universität wien

Diffusion Models for Molecular Dynamics | Optimization Seminar

Bogdan Chuzhinov – a12331995@unet.univie.ac.at

February 15, 2026, Vienna

Contents

1 Introduction

One of the central challenges of modern computational biology is the description of the behavior of large biological molecules. Proteins represent the most important class of such molecules, as their functional roles in living organisms are highly diverse and include, among others, catalytic activity, molecular transport, protective and signaling mechanisms.

A protein is a sequence of amino acids linked by peptide bonds and characterized by specific physical and chemical properties that depend on the geometric organization of this sequence in three-dimensional space. The linear sequence of 20 amino acid types, considered independently of any spatial conformation, is referred to as the primary structure of a protein. Such sequences are typically long and may consist of thousands amino acid residues. The primary structure alone does not uniquely determine the structure and, consequently, the function of a protein. To understand the function of a particular protein, it is necessary to know how this sequence is arranged in space — its so-called tertiary structure. In the case of complex molecules composed of multiple polypeptide chains, it is additionally necessary to understand how these chains interact with each other, forming the quaternary structure.

In the late 1950s, Perutz et al. (1960) and Kendrew et al. (1958), using an improved method of X-ray crystallography developed by them, were the first to determine the three-dimensional structures of hemoglobin and myoglobin — proteins that play a central role in oxygen transport in vertebrate organisms. This discovery marked the beginning of the final stage in the transition of biology from a descriptive science to a molecular one.

As computational technologies have advanced, costly and technically difficult experimental approaches have increasingly been complemented or replaced by cheaper and more easily reproducible computational simulations. Monte Carlo methods and Newtonian *Molecular Dynamics* (MD) have proven to be particularly effective for modeling molecular interactions in large biological systems.

At a high level, the aim of molecular dynamics is to integrate the equations of motion

$$M_i \ddot{x}_i = -\nabla_{x_i} U(x_1, \ldots, x_N),$$

for each particle i in a molecular configuration $(x_1, \ldots, x_N) \in \mathbb{R}^{3N}$, where M_i is the mass and U is the potential energy function $U : \mathbb{R}^{3N} \to \mathbb{R}$. In classical MD, the potential energy $U(x_1, \ldots, x_N)$ is commonly decomposed into bonded and non-bonded contributions:

$$U = U_{\text{bond}} + U_{\text{angle}} + U_{\text{dihedral}} + U_{\text{Coulomb}} + U_{\text{vdW}}.$$

Each term corresponds to a force of distinct physical origin, as detailed below.

Bond stretching. Bonded pairs of atoms (i, j) contribute harmonic terms of the form

$$U_{\text{bond}} = \sum_{(i,j)\in\mathcal{B}} \frac{k_{ij}^{(r)}}{2} \left(r_{ij} - r_{ij}^0\right)^2,$$

where $r_{ij} = \|x_i - x_j\|$ is the inter-atomic distance, r_{ij}^0 is the equilibrium bond length, and $k_{ij}^{(r)}$ is a force constant.

Angle bending. Triplets of bonded atoms (i, j, k) give rise to angle terms

$$U_{\text{angle}} = \sum_{(i,j,k)\in\mathcal{A}} \frac{k_{ijk}^{(\theta)}}{2} \left(\theta_{ijk} - \theta_{ijk}^0\right)^2,$$

where θ_{ijk} is the bond angle at atom j.

Dihedral (torsional) interactions. Quadruplets (i, j, k, l) define dihedral angles ϕ_{ijkl}, modeled via periodic potentials:

$$U_{\text{dihedral}} = \sum_{(i,j,k,l)\in\mathcal{D}} k_{ijkl}^{(\phi)} \left(1 + \cos(n\phi_{ijkl} - \delta)\right),$$

where n is the multiplicity and δ a phase offset.

Electrostatic interactions. Non-bonded pairs contribute Coulomb interactions:

$$U_{\text{Coulomb}} = \sum_{i<j} \frac{q_i q_j}{4\pi\varepsilon_0 r_{ij}},$$

where q_i are partial atomic charges.

Van der Waals interactions. Short-range repulsion and dispersion attraction are typically modeled using a Lennard–Jones potential:

$$U_{\text{vdW}} = \sum_{i<j} 4\varepsilon_{ij} \left[\left(\frac{\sigma_{ij}}{r_{ij}} \right)^{12} - \left(\frac{\sigma_{ij}}{r_{ij}} \right)^{6} \right].$$

The total force acting on atom i is $F_i = -\nabla_{x_i} U$, yielding a highly nonlinear, high-dimensional dynamical system. The bonded terms enforce local geometric structure, while non-bonded terms encode long-range physical interactions, together determining the conformational landscape explored by the system.

Numerical integration of Newton's equations of motion at atomic scales is a standard tool for investigating molecular phenomena in chemistry, biology, and related disciplines (Alder & Wainwright, 1959). MD is general and flexible method but, at the same time, computationally expensive, since, when using a time step of picosecond scale, physically relevant processes require the simulation of dynamics on the millisecond timescale.

A major breakthrough in protein structure prediction was achieved with the development of neural network *AlphaFold*, whose latest version is described in (Abramson et al., 2024). By leveraging deep learning techniques and large-scale structural data, AlphaFold demonstrated high accuracy in predicting the three-dimensional structures of proteins directly from their amino acid sequences.

Following this success, some attempts (Jing, Berger, & Jaakkola, 2024) were made to use AlphaFold-derived structures directly within MD simulations. The underlying idea was that highly accurate static structural predictions could serve as reliable initial configurations for dynamical simulations, potentially accelerating studies of protein folding, conformational transitions, and functional mechanisms. However, while AlphaFold excels at predicting stable conformations, it does not explicitly model thermodynamic ensembles or time-dependent dynamics, which limits its direct applicability to fully dynamical modeling.

Modern generative modeling aims to construct a transformation that maps a simple reference distribution ρ_0 (e.g. a Gaussian prior) to a complex target distribution ρ_1 supported on high-dimensional data. Formally, this can be viewed as constructing a *homotopy* (or continuous interpolation) between ρ_0 and ρ_1, that is, a family of intermediate distributions $(\rho_t)_{t\in[0,1]}$ such that $\rho_{t=0} = \rho_0$ and $\rho_{t=1} = \rho_1$.

Diffusion models realize this idea by introducing a stochastic process that gradually transforms samples from ρ_0 into samples from ρ_1 through a time-dependent dynamics. From a geometric perspective, generative modeling can therefore be interpreted as learning a path in the space of probability measures. This homotopy viewpoint connects diffusion models to optimal transport, stochastic control, and optimization. In particular, optimal transport corresponds to selecting a path that minimizes kinetic energy in the space of measures, while diffusion models may be interpreted as entropy-regularized transport problems. Such a perspective provides a unifying mathematical framework for understanding the structure, training objectives, and stability properties of modern generative models.

In this seminar work, we consider MDGEN, a general-purpose and computationally efficient surrogate modeling framework for MD developed by Jing, Stärk, et al. (2024) based on *direct generative modeling of simulated trajectories*. Instead of directly incorporating AlphaFold-like structure predictors into denoising or diffusion-based frameworks, the authors adopt a fundamentally different perspective. Rather than modeling single static conformations, they formulate end-to-end generative modeling of complete molecular trajectories, viewed as time series of three-dimensional structures. Inspired by the extension of image generative models to video generation (Ho et al., 2022), their formulation augments single-structure generative models with an explicit temporal dimension.

2 Stochastic Interpolants

As discussed in the introduction, diffusion models can be interpreted as constructing a homotopy between two probability measures: a tractable reference distribution ρ_0 and a target distribution ρ_1 we are interested in. This can equivalently be formulated as the problem of finding a time-differentiable interpolant $I_t \colon \Omega \times \Omega \to \Omega$ for some $\Omega \subset \mathbb{R}^d$ such that $I_{t=0}(x_0, x_1) = x_0$, $I_{t=1}(x_0, x_1) = x_1$. Albergo and Vanden-Eijnden (2022) analyzed the properties of such mappings in the case where $x_0 \sim \rho_0$ and $x_1 \sim \rho_1$ are independent random variables. The framework was later extended to a broader class of transformations by relaxing the independence assumption and incorporating latent variables. We now turn to a discussion of this extended formulation.

Unless stated otherwise, the statements and proofs in this chapter follow those of Albergo et al. (2025), and we reproduce the main arguments for completeness and clarity.

2.1 Definitions and Assumptions

We first introduce the stochastic processes central to this framework.

Definition 2.1 (Stochastic interpolant)**.** Given two probability density functions $\rho_0, \rho_1 : \mathbb{R}^d \to \mathbb{R}_{\geq 0}$, a *stochastic interpolant* between ρ_0 and ρ_1 is a stochastic process $(x_t)_{t\in[0,1]}$ defined as

$$x_t = I(t, x_0, x_1) + \gamma(t)z, \qquad t \in [0, 1], \tag{1}$$

where:

1. [1] $I \in C^2\big([0,1]; C^2(\mathbb{R}^d \times \mathbb{R}^d; \mathbb{R}^d)\big)$ satisfies the boundary conditions

$$I(0, x_0, x_1) = x_0, \qquad I(1, x_0, x_1) = x_1,$$

as well as

$$\exists\, C_1 < \infty \ : \ |\partial_t I(t, x_0, x_1)| \leq C_1 |x_0 - x_1|, \quad \forall (t, x_0, x_1) \in [0, 1] \times \mathbb{R}^d \times \mathbb{R}^d; \tag{2}$$

2. $\gamma : [0, 1] \to \mathbb{R}$ satisfies $\gamma(0) = \gamma(1) = 0$, $\gamma(t) > 0$ for all $t \in (0, 1)$, and $\gamma^2 \in C^2([0, 1])$;

3. The pair (x_0, x_1) is drawn from a probability measure ν with marginals ρ_0 and ρ_1, i.e.

$$\nu(dx_0, \mathbb{R}^d) = \rho_0(x_0)\, dx_0, \qquad \nu(\mathbb{R}^d, dx_1) = \rho_1(x_1)\, dx_1;$$

4. z is a Gaussian random variable independent of (x_0, x_1), i.e.

$$z \sim \mathcal{N}(0, I_d), \qquad z \perp (x_0, x_1).$$

The Lipschitz-continuity condition (2) ensures that $I(t, x_0, x_1)$ does not vary too rapidly along the path connecting x_0 at $t = 0$ to x_1 at $t = 1$. In particular, the interpolant remains controlled and does not drift excessively far from the endpoints.

The probability measure ν specifies a coupling between ρ_0 and ρ_1, and thus influences the properties of the resulting stochastic interpolant. A natural and simple choice is the product measure

$$\nu(dx_0, dx_1) = \rho_0(x_0)\rho_1(x_1)\, dx_0 dx_1,$$

in which case x_0 and x_1 are independent.

In what follows, we assume the following conditions on the densities ρ_0, ρ_1, as well as on the coupling between the measure ν and the functional I.

[1] Given a function $b : [0, 1] \times \mathbb{R}^d \to \mathbb{R}^d$, $(t, x) \mapsto b(t, x)$, we write $b \in C^1\big([0, 1]; C^2(\mathbb{R}^d; \mathbb{R}^d)\big)$ to indicate that b is continuously differentiable in t for all $(t, x) \in [0, 1] \times \mathbb{R}^d$ and that $b(t, \cdot) \in C^2(\mathbb{R}^d; \mathbb{R}^d)$ for all $t \in [0, 1]$.

Assumption 2.2. The densities ρ_0 and ρ_1 are strictly positive elements of $C^2(\mathbb{R}^d)$ and are such that

$$\int_{\mathbb{R}^d} |\nabla \log \rho_0(x)|^2 \rho_0(x)\, dx < \infty, \qquad \int_{\mathbb{R}^d} |\nabla \log \rho_1(x)|^2 \rho_1(x)\, dx < \infty.$$

The measure ν and the function I are such that

$$\exists\, M_1, M_2 < \infty \;:\; \mathbb{E}\big[|\partial_t I(t, x_0, x_1)|^4\big] \le M_1, \qquad \mathbb{E}\big[|\partial_t^2 I(t, x_0, x_1)|^2\big] \le M_2, \quad \forall\, t \in [0,1], \tag{3}$$

where the expectation is taken over $(x_0, x_1) \sim \nu$.

We now formalize the notion of the score function associated with a differentiable probability density.

Definition 2.3 (Score function)**.** Let p be a probability density on $\mathbb{R}^d$ that is differentiable and strictly positive on its support. The *score function* of p is

$$s(x) := \nabla_x \log p(x).$$

In score-based diffusion models, the time-dependent score $s_t(x) = \nabla_x \log p_t(x)$ determines the reverse-time stochastic dynamics associated with the forward diffusion process. Accurate estimation of s_t enables sampling from the data distribution via the corresponding reverse SDE or probability flow ODE.

2.2 Transport Equation and Quadratic Objectives

We now state the fundamental property of the stochastic process x_t.

Theorem 2.4 (Stochastic interpolant properties)**.** *The probability distribution of the stochastic interpolant x_t defined in (1) is absolutely continuous with respect to the Lebesgue measure for all $t \in [0,1]$, and its time-dependent density $\rho(t)$ satisfies*

$$\rho(0) = \rho_0, \qquad \rho(1) = \rho_1,$$
$$\rho \in C^1\big([0,1]; C^p(\mathbb{R}^d)\big) \quad \text{for any } p \in \mathbb{N}, \qquad \rho(t,x) > 0 \quad \forall (t,x) \in [0,1] \times \mathbb{R}^d.$$

In addition, ρ solves the transport equation

$$\partial_t \rho + \nabla \cdot (b\rho) = 0, \tag{4}$$

where the velocity field is defined by

$$b(t,x) = \mathbb{E}[\dot{x}_t \mid x_t = x] = \mathbb{E}[\partial_t I(t, x_0, x_1) + \dot{\gamma}(t) z \mid x_t = x]\,. \tag{5}$$

The velocity b belongs to $C^0\big([0,1]; C^p(\mathbb{R}^d; \mathbb{R}^d)\big)$ for any $p \in \mathbb{N}$.

Proof. Let

$$g(t,k) := \mathbb{E}\Big[e^{ik\cdot x_t}\Big], \qquad k \in \mathbb{R}^d,$$

be the characteristic function of $\rho(t,x)$. From the definition of x_t in (1), we have

$$g(t,k) = \mathbb{E}\Big[e^{ik\cdot(I(t,x_0,x_1)+\gamma(t)z)}\Big]\,. \tag{6}$$

Using the independence between (x_0, x_1) and z, we obtain

$$g(t,k) = \mathbb{E}\Big[e^{ik\cdot I(t,x_0,x_1)}\Big]\,\mathbb{E}\Big[e^{i\gamma(t)k\cdot z}\Big] = g_0(t,k)\, e^{-\frac{1}{2}\gamma^2(t)|k|^2}, \tag{7}$$

where we defined

$$g_0(t,k) = \mathbb{E}\Big[e^{ik\cdot I(t,x_0,x_1)}\Big]\,.$$

The function $g_0(t,k)$ is the characteristic function of $I(t,x_0,x_1)$ with $(x_0,x_1)\sim\nu$. From (7), we have

$$|g(t,k)| = |g_0(t,k)|e^{-\frac{1}{2}\gamma^2(t)|k|^2} \leq e^{-\frac{1}{2}\gamma^2(t)|k|^2}.$$

Since $\gamma(t)>0$ for all $t\in(0,1)$ by assumption, this shows that

$$\forall\, p\in\mathbb{N} \text{ and } t\in(0,1): \qquad \int_{\mathbb{R}^d}|k|^p|g(t,k)|\,dk<\infty,$$

implying that $\rho(t,\cdot)$ is in $C^p(\mathbb{R}^d)$ for any $p\in\mathbb{N}$ and all $t\in(0,1)$. From (7), we also have

$$\begin{aligned}
|\partial_t g(t,k)|^2 &= \Big|\mathbb{E}\Big[\big(ik\cdot\partial_t I(t,x_0,x_1)-\gamma(t)\dot\gamma(t)|k|^2\big)e^{ik\cdot I(t,x_0,x_1)}\Big]\Big|^2 e^{-\gamma^2(t)|k|^2}\\
&\leq 2\Big(|k|^2\mathbb{E}\big[|\partial_t I(t,x_0,x_1)|^2\big]+|\gamma(t)\dot\gamma(t)|^2|k|^4\Big)e^{-\gamma^2(t)|k|^2}\\
&\leq 2\Big(|k|^2M_1+4|\gamma(t)\dot\gamma(t)|^2|k|^4\Big)e^{-\gamma^2(t)|k|^2},
\end{aligned}$$

and

$$\begin{aligned}
|\partial_t^2 g(t,k)|^2 &\leq 4\Big(|k|^2\mathbb{E}\big[|\partial_t^2 I(t,x_0,x_1)|^2\big]+\big(|\dot\gamma(t)|^2+\gamma(t)\ddot\gamma(t)\big)^2|k|^4\Big)e^{-\gamma^2(t)|k|^2}\\
&\quad+8\Big(|k|^2\mathbb{E}\big[|\partial_t I(t,x_0,x_1)|^4\big]+\big(\gamma(t)\dot\gamma(t)\big)^4|k|^8\Big)e^{-\gamma^2(t)|k|^2}\\
&\leq 4\Big(|k|^2M_2+\big(|\dot\gamma(t)|^2+\gamma(t)\ddot\gamma(t)\big)^2|k|^4\Big)e^{-\gamma^2(t)|k|^2}\\
&\quad+8\Big(|k|^2M_1+\big(\gamma(t)\dot\gamma(t)\big)^4|k|^8\Big)e^{-\gamma^2(t)|k|^2},
\end{aligned}$$

where in both cases we used (3) in Assumption 2.2 to obtain the last inequalities. These imply that

$$\forall\, p\in\mathbb{N} \text{ and } t\in(0,1): \qquad \int_{\mathbb{R}^d}|k|^p|\partial_t g(t,k)|\,dk<\infty, \qquad \int_{\mathbb{R}^d}|k|^p|\partial_t^2 g(t,k)|\,dk<\infty.$$

This shows that $\partial_t\rho(t,\cdot)$ and $\partial_t^2\rho(t,\cdot)$ belong to $C^p(\mathbb{R}^d)$ for any $p\in\mathbb{N}$, i.e. $\rho\in C^1\big((0,1);C^p(\mathbb{R}^d)\big)$ as claimed.

To show that ρ is also positive, denote by $\mu_0(t,dx)$ the unique (by the Fourier inversion theorem) probability measure associated with $g_0(t,k)$, i.e. the measure such that

$$g_0(t,k)=\int_{\mathbb{R}^d}e^{ik\cdot x}\,\mu_0(t,dx).$$

From (7) and the convolution theorem it follows that we can express ρ as

$$\rho(t,x)=\int_{\mathbb{R}^d}\frac{e^{-|x-y|^2/(2\gamma^2(t))}}{(2\pi\gamma^2(t))^{d/2}}\,\mu_0(t,dy).$$

This shows that $\rho>0$ for all $(t,x)\in(0,1)\times\mathbb{R}^d$. Since $x_{t=0}=x_0$ and $x_{t=1}=x_1$ by definition of the interpolant, we also have $\rho(0)=\rho_0$ and $\rho(1)=\rho_1$, which shows that ρ is also positive and in $C^p(\mathbb{R}^d)$ at $t=0,1$ by Assumption 2.2. Note that since $\rho\in C^1\big((0,1);C^p(\mathbb{R}^d)\big)$ and is positive, we immediately deduce that $s=\nabla\log\rho=\nabla\rho/\rho\in C^1\big((0,1);C^p(\mathbb{R}^d;\mathbb{R}^d)\big)$.

To show that ρ satisfies the transport equation (4), we take the time derivative of (6) to deduce that

$$\partial_t g(t,k)=ik\cdot m(t,k), \tag{8}$$

where $m:[0,1]\times\mathbb{R}^d\to\mathbb{C}^d$ is the vector-valued function defined as

$$m(t,k)=\mathbb{E}\Big[\big(\partial_t I(t,x_0,x_1)+\dot\gamma(t)z\big)e^{ik\cdot x_t}\Big].$$

By definition of the conditional expectation, $m(t,k)$ can be expressed as

$$\begin{aligned} m(t,k) &= \int_{\mathbb{R}^d} \mathbb{E}\Big[\big(\partial_t I(t,x_0,x_1) + \dot\gamma(t) z\big) e^{ik\cdot x_t} \,\Big|\, x_t = x\Big]\, \rho(t,x)\, dx \\ &= \int_{\mathbb{R}^d} e^{ik\cdot x} \mathbb{E}[\partial_t I(t,x_0,x_1) + \dot\gamma(t) z \,|\, x_t = x]\, \rho(t,x)\, dx \\ &= \int_{\mathbb{R}^d} e^{ik\cdot x}\, b(t,x)\, \rho(t,x)\, dx, \end{aligned} \tag{9}$$

where the last equality follows from the definition of b in (5). Inserting (9) into (8), we deduce that this equation can be written in real space as the transport equation (4).

Let us now investigate the regularity of b. To that end, we go back to m and use the independence between x_0, x_1, and z, as well as Gaussian integration by parts, to deduce that

$$m(t,k) = \mathbb{E}\Big[\big(\partial_t I(t,x_0,x_1) - i\gamma(t)\dot\gamma(t)k\big) e^{ik\cdot I(t,x_0,x_1)}\Big]\, e^{-\frac{1}{2}\gamma^2(t)|k|^2}.$$

As a result,

$$\begin{aligned} |m(t,k)|^2 &= \Big|\mathbb{E}\Big[\big(\partial_t I(t,x_0,x_1) - i\gamma(t)\dot\gamma(t)k\big) e^{ik\cdot I(t,x_0,x_1)}\Big]\Big|^2 e^{-\gamma^2(t)|k|^2} \\ &\le 2\Big(\mathbb{E}\big[|\partial_t I(t,x_0,x_1)|^2\big] + |\gamma(t)\dot\gamma(t)|^2|k|^2\Big) e^{-\gamma^2(t)|k|^2} \\ &\le 2M_1\, e^{-\gamma^2(t)|k|^2}, \end{aligned}$$

and

$$\begin{aligned} |\partial_t m(t,k)|^2 &\le 4\Big(\mathbb{E}\big[|\partial_t^2 I(t,x_0,x_1)|^2\big] + \big(\gamma(t)\ddot\gamma(t) + \dot\gamma^2(t)\big)^2\Big) e^{-\gamma^2(t)|k|^2} \\ &\quad + 8\Big(|k|^2\mathbb{E}\big[|\partial_t I(t,x_0,x_1)|^4\big] + |\gamma(t)\dot\gamma(t)|^4|k|^4\Big) e^{-\gamma^2(t)|k|^2} \\ &\le 4\Big(M_2 + \big(\gamma(t)\ddot\gamma(t) + \dot\gamma^2(t)\big)^2\Big) e^{-\gamma^2(t)|k|^2} \\ &\quad + 8\Big(|k|^2 M_1 + |\gamma(t)\dot\gamma(t)|^4|k|^4\Big) e^{-\gamma^2(t)|k|^2}, \end{aligned}$$

where in both cases the last inequalities follow from (3). Therefore,

$$\forall\, p \in \mathbb{N},\ t \in (0,1): \qquad \int_{\mathbb{R}^d} |k|^p |m(t,k)|\, dk < \infty, \qquad \int_{\mathbb{R}^d} |k|^p |\partial_t m(t,k)|\, dk < \infty.$$

This implies that the inverse Fourier transform of m is a function $j : [0,1]\times\mathbb{R}^d \to \mathbb{R}^d$ that satisfies

$$j(t,\cdot) \in C^p(\mathbb{R}^d;\mathbb{R}^d) \quad \text{for any } p \in \mathbb{N},\ t \in (0,1),$$

and can be expressed as

$$\begin{aligned} j(t,x) &= (2\pi)^{-d} \int_{\mathbb{R}^d} e^{-ik\cdot x} m(t,k)\, dk \\ &= \mathbb{E}[\partial_t I(t,x_0,x_1) + \dot\gamma(t) z \,|\, x_t = x]\, \rho(t,x) = b(t,x)\rho(t,x), \end{aligned}$$

where the last equality follows from the definition of b in (5). Finally, we deduce that

$$b \in C^0\big([0,1]; C^p(\mathbb{R}^d;\mathbb{R}^d)\big) \quad \text{for any } p \in \mathbb{N},$$

since $j \in C^0\big([0,1]; C^p(\mathbb{R}^d;\mathbb{R}^d)\big)$, $\rho \in C^1\big([0,1]; C^p(\mathbb{R}^d)\big)$, and $\rho > 0$. □

The next theorem identifies the score of ρ and shows that it arises as the unique minimizer of a quadratic objective functional.

Theorem 2.5 (Score)**.** *The score of the probability density ρ specified in Theorem 2.4 belongs to*

$$C^1\big([0,1]; C^p(\mathbb{R}^d;\mathbb{R}^d)\big) \quad \text{for any } p \in \mathbb{N},$$

and is given by

$$s(t,x) = \nabla \log \rho(t,x) = -\gamma^{-1}(t)\,\mathbb{E}[z \mid x_t = x]\,, \qquad \forall (t,x) \in (0,1)\times\mathbb{R}^d. \tag{10}$$

In addition, it satisfies

$$\forall\, t \in [0,1]: \qquad \int_{\mathbb{R}^d} |s(t,x)|^2 \rho(t,x)\,dx < \infty, \tag{11}$$

and is the unique minimizer in $C^1\big([0,1]; C^1(\mathbb{R}^d;\mathbb{R}^d)\big)$ of the quadratic objective

$$\mathcal{L}_s[\hat{s}] = \int_0^1 \mathbb{E}\left[\frac{1}{2}|\hat{s}(t,x_t)|^2 + \gamma^{-1}(t) z\cdot \hat{s}(t,x_t)\right] dt, \tag{12}$$

where x_t is defined in (1) and the expectation is taken independently over $(x_0,x_1)\sim\nu$ and $z\sim\mathcal{N}(0,I_d)$.

Proof. Since $\rho \in C^1((0,1); C^p(\mathbb{R}^d))$ and is positive by Theorem 2.4, we already know that

$$s = \nabla \log \rho = \frac{\nabla \rho}{\rho} \in C^1\big((0,1); C^p(\mathbb{R}^d;\mathbb{R}^d)\big).$$

To establish (10), note that for $t\in(0,1)$ where $\gamma(t)>0$, we have

$$\begin{aligned}\mathbb{E}\Big[z e^{i\gamma(t)k\cdot z}\Big] &= -\gamma^{-1}(t)(i\partial_k)\mathbb{E}\Big[e^{i\gamma(t)k\cdot z}\Big] \\ &= -\gamma^{-1}(t)(i\partial_k) e^{-\frac{1}{2}\gamma^2(t)|k|^2} = i\gamma(t)k\, e^{-\frac{1}{2}\gamma^2(t)|k|^2}.\end{aligned}$$

Using the independence between x_0, x_1 and z, we obtain

$$\mathbb{E}\Big[z e^{ik\cdot x_t}\Big] = i\gamma(t)k\, g(t,k), \tag{13}$$

where g is the characteristic function of x_t defined in (6). Using conditional expectation,

$$\mathbb{E}\Big[z e^{ik\cdot x_t}\Big] = \int_{\mathbb{R}^d} \mathbb{E}[z \mid x_t = x] e^{ik\cdot x}\rho(t,x)\,dx.$$

Since the left-hand side of (13) is the Fourier transform of $-\gamma(t)\nabla\rho(t,x)$, we deduce

$$\mathbb{E}[z \mid x_t = x]\,\rho(t,x) = -\gamma(t)\nabla\rho(t,x) = -\gamma(t)s(t,x)\rho(t,x).$$

Since $\rho(t,x)>0$, this proves (10).

To establish (11), observe that

$$\begin{aligned}\int_{\mathbb{R}^d} |s(t,x)|^2\rho(t,x)\,dx &= \int_{\mathbb{R}^d} \gamma^{-2}(t)\,|\mathbb{E}[z \mid x_t = x]|^2\,\rho(t,x)\,dx \\ &\le \gamma^{-2}(t)\int_{\mathbb{R}^d} \mathbb{E}\big[|z|^2 \mid x_t = x\big]\,\rho(t,x)\,dx \\ &= \gamma^{-2}(t)\mathbb{E}[|z|^2] = d\,\gamma^{-2}(t),\end{aligned}$$

which is finite for $t\in(0,1)$. Continuity at $t=0,1$ follows as in Theorem 2.4, hence (11) holds.

The objective (12) can be written as

$$\begin{aligned}\mathcal{L}_s[\hat{s}] &= \int_0^1\int_{\mathbb{R}^d}\left(\frac{1}{2}|\hat{s}(t,x)|^2 + \gamma^{-1}(t)\mathbb{E}[z \mid x_t = x]\cdot\hat{s}(t,x)\right)\rho(t,x)\,dx\,dt \\ &= \int_0^1\int_{\mathbb{R}^d}\left(\frac{1}{2}|\hat{s}(t,x)|^2 - s(t,x)\cdot\hat{s}(t,x)\right)\rho(t,x)\,dx\,dt,\end{aligned}$$

where we used (10). Completing the square,

$$\begin{aligned}\mathcal{L}_s[\hat{s}] &= \frac{1}{2}\int_0^1 \int_{\mathbb{R}^d} |\hat{s}(t,x) - s(t,x)|^2 \rho(t,x)\,dx\,dt \\ &\quad - \frac{1}{2}\int_0^1 \int_{\mathbb{R}^d} |s(t,x)|^2 \rho(t,x)\,dx\,dt \\ &\geq -\frac{1}{2}\int_0^1 \int_{\mathbb{R}^d} |s(t,x)|^2 \rho(t,x)\,dx\,dt > -\infty,\end{aligned}$$

where the last inequality follows from (11). Since $\rho > 0$, the minimizer is unique and given by $\hat{s} = s$. □

Now we can decompose the velocity b defined by (5) in the following way:

$$b(t,x) = \mathbb{E}[\partial_t I(t,x_0,x_1) \mid x_t = x] - \dot{\gamma}(t)\gamma(t)s(t,x) = v(t,x) - \dot{\gamma}(t)\gamma(t)s(t,x), \tag{14}$$

where s is the score given in (10) and we define the velocity field

$$v(t,x) = \mathbb{E}(\partial_t I(t,x_0,x_1) \mid x_t = x)\,. \tag{15}$$

The velocity field $v \in C^0\big([0,1]; C^p(\mathbb{R}^d;\mathbb{R}^d)\big)$ for any $p \in \mathbb{N}$ and can be characterized (it will follow from Thm. 2.9) as the unique minimizer of the objective

$$\mathcal{L}_v[\hat{v}] = \int_0^1 \mathbb{E}\left(\frac{1}{2}|\hat{v}(t,x_t)|^2 - \partial_t I(t,x_0,x_1)\cdot\hat{v}(t,x_t)\right) dt. \tag{16}$$

Learning this velocity field and the score separately may be useful in practice.

The representation (14) allows us to establish the following property of the velocity field b, thereby completing the statements of Theorem 2.4.

Corollary 2.6. *The velocity b defined by (5) in Theorem 2.4 satisfies*

$$\forall\, t \in [0,1]: \qquad \int_{\mathbb{R}^d} |b(t,x)|^2 \rho(t,x)\,dx < \infty. \tag{17}$$

Proof. By (3), we have

$$\begin{aligned}\int_{\mathbb{R}^d} |b(t,x)|^2\rho(t,x)\,dx &= \int_{\mathbb{R}^d} \mathbb{E}\Big[\big|\partial_t I(t,x_0,x_1) + \dot{\gamma}(t)z\big|^2 \,\Big|\, x_t = x\Big]\,\rho(t,x)\,dx \\ &\leq 2\int_{\mathbb{R}^d} \mathbb{E}\big[|\partial_t I(t,x_0,x_1)|^2 + |\dot{\gamma}(t)|^2|z|^2 \,\big|\, x_t = x\big]\,\rho(t,x)\,dx \\ &\leq 2\,\mathbb{E}\big[|\partial_t I(t,x_0,x_1)|^2 + |\dot{\gamma}(t)|^2|z|^2\big] \\ &< 2M_1^{1/2} + 2d|\dot{\gamma}(t)|^2,\end{aligned} \tag{18}$$

so that this integral is bounded for all $t \in (0,1)$. To analyze its behavior at the endpoints, note that the decomposition (14) implies that

$$\begin{aligned}b_0(x) &:= \lim_{t\to 0} b(t,x) = \mathbb{E}_1\big[\partial_t I(0,x,x_1)\big] - \lim_{t\to 0}\dot{\gamma}(t)\gamma(t)s_0(x), \\ b_1(x) &:= \lim_{t\to 1} b(t,x) = \mathbb{E}_0\big[\partial_t I(1,x_0,x)\big] - \lim_{t\to 1}\dot{\gamma}(t)\gamma(t)s_1(x),\end{aligned}$$

where $s_0 = \nabla\log\rho_0$, $s_1 = \nabla\log\rho_1$, $\mathbb{E}_0$ and $\mathbb{E}_1$ denote expectations over $x_0 \sim \rho_0$ and $x_1 \sim \rho_1$, respectively, and we used the fact that $x_{t=0} = x_0$ and $x_{t=1} = x_1$. Since $\lim_{t\to 0,1}\dot{\gamma}(t)\gamma(t)$ exists by our assumption that $\gamma^2 \in C^1([0,1])$, b_0 and b_1 are well defined, and

$$\int_{\mathbb{R}^d} |b_0(x)|^2\rho_0(x)\,dx < \infty, \qquad \int_{\mathbb{R}^d} |b_1(x)|^2\rho_1(x)\,dx < \infty,$$

by Assumption 2.2. As a result, the integral in (18) is continuous at $t = 0$ and $t = 1$, so it must be integrable on $[0,1]$, and (17) holds. □

The availability of the score permits a reformulation of the transport equation (4) into the *forward* and *backward Fokker–Planck equations*, which will be used in the subsequent analysis.

Corollary 2.7 (Fokker–Planck equations)**.** *For any $\varepsilon \in C^0([0,1])$ with $\varepsilon(t) \geq 0$ for all $t \in [0,1]$, the probability density ρ specified in Theorem 2.4 satisfies:*

1. ***The forward Fokker–Planck equation***

$$\partial_t \rho + \nabla \cdot (b_F \rho) = \varepsilon(t) \Delta \rho, \qquad \rho(0) = \rho_0, \tag{19}$$

where the forward drift is defined by

$$b_F(t,x) = b(t,x) + \varepsilon(t) s(t,x). \tag{20}$$

Equation (19) is well-posed when solved forward in time from $t = 0$ to $t = 1$, and its solution with initial condition $\rho(t = 0) = \rho_0$ satisfies $\rho(1) = \rho_1$.

2. ***The backward Fokker–Planck equation***

$$\partial_t \rho + \nabla \cdot (b_B \rho) = -\varepsilon(t) \Delta \rho, \qquad \rho(1) = \rho_1, \tag{21}$$

where the backward drift is defined by

$$b_B(t,x) = b(t,x) - \varepsilon(t) s(t,x). \tag{22}$$

Equation (21) is well-posed when solved backward in time from $t = 1$ to $t = 0$, and its solution with final condition $\rho(t = 1) = \rho_1$ satisfies $\rho(0) = \rho_0$.

Proof. The forward equation (19) and the backward equation (21) are direct consequences of the transport equation (4) and the score identity (10).

Indeed, since

$$\Delta \rho = \nabla \cdot (\nabla \rho) = \nabla \cdot (\rho \nabla \log \rho) = \nabla \cdot (\rho s),$$

we have

$$\varepsilon(t) \Delta \rho = \varepsilon(t) \nabla \cdot (\rho s),$$

which allows one to convert between

$$\partial_t \rho + \nabla \cdot (b \rho) = 0$$

and the forward/backward Fokker–Planck forms by absorbing $\pm \varepsilon(t) s$ into the drift. □

Taking into account the correspondence between the Fokker–Planck equations (19), (21) and the associated stochastic differential equations, we arrive at the following result, which plays a central role in practical applications.

Corollary 2.8 (Generative models)**.** *At any time $t \in [0,1]$, the law of the stochastic interpolant x_t coincides with the law of the three processes X_t, X_t^{F}, and X_t^{B}, defined as follows:*

1. The solutions of the probability flow associated with the transport equation (4)

$$\frac{d}{dt} X_t = b(t, X_t), \tag{23}$$

where b is defined by (5), solved either forward in time from the initial data $X_{t=0} \sim \rho_0$ or backward in time from the final data $X_{t=1} = x_1 \sim \rho_1$;

2. The solutions of the forward SDE associated with the Fokker–Planck equation (19)

$$dX_t^{\mathrm{F}} = b_{\mathrm{F}}(t, X_t^{\mathrm{F}})\,dt + \sqrt{2\varepsilon(t)}\,dW_t, \tag{24}$$

where b_{F} is defined by (20), solved forward in time from the initial data $X_{t=0}^{\mathrm{F}} \sim \rho_0$, independent of the standard Brownian motion W;

3. The solutions of the backward SDE associated with the backward Fokker–Planck equation (21)

$$dX_t^{\mathrm{B}} = b_{\mathrm{B}}(t, X_t^{\mathrm{B}})\,dt + \sqrt{2\varepsilon(t)}\,dW_t^{\mathrm{B}}, \qquad W_t^{\mathrm{B}} = -W_{1-t}, \tag{25}$$

where b_{B} is defined by (22), solved backward in time from the final data $X_{t=1}^{\mathrm{B}} \sim \rho_1$, independent of the standard Brownian motion W.

The solution of (25) is by definition $X_t^{\mathrm{B}} = Z_{1-t}^{\mathrm{F}}$, where Z_t^{F} satisfies

$$dZ_t^{\mathrm{F}} = -b_{\mathrm{B}}(1-t, Z_t^{\mathrm{F}})\,dt + \sqrt{2\varepsilon(t)}\,dW_t,$$

solved forward in time from the initial data $Z_{t=0}^{\mathrm{F}} \sim \rho_1$, independent of W.

Finally, following the approach of Theorem 2.5, we prove the next result, which further enables us to obtain quantitative bounds on the Kullback–Leibler divergence between the ground-true distribution and the approximated one, a feature that is essential for practical applications.

Theorem 2.9 (Objective)**.** *The velocity b defined in (5) is the unique minimizer in $C^0\big([0,1]; C^1(\mathbb{R}^d;\mathbb{R}^d)\big)$ of the quadratic objective*

$$\mathcal{L}_b[\hat{b}] = \int_0^1 \mathbb{E}\left[\frac{1}{2}|\hat{b}(t,x_t)|^2 - \big(\partial_t I(t,x_0,x_1) + \dot{\gamma}(t)z\big)\cdot\hat{b}(t,x_t)\right] dt, \tag{26}$$

where x_t is defined in (1) and the expectation is taken independently over $(x_0,x_1)\sim\nu$ and $z\sim\mathcal{N}(0,I_d)$.

Proof. By definition of ρ, the objective $\mathcal{L}_b$ defined in (26) can be written as

$$\begin{aligned}\mathcal{L}_b[\hat{b}] &= \int_0^1\int_{\mathbb{R}^d}\Big(\frac{1}{2}|\hat{b}(t,x)|^2 - \mathbb{E}[\partial_t I(t,x_0,x_1) + \dot{\gamma}(t)z \,|\, x_t = x]\cdot\hat{b}(t,x)\Big)\rho(t,x)\,dx\,dt \\ &= \int_0^1\int_{\mathbb{R}^d}\Big(\frac{1}{2}|\hat{b}(t,x)|^2 - b(t,x)\cdot\hat{b}(t,x)\Big)\rho(t,x)\,dx\,dt,\end{aligned} \tag{27}$$

where we used the definition of b in (5). This quadratic objective is bounded from below since

$$\begin{aligned}\mathcal{L}_b[\hat{b}] &= \frac{1}{2}\int_0^1\int_{\mathbb{R}^d}|\hat{b}(t,x)-b(t,x)|^2\rho(t,x)\,dx\,dt - \frac{1}{2}\int_0^1\int_{\mathbb{R}^d}|b(t,x)|^2\rho(t,x)\,dx\,dt \\ &\geq -\frac{1}{2}\int_0^1\int_{\mathbb{R}^d}|b(t,x)|^2\rho(t,x)\,dx\,dt > -\infty,\end{aligned}$$

where the finiteness follows from (17). Since $\rho(t,x) > 0$, we conclude $\hat{b} = b$ is the unique minimizer of (27). □

2.3 Likelihood Control

It can be shown (Albergo et al., 2025, Lemma 22) that, unlike in the case of transport equations, the KL-divergence between the solutions of two Fokker–Planck equations is controlled by the error in their drifts.

We can now state the following result, which demonstrates that the losses (12) and (26) control the likelihood of learned approximations to the FPE (19).

Theorem 2.10. *Let ρ denote the solution of the Fokker–Planck equation (19) with $\varepsilon(t) = \varepsilon > 0$ constant. Given two velocity fields $\hat{b}, \hat{s} \in C^0\big([0,1]; C^1(\mathbb{R}^d;\mathbb{R}^d)\big)$, define*

$$\hat{b}_{\mathrm{F}}(t,x) = \hat{b}(t,x) + \varepsilon \hat{s}(t,x), \qquad \hat{v}(t,x) = \hat{b}(t,x) + \gamma(t)\dot{\gamma}(t)\hat{s}(t,x).$$

Assume that the function γ satisfies the properties listed in Definition 2.1. Let $\hat{\rho}$ denote the solution to the Fokker–Planck equation

$$\partial_t \hat{\rho} + \nabla \cdot (\hat{b}_{\mathrm{F}} \hat{\rho}) = \varepsilon \Delta \hat{\rho}, \qquad \hat{\rho}(0) = \rho_0.$$

Then

$$\mathrm{KL}(\rho_1 \mid \hat{\rho}(1)) \le \frac{1}{2\varepsilon}\left(\mathcal{L}_b[\hat{b}] - \min_{\hat{b}} \mathcal{L}_b[\hat{b}]\right) + \frac{\varepsilon}{2}\left(\mathcal{L}_s[\hat{s}] - \min_{\hat{s}} \mathcal{L}_s[\hat{s}]\right),$$

where $\mathcal{L}_b[\hat{b}]$ and $\mathcal{L}_s[\hat{s}]$ are the objective functions defined in (26) and (12). Moreover,

$$\mathrm{KL}(\rho_1 \mid \hat{\rho}(1)) \le \frac{1}{2\varepsilon}\left(\mathcal{L}_v[\hat{v}] - \min_{\hat{v}} \mathcal{L}_v[\hat{v}]\right) + \frac{\sup_{t\in[0,1]}(\gamma(t)\dot{\gamma}(t) - \varepsilon)^2}{2\varepsilon}\left(\mathcal{L}_s[\hat{s}] - \min_{\hat{s}} \mathcal{L}_s[\hat{s}]\right),$$

where $\mathcal{L}_v[\hat{v}]$ is the objective function defined in (16).

2.4 Spatially Linear One-Sided Interpolants

Much of the discussion above generalizes to the so called one-sided linear interpolants, which are defined as follow

$$x_t^{\mathrm{os,lin}} = \alpha(t) z + \beta(t) x_1, \qquad t \in [0,1],$$

where $\alpha^2, \beta \in C^2([0,1])$ and $\alpha(0) = \beta(1) = 1, \alpha(1) = \beta(0) = 0$, and $\alpha(t) > 0$ for all $t \in [0,1]$. We focus on this particular class of interpolants due to their significant practical relevance.

The velocity b and the score s defined in (5) and (10) can now be expressed as

$$b(t,x) = \dot{\alpha}(t)\eta_z^{\mathrm{os}}(t,x) + \dot{\beta}(t)\eta_1^{\mathrm{os}}(t,x), \qquad s(t,x) = -\alpha^{-1}(t)\eta_z^{\mathrm{os}}(t,x), \tag{28}$$

where the second expression holds for all $t \in [0,1]$ and we defined:

$$\eta_z^{\mathrm{os}}(t,x) = \mathbb{E}[z \mid x_t^{\mathrm{os,lin}} = x], \qquad \eta_1^{\mathrm{os}}(t,x) = \mathbb{E}[x_1 \mid x_t^{\mathrm{os,lin}} = x]. \tag{29}$$

Note that, by definition of the conditional expectation, η_z^{os} and η_1^{os} satisfy

$$\forall (t,x) \in [0,1] \times \mathbb{R}^d : \qquad \alpha(t)\eta_z^{\mathrm{os}}(t,x) + \beta(t)\eta_1^{\mathrm{os}}(t,x) = x. \tag{30}$$

As a result, only one of them needs to be estimated. For example, we can express η_1^{os} as a function of η_z^{os} for all t such that $\beta(t) \neq 0$, and use the result to express the velocity (28) as

$$b(t,x) = \dot{\beta}(t)\beta^{-1}(t)x + \left(\dot{\alpha}(t) - \alpha(t)\dot{\beta}(t)\beta^{-1}(t)\right)\eta_z^{\mathrm{os}}(t,x), \qquad \forall t:\ \beta(t) \neq 0. \tag{31}$$

Assuming that $\beta(t) \neq 0$ for all $t \in (0,1]$, this formula only needs to be supplemented at $t = 0$ with

$$b(0,x) = \dot{\alpha}(0)x + \dot{\beta}(0)\mathbb{E}[x_1],$$

which follows from (28) since $x_{t=0}^{\mathrm{os,lin}} = z$.

Finally note that η_z and/or η_1 can be estimated using the following two objective functions, respectively:

$$\mathcal{L}_{\eta_z}(\hat{\eta}_z^{\mathrm{os}}) = \int_0^1 \mathbb{E}\left[\frac{1}{2}\left\|\hat{\eta}_z^{\mathrm{os}}(t, x_t^{\mathrm{os,lin}}) - z\right\|^2\right] dt,$$

$$\mathcal{L}_{\eta_1}(\hat{\eta}_1^{\mathrm{os}}) = \int_0^1 \mathbb{E}\left[\frac{1}{2}\left\|\hat{\eta}_1^{\mathrm{os}}(t, x_t^{\mathrm{os,lin}}) - x_1\right\|^2\right] dt.$$

Now we can prove the main practical result, which allows us to simulate the stochastic interpolants approach on the computer. Taking infinitesimal steps, we obtain a generative model consistent with the probability flow equation (23) associated with $x_t^{\mathrm{os,lin}}$.

Theorem 45. Let $t_j = j/N$ with $j \in \{1, \dots, N\}$, set $X_1^{\mathrm{den}} = z$, and define for $j = 1, \dots, N-1$,

$$X_{j+1}^{\mathrm{den}} = \frac{\beta(t_{j+1})}{\beta(t_j)} X_j^{\mathrm{den}} + \left(\alpha(t_{j+1}) - \frac{\alpha(t_j)\beta(t_{j+1})}{\beta(t_j)} \right) \eta_z^{\mathrm{os}}(t_j, X_j^{\mathrm{den}}). \tag{32}$$

Then (32) is a consistent integration scheme for the probability flow equation (23) associated with the velocity field (28) expressed as in (31). That is, if $N, j \to \infty$ with $j/N \to t \in [0,1]$, then $X_j^{\mathrm{den}} \to X_t$ where

$$\dot{X}_t = b(t, X_t) = \frac{\dot{\beta}(t)}{\beta(t)} X_t + \left(\dot{\alpha}(t) - \frac{\alpha(t)\dot{\beta}(t)}{\beta(t)} \right) \eta_z^{\mathrm{os}}(t, X_t), \qquad X_{t=0} = z. \tag{33}$$

In particular, if $z \sim \mathcal{N}(0, I_d)$, then $X_N^{\mathrm{den}} \to x_1 \sim \rho_1$ in this limit.

Proof. Use

$$\beta(t_{j+1})\beta^{-1}(t_j) = 1 + \dot{\beta}(t_j)\beta^{-1}(t_j)(t_{j+1} - t_j) + O\big((t_{j+1} - t_j)^2\big)$$

and

$$\alpha(t_{j+1}) - \alpha(t_j)\beta(t_{j+1})\beta^{-1}(t_j) = \Big(\dot{\alpha}(t_j) - \alpha(t_j)\dot{\beta}(t_j)\beta^{-1}(t_j) \Big) (t_{j+1} - t_j) + O\big((t_{j+1} - t_j)^2\big)$$

to deduce that (32) implies

$$\begin{aligned} X_{j+1}^{\mathrm{den}} = X_j^{\mathrm{den}} &+ \dot{\beta}(t_j)\beta^{-1}(t_j) X_j^{\mathrm{den}} (t_{j+1} - t_j) \\ &+ \Big(\dot{\alpha}(t_j) - \alpha(t_j)\dot{\beta}(t_j)\beta^{-1}(t_j) \Big) \eta_z^{\mathrm{os}}(t_j, X_j^{\mathrm{den}})(t_{j+1} - t_j) + O\big((t_{j+1} - t_j)^2\big). \end{aligned}$$

Or equivalently,

$$\frac{X_{j+1}^{\mathrm{den}} - X_j^{\mathrm{den}}}{t_{j+1} - t_j} = \dot{\beta}(t_j)\beta^{-1}(t_j) X_j^{\mathrm{den}} + \Big(\dot{\alpha}(t_j) - \alpha(t_j)\dot{\beta}(t_j)\beta^{-1}(t_j) \Big) \eta_z^{\mathrm{os}}(t_j, X_j^{\mathrm{den}}) + O(t_{j+1} - t_j).$$

Taking the limit as $N, j \to \infty$ with $j/N \to t \in [0,1]$, we recover (33) and deduce that $X_j^{\mathrm{den}} \to X_t$.

3 Architecture of MDGen

In this section, we examine the architecture of a contemporary diffusion-based model for molecular dynamics generation, specifically MDGen. The implementation and detailed usage instructions are described in (Jing, Stärk, et al., 2024) and made publicly available through the project repository.

3.1 Tokenizing Molecular Trajectories

Firstly, we need to represent a polypeptide chain in three-dimensional space as a mathematical object. The most straightforward approach is to use the coordinates of all atoms in the protein, thereby forming a vector in $\mathbb{R}^{3N}$, where N denotes the total number of atoms. However, this representation has a fundamental drawback. The physical properties of a protein do not depend on its position or orientation in space: translations and rotations of the molecule do not change its intrinsic structure. Therefore, a suitable mathematical representation must be invariant under rigid motions in $\mathbb{R}^3$.

We now describe a possible construction of such a representation introduces in (Yim et al., 2023). The authors adopted the backbone frame parameterization used in AlphaFold2 (Jumper et al., 2021). Here, an L-residue backbone is parameterized by a collection of L orientation-preserving rigid transformations, or *frames*, that map from fixed coordinates $\mathrm{N}^*, \mathrm{C}_\alpha^*, \mathrm{C}^*, \mathrm{O}^* \in \mathbb{R}^3$ centered at $\mathrm{C}_\alpha^* = (0,0,0)$ ([2]Fig. 1).

[2]Image source: (Yim et al., 2023, Figure 1A)

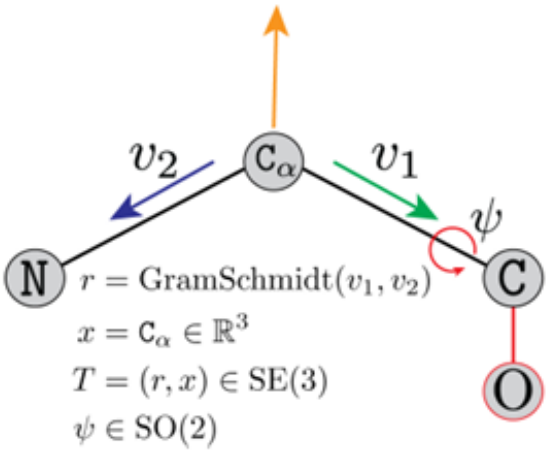

Figure 1: A frame.

Each fixed coordinate assumes chemically idealized bond angles and lengths measured experimentally. These values differ slightly per amino acid type. Since we do not model sequence, we can take the idealized values of Alanine, which are

$$\mathrm{N}^* = (-0.525,\ 1.363,\ 0.0),$$
$$\mathrm{C}_\alpha^* = (0.0,\ 0.0,\ 0.0),$$
$$\mathrm{C}^* = (1.526,\ 0.0,\ 0.0),$$
$$\mathrm{O}^* = (0.627,\ 1.062,\ 0.0).$$

For each residue indexed by l, the backbone main atom coordinates are given by

$$[\mathrm{N}_l, \mathrm{C}_l, (\mathrm{C}_\alpha)_l] = g_l \cdot [\mathrm{N}^*, \mathrm{C}^*, \mathrm{C}_\alpha^*],$$

where g_l is a member of the special Euclidean group $\mathrm{SE}(3)$, the set of orientation-preserving rigid transformations in Euclidean space. Each g_l may be decomposed into two components $g_l = (R_l, x_l)$ where $R_l \in \mathrm{SO}(3)$ is a 3×3 rotation matrix and $x_l \in \mathbb{R}^3$ represents a translation. For a vector $v \in \mathbb{R}^3$, the action of g_l is given by

$$g_l \cdot v = R_l v + x_l.$$

With an additional torsion angle $\psi_l \in \mathrm{SO}(2)$, we can construct the backbone oxygen by rotating O^* around the $\mathrm{C} - \mathrm{C}_\alpha$ bond:

$$\mathrm{O}_l = g_l \cdot g_{\text{psi}}^*(\psi_l) \cdot \mathrm{O}^*,$$

where $g_{\text{psi}}^*(\psi_l) = (R_x(\psi_l), x_{\text{psi}})$ is a Euclidean transformation from the central frame g_l to a new frame $g_l \cdot g_{\text{psi}}^*$ centered at C and rotated around the x-axis by ψ_l,

$$R_x(\psi_l) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \psi_{l,1} & -\psi_{l,2} \\ 0 & \psi_{l,2} & \psi_{l,1} \end{pmatrix},$$

$$x_{\text{psi}} = (1.526,\ 0.0,\ 0.0).$$

Recall ψ_l is a point on the unit circle, $\psi_l = [\psi_{l,1}, \psi_{l,2}]$ where $(\psi_{l,1})^2 + (\psi_{l,2})^2 = 1$.

The mapping from frames to idealized coordinates, `frame2atom`, is given by

$$[\mathrm{N}_l, \mathrm{C}_l, (\mathrm{C}_\alpha)_l, \mathrm{O}_l] = \text{frame2atom}(g_l, \psi_l).$$

We next describe constructing frames from coordinates. Each residue's frames are obtained as sketched in Fig. 1 and the `rigidFrom3Point` algorithm in AlphaFold2 (Jumper et al., 2021),

$$v_1 = \mathrm{C}_l - (\mathrm{C}_\alpha)_l, \qquad v_2 = \mathrm{N}_l - (\mathrm{C}_\alpha)_l,$$
$$e_1 = \frac{v_1}{\|v_1\|}, \qquad u_2 = v_2 - e_1 \left(e_1^\top v_2\right),$$
$$e_2 = \frac{u_2}{\|u_2\|}, \qquad e_3 = e_1 \times e_2.$$

$$R_l = \mathrm{concat}(e_1, e_2, e_3), \qquad x_l = (\mathrm{C}_\alpha)_l, \qquad g_l = (R_l, x_l).$$

where the first four lines follow from Gram–Schmidt. The operation of going from coordinates to frames is called `atom2frame`,

$$g_l = \mathrm{atom2frame}(\mathrm{N}_l, \mathrm{C}_l, (\mathrm{C}_\alpha)_l).$$

So far, using Alanine as an illustrative example, we conclude that an amino acid residue can be represented by an element $g \in \mathrm{SE}(3)$ of the special Euclidean group together with a torsion angle $\psi \in \mathbb{S}^1$.

For other amino acids, additional atomic positions besides the oxygen atom must be specified. In general, the number of torsion angles required to describe a residue may be as large as seven.

Given a chemical specification of a molecular system with L amino acid residuals developing in T time steps, we have the following representation of the molecular trajectory:

$$\mathcal{G}_l^t = \big((R_l, x_l),\ (\psi_l, \phi_l, \omega_l, \chi_{(l,1)}, \ldots, \chi_{(l,4)})\big), \qquad \mathcal{G} \in \Big(\big[\mathrm{SE}(3) \times \mathbb{T}^7\big]^L\Big)^T,$$

where subscripts indicate residue and superscripts time indices. The undefined torsion angles can be randomized and are unsupervised for residues with fewer than seven torsion angles.

To learn a generative model on the space of roto-translations and torsion angles, we exploit the fact that the problem can be formulated as one of *conditional trajectory generation*. In particular, each trajectory contains at least one so-called *key frame*, whose roto-translations are known initially. Since these key frames are given, they do not need to be generated and can instead be used as references in the modeling process. The roto-translations of the remaining structures are therefore represented as *offsets relative to the key frames*.

More precisely, given K key frames $t_1, \ldots, t_K$, we tokenize residue l in frame t as follows:

$${}^K\mathcal{G}_l^t = \Big([g_l^{t_1}]^{-1} g_l^t, \ldots, [g_l^{t_K}]^{-1} g_l^t, {\tau_l}^t\Big) \in \mathrm{SE}(3)^K \times \mathbb{T}^7 = (\hat{\mathbb{Q}}^+ \oplus \mathbb{R}^3)^K \times (\mathbb{S}^1)^7 \subset \mathbb{R}^{7K+14}.$$

where $g_l^t \in \mathrm{SE}(3)$ represents the roto-translation and ${\tau_l}^t$ the torsion angles of residue l at frame t.

Namely, we convert the relative roto-translational offsets $[g_l^{t_i}]^{-1} g_l^t$ to unit quaternions with positive real part $\hat{\mathbb{Q}}^+ \subset \mathbb{R}^4$ and translation vectors in $\mathbb{R}^3$, and convert torsion angles to points on the unit circle, obtaining a $(7K+14)$-dimensional token for each residue in every frame. The offsets and torsion angles are $\mathrm{SE}(3)$-invariant; thus, we obtain a representation of molecular trajectories as an $(T \times L)$-array of $\mathrm{SE}(3)$-invariant tokens.

3.2 The Main Blocks

We now discuss the main mechanisms underlying the MDGen architecture, as well as several essential building blocks of the model.

Diffusion Block. The generative model for MD trajectories is formulated within the stochastic interpolants framework described in Section 2.4. Given a continuous distribution $\rho_1 \equiv p_{\mathrm{data}}$ over $\mathbb{R}^n$, consider the following one-sided linear stochastic interpolant

$$x_t = \alpha_t z + \beta_t x_1$$

transporting a prior distribution $\rho_0 = \mathcal{N}(0, I)$ to the data ρ_1, where $x_1 \sim \rho_1$ and the interpolation path satisfies $\alpha_0 = \beta_1 = 1$ and $\alpha_1 = \beta_0 = 0$. The aim is to construct a neural network $v_\theta \colon [0,1] \times \mathbb{R}^n \to \mathbb{R}^n$ is trained to approximate the time-evolving flow field defined by (15):

$$v_\theta(t, x_t) \approx \eta_1^{\mathrm{os}}(t, x_t) \equiv \mathbb{E}_{z, x_1 | x_t}\big[\dot{\alpha}_t z + \dot{\beta}_t x_1\big].$$

Using (28) and (30) we can derive the exact formula for the score function (Ma et al., 2024, Appendix A.4) and then use its neural network approximation:

$$s_\theta(t, x_t) = \alpha_t^{-1} \frac{\beta_t\, v_\theta(t, x_t) - \dot{\beta}_t\, x_t}{\dot{\beta}_t \alpha_t - \beta_t \dot{\alpha}_t}. \tag{34}$$

Our base modeling task is to generate a distribution over $\mathbb{R}^{T\times L\times(7K+14)}$ conditioned on roto-translations of one or more key frames $g^{t_1},\dots,g^{t_K}$ and amino acid identities $\mathcal{A}$. To do so, we learn parameterize a velocity network

$$v_\theta(\cdot \mid g^{t_1},\dots,g^{t_K},\mathcal{A})\colon [0,1]\times\mathbb{R}^{T\times L\times(7K+14)}\to\mathbb{R}^{T\times L\times(7K+14)}. \tag{35}$$

The architecture of such a network will be discussed in the subsequent paragraphs.

Once the velocity field has been trained, trajectory generation can be performed using standard deterministic numerical integration methods applied to (32).

Another approach to generating MD trajectories is based on the Fokker–Planck equation (24), which in this setting takes the form

$$dX_t=\eta_1^{\mathrm{os}}(t,X_t)\,dt+\varepsilon_t\eta_z^{\mathrm{os}}(t,X_t)\,dt+\sqrt{2\varepsilon_t}\,dW_t,$$

here η_z^{os} and η_1^{os} are defined in (29). One can use a stochastic numerical integrator, for example Alg. 1 described in (Ma et al., 2024, Appendix E).

Algorithm 1 Stochastic Euler–Maruyama Sampler

procedure EULERSAMPLER($v_\theta(t,x,y)$, ε_t, $\{t_i\}_{i=0}^N$, T, α_t, σ_t) ▷ y denotes a conditioning here
  Sample $x_0\sim\mathcal{N}(0,I)$ ▷ Generate initial sample
  $s_\theta\leftarrow$ convert from v_θ using (34)
  $\Delta t\leftarrow t_1-t_0$ ▷ Fixed step size
  for $i=0$ to $N-1$ **do**
    Sample $\epsilon_i\sim\mathcal{N}(0,I)$
    $d\epsilon_i\leftarrow\epsilon_i\sqrt{\Delta t}$
    $d_i\leftarrow v_\theta(t_i,x_i,y)+\varepsilon_{t_i}s_\theta(t_i,x_i,y)$ ▷ Drift at t_i
    $x_{i+1}\leftarrow x_i+\Delta t\,d_i+\sqrt{2\varepsilon_{t_i}}\,d\epsilon_i$ ▷ Diffusion at t_i
  end for
  return x
end procedure

Attention with rotary position embeddings. The remarkable performance of modern generative models can largely be attributed to the attention mechanism introduced by Vaswani et al. (2023). We briefly review this mechanism following Su et al. (2023).

Let $S_N=(w_1,\dots,w_N)$ be a sequence of N input tokens. We denote by $E_N=(x_1,\dots,x_N)$ the corresponding sequence of embeddings, where each $x_i\in\mathbb{R}^d$ is the d-dimensional vector representation of token w_i, prior to the addition of positional encodings.

Self-attention first incorporates positional information into the word embeddings and transforms them into queries, keys, and value representations:

$$\begin{aligned}q_m&=f_q(x_m,m),\\ k_n&=f_k(x_n,n),\\ v_n&=f_v(x_n,n),\end{aligned} \tag{36}$$

where q_m, k_n, and v_n incorporate the m-th and n-th positions through f_q, f_k, and f_v, respectively. The query and key vectors are then used to compute the attention weights, and the output is computed as the weighted sum over the value representations:

$$a_{m,n}=\frac{\exp\left(\frac{q_m^\top k_n}{\sqrt{d}}\right)}{\sum_{j=1}^N\exp\left(\frac{q_m^\top k_j}{\sqrt{d}}\right)}, \tag{37}$$

$$o_m=\sum_{n=1}^N a_{m,n}v_n. \tag{38}$$

After that the output vectors o_m defined by (38) are used in the learning process.

While the in the standard self-attention architecture, positional encodings are added to the input embeddings in order to inject information about token order, such additive encodings do not explicitly model relative positional relationships within the attention operation itself.

To address this limitation, *Rotary Positional Embeddings* (RoPE) introduce positional information directly into the attention mechanism by applying a position-dependent rotation to the query and key representations. This construction preserves the inner-product structure underlying attention while naturally encoding relative positional information through phase differences. As a result, Attention with RoPE provides improved extrapolation to longer sequences and a more principled integration of positional structure into the attention computation.

More precisely, for any $x_i \in \mathbb{R}^d$ with even d, we divide the d-dimensional space into $d/2$ two-dimensional subspaces and consider $f_{\{q,k\}}$ from (36) in the form

$$f_{\{q,k\}}(x_m, m) = R^d_{\Omega,m} W_{\{q,k\}} x_m,$$

where

$$R^d_{\Omega,m} = \begin{pmatrix} \cos m\omega_1 & -\sin m\omega_1 & 0 & 0 & \cdots & 0 & 0 \\ \sin m\omega_1 & \cos m\omega_1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cos m\omega_2 & -\sin m\omega_2 & \cdots & 0 & 0 \\ 0 & 0 & \sin m\omega_2 & \cos m\omega_2 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & \cos m\omega_{d/2} & -\sin m\omega_{d/2} \\ 0 & 0 & 0 & 0 & \cdots & \sin m\omega_{d/2} & \cos m\omega_{d/2} \end{pmatrix}.$$

This is a block-diagonal rotation matrix with predefined parameters

$$\Omega = \left\{ \omega_i = 10000^{-2(i-1)/d} \;\middle|\; i = 1, 2, \ldots, d/2 \right\}.$$

Applying RoPE to the scalar product in Eq. (37), we obtain

$$q_m^\top k_n = \left(R^d_{\Omega,m} W_q x_m\right)^\top \left(R^d_{\Omega,n} W_k x_n\right) = x_m^\top W_q^\top R^d_{\Omega,n-m} W_k x_n,$$

where we used the fact that R^d_Ω is an orthogonal matrix:

$$R^d_{\Omega,n-m} = \left(R^d_{\Omega,m}\right)^\top R^d_{\Omega,n}.$$

Finally, the output vectors (38) have the form

$$o_m = \sum_{n=1}^{N} \frac{\exp\left(\frac{x_m^\top W_q^\top R^d_{\Omega,n-m} W_k x_n}{\sqrt{d}}\right) v_n}{\sum_{j=1}^{N} \exp\left(\frac{x_m^\top W_q^\top R^d_{\Omega,j-m} W_k x_j}{\sqrt{d}}\right)}. \tag{39}$$

Invariant Point Attention. We now consider another modification of the attention mechanism, namely the so-called *Invariant Point Attention* (IPA), introduced in Jumper et al. (2021, Supplementary Material, Algorithm 22). The key motivation behind this construction is to incorporate geometric structure directly into the attention computation while preserving invariance with respect to rigid motions in three-dimensional space. In contrast to standard attention, which operates purely in the space of feature vectors, IPA augments the representation with learnable points embedded in $\mathbb{R}^3$. The attention weights are then influenced not only by feature similarity but also by the relative spatial configuration of these points. This enables

the model to reason about three-dimensional structure in a manner that is equivariant under rotations and translations, making it particularly suitable for molecular and protein modeling tasks.

We now introduce the notation used in Alg. 2. Let $\{\mathbf{s}_i\}_{i=1}^{L}$ denote the single per-residue embeddings and $\{\mathbf{z}_{ij}\}_{i,j=1}^{L}$ the pair representations corresponding, for example, to electrostatic interactions. Here, as usual, L is the total number of residues in a polypeptide chain. Each residue i is additionally associated with a rigid-body transformation $g_i \in \mathrm{SE}(3)$, which maps points from the local coordinate frame of the residue to the global three-dimensional coordinate system.

For each attention head h, the single representations $\mathbf{s}_i$ are linearly projected to scalar queries, keys, and values $\mathbf{q}_i^h, \mathbf{k}_i^h, \mathbf{v}_i^h \in \mathbb{R}^c$, as well as to sets of learnable three-dimensional query and key points $\{\vec{\mathbf{q}}_i^{\,hp}\}$ and $\{\vec{\mathbf{k}}_i^{\,hp}\}$ whit $p = 1, \ldots, N_{\text{query points}}$, and three-dimensional value points $\{\vec{\mathbf{v}}_i^{\,hp}\}$ with $p = 1, \ldots, N_{\text{point values}}$. These points are defined in the local residue frame and are transformed to global coordinates via g_i during the attention computation. The attention weights are computed by combining three contributions:

- the standard dot-product similarity between scalar queries and keys,
- a pairwise bias term derived from $\mathbf{z}_{ij}$, and
- a geometric term depending on the squared Euclidean distances between the transformed query and key points in $\mathbb{R}^3$.

This geometric component ensures that the attention mechanism is sensitive to spatial relationships while remaining equivariant with respect to rigid motions.

Finally, the outputs (Alg. 2, lines 7–9) are obtained by aggregating both the scalar values and the transformed value points using the computed attention weights. The resulting features are concatenated and projected to produce the updated single representations $\{\tilde{\mathbf{s}}_i\}$.

Algorithm 2 Invariant Point Attention

1: **Input:** $\{\mathbf{s}_i\}, \{\mathbf{z}_{ij}\}, \{g_i\}, N_{\text{head}} = 12, c = 16, N_{\text{query points}} = 4, N_{\text{point values}} = 8$
2: $\mathbf{q}_i^h, \mathbf{k}_i^h, \mathbf{v}_i^h = \mathrm{LinearNoBias}(\mathbf{s}_i)$ ▷ $\mathbf{q}_i^h, \mathbf{k}_i^h, \mathbf{v}_i^h \in \mathbb{R}^c,\ h = 1, \ldots, N_{\text{head}}$
3: $\vec{\mathbf{q}}_i^{\,hp}, \vec{\mathbf{k}}_i^{\,hp} = \mathrm{LinearNoBias}(\mathbf{s}_i)$ ▷ $\vec{\mathbf{q}}_i^{\,hp}, \vec{\mathbf{k}}_i^{\,hp} \in \mathbb{R}^3,\ p = 1, \ldots, N_{\text{query points}}$
4: $\vec{\mathbf{v}}_i^{\,hp} = \mathrm{LinearNoBias}(\mathbf{s}_i)$ ▷ $\vec{\mathbf{v}}_i^{\,hp} \in \mathbb{R}^3,\ p = 1, \ldots, N_{\text{point values}}$
5: $b_{ij}^h = \mathrm{LinearNoBias}(\mathbf{z}_{ij})$
6: $a_{ij}^h = \mathrm{softmax}_j\left(\frac{1}{\sqrt{c}}\mathbf{q}_i^{h\top}\mathbf{k}_j^h + b_{ij}^h - \frac{1}{2}\sum_p \left\| g_i \circ \vec{\mathbf{q}}_i^{\,hp} - g_j \circ \vec{\mathbf{k}}_j^{\,hp} \right\|^2\right)$
7: $\tilde{\mathbf{o}}_i^h = \sum_j a_{ij}^h \mathbf{z}_{ij}$
8: $\mathbf{o}_i^h = \sum_j a_{ij}^h \mathbf{v}_j^h$
9: $\bar{\mathbf{o}}_i^{hp} = g_i^{-1} \circ \sum_j a_{ij}^h \left(g_j \circ \vec{\mathbf{v}}_j^{\,hp}\right)$
10: $\tilde{\mathbf{s}}_i = \mathrm{Linear}\left(\mathrm{concat}_{h,p}\left(\tilde{\mathbf{o}}_i^h,\ \mathbf{o}_i^h,\ \bar{\mathbf{o}}_i^{hp},\ \|\bar{\mathbf{o}}_i^{hp}\|\right)\right)$
11: **return** $\{\tilde{\mathbf{s}}_i\}$

The proof of invariance is straightforward. The global transformation cancels in the affinity computation (Alg. 2, line 6), because the ℓ_2-norm of a vector is invariant under rigid transformations:

$$\begin{aligned}\left\|(g_{\text{global}} \circ g_i) \circ \vec{\mathbf{q}}_i^{\,hp} - (g_{\text{global}} \circ g_j) \circ \vec{\mathbf{k}}_j^{\,hp}\right\|^2 &= \left\|g_{\text{global}} \circ \left(g_i \circ \vec{\mathbf{q}}_i^{hp} - g_j \circ \vec{\mathbf{k}}_j^{hp}\right)\right\|^2 \\ &= \left\|g_i \circ \vec{\mathbf{q}}_i^{hp} - g_j \circ \vec{\mathbf{k}}_j^{hp}\right\|^2.\end{aligned}$$

In the computation of the output points (Alg. 2, line 9), the global transformation cancels when mapping

back to the local frame:

$$
\begin{aligned}
&(g_{\text{global}} \circ g_i)^{-1} \circ \sum_j a_{ij}^h \left((g_{\text{global}} \circ g_j) \circ \vec{\mathbf{v}}_j^{\,hp} \right) \\
&= g_i^{-1} \circ g_{\text{global}}^{-1} \circ g_{\text{global}} \circ \sum_j a_{ij}^h \left(g_j \circ \vec{\mathbf{v}}_j^{hp} \right) \\
&= g_i^{-1} \circ \sum_j a_{ij}^h \left(g_j \circ \vec{\mathbf{v}}_j^{hp} \right).
\end{aligned}
$$

The invariance with respect to the global reference frame implies that applying a shared rigid motion to all residues, while keeping the embeddings fixed, leads to the same update in the local frames. Therefore, the updated structure transforms under the same shared rigid motion, demonstrating that the update rule is equivariant under rigid motions.

Velocity Denoiser Now we can, finally, construct the architecture for learning the model (35).

The input of Alg. 3 consists of noisy tokens $G \in \mathbb{R}^{T\times L\times(7K+14)}$ defined in (3.1), conditioning tokens $G_{\text{cond}} \in \mathbb{R}^{T\times L\times(7K+14)}$ corresponding to given information about molecular configuration during the dynamics, and a set of key-frame rigid transformations $\{g^1,\dots,g^K\} \in (\mathrm{SE}(3)^L)^K$, which provide geometric reference frames. In addition, the model receives amino acid identities $A \in \{1,\dots,20\}^L$, and a conditioning mask $\mathbf{m} \in \{0,1\}^{T\times L\times(7K+14)}$.

The network first embeds the time variable and the amino acid identities (Alg. 3, line 5), and then processes the representations associated with each key frame using a stack of IPA layers. These layers incorporate geometric information by leveraging the rigid-body transformations g^{t_k}. The resulting representations are aggregated across key frames and combined with conditioning features (Alg. 3, line 10). The fused representation is subsequently refined by a stack of diffusion transformer attention layers, which model temporal and inter-residue dependencies in the trajectory. Finally, a linear projection produces the predicted velocity field $v_\theta \in \mathbb{R}^{T\times L\times(7K+14)}$.

Algorithm 3 Velocity network

1: **Input:**
noisy tokens $\mathcal{G} \in \mathbb{R}^{T\times L\times(7K+14)}$,
conditioning tokens $\mathcal{G}_{\text{cond}} \in \mathbb{R}^{T\times L\times(7K+14)}$,
key frame roto-translations $g^{t_1},\dots,g^{t_K} \in (SE(3)^L)^K$,
flow matching time t, amino acid identities $\mathcal{A} \in \{1,\dots,20\}^L$,
conditioning mask $\mathbf{m} \in \{0,1\}^{T\times L\times(7K+14)}$
2: **Output:** flow velocity $v_\theta \in \mathbb{R}^{T\times L\times(7K+14)}$
3: $t \leftarrow \text{Embed}(t)$
4: **for** $k \leftarrow 1$ to K **do**
5: $\quad \mathbf{x}_k \leftarrow \text{Embed}(\mathbf{A}) + \sum_{k'} \text{Linear}([g^t]^{-1} g^{t_{k'}})$
6: $\quad$ **for** $\ell \leftarrow 1$ to *num_ipa_layers* **do**
7: $\quad\quad \mathbf{x}_k \leftarrow \text{InvariantPointAttentionLayer}(\mathbf{x}_k, g^{t_k}, t)$ ▷ Alg. 4
8: $\quad$ **end for**
9: **end for**
10: $\mathbf{x} \leftarrow \sum_k \mathbf{x}_k + \text{Linear}(\mathbf{x}) + \text{Linear}(\mathcal{G}_{\text{cond}} \odot \mathbf{m}) + \text{Embed}(\mathbf{m})$
11: **for** $\ell \leftarrow 1$ to *num_transformer_layers* **do**
12: $\quad \mathbf{x} \leftarrow \text{DiffusionTransformerAttentionLayer}(\mathbf{x}, t)$ ▷ Alg. 5
13: **end for**
14: **return** $\text{DiffusionTransformerFinalLayer}(\mathbf{x}, t)$

Algorithm 4 InvariantPointAttentionLayer

Input: $\mathbf{x} \in \mathbb{R}^{L \times C}$, time conditioning t, roto-translations $g \in SE(3)^L$
$(\alpha, \beta, \gamma)_{\ell,f} = \mathrm{Linear}(t)$
$\mathbf{x} \mathrel{+}= \mathrm{InvariantPointAttention}(\mathrm{LayerNorm}(\mathbf{x}), g)$
$\mathbf{x} \mathrel{+}= g_\ell \odot \mathrm{AttentionWithRoPE}(\gamma_\ell \odot \mathrm{LayerNorm}(\mathbf{x}) + \beta_\ell)$
$\mathbf{x} \mathrel{+}= g_m \odot \mathrm{MLP}(\gamma_m \odot \mathrm{LayerNorm}(\mathbf{x}) + \beta_m)$
return $\mathbf{x}$

Algorithm 5 DiffusionTransformerAttentionLayer

Input: $\mathbf{x} \in \mathbb{R}^{T \times L \times C}$, time conditioning t
$(\alpha, \beta, \gamma)_{t,\ell,f} = \mathrm{Linear}(t)$
$\mathbf{x} \mathrel{+}= g_\ell \odot \mathrm{AttentionWithRoPE}(\gamma_\ell \odot \mathrm{LayerNorm}(\mathbf{x}) + \beta_\ell, \mathsf{dim} = 1)$
$\mathbf{x} \mathrel{+}= g_t \odot \mathrm{AttentionWithRoPE}(\gamma_t \odot \mathrm{LayerNorm}(\mathbf{x}) + \beta_t, \mathsf{dim} = 0)$
$\mathbf{x} \mathrel{+}= g_m \odot \mathrm{MLP}(\gamma_m \odot \mathrm{LayerNorm}(\mathbf{x}) + \beta_m)$
return $\mathbf{x}$

Conclusion

In this seminar work, we examined the application of modern diffusion-based generative modeling techniques to molecular dynamics. Starting from the classical formulation of MD in Sec. 1 as a high-dimensional nonlinear dynamical system governed by Newtonian mechanics, we discussed the computational challenges associated with simulating long-timescale molecular processes.

We then considered the framework of stochastic interpolants (Def. 2.1), which provides a principled mathematical foundation for constructing continuous transformations between probability measures. Within this framework, we analyzed the associated transport equation (Thm. 2.4), the corresponding Fokker–Planck formulations (Cor. 2.7), and the objectives whose minimizers characterize the velocity field (Thm. 2.9) and the score function (Thm. 2.5). In particular, we have seen (Cor. 2.8) how the generative modeling problem can be interpreted as learning a time-dependent flow in the space of probability distributions.

Building on this theoretical foundation, we described the MDGen architecture (Algs. 3–5), which formulates molecular trajectory generation as a conditional generative modeling task. A key aspect of this approach is the use of $\mathrm{SE}(3)$-invariant representations of protein structures via backbone frames and torsion angles (Sec. 3.1), allowing the model to respect the geometric symmetries of molecular systems. Furthermore, the integration of attention mechanisms, including rotary positional embeddings (39) and invariant point attention (Alg. 2), enables the network to capture both long-range dependencies and three-dimensional geometric structure.

Overall, diffusion-based surrogate modeling of molecular dynamics offers a computationally efficient alternative to direct numerical integration of physical equations of motion, and provide a flexible framework for learning complex trajectory distributions.

References

Abramson, J., Adler, J., Dunger, J., Evans, R., Green, T., Pritzel, A., Ronneberger, O., Willmore, L., Ballard, A. J., Bambrick, J., Bodenstein, S. W., Evans, D. A., Hung, C.-C., O'Neill, M., Reiman, D., Tunyasuvunakool, K., Wu, Z., Žemgulytė, A., Arvaniti, E., … Jumper, J. M. (2024). Accurate structure prediction of biomolecular interactions with alphafold 3. *Nature*, *630*(8016), 493–500. https://doi.org/10.1038/s41586-024-07487-w

Albergo, M. S., 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. http://jmlr.org/papers/v26/23-1605.html

Albergo, M. S., & Vanden-Eijnden, E. (2022). Building normalizing flows with stochastic interpolants. *ArXiv*, *abs/2209.15571*. https://api.semanticscholar.org/CorpusID:252668615

Alder, B. J., & Wainwright, T. E. (1959). Studies in molecular dynamics. i. general method. *Journal of Chemical Physics*, *31*, 459–466. https://api.semanticscholar.org/CorpusID:44487491

Ho, J., Salimans, T., Gritsenko, A., Chan, W., Norouzi, M., & Fleet, D. J. (2022). Video diffusion models. https://arxiv.org/abs/2204.03458

Jing, B., Berger, B., & Jaakkola, T. (2024). Alphafold meets flow matching for generating protein ensembles. https://arxiv.org/abs/2402.04845

Jing, B., Stärk, H., Jaakkola, T., & Berger, B. (2024). Generative modeling of molecular dynamics trajectories. https://arxiv.org/abs/2409.17808

Jumper, J., Evans, R., Pritzel, A., Green, T., Figurnov, M., Ronneberger, O., Tunyasuvunakool, K., Bates, R., Žídek, A., Potapenko, A., Bridgland, A., Meyer, C., Kohl, S. A. A., Ballard, A. J., Cowie, A., Romera-Paredes, B., Nikolov, S., Jain, R., Adler, J., … Hassabis, D. (2021). Highly accurate protein structure prediction with alphafold. *Nature*, *596*(7873), 583–589. https://doi.org/10.1038/s41586-021-03819-2

Kendrew, J. C., Bodo, G., Dintzis, H. M., Parrish, R. G., Wyckoff, H., & Phillips, D. C. (1958). A three-dimensional model of the myoglobin molecule obtained by x-ray analysis. *Nature*, *181*(4610), 662–666. https://doi.org/10.1038/181662a0

Ma, N., Goldstein, M., Albergo, M. S., Boffi, N. M., Vanden-Eijnden, E., & Xie, S. (2024). Sit: Exploring flow and diffusion-based generative models with scalable interpolant transformers. https://arxiv.org/abs/2401.08740

Perutz, M. F., Rossmann, M. G., Cullis, A. F., Muirhead, H., Will, G., & North, A. C. T. (1960). Structure of hæmoglobin: A three-dimensional fourier synthesis at 5.5-Å. resolution, obtained by x-ray analysis. *Nature*, *185*(4711), 416–422. https://doi.org/10.1038/185416a0

Su, J., Lu, Y., Pan, S., Murtadha, A., Wen, B., & Liu, Y. (2023). Roformer: Enhanced transformer with rotary position embedding. https://arxiv.org/abs/2104.09864

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2023). Attention is all you need. https://arxiv.org/abs/1706.03762

Yim, J., Trippe, B. L., Bortoli, V. D., Mathieu, E., Doucet, A., Barzilay, R., & Jaakkola, T. (2023). Se(3) diffusion model with application to protein backbone generation. https://arxiv.org/abs/2302.02277