Strong Stochastic Flow Maps

Anonymous Authors[1]

Abstract

Flow and diffusion models generate high-quality samples in many modalities; however, many network evaluations are required during inference due to numerical integration of an underlying differential equation. Flow maps alleviate this problem by learning the solution map of the differential equation directly, enabling few-step sampling. Yet, current methods are restricted to approximating the solution map of ODEs. These methods can be used to learn the transition kernel of an SDE, thereby obtaining a solution map that recovers the marginal distributions of the process (weak convergence) rather than the solution path (strong convergence). We propose STRONG STOCHASTIC FLOW MAPS (SSFMs) as a novel framework for learning the *strong* solution map of additive-noise SDEs, directly generalizing deterministic flow maps to the stochastic setting. A polynomial approximation to Brownian motion is introduced and shown to converge pathwise. These results enable a simulation-free training objective for the solution map of diffusion models. We demonstrate that SSFMs outperform previous flow map methods on image generation and enable few-step sampling of molecular systems.

1. Introduction

Deep generative models based on *neural differential equations* (Chen et al., 2018; Kidger, 2022) have become one of the most successful model classes for solving a variety of problems such as generative modeling (Liu et al., 2023; Song, Sohl-Dickstein, et al., 2021) and time-series data (Oh et al., 2024; Walker et al., 2024). The application of neural differential equations to generative modeling—diffusion models (Ho et al., 2020; Sohl-Dickstein et al., 2015; Song, Sohl-Dickstein, et al., 2021) and flow matching models (Lipman, Chen, et al., 2023; Liu et al., 2023; Peluchetti, 2021)—have become state-of-the-art for many different tasks including image (Rombach et al., 2022) and video (Blattmann et al., 2023) generation, protein design (Rector-Brooks et al., 2026; Watson et al., 2023), and Boltzmann sampling (Rehman et al., 2026).

While quite expressive generative models, neural differential equations often require a large *number of function evaluations* (NFEs) of the learned vector fields to integrate the underlying differential equation. As such there has been great interest in learning how to improve the computational efficiency of these models, *e.g.*, proposing better numerical schemes to reduce the number of NFEs whilst maintaining similar performance (Gonzalez et al., 2023; Lu et al., 2022; Zhang & Chen, 2023). Recently, another direction has looked at how to learn the solution, or flow, map associated with a neural *ordinary differential equation* (ODE) (Boffi et al., 2024; Geng, Deng, et al., 2025; Heek et al., 2024; Kim et al., 2024; Liu et al., 2023; Sabour et al., 2025; Song, Dhariwal, et al., 2023). These methods, which learn a *neural flow map*, have obtained state-of-the-art performance with low NFEs in image generation (Geng, Deng, et al., 2025; Geng, Lu, et al., 2025) and Boltzmann sampling (Rehman et al., 2026) compared to the 100s of steps required with diffusion models.

Previous work has focused on learning a *deterministic* map from the source noise to the target distribution. Recent work has extended these deterministic maps to the stochastic setting by approximating the transition distributions of a stochastic process (Holderrieth, Chen, et al., 2026; Passaro et al., 2026; Potaptchik et al., 2026). However, such approaches do not allow for the estimation of pathwise observables as they are fundamentally decoupled from the underlying stochastic differential equation (SDE) and are only capable of computing *weak* (convergence in distribution) solutions of the stochastic process (Øksendal, 2003).

We introduce STRONG STOCHASTIC FLOW MAPS (SSFMs), a novel framework which obtains the *strong* (convergence in path) solution map to additive-noise SDEs. This naturally extends ODE-based flow maps, which are pathwise solution maps, to the stochastic setting. Specifically, given a realization of the Brownian path and an initial condition, we learn the pathwise solution map of the additive-noise

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

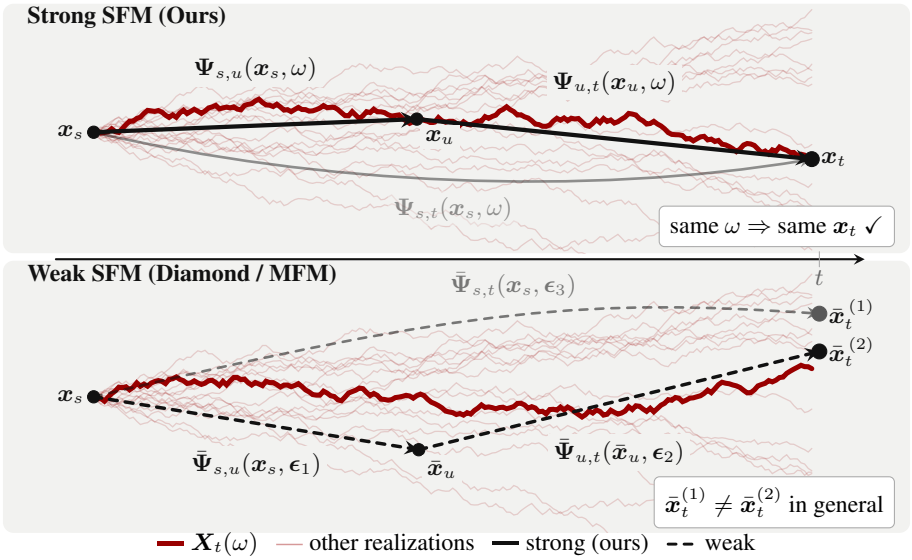

Figure 1. **Strong *vs*. weak stochastic flow maps.** *Top*: The strong stochastic flow map solution is consistent pathwise for a given realization of the Brownian motion $\boldsymbol{W}_t(\omega)$. *Bottom*: The weak stochastic flow map samples independently from the marginal distribution at each time.

SDE. To efficiently implement such a model we require a novel set of tools which we construct in this work. Our main contributions are summarized as follows:

- We develop a novel framework for learning the strong solution to additive-noise SDEs in contrast with prior works which consider weak approximations.
- We formalize a training objective for learning this map which admits a simulation-free algorithm for obtaining the strong solution map of diffusion models.
- We introduce a polynomial approximation of the Brownian motion and prove that it converges in α-Hölder distance, has tractable coefficients, and admits closed form Chen relations.
- We demonstrate this formulation achieves state-of-the-art performance on CIFAR-10 and enables generation of equilibrium molecular conformations with as few as 2 NFEs for Alanine-Dipeptide.

2. Background and related work

Neural SDEs. Neural SDEs are the stochastic extension of neural ODEs (Chen et al., 2018), independently proposed by Kidger et al. (2021) and Li et al. (2020) which aim to learn the drift and diffusion coefficients for an arbitrary SDE. In this work we will focus specifically on the case of additive-noise SDEs with known diffusion coefficients, given by

$$\mathrm{d}\boldsymbol{X}_t = \boldsymbol{f}^\theta(t, \boldsymbol{X}_t)\,\mathrm{d}t + g(t)\,\mathrm{d}\boldsymbol{W}_t, \tag{1}$$

where $\boldsymbol{f}^\theta$ is the drift coefficient we aim to learn. Previous work has focused on learning these coefficients by integrating the SDE numerically and then performing backpropagation through the numerical solution. However, for many SDEs studied in generative modelling we can learn the drift coefficient via *simulation-free* training which has underpinned the popularity of different *matching* objectives such as score matching and flow matching. In this work, we focus on diffusion SDEs; however, our results hold for any additive-noise SDE.

Flow matching and diffusion. In this section we introduce the necessary background on flow/diffusion models (Albergo et al., 2025; Lipman, Chen, et al., 2023; Liu et al., 2023; Peluchetti, 2021; Tong et al., 2024). We denote data samples as $\boldsymbol{X}_1 \in \mathbb{R}^d$ drawn from the data distribution with density $p_1 \equiv q(\boldsymbol{x})$. We take the source distribution to be a unit Gaussian with density $p_0 \equiv p(\boldsymbol{x})$ with $\boldsymbol{X}_0 \sim p_0$. Following Lipman, Havasi, et al. (2024) we consider the scenario of *affine Gaussian probability paths* where we define a random variable $\boldsymbol{X}_t := \alpha_t \boldsymbol{X}_1 + \sigma_t \boldsymbol{X}_0$ with noise schedule (α_t, σ_t), where $\alpha_t, \sigma_t \geq 0$ with $\alpha_0 = \sigma_1 = 0$, $\alpha_1 = \sigma_0 = 1$, and α_t (σ_t resp.) is strictly monotonically increasing (decreasing resp.) and continuously differentiable.

In the flow matching framework we learn the marginal vector field as

$$\begin{aligned} \boldsymbol{u}_t(\boldsymbol{X}) &= \int_{\mathbb{R}^d} \boldsymbol{u}_{t|1}(\boldsymbol{X}|\boldsymbol{X}_1) p_{1|t}(\boldsymbol{X}_1|\boldsymbol{X})\,\mathrm{d}\boldsymbol{X}_1, \\ p_{1|t}(\boldsymbol{X}_1|\boldsymbol{X}) &= \frac{p_{t|1}(\boldsymbol{X}|\boldsymbol{X}_1) p_1(\boldsymbol{X}_1)}{p_t(\boldsymbol{X})}, \end{aligned} \tag{2}$$

where $\boldsymbol{u}_{t|1}(\cdot|\boldsymbol{X}_1)$ is the vector field conditioned on a data sample $\boldsymbol{X}_1$. This vector field can be shown to satisfy,

$$\boldsymbol{X}_0 \sim p_0, \quad \boldsymbol{X}_0 + \int_0^t \boldsymbol{u}_s(\boldsymbol{X}_s)\,\mathrm{d}s = \boldsymbol{X}_t \sim p_t, \tag{3}$$

such that the solution to (3) at time $t = 1$ yields samples from the data distribution $q \equiv p_1$.

The ODE in Equation (3) is referred to as the *probability flow ODE* (Song, Sohl-Dickstein, et al., 2021) and can be written as an SDE with the same marginal distributions, given by

$$\begin{aligned} \mathrm{d}\boldsymbol{X}_t &= \left[2\boldsymbol{u}_t(\boldsymbol{X}_t) - \frac{\dot{\alpha}_t}{\alpha_t}\boldsymbol{X}_t\right]\,\mathrm{d}t + \nu_t\,\mathrm{d}\boldsymbol{W}_t, \\ \nu_t^2 &= 2\frac{\dot{\alpha}_t}{\alpha_t}\sigma_t^2 - 2\sigma_t\dot{\sigma}_t, \end{aligned} \tag{4}$$

where $\boldsymbol{W}_t$ is the standard Brownian motion. This derivation follows straightforwardly from Anderson (1982) and has been discussed more recently in the context of diffusion models (Holderrieth, Singer, et al., 2026; Maoutsa et al., 2020; Song, Sohl-Dickstein, et al., 2021). The implications for model performance when sampling from either the SDE or ODE formulation with equivalent marginals is discussed in (Albergo et al., 2025; Nie et al., 2024).

Table 1. Comparison of flow-based models.

Method	Few-step	Pathwise	Stochastic
Flow models	✗	✓	✗
Diffusion models	✗	✓	✓
GLASS flows	✗	✗	✓
Flow maps	✓	✓	✗
Weak SFMs	✓	✗	✓
SSFM (ours)	✓	✓	✓

Flow maps. Since integrating the generative models in (3) or (4) requires many function evaluations, recent work has proposed to instead learn the integral, or solution map, directly at training time. These works include consistency models (Heek et al., 2024; Kim et al., 2024; Song, Dhariwal, et al., 2023), shortcut models (Frans et al., 2025), mean flows (Geng, Deng, et al., 2025), and flow maps (Boffi et al., 2024, 2025; Sabour et al., 2025). There exists a variety of techniques for training such models. However, training can largely be broken down into two loss components: one term learns the instantaneous behavior at $s = t$ and another learns the flow map $t > s$. Training via such an objective is referred to as self-distillation (Boffi et al., 2025).

Weak stochastic flow maps. Recently, several works have explored a stochastic extension of deterministic flow maps (Holderrieth, Chen, et al., 2026; Passaro et al., 2026; Potaptchik et al., 2026), which can be characterized as learning the transition kernel $p_{t|s}(\boldsymbol{X}_t|\boldsymbol{X}_s)$ of some underlying SDE. Specifically, these methods proceed by defining a deterministic ODE flow map where each step of the map is defined by an inner flow model, given by

$$\boldsymbol{X}_s + \int_s^t \bar{\boldsymbol{u}}_\tau(\bar{\boldsymbol{X}}_\tau|\boldsymbol{X}_s, s)\,\mathrm{d}\tau = \bar{\boldsymbol{X}}_t \sim p_{t|s}(\cdot|\boldsymbol{X}_s). \tag{5}$$

Since these methods learn the transition kernel $p_{t|s}$ they are described as exhibiting *weak* (in distribution) convergence (Øksendal, 2003). This is in contrast to the pathwise solution map of the underlying additive-noise SDE which generates $\boldsymbol{X}_t$ given an initial condition $\boldsymbol{X}_s$ and a realization of the Brownian motion $\{\boldsymbol{W}_u\}_{s\le u\le t}$. We summarize the relationship between SSFMs and prior methods in Table 1.

3. Strong Stochastic Flow Maps

In this section we describe the transition from deterministic flow maps to strong stochastic flow maps. We present a natural extension of deterministic flow maps by learning the solution map from both an initial condition and a realization of the Brownian path. We then show how this solution map can be obtained by minimization of an appropriate self-distillation objective.

3.1. The Itô map

Consider the following additive-noise Itô SDE,

$$\mathrm{d}\boldsymbol{X}_t = \boldsymbol{f}(t, \boldsymbol{X}_t)\,\mathrm{d}t + \boldsymbol{g}(t)\,\mathrm{d}\boldsymbol{W}_t, \tag{6}$$

where $\boldsymbol{X}_t \in \mathbb{R}^d$ is a continuous-valued stochastic process with initial condition $\boldsymbol{X}_0$, $\boldsymbol{f} : \mathbb{R} \times \mathbb{R}^d \to \mathbb{R}^d$ is the drift function, $\boldsymbol{g} : \mathbb{R} \to \mathbb{R}^{d\times w}$ is the diffusion coefficient and $\boldsymbol{W}_t \in \mathbb{R}^w$ is Brownian motion. It is assumed that $\boldsymbol{f}$ and $\boldsymbol{g}$ are suitably regular such that a unique strong solution $\boldsymbol{X}_t$ exists (see Assumption A.1).

It is natural to consider the solution map of this SDE, $\boldsymbol{\Psi}_{s,t} : (\boldsymbol{X}_s, \boldsymbol{W}_{[s,t]}) \mapsto \boldsymbol{X}_t$, where the Brownian path is written as $\boldsymbol{W}_{[s,t]} = \{\boldsymbol{W}_u\}_{s\le u\le t}$. This map $\boldsymbol{\Psi}_{s,t}$ is known as the Itô map.[1] For an SDE with additive noise and suitably regular coefficients, the Itô map is well-posed and continuous (Friz & Victoir, 2010). It is this map that we aim to approximate with a neural network, $\boldsymbol{\Psi}^\theta_{s,t}$.

Since the Itô map is defined with respect to both an initial condition, $\boldsymbol{X}_0$, and a realization of $\boldsymbol{W}_{[s,t]}$, the convergence to the solution $\boldsymbol{X}_t$ is pathwise. In stochastic analysis, a process approximating such a pathwise solution is called

[1] Note that for a general SDE with state-dependent diffusion, $\boldsymbol{g}(t, \boldsymbol{X}_t)$, the Itô map is not a well-defined continuous function. This is solved by rough path theory and the Itô-Lyons map (Lyons, 1998).

strongly convergent (Øksendal, 2003). We therefore refer to our method as *Strong Stochastic Flow Maps*.

3.1.1. Constructing the Itô map

Analogously to the deterministic flow map, we can derive a tangent condition that the Itô map satisfies. This result will allow us to associate the vector fields of an SDE with the Itô map.

Lemma 3.1 (Tangent Condition)**.** *Let $\boldsymbol{\Psi}_{s,t}(\boldsymbol{X}_s, \boldsymbol{W}_{[s,t]})$ denote the Itô map for (6). Then,*

$$\lim_{s \to t} \mathrm{d}\boldsymbol{\Psi}_{s,t}(\boldsymbol{X}_s, \boldsymbol{W}_{[s,t]}) = \boldsymbol{f}(t, \boldsymbol{X}_t)\,\mathrm{d}t + \boldsymbol{g}(t)\,\mathrm{d}\boldsymbol{W}_t. \tag{7}$$

To construct the Itô map we therefore propose an Euler-Maruyama step-like object that satisfies the tangent condition. We will show that such a parameterization attains the Itô map when trained according to the objective in (14).

Proposition 3.2 (Strong Stochastic Flow Map)**.** *Consider an Euler-Maruyama parameterization of the Itô map,*

$$\begin{aligned}\boldsymbol{\Psi}_{s,t}(\boldsymbol{X}_s, \boldsymbol{W}_{[s,t]}) = \boldsymbol{X}_s &+ \boldsymbol{f}_{s,t}(\boldsymbol{X}_s, \boldsymbol{W}_{[s,t]})(t-s) \\ &+ \boldsymbol{g}_{s,t}(\boldsymbol{W}_{[s,t]})(\boldsymbol{W}_t - \boldsymbol{W}_s),\end{aligned} \tag{8}$$

where $\boldsymbol{f}, \boldsymbol{g}$ are twice continuously differentiable, Lipschitz in both time arguments, satisfy the conditions that $\boldsymbol{f}_{t,t}(\boldsymbol{X}_t, \boldsymbol{W}_{[s,t]}) = \boldsymbol{f}_{t,t}(\boldsymbol{X}_t)$, and $\boldsymbol{g}_{t,t}(\boldsymbol{W}_{[s,t]}) = \boldsymbol{g}_{t,t}$ (i.e. the coefficients are independent of $\boldsymbol{W}_{[s,t]}$ for $s = t$). Then, $\boldsymbol{\Psi}_{s,t}$ satisfies the tangent condition (7) if and only if

$$\boldsymbol{f}_{t,t}(x_t) = \boldsymbol{f}(t, x_t), \quad \boldsymbol{g}_{t,t} = \boldsymbol{g}(t).$$

The proposed stochastic flow map in Proposition 3.2 introduces a drift $\boldsymbol{f}_{s,t}$ and a diffusion $\boldsymbol{g}_{s,t}$. The intuition behind these functions is that they act as the normalized drift and diffusion integrals, respectively. Therefore, they must depend on the underlying driving path $\boldsymbol{W}_{[s,t]}$.

We see from Proposition 3.2 that for the stochastic flow map to satisfy the tangent condition, the drift integral must collapse to the drift function, $\boldsymbol{f}_{t,t}(x_t) = \boldsymbol{f}(t, x_t)$, and the diffusion integral must collapse to the diffusion coefficient, $\boldsymbol{g}_{t,t} = \boldsymbol{g}(t)$, as $s \to t$. This indicates that $\boldsymbol{f}_{t,t}(x_t)$ and $\boldsymbol{g}_{t,t}$ can be estimated by the matching objective in (14).

Next, we must establish an objective that constrains the finite-time, $t > s$, behaviour of the stochastic flow map. Consider the semigroup condition,

$$\boldsymbol{\Psi}_{s,t}(\boldsymbol{X}_s, \boldsymbol{W}_{[s,t]}) = \boldsymbol{\Psi}_{u,t}(\boldsymbol{\Psi}_{s,u}(\boldsymbol{X}_s, \boldsymbol{W}_{[s,u]}), \boldsymbol{W}_{[u,t]}), \tag{9}$$

for $s < u < t$. Note this condition is satisfied by the Itô map (see the proof of Proposition 3.3). It is possible to show that if the strong stochastic flow map construction from Proposition 3.2 satisfies the tangent condition (7) and the semigroup property, then this map *is* the Itô map. This is given by Proposition 3.3.

Proposition 3.3 (Semigroup condition)**.** *Let $\boldsymbol{\Psi}_{s,t}(\boldsymbol{X}_s, \boldsymbol{W}_{[s,t]})$ denote the strong stochastic flow map satisfying (7) and (8). Then $\boldsymbol{\Psi}_{s,t}(\boldsymbol{X}_s, \boldsymbol{W}_{[s,t]})$ is the Itô map if and only if the semigroup property (9) holds.*

3.1.2. Constructing the Brownian motion

Since the Itô map requires a realization of the Brownian path $\boldsymbol{W}_{[s,t]}$, so too do the drift and diffusion integral functions $\boldsymbol{f}_{s,t}, \boldsymbol{g}_{s,t}$. These functions will ultimately be parameterized as neural networks. We therefore require an efficient characterization of $\boldsymbol{W}_{[s,t]}$ that we can pass as input.

A naïve approach is to simply pass a piecewise linear approximation to $\boldsymbol{W}_{[s,t]}$ over N intervals. However, to obtain any reasonable approximation to the path, too many intervals would be required. We instead look to pass the coefficients of a polynomial expansion of $\boldsymbol{W}_{[s,t]}$. Specifically, we consider a piecewise polynomial expansion to Brownian motion in terms of shifted Legendre polynomials (Foster et al., 2020; Habermann, 2021).

This polynomial expansion has a number of desirable properties: (1) the coefficients of this expansion appear in the stochastic Taylor expansion of the SDE in (6) enabling larger time-step approximation; (2) the coefficients are independently and normally distributed allowing for tractable and exact sampling; (3) the coefficients admit closed-form Chen relations (Chen, 1954, 1957) such that two coefficients over the sub-intervals $[s, u]$, $[u, t]$ combine into a single coefficient over $[s, t]$. This property enables the semigroup objective to be implemented. These results are shown in Theorem 3.4.

The following results are theoretically dense, relying on elements of rough path theory (Lyons, 1998). However, the polynomial expansion of Brownian motion is visually intuitive; see Figure 3.

Definition 3.1 (Polynomial approximation for Brownian motion)**.** Let $\widetilde{P}_n : [0, 1] \to \mathbb{R}$ denote the n-th shifted Legendre polynomial on $[0, 1]$. We define the

n-th coefficient of the polynomial expansion as,

$$\boldsymbol{I}_{s,t}^{(n)} = \int_s^t \widetilde{P}_n\left(\frac{u-s}{t-s}\right) \mathrm{d}\boldsymbol{W}_u. \tag{10}$$

As introduced in (Foster et al., 2020; Habermann, 2021), the degree-N polynomial approximation of the Brownian motion on $[s,t]$ takes the form,

$$\boldsymbol{W}_{u,v}^{(N)} = \sum_{n=0}^{N-1} \frac{2n+1}{t-s} \boldsymbol{I}_{s,t}^{(n)} \int_u^v \widetilde{P}_n\left(\frac{r-s}{t-s}\right) \mathrm{d}r, \tag{11}$$

for each increment $[u,v] \subseteq [s,t]$.

Theorem 3.4 (Properties of $\boldsymbol{W}_{u,v}^{(N)}$)**.** *The polynomial approximation of the Brownian motion in Definition 3.1 has the following properties:*

1. *Converges to Brownian motion in the α-Hölder distance, with $\alpha \in [0, \frac{1}{2})$,*
2. *The coefficients $\boldsymbol{I}_{s,t}^{(n)}$ are independently and normally distributed with*

$$\boldsymbol{I}_{s,t}^{(n)} \sim \mathcal{N}\left(\boldsymbol{0}, \frac{(t-s)}{2n+1}\boldsymbol{I}\right), \tag{12}$$

3. *The coefficients $\boldsymbol{I}_{s,t}^{(n)}$ admit closed form Chen relations.*

3.2. Training

Given the polynomial approximation in Definition 3.1, we can write the Strong Stochastic Flow Map as

$$\begin{aligned}\boldsymbol{\Psi}_{s,t}^{\theta}(\boldsymbol{X}_s, \boldsymbol{I}_{s,t}^{(N)}) = \boldsymbol{X}_s &+ \boldsymbol{f}_{s,t}^{\theta}(\boldsymbol{X}_s, \boldsymbol{I}_{s,t}^{(N)})(t-s) \\ &+ \boldsymbol{g}_{s,t}^{\theta}(\boldsymbol{I}_{s,t}^{(N)})(\boldsymbol{W}_t - \boldsymbol{W}_s),\end{aligned} \tag{13}$$

where the coefficients up to degree N, given by $\boldsymbol{I}_{s,t}^{(N)} = \{\boldsymbol{I}_{s,t}^{(n)}\}_{n \le N}$, are passed to the neural network terms $\boldsymbol{f}_{s,t}^{\theta}, \boldsymbol{g}_{s,t}^{\theta}$. Note that $\boldsymbol{W}_t - \boldsymbol{W}_s = \boldsymbol{I}_{s,t}^{(0)}$.

3.2.1. Self-distillation objective

We have seen that the stochastic flow map in (8) *is* the Itô map if and only if the diagonal integrals are equal to the SDE coefficients, $\boldsymbol{f}_{t,t}(x_t) = \boldsymbol{f}(t, x_t), \boldsymbol{g}_{t,t} = \boldsymbol{g}(t)$, and the semigroup condition is satisfied.

These two properties allow us to write down an objective for the Itô map. This is shown by Theorem 3.5.

Theorem 3.5 (Self-distillation Objective)**.** *Let $\boldsymbol{\Psi}_{s,t}(\boldsymbol{X}_s, \boldsymbol{W}_{[s,t]})$ denote the Itô map for (6). Then, this map is given by the strong stochastic flow map in (8) where $\boldsymbol{v}_{s,t} = [\boldsymbol{f}_{s,t}, \boldsymbol{g}_{s,t}]$ is the unique global minimizer over $\hat{\boldsymbol{v}}$ of*

$$\mathcal{L}_{SD}(\hat{\boldsymbol{v}}) = \mathcal{L}_{\boldsymbol{f},\boldsymbol{g}}(\hat{\boldsymbol{v}}) + \mathcal{L}_D(\hat{\boldsymbol{v}}), \tag{14}$$

where

$$\begin{aligned}\mathcal{L}_{\boldsymbol{f},\boldsymbol{g}}(\hat{\boldsymbol{v}}) = \mathbb{E}_{t,\boldsymbol{X}_t}\Big[&\|\boldsymbol{f}(t,\boldsymbol{X}_t) - \hat{\boldsymbol{f}}_{t,t}(\boldsymbol{X}_t)\|_2^2 \\ &+ \|\boldsymbol{g}(t) - \hat{\boldsymbol{g}}_{t,t}\|_2^2\Big],\end{aligned} \tag{15}$$

$$\begin{aligned}\mathcal{L}_D(\hat{\boldsymbol{v}}) = \mathbb{E}_{s,u,t,\boldsymbol{X}_s,\boldsymbol{W}_{[s,t]}}\Big[&\|\hat{\boldsymbol{\Psi}}_{s,t}(\boldsymbol{X}_s, \boldsymbol{W}_{[s,t]}) \\ &- \hat{\boldsymbol{\Psi}}_{u,t}(\hat{\boldsymbol{\Psi}}_{s,u}(\boldsymbol{X}_s, \boldsymbol{W}_{[s,u]}), \boldsymbol{W}_{[u,t]})\|_2^2\Big].\end{aligned} \tag{16}$$

The objective in (14) is written in terms of the drift and diffusion coefficients of the SDE for which we wish to learn the Itô map. In the case of flow models (flow matching, diffusion, stochastic interpolants) the drift coefficient is known conditioned on an end-point sample, $\boldsymbol{X}_1$ (and/or $\boldsymbol{X}_0$), obtained from the marginal at $t=1$ (and/or $t=0$).

The results presented in this section are proved in Appendix A.

3.2.2. General training method

By Theorems 3.5 and 3.4, this parameterization trained according to (14) converges to the Itô map as $N \to \infty$. In practice, as the variance of the coefficient terms decay with $1/n$, we can obtain a sufficient approximation using a finite number of coefficients.

Note that $\mathcal{L}_{\boldsymbol{f},\boldsymbol{g}}$ provides a training signal for $s=t$ (*i.e.* $\boldsymbol{f}_{t,t}^{\theta}, \boldsymbol{g}_{t,t}^{\theta}$), but $\mathcal{L}_{\mathrm{D}}$ evaluates the model at $t > s$. This forces the network to generalize from $s=t$ to $t>s$ via continuity. To alleviate this, we instead match a small Euler-Maruyama step (with $t>s$) of the ground truth SDE rather than matching coefficients. This can be shown to result in a weighted coefficient matching objective (see Lemma B.1).

The general training algorithm is given in Algorithm 1. This requires a ground truth for $\boldsymbol{f}$ and $\boldsymbol{g}$ which can be constructed based on the task; for diffusion this is obtained by the reverse SDE with $\boldsymbol{f}(t, \boldsymbol{X}_t)$ derived from the conditional score – resulting in a simulation-free objective (see Appendix B.2).

Algorithm 1 Strong Stochastic Flow Map Training

Require: Batch size M, split $\eta \in (0,1)$, polynomial degree N, threshold Δt, EMA decay $\beta \in (0,1)$

repeat
  Sample $s \sim \mathcal{U}[0,1]$, $\boldsymbol{X}_s \sim p_s$ ▷ Simulate or interpolate
  ▷ Matching objective (batch size $\lfloor \eta M \rfloor$)
  Sample $t \sim \mathcal{U}[s,\ s{+}\Delta t]$ and $\boldsymbol{I}_{s,t}^{(N)} \sim$ (12)
  $\hat{\boldsymbol{X}}_t \leftarrow \boldsymbol{X}_s + \boldsymbol{f}(s,\boldsymbol{X}_s)(t{-}s) + \boldsymbol{g}(s)(\boldsymbol{W}_t{-}\boldsymbol{W}_s)$
  $\mathcal{L}_{\boldsymbol{f},\boldsymbol{g}} \leftarrow (t{-}s)^{-1} \left\| \hat{\boldsymbol{X}}_t - \boldsymbol{\Psi}_{s,t}^{\theta}(\boldsymbol{X}_s, \boldsymbol{I}_{s,t}^{(N)}) \right\|^2$
  ▷ Distillation objective (batch size $M - \lfloor \eta M \rfloor$)
  Sample $t \sim \mathcal{U}[s{+}\Delta t,\ 1]$ and set $u \leftarrow \frac{1}{2}(s{+}t)$
  Sample $\boldsymbol{I}_{s,u}^{(N)}, \boldsymbol{I}_{u,t}^{(N)} \sim$ (12) and compute $\boldsymbol{I}_{s,t}^{(N)}$ via (22)
  $\boldsymbol{X}_{\text{tgt}} \leftarrow \texttt{stopgrad}\big(\boldsymbol{\Psi}_{s,t}^{\theta}(\boldsymbol{X}_s, \boldsymbol{I}_{s,t}^{(N)})\big)$
  $\boldsymbol{X}_{\text{pred}} \leftarrow \boldsymbol{\Psi}_{u,t}^{\theta}\big(\boldsymbol{\Psi}_{s,u}^{\theta}(\boldsymbol{X}_s, \boldsymbol{I}_{s,u}^{(N)}),\ \boldsymbol{I}_{u,t}^{(N)}\big)$
  $\mathcal{L}_{\text{D}} \leftarrow (t{-}s)^{-1} \left\| \boldsymbol{X}_{\text{tgt}} - \boldsymbol{X}_{\text{pred}} \right\|^2$
  ▷ Update
  $\theta \leftarrow \theta - \lambda \nabla_\theta (\mathcal{L}_{\boldsymbol{f},\boldsymbol{g}} + \mathcal{L}_{\text{D}})$
  $\hat{\theta} \leftarrow \beta\hat{\theta} + (1{-}\beta)\,\theta$
until θ converges =0

4. Experiments

We consider three experiments that demonstrate the properties and performance of the SSFM model. Firstly, we ablate the algorithmic properties on a non-linear SDE and verify the effectiveness of the polynomial approximation to Brownian motion. Second, we apply the model to CIFAR-10 image generation and show that SSFMs outperform previous deterministic and stochastic flow maps. Third, we consider generation of equilibrium molecular conformations on the Alanine-Dipeptide dataset, where the SSFM model is capable of generating accurate samples in as few as two network evaluations.

4.1. Non-linear SDE

We first investigate the performance of the SSFM model on a toy system: a non-linear drift, additive-noise SDE, given by

$$\mathrm{d}\boldsymbol{X}_t = [\boldsymbol{X}_t - \boldsymbol{X}_t^3]\,\mathrm{d}t + \sqrt{\beta_t}\,\mathrm{d}\boldsymbol{W}_t, \tag{17}$$

where β_t is a linear interpolation between $\beta_{\text{min}} = 0.1$ and $\beta_{\text{max}} = 20$. This system is intended to mimic the variance preserving reverse diffusion SDE. The SSFM is learned via Algorithm 1 with the ground truth constructed from (17).

In Figure 2, we ablate the SSFM accuracy as a function of the polynomial degree. As the number of coefficients passed to the model increases, the strong convergence error decreases; this is most pronounced at larger step sizes. It can also been seen that the accuracy gained by each additional coefficient added diminishes, but is far from saturated at $N = 4$.

In Figure 3, we show the underlying 4-th degree polynomial expansion of Brownian motion and the learned flow map evaluated at 16-steps. We see that the SSFM is capable

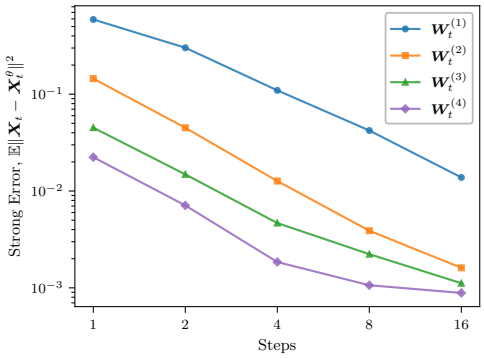

Figure 2. Strong error of the SSFM as a function of the polynomial degree.

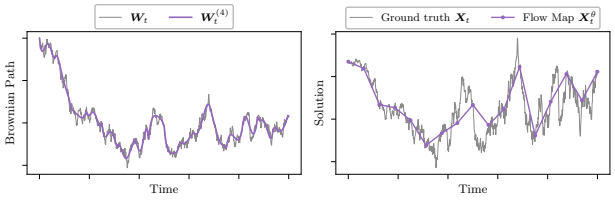

Figure 3. Left: Ground-truth Brownian path $\boldsymbol{W}_t$ and 4-th degree polynomial approximation $\boldsymbol{W}_t^{(4)}$ over 16 intervals. *Right*: Ground-truth SDE solution driven by $\boldsymbol{W}_t$ and learned 16-step flow map driven by $\boldsymbol{W}_t^{(4)}$.

Table 2. FID (↓) on CIFAR-10 across NFE step counts.

Method	NFE 2	4	8	16
Deterministic flows				
Consistency training (Song, Dhariwal, et al., 2023)	5.83	—	—	—
Flow map (LSD) (Boffi et al., 2025)	**4.37**	**3.34**	3.33	3.57
Flow map (PSD-M) (Boffi et al., 2025)	8.43	5.96	5.07	4.64
Flow map (PSD-U) (Boffi et al., 2025)	7.95	6.03	5.32	5.16
Flow map (Holderrieth, Chen, et al., 2026)	4.60	4.18	4.88	—
Weak stochastic flows				
GLASS (Holderrieth, Singer, et al., 2026)	157.55	39.47	11.60	—
Diamond Map (Holderrieth, Chen, et al., 2026)	5.80	5.80	6.73	—
Strong stochastic flows				
SSFM (Ours)	4.93	3.49	**3.29**	**3.35**

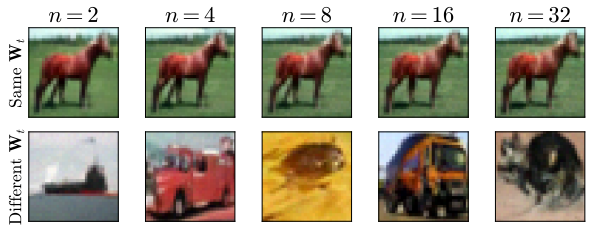

Figure 4. Top: SSFM with fixed $\boldsymbol{X}_0$ and same $\boldsymbol{W}_t$ across step counts. *Bottom*: SSFM with fixed $\boldsymbol{X}_0$, different $\boldsymbol{W}_t$ across step counts.

of accurately coarsening the SDE dynamics. We include additional plots for this system in Appendix D.1.

4.2. Image generation

We demonstrate the strong stochastic flow map construction for image generation on the CIFAR-10 dataset (Krizhevsky, Hinton, et al., 2009). The training procedure follows Algorithm 1, where the ground truth SDE is obtained via the reverse time variance preserving diffusion SDE (Song, Sohl-Dickstein, et al., 2021). See Appendix B.2 for a description of this SDE. The $\boldsymbol{f}^\theta, \boldsymbol{g}^\theta$ networks are independently parameterized with the EDM2 architecture (Karras et al., 2024). The number of polynomial coefficients was chosen to be $N = 3$.

The results can be seen in Table 2. The strong stochastic flow map models outperform the weak formulations across all step sizes considered. Additionally, under the strong construction we find that stochastic flow maps become competitive with deterministic flow maps and often outperform.

We verify the strong convergence property of the SSFM in Figure 4. Here, we see that for a fixed initial condition $\boldsymbol{X}_0$ and the same Brownian path $\boldsymbol{W}_t$, the resulting sample is the same for all step counts. Further, the stochasticity of the flow map is retained; for the same $\boldsymbol{X}_0$ with different $\boldsymbol{W}_t$, we obtain different generated samples.

4.3. Molecular systems

We demonstrate the strong stochastic flow map on the molecular system Alanine Dipeptide (ALDP) (Plainer et al., 2025). See Appendix D.3 for full details. We compare SSFM to regular diffusion baselines established in (Plainer et al., 2025). The results can be seen in Table 3. As expected, the SSFM enables markedly more efficient sampling than the diffusion baselines for step counts < 1000. As Steps $\to 1000$, the methods converge to achieving comparable performance. Notably, the SSFM at 100 steps is competitive with the diffusion baselines at 1000 steps, a factor 10 reduction in NFEs.

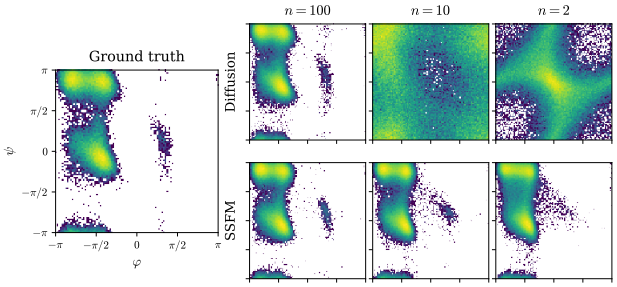

Figure 5. Ramachandran plots for Alanine-Dipeptide showing ground truth data, diffusion mixture baseline and SSFM prediction across step counts.

For illustration, the dihedral angles of the SSFM generated samples are shown in Figure 5. The SSFM samples accurate molecular configurations and is able to capture all modes even at just 2 coarse steps. In comparison, the diffusion baseline is unable to sample accurate configurations with fewer than 100 steps.

Table 3. PMF squared error (↓) and JS divergence (↓) on Alanine-Dipeptide across step counts.

		NFE					
		2	4	10	20	100	1000
PMF Error	Two-for-One	16.682	15.714	11.410	3.564	0.087	0.068
	Diffusion	16.728	15.667	11.347	3.400	0.084	0.066
	Mixture	22.070	17.266	11.556	3.394	0.078	**0.058**
	Fokker-Planck	15.475	15.709	11.345	3.498	0.092	0.069
	Both	21.830	17.094	11.571	3.393	0.087	0.065
	SSFM	**0.235**	**0.168**	**0.101**	**0.089**	**0.067**	0.062
JS ($\times 10^{-2}$)	Two-for-One	48.188	48.569	38.476	16.783	0.813	0.665
	Diffusion	48.735	48.377	38.309	16.293	0.787	0.618
	Mixture	52.432	49.500	38.709	16.279	0.770	0.609
	Fokker-Planck	47.157	48.250	38.307	16.569	0.836	0.638
	Both	52.340	49.270	38.660	16.370	0.830	0.640
	SSFM	**1.990**	**1.480**	**0.950**	**0.860**	**0.640**	**0.590**

5. Conclusion

In this work, we have introduced a novel theoretical framework for learning the Itô map to any additive-noise SDE. This framework was used to construct a class of flow maps, termed STRONG STOCHASTIC FLOW MAPS (SSFMs), that approximate the Itô map to compute a strong solution to additive-noise SDEs. On image generation experiments, this parameterization was shown to outperform both deterministic and stochastic flow map models. When applied to sampling molecular conformations, the model obtained accurate results in as few as two network evaluations and matched the performance of current diffusion based generative models using 10 times fewer steps. We expect this work to open up further improvements in reward alignment of generative models via pathwise estimators, and to accelerate molecular simulation and generative modeling tasks.

References

Albergo, M., Boffi, N. M., & Vanden-Eijnden, E. (2025). Stochastic interpolants: A unifying framework for flows and diffusions. *Journal of Machine Learning Research*, *26*(209), 1–80 (cit. on pp. 2, 3).

Anderson, B. D. (1982). Reverse-time diffusion equation models. *Stochastic Processes and their Applications*, *12*(3), 313–326 (cit. on p. 3).

Blasingame, Z. W., & Liu, C. (2026). Rex: A family of reversible exponential (stochastic) runge-kutta solvers. *Forty-third International Conference on Machine Learning*. https://openreview.net/forum?id=7pQIzVNctu (cit. on p. 21).

Blattmann, A., Rombach, R., Ling, H., Dockhorn, T., Kim, S. W., Fidler, S., & Kreis, K. (2023). Align your latents: High-resolution video synthesis with latent diffusion models. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 22563–22575 (cit. on p. 1).

Boffi, N. M., Albergo, M. S., & Vanden-Eijnden, E. (2024). Flow map matching. *arXiv preprint arXiv:2406.07507* (cit. on pp. 1, 3).

Boffi, N. M., Albergo, M. S., & Vanden-Eijnden, E. (2025). How to build a consistency model: Learning flow maps via self-distillation. *arXiv preprint arXiv:2505.18825* (cit. on pp. 3, 7, 21).

Chen, K.-T. (1954). Iterated integrals and exponential homomorphisms. *Proceedings of the London Mathematical Society*, *3*(1), 502–512 (cit. on p. 4).

Chen, K.-T. (1957). Integration of paths, geometric invariants and a generalized baker-hausdorff formula. *Annals of Mathematics*, *65*(1), 163–178 (cit. on p. 4).

Chen, R. T. Q., Rubanova, Y., Bettencourt, J., & Duvenaud, D. K. (2018). Neural ordinary differential equations. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, & R. Garnett (Eds.), *Advances in neural information processing systems* (Vol. 31). Curran Associates, Inc. https://proceedings.neurips.cc/paper_files/paper/2018/file/69386f6bb1dfed68692a24c8686939b9-Paper.pdf (cit. on pp. 1, 2).

Foster, J., Lyons, T., & Oberhauser, H. (2020). An optimal polynomial approximation of brownian motion. *SIAM Journal on Numerical Analysis*, *58*(3), 1393–1421 (cit. on pp. 4, 5, 17, 18).

Frans, K., Hafner, D., Levine, S., & Abbeel, P. (2025). One step diffusion via shortcut models. *The Thirteenth International Conference on Learning Representations*. https://openreview.net/forum?id=OlzB6LnXcS (cit. on pp. 3, 21).

Friz, P. K., & Victoir, N. B. (2010). *Multidimensional stochastic processes as rough paths: Theory and applications* (Vol. 120). Cambridge University Press. (Cit. on pp. 3, 17).

Garsia, A. M., Rodemich, E., Rumsey, H., & Rosenblatt, M. (1970). A real variable lemma and the continuity of paths of some gaussian processes. *Indiana University Mathematics Journal*, *20*(6), 565–578. Retrieved May 6, 2026, from http://www.jstor.org/stable/24890119 (cit. on p. 17).

Geng, Z., Deng, M., Bai, X., Kolter, J. Z., & He, K. (2025). Mean flows for one-step generative modeling. *The Thirty-ninth Annual Conference on Neural Information Processing Systems*. https://openreview.net/forum?id=uWj4s7rMnR (cit. on pp. 1, 3, 21).

Geng, Z., Lu, Y., Wu, Z., Shechtman, E., Kolter, J. Z., & He, K. (2025). Improved mean flows: On the challenges of fastforward generative models. *arXiv preprint arXiv:2512.02012* (cit. on pp. 1, 21).

Gonzalez, M., Fernandez, N., Tran, T. V. D., Gherbi, E., Hajri, H., & Masmoudi, N. (2023). SEEDS: Exponential SDE solvers for fast high-quality sampling from diffusion models. *Thirty-seventh Conference on Neural Information Processing Systems*. https://openreview.net/forum?id=V6IgkYKD8P (cit. on pp. 1, 21).

Habermann, K. (2021). A semicircle law and decorrelation phenomena for iterated kolmogorov loops. *Journal of the London Mathematical Society*, *103*(2), 558–586. https://doi.org/10.1112/jlms.12384 (cit. on pp. 4, 5).

Heek, J., Hoogeboom, E., & Salimans, T. (2024). Multistep consistency models. https://openreview.net/forum?id=d7DZRNe2xG (cit. on pp. 1, 3).

Ho, J., Jain, A., & Abbeel, P. (2020). Denoising diffusion probabilistic models. In H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, & H. Lin (Eds.), *Advances in neural information processing systems* (pp. 6840–6851, Vol. 33). Curran Associates, Inc. https://proceedings.neurips.cc/paper_files/paper/2020/file/4c5bcfec8584af0d967f1ab10179ca4b-Paper.pdf (cit. on p. 1).

Holderrieth, P., Chen, D., Eyring, L., Shah, I., Anantharaman, G., He, Y., Akata, Z., Jaakkola, T., Boffi, N. M., & Simchowitz, M. (2026). Diamond maps: Efficient reward alignment via stochastic flow maps. *arXiv preprint arXiv:2602.05993* (cit. on pp. 1, 3, 7, 21).

Holderrieth, P., Havasi, M., Yim, J., Shaul, N., Gat, I., Jaakkola, T., Karrer, B., Chen, R. T. Q., & Lipman, Y. (2025). Generator matching: Generative modeling with arbitrary markov processes. *The Thirteenth International Conference on Learning Representations*. https://openreview.net/forum?id=RuP17cJtZo (cit. on p. 21).

Holderrieth, P., Singer, U., Jaakkola, T., Chen, R. T. Q., Lipman, Y., & Karrer, B. (2026). GLASS flows: Efficient inference for reward alignment of flow and diffusion models. *The Fourteenth International Conference on Learning Representations*. https://openreview.net/forum?id=vH7OAPZ2dR (cit. on pp. 3, 7).

Jelinčič, A., Foster, J., & Kidger, P. (2024). Single-seed generation of brownian paths and integrals for adaptive and high order sde solvers. *arXiv preprint arXiv:2405.06464* (cit. on p. 21).

Jiang, L., Ge, W., Cariou-Kotlarek, N., Yi, M., Chen, P.-Y., Yang, L., Buet-Golfouse, F., Mittal, G., & Ni, H. (2025). Sig-deg for distillation: Making diffusion models faster and lighter. *arXiv preprint arXiv:2508.16939* (cit. on p. 21).

Karras, T., Aittala, M., Lehtinen, J., Hellsten, J., Aila, T., & Laine, S. (2024). Analyzing and improving the training dynamics of diffusion models. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 24174–24184 (cit. on pp. 7, 23).

Kidger, P. (2022). *On neural differential equations* [Ph.D. thesis]. Oxford University [Available at https://arxiv.org/abs/2202.02435]. (Cit. on pp. 1, 21).

Kidger, P., Foster, J., Li, X., & Lyons, T. (2021). Efficient and accurate gradients for neural SDEs. In A. Beygelzimer, Y. Dauphin, P. Liang, & J. W. Vaughan (Eds.), *Advances in neural information processing systems*. https://openreview.net/forum?id=b2bkE0Qq8Ya (cit. on p. 2).

Kim, D., Lai, C.-H., Liao, W.-H., Murata, N., Takida, Y., Uesaka, T., He, Y., Mitsufuji, Y., & Ermon, S. (2024). Consistency trajectory models: Learning probability flow ODE trajectory of diffusion. *The Twelfth International Conference on Learning Representations*. https://openreview.net/forum?id=ymjI8feDTD (cit. on pp. 1, 3).

Köhler, J., Krämer, A., & Noé, F. (2021). Smooth normalizing flows. *Advances in Neural Information Processing Systems*, *34*, 2796–2809 (cit. on p. 23).

Krizhevsky, A., Hinton, G., et al. (2009). Learning multiple layers of features from tiny images (cit. on p. 7).

Li, X., Wong, T.-K. L., Chen, R. T. Q., & Duvenaud, D. K. (2020, August). Scalable gradients and variational inference for stochastic differential equations. In C. Zhang, F. Ruiz, T. Bui, A. B. Dieng, & D. Liang (Eds.), *Proceedings of the 2nd symposium on advances in approximate bayesian inference* (pp. 1–28, Vol. 118). PMLR. https://proceedings.mlr.press/v118/li20a.html (cit. on p. 2).

Lipman, Y., Chen, R. T. Q., Ben-Hamu, H., Nickel, M., & Le, M. (2023). Flow matching for generative modeling. *The Eleventh International Conference on Learning Representations*. https://openreview.net/forum?id=PqvMRDCJT9t (cit. on pp. 1, 2).

Lipman, Y., Havasi, M., Holderrieth, P., Shaul, N., Le, M., Karrer, B., Chen, R. T., Lopez-Paz, D., Ben-Hamu, H., & Gat, I. (2024). Flow matching guide and code. (Cit. on p. 2).

Liu, X., Gong, C., & Liu, Q. (2023). Flow straight and fast: Learning to generate and transfer data with rectified flow. *The Eleventh International Conference on Learning Representations*. https://openreview.net/forum?id=XVjTT1nw5z (cit. on pp. 1, 2).

Lu, C., Zhou, Y., Bao, F., Chen, J., Li, C., & Zhu, J. (2022). DPM-solver: A fast ODE solver for diffusion probabilistic model sampling in around 10 steps. In A. H. Oh, A. Agarwal, D. Belgrave, & K. Cho (Eds.), *Advances in neural information processing systems*. https://openreview.net/forum?id=2uAaGwlP_V (cit. on p. 1).

Lyons, T. J. (1998). Differential equations driven by rough signals. *Revista Matemática Iberoamericana*, *14*(2), 215–310 (cit. on pp. 3, 4).

Maoutsa, D., Reich, S., & Opper, M. (2020). Interacting particle solutions of fokker–planck equations through gradient–log–density estimation. *Entropy*, *22*(8), 802 (cit. on p. 3).

Nie, S., Guo, H. A., Lu, C., Zhou, Y., Zheng, C., & Li, C. (2024). The blessing of randomness: SDE beats ODE in general diffusion-based image editing. *The Twelfth International Conference on Learning Representations*. https://openreview.net/forum?id=DesYwmUG00 (cit. on p. 3).

Oh, Y., Lim, D., & Kim, S. (2024). Stable neural stochastic differential equations in analyzing irregular time series data. *The Twelfth International Conference on Learning Representations*. https://openreview.net/forum?id=4VIgNuQ1pY (cit. on p. 1).

Øksendal, B. (2003, July). *Stochastic differential equations: An introduction with applications*. Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-14394-6 (cit. on pp. 1, 3, 4, 12, 15).

Passaro, R., Blasingame, Z. W., Bronstein, M. M., & Tong, A. (2026). Stochastic few-step models. *ICLR 2026 2nd Workshop on Deep Generative Model in Machine Learning: Theory, Principle and Efficacy*. https://openreview.net/forum?id=nmczKNW73P (cit. on pp. 1, 3, 21).

Peluchetti, S. (2021). Non-denoising forward-time diffusions. (Cit. on pp. 1, 2).

Plainer, M., Wu, H., Klein, L., Günnemann, S., & Noe, F. (2025). Consistent sampling and simulation: Molecular dynamics with energy-based diffusion models. *The Thirty-ninth Annual Conference on Neural Information Processing Systems* (cit. on pp. 7, 23).

Poli, M., Massaroli, S., Yamashita, A., Asama, H., & Park, J. (2020). Hypersolvers: Toward fast continuous-depth models. In H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, & H. Lin (Eds.), *Advances in neural information processing systems* (pp. 21105–21117, Vol. 33). Curran Associates, Inc. https://proceedings.neurips.cc/paper_files/paper/2020/file/f1686b4badcf28d33ed632036c7ab0b8-Paper.pdf (cit. on p. 21).

Potaptchik, P., Saravanan, A., Mammadov, A., Prat, A., Albergo, M. S., & Teh, Y. W. (2026). Meta flow maps enable scalable reward alignment. *arXiv preprint arXiv:2601.14430* (cit. on pp. 1, 3, 21).

Rector-Brooks, J., Lambert, T., Skreta, M., Roth, D., Long, Y., Li, Z.-Q., Zhang, X., Cretu, M., Li, F.-Z., Ganapathy, T., Jin, E., Bose, A. J., Yang, J., Neklyudov, K., Bengio, Y., Tong, A., Arnold, F. H., & Liu, C.-H. (2026). General multimodal protein design enables dna-encoding of chemistry. https://arxiv.org/abs/2604.05181 (cit. on p. 1).

Rehman, D., Akhound-Sadegh, T., Gazizov, A., Bengio, Y., & Tong, A. (2026). FALCON: Few-step accurate likelihoods for continuous flows. *The Fourteenth International Conference on Learning Representations*. https://openreview.net/forum?id=FbssShlI4N (cit. on p. 1).

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., & Ommer, B. (2022). High-resolution image synthesis with latent diffusion models. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 10684–10695 (cit. on p. 1).

Sabour, A., Fidler, S., & Kreis, K. (2025). Align your flow: Scaling continuous-time flow map distillation. *The Thirty-ninth Annual Conference on Neural Information Processing Systems*. https://openreview.net/forum?id=pzHuesCvcO (cit. on pp. 1, 3).

Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N., & Ganguli, S. (2015). Deep unsupervised learning using nonequilibrium thermodynamics. *International conference on machine learning*, 2256–2265 (cit. on p. 1).

Song, Y., & Dhariwal, P. (2024). Improved techniques for training consistency models. *The Twelfth International Conference on Learning Representations*. https://openreview.net/forum?id=WNzy9bRDvG (cit. on p. 21).

Song, Y., Dhariwal, P., Chen, M., & Sutskever, I. (2023, 23–29 Jul). Consistency models. In A. Krause, E. Brunskill,

K. Cho, B. Engelhardt, S. Sabato, & J. Scarlett (Eds.), *Proceedings of the 40th international conference on machine learning* (pp. 32211–32252, Vol. 202). PMLR. https://proceedings.mlr.press/v202/song23a.html (cit. on pp. 1, 3, 7, 21).

Song, Y., Sohl-Dickstein, J., Kingma, D. P., Kumar, A., Ermon, S., & Poole, B. (2021). Score-based generative modeling through stochastic differential equations. *International Conference on Learning Representations*. https://openreview.net/forum?id=PxTIG12RRHS (cit. on pp. 1, 3, 7).

Tong, A., FATRAS, K., Malkin, N., Huguet, G., Zhang, Y., Rector-Brooks, J., Wolf, G., & Bengio, Y. (2024). Improving and generalizing flow-based generative models with minibatch optimal transport [Expert Certification]. *Transactions on Machine Learning Research*. https://openreview.net/forum?id=CD9Snc73AW (cit. on p. 2).

Walker, B., McLeod, A. D., Qin, T., Cheng, Y., Li, H., & Lyons, T. (2024). Log neural controlled differential equations: The lie brackets make a difference. *Forty-first International Conference on Machine Learning*. https://openreview.net/forum?id=0tYrMtQyPT (cit. on p. 1).

Watson, J. L., Juergens, D., Bennett, N. R., Trippe, B. L., Yim, J., Eisenach, H. E., Ahern, W., Borst, A. J., Ragotte, R. J., Milles, L. F., et al. (2023). De novo design of protein structure and function with rfdiffusion. *Nature*, *620*(7976), 1089–1100 (cit. on p. 1).

Zhang, Q., & Chen, Y. (2023). Fast sampling of diffusion models with exponential integrator. *The Eleventh International Conference on Learning Representations*. https://openreview.net/forum?id=Loek7hfb46P (cit. on pp. 1, 21).

Appendices

A. Proofs

In this section we prove the results provided in the main text. Throughout, we assume that the stochastic differential equations studied satisfy the following assumption.

Assumption A.1 (Unique strong solution)**.** *We assume that the additive-noise stochastic differential equation given by*

$$d\boldsymbol{X}_t = \boldsymbol{f}(t, \boldsymbol{X}_t)\,\mathrm{d}t + \boldsymbol{g}(t)\,\mathrm{d}\boldsymbol{W}_t,$$

where $\boldsymbol{f} : \mathbb{R} \times \mathbb{R}^d \to \mathbb{R}^d, \boldsymbol{g} : \mathbb{R} \to \mathbb{R}^{d\times w}$ *with* $\boldsymbol{W}_t \in \mathbb{R}^w$ *and initial condition* $\boldsymbol{X}_0$ *with finite second moment, has the following properties such that a unique continuous solution* $\boldsymbol{X}_t$ *exists. Let* $\boldsymbol{f}, \boldsymbol{g}$ *be measurable functions satisfying*

$$|\boldsymbol{f}(t, \boldsymbol{x})| + |\boldsymbol{g}(t)| \leq C(1 + |\boldsymbol{x}|)$$
$$|\boldsymbol{f}(t, \boldsymbol{x}) - \boldsymbol{f}(t, \boldsymbol{y})| \leq D|\boldsymbol{x} - \boldsymbol{y}|$$

for constants C, D*. See Øksendal (2003) for a full discussion on the existence and unique of strong solutions to SDEs.*

A.1. Constructing the Itô map

A.1.1. Proof of Lemma 3.1

We first prove a simple lemma that relates the Itô map to the SDE.

Lemma A.1 (Stochastic Lagrangian equation)**.** *Let* $\boldsymbol{\Psi}_{s,t}(\boldsymbol{X}_s, \boldsymbol{W}_{[s,t]})$ *denote the Itô map for (6). Then,*

$$\mathrm{d}\boldsymbol{\Psi}_{s,t}(\boldsymbol{X}_s, \boldsymbol{W}_{[s,t]}) = \boldsymbol{f}(t, \boldsymbol{\Psi}_{s,t}(\boldsymbol{X}_s, \boldsymbol{W}_{[s,t]}))\mathrm{d}t + \boldsymbol{g}(t)\mathrm{d}\boldsymbol{W}_t. \tag{18}$$

Proof. Consider the derivative of the stochastic flow map,

$$\begin{aligned}\mathrm{d}\boldsymbol{\Psi}_{s,t}(\boldsymbol{X}_s, \boldsymbol{W}_{[s,t]}) &= \mathrm{d}\boldsymbol{X}_t, \\ &= \boldsymbol{f}(t, \boldsymbol{X}_t)\mathrm{d}t + \boldsymbol{g}(t)\mathrm{d}\boldsymbol{W}_t, \\ &= \boldsymbol{f}(t, \boldsymbol{\Psi}_{s,t}(\boldsymbol{X}_s, \boldsymbol{W}_{[s,t]}))\mathrm{d}t + \boldsymbol{g}(t)\mathrm{d}\boldsymbol{W}_t.\end{aligned}$$

□

Now, we are able to prove Lemma 3.1 which we restate here.

Lemma 3.1 (Tangent Condition)**.** *Let $\boldsymbol{\Psi}_{s,t}(\boldsymbol{X}_s, \boldsymbol{W}_{[s,t]})$ denote the Itô map for (6). Then,*

$$\lim_{s\to t} \mathrm{d}\boldsymbol{\Psi}_{s,t}(\boldsymbol{X}_s, \boldsymbol{W}_{[s,t]}) = \boldsymbol{f}(t, \boldsymbol{X}_t)\ \mathrm{d}t + \boldsymbol{g}(t)\ \mathrm{d}\boldsymbol{W}_t. \tag{7}$$

Proof. By Lemma A.1 we have that the Itô map satisfies the stochastic Lagrangian equation. Taking the limit as $s \to t$, then given the continuity of the Itô map we have

$$\begin{aligned}\lim_{s\to t} \mathrm{d}\boldsymbol{\Psi}_{s,t}(\boldsymbol{X}_s, \boldsymbol{W}_{[s,t]}) &= \lim_{s\to t} \boldsymbol{f}(t, \boldsymbol{\Psi}_{s,t}(\boldsymbol{X}_s, \boldsymbol{W}_{[s,t]}))\mathrm{d}t + \boldsymbol{g}(t)\mathrm{d}\boldsymbol{W}_t \\ &= \boldsymbol{f}(t, \boldsymbol{\Psi}_{t,t}(\boldsymbol{X}_t, \boldsymbol{W}_{[t,t]}))\mathrm{d}t + \boldsymbol{g}(t)\mathrm{d}\boldsymbol{W}_t \\ &= \boldsymbol{f}(t, \boldsymbol{X}_t)\mathrm{d}t + \boldsymbol{g}(t)\mathrm{d}\boldsymbol{W}_t\end{aligned}$$

□

A.1.2. Proof of Proposition 3.2

Next, we prove Proposition 3.2.

Proposition 3.2 (Strong Stochastic Flow Map)**.** *Consider an Euler-Maruyama parameterization of the Itô map,*

$$\begin{aligned}\boldsymbol{\Psi}_{s,t}(\boldsymbol{X}_s, \boldsymbol{W}_{[s,t]}) = \boldsymbol{X}_s &+ \boldsymbol{f}_{s,t}(\boldsymbol{X}_s, \boldsymbol{W}_{[s,t]})(t-s) \\ &+ \boldsymbol{g}_{s,t}(\boldsymbol{W}_{[s,t]})(\boldsymbol{W}_t - \boldsymbol{W}_s),\end{aligned} \tag{8}$$

where $\boldsymbol{f}, \boldsymbol{g}$ are twice continuously differentiable, Lipschitz in both time arguments, satisfy the conditions that $\boldsymbol{f}_{t,t}(\boldsymbol{X}_t, \boldsymbol{W}_{[s,t]}) = \boldsymbol{f}_{t,t}(\boldsymbol{X}_t)$, and $\boldsymbol{g}_{t,t}(\boldsymbol{W}_{[s,t]}) = \boldsymbol{g}_{t,t}$ (i.e. the coefficients are independent of $\boldsymbol{W}_{[s,t]}$ for $s = t$). Then, $\boldsymbol{\Psi}_{s,t}$ satisfies the tangent condition (7) if and only if

$$\boldsymbol{f}_{t,t}(x_t) = \boldsymbol{f}(t, x_t), \quad \boldsymbol{g}_{t,t} = \boldsymbol{g}(t).$$

Proof. Consider the application of Itô's lemma to $\boldsymbol{\Psi}_{s,t}(\boldsymbol{X}_s, \boldsymbol{W}_{[s,t]})$, given by

$$\begin{aligned}\mathrm{d}\boldsymbol{\Psi}_{s,t}(\boldsymbol{X}_s, \boldsymbol{W}_{[s,t]}) &= \left(\partial_t \boldsymbol{\Psi}_{s,t}(\boldsymbol{X}_s, \boldsymbol{W}_{[s,t]}) + \frac{1}{2}\partial^2_{\boldsymbol{W}_t} \boldsymbol{\Psi}_{s,t}(\boldsymbol{X}_s, \boldsymbol{W}_{[s,t]})\right)\mathrm{d}t + \partial_{\boldsymbol{W}_t} \boldsymbol{\Psi}_{s,t}(\boldsymbol{X}_s, \boldsymbol{W}_{[s,t]})\mathrm{d}\boldsymbol{W}_t, \\ &= \Big(\boldsymbol{f}_{s,t}(\boldsymbol{X}_s, \boldsymbol{W}_{[s,t]}) + (t-s)\partial_t \boldsymbol{f}_{s,t}(\boldsymbol{X}_s, \boldsymbol{W}_{[s,t]}) + (\boldsymbol{W}_t - \boldsymbol{W}_s)\partial_t \boldsymbol{g}_{s,t}(\boldsymbol{W}_{[s,t]}) \\ &+ \frac{1}{2}(t-s)\partial^2_{\boldsymbol{W}_t} \boldsymbol{f}_{s,t}(\boldsymbol{X}_s, \boldsymbol{W}_{[s,t]}) + \frac{1}{2}(\boldsymbol{W}_t - \boldsymbol{W}_s)\partial^2_{\boldsymbol{W}_t} \boldsymbol{g}_{s,t}(\boldsymbol{W}_{[s,t]}) + \partial_{\boldsymbol{W}_t} \boldsymbol{g}_{s,t}(\boldsymbol{W}_{[s,t]})\Big)\mathrm{d}t \\ &+ \Big((t-s)\partial_{\boldsymbol{W}_t} \boldsymbol{f}_{s,t}(\boldsymbol{X}_s, \boldsymbol{W}_{[s,t]}) + \boldsymbol{g}_{s,t}(\boldsymbol{W}_{[s,t]}) + (\boldsymbol{W}_t - \boldsymbol{W}_s)\partial_{\boldsymbol{W}_t} \boldsymbol{g}_{s,t}(\boldsymbol{W}_{[s,t]})\Big)\mathrm{d}\boldsymbol{W}_t.\end{aligned}$$

Now, taking the limit $s \to t$, we obtain

$$\lim_{s\to t} \mathrm{d}\boldsymbol{\Psi}_{s,t}(\boldsymbol{X}_s, \boldsymbol{W}_{[s,t]}) = \Big(\boldsymbol{f}_{t,t}(\boldsymbol{X}_t, \boldsymbol{W}_{[t,t]}) + \partial_{\boldsymbol{W}_t} \boldsymbol{g}_{t,t}(\boldsymbol{W}_{[t,t]})\Big)\mathrm{d}t + \boldsymbol{g}_{t,t}\mathrm{d}\boldsymbol{W}_t.$$

Finally, using the conditions that $\boldsymbol{f}_{t,t}(\boldsymbol{X}_t, \boldsymbol{W}_{[t,t]}) = \boldsymbol{f}_{t,t}(\boldsymbol{X}_t)$ and $\boldsymbol{g}_{t,t}(\boldsymbol{W}_{[t,t]}) = \boldsymbol{g}_{t,t}$, this reduces to

$$\lim_{s \to t} \mathrm{d}\boldsymbol{\Psi}_{s,t}(\boldsymbol{X}_s, \boldsymbol{W}_{[s,t]}) = \boldsymbol{f}_{t,t}(\boldsymbol{X}_t)\mathrm{d}t + \boldsymbol{g}_{t,t}\mathrm{d}\boldsymbol{W}_t.$$

To satisfy the tangent condition (7) we must have $\boldsymbol{f}_{t,t}(\boldsymbol{X}_t) = \boldsymbol{f}(t, \boldsymbol{X}_t)$ and $\boldsymbol{g}_{t,t} = \boldsymbol{g}(t)$. Conversely, substituting $\boldsymbol{f}_{t,t}(\boldsymbol{X}_t) = \boldsymbol{f}(t, \boldsymbol{X}_t), \boldsymbol{g}_{t,t} = \boldsymbol{g}(t)$ recovers the tangent condition. □

A.1.3. Proof of Proposition 3.3

Now onto the proof that the strong stochastic flow map attains the Itô map when satisfying the tangent and semigroup conditions.

Proposition 3.3 (Semigroup condition)**.** *Let $\boldsymbol{\Psi}_{s,t}(\boldsymbol{X}_s, \boldsymbol{W}_{[s,t]})$ denote the strong stochastic flow map satisfying (7) and (8). Then $\boldsymbol{\Psi}_{s,t}(\boldsymbol{X}_s, \boldsymbol{W}_{[s,t]})$ is the Itô map if and only if the semigroup property (9) holds.*

Proof. First observe that the Itô map satisfies the semigroup condition,

$$\begin{aligned}\boldsymbol{\Psi}_{s,t}(\boldsymbol{X}_s, \boldsymbol{W}_{[s,t]}) &= \boldsymbol{\Psi}_{u,t}(\boldsymbol{\Psi}_{s,u}(\boldsymbol{X}_s, \boldsymbol{W}_{[s,u]}), \boldsymbol{W}_{[u,t]}) \\ &= \boldsymbol{\Psi}_{u,t}(\boldsymbol{X}_u, \boldsymbol{W}_{[u,t]}) \\ &= \boldsymbol{X}_t\end{aligned}$$

Next, we want to show the inverse implication. Let $\boldsymbol{\Psi}_{s,t}(\boldsymbol{X}_s, \boldsymbol{W}_{[s,t]})$ denote the strong stochastic flow map satisfying (7), (8) and (9). We show that this map is the Itô map for (6). This follows by considering the semigroup condition,

$$\begin{aligned}\boldsymbol{\Psi}_{s,t+h}(\boldsymbol{X}_s, \boldsymbol{W}_{[s,t+h]}) &= \boldsymbol{\Psi}_{t,t+h}(\boldsymbol{\Psi}_{s,t}(\boldsymbol{X}_s, \boldsymbol{W}_{[s,t]}), \boldsymbol{W}_{[t,t+h]}) \\ &= \boldsymbol{\Psi}_{s,t}(\boldsymbol{X}_s, \boldsymbol{W}_{[s,t]}) + h\boldsymbol{f}_{t,t+h}(\boldsymbol{\Psi}_{s,t}(\boldsymbol{X}_s, \boldsymbol{W}_{[s,t]}), \boldsymbol{W}_{[t,t+h]}) \\ &\quad + (\boldsymbol{W}_{t+h} - \boldsymbol{W}_t)\boldsymbol{g}_{t,t+h}(\boldsymbol{W}_{[t,t+h]})\end{aligned}$$

Define $\hat{\boldsymbol{X}}_t = \boldsymbol{\Psi}_{s,t}(\boldsymbol{X}_s, \boldsymbol{W}_{[s,t]})$ and re-arrange to get

$$\hat{\boldsymbol{X}}_{t+h} - \hat{\boldsymbol{X}}_t = h\boldsymbol{f}_{t,t+h}(\hat{\boldsymbol{X}}_t, \boldsymbol{W}_{[t,t+h]}) + (\boldsymbol{W}_{t+h} - \boldsymbol{W}_t)\boldsymbol{g}_{t,t+h}(\boldsymbol{W}_{[t,t+h]}).$$

Next consider taking the following sum over a partition $t = u_0 < u_1 < \cdots < u_n = t + h$,

$$\sum_i \hat{\boldsymbol{X}}_{u_{i+1}} - \hat{\boldsymbol{X}}_{u_i} = \sum_i (u_{i+1} - u_i)\boldsymbol{f}_{u_i,u_{i+1}}(\hat{\boldsymbol{X}}_{u_i}, \boldsymbol{W}_{[u_i,u_{i+1}]}) + \sum_i (\boldsymbol{W}_{u_{i+1}} - \boldsymbol{W}_{u_i})\boldsymbol{g}_{u_i,u_{i+1}}(\boldsymbol{W}_{[u_i,u_{i+1}]})$$

The LHS of this sum telescopes to $\hat{\boldsymbol{X}}_{t+h} - \hat{\boldsymbol{X}}_t$. We now take the limit of this partition with $\max_i(u_{i+1} - u_i) \to 0$.

Since we have $\boldsymbol{f}_{u,u}(\hat{\boldsymbol{X}}_u, \boldsymbol{W}_{[u,u]}) = \boldsymbol{f}_{u,u}(\hat{\boldsymbol{X}}_u)$ and $\boldsymbol{g}_{u,u}(\boldsymbol{W}_{[u,u]}) = \boldsymbol{g}_{u,u}$ by the independence assumption on (8), consider the following equivalent expression obtained by adding and subtracting the diagonal coefficients,

$$\begin{aligned}\hat{\boldsymbol{X}}_{t+h} - \hat{\boldsymbol{X}}_t = &\sum_i (u_{i+1} - u_i)\boldsymbol{f}_{u_i,u_i}(\hat{\boldsymbol{X}}_{u_i}) + \sum_i (u_{i+1} - u_i)\Big(\boldsymbol{f}_{u_i,u_{i+1}}(\hat{\boldsymbol{X}}_{u_i}, \boldsymbol{W}_{[u_i,u_{i+1}]}) - \boldsymbol{f}_{u_i,u_i}(\hat{\boldsymbol{X}}_{u_i})\Big) \\ &+ \sum_i (\boldsymbol{W}_{u_{i+1}} - \boldsymbol{W}_{u_i})\boldsymbol{g}_{u_i,u_i} + \sum_i (\boldsymbol{W}_{u_{i+1}} - \boldsymbol{W}_{u_i})\Big(\boldsymbol{g}_{u_i,u_{i+1}}(\boldsymbol{W}_{[u_i,u_{i+1}]}) - \boldsymbol{g}_{u_i,u_i}\Big)\end{aligned}$$

The two difference terms (in both $\boldsymbol{f}$ and $\boldsymbol{g}$) decay to zero as $\max_i(u_{i+1} - u_i) \to 0$. This follows from the Lipschitz-in-time assumption on $\boldsymbol{f}, \boldsymbol{g}$. Specifically, for the $\boldsymbol{f}$ residual we have

$$\begin{aligned}\Big|\sum_i (u_{i+1} - u_i)\Big(\boldsymbol{f}_{u_i,u_{i+1}}(\hat{\boldsymbol{X}}_{u_i}, \boldsymbol{W}_{[u_i,u_{i+1}]}) - \boldsymbol{f}_{u_i,u_i}(\hat{\boldsymbol{X}}_{u_i})\Big)\Big| &\leq L_f \sum_i |u_{i+1} - u_i||u_{i+1} - u_i|, \\ &\leq L_f \max_i(|u_{i+1} - u_i|) \sum_i |u_{i+1} - u_i|, \\ &\leq L_f h \max_i(|u_{i+1} - u_i|),\end{aligned}$$

which tends to zero as $\max_i(u_{i+1} - u_i) \to 0$. For the $\boldsymbol{g}$ residual, we have

$$\begin{aligned}\Big|\sum_i (\boldsymbol{W}_{u_{i+1}} - \boldsymbol{W}_{u_i})\Big(\boldsymbol{g}_{u_i,u_{i+1}}(\boldsymbol{W}_{[u_i,u_{i+1}]}) - \boldsymbol{g}_{u_i,u_i}\Big)\Big| &\leq L_g \sum_i |(\boldsymbol{W}_{u_{i+1}} - \boldsymbol{W}_{u_i})||u_{i+1} - u_i| \\ &\leq L_g \max_i(|\boldsymbol{W}_{u_{i+1}} - \boldsymbol{W}_{u_i}|) \sum_i |u_{i+1} - u_i|, \\ &= hL_g \max_i(|\boldsymbol{W}_{u_{i+1}} - \boldsymbol{W}_{u_i}|),\end{aligned}$$

which tends to zero as $\max_i(u_{i+1} - u_i) \to 0$ by the uniform continuity of $\boldsymbol{W}_t$. We are left with

$$\hat{\boldsymbol{X}}_{t+h} - \hat{\boldsymbol{X}}_t = \lim_{\max_i(u_{i+1}-u_i)\to 0} \sum_i (u_{i+1} - u_i)\boldsymbol{f}_{u_i,u_i}(\hat{\boldsymbol{X}}_{u_i}) + \sum_i (\boldsymbol{W}_{u_{i+1}} - \boldsymbol{W}_{u_i})\boldsymbol{g}_{u_i,u_i}.$$

Therefore, both Riemann and Itô (Stratonovich, equivalently) converge (Øksendal, 2003) to give

$$\begin{aligned}\hat{\boldsymbol{X}}_{t+h} &= \hat{\boldsymbol{X}}_t + \int_t^{t+h} \boldsymbol{f}_{u,u}(\hat{\boldsymbol{X}}_u)\mathrm{d}u + \int_t^{t+h} \boldsymbol{g}_{u,u}\mathrm{d}\boldsymbol{W}_u \\ &= \hat{\boldsymbol{X}}_t + \int_t^{t+h} \boldsymbol{f}(u, \hat{\boldsymbol{X}}_u)\mathrm{d}u + \int_t^{t+h} \boldsymbol{g}(u)\mathrm{d}\boldsymbol{W}_u,\end{aligned}$$

where $\boldsymbol{f}_{t,t}(\boldsymbol{X}_t) = \boldsymbol{f}(t, \boldsymbol{X}_t)$ and $\boldsymbol{g}_{t,t} = \boldsymbol{g}(t)$ follows from Proposition 3.2. Since this holds for any $t \geq s$ and $h \geq 0$, $\hat{\boldsymbol{X}}_t$ is a strong solution to the SDE with coefficients $\boldsymbol{f}(t, \boldsymbol{X}_t), \boldsymbol{g}(t)$. By uniqueness, this implies that $\boldsymbol{\Psi}_{s,t}(\boldsymbol{X}_s, \boldsymbol{W}_{[s,t]})$ is the Itô map for (6). □

A.1.4. Proof of Theorem 3.5

Given the results above, we are able to prove Theorem 3.5.

Theorem 3.5 (Self-distillation Objective)**.** *Let $\boldsymbol{\Psi}_{s,t}(\boldsymbol{X}_s, \boldsymbol{W}_{[s,t]})$ denote the Itô map for (6). Then, this map is given by the strong stochastic flow map in (8) where $\boldsymbol{v}_{s,t} = [\boldsymbol{f}_{s,t}, \boldsymbol{g}_{s,t}]$ is the unique global minimizer over $\hat{\boldsymbol{v}}$ of*

$$\mathcal{L}_{SD}(\hat{\boldsymbol{v}}) = \mathcal{L}_{\boldsymbol{f},\boldsymbol{g}}(\hat{\boldsymbol{v}}) + \mathcal{L}_D(\hat{\boldsymbol{v}}), \tag{14}$$

where

$$\begin{aligned}\mathcal{L}_{\boldsymbol{f},\boldsymbol{g}}(\hat{\boldsymbol{v}}) = \mathbb{E}_{t,\boldsymbol{X}_t}\Big[&\|\boldsymbol{f}(t, \boldsymbol{X}_t) - \hat{\boldsymbol{f}}_{t,t}(\boldsymbol{X}_t)\|_2^2 \\ &+ \|\boldsymbol{g}(t) - \hat{\boldsymbol{g}}_{t,t}\|_2^2\Big],\end{aligned} \tag{15}$$

$$\begin{aligned}\mathcal{L}_D(\hat{\boldsymbol{v}}) = \mathbb{E}_{s,u,t,\boldsymbol{X}_s,\boldsymbol{W}_{[s,t]}}\Big[&\|\hat{\boldsymbol{\Psi}}_{s,t}(\boldsymbol{X}_s, \boldsymbol{W}_{[s,t]}) \\ &- \hat{\boldsymbol{\Psi}}_{u,t}(\hat{\boldsymbol{\Psi}}_{s,u}(\boldsymbol{X}_s, \boldsymbol{W}_{[s,u]}), \boldsymbol{W}_{[u,t]})\|_2^2\Big].\end{aligned} \tag{16}$$

Proof. We have that for any $\hat{\boldsymbol{v}}$,

$$\begin{aligned}\mathcal{L}_{\boldsymbol{f},\boldsymbol{g}}(\hat{\boldsymbol{v}}_{t,t}) \geq \mathcal{L}_{\boldsymbol{f},\boldsymbol{g}}(\boldsymbol{v}_{t,t}) = 0, \\ \mathcal{L}_{\mathrm{D}}(\hat{\boldsymbol{v}}) \geq 0.\end{aligned}$$

This follows since $\mathcal{L}_{\boldsymbol{f},\boldsymbol{g}}$ is convex in $\hat{\boldsymbol{v}}_{t,t}$ with unique global minimizer $\boldsymbol{v}_{t,t}$. Note that the Itô map satisfies $\mathcal{L}_{\mathrm{D}}(\boldsymbol{v}) = 0$. This implies that for the Itô map we obtain the minimum of the objective, given by

$$\mathcal{L}_{\mathrm{SD}}(\boldsymbol{v}) = 0.$$

To show that the minimizer is unique, consider any $\hat{\boldsymbol{v}}$ (and associated $\hat{\boldsymbol{\Psi}}$) that obtains the minimum,

$$\mathcal{L}_{SD}(\hat{\boldsymbol{v}}) = 0$$

Then we must have

$$\hat{\boldsymbol{v}}_{t,t} = \boldsymbol{v}_{t,t}$$
$$\mathcal{L}_{\mathrm{D}}(\hat{\boldsymbol{v}}) = 0.$$

By Propositions 3.2 (since $\hat{\boldsymbol{v}}_{t,t} = \boldsymbol{v}_{t,t}$, the tangent condition holds) and 3.3 (since $\mathcal{L}_{\mathrm{D}} = 0$, the semigroup condition holds), this implies $\hat{\boldsymbol{\Psi}}$ is the Itô map for (6). □

A.2. Constructing the Brownian motion

A.2.1. Primer on rough path theory

A (step-2) rough path is an element of $G^2(\mathbb{R}^d) = \mathbb{R}^d \oplus \mathbb{R}^{d\times d}$. We define a metric on $\mathbb{X} = (\mathbb{X}^{(1)}, \mathbb{X}^{(2)}) \in G^2(\mathbb{R}^d)$ by,

$$d(\mathbb{X}_s, \mathbb{X}_t) \coloneqq |\mathbb{X}^{(1)}_{s,t}| + |\mathbb{X}^{(2)}_{s,t}|^{\frac{1}{2}},$$

where we have used $|\cdot|$ to indicate the *Euclidean norm on* $\mathbb{R}^{\otimes k}$ for the appropriate choice of $k \in \{1, 2\}$. To measure the "difference" between two rough paths $\mathbb{X}, \mathbb{Y} \in G^2(\mathbb{R}^2)$ we define a pathwise metric, for each $\alpha \in [0, \frac{1}{2})$, the α-Hölder metric by,

$$d_{\alpha\text{-Höl}}(\mathbb{X}, \mathbb{Y}) \coloneqq \sup_{0\le s<t\le 1} \Big(\frac{|\mathbb{X}^{(1)}_{s,t} - \mathbb{Y}^{(1)}_{s,t}| + |\mathbb{X}^{(2)}_{s,t} - \mathbb{Y}^{(2)}_{s,t}|^{\frac{1}{2}}}{|t-s|^{\alpha}} \Big).$$

To avoid ambiguity of indices being used to represent both the *level* of a rough path, and in Definition 3.1 the order of our approximation, we will sometimes use the projection operator, $\pi_k : G^2(\mathbb{R}^d) \to (\mathbb{R}^d)^{\otimes k}$ for $k = 1, 2$, to explicitly distinguish the first and second levels of the rough path respectively, i.e. $\pi_k(\mathbb{X}) = \mathbb{X}^{(k)}$.

For both Brownian motion $\boldsymbol{W}_t$ and the polynomial approximation $\boldsymbol{W}^{(N)}_t$ from Definition 3.1, we lift the paths to rough paths $\mathbb{W}$ and $\mathbb{W}^{(N)}$ respectively by defining the canonical Stratonovich lifts,

$$\pi_2(\mathbb{W}_{s,t}) \coloneqq \int_s^t \boldsymbol{W}_{s,u} \otimes \mathrm{d}\boldsymbol{W}_u,$$
$$\pi_2(\mathbb{W}^{(N)}_{s,t}) \coloneqq \int_s^t \boldsymbol{W}^{(N)}_{s,u} \otimes \mathrm{d}\boldsymbol{W}^{(N)}_u.$$

A.2.2. Proof of Theorem 3.4

Theorem 3.4 (Properties of $\boldsymbol{W}^{(N)}_{u,v}$)**.** *The polynomial approximation of the Brownian motion in Definition 3.1 has the following properties:*

1. *Converges to Brownian motion in the α-Hölder distance, with $\alpha \in [0, \frac{1}{2})$,*
2. *The coefficients $\boldsymbol{I}^{(n)}_{s,t}$ are independently and normally distributed with*

$$\boldsymbol{I}^{(n)}_{s,t} \sim \mathcal{N}\left(\boldsymbol{0}, \frac{(t-s)}{2n+1}\boldsymbol{I}\right), \tag{12}$$

3. *The coefficients $\boldsymbol{I}^{(n)}_{s,t}$ admit closed form Chen relations.*

Proof. Property 2 is simply an application of Itô's isometry, using the fact that the n^{th} (shifted) Legendre polynomial has a normalisation constant of $\frac{1}{2n+1}$,

$$\boldsymbol{I}^{(n)}_{s,t} = \int_s^t \widetilde{P}_n\Big(\frac{r-s}{t-s}\Big)\,\mathrm{d}\boldsymbol{W}_r \sim \mathcal{N}\big(0, K(s,t)\big),$$

where,

$$\begin{aligned} K(s,t) &= \mathbb{E}\Big[\int_s^t \widetilde{P}_n\Big(\frac{r-s}{t-s}\Big)\,\mathrm{d}\boldsymbol{W}_r \otimes \int_s^t \widetilde{P}_n\Big(\frac{r-s}{t-s}\Big)\,\mathrm{d}\boldsymbol{W}_r\Big] \\ &= \boldsymbol{I}\int_s^t \widetilde{P}_n\Big(\frac{r-s}{t-s}\Big)^2\,\mathrm{d}r \\ &= \frac{t-s}{2n+1}\boldsymbol{I}. \end{aligned}$$

Property 1 and 3 require slightly more work, so are proven individually as Proposition A.3 and Proposition A.5 respectively. □

The following lemma will be required to establish rough-path convergence of the polynomial approximation $\boldsymbol{W}_t$.

Lemma A.2 (Uniform bounds on $\mathbb{W}$)**.** *For $\alpha \in [0, \frac{1}{2})$, there exists $C_\alpha \in L^2(\mathbb{P})$ such that for all $[u,v] \subseteq [s,t]$,*

$$\|\mathbb{W}_{u,v}\| \le C_\alpha |v-u|^\alpha \tag{19}$$

where $\|\cdot\| : G^2(\mathbb{R}^d) \to \mathbb{R}$ denotes the norm,

$$\|\mathbb{X}_{u,v}\| = \max\Big(\big\|\mathbb{X}^{(1)}_{u,v}\big\|_{L^2}, \big\|\mathbb{X}^{(2)}_{u,v}\big\|^{\frac{1}{2}}_{L^2}\Big).$$

Proof. This is a standard application of the Garsia–Rodemich–Rumsey lemma (Garsia et al., 1970, Lemma 1.1), which can be generalised to any metric as seen in (Friz & Victoir, 2010, Proposition A.8). Choosing $p(u) = |u|^{1/2}$, $\Psi(u) = u^q$ for some $q \ge \frac{4}{1-2\alpha}$ and $B = \int_0^1\int_0^1 \Psi\big(\frac{\|\mathbb{X}_{u,v}\|}{p(v-u)}\big)\,\mathrm{d}u\,\mathrm{d}v$, we obtain,

$$\|\mathbb{W}_{u,v}\| \le C_\alpha |v-u|^{1/2-2/q} \le C_\alpha |v-u|^\alpha,$$

where $C_\alpha := \frac{8q(4B)^{1/q}}{q-4}$. To verify $C_\alpha \in L^2(\mathbb{P})$ we use $\|\mathbb{W}_{u,v}\|_{L^2(\mathbb{P})} = |v-u|^{1/2}$

$$\begin{aligned} \mathbb{E}[C_\alpha^2] &\lesssim \mathbb{E}[B^{2/q}] \\ &\le \int_0^1\int_0^1 \frac{\|\mathbb{W}_{u,v}\|^2_{L^2(\mathbb{P})}}{|v-u|^{2\alpha}}\,\mathrm{d}u\,\mathrm{d}v \\ &= \int_0^1\int_0^1 |v-u|^{1-2\alpha}\,\mathrm{d}u\,\mathrm{d}v \\ &< \infty. \end{aligned}$$

□

Proposition A.3 (Rough path convergence of the polynomial approximation)**.** *The polynomial expansion of Brownian motion given in Definition 3.1 converges in the rough-path sense, i.e. for any $\alpha \in [0, \frac{1}{2})$,*

$$d_{\alpha-H\ddot{o}l}(\mathbb{W}, \mathbb{W}^N) \xrightarrow{a.s.} 0,$$

as $N \to \infty$.

Proof. As shown in (Foster et al., 2020, Theorem 2.4), we can define the following filtration,

$$\big\{\mathscr{F}_N := \sigma(\{\boldsymbol{I}^{(n)}_{s,t} \;:\; n = 0,1,\cdots,N\})\big\}_{N\ge 0},$$

so that the polynomial approximation admits the representation,

$$\mathbb{W}^{(N)}_{u,v} = \mathbb{E}[\mathbb{W}_{u,v} \,|\, \mathscr{F}_N],$$

for any $[u, v] \subseteq [s, t]$. Thus, by taking expectations with respect to $\mathscr{F}_N$ in Lemma A.2,

$$\|\mathbb{W}_{u,v}^{(N)}\| \leq \widetilde{C}_\alpha |v-u|^\alpha, \tag{20}$$

where $\widetilde{C}_\alpha \coloneqq \sup_{n\geq 0} \mathbb{E}[C_\alpha \,|\, \mathscr{F}_n]$ and where we continue to use $\|\cdot\|$ as defined in Lemma A.2. By Doob's maximal inequality $\widetilde{C}_\alpha$ is finite a.s. since $\mathbb{E}[C_\alpha \,|\, \mathscr{F}_n]$ is a (discrete) martingale.

Combining Equation (19) and Equation (20), the sequence $\{\mathbb{W}_{u,v} - \mathbb{W}_{u,v}^{(N)}\}_{N\geq 0}$ is uniformly bounded and uniformly equicontinuous. Thus, invoking the Arzelà-Ascoli theorem, we conclude that there exists a uniformly convergent subsequence and since pointwise convergence is proven in (Foster et al., 2020), we know that this limit is zero, so,

$$\mathbb{W}_{u,v}^{(N)} \xrightarrow{a.s.} \mathbb{W}_{u,v} \qquad \text{uniformly as } N \to \infty.$$

Finally, defining $C^* = \max(C_\beta, \widetilde{C}_\beta)^{\frac{\alpha}{\beta}}$, use the following inequality,

$$\begin{aligned}
\frac{\|\pi_k(\mathbb{W}_{u,v} - \mathbb{W}_{u,v}^{(N)})\|_{L_2}}{|v-u|^{k\alpha}} &\leq \left(\frac{\|\pi_k(\mathbb{W}_{u,v} - \mathbb{W}_{u,v}^{(N)})\|_{L_2}}{|v-u|^{\beta}}\right)^{\frac{\alpha}{\beta}} \Big(\sup_{0\leq s<t\leq 1} \|\pi_k(\mathbb{W}_{u,v} - \mathbb{W}_{u,v}^{(N)})\|_{L_2}\Big)^{1-\frac{\alpha}{\beta}} \\
&\leq C^* \Big(\sup_{0\leq u<v\leq 1} \|\pi_k(\mathbb{W}_{u,v} - \mathbb{W}_{u,v}^{(N)})\|_{L_2}\Big)^{1-\frac{\alpha}{\beta}} \\
&\xrightarrow{a.s.} 0 \qquad \text{uniformly as } N \to \infty.
\end{aligned}$$

Thus, $d_{\alpha-\text{Höl}}(\mathbb{W}, \mathbb{W}^{(N)}) \xrightarrow{a.s.} 0$ as $N \to \infty$. □

Finally, we move to deriving the explicit Chen relations for the integrals $\boldsymbol{I}_{s,t}^{(n)}$ To achieve this, we first establish the following related dilation rules for the (shifted) Legendre polynomials.

Lemma A.4 (Dilation rule for shifted Legendre polynomials)**.** *Let $\widetilde{P}_n(x)$ denote the n^{th} shifted Legendre polynomial on $[0, 1]$. Then,*

$$\widetilde{P}_n(x) = \sum_{m=0}^{n} c_{n,m}\, \widetilde{P}_m(2x) = \sum_{m=0}^{n} (-1)^{n+m} c_{n,m} \widetilde{P}_m(2x-1), \tag{21}$$

where,

$$c_{n,m} \coloneqq (-1)^n (2m+1) \sum_{k=m}^{n} \Big(-\frac{1}{2}\Big)^k \frac{(n+k)!}{(n-k)!(k-m)!(k+m+1)!}.$$

Proof. Using the orthogonality relations for Legendre polynomials, we derive the following integral expression for $c_{n,m}$,

$$(2m+1)\int_0^1 \widetilde{P}_n\Big(\frac{u}{2}\Big)\widetilde{P}_m(u)\,\mathrm{d}u = (2m+1)\sum_{m=0}^{n} c_{n,m}\int_0^1 \widetilde{P}_n(u)\widetilde{P}_k(u)\,\mathrm{d}u = c_{n,m}.$$

We can evaluate the integral explicitly. By substituting Rodrigues' formula for $\widetilde{P}_m(u)$, before applying integration by parts m-times (noting that the boundary terms always vanish),

$$\begin{aligned}
\int_0^1 \widetilde{P}_n\Big(\frac{u}{2}\Big) P_m(u)\,\mathrm{d}u &= \frac{1}{m!}\int_0^1 \widetilde{P}_n\Big(\frac{u}{2}\Big)\frac{\mathrm{d}^m}{\mathrm{d}u^m}[x^m(x-1)^m]\,\mathrm{d}u \\
&= \frac{1}{m!}\int_0^1 x^m(1-x)^m \frac{\mathrm{d}^m}{\mathrm{d}u^m}\widetilde{P}_n\Big(\frac{u}{2}\Big)\,\mathrm{d}u.
\end{aligned}$$

Next, substituting the explicit form for the shifted Legendre polynomial,

$$\widetilde{P}_n\Big(\frac{u}{2}\Big) = (-1)^n \sum_{k=0}^{n} \binom{n}{k}\binom{n+k}{k}\Big(-\frac{x}{2}\Big)^k,$$

we obtain,

$$\begin{aligned}
\int_0^1 \widetilde{P}_n\Big(\frac{u}{2}\Big) P_m(u)\,\mathrm{d}u &= \frac{1}{m!}\int_0^1 x^m(1-x)^m \frac{\mathrm{d}^m}{\mathrm{d}u^m}\Big[(-1)^n \sum_{k=0}^n \binom{n}{k}\binom{n+k}{k}\Big(-\frac{x}{2}\Big)^k\Big]\,\mathrm{d}u \\
&= \frac{(-1)^n}{m!}\sum_{k=m}^n \Big(-\frac{1}{2}\Big)^k \binom{n}{k}\binom{n+k}{k}\int_0^1 x^m(1-x)^m \frac{\mathrm{d}^m}{\mathrm{d}u^m}[x^k]\,\mathrm{d}u \\
&= \frac{(-1)^n}{m!}\sum_{k=m}^n \Big(-\frac{1}{2}\Big)^k \frac{k!}{(k-m)!}\binom{n}{k}\binom{n+k}{k}\int_0^1 x^k(1-x)^m\,\mathrm{d}u \\
&= \frac{(-1)^n}{m!}\sum_{k=m}^n \Big(-\frac{1}{2}\Big)^k \frac{k!}{(k-m)!}\binom{n}{k}\binom{n+k}{k}\frac{k!\,m!}{(k+m+1)!} \\
&= (-1)^n \sum_{k=m}^n \Big(-\frac{1}{2}\Big)^k \frac{(n+k)!}{(n-k)!(k-m)!(k+m+1)!}.
\end{aligned}$$

For the second equation, use the symmetry of the Legendre polynomial,

$$\begin{aligned}
\widetilde{P}_n(x) &= (-1)^n \widetilde{P}_n(1-x) \\
&= (-1)^n \sum_{m=0}^n c_{n,m}\widetilde{P}_m\big(2(1-x)\big) \\
&= (-1)^n \sum_{m=0}^n c_{n,m}\widetilde{P}_m\big(1-(2x-1)\big) \\
&= \sum_{m=0}^n (-1)^{n+m} c_{n,m}\widetilde{P}_m(2x-1).
\end{aligned}$$

□

The proof of Proposition A.5 now follows naturally, by splitting the domain of the integral $\boldsymbol{I}_{s,t}^{(n)}$ into two and applying the relations we have just derived.

Proposition A.5 (Chen relations for $\boldsymbol{I}^{(n)}$)**.** *Let $u = \dfrac{s+t}{2}$ be the midpoint of times $s \le t$. Then,*

$$\boldsymbol{I}_{s,t}^{(n)} = \sum_{m=0}^n c_{n,m}\Big(\boldsymbol{I}_{s,u}^{(m)} + (-1)^{n+m}\boldsymbol{I}_{u,t}^{(m)}\Big), \tag{22}$$

where $c_{n,m}$ are defined as in Lemma A.4.

Proof. By splitting $[s,t]$ into the two half-domains $[s,u]$ and $[u,t]$, we can apply the dilation rules for the Legendre polynomials given in Lemma A.4. We use the identities,

$$2\Big(\frac{r-s}{t-s}\Big) = \frac{r-s}{u-s},$$

$$2\Big(\frac{r-s}{t-s}\Big) - 1 = \frac{r-u}{t-u},$$

so that,

$$\begin{aligned}\boldsymbol{I}_{s,t}^{(n)} &= \int_s^t \widetilde{P}_n\Big(\frac{r-s}{t-s}\Big)\,\mathrm{d}\boldsymbol{W}_r \\ &= \sum_{m=0}^{n} c_{n,m}\Big(\int_s^u \widetilde{P}_m\Big(\frac{r-s}{u-s}\Big)\,\mathrm{d}\boldsymbol{W}_r + (-1)^{n+m}\int_u^t \widetilde{P}_m\Big(\frac{r-u}{t-u}\Big)\,\mathrm{d}\boldsymbol{W}_r\Big) \\ &= \sum_{m=0}^{n} c_{n,m}\Big(\boldsymbol{I}_{s,u}^{(m)} + (-1)^{n+m}\boldsymbol{I}_{u,t}^{(m)}\Big),\end{aligned}$$

as required. □

B. Training

B.1. Euler-Maruyama step objective

Here we show that matching an Euler-Maruyama step with the strong stochastic flow map model is equivalent to a weighted coefficient loss.

Lemma B.1 (Euler-Maruyama objective)**.** *Consider the following objective*

$$\mathcal{L}_{EM} = \big\|\boldsymbol{X}_t - \hat{\boldsymbol{X}}_t\big\|^2,$$

where $\hat{\boldsymbol{X}}_t = \boldsymbol{\Psi}_{s,t}(\boldsymbol{X}_s, \boldsymbol{W}_{[s,t]})$ *via* (8) *and*

$$\boldsymbol{X}_t = \boldsymbol{X}_s + \boldsymbol{f}(s, \boldsymbol{X}_s)(t-s) + \boldsymbol{g}(s)(\boldsymbol{W}_t - \boldsymbol{W}_s).$$

Since the true coefficients $\boldsymbol{f}(s, \boldsymbol{X}_s), \boldsymbol{g}(s)$ *depend only on* s*, it suffices to restrict the model to coefficients* $\boldsymbol{f}_{s,t}(\boldsymbol{X}_s)$ *and* $\boldsymbol{g}_{s,t}$ *that are independent of* $\boldsymbol{W}_{[s,t]}$*. Then*

$$\mathbb{E}[\mathcal{L}_{EM} \mid \boldsymbol{X}_s] = (t-s)^2\mathcal{L}_f + (t-s)\mathcal{L}_g,$$

where $\mathcal{L}_f = \big(\boldsymbol{f}(s, \boldsymbol{X}_s) - \boldsymbol{f}_{s,t}(\boldsymbol{X}_s)\big)^2$ *and* $\mathcal{L}_g = \big(\boldsymbol{g}(s) - \boldsymbol{g}_{s,t}\big)^2$.

Proof. Under the independence assumption, $\boldsymbol{f}_{s,t}(\boldsymbol{X}_s)$ and $\boldsymbol{g}_{s,t}$ are independent of $\boldsymbol{W}_{[s,t]}$, so

$$\boldsymbol{X}_t - \hat{\boldsymbol{X}}_t = \big(\boldsymbol{f}(s, \boldsymbol{X}_s) - \boldsymbol{f}_{s,t}(\boldsymbol{X}_s)\big)\,(t-s) + \big(\boldsymbol{g}(s) - \boldsymbol{g}_{s,t}\big)\,(\boldsymbol{W}_t - \boldsymbol{W}_s).$$

Expanding the square,

$$\begin{aligned}\mathcal{L}_{\text{EM}} = (t-s)^2\,&\big(\boldsymbol{f}(s, \boldsymbol{X}_s) - \boldsymbol{f}_{s,t}(\boldsymbol{X}_s)\big)^2 \\ &+ 2(t-s)\,\big(\boldsymbol{f}(s, \boldsymbol{X}_s) - \boldsymbol{f}_{s,t}(\boldsymbol{X}_s)\big)\,\big(\boldsymbol{g}(s) - \boldsymbol{g}_{s,t}\big)\,(\boldsymbol{W}_t - \boldsymbol{W}_s) \\ &+ \big(\boldsymbol{g}(s) - \boldsymbol{g}_{s,t}\big)^2\,(\boldsymbol{W}_t - \boldsymbol{W}_s)^2.\end{aligned}$$

Taking $\mathbb{E}[\cdot \mid \boldsymbol{X}_s]$ and using $\mathbb{E}[\boldsymbol{W}_t - \boldsymbol{W}_s \mid \boldsymbol{X}_s] = 0$ and $\mathbb{E}[(\boldsymbol{W}_t - \boldsymbol{W}_s)^2 \mid \boldsymbol{X}_s] = t - s$,

$$\mathbb{E}[\mathcal{L}_{\text{EM}} \mid \boldsymbol{X}_s] = (t-s)^2\,\big(\boldsymbol{f}(s, \boldsymbol{X}_s) - \boldsymbol{f}_{s,t}(\boldsymbol{X}_s)\big)^2 + (t-s)\,\big(\boldsymbol{g}(s) - \boldsymbol{g}_{s,t}\big)^2.$$

Substituting for $\mathcal{L}_f, \mathcal{L}_g$ gives the result. □

B.2. Diffusion SDEs

To apply Algorithm 1 to diffusion SDEs, we must derive the ground truth reverse diffusion SDE. Consider the variance preserving formulation,

$$\mathrm{d}\boldsymbol{X}_t = -\frac{1}{2}\beta_t\boldsymbol{X}_t\,\mathrm{d}t + \sqrt{\beta_t}\,\mathrm{d}\boldsymbol{W}_t.$$

The reverse diffusion SDE is given by

$$\mathrm{d}\boldsymbol{X}_t = [-\frac{1}{2}\beta_t \boldsymbol{X}_t - \beta_t \nabla_{\boldsymbol{X}_t} \log p(t, \boldsymbol{X}_t)]\,\mathrm{d}t + \sqrt{\beta_t}\,\mathrm{d}\boldsymbol{W}_t,$$

where $\nabla_{\boldsymbol{X}_t} \log p(t, \boldsymbol{X}_t)$ is the score.

Given a data sample $\boldsymbol{X}_1$, an expression for the score can be obtained. Specifically, we sample the forward process via $\boldsymbol{X}_t = \alpha_t \boldsymbol{X}_1 + \sigma_t \epsilon$, $\epsilon \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$, where

$$\alpha_t = \exp\left(-\tfrac{1}{2}\int_0^t \beta_s\, ds\right), \quad \sigma_t^2 = 1 - \alpha_t^2.$$

Then, the score is given by

$$\nabla_{\boldsymbol{X}_t} \log p(t, \boldsymbol{X}_t) = -\frac{\epsilon}{\sigma_t}.$$

We are therefore able to build the ground truth SDE required for training the stochastic flow map by

$$\mathrm{d}\boldsymbol{X}_t = [-\frac{1}{2}\beta_t \boldsymbol{X}_t + \beta_t \frac{\epsilon}{\sigma_t}]\,\mathrm{d}t + \sqrt{\beta_t}\,\mathrm{d}\boldsymbol{W}_t.$$

C. Extended related work

C.1. Few-step models

The training objectives for SSFMs with small step sizes are related to generator matching, *i.e.*, matching the drift and diffusion coefficients (Holderrieth, Havasi, et al., 2025). The semigroup loss is the stochastic analogue to the one used in deterministic flow maps studied by (Boffi et al., 2025; Frans et al., 2025). Consistency models (Song & Dhariwal, 2024; Song, Dhariwal, et al., 2023) use a small Euler step which is related to our small stochastic Euler-Maruyama step; however, the actual loss itself is quite different as we predict the small jump rather than using consistency. Other methods (Boffi et al., 2025; Geng, Deng, et al., 2025; Geng, Lu, et al., 2025) train with losses which require time derivatives of the flow map, a requirement that does not extend naturally to the stochastic setting, as the Brownian motion is nowhere differentiable w.r.t. time.

C.2. Few-step sampling of SDEs

An active area of research has been accelerating inference with diffusion SDEs via more efficient numerical schemes (Blasingame & Liu, 2026; Gonzalez et al., 2023; Zhang & Chen, 2023), thereby decreasing the NFE to use these models; however, all these techniques are at inference time and use the same models. Recent work by Jiang et al. (2025) looks at using *partial signatures* of the Brownian motion to distill pre-trained diffusion SDEs into taking larger step sizes by learning a hyper-solver (Poli et al., 2020). This is significantly different from our work as it requires numerically integrating the entire SDE, *i.e.*, it does not have a simulation-free manner of training even for flow/diffusion models and relies on a pre-trained diffusion model teacher. SSFMs on the other hand enable scalable training of stochastic flow maps and can be trained without a pre-trained teacher model.

C.3. Stochastic few-step models

Recent work by Holderrieth, Chen, et al. (2026), Passaro et al. (2026), and Potaptchik et al. (2026) has looked at weak approximations to the diffusion SDE. As mentioned in the main text these works all learn an "inner" flow map to learn the transition kernel; Passaro et al. (2026) discusses the nuances between these works in more detail. Beyond providing a strong solution, SSFMs also work for arbitrary additive-noise SDEs, whereas existing weak approaches are typically formulated for specific diffusion model SDEs.

D. Experimental details and further experiments

In the image and molecule generation experiments we take the number of polynomial coefficients $N = 3$. We use the Virtual Brownian Tree (VBT) implementation in Diffrax to generate the coefficients $\boldsymbol{I}_{s,t}^{(3)}$ (Jelinčič et al., 2024; Kidger, 2022). It should be noted that the VBT implementation introduces a constant scaling factor on each coefficient.

D.1. Non-linear SDE

We include additional informative plots for the non-linear SDE experiment that would not fit in the main text. In Figure 6 we show the polynomial approximation of Brownian motion for varying degree polynomials. In Figure 7 we show the SSFM predictions for many step sizes when trained with the $N = 4$ degree polynomial.

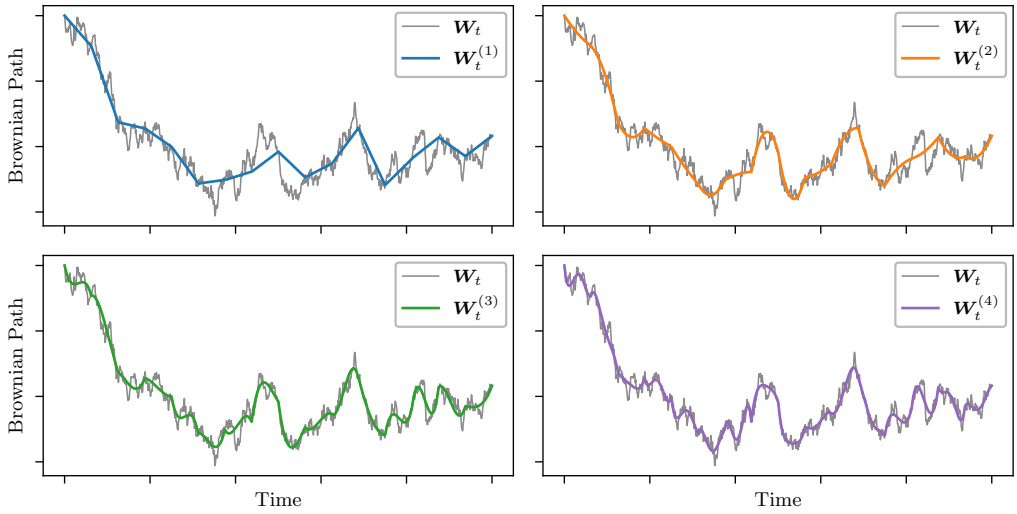

Figure 6. Brownian path $\boldsymbol{W}_t$ and associated polynomial approximations $\boldsymbol{W}_t^{(N)}$ over 16-steps.

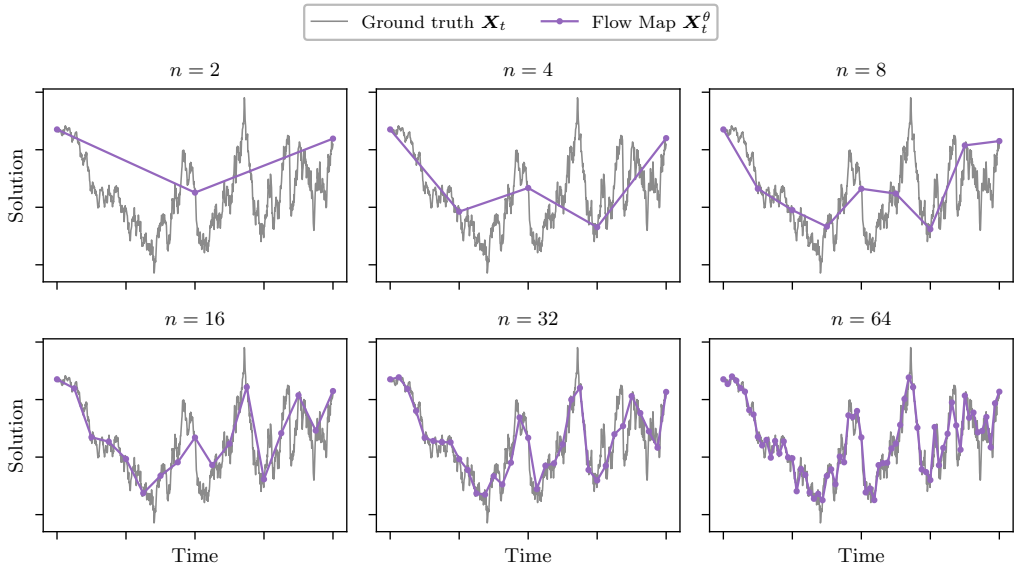

Figure 7. Ground truth SDE and SSFM prediction for many step counts.

D.2. Image generation

For the image generation experiments we setup the variance preserving SDE (see Appendix B.2) as the ground truth and follow Algorithm 1 to train the SSFM. The drift and diffusion networks are parameterized with the EDM2 architecture (Config C) (Karras et al., 2024) with hyperparameters given in Table 4.

For evaluation, we generate 50k samples with the SSFM and compute FID against the CIFAR-10 dataset. Uniform step placement is used so that for n steps, the step size is given by $(1 - t_\epsilon)/N$, where $t_\epsilon = 10^{-5}$.

On two NVIDIA RTX A6000s, this model takes ~ 5.5 days to train.

Table 4. Image generation hyperparameters.

Hyperparameter	CIFAR-10
Drift network (EDM2 U-Net)	
Base channels	128
Channel multipliers	$(2, 2, 2)$
Attention resolutions	16×16
Attention head dimension	64
GroupNorm groups	8
Dropout	0.13
Diffusion network (EDM2 U-Net)	
Base channels	64
Channel multipliers	$(2, 2, 2)$
Attention resolutions	16×16
Attention head dimension	64
GroupNorm groups	8
Dropout	0
Loss	
Step size split Δt	0.01
Max distillation step $h_{\max}$	0.52
Batch split η	0.75
Optimization	
Optimizer	Adam ($\beta_2 = 0.99$)
Peak learning rate	10^{-3}
Min learning rate	10^{-5}
LR schedule	warmup + cosine decay
Warmup steps	5,000
Gradient clip (global norm)	10
Batch size	512
Training steps	400,000
EMA decay	0.999

D.3. Molecular systems

The alanine dipeptide dataset follows (Plainer et al., 2025) with 50k samples from a molecular dynamics simulation in implicit solvent (Köhler et al., 2021), coarse grained to five atoms [C, N, CA, C, N]. It is available from the *ScoreMD* repository accompanying (Plainer et al., 2025).

The evaluation metrics calculated are the potential of mean force (PMF) squared error and the Jensen-Shannon (JS) divergence. These metrics both compare the difference in the equilibrium free energy surfaces of the ground truth system and the model prediction.

A graph transformer architecture is used with hyperparameters listed in Table 5. The variance preserving SDE is used as the ground truth to construct the SSFM following Algorithm 1.

The diffusion baseline models were obtained from the *ScoreMD* repository and results collected via the evaluation code provided. The SSFM model evaluation was also computed from the *ScoreMD* code to ensure a fair comparison.

On one NVIDIA RTX A6000s, this model takes $\sim$ 6 hours to train.

Table 5. Molecule generation hyperparameters.

Hyperparameter	ALDP
Drift/Diffusion network (graph-transformer)	
Hidden dim	96
Transformer blocks	3/2
Attention heads	8
Head dim	64
Feed-forward multiplier	4
Time-embedding dim	64
Uncertainty MLP Fourier dim	64
Loss	
Step size split Δt	10^{-3}
Max distillation step $h_{\max}$	0.52
Batch split η	0.75
Optimization	
Optimizer	AdamW
Peak learning rate	10^{-3}
Min learning rate	10^{-5}
LR schedule	warmup + cosine decay
Warmup steps	1,000
Gradient clip (global norm)	10
Batch size	1,024
Training steps	400,000
EMA decay	0.999

D.4. Hardware

All experiments were run on one/two NVIDIA RTX A6000 GPUs.

D.5. Repositories

We made use of the following repositories and resources:

1. `patrick-kidger/diffrax` (for VBT)
2. `patrick-kidger/equinox` (for neural networks in JAX)
3. `Lightning-AI/torchmetrics` (for FID)
4. `noegroup/ScoreMD` (for Alanine-Dipeptide dataset and diffusion baselines)

E. Discussions

E.1. Broader impacts

We propose a framework for few-step generation of additive-noise SDEs, with demonstrated applications in image generation and molecular dynamics. The primary positive impacts of this work is the acceleration of molecular dynamic simulations relevalent to drug discovery and other ai4science applications which use diffusion models. Improvements to image generation efficiency also reduce inference-time energy consumption. As with all advances in generative modelling, there is a risk that

improved image generation could be misused to produce harmful synthetic media; however, SSFMs represent an efficiency improvement to existing pipelines rather than a new capability, and we do not believe this work introduces risks beyond those already present in deployed diffusion models.

E.2. Limitations

The current framework is restricted to additive-noise SDEs; extending this framework to state-dependent SDEs would require the full machinery associated with the Itô-Lyons map from the theory of rough paths and may require approximating the space-space Lévy area. We leave this to future work. The polynomial approximation of the Brownian path introduces a truncation error controlled by the degree N, which in practice requires tuning as a hyperparameter. Our image generation experiments are conducted on CIFAR-10; scaling to larger datasets and higher resolutions remains to be demonstrated. Finally, the empirical comparison to weak stochastic flow maps focuses primarily on generative performance metrics; a more detailed empirical study of the pathwise consistency properties and their downstream implications is left to future work.