\right)^2\right] = \frac{1}{N}\mathrm{Var}\left(\widehat{J}_{h,\tau}^{(1)}(\theta)\right) \leq \frac{1}{N}\mathbb{E}\left[\left(\widehat{J}_{h,\tau}^{(1)}(\theta)\right)^2\right]. \quad (22)$$

We want to approximate the right-hand side with the continuous-time form of the objective for simplicity. The error that we commit by using the continuous approximation

$$\widehat{J}_\tau^{(1)}(\theta; t_1) := \frac{1}{\tau}\int_{t_1}^{t_1+\tau} \|v_\theta(s, X_\omega^{(1)}(s))\|_2^2 \mathrm{d}s - \frac{2}{\tau}\int_{t_1}^{t_1+\tau} v_\theta(s, X_\omega^{(1)}(s)) \circ \mathrm{d}X_\omega(s)\,,$$

can be written as $\widehat{J}_{h,\tau}^{(1)}(\theta) = \widehat{J}_\tau^{(1)}(\theta; t_1) + E_h$, with mean squared approximation error of the trapezoidal rule $\mathbb{E}[E_h^2] \leq \tilde{C}h$ (strong convergence order $1/2$ in general [32]; here, again, $\tilde{C}$ depends on b, A, v_θ and derivative bounds; and we could also get an $O(h^2)$ bound e.g. for commutative noise). Since $(a+b)^2 \leq 2a^2+2b^2$, we have $\mathbb{E}\left[\left(\widehat{J}_{h,\tau}^{(1)}(\theta)\right)^2\right] \leq 2\mathbb{E}\left[\left(\widehat{J}_\tau^{(1)}(\theta; t_1)\right)^2\right] + 2\tilde{C}h$. Then it remains to estimate the second moment of the continuous objective (again with $(a-b)^2 \leq 2a^2+2b^2$):

$$\mathbb{E}\left[\left(\widehat{J}_\tau^{(1)}(\theta)\right)^2\right] \leq \underbrace{2\mathbb{E}_{t\sim\mathcal{U}([0,T-\tau]),X(t)\sim\rho(t)}\left[\left(\frac{1}{\tau}\int_t^{t+\tau}\|v_\theta(s,X(s))\|_2^2\mathrm{d}s\right)^2\right]}_{\leq 2V^4}$$
$$+ 8\mathbb{E}_{t\sim\mathcal{U}([0,T-\tau]),\omega\sim\mathbb{P}}\left[\left(\frac{1}{\tau}\int_t^{t+\tau} v_\theta(s,X_\omega(s))\circ\mathrm{d}X_\omega(s)\right)^2\right].$$

To bound the expectation of the squared Stratonovich integral, fix t in the following, i.e., use the law of total expectation. Convert the Stratonovich integral back to an Itô integral:

$$\frac{1}{\tau}\int_t^{t+\tau} v_\theta(s,X_\omega(s))\circ\mathrm{d}X_\omega(s) = \frac{1}{\tau}\int_t^{t+\tau}\langle v_\theta(s,X_\omega(s)), b(s,X_\omega(s))\rangle\mathrm{d}s$$
$$+\frac{1}{\tau}\int_t^{t+\tau}\langle v_\theta(s,X_\omega(s)), A(s)\mathrm{d}W_\omega(s)\rangle + \frac{1}{2\tau}\int_t^{t+\tau}\mathrm{Tr}\left[\Sigma(s)\nabla v_\theta(s,X_\omega(s))\right]\mathrm{d}s\,.$$

Using the inequality $(a+b+c)^2 \leq 3(a^2+b^2+c^2)$ (by Cauchy-Schwarz), we bound the expectations over ω of the individual terms squared:

1. Using the Cauchy-Schwarz inequality for integrals and the uniform bounds $\|b(t,x)\|_2 \leq b_{\text{max}}$ and $\|v_\theta(t,x)\|_2 \leq V$, we get

$$\mathbb{E}_\omega\left[\left(\frac{1}{\tau}\int_t^{t+\tau}\langle v_\theta(s,X_\omega(s)), b(s,X_\omega(s))\rangle\mathrm{d}s\right)^2\right] \leq V^2 b_{\text{max}}^2\,.$$

2. Using the Itô isometry and the bound $\Sigma(t) = A(t)A(t)^\top \preceq \sigma_{\text{max}} I_d$, we have

$$\mathbb{E}_\omega\left[\left(\frac{1}{\tau}\int_t^{t+\tau}\langle v_\theta(s,X_\omega(s)), A(s)\mathrm{d}W_\omega(s)\rangle\right)^2\right]$$
$$= \frac{1}{\tau^2}\mathbb{E}_\omega\left[\int_t^{t+\tau}\langle v_\theta(s,X_\omega(s)), \Sigma(s)v_\theta(s,X_\omega(s))\rangle\mathrm{d}s\right]$$
$$\leq \frac{1}{\tau^2}\mathbb{E}_\omega\left[\int_t^{t+\tau}\sigma_{\max}\|v_\theta(s,X_\omega(s))\|_2^2\mathrm{d}s\right] \leq \frac{\sigma_{\max}V^2}{\tau}\,.$$

3. Lastly, using the assumption $C_{\text{div}} = \sup_{t,x}\frac{1}{2}|\mathrm{Tr}\left[\Sigma(t)\nabla v_\theta(t,x)\right]| < \infty$, we have

$$\mathbb{E}_\omega\left[\left(\frac{1}{2\tau}\int_t^{t+\tau}\mathrm{Tr}\left[\Sigma(s)\nabla v_\theta(s,X_\omega(s))\right]\mathrm{d}s\right)^2\right] \leq C_{\text{div}}^2\,,$$

which completes the proof since all of these bounds are independent of $t \sim \mathcal{U}([0,T-\tau])$. □

C.3 Variance of the one-step FTM loss as $h \downarrow 0$, and relation to symmetric differences

In this section, we discuss the one-step FTM objective (16) from the main text. We first note that the loss $J_{\mathrm{FTM}}(\theta)$, prior to empiricalization, can be written as

$$\begin{aligned} J_{\mathrm{FTM}}(\theta) &= h^{-1}\mathbb{E}_{t\sim\mathcal{U}([h,T-h]),\omega\sim\mathbb{P}}\left[\|v_\theta(t,X_\omega(t))\|_2^2 h - \langle v_\theta(t,X_\omega(t)), X_\omega(t+h)-X_\omega(t-h)\rangle\right] \\ &= \mathbb{E}_{t\sim\mathcal{U}([h,T-h]),\omega\sim\mathbb{P}}\left[\|v_\theta(t,X_\omega(t))\|_2^2 - 2\langle Y_{h,\omega}(t), v_\theta(t,X_\omega(t))\rangle\right] , \end{aligned} \tag{23}$$

where we denote the symmetric finite difference of the SDE paths with step size h by

$$Y_{h,\omega}(t) := \frac{X_\omega(t+h)-X_\omega(t-h)}{2h} .$$

Indeed, while the one-step FTM objective (16) has been obtained from (13) in the main text in the limiting case $\tau = 2h$ and midpoint discretization of the Stratonovich integral, we can see from (23) that there is another, in some sense complementary way of viewing (16) based on the discussion in Section 3.1 of the main text: In (23), by the law of total expectation, we have

$$\mathbb{E}_{\omega\sim\mathbb{P}}\left[\langle Y_{h,\omega}(t), v_\theta(t,X_\omega(t))\rangle\right] = \mathbb{E}_{\omega\sim\mathbb{P}}[\langle \underbrace{\mathbb{E}\left[Y_{h,\omega}(t) \mid X_\omega(t)\right]}_{\overset{(6)}{=} v_h(t,X_\omega(t))}, v_\theta(t,X_\omega(t))\rangle] ,$$

i.e., we recognize here the approximation of the probability current velocity via symmetric differences as in (6). For $h \downarrow 0$, this term converges to $\mathbb{E}_{\omega\sim\mathbb{P}}\left[\langle v(t,X_\omega(t)), v_\theta(t,X_\omega(t))\rangle\right]$ by Section 3.1. In summary, as an alternative to the route to the FTM loss (16) via the Stratonovich integral identity (11) and subsequent discretization and $\tau = 2h$ as presented in the main text, we could hence have started directly from the intractable loss (10), approximated the true, unknown probability current velocity v by the symmetric difference v_h at finite $h > 0$, and then used the law of total expectation to arrive at (16). We chose to start from the Stratonovich integral identity (11) in the main text to closely connect the loss to the available literature in statistical physics and stochastic thermodynamics [24, 11, 41], and since chunking – which prevents the variance blowup as $h \downarrow 0$ at fixed N – is somewhat more natural in this setting. In hindsight, then, it is easy to realize that *averaging* the symmetric difference estimator over K_τ time steps is equivalent, up to negligible boundary terms for $K_\tau \gg 1$, to the chunked estimator (14).

Let us discuss the aforementioned variance blowup of the one-step FTM loss in more detail. While in principle, this is a special case of the "chunked" result in Proposition 1, up to the different choice of discretization rule, we prefer to present a separate analysis and further comments on it for clarity. We consider the discretized objective

$$J_h(\theta) = \mathbb{E}_{t\sim\mathcal{U}([h,T-h]),\omega\sim\mathbb{P}}\left[\|v_\theta(t,X_\omega(t))\|_2^2 - 2\langle Y_{h,\omega}(t), v_\theta(t,X_\omega(t))\rangle\right] ,$$

and the corresponding empirical estimator

$$\widehat{J}_h(\theta) = \frac{1}{N}\sum_{i=1}^{N} \|v_\theta(t_i, X^{(i)}(t_i))\|_2^2 - 2\langle Y_h^{(i)}(t_i), v_\theta(t_i, X^{(i)}(t_i))\rangle , \tag{24}$$

which provides a practically tractable objective to replace $J(\theta)$ from (10) in the main text. In the following, we will make the two assumptions:

1. For fixed $\theta \in \mathbb{R}^p$, the velocity field v_θ is bounded in the sense that there exists $V > 0$ such that
$$\|v_\theta(t,x)\|_2 \le V \quad \text{for all } t\in[0,T],\ x\in\mathbb{R}^d .$$
2. The symmetric increment (6) is first-order accurate in the sense
$$\|v_h(t,x) - v(t,x)\|_2 \le Ch , \tag{25}$$
for a constant $C > 0$ and all $t \in [h, T-h]$, $x \in \mathbb{R}^d$. We formulate this as an assumption here for brevity, but such a bound can be derived by using standard weak Taylor arguments for the forward and reverse process [32].

We have the following upper bound on the mean-squared error of (24) under these assumptions:

Proposition 6. *We have*

$$\mathbb{E}\left[(\widehat{J}_h(\theta) - J(\theta))^2\right] \leq 4C^2V^2h^2 + 6V^2 d\sigma_{\max}(Nh)^{-1} + \left(3V^4 + 6b_{\max}^2V^2\right)N^{-1}\,. \tag{26}$$

The first term on the right-hand side of (26) bounds the squared bias $(J_h(\theta) - J(\theta))^2 \lesssim h^2$, and the other terms bound the variance $\mathbb{E}\left[(\widehat{J}_h(\theta) - J_h(\theta))^2\right] \lesssim (Nh)^{-1}$ for small h, which diverges at fixed N as $h \downarrow 0$. This suggests that as the time step size h decreases, more and more samples N are necessary to keep the mean-squared error bounded. A possible choice is to balance the bias and variance by picking $h^2 \approx (Nh)^{-1}$, or $N \approx 1/h^3$ (ignoring constants). While Proposition 6 a priori only gives an upper bound on the mean-squared error, a matching lower bound $\gtrsim (Nh)^{-1}$ can be shown, e.g., when assuming uniform ellipticity $\Sigma(t) \succeq \sigma_{\min} I_d$ for all $t \in [0,T]$ and $\sigma_{\min} > 0$. We find it more instructive to instead present an explicit calculation of the mean-squared error of the symmetric finite difference objective for the special case of $(X(t))_{t\in[0,T]}$ a one-dimensional Brownian motion, which we formulate as a lemma.

Lemma 1. *Consider the $(d=1)$-dimensional SDE $\mathrm{d}X(t) = \mathrm{d}W(t)$, $X(0) \sim \mathcal{N}(0,1)$, on the time interval $t \in [0,1]$. Then we have the sharp asymptotic equivalence $\mathbb{E}\left[(\widehat{J}_h(\theta) - J(\theta))^2\right] \sim c(Nh)^{-1}$ as $h \downarrow 0$ for a constant $c > 0$ (explicitly given by $c = 2\int_0^1 \mathbb{E}_{Z\sim\mathcal{N}(0,1)}\left[v_\theta(t, \sqrt{1+t}Z)^2\right]\mathrm{d}t$ here).*

Intuitively, then, since at small h only the short-time behavior of the diffusion process matters also for more general SDEs driven by Brownian noise, this will remain true in higher dimensions and with nonlinear drift vector field and anisotropic diffusion matrix. Finally, as discussed further in the main text, we remark that constants do matter in practice, in the sense that the time step h in many applications may not actually be so small as to see the blow-up of the variance at fixed N as $h \downarrow 0$ of (24). In such cases, the objective (24) does provide a viable option for learning the probability current velocity.

Proof of Proposition 6. Decompose the difference $\widehat{J}_h(\theta) - J(\theta)$ as

$$\widehat{J}_h(\theta) - J(\theta) = (\widehat{J}_h(\theta) - J_h(\theta)) + (J_h(\theta) - J(\theta))\,. \tag{27}$$

The first term in (27) has mean zero because $\widehat{J}_h(\theta)$ is a Monte Carlo estimator of $J_h(\theta)$ with independent samples. We now use that $\mathbb{E}[(Z_1+Z_2)^2] = \mathbb{E}[Z_1^2] + 2\mathbb{E}[Z_1Z_2] + \mathbb{E}[Z_2^2]$ and identify Z_1 with the first term of (27) and Z_2 with the second term of (27). Notice that Z_2 (the second term of (27)) is deterministic so that

$$\mathbb{E}\left[\left(\widehat{J}_h(\theta) - J(\theta)\right)^2\right] = \mathbb{E}\left[(\widehat{J}_h(\theta) - J_h(\theta))^2\right] + (J_h(\theta) - J(\theta))^2\,. \tag{28}$$

Let us first consider the squared bias term $(J_h(\theta) - J(\theta))^2$ in (28). With the linearity of the expectation, we obtain

$$J_h(\theta) - J(\theta) = -2\mathbb{E}_{t\sim\mathcal{U}([h,T-h]),\omega\sim\mathbb{P}}\left[\langle Y_{h,\omega}(t) - v(t, X_\omega(t)), v_\theta(t, X_\omega(t))\rangle\right]\,.$$

With the law of total probability, we obtain

$$\begin{aligned}\mathbb{E}\left[\langle Y_{h,\omega}(t) - v(t, X_\omega(t)), v_\theta(t, X_\omega(t))\rangle\right] =& \mathbb{E}\left[\mathbb{E}\left[\langle Y_{h,\omega}(t) - v(t, X_\omega(t)), v_\theta(t, X_\omega(t))\rangle\,|X_\omega(t)\right]\right]\\ =& \mathbb{E}\left[\langle \mathbb{E}\left[Y_{h,\omega}(t) - v(t, X_\omega(t))|X_\omega(t)\right], v_\theta(t, X_\omega(t))\rangle\right]\,.\end{aligned}$$

Now use (25) and the boundedness of v_θ to obtain with the Cauchy-Schwarz inequality

$$|J_h(\theta) - J(\theta)| \leq 2\mathbb{E}\left[\|\mathbb{E}[Y_{h,\omega}(t) - v(t, X_\omega(t))|X_\omega(t)]\|_2 \|v_\theta(t, X_\omega(t))\|_2\right] \leq 2ChV\,.$$

Let us now bound the first term of (28). Because $\widehat{J}_h(\theta)$ is an unbiased estimator of $J_h(\theta)$ and all samples in the estimator are independent and identically distributed, we obtain

$$\mathbb{E}\left[\left(\widehat{J}_h(\theta) - J_h(\theta)\right)^2\right] = \frac{1}{N}\operatorname{Var}_{t\in\mathcal{U}([h,T-h]),\omega\sim\mathbb{P}}\left[\|v_\theta(t, X_\omega(t))\|_2^2 - 2\langle Y_{h,\omega}(t), v_\theta(t, X_\omega(t))\rangle\right]\,.$$

It will be convenient to decompose the random variable

$$\begin{aligned}&\|v_\theta(t, X_\omega(t))\|_2^2 - 2\langle Y_{h,\omega}(t), v_\theta(t, X_\omega(t))\rangle\\ =&\|v_\theta(t, X_\omega(t))\|_2^2 - 2\langle Dh, \omega(t), v_\theta(t, X_\omega(t))\rangle - 2\langle \eta_{h,\omega}(t), v_\theta(t, X_\omega(t))\rangle\,,\end{aligned} \tag{29}$$

where we decomposed $Y_{h,\omega}(t)$ into

$$Y_{h,\omega}(t) = D_{h,\omega}(t) + \eta_{h,\omega}(t)\,, \quad \text{with} \begin{cases} D_{h,\omega}(t) & = \frac{1}{2h}\int_{t-h}^{t+h} b(s, X_\omega(s))\mathrm{d}s\,, \\ \eta_{h,\omega}(t) & = \frac{1}{2h}\int_{t-h}^{t+h} A(s)\mathrm{d}W_\omega(s)\,. \end{cases}$$

Consider the term $\langle \eta_{h,\omega}(t), v_\theta(t, X_\omega(t))\rangle$ in (29) first. For a fixed $h > 0$, this term is square integrable because the Cauchy-Schwarz inequality gives

$$|\langle \eta_{h,\omega}(t), v_\theta(t, X_\omega(t))\rangle|^2 \leq \|\eta_{h,\omega}(t)\|^2 \|v_\theta(t, X_\omega(t))\|^2\,,$$

and with the boundedness of v_θ and diffusion matrix $\Sigma(t) \preceq \sigma_{\max} I_d$, we obtain via Itô isometry:

$$\mathbb{E}[|\langle \eta_{h,\omega}(t), v_\theta(t, X_\omega(t))\rangle|^2] \leq V^2 \frac{d\sigma_{\max}}{h} < \infty\,.$$

Correspondingly, we bound the variance as

$$\mathrm{Var}[-2\langle \eta_{h,\omega}(t), v_\theta(t, X_\omega(t))\rangle] \leq \mathbb{E}[|2\langle \eta_{h,\omega}(t), v_\theta(t, X_\omega(t))\rangle|^2] \leq 4V^2 \frac{d\sigma_{\max}}{2h}\,. \tag{30}$$

Let us now consider the term $-2\langle D_{h,\omega}(t), v_\theta(t, X_\omega(t))\rangle$ of (29), and obtain

$$|-2\langle D_{h,\omega}(t), v_\theta(t, X_\omega(t))\rangle| \leq 2\|D_{h,\omega}(t)\|_2 \|v_\theta(t, X_\omega(t))\|_2 \leq 2b_{\max} V\,,$$

where we used that the drift term b is upper bound by $b_{\max}$ and v_θ by V. The variance is then

$$\mathrm{Var}\left[-2\langle D_{h,\omega}(t), v_\theta(t, X_\omega(t))\rangle\right] \leq \mathbb{E}\left[|-2\langle D_{h,\omega}(t), v_\theta(t, X_\omega(t))\rangle|^2\right] \leq 4b_{\max}^2 V^2\,. \tag{31}$$

Analogously, we bound

$$\mathrm{Var}\left[\|v_\theta(t, X_\omega(t))\|_2^2\right] \leq \mathbb{E}\left[\|v_\theta(t, X_\omega(t))\|_2^4\right] \leq V^4\,. \tag{32}$$

Combining (30), (31), and (32) with $\mathrm{Var}[Z_1 + Z_2 + Z_3] \leq 3(\mathrm{Var}[Z_1] + \mathrm{Var}[Z_2] + \mathrm{Var}[Z_3])$ leads to (26). □

Proof of Lemma 1. It suffices to consider the variance $\mathbb{E}\left[(\widehat{J}_h(\theta) - J_h(\theta))^2\right]$, and, by independence of the samples in (24) and the law of total variance to condition on the time $t \sim \mathcal{U}([h, 1-h])$, to compute $\mathrm{Var}\,\widehat{J}_h(\theta; t)$ for

$$\widehat{J}_h(\theta; t) := v_\theta(t, X_\omega(t))^2 - 2v_\theta(t, X_\omega(t))Y_{h,\omega}(t) \tag{33}$$

at a fixed time $t \in [h, 1-h]$. First, we characterize the joint distribution of $(X_\omega(t), Y_{h,\omega}(t))$. Clearly $X_\omega(t) \sim \mathcal{N}(0, 1+t)$, and

$$\begin{aligned} Y_{h,\omega}(t) &= \frac{X_\omega(t+h) - X_\omega(t-h)}{2h} = \frac{(X_\omega(t+h) - X_\omega(t)) + (X_\omega(t) - X_\omega(t-h))}{2h} \\ &= \frac{\Delta B_\omega(t) + \Delta B_\omega(t-h)}{2h} \sim \mathcal{N}\left(0, \frac{1}{2h}\right) \end{aligned}$$

marginally. Their correlation is then

$$\mathbb{E}\left[X_\omega(t) Y_{h,\omega}(t)\right] = \frac{1}{2h}\mathbb{E}\left[(X_\omega(t-h) + \Delta B_\omega(t-h))\,(\Delta B_\omega(t) + \Delta B_\omega(t-h))\right] = \frac{1}{2}$$

such that in total

$$\begin{pmatrix} X_\omega(t) \\ Y_{h,\omega}(t) \end{pmatrix} \sim \mathcal{N}\left(0, \begin{pmatrix} 1+t & 1/2 \\ 1/2 & 1/(2h) \end{pmatrix}\right)\,.$$

We can hence also write

$$\begin{pmatrix} X_\omega(t) \\ Y_{h,\omega}(t) \end{pmatrix} \overset{d}{=} \begin{pmatrix} \sqrt{1+t} Z_1 \\ \frac{1}{2\sqrt{1+t}} Z_1 + \sqrt{\frac{1}{2h} - \frac{1}{4(1+t)}} Z_2 \end{pmatrix} \tag{34}$$

for Z_1, Z_2 iid $\mathcal{N}(0,1)$. The other ingredient we need is that for any $k \geq 0$, by the law of total probability, we have

$$\mathbb{E}\left[v_\theta(t, X_\omega(t))^k Y_{h,\omega}(t)\right] = \mathbb{E}\left[v_\theta(t, X_\omega(t))^k \mathbb{E}\left[Y_{h,\omega}(t) \mid X_\omega(t)\right]\right] = \mathbb{E}\left[v_\theta(t, X_\omega(t))^k v_h(X_\omega(t), t)\right] \tag{35}$$

where $\lim_{h\to 0} v_h(t,x) = v(t,x)$ is the true probability current velocity of the process X. In the present example

$$v(t,x) = b(t,x) - \frac{1}{2}\Sigma(t)\nabla \log \rho(t,x) = \frac{x}{2(1+t)}\,.$$

With that, we can now compute

$$\begin{aligned}\operatorname{Var}\widehat{J}_h(\theta;t) = \underbrace{\operatorname{Var}\left(v_\theta(t,X_\omega(t))^2\right)}_{=:V_1} + 4\underbrace{\operatorname{Var}\left(v_\theta(t,X_\omega(t))Y_{h,\omega}(t)\right)}_{=:V_2} \\ -4\underbrace{\operatorname{Cov}\left(v_\theta(t,X_\omega(t))^2, v_\theta(t,X_\omega(t))Y_{h,\omega}(t)\right)}_{=:C_{12}}\,.\end{aligned}$$

The first term V_1 is a constant in h. The last term is

$$\begin{aligned}C_{12} &= \mathbb{E}\left[v_\theta(t,X_\omega(t))^3 Y_{h,\omega}(t)\right] - \mathbb{E}\left[v_\theta(t,X_\omega(t))^2\right]\mathbb{E}\left[v_\theta(t,X_\omega(t))Y_{h,\omega}(t)\right] \\ &\overset{(35)}{=} \mathbb{E}\left[v_\theta(t,X_\omega(t))^3 v_h(t,X_\omega(t))\right] - \mathbb{E}\left[v_\theta(t,X_\omega(t))^2\right]\mathbb{E}\left[v_\theta(t,X_\omega(t))v_h(t,X_\omega(t))\right]\,,\end{aligned}$$

which evidently converges to a well-defined limiting value as $h \to 0$. Lastly, for V_2, we find

$$\begin{aligned}V_2 &= \mathbb{E}\left[v_\theta(t,X_\omega(t))^2\left(Y_{h,\omega}(t)\right)^2\right] - \mathbb{E}\left[v_\theta(t,X_\omega(t))Y_{h,\omega}(t)\right]^2 \\ &\overset{(34),(35)}{=} \frac{1}{4(1+t)}\mathbb{E}\left[v_\theta(t,\sqrt{1+t}Z_1)^2 Z_1^2\right] + \left(\frac{1}{2h} - \frac{1}{4(1+t)}\right)\mathbb{E}\left[v_\theta(t,\sqrt{1+t}Z_1)^2\right] \\ &\quad - \mathbb{E}\left[v_\theta(t,X_\omega(t))v_h(t,X_\omega(t))\right]^2\,.\end{aligned}$$

Overall, because of the term $1/(2h)$ term, we hence have shown

$$\operatorname{Var}\widehat{J}_h(\theta;t) \overset{h\to 0}{\sim} \frac{c(t)}{h}\,,$$

where the constant $c(t) > 0$ is explicitly given by $c(t) = 2\mathbb{E}\left[v_\theta(t,\sqrt{1+t}Z)^2\right]$ for $Z \sim \mathcal{N}(0,1)$. $\square$

C.4 Proof of Proposition 2

We denote in this section by $(X(t))_{t\in[0,T]}$ sample paths of the SDE (1), by $(\hat{x}_\theta(t))_{t\in[0,T]}$ the trajectories of the ODE (3) with (learned) velocity field $u = v_\theta$, and by $(\hat{x}(t))_{t\in[0,T]}$ the trajectories of the ODE (3) with $u = v$ the true probability current velocity (7). We couple the distributions of these by picking *the same starting point* $X(0) = \hat{x}_\theta(0) = \hat{x}(0) \sim \rho(0)$, state pathwise errors in Lemma 2 and Lemma 3 via Grönwall's lemma, and conclude with the Wasserstein-2 bound of Proposition 2 as a corollary. As in the main text in Proposition 2, we suppose here that the probability current velocity v is uniformly Lipschitz on $\mathbb{R}^d$ for all times with Lipschitz constant $L_v > 0$. Then we have:

Lemma 2. $\mathbb{E}\left[\|\hat{x}(t) - X(t)\|_2^2\right] \le \exp((2L_v+1)t)\int_0^t \left(\operatorname{Tr}\Sigma(s) + \mathbb{E}\left[\|\frac{1}{2}\Sigma(s)\nabla\log\rho(s,X(s))\|_2^2\right]\right)\mathrm{d}s.$

Lemma 3. $\mathbb{E}\left[\|\hat{x}_\theta(t) - \hat{x}(t)\|_2^2\right] \le \exp((2L_v+1)t)\int_0^t \mathbb{E}\left[\|v_\theta(s,\hat{x}_\theta(s)) - v(s,\hat{x}_\theta(s))\|_2^2\right]\mathrm{d}s.$

We remark that combining these two lemmas using the triangle inequality also provides an upper bound on the mean squared distance of $X(t)$ and $\hat{x}_\theta(t)$ if desired, though we come back to this problem in more detail in Proposition 3 and Appendix C.5. Now, since the time-marginal distributions $\rho(t)$ generated by $(\hat{x}(t))_{t\in[0,T]}$ are the true ones, and we have constructed a particular coupling of $\rho(t)$ and $\hat{\rho}_\theta(t)$ via the two ODEs with the same initial conditions, Lemma 3 directly implies Proposition 2:

$$W_2^2(\rho(t),\hat{\rho}_\theta(t)) = \inf_{\gamma\in\Gamma(\rho(t),\hat{\rho}_\theta(t))}\mathbb{E}_{(x,y)\sim\gamma}\left[\|x-y\|_2^2\right] \le \mathbb{E}\left[\|\hat{x}_\theta(t) - \hat{x}(t)\|_2^2\right]\,,$$

where $\Gamma(\rho(t),\hat{\rho}_\theta(t))$ denotes the set of all couplings of $\rho(t)$ and $\hat{\rho}_\theta(t)$. Finally, then, as mentioned in the main text, if $\phi\colon \mathbb{R}^d \to \mathbb{R}$ is a scalar QoI which is Lipschitz with constant L_ϕ, then we obtain by Kantorovich–Rubinstein duality for the W_1 metric, and $W_1 \le W_2$:

$$|\mathbb{E}_{X(t)\sim\rho(t)}[\phi(X(t))] - \mathbb{E}_{\hat{x}_\theta(t)\sim\hat{\rho}_\theta(t)}[\phi(\hat{x}_\theta(t))]| \le L_\phi\, W_1(\rho(t),\hat{\rho}_\theta(t)) \le L_\phi\, W_2(\rho(t),\hat{\rho}_\theta(t))\,,$$

such that Proposition 2 also provides control over the error for time-local QoIs.

Proof of Lemma 2. Denote the difference of the ODE and SDE trajectory by $E(t) := \hat{x}(t) - X(t)$. By construction we have $E(0) = 0$, and E satisfies the SDE

$$\begin{aligned}\mathrm{d}E(t) &= v(t, \hat{x}(t))\mathrm{d}t - b(t, X(t))\mathrm{d}t - A(t)\mathrm{d}W(t) \\ &\overset{(7)}{=} \left[v(t, \hat{x}(t)) - v(t, X(t)) - q(t)\right]\mathrm{d}t - A(t)\mathrm{d}W(t)\,,\end{aligned}$$

where we abbreviate $q(t) := \frac{1}{2}\Sigma(t)\nabla \log \rho(t, X(t))$. By Itô's lemma, we then have

$$\mathrm{d}\|E(t)\|_2^2 = \left(2\langle E(t), v(t, \hat{x}(t)) - v(t, X(t)) - q(t)\rangle + \operatorname{Tr}\Sigma(t)\right)\mathrm{d}t - 2\langle E(t), A(t)\mathrm{d}W(t)\rangle\,,$$

and by taking expectations:

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbb{E}\left[\|E(t)\|_2^2\right] = 2\mathbb{E}\left[\langle E(t), v(t, \hat{x}(t)) - v(t, X(t))\rangle\right] + 2\mathbb{E}\left[\langle E(t), q(t)\rangle\right] + \operatorname{Tr}\Sigma(t)\,.$$

Using the Cauchy–Schwarz inequality and Lipschitz property of v for the first term on the right-hand side and Young's inequality for the second term, we obtain

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbb{E}\left[\|E(t)\|_2^2\right] \le (2L_v + 1)\left[\|E(t)\|_2^2\right] + \mathbb{E}\left[\|q(t)\|_2^2\right] + \operatorname{Tr}\Sigma(t)\,,$$

such that Grönwall's lemma, noting that $\mathbb{E}\left[\|E(0)\|_2^2\right] = 0$, finishes the proof. □

Proof of Lemma 3. Define $e(t) = \hat{x}_\theta(t) - \hat{x}(t)$ with $\frac{\mathrm{d}}{\mathrm{d}t}e(t) = v_\theta(t, \hat{x}_\theta(t)) - v(t, \hat{x}(t)) = (v_\theta(t, \hat{x}_\theta(t)) - v(t, \hat{x}_\theta(t))) + (v(t, \hat{x}_\theta(t)) - v(t, \hat{x}(t)))$ and proceed analogously to the previous proof. □

C.5 Proof of Proposition 3

For the reader's convenience, we state again all necessary assumptions for Proposition 3 in the following. Let $\rho(t)$ denote the law of the SDE (1), and let $\hat{\rho}_\theta(t)$ denote the law generated by the learned ODE

$$\frac{\mathrm{d}}{\mathrm{d}t}\hat{x}_\theta(t) = v_\theta(t, \hat{x}_\theta(t)), \qquad \hat{x}_\theta(0) \sim \rho(0).$$

Assume that the probability current velocity v is uniformly Lipschitz in x with constant $L_v > 0$, and that we have pointwise error control

$$\|v_\theta(t, x) - v(t, x)\|_2 \le \epsilon \qquad \text{for all } t \in [0, T],\ x \in \mathbb{R}^d.$$

Let $\phi : [0, T] \times \mathbb{R}^d \to \mathbb{R}$ be a bounded and Lipschitz vector field:

$$\|\phi(t, x)\|_2 \le B_\phi, \qquad \|\phi(t, x) - \phi(t, y)\|_2 \le L_\phi\|x - y\|_2 \qquad \text{for all } t \in [0, T],\ x, y \in \mathbb{R}^d\,.$$

Assume similarly that v_θ is uniformly bounded and Lipschitz in x:

$$\|v_\theta(t, x)\|_2 \le B_{v_\theta}, \qquad \|v_\theta(t, x) - v_\theta(t, y)\|_2 \le L_{v_\theta}\|x - y\|_2 \qquad \text{for all } t \in [0, T],\ x, y \in \mathbb{R}^d\,.$$

Set $\tilde{C} := L_\phi B_{v_\theta} + B_\phi L_{v_\theta}$. We will show below that the error in the path-dependent QoI $\Phi(X) := \int_0^T \phi(t, X(t)) \circ \mathrm{d}X(t)$ under the true SDE and the learned ODE rollout satisfies

$$|\mathbb{E}[\Phi(X)] - \mathbb{E}[\Phi(\hat{x}_\theta)]| \le \int_0^T \left(\|v(t, \cdot) - v_\theta(t, \cdot)\|_{L^2(\rho(t))}\|\phi(t, \cdot)\|_{L^2(\rho(t))} + \tilde{C}W_2(\rho(t), \hat{\rho}_\theta(t))\right)\mathrm{d}t\,. \tag{36}$$

Using Proposition 2, this estimate directly implies Proposition 3. We also mention here (cf. (37) in the proof below) that the local transport term, i.e., the first term on the right-hand side of (36), can alternatively be controlled directly by the FTM excess risk

$$J(\theta) - J^* = \frac{1}{T}\int_0^T \|v_\theta(t, \cdot) - v(t, \cdot)\|^2_{L^2(\rho(t))}\,\mathrm{d}t\,,$$

where

$$\begin{cases} J(\theta) &= \frac{1}{T}\int_0^T \int_{\mathbb{R}^d} \left(\|v_\theta(t, x)\|_2^2 - 2\langle v(t, x), v_\theta(t, x)\rangle\right)\rho(t, x)\,\mathrm{d}x\,\mathrm{d}t\,, \\ J^* &:= \frac{1}{T}\int_0^T \int_{\mathbb{R}^d} \left(\|v(t, x)\|_2^2 - 2\|v(t, x)\|_2^2\right)\rho(t, x)\,\mathrm{d}x\,\mathrm{d}t \\ &= -\frac{1}{T}\int_0^T \int_{\mathbb{R}^d} \|v(t, x)\|_2^2\rho(t, x)\,\mathrm{d}x\,\mathrm{d}t\,, \end{cases}$$

since by using Cauchy–Schwarz twice:

$$\frac{1}{T}\int_0^T \left|\mathbb{E}_{X(t)\sim\rho(t)}\left[\langle\phi(t,X(t)), v(t,X(t)) - v_\theta(t,X(t))\rangle\right]\right| \mathrm{d}t$$
$$\leq (J(\theta) - J^*)^{1/2} \left(\frac{1}{T}\int_0^T \|\phi(t,\cdot)\|^2_{L^2(\rho(t))}\,\mathrm{d}t\right)^{1/2}.$$

Proof of Proposition 3. We first use the Stratonovich integral identity (11) for the SDE sample paths to simplify the difference of the path-dependent QoI expectations:

$$\begin{aligned}
|\mathbb{E}[\Phi(X)] - \mathbb{E}[\Phi(\hat{x}_\theta)]| &= \left|\mathbb{E}\left[\int_0^T \phi(t,X(t)) \circ \mathrm{d}X(t) - \int_0^T \left\langle \phi(t,\hat{x}_\theta(t)), \tfrac{\mathrm{d}}{\mathrm{d}t}\hat{x}_\theta(t)\right\rangle \mathrm{d}t\right]\right| \\
&= \left|\mathbb{E}\left[\int_0^T \Big(\langle\phi(t,X(t)), v(t,X(t))\rangle - \langle\phi(t,\hat{x}_\theta(t)), v_\theta(t,\hat{x}_\theta(t))\rangle\Big)\mathrm{d}t\right]\right| \\
&\leq \int_0^T \left|\mathbb{E}\left[\langle\phi(t,X(t)), v(t,X(t))\rangle - \langle\phi(t,\hat{x}_\theta(t)), v_\theta(t,\hat{x}_\theta(t))\rangle\right]\right| \mathrm{d}t
\end{aligned}$$

Adding and subtracting $\int_{\mathbb{R}^d}\langle\phi(t,x), v_\theta(t,x)\rangle\rho(t,x)\,\mathrm{d}x$, we obtain

$$\begin{aligned}
|\mathbb{E}[\Phi(X)] - \mathbb{E}[\Phi(\hat{x}_\theta)]| \leq \int_0^T \Bigg(&\left|\int_{\mathbb{R}^d}\langle\phi(t,x), v(t,x) - v_\theta(t,x)\rangle\rho(t,x)\,\mathrm{d}x\right| \\
&+ \left|\int_{\mathbb{R}^d}\langle\phi(t,x), v_\theta(t,x)\rangle\big(\rho(t,x) - \hat{\rho}_\theta(t,x)\big)\,\mathrm{d}x\right|\Bigg)\mathrm{d}t. \qquad (37)
\end{aligned}$$

We bound the two terms separately. For the first term in (37), Cauchy–Schwarz gives

$$\begin{aligned}
&\left|\int_{\mathbb{R}^d}\langle\phi(t,x), v(t,x) - v_\theta(t,x)\rangle\rho(t,x)\,\mathrm{d}x\right| \\
&\leq \left(\int_{\mathbb{R}^d}\|v(t,x) - v_\theta(t,x)\|_2^2\rho(t,x)\,\mathrm{d}x\right)^{1/2}\left(\int_{\mathbb{R}^d}\|\phi(t,x)\|_2^2\rho(t,x)\,\mathrm{d}x\right)^{1/2} \\
&= \|v(t,\cdot) - v_\theta(t,\cdot)\|_{L^2(\rho(t))}\|\phi(t,\cdot)\|_{L^2(\rho(t))}.
\end{aligned}$$

For the second term in (37), define, for fixed $t\in[0,T]$, $f_t(x) := \langle\phi(t,x), v_\theta(t,x)\rangle$. We show that f_t is Lipschitz with $\mathrm{Lip}(f_t)\leq\tilde{C}$. For any $x, y\in\mathbb{R}^d$:

$$\begin{aligned}
|f_t(x) - f_t(y)| &= |\langle\phi(t,x), v_\theta(t,x)\rangle - \langle\phi(t,y), v_\theta(t,y)\rangle| \\
&\leq |\langle\phi(t,x) - \phi(t,y), v_\theta(t,x)\rangle| + |\langle\phi(t,y), v_\theta(t,x) - v_\theta(t,y)\rangle| \\
&\leq \|\phi(t,x) - \phi(t,y)\|_2\|v_\theta(t,x)\|_2 + \|\phi(t,y)\|_2\|v_\theta(t,x) - v_\theta(t,y)\|_2 \\
&\leq L_\phi\|x-y\|_2 B_{v_\theta} + B_\phi L_{v_\theta}\|x-y\|_2 \\
&= \tilde{C}\|x-y\|_2.
\end{aligned}$$

Then, by Kantorovich–Rubinstein duality and $W_1\leq W_2$,

$$\begin{aligned}
\left|\int_{\mathbb{R}^d}\langle\phi(t,x), v_\theta(t,x)\rangle\big(\rho(t,x) - \hat{\rho}(t,x)\big)\,\mathrm{d}x\right| &= \left|\int_{\mathbb{R}^d} f_t(x)\rho(t,x)\,\mathrm{d}x - \int_{\mathbb{R}^d} f_t(x)\hat{\rho}(t,x)\,\mathrm{d}x\right| \\
&\leq \mathrm{Lip}(f_t)W_1(\rho(t),\hat{\rho}(t)) \leq \tilde{C}W_2(\rho(t),\hat{\rho}(t)).
\end{aligned}$$

This finishes the proof of (36) from (37).

□

D Details about experiments

We provide the details for the four benchmark systems used in Section 4. In all examples, FTM is trained from trajectory data using the one-step loss (16), except otherwise noted (in particular in Figure 2). At test time, we generate ensembles by integrating the learned ODE.

Trajectory-dependent probability-current QoIs We evaluate whether the learned model recovers trajectory-dependent probability currents. Given a smooth test function

$$\phi : [0, T] \times \mathbb{R}^d \to \mathbb{R}^d,$$

we define the Stratonovich path functional

$$Q_\phi = \mathbb{E}\left[\int_0^T \phi(t, X(t)) \circ \mathrm{d}X(t)\right]. \tag{38}$$

For discrete trajectories $\{X^{(i)}(t_k)\}_{i=1,k=0}^{N,K}$, we estimate (38) using the midpoint Stratonovich rule

$$\widehat{Q}_\phi = \frac{1}{N}\sum_{i=1}^{N}\sum_{k=0}^{K-1} \phi\left(\frac{t_k + t_{k+1}}{2}, \frac{X^{(i)}(t_k) + X^{(i)}(t_{k+1})}{2}\right)^\top \left(X^{(i)}(t_{k+1}) - X^{(i)}(t_k)\right). \tag{39}$$

The same estimator is applied to trajectories generated by each learned model, and the reported trajectory-QoI error is the absolute difference from the reference value computed on ground-truth trajectories.

D.1 Duffing oscillator

Dynamics We consider the stochastically forced Duffing oscillator

$$\mathrm{d}X_1(t) = X_2(t)\,\mathrm{d}t, \tag{40}$$

$$\mathrm{d}X_2(t) = \left(-2\xi\omega X_2(t) + \omega^2 X_1(t) - \omega^2\gamma X_1(t)^3\right)\mathrm{d}t + \sigma\,\mathrm{d}W(t), \tag{41}$$

with parameters

$$\xi = 0.2, \qquad \gamma = 0.2, \qquad \omega = 1, \qquad \sigma = 0.5.$$

Equivalently, the deterministic force corresponds to a double-well potential of the form

$$V(x_1) = -\frac{1}{2}x_1^2 + \frac{1}{20}x_1^4,$$

up to the parameter scaling above. The random forcing acts only on the acceleration equation. Initial conditions are sampled as

$$X(0) \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ -10 \end{bmatrix}, I_2\right).$$

Probability-current QoI The Duffing oscillator has two metastable wells separated by the barrier near $x_1 = 0$. To measure the net probability current across this barrier, we use the test function

$$\phi(x) = \begin{bmatrix} \dfrac{1}{\sqrt{2\pi}\epsilon}\exp\left(-\dfrac{x_1^2}{2\epsilon^2}\right) \\ 0 \end{bmatrix}, \qquad \epsilon = 1. \tag{42}$$

This localizes the Stratonovich line integral near the separating surface $x_1 = 0$ and therefore measures the signed probability flux moving between the two wells. We estimate Q_ϕ using (39).

Training data generation We generate training trajectories by integrating the Duffing Oscillator SDE with the Euler–Maruyama scheme on a uniform time grid with step size $\Delta t = 0.01$ over the time interval $[0, 12]$. Initial conditions are drawn from the distribution specified above, and we generate 5000 training trajectories. For testing, we draw a new set of initial conditions from the same initial distribution and generate 5000 testing trajectories.

Reported metrics For the Duffing oscillator, we report both a distributional error between time marginals and the trajectory-dependent current error associated with (42). The distributional error is computed using a sliced Wasserstein-2 distance between generated and reference ensembles, averaged over the evaluated time points.

D.2 Stochastic Rayleigh–Bénard convection

Dynamics We use the nine-dimensional chaotic Rayleigh–Bénard convection model of [51]. The state is

$$X(t) = (X_1(t), \dots, X_9(t)) \in \mathbb{R}^9,$$

and the dynamics are given by

$$\mathrm{d}X(t) = f_\mu(X(t))\,\mathrm{d}t + \sigma\,\mathrm{d}W(t), \tag{43}$$

where f_μ is the deterministic nine-mode convection vector field and μ is the Rayleigh-type control parameter. In the experiments we use the chaotic regime with

$$\sigma = \frac{1}{20},$$

and treat μ as a control parameter that is varied across training and testing.

Rotational probability-current QoI The Rayleigh–Bénard dynamics exhibit persistent rotational motion in low-dimensional projections. To test whether a learned model reproduces this rotational current, we consider the projection

$$P : \mathbb{R}^9 \to \mathbb{R}^2, \qquad Px = (x_1, x_2),$$

and the counterclockwise rotation matrix

$$R = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}.$$

We define

$$\phi(x) = P^\top R P x = (-x_2, x_1, 0, \dots, 0)^\top. \tag{44}$$

The corresponding quantity Q_ϕ measures the component of the probability current aligned with rotation in the (x_1, x_2) plane. As in the Duffing example, we estimate it by the discrete Stratonovich estimator (39).

Training data generation We generate training trajectories by integrating the Rayleigh–Bénard SDE (43) with the Euler–Maruyama scheme on a uniform time grid with step size $\Delta t = 0.01$ over the time interval $[0, 20]$. Initial conditions are drawn from an isotropic Gaussian with mean 0 and standard deviation 0.02 in each coordinate. For each training parameter value

$$\mu \in \{13.5,\ 13.6,\ 13.7,\ 13.8,\ 13.9,\ 14.0,\ 14.1,\ 14.2\},$$

we generate $20{,}000$ training trajectories, and we evaluate the learned model at the held-out test value $\mu = 13.65$. For testing, we draw a new set of initial conditions from the same Gaussian distribution and generate $20{,}000$ testing trajectories at $\mu = 13.65$.

Reported metrics We report a distributional error between generated and reference time marginals, together with the rotational-current error induced by (44). The latter distinguishes methods that reproduce the correct projected distribution from those that also reproduce the trajectory-induced circulation.

D.3 Stochastically forced Burgers turbulence

Dynamics We consider the one-dimensional stochastic Burgers equation on the periodic domain $[0, 1)$,

$$\mathrm{d}u(t,x) = \left(\nu\,\partial_{xx}u(t,x) - u(t,x)\,\partial_x u(t,x)\right)\mathrm{d}t + \sigma\,\mathrm{d}W(t,x), \qquad x \in [0,1), \tag{45}$$

with viscosity

$$\nu = 0.007.$$

The stochastic forcing is white in time and colored in space. Specifically, we use a truncated Fourier expansion on the first ten modes,

$$\mathrm{d}W(t,x) = \sum_{\iota=1}^{10} \frac{1}{\iota}\left(a_\iota(x)\,\mathrm{d}B_\iota^{(1)}(t) + b_\iota(x)\,\mathrm{d}B_\iota^{(2)}(t)\right),$$

where a_ι and b_ι denote the corresponding sine and cosine modes, and the Brownian motions are independent. Unless otherwise stated, the noise amplitude is

$$\sigma = \frac{2}{50}.$$

Initial conditions Initial conditions are Gaussian bumps with additive colored perturbations,

$$u(0,x) = \exp\bigl(-c(x-1/2)^2\bigr) + \eta(x), \tag{46}$$

where η is drawn from the same spatially colored Fourier family as the forcing, with perturbation strength $\sigma_0 = 0.015$. The constant c controls the width of the initial bump.

Training data generation We solve (45) using a method-of-lines discretization with second-order centered finite differences in space on a uniform grid of $N = 64$ points over the periodic domain $[0, 1)$, combined with an Euler–Maruyama scheme in time with step size $\Delta t = 5 \times 10^{-4}$ over the interval $[0, 4]$. Trajectories are subsampled to $T = 800$ uniformly spaced output times (output spacing $\Delta t_{\text{out}} = 5 \times 10^{-3}$). Initial conditions are drawn according to (46) with bump width $c = 20$. We generate $4{,}096$ training trajectories and $4{,}096$ testing trajectories, with testing initial conditions drawn independently from the same distribution.

Physical diagnostics For Burgers, the main evaluation metrics are the energy and enstrophy of the field,

$$E(t) = \frac{1}{2}\int_0^1 |u(t,x)|^2 \,\mathrm{d}x, \tag{47}$$

$$Z(t) = \frac{1}{2}\int_0^1 |\partial_x u(t,x)|^2 \,\mathrm{d}x. \tag{48}$$

On an equidistant grid of 64 grid points, these quantities are computed by the corresponding trapezoidal or periodic finite-difference approximations. We compare the ensemble mean and ensemble standard deviation of $E(t)$ and $Z(t)$ over generated and reference trajectories.

Noise-sensitivity experiment To test sensitivity to stochastic forcing, we repeat the Burgers experiment for multiple values of the forcing amplitude σ. This comparison is used to verify that the benchmark contains genuine stochastic variability: standard deterministic time-steppers trained with mean-squared error tend to collapse toward conditional-mean dynamics as σ increases, whereas FTM is evaluated by its ability to maintain accurate ensemble statistics.

D.4 Stochastically forced turbulence (Navier-Stokes)

Dynamics and numerical solver We consider two-dimensional incompressible Navier–Stokes flow on the periodic domain

$$\Omega = [0, 2\pi]^2.$$

The simulations are based on the spectral forced-turbulence example from JAX-CFD [33, 22], modified to include additive stochastic forcing in place of deterministic Kolmogorov forcing. We use the vorticity formulation

$$\mathrm{d}\omega + (u \cdot \nabla\omega)\,\mathrm{d}t = (\nu\,\Delta\omega - \alpha\,\omega)\,\mathrm{d}t + f_{\text{sto}}(t,x)\,\mathrm{d}t, \qquad u = \nabla^{\perp}\Delta^{-1}\omega, \tag{49}$$

with viscosity $\nu = 10^{-3}$ and linear drag coefficient $\alpha = 0.1$.

Stochastic coarse-mode forcing The driving stochastic forcing acts on a finite set of low-frequency Fourier modes. In vorticity form, the stochastic forcing is

$$f_{\text{sto}}(t,x)\,\mathrm{d}t = \sigma \sum_{\kappa \in \mathcal{K}_{\text{sto}}} w_\kappa \left(a_\kappa(x)\,\mathrm{d}B_\kappa^{(1)}(t) + b_\kappa(x)\,\mathrm{d}B_\kappa^{(2)}(t) \right), \tag{50}$$

where $\mathcal{K}_{\text{sto}}$ contains only coarse Fourier modes, a_κ and b_κ are the corresponding real sine and cosine Fourier basis functions, the Brownian motions are independent, and the spectral weights are $w_\kappa \propto |\kappa|^{-1/2}$. Thus the unresolved random forcing acts at large spatial scales, while the nonlinear Navier–Stokes dynamics transfer this variability across scales. In all our experiments, we set the stochastic forcing strength to be $\sigma = 0.3$.

Initial conditions Trajectories are initialized from filtered random velocity fields as in the JAX-CFD forced-turbulence setup, rescaled so that the maximum velocity magnitude is approximately 7. The combination of random initial conditions and stochastic coarse-mode forcing yields an ensemble of turbulent rollouts.

Training data generation Equation (49) is integrated on a 256×256 uniform grid over $[0, 2\pi]^2$ using a pseudo-spectral method with $2/3$-rule dealiasing. Time integration is performed via an implicit–explicit scheme: the linear (viscous and drag) terms are treated implicitly with Crank–Nicolson, while advection and stochastic forcing are advanced explicitly with a five-stage low-storage Carpenter–Kennedy Runge–Kutta-4 scheme. We use an internal timestep $\Delta t_{\mathrm{sim}} = 10^{-3}$ and integrate up to $T_{\mathrm{end}} = 25$. Each trajectory is then subsampled in time to $T = 1000$ evenly-spaced snapshots and spectrally subsampled in space to a 64×64 output grid. We generate 2048 training trajectories and a separate set of 1024 test trajectories.

Physical diagnostics Same as for the Burgers example; see Appendix D.3.

D.5 Setup and details of numerical experiments

Architectures for FTM

Duffing oscillator. We use a 3-layer MLP with hidden dimension 128 and ReLU activations. Time is provided by concatenation to the input. Training uses Adam with learning rate 5×10^{-4} under a warmup–cosine-decay schedule, batch size 8192, for 10^5 steps.

Rayleigh–Bénard. We use a 7-layer MLP with hidden dimension 128 and ReLU activations. Time is provided by concatenation to the input. Training uses Adam with learning rate 5×10^{-4} under a warmup–cosine-decay schedule, batch size 8192, for 10^5 steps.

Stochastic Burgers. We use a 1D U-Net with channel widths $[32, 64, 128]$, two residual blocks per scale, one middle residual block, and GeLU activations. Time is embedded via a 2-layer MLP of width 128 and injected into each residual block via a learned linear projection: the projected vector is added to the feature map both before the first convolution and before the second convolution, broadcast across the spatial dimension. Training uses Adam with learning rate 1×10^{-4} under a warmup–cosine-decay schedule, batch size 2048, for 2×10^5 steps.

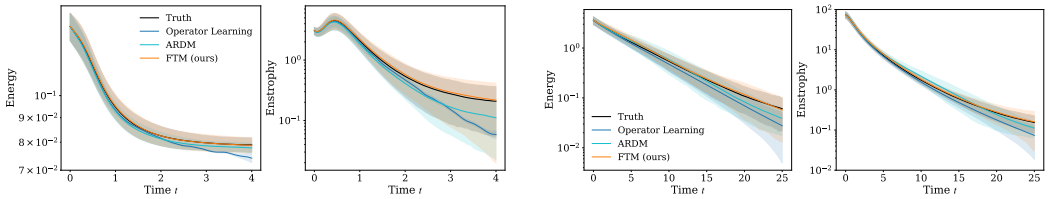
Stochastic Navier–Stokes. We use a 2D U-Net with channel widths $[128, 256, 512, 1024]$, two residual blocks per scale, one middle residual block, and GeLU activations, conditioned on three consecutive frames. Time is embedded via a 2-layer MLP of width 512 and injected into each residual block via a learned linear projection: the projected vector is added to the feature map both before the first convolution and before the second convolution, broadcast across the spatial dimensions. Training uses Adam with learning rate 1×10^{-4} under a warmup–cosine-decay schedule, batch size 128, for 10^6 steps.

Compute We train all models, including the other baselines, on compute nodes with H200 GPUs and 60GB of memory and 2TB of node main memory; and also use the same H200 GPUs for inference tasks. All runtime results are measured on H200 GPUs. Batch sizes are chosen so that they fit into the memory of the GPUs.

D.6 Additional description and details about experiments

Figure 2 The cost reported in panel (c) is the runtime of one loss evaluation, which has been averaged over 100 runs. The shaded area shows the standard deviation over these 100 runs. The time-step size $h = 10^{-2}$ and fixed over all τ, which is the setup used in the Duffing oscillator experiment.

Table 1 The W_2 error for the Duffing and Rayleigh -Benard convection is computed for each time step separately and then these errors are averaged, which is the reported number. The number in brackets is the standard deviation of the error over all time steps, which indicates how robustly the time marginals are matched over time. For the trajectory-based QoI error, the QoI is computed (Appendices D.1 and D.2) over all test trajectories. The reported number is the average relative error. The number in brackets is the standard deviation of the Monte Carlo estimator.



(a) stochastic Burgers turbulence (Burgulence)

(b) stochastically forced turbulence (Navier-Stokes)

Figure 7: FTM predicts ensembles with accurate energy and enstrophy statistics of stochastic Burgers (left panel) and stochastically forced turbulence (right panel) solution trajectories.

Table 5: Comparison of methods on the stochastic Burgers equation and stochastically forced turbulence (two-dim. Navier-Stokes).

Stochastic Burgers			
Method	error energy	error enstrophy	NFEs/time step
Operator learning [58]	$2.21\text{e}{-2}$ $(\pm 1.54\text{e}{-2})$	$2.55\text{e}{-1}$ $(\pm 2.34\text{e}{-1})$	1
ARDM 50 steps[34]	$1.46\text{e}{-2}$ $(\pm 3.39\text{e}{-3})$	$2.49\text{e}{-1}$ $(\pm 1.43\text{e}{-1})$	50
ARDM 75 steps [34]	$1.36\text{e}{-2}$ $(\pm 3.49\text{e}{-3})$	$2.30\text{e}{-1}$ $(\pm 1.26\text{e}{-3})$	75
ARDM 100 steps [34]	$1.24\text{e}{-2}$ $(\pm 3.12\text{e}{-3})$	$2.11\text{e}{-1}$ $(\pm 1.14\text{e}{-1})$	100
CFM 20 steps [2, 38]	$2.71\text{e}{-3}$ $(\pm 1.96\text{e}{-3})$	$1.53\text{e}{-1}$ $(\pm 1.28\text{e}{-1})$	20
FTM (ours)	$\mathbf{1.95}\text{e}{-\mathbf{3}}$ $(\pm 1.93\text{e}{-3})$	$\mathbf{5.32}\text{e}{-\mathbf{2}}$ $(\pm 2.23\text{e}{-2})$	1
stochastically forced turbulence (2D Navier-Stokes)			
Method	error energy	error enstrophy	NFEs/time step
Operator learning [58]	$2.13\text{e}{-1}$ $(\pm 1.58\text{e}{-1})$	$1.92\text{e}{-1}$ $(\pm 1.48\text{e}{-1})$	1
ARDM 50 steps [34]	$3.14\text{e}{-1}$ $(\pm 1.48\text{e}{-1})$	$2.82\text{e}{-1}$ $(\pm 9.60\text{e}{-2})$	50
ARDM 75 steps [34]	$2.75\text{e}{-1}$ $(\pm 1.40\text{e}{-1})$	$1.71\text{e}{-1}$ $(\pm 9.39\text{e}{-2})$	75
ARDM 100 steps [34]	$1.16\text{e}{-1}$ $(\pm 1.09\text{e}{-1})$	$1.34\text{e}{-1}$ $(\pm 8.46\text{e}{-2})$	100
CFM 20 steps [2, 38]	$1.39\text{e}{-1}$ $(\pm 6.30\text{e}{-2})$	$1.04\text{e}{-1}$ $(\pm 7.70\text{e}{-2})$	20
FTM (ours)	$\mathbf{3.40}\text{e}{-\mathbf{2}}$ $(\pm 2.50\text{e}{-2})$	$\mathbf{3.31}\text{e}{-\mathbf{2}}$ $(\pm 1.94\text{e}{-2})$	1

Table 2 and Table 5 For computing the energy and enstrophy estimates, see Appendix D.3. We report the relative error averaged over the whole time range. In brackets, we report the standard deviation of the errors over time.

Figure 3(right) The right panel shows the error of energy and enstrophy of FTM for the Burgers example for varying noise level σ.

Figure 5 The spectrum $E(k)$ shown in the plots on the right is computed by taking the Fourier decomposition of the generated field at times $t = 15$ and $t = 25$ and then plotting the coefficients.

D.7 Additional experiments

Table 5 shows more extensive results for the stochastic Burgers example and the stochastically forced turbulence example. Table 6 shows extensive results for using a distilled auto-regressive conditional flow matching (CFM) model, using mean-flows [26] for distillation.

Table 7 shows the standard deviation of the time-averaged mean errors for the stochastically forced turbulence example, demonstrating that FTM's improvements are substantially larger than the observed variability across samples.

Figure 7 shows the mean and standard deviation of energy and enstrophy of the ensemble trajectories compared to the ground truth for stochastic Burgers and stochastic turbulence examples.

E Baselines

All baselines are trained on the respective same data sets and comparable architectures as FTM to ensure a fair comparison.

Neural time stepper We follow the standard learned-simulator baseline used in, e.g., [48] and fit a deterministic one-step map $f_\theta : (t_k, x_k) \mapsto x_{k+1}$ by minimizing the regression loss

$$\mathcal{L}_{\text{step}}(\theta) = \frac{1}{NK} \sum_{i=1}^{N} \sum_{k=0}^{K-1} \left\| f_\theta(t_k, X^{(i)}(t_k)) - X^{(i)}(t_{k+1}) \right\|_2^2 .$$

At inference time, trajectories are generated autoregressively by setting $\hat{x}(t_{k+1}) = f_\theta(t_k, \hat{x}(t_k))$. For stochastic data, this baseline estimates the conditional mean of the next state under squared-error training analogous to operator learning, and therefore serves as a deterministic time-stepper baseline.

For each of the experiments, we use the same architecture and hyperparameter sets mentioned in Appendix D.

SDE learning We implement the neural Euler–Maruyama likelihood baseline of [23] for the Duffing and Rayleigh–Bénard problems. Two MLPs are trained jointly: a drift network $b_\theta(x, t)$ and a diffusion network $\sigma_\theta(t)$, the latter taking only time as input with a softplus output to enforce positivity. Given consecutive trajectory pairs (X_k, X_{k+1}) with step size Δt, the model minimizes the negative log-likelihood under the Euler–Maruyama Gaussian transition $X_{k+1} \mid X_k \sim \mathcal{N}(X_k + b_\theta(X_k, t_k), \Delta t, ; \sigma_\theta(t_k)^2, \Delta t \cdot I)$:

$$\mathcal{L} = \mathbb{E}\left[\log\left(\sigma_\theta^2, \Delta t\right) + \frac{|X_{k+1} - X_k - b_\theta, \Delta t|^2}{\sigma_\theta^2, \Delta t} \right].$$

After training, rollouts are generated by simulating the learned SDE with Euler–Maruyama. This baseline therefore learns a full stochastic model, rather than only the probability current velocity.

The hyperparameters for the MLPs for each of the experiments are the same as mentioned in Appendix Appendix D.

SDE matching We use the codebase published by the authors of the original paper [4] on GitHub. The model is trained in the state space, which is tractable for the Duffing oscillator and Rayleigh–Bénard convection. For the Duffing oscillator 128 hidden dimensions were used with 3 layers, and for the Rayleigh–Bénard problem 128 hidden dimensions with 7 layers was used. The learning rate for both was 5×10^{-4}, and the latent dimension for each was set to be equal to the state dimension (2 for Duffing oscillator, 9 for Rayleigh–Bénard convection). The Duffing oscillator was trained for $150,000$ training steps and the Rayleigh–Bénard convection was trained for $300,000$ training steps.

Direct ("plug-in") estimation of v We estimate the drift and diffusion coefficient as in SDE learning (see paragraph above) and additionally estimate the score conditioned on time with [28] from the samples of the time marginals. The setup for estimating the drift and diffusion coefficient is the same as in the SDE learning (see paragraph above). For learning the score, we use denoising score matching. Concretely, we train a time-conditioned denoising network with the v-parametrization on $T = 1000$ diffusion steps with a linear β-schedule. At inference we query at a small diffusion step $t = 3$ and recover the score. The score network uses the same MLP backbone as the drift and diffusion networks.

DICE We directly used the code that comes with the paper [7]. For the duffing oscillator, we use an MLP backbone with 128 hidden dimensions and 5 layers with swish activation functions. We use learning rate 5×10^{-4} with a cosine scheduler, and batchsize 256×256. For the Rayleigh–Bénard system, we use the same hyperparameters and architecture in the DICE paper.

Autoregressive conditional diffusion model (ARDM) For the Burgers equation example, we implement an ARDM baseline in the same spirit as [34]. The model learns the conditional transition law $X(t_{k+1})|X(t_k)$ rather than a deterministic one-step map. Concretely, we train a conditional score network with the denoising score-matching setup. At inference time, a next state is generated

Table 6: Comparison of distillation with mean flow to FTM and other few-step methods on the stochastic Burgers equation and stochastically forced turbulence.

Stochastic Burgers			
Method	error energy	error enstrophy	NFEs/time step
CFM 20 steps [2, 38]	$2.71\text{e}{-3}$ ($\pm 1.96\text{e}{-3}$)	$1.53\text{e}{-1}$ ($\pm 1.28\text{e}{-1}$)	20
MeanFlow 1 step[26]	$8.22\text{e}{-1}$ ($\pm 2.88\text{e}{-1}$)	$3.59\text{e}{+2}$ ($\pm 2.78\text{e}{+2}$)	1
MeanFlow 2 steps [26]	$1.69\text{e}{-1}$ ($\pm 5.77\text{e}{-2}$)	$9.81\text{e}{+1}$ ($\pm 8.05\text{e}{+1}$)	2
MeanFlow 5 steps [26]	$3.41\text{e}{-2}$ ($\pm 2.22\text{e}{-2}$)	$3.15\text{e}{-1}$ ($\pm 1.81\text{e}{-1}$)	5
MeanFlow 10 steps[26]	$2.06\text{e}{-2}$ ($\pm 1.58\text{e}{-2}$)	$3.15\text{e}{-1}$ ($\pm 1.81\text{e}{-1}$)	10
MeanFlow 20 steps [26]	$9.71\text{e}{-3}$ ($\pm 2.91\text{e}{-3}$)	$2.46\text{e}{-1}$ ($\pm 2.57\text{e}{-1}$)	20
Consistency Model 1 step [56]	$5.46\text{e}{-1}$ ($\pm 2.19\text{e}{-1}$)	$7.10\text{e}{+1}$ ($\pm 5.38\text{e}{+1}$)	1
Reflow 1 step [39]	$1.17\text{e}{-1}$ ($\pm 8.74\text{e}{-2}$)	$9.12\text{e}{+0}$ ($\pm 1.05\text{e}{+1}$)	1
Reflow + Distill 1 step [39]	$4.26\text{e}{-3}$ ($\pm 3.81\text{e}{-3}$)	$1.51\text{e}{-1}$ ($\pm 1.39\text{e}{-1}$)	1
FTM (ours)	$\mathbf{1.95}\text{e}{-\mathbf{3}}$ ($\pm 1.93\text{e}{-3}$)	$\mathbf{5.32}\text{e}{-\mathbf{2}}$ ($\pm 2.23\text{e}{-2}$)	1
stochastically forced turbulence (2D Navier-Stokes)			
CFM 20 steps [2, 38]	$1.39\text{e}{-1}$ ($\pm 6.30\text{e}{-2}$)	$1.04\text{e}{-1}$ ($\pm 7.70\text{e}{-2}$)	20
MeanFlow 1 step [26]	$2.46\text{e}{-1}$ ($\pm 1.25\text{e}{-1}$)	$4.37\text{e}{-1}$ ($\pm 2.78\text{e}{-1}$)	1
MeanFlow 2 steps [26]	$1.40\text{e}{-1}$ ($\pm 6.61\text{e}{-2}$)	$9.81\text{e}{-2}$ ($\pm 2.97\text{e}{-2}$)	2
MeanFlow 4 steps [26]	$1.22\text{e}{-1}$ ($\pm 8.45\text{e}{-2}$)	$6.74\text{e}{-2}$ ($\pm 3.66\text{e}{-2}$)	4
Consistency Model 1 step [56]	$1.59\text{e}{+0}$ ($\pm 5.52\text{e}{-1}$)	$7.30\text{e}{+0}$ ($\pm 5.90\text{e}{+0}$)	1
Reflow 1 step [39]	$6.09\text{e}{-1}$ ($\pm 6.55\text{e}{-1}$)	$4.26\text{e}{-1}$ ($\pm 4.21\text{e}{-1}$)	1
Reflow + Distill 1 step [39]	$1.33\text{e}{-1}$ ($\pm 6.90\text{e}{-2}$)	$1.09\text{e}{-1}$ ($\pm 8.47\text{e}{-2}$)	1
FTM (ours)	$\mathbf{3.40}\text{e}{-\mathbf{2}}$ ($\pm 2.50\text{e}{-2}$)	$\mathbf{3.31}\text{e}{-\mathbf{2}}$ ($\pm 1.94\text{e}{-2}$)	1

by initializing the sampler from the reference distribution and numerically integrating the learned reverse-time sampling dynamics conditioned on the current state x_k. Repeating this procedure autoregressively produces a trajectory. For the two-dimensional Navier–Stokes experiments in Appendix D.4, we directly use the codebase released with [34].

For the Burgers implementation, we used a 1D UNet with feature depths $[64, 128, 256]$, four residual blocks per scale, one middle residual block, and GeLU activations. The noise schedule is the linear β-schedule with $T = 1000$ levels. The loss is the same as the denoising score matching via the v-parametrization. The optimizer used is Adam with learning rate 1×10^{-4}, with a warmup and then cosine decay. Training is done for $650,000$ gradient steps. At inference, DDIM sampling is used with $\eta = 1.0$, thus making it stochastic.

For Navier Stokes, we use a 2D U-Net with ConvNext blocks and feature depths $[42, 64, 128, 256]$, two ConvNext residual blocks per scale, and GELU activations. Each ConvNext block applies a depthwise 7×7 convolution followed by two 3×3 convolutions with GroupNorm and GELU. Linear attention is applied at each resolution level and full self-attention at the bottleneck. The network is conditioned on three consecutive vorticity frames (clean conditioning: conditioning frames are passed without noise during both training and inference); the noisy target frame is concatenated to the conditioning frames to form the 4-channel UNet input. Time is embedded via sinusoidal embeddings projected by a two-layer MLP (widths $[64, 256]$ with GELU), and injected into each ConvNext block via a learned linear projection added to the feature map after the depthwise convolution, broadcast across spatial dimensions. The forward diffusion uses a linear noise schedule with $T{=}100$ steps ($\beta_0{=}5\times10^{-4}$, $\beta_T{=}0.1$, scaled from the standard 1000-step parametrisation). Training uses Adam with learning rate 1×10^{-4}, no weight decay, batch size 64, for $2,000,000$ training steps. Inference is done in the same way as in Burgers.

Autoregressive conditional flow matching and mean-flow model Analogously to ARDM, we implement an autoregressive conditional generative baseline, but replace the reverse diffusion sampler by a conditional flow matching (CFM) ODE [2, 38]. We also implemented a mean-flow-style distillation baseline [26, 8], but we used a two-stage procedure. We first train the autoregressive CFM model and then distill its sampling-time flow into a model of the average velocity.

We train the CFM model to map the standard Gaussian to the target distribution, which is the conditional transition law. For each training pair (x_t, x_{t+1}) we draw $z \sim \mathcal{N}(0, I)$ and a flow time

$\tau \sim \mathrm{Uniform}(0,1)$, construct the interpolant $x_\tau = \tau x_{t+1} + (1-\tau)z$, and regress the vector field onto the straight-line target $u_\tau = x_{t+1} - z$:

$$\mathcal{L}(\theta) = \mathbb{E}_{\tau,\, z,\, (x_t, x_{t+1})}\big[\|v_\theta(x_\tau, \tau \mid x_{\mathrm{cond}}) - \mathrm{sg}(u_\tau)\|^2\big]. \tag{51}$$

For the Stochastic Burgers experiments, we use a 1D U-Net with channel widths $[96, 192, 384, 384]$, one residual block per encoder and decoder level, one bottleneck block, and SiLU activations; all convolutions apply circular padding to respect the periodic spatial domain, and skip connections are merged by concatenation. Time τ is embedded via a 128-dimensional sinusoidal encoding projected to dimension 512 through a two-layer MLP (Linear $\to$ SiLU $\to$ Linear). Training uses Adam with gradient clipping (global norm 1.0), a peak learning rate of 3×10^{-4} under a cosine decay schedule with a 1000-step linear warm-up, batch size 256, and runs for $500{,}000$ gradient steps.

For Navier Stokes experiments, we use a 2D U-Net with channel widths $[128, 256, 512, 512]$ (doubling per level, capped at $4\times$ base), one residual block (SiLU, 3×3 convolution) per encoder and decoder level, and skip connections merged by channel concatenation. Time τ is embedded via a 128-dimensional sinusoidal encoding projected to dimension 512 through a two-layer MLP (Linear $\to$ SiLU $\to$ Linear), and conditioning is done by simple channel concatenation of the prior 3 frames; model parameters are maintained with an exponential moving average of decay 0.9999 and EMA weights are used at inference. Training uses Adam with gradient clipping (global norm 1.0), a peak learning rate of 3×10^{-4} under a cosine decay schedule with a 1000-step linear warm-up, batch size 256, and runs for $1{,}000{,}000$ gradient steps.

We distill the trained CFM vector field v_θ into a mean-velocity network $u_\psi(x_r, r, t \mid x_{\mathrm{cond}})$ following the ODE-regression formulation of [26]:

$$u_\psi(x_r, r, t) \;\approx\; \frac{\hat{x}_t - x_r}{t - r}, \tag{52}$$

where $\hat{x}_t$ is the endpoint obtained by integrating the *frozen* oracle v_θ forward from x_r over the interval $[r, t]$. At inference a single network evaluation suffices: $x_1 = z + u_\psi(z, 0, 1 \mid x_{\mathrm{cond}})$, but one can do multiple evaluations uniformly along the $[0, 1]$ interval. The MeanFlow loss is a regression task of the form:

$$\mathcal{L}(\psi) = \mathbb{E}_{r \le t,\, z,\, (x_t, x_{t+1})}\big[\|u_\psi(x_r, r, t \mid x_{\mathrm{cond}}) - \mathrm{sg}(u_{\mathrm{tgt}})\|^2\big]. \tag{53}$$

The network u_ψ shares the U-Net body of v_θ exactly. The only architectural change is the time-conditioning module: both interval endpoints r and t are embedded independently with the same sinusoidal encoding of dimension 128, concatenated to a 256-dimensional vector, and projected to 512 via a two-layer MLP. Training uses Adam with a peak learning rate of 10^{-4} under a cosine decay schedule with a 10000-step warm-up, batch size 256, and runs for 1,000,000 gradient steps.

Consistency Models We compare against consistency models trained via consistency distillation [56], which compress a pretrained diffusion model into a one-step generative model. Let $f_\theta(y, \tau \mid x_{\mathrm{cond}})$ denote the consistency model, which maps a noisy sample y at noise level τ to an estimate of the clean sample x_{t+1}. Given a pretrained diffusion model, we discretize the noise interval into levels $\tau_1 < \cdots < \tau_N$. For a data sample x_{t+1} and $z \sim \mathcal{N}(0, I)$, we construct

$$y_{\tau_{i+1}} = x_{t+1} + \tau_{i+1} z.$$

A single backward step of the teacher's probability-flow ODE produces $\hat{y}_{\tau_i}$ at the adjacent noise level τ_i. The consistency model is trained by minimizing

$$\begin{aligned}\mathcal{L}(f_\theta) = \mathbb{E}_{i,\, z,\, x_{t+1},\, x_{\mathrm{cond}}}\Big[\lambda(\tau_i)\big\| & f_\theta(y_{\tau_{i+1}}, \tau_{i+1} \mid x_{\mathrm{cond}}) \\ & - \mathrm{sg}\left(f_{\theta^-}(\hat{y}_{\tau_i}, \tau_i \mid x_{\mathrm{cond}})\right)\big\|_2^2\Big],\end{aligned} \tag{54}$$

where θ^- denotes the parameters of an exponential moving average of the consistency model and λ is a weighting function. At inference, samples are generated from Gaussian noise conditioned on x_{cond} using a single network evaluation.

All consistency models use the same U-Net as the corresponding pretrained diffusion model, with 100 noise discretization steps. For Stochastic Burgers experiments, training uses Adam with a peak learning rate of 2×10^{-5} under a cosine decay schedule to 10^{-6} with a 500-step warm-up and batch

Table 7: Robustness of time-averaged relative errors in energy and enstrophy for the stochastically forced turbulence example. Standard deviations are computed across independent samples.

Method	Energy error		Enstrophy error	
	Mean	Std. dev.	Mean	Std. dev.
Operator Learning	2.13e-01	3.9441e-03	1.92e-01	2.6734e-02
CFM (20 steps)	2.61e-01	3.6386e-03	2.71e-01	2.4079e-02
ARDM (100 steps)	1.16e-01	4.1512e-03	1.34e-01	2.9114e-02
FTM (ours)	**3.43e-02**	4.8782e-03	**3.30e-02**	3.0782e-02

size 4096; the model is trained for $100{,}000$ gradient steps.
For Navier-Stokes experiments, training uses Adam with a peak learning rate of 10^{-4} under a cosine decay schedule with a $1{,}000$-step warm-up and batch size 128; the model is trained for $100{,}000$ gradient steps.

ReFlow Models and distillation In addition to MeanFlow and Consistency Models, we compare against one-step models obtained by the Reflow distillation procedure [39]. The original CFM model is trained on the independent coupling (z, x_{t+1}), with $z \sim \mathcal{N}(0, I)$ drawn independently of x_{t+1}. For $j \geq 0$, let $\Psi^{(j)}(z \mid x_{\mathrm{cond}}) := y(1)$ denote the time-one flow map of the sampling ODE $\frac{\mathrm{d}}{\mathrm{d}\tau} y(\tau) = v^{(j)}(y(\tau), \tau \mid x_{\mathrm{cond}})$, $y(0) = z$, where $v^{(0)}$ denotes the original CFM vector field. One Reflow step replaces the independent coupling by the deterministic coupling $(z, \hat{y}_1)$ with $\hat{y}_1 := \Psi^{(j)}(z \mid x_{\mathrm{cond}})$, and fits a new vector field $v^{(j+1)}$ with the following loss on the linear interpolants $y_\tau = \tau \hat{y}_1 + (1-\tau) z$:

$$\mathcal{L}\big(v^{(j+1)}\big) = \mathbb{E}_{\tau,\, z,\, x_{\mathrm{cond}}} \Big[\big\| v^{(j+1)}(y_\tau, \tau \mid x_{\mathrm{cond}}) - \mathrm{sg}\left(\hat{y}_1 - z\right) \big\|_2^2 \Big]. \tag{55}$$

We perform three such stages and generate trajectories from the final stage with a single Euler step (Reflow 1 step in Table 6). For Reflow + Distill, we additionally train a one-step model v_ϕ, initialized from $v^{(3)}$, by regressing its single Euler step directly onto the endpoint of the Reflow model:

$$\mathcal{L}(v_\phi) = \mathbb{E}_{z,\, x_{\mathrm{cond}}} \Big[\big\| z + v_\phi(z, 0 \mid x_{\mathrm{cond}}) - \mathrm{sg}\left(\Psi^{(3)}(z \mid x_{\mathrm{cond}})\right) \big\|_2^2 \Big]. \tag{56}$$

All Reflow and distilled models use the same U-Net as the corresponding CFM model, with $\Psi^{(j)}$ evaluated with 20 Euler steps. For Stochastic Burgers experiments, training uses Adam with a peak learning rate of 10^{-4} under a cosine decay schedule with a 500-step warm-up and batch size 512; each Reflow stage is trained for $100{,}000$ gradient steps, and the distillation for $50{,}000$ gradient steps. For Navier-Stokes experiments, training uses Adam with a peak learning rate of 10^{-4} under a cosine decay schedule with a $1{,}000$-step warm-up and batch size 256; each Reflow stage is trained for $100{,}000$ gradient steps, and the distillation for $100{,}000$ gradient steps.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction state the central contributions: learning the probability current velocity from trajectories, deriving a scalable one-step FTM loss, analyzing its stability, and evaluating the method on stochastic and turbulent systems. The claims are scoped to ensemble prediction, first-order transport, and fast deterministic rollouts rather than recovery of the full stochastic path law; see the Abstract, Introduction, and Sections 4 and the limitations section.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper includes a "Conclusions and limitations" section. It explicitly states that FTM does not recover the full stochastic path law, that the theory is limited to finite-dimensional SDEs on $\mathbb{R}^d$ with additive noise and regularity assumptions, and that the one-step loss has a data-resolution tradeoff scaling like $1/(Nh)$.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best

judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: The paper states assumptions in the setup and in the statements of the theoretical results, including smoothness, boundedness, additive noise, and Lipschitz-type assumptions. Proofs and derivations are provided or referenced in the appendices, including the Stratonovich identity, the MSE bound for the empirical FTM loss, marginal error bounds, and path-dependent QoI bounds.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The main text contains descriptions of the benchmarks, metrics, and baselines (Section 4). The appendix (Appendices D and E) contains details about architecture specifications, optimizer settings, concrete setup of baselines. Code is available in the supplemental.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.

(b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
(c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: We provide anonymized supplementary code and data. Code will be released on GitHub if the paper gets accepted.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: The main important points of the experiments are presented in Section 4. We provide additional technical details in Appendix D.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: For results that report the main claims of the paper, the reported errors are equipped with standard deviations (e.g., Table 1, Figure 4) and results are presented over multiple replicates where appropriate (Figure 2d).

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We report computer architecture, GPU memory, and node main memory; see Appendix D.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The work studies algorithms for simulated stochastic and turbulent dynamical systems and does not involve human subjects, private data, or deployment in a sensitive domain. To the best of the authors' knowledge, the research conforms to the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.

- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We do not expect negative societal impacts that are specific to our method (see Section 5).

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The work is a methodological contribution evaluated on simulated scientific systems.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: Original papers that produced code package and/or data sets that are used are cited throughout the main text (e.g., Table 1 where baselines are cited).

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [Yes]

Justification: The code and data sets in the supplemental material are documented. No other assets (pre-trained models, etc) are released.

Guidelines:

- The answer [N/A] means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [N/A]

Justification: The paper does not involve crowdsourcing or research with human subjects.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing or research with human subjects, so IRB or equivalent human-subjects review is not applicable.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: The core methodology does not use LLMs as an important, original, or non-standard component.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.