A Score-based Geometric Model for Molecular Dynamics Simulations

Fang Wu∗
Columbia University
New York, USA
fw2359@columbia.edu

Qiang Zhang∗
Zhejiang University
Hangzhou, China
qiang.zhang.cs@zju.edu.cn

Xurui Jin
Mindrank
Hangzhou, China
xurui@mindrank.ai

Yinghui Jiang
Mindrank
Hangzhou, China
yinghui@mindrank.ai

Stan Z. Li†
Westlake University
Hangzhou, China
stan.zq.li@westlake.edu.cn

Abstract
Molecular dynamics (MD) has long been the de facto choice for modeling complex atomistic systems from first principles, and recently deep learning become a popular way to accelerate it. Notwithstanding, preceding approaches depend on intermediate variables such as the potential energy or force fields to update atomic positions, which requires additional computations to perform back-propagation. To waive this requirement, we propose a novel model called ScoreMD by directly estimating the gradient of the log density of molecular conformations. Moreover, we analyze that diffusion processes highly accord with the principle of enhanced sampling in MD simulations, and is therefore a perfect match to our sequential conformation generation task. That is, ScoreMD perturbs the molecular structure with a conditional noise depending on atomic accelerations and employs conformations at previous timeframes as the prior distribution for sampling. Another challenge of modeling such a conformation generation process is that the molecule is kinetic instead of static, which no prior studies strictly consider. To solve this challenge, we introduce a equivariant geometric Transformer as a score function in the diffusion process to calculate the corresponding gradient. It incorporates the directions and velocities of atomic motions via 3D spherical Fourier-Bessel representations. With multiple architectural improvements, we outperforms state-of-the-art baselines on MD17 and isomers of C7O2H10. This research provides new insights into the acceleration of new material and drug discovery.

1 Introductions

Molecular dynamics (MD) [18] [33] [86], an in silico tool that simulates complex atomic systems based on first principles, has exerted dramatic impacts in scientific research. Instead of yielding an average structure by experimental approaches including X-ray crystallography and cryo-EM, MD simulations can capture the sequential behavior of molecules in full atomic details at very fine temporal resolution, and thus allows researchers to quantify how much various regions of the molecule move at equilibrium and what types of structural fluctuations they undergo. In the areas of molecular biology and drug discovery [26], the most basic and intuitive application of MD is to assess the mobility or flexibility

∗Equal contributions.
†The corresponding author.

Preprint. Under review.
of various regions of a molecule. MD substantially accelerates the studies to observe biomolecular process in action, particularly important functional processes such as ligand binding [77], ligand- or voltage-induced conformational change [17], protein folding [44], or membrane transport [42, 84].

Nevertheless, the computational cost of MD generally scales cubically with respect to the number of electronic degrees of freedom [31]. Besides, important biomolecular processes like the conformational change often take place on timescales longer than those accessible by classical all-atom MD simulations [26]. Although a wide variety of enhanced sampling techniques have been proposed to capture longer-timescale events [49, 75], none of them is a panacea for timescale limitations [6] and might additionally cause decreased accuracy [61]. Thus, it is an urgent demand to fundamentally boost the efficiency of MD while keeping accuracy.

Recently, deep learning-based MD (DLMD) models provide a new paradigm to meet the pressing demand [5, 9, 30, 76]. The accuracy of those models stems from not only the distinctive ability of neural networks to approximate high-dimensional functions [4, 47], but the proper treatment of physical requirements like symmetry constraints and the concurrent learning scheme that generates a compact training dataset [31]. Despite their success, current DLMD models primarily suffer from the following three issues. First, most DLMD models still rely on intermediate variables (e.g., the potential energy) and multiple stages to generate subsequent biomolecular conformations. This substantially raises the computational expenditure and hinder the inference efficiency, since the inverse Hessian scales as cubically with the number of atom coordinates [14]. Second, existing DLMD models regard the DL module as a black-box to predict atomic attributes and never inosculate the neural architecture with the theory of thermodynamics. Last but not least, the majority of prevailing geometric methods [21, 74, 36, 45] are designed for immobile molecules and not suitable for dynamic systems where the directions and velocities of atomic motions count.

This paper proposes ScoreMD that aims to address the above-mentioned issues. First, ScoreMD is an one-stage procedure and forecasts the simulation trajectories without any dependency on the potential energy or forces. For the second issue, inspired by the consistency of diffusion processes in nonequilibrium thermodynamics [15] and probabilistic generative models [80, 81], ScoreMD adopts a score-based denoising diffusion generative model [83] with the exploration of various stochastic differential equations (SDEs). It sequentially corrupts training data by slowly increasing noise and then learns to reverse this corruption. This generative process highly accords with the enhanced sampling mechanism in MD [52], where a boost potential is added conditionally to smooth biomolecular potential energy surface and decrease energy barriers. Besides, to make geometric models aware of atom mobility, we introduce an equivariant geometric Transformer (EGT) as the score function for our ScoreMD. It refines the self-attention mechanism [89] with 3D spherical Fourier-Bessel representations to incorporate both the intersection and dihedral angles between each pair of atoms and their associated velocities.

We conduct comprehensive experiments on multiple standard MD simulation datasets including MD17 and C7O2H10 isomers. Numerical results demonstrate ScoreMD constantly outperforms state-of-the-art DLMD models by a large margin. The significantly superior performance illustrates the high capability of our ScoreMD to accurately produce MD trajectories for microscopic systems.

2 Background

2.1 Preliminaries

We consider a MD trajectory of a molecule with \(T\) timeframes. \(\mathcal{M}(t) = (\mathbf{x}^{(t)}, \mathbf{h}^{(t)}, \mathbf{v}^{(t)})\) denotes the conformation at time \(t \in [T]\) and is assumed to have \(N\) atoms. There \(\mathbf{x}^{(t)} \in \mathbb{R}^{N \times 3}\) and \(\mathbf{h}^{(t)} \in \mathbb{R}^{N \times \psi_h}\) denote the 3D coordinates and \(\psi_h\)-dimension roto-translational invariant features (e.g., atom types) associated with each atom, respectively. \(\mathbf{v}^{(t)} \in \mathbb{R}^{N \times 3}\) corresponds to the atomic velocities. We denote a vector norm by \(x = ||x||_2\), its direction by \(\hat{x} = x/\|x\|\), and the relative position by \(\mathbf{x}_{ij} = \mathbf{x}_i - \mathbf{x}_j\).

2.2 Molecular Dynamics

MD with classical potentials. The fundamental idea behind MD simulations is to study the time-dependent behavior of a microscopic system [16]. It generates the atomic trajectories for a specific
interatomic potential with certain initial conditions and boundary conditions [40]. This is obtained by solving the first-order differential equation of the Newton’s second law:

\[ F^{(t)}_i = m_i a^{(t)}_i = -\frac{\partial U(x^{(t)})}{\partial x^{(t)}_i}, \]

where \( F^{(t)}_i \) is the net force acting on the \( i \)-th atom of the system at a given point in the \( t \)-th timeframe, \( a^{(t)}_i \) is the corresponding acceleration, and \( m_i \) is the mass. \( U(x) \) is the potential energy function. The classic force field (FF) defines the potential energy function in Eq. [16] in Appendix [A]. Then numerical methods are utilized to advance the trajectory over small time increments \( \Delta t \) with the assistance of some integrator (see more introductions about MD in Appendix [A]).

Enhanced sampling in MD. Enhanced sampling methods [11] have been developed to accelerate MD and retrieve useful thermodynamic and kinetic data [69]. These methods exploit the fact that the free energy is a state function; thus, differences in free energy is independent upon the path between states [16]. Several techniques such as free-energy perturbation, umbrella sampling, tempering, and metadynamics are invented to reduce the energy barrier and smooth the potential energy surface [46, 43].

### 2.3 Score-based Generative Model

Score-based generative models Song et al. [83] refer to the score matching with Langevin dynamics [81] and the denoising diffusion probabilistic modeling [80]. They have shown effectiveness in the generation of images [25], graphs [58], shape [10], and molecular conformations [78].

**Diffusion process.** Assume a diffusion process \( \{x(s)\}_{s=0}^{S} \) indexed by a continuous time variable \( s \in [0, S] \), such that \( x(0) \sim p_0 \), for which we have a dataset of i.i.d. samples, and \( x(S) \sim p_S \), for which we have a tractable form to generate samples efficiently. Let \( p_s(x) \) be the probability density of \( x(s) \), and \( p(x(s_1) \mid x(s_0)) \) be the transition kernel from \( x(s_0) \) to \( x(s_1) \), where \( 0 \leq s_0 < s_1 \leq T \). Then the diffusion process is modeled as the solution to an Ito SDE [83]:

\[ dx = f(x,s)ds + g(s)d\mathbf{w}, \]

where \( \mathbf{w} \) is a standard Wiener process, \( f(\cdot, s) : \mathbb{R}^d \rightarrow \mathbb{R}^d \) is a vector-valued function called the drift coefficient of \( x(s) \), and \( g(\cdot) : \mathbb{R} \rightarrow \mathbb{R} \) is a scalar function known as the diffusion coefficient of \( x(s) \).

**Reverse process.** By starting from samples of \( x(S) \sim p_S \) and reversing the diffusion process, we can obtain samples \( x(0) \sim p_0 \). The reverse-time SDE can be acquired based on the result from Anderson [3] that the reverse of a diffusion process is also a diffusion process as:

\[ dx = [f(x, s) - g(s)^2 \nabla_x \log p_s(x)] ds + g(s)d\mathbf{w}, \]

where \( \mathbf{w} \) is a standard Wiener process when time flows backwards from \( S \) to \( 0 \), and \( ds \) is an infinitesimal negative timeframe. The score of a distribution can be estimated by training a score-based model on samples with score matching [29, 81]. To estimate \( \nabla_x \log p_s(x) \), one can train a time-dependent score-based model \( s_\theta(x, s) \) via a continuous generalization to the denoising score matching objective [83]:

\[ \theta^* = \arg \min_\theta \mathbb{E}_s \left\{ \lambda(s)\mathbb{E}_{x(0) \sim \mathbb{P}_{x(0)}(x(0))} \left[ \left\| s_\theta(x(s), s) - \nabla_x \log p_s(x(s) \mid x(0)) \right\|^2 \right] \right\}, \]

Here \( \lambda : [0, S] \rightarrow \mathbb{R}^+ \) is a positive weighting function, \( s \) is uniformly sampled over \([0, T] \), \( x(0) \sim p_0(x) \), and \( x(s) \sim p_{0 \theta}(x(s) \mid x(0)) \). With sufficient data and model capacity, score matching ensures that the optimal solution to Eq. [4] denoted by \( s_\theta^*(x, s) \), equals \( \nabla_x \log p_s(x) \) for almost all \( x \) and \( s \). We can typically choose \( \lambda \propto 1/\mathbb{E} \left[ \left\| \nabla_x \log p_{0\theta}(x(s) \mid x(0)) \right\|^2 \right] \) [83].

### 3 Proposed Method

#### 3.1 Overview

Most prior DLMD studies such as [98] rely on the potential energy \( U \) as the intermediate variable to acquire atomic forces and update positions, which demands an additional backpropagation calculation
and significantly increases the computational costs \cite{91}. Some recent work starts to abandon the two-stage manner and choose the atom-level force $F$ as the prediction target of deep networks \cite{65}. However, they all rely on the integrator from external computational tools to renew the positions in accordance with pre-calculated energy or forces. None embraces a straightforward paradigm to immediately forecast the 3D coordinates in a microscopic system concurrently based on previous available timeframes, i.e., $p(x(t+1) | \{M(i)\}_{i=0}^t)$. To bridge this gap, we seek to generate trajectories without any transitional integrator.

Several MD simulation frameworks assume the Markov property on biomolecular conformational dynamics \cite{13,48} for ease of representation, i.e., $p(x(t+1) | \{M(i)\}_{i=0}^t) = p(x(t+1) | M(t))$. We also hold this assumption and aim to estimate the gradient field of the log density of atomic positions at each timeframe, i.e., $\nabla_x \log p(x(t+1))$. In this setting, we design a score network based on the Transformer architecture to learn the scores of the position distribution, i.e., $s_{\theta}(M_{t+1}) = \nabla_x \log p(x(t+1))$. During the inference period, we regard the conformation of the previous frame $M_t$ as the prior distribution, from which $x_{t+1}$ is sampled. Note that $s_{\theta}(M_{t+1}) \in \mathbb{R}^N$, we formulate it as a node regression problem. The whole procedure of ScoreMD is depicted in Fig. 1.

### 3.2 Score-based Generative Models for Positions

The motivation for our extending the denoising diffusion models \cite{90} to MD simulations is their resemblance to the enhanced sampling mechanism. Inspired by non-equilibrium statistical physics, these models first systematically and slowly destroy structures in a distribution through an iterative forward diffusion process and then reverse it, similar to the behavior of perturbing the free energy in the system and striving to minimizing the overall energy \cite{32}.

**Perturbing data conditionally with SDEs.** Our goal is to construct a diffusion process \[ \{x(t+1)(s)\}_{s=0}^S \] indexed by a continuous time variable $s \in [0, S]$, such that $x(t+1)(0) \sim p_0$ and $x(t+1)(S) \sim p_S$. There, $p_0$ and $p_S$ are the data distribution and the prior distribution of atomic positions respectively, as Equation \ref{equation:2}.

How to incorporate noise remains critical to the success of the generation Godwin et al. \cite{22}, which ensures the resulting distribution does not collapse to a low dimensional manifold \cite{81}. Conventionally, $p_S$ is an unstructured prior distribution, such as a Gaussian distribution with fixed mean and variance \cite{83}, which is uninformative for $p_0$. This construction of $p_S$ improves the sample variety for image generation \cite{2} but may not work well for MD. One reason is corrupting molecular conformations unconditionally would trigger severe turbulence to the microscopic system; besides, it ignores the fact that molecular conformations of neighboring frames $M(t)$ and $M(t+1)$ are close to

---

**Figure 1:** The overall procedure of our ScoreMD. Starting from the conformation of the last time step, atomic locations are sequentially updated with the gradient information from the score network.
each other and their divergence is dependent on the status of the former one. Therefore, it is necessary to formulate $p_S$ with the prior knowledge of $\mathcal{M}(t)$.

To be explicit, the noise does not constantly grow along with $s$, but depends on prior states. This strategy aligns with the Gaussian accelerated MD (GaMD) mechanism (details in Appendix A) and serve a more practical way to inject turbulence into $p_0$. Driven by foregoing analysis, we introduce a conditional noise in compliance with the accelerations at prior frames and choose the following SDE:

$$dx^{(t+1)} = \sigma^{s}_{\alpha(t)} dw, \; s \in [0, 1], \quad (5)$$

where the noise term $\sigma^{s}_{\alpha(t)}$ is dynamically adjusted as:

$$\sigma^{s}_{\alpha(t)} = \sigma_a \eta_{\alpha} \left( \left\| \alpha^{(t-1)} \right\|_{2}^2 - \bar{a} \right)^{2}, \quad \left\| \alpha^{(t-1)} \right\|_{2}^2 < \bar{a}, \quad (6)$$

Here $\eta_{\alpha}$ is the harmonic acceleration constant and $\bar{a}$ represents the acceleration threshold. Once the system has a slow variation trend of motion (i.e., the systematical energy is low), it will be supplied with a large level of noise, and vice versa. Thus, the conditional noise is inversely proportional to $\alpha^{(t-1)}$ and inherits the merits of enhanced sampling.

**Generating samples through reversing the SDE.** Following the reverse-time SDE [3], samples of the next timeframe $x^{(t+1)}$ can be attained by reversing the diffusion process as:

$$dx^{(t+1)} = \left[ f \left( x^{(t+1)}, s \right) - g(s)^2 \nabla \alpha^{(t+1)} \log p_s \left( x^{(t+1)} \right) \right] ds + g(s) dw. \quad (7)$$

Once the score of each marginal distribution, $\nabla \alpha^{(t+1)} \log p_s \left( x^{(t+1)} \right)$, is known for all $s$, we can simulate the reverse diffusion process to sample from $p_0$. The workflow is summarized in Fig. 2.

**Estimating scores for the SDE.** Intuitively, the optimal parameters $\theta^*$ of the conditional score network $s_{\theta} \left( \mathcal{M}^{(t+1)} \right)$ can be trained directly by minimizing the following formula:

$$\mathbb{E}_s \left\{ \lambda(s) \mathbb{E}_{x^{(t+1)}(0)} \mathbb{E}_{x^{(t+1)}(s)} \left[ \left\| s_{\theta} \left( \mathcal{M}^{(t+1)}(s), s \right) - \nabla \alpha^{(t+1)} \log p_{0s} \left( x^{(t+1)}(s) \mid x^{(t+1)}(0) \right) \right\|_{2}^2 \right] \right\}. \quad (8)$$

Here $x^{(t+1)}(0) \sim p_0 \left( x^{(t+1)} \right)$ and $x^{(t+1)}(s) \sim p_{0s} \left( x^{(t+1)}(s) \mid x^{(t+1)}(0) \right)$. $\mathcal{M}^{(t+1)}$ stands for the disturbed conformation with the noised geometric position $x^{(t+1)}$. In order to efficiently solve Eq. 8 it is required to know the transition kernel $p_{0s} \left( x^{(t+1)}(s) \mid x^{(t+1)}(0) \right)$. When $f(\cdot, s)$ is affine, this transition kernel is always a Gaussian distribution, where its mean and variance are in closed-forms by means of standard techniques [70], as shown in Appendix B.1

### 3.3 Equivariant Geometric Score Network

Equivariance is an ubiquitous symmetry, which complies with the fact that physical laws hold regardless of the coordinate system. It has shown efficacy to integrate such inductive bias into model parameterization for modeling 3D geometry [38, 35]. Hence, we consider building the score network $s_{\theta}$ equivariant to rotation and translation transformations.
We first calculate the spherical Fourier-Bessel bases to integrate all available geometric information: \[ e_{SBF,1,0m}^{l} (x_{ab}^{l}, \varphi_{a,ab}^{l}, \theta_{a,ab}^{l}) = \sqrt{\frac{2}{c_{int}^{2}j_{o+1}(z_{on})}} j_{o} \left( \frac{z_{on}}{c_{int}^{2}j_{o+1}(z_{on})} \right) Y_{om} (\varphi_{a,ab}^{l}, \theta_{a,ab}^{l}), \]
\[ e_{SBF,2,0m}^{l} (x_{ab}^{l}, \varphi_{a,ab}^{l}, \theta_{a,ab}^{l}) = \sqrt{\frac{2}{c_{int}^{2}j_{o+1}(z_{on})}} j_{o} \left( \frac{z_{on}}{c_{int}^{2}j_{o+1}(z_{on})} \right) Y_{om} (\varphi_{a,ab}^{l}, \theta_{a,ab}^{l}), \]
where \( o \in [N_{	ext{CBF}}], n \in [N_{	ext{RBF}}], \) and \( m \in [N_{	ext{SBF}}] \) control the degree, root, and order of the radial basis functions, respectively. \( c_{int} \) is the interaction cutoff. \( j_{o} \) is the spherical Bessel functions. \( z_{on} \) is the \( n \)-th root of the \( o \)-degree Bessel functions. \( Y_{om} \) is the real spherical harmonics with degree \( o \) and order \( m \). Remarkably, 3D spherical Fourier-Bessel representations including \( e_{SBF,1} \) and \( e_{SBF,2} \) enjoy the roto-translation invariant property due to their exploitation of the relative distance as well as the invariant angles. Then those directional vectors are fed into EGL as:
\[ q_{i} = [f_{q} (h_{i}^{(l)}) \circ e_{SBF,1}^{l}] W_{SBF,1}, \]
\[ k_{i} = [f_{k} (h_{i}^{(l)}) \circ e_{SBF,2}^{l}] W_{SBF,2}, \]
\[ m_{i} = f_{m} (h_{i}^{(l)}), \]
\[ a_{ij} = \frac{q_{i} k_{j}^{T}}{\sqrt{\psi_{at}}}, \]
\[ v_{i}^{(t+1)} = f_{v} (h_{i}^{(l)}) v_{i}^{(t)} + \sum_{j=1}^{N} \phi (a_{ij}) x_{ij}^{(t)}, \]
\[ x_{i}^{(t+1)} = x_{i}^{(t)} + \frac{1}{L} v_{i}^{(t+1)}, h_{i}^{(t+1)} = f_{h} \left( \sum_{j=1}^{N} \phi (a_{ij}) m_{j} \right). \]
Here \( \circ \) denotes concatenation and \( L \) is the number of total layers in EGT. \( f_{q}, f_{k}, f_{m}, f_{v}, \) and \( f_{h} \) are linear transformations. \( f_{v} \) and \( f_{h} \) are velocity and feature operations, which are commonly approximately by multi-layer perceptrons (MLPs). \( q_{i} \), \( k_{i} \), and \( m_{i} \) are respectively the query, key, and value vectors with the same dimension \( \psi_{at} \). The weight matrix \( W_{SBF,1} \) and \( W_{SBF,2} \) are learnable, transferring dimensions of the concatenated vectors back to \( \psi_{at} \). \( a_{ij} \) is the attention that the token \( i \) pays to the token \( j \). \( \phi \) denotes the Softmax function. Finally, \( x_{i}^{(t)} \) at the last layer immediately draw the gradient field of locations, i.e., \( \nabla x_{i}^{(t)} \log p (x^{(t)}) \).
We construct the following two tasks and empirically evaluate the proposed method on them:

After training a time-dependent score-based model $s$ for ARMSE of all snapshots at a given version of $x$. Besides, we include a naive baseline that uses SCFNN, GMN, EGNN Flows [37].

4 Experiments

We construct the following two tasks and empirically evaluate the proposed method on them:

Short-term-to-long-term (S2L) Trajectory Generation. In this task setting, models are first trained on some short-term trajectories and are required to produce long-term trajectories of the same molecule given the starting conformation $x(t_0)$ as $p(x(t_{0}), ..., x(t_{1}), x(t_{2}))$. This extrapolation over time aims to examine the model’s capacity of generalization from the temporal view.

One-to-others (O2O) Trajectory Generation. In the O2O task, models are trained on the entire trajectories of some molecules and examined on other molecules from scratch. This evaluates model’s eligibility to generalize to conformations of different molecules, namely, the discrepancy with respect to different molecular types.

4.1 Experiment Setup

Evaluation metric. We follow Rao et al. [68] and adopt the accumulative root-mean-square-error (ARMSE) of all snapshots at a given $n$-step time period $\{t_i\}_{i=1}^n$ as the evaluation metric. ARMSE evaluates the generated conformations as: $\text{ARMSE} = \left( \frac{1}{n} \sum_{i=1}^{t_n} \left\| \ddot{x}^{(i)} - x^{(i)} \right\|^2 \right)^{\frac{1}{2}}$.

Baselines. We compare ScoreMD with several state-of-the-art methods for the MD trajectory prediction. Specifically, Tensor Field Network (TFN) [85] adopts filters built from spherical harmonics to achieve equivariance. Radial Field (RF) is a GNN drawn from Equivariant Flows [77]. SE(3)-Transformer [19] is an equivariant variant of the self-attention module for 3D point-clouds. EGNW [71] learns GNNs equivariant to rotations, translations, reflections and permutations. GMN [27] resorts to generalized coordinates to impose geometrical constraints on graphs. SCFNN [20] is a self-consistent field NN for learning long-range response of molecular systems. Besides, we include a naive baseline that uses $x(t_0)$ as the prediction for all the subsequent timeframes.
in order to show the intensity of fluctuation during the test period. See Appendix D.1 for full details of the experiments.

4.2 Short-term-to-long-term Trajectory Generation

Data. MD17 \cite{12} contains trajectories of eight thermalized molecules, and all are calculated at a temperature of 500K and a resolution of 0.5 femtosecond (ft). We use the first 20K frame pairs as the training set, and split the next 20K frame pairs equally into validation and test sets. Unfortunately, MD17 does not include velocities of particles, for which we use \( v(t) = x(t) - x(t-1) \) as a substitution, similarly to GMN.

Results. Table 1 documents the performance of baselines and our ScoreMD in S2L, where the best performance is marked bold and the second best is underlined for clear comparison. Note that floating overflow is encountered by RF (denoted as NA). We also provide the training losses in Appendix D.2. For all eight organic molecules, ScoreMD achieves the lowest ARMSEs, and is the only approach that outperforms the naive baseline. It is also found that though those baseline methods can realize a very low training loss, all seriously suffer from error propagation along the inference time, especially for RFN and RF. In contrast, ScoreMD controls the propagation of uncertainty to the largest extent and possesses satisfactory extrapolation over the temporal horizon. Moreover, different organic molecules perform in different manners during MD. Particularly, the benzene moves most actively than other molecules, which leads to the highest prediction errors. After a thorough investigation of its trajectory via the VMD program \cite{28}, we find that the inner structure of benzene is stable but MD17 ignores the periodic boundary condition (PBC). Benzene rotates and translates freely in the space, which vastly magnifies the task difficulty (see Appendix D.3).

Table 1: Extrapolation performance on MD17. Note the extrapolation errors for TFN are not available (NA) due to the floating number overflow.

| Methods   | Aspirin | Benzene | Ethanol | Malonaldehyde | Naphthalene | Salicylic | Toluene | Uracil |
|-----------|---------|---------|---------|---------------|-------------|-----------|---------|--------|
| Naive     | 0.716   | 2.507   | 0.616   | 0.815         | 0.153       | 0.182     | 0.958   | 0.170  |
| TFN       | NA      | NA      | NA      | NA            | NA          | NA        | NA      | NA     |
| RF        | 3.707   | 19.724  | 5.963   | 18.532        | 13.791      | 2.071     | 4.052   | 2.382  |
| SE(3)-Tr. | 0.813   | 2.415   | 0.678   | 1.183         | 1.834       | 1.230     | 1.312   | 0.691  |
| EGNN      | 0.868   | 2.518   | 0.719   | 0.889         | 0.484       | 0.632     | 1.034   | 0.464  |
| GMN       | 0.814   | 2.528   | 0.751   | 0.880         | 0.832       | 0.895     | 1.018   | 0.494  |
| SCFFN     | 1.151   | 2.832   | 1.084   | 1.096         | 0.923       | 0.918     | 1.229   | 0.857  |
| ScoreMD   | 0.648   | 2.465   | 0.637   | 0.784         | 0.298       | 0.471     | 0.820   | 0.393  |

4.3 One-to-others Trajectory Generation

Data. C7O2H10 \cite{8} is a dataset that consists of the trajectories of 113 randomly selected C7O2H10 isomers, which are calculated at a temperature of 100K and resolution of 1 fs using density functional theory with the PBE exchange-correlation potential. We select the top-5 isomers that have the largest ARMSEs out of 113 samples, computed based on the naive baseline, as the validation targets and take the rest as the training set. Same as the MD17 case, we compute the distance vector between neighboring frames as the velocities.

Results. Table 4 reports ARMSE of baselines and our ScoreMD on the five isomers from C7O2H10. ScoreMD exceeds all baselines with a large margin and beats the naive baselines for all target molecules. Unlike MD17, several methods including EGNN and GMN surpass the naive baseline.

Figure 4: Performance on the five isomers in C7O2H10.

| Methods   | ISO_1004 | ISO_2134 | ISO_2126 | ISO_3001 | ISO_1007 |
|-----------|----------|----------|----------|----------|----------|
| Naive     | 1.123    | 1.013    | 0.961    | 0.941    | 0.887    |
| TFN       | 7.390    | 10.990   | 10.412   | 4.697    | 10.677   |
| RF        | 4.772    | 4.364    | 21.576   | 9.077    | 11.049   |
| SE(3)-Tr. | 5.253    | 6.186    | 4.334    | 5.304    | 7.514    |
| EGNN      | 1.142    | 0.578    | 0.928    | 1.017    | 1.035    |
| GMN       | 1.205    | 0.363    | 0.988    | 1.055    | 1.154    |
| SCFFN     | 1.781    | 1.693    | 1.785    | 2.842    | 2.264    |

ScoreMD | 1.127 | 0.278 | 0.929 | 0.917 | 0.878 |

\footnote{Both MD17 and C7O2H10 datasets are available at \url{http://quantum-machine.org/datasets/}}
for ISO_2134. We plot the error propagation curves in Appendix D.4 and snapshots at different timeframes in Appendix D.5.

Closer inspection on the generated trajectories shows that, several baselines have worse generation quality because their conformations are not geometrically and biologically constrained. On the contrary, generated conformations by models like EGNN and GMN are geometrically legal, but their variations are minute. Interestingly, we discover that conformations generated by SCFNN remains unchanged after a few timeframes, which indicates the network is stuck in a fixed point.

5 Related Work

Molecular Dynamics with Deep Learning Recently, various DL models have become easy-to-use tools for fascinating MD with *ab initio* accuracy. Behler-Parrinello network [5] is one of the first models to learn potential surfaces from MD data. After that, Deep-Potential net [24] and ANI network [79] are further developed by extending to more advanced functions involving two neighbors. Inspired by Word2Vec [56], DTNN [73] and SchNet [74] achieve highly competitive prediction performance across the chemical compound space and the configuration space in order to simulate MD [59]. However, they still follow the routine of multi-stage simulations and rely on forces or energy as the prediction target [65, 99, 88]. Huang et al. [27] proposes an end-to-end GMN to characterise constrained systems of interacting objects, where molecules are defined as a set of rigidly connected particles with sticks and hinges. In spite of that, their experiments fail to be realistic and the constraint strongly violates the nature of MD, since no distance between any pair of atoms are fixed.

Conformation Generation. Researchers are increasingly interested conformation generation. Some works start from 2D molecular graphs to gain their corresponding 3D structures via bi-level programming [96] and continuous flows [95]. Some others concentrate on the inverse design to create the conformations of drug-like molecules [41] or crystals with desired properties [60, 94]. Recently, Gao and Remsing [20] propose a SCFNN that perturbs positions of the Wannier function centers induced by external electric fields. Latterly, diffusion models become a favored choice in conformation generation [78]. Xu et al. [97] introduces a GeoDiff by progressively injecting and eliminating small noises. However, its perturbations evolve over discrete times. A better approach would be to express dynamics as a set of differential equations since time is actually continuous [23]. Furthermore, these studies leverage diffusion models in recovering conformations from molecular graphs instead of generating sequential conformations. We fill in the gap by applying them to yield MD trajectories.

6 Conclusion and Future Work

In this paper, we propose ScoreMD, a novel principle for sequentially generating molecular conformations in MD simulations. ScoreMD marries denoising diffusion models with an equivariant geometric Transformer, which enables self-attention to leverage directional information in addition to the interatomic distances. Extensive experiments over multiple tasks on different datasets demonstrate that ScoreMD is superior to existing state-of-the-art models. This research greatly accelerate the discovery of new drugs and materials.
References

[1] Abrams, C., Bussi, G., 2014. Enhanced sampling in molecular dynamics using metadynamics, replica-exchange, and temperature-acceleration. Entropy 16, 163–199.

[2] Allen, M.P., Tildesley, D.J., 2012. Computer simulation in chemical physics. volume 397. Springer Science & Business Media.

[3] Anderson, B.D., 1982. Reverse-time diffusion equation models. Stochastic Processes and their Applications 12, 313–326.

[4] Barron, A.R., 1993. Universal approximation bounds for superpositions of a sigmoidal function. IEEE Transactions on Information theory 39, 930–945.

[5] Behler, J., Parrinello, M., 2007. Generalized neural-network representation of high-dimensional potential-energy surfaces. Physical review letters 98, 146401.

[6] Bernardi, R.C., Melo, M.C., Schulten, K., 2015. Enhanced sampling techniques in molecular dynamics simulations of biological systems. Biochimica et Biophysica Acta (BBA)-General Subjects 1850, 872–877.

[7] Brock, A., Donahue, J., Simonyan, K., 2018. Large scale gan training for high fidelity natural image synthesis. arXiv preprint arXiv:1809.11096.

[8] Brockherde, F., Vogt, L., Li, L., Tuckerman, M.E., Burke, K., Müller, K.R., 2017. Bypassing the kohn-sham equations with machine learning. Nature communications 8, 1–10.

[9] Brooks, B.R., Brooks III, T.E., Darden, T., Gohlke, H., Luo, R., Merz Jr, K.M., Onufriev, A., Simmerling, C., Wang, B., Woods, R.J., 2005. The amber biomolecular simulation programs. Journal of computational chemistry 26, 1668–1688.

[10] Case, D.A., Cheatham III, T.E., Darden, T., Gohlke, H., Luo, R., Merz Jr, K.M., Onufriev, A., Simmerling, C., Wang, B., Woods, R.J., 2005. The amber biomolecular simulation programs. Journal of computational chemistry 26, 1668–1688.

[11] Chodera, J.D., Noé, F., 2014. Markov state models of biomolecular conformational dynamics. Current opinion in structural biology 25, 135–144.

[12] De Groot, S.R., Mazur, P., 2013. Non-equilibrium thermodynamics. Courier Corporation.

[13] De Vivo, M., Masetti, M., Bottegoni, G., Cavalli, A., 2016. Role of molecular dynamics and related methods in drug discovery. Journal of medicinal chemistry 59, 4035–4061.

[14] Dror, R.O., Arlow, D.H., Maragakis, P., Mildorf, T.J., Pan, A.C., Xu, H., Borhani, D.W., Shaw, D.E., 2011. Activation mechanism of the β2-adrenergic receptor. Proceedings of the National Academy of Sciences 108, 18684–18689.

[15] Frenkel, D., Smit, B., 2001. Understanding molecular simulation: from algorithms to applications. volume 1. Elsevier.

[16] Fuchs, F., Worrall, D., Fischer, V., Welling, M., 2020. Se (3)-transformers: 3d roto-translation equivariant attention networks. Advances in Neural Information Processing Systems 33, 1970–1981.
[20] Gao, A., Remsing, R.C., 2022. Self-consistent determination of long-range electrostatics in neural network potentials. Nature Communications 13, 1–11.

[21] Gilmer, J., Schoenholz, S.S., Riley, P.F., Vinyals, O., Dahl, G.E., 2017. Neural message passing for quantum chemistry, in: International conference on machine learning, PMLR. pp. 1263–1272.

[22] Godwin, J., Schaarschmidt, M., Gaunt, A.L., Sanchez-Gonzalez, A., Rubanova, Y., Veličković, P., Kirkpatrick, J., Battaglia, P., 2021. Simple gnn regularisation for 3d molecular property prediction and beyond, in: International Conference on Learning Representations.

[23] Greydanus, S., Dzamba, M., Yosinski, J., 2019. Hamiltonian neural networks. Advances in Neural Information Processing Systems 32.

[24] Han, J., Zhang, L., Car, R., et al., 2017. Deep potential: A general representation of a many-body potential energy surface. arXiv preprint arXiv:1707.01478.

[25] Ho, J., Jain, A., Abbeel, P., 2020. Denoising diffusion probabilistic models. Advances in Neural Information Processing Systems 33, 6840–6851.

[26] Hollingsworth, S.A., Dror, R.O., 2018. Molecular dynamics simulation for all. Neuron 99, 1129–1143.

[27] Huang, W., Han, J., Rong, Y., Xu, T., Sun, F., Huang, J., 2021. Constrained graph mechanics networks, in: International Conference on Learning Representations.

[28] Humphrey, W., Dalke, A., Schulten, K., 1996. Vmd: visual molecular dynamics. Journal of molecular graphics 14, 33–38.

[29] Hyvärinen, A., Dayan, P., 2005. Estimation of non-normalized statistical models by score matching. Journal of Machine Learning Research 6.

[30] Jelinek, B., Groh, S., Horstemeyer, M.F., Houze, J., Kim, S.G., Wagner, G.J., Moitra, A., Baskes, M.I., 2012. Modified embedded atom method potential for al, si, mg, cu, and fe alloys. Physical Review B 85, 245102.

[31] Jia, W., Wang, H., Chen, M., Lu, D., Lin, L., Car, R., Weinan, E., Zhang, L., 2020. Pushing the limit of molecular dynamics with ab initio accuracy to 100 million atoms with machine learning, in: SC20: International conference for high performance computing, networking, storage and analysis, IEEE. pp. 1–14.

[32] Jorgensen, W.L., Ravimohan, C., 1985. Monte carlo simulation of differences in free energies of hydration. The Journal of chemical physics 83, 3050–3054.

[33] Karplus, M., McCammon, J.A., 2002. Molecular dynamics simulations of biomolecules. Nature structural biology 9, 646–652.

[34] Kingma, D.P., Ba, J., 2014. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.

[35] Klicpera, J., Becker, F., Günnemann, S., 2021. Gemnet: Universal directional graph neural networks for molecules. Advances in Neural Information Processing Systems 34.

[36] Klicpera, J., Groß, J., Günnemann, S., 2020. Directional message passing for molecular graphs. arXiv preprint arXiv:2003.03123.

[37] Köhler, J., Klein, L., Noé, F., 2019. Equivariant flows: sampling configurations for multi-body systems with symmetric energies. arXiv preprint arXiv:1910.00753.

[38] Köhler, J., Klein, L., Noé, F., 2020. Equivariant flows: exact likelihood generative learning for symmetric densities, in: International Conference on Machine Learning, PMLR. pp. 5361–5370.

[39] Lambert, J.D., 1991. Numerical methods for ordinary differential systems: the initial value problem. John Wiley & Sons, Inc.
[40] Li, J., 2005. Basic molecular dynamics, in: Handbook of Materials Modeling. Springer, pp. 565–588.

[41] Li, Y., Pei, J., Lai, L., 2021. Learning to design drug-like molecules in three-dimensional space using deep generative models. arXiv preprint arXiv:2104.08474.

[42] Liang, R., Swanson, J.M., Madsen, J.J., Hong, M., DeGrado, W.F., Voth, G.A., 2016. Acid activation mechanism of the influenza a m2 proton channel. Proceedings of the National Academy of Sciences 113, E6955–E6964.

[43] Liao, Q., 2020. Enhanced sampling and free energy calculations for protein simulations, in: Progress in molecular biology and translational science. Elsevier. volume 170, pp. 177–213.

[44] Lindorff-Larsen, K., Piana, S., Dror, R.O., Shaw, D.E., 2011. How fast-folding proteins fold. Science 334, 517–520.

[45] Luo, R., Zhang, Q., Yang, Y., Wang, J., 2020. Replica-exchange nos\'e-hoover dynamics for bayesian learning on large datasets. Advances in Neural Information Processing Systems 33, 17874–17885.

[46] Ma, C., Wu, L., et al., 2020. Machine learning from a continuous viewpoint, i. Science China Mathematics 63, 2233–2266.

[47] Malmstrom, R.D., Lee, C.T., Van Wart, A.T., Amaro, R.E., 2014. Application of molecular-dynamics based markov state models to functional proteins. Journal of chemical theory and computation 10, 2648–2657.

[48] Marrink, S.J., Tieleman, D.P., 2013. Perspective on the martini model. Chemical Society Reviews 42, 6801–6822.

[49] Martys, N.S., Mountain, R.D., 1999. Velocity verlet algorithm for dissipative-particle-dynamics-based models of suspensions. Physical Review E 59, 3733.

[50] Miao, Y., Bhattarai, A., Wang, J., 2020. Ligand gaussian accelerated molecular dynamics (ligamd): Characterization of ligand binding thermodynamics and kinetics. Journal of chemical theory and computation 16, 5526–5547.

[51] Miao, Y., Feher, V.A., McCammon, J.A., 2015a. Gaussian accelerated molecular dynamics: unconstrained enhanced sampling and free energy calculation. Journal of chemical theory and computation 11, 3584–3595.

[52] Miao, Y., Feixas, F., Eun, C., McCammon, J.A., 2015b. Accelerated molecular dynamics simulations of protein folding. Journal of computational chemistry 36, 1536–1549.

[53] Miao, Y., McCammon, J.A., 2016. Graded activation and free energy landscapes of a muscarinic g-protein–coupled receptor. Proceedings of the National Academy of Sciences 113, 12162–12167.

[54] Mikolov, T., Chen, K., Corrado, G., Dean, J., 2013. Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781.

[55] Neal, R.M., et al., 2011. Mcmc using hamiltonian dynamics. Handbook of markov chain monte carlo 2, 2.

[56] Niu, C., Song, Y., Song, J., Zhao, S., Grover, A., Ermon, S., 2020. Permutation invariant graph generation via score-based generative modeling, in: International Conference on Artificial Intelligence and Statistics, PMLR. pp. 4474–4484.
[59] Noé, F., Tkatchenko, A., Müller, K.R., Clementi, C., 2020. Machine learning for molecular simulation. Annual review of physical chemistry 71, 361–390.

[60] Noh, J., Kim, J., Stein, H.S., Sanchez-Lengeling, B., Gregoire, J.M., Aspuru-Guzik, A., Jung, Y., 2019. Inverse design of solid-state materials via a continuous representation. Matter 1, 1370–1384.

[61] de Oliveira, C.A.F., Hamelberg, D., McCammon, J.A., 2006. On the application of accelerated molecular dynamics to liquid water simulations. The journal of physical chemistry B 110, 22695–22701.

[62] Pang, T., Xu, K., Li, C., Song, Y., Ermon, S., Zhu, J., 2020. Efficient learning of generative models via finite-difference score matching. Advances in Neural Information Processing Systems 33, 19175–19188.

[63] Pang, Y.T., Miao, Y., Wang, Y., McCammon, J.A., 2017. Gaussian accelerated molecular dynamics in namd. Journal of chemical theory and computation 13, 9–19.

[64] Parisi, G., 1981. Correlation functions and computer simulations. Nuclear Physics B 180, 378–384.

[65] Park, C.W., Kornbluth, M., Vandermause, J., Wolverton, C., Kozinsky, B., Mailoa, J.P., 2021. Accurate and scalable graph neural network force field and molecular dynamics with direct force architecture. npj Computational Materials 7, 1–9.

[66] Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al., 2019. Pytorch: An imperative style, high-performance deep learning library. Advances in neural information processing systems 32.

[67] Platen, E., Bruti-Liberati, N., 2010. Numerical solution of stochastic differential equations with jumps in finance. volume 64. Springer Science & Business Media.

[68] Rao, C., Sun, H., Liu, Y., 2021. Embedding physics to learn spatiotemporal dynamics from sparse data. arXiv preprint arXiv:2106.04781.

[69] Rocchia, W., Masetti, M., Cavalli, A., 2012. Enhanced sampling methods in drug design. Physico-Chemical and Computational Approaches to Drug Discovery. The Royal Society of Chemistry, 273–301.

[70] Särkkä, S., Solin, A., 2019. Applied stochastic differential equations. volume 10. Cambridge University Press.

[71] Satorras, V.G., Hoogeboom, E., Welling, M., 2021. E (n) equivariant graph neural networks, in: International Conference on Machine Learning, PMLR. pp. 9323–9332.

[72] Schlick, T., 2010. Molecular modeling and simulation: an interdisciplinary guide. volume 2. Springer.

[73] Schütt, K.T., Arbabzadah, F., Chmiela, S., Müller, K.R., Tkatchenko, A., 2017. Quantum-chemical insights from deep tensor neural networks. Nature communications 8, 1–8.

[74] Schütt, K.T., Sauceda, H.E., Kindermans, P.J., Tkatchenko, A., Müller, K.R., 2018. Schnet—a deep learning architecture for molecules and materials. The Journal of Chemical Physics 148, 241722.

[75] Schwantes, C.R., McGibbon, R.T., Pande, V.S., 2014. Perspective: Markov models for long-timescale biomolecular dynamics. The Journal of chemical physics 141, 09B201_1.

[76] Senftle, T.P., Hong, S., Islam, M.M., Kylasa, S.B., Zheng, Y., Shin, Y.K., Junkermeier, C., Engel-Herbert, R., Janik, M.J., Aktulga, H.M., et al., 2016. The reaxff reactive force-field: development, applications and future directions. npj Computational Materials 2, 1–14.

[77] Shan, Y., Kim, E.T., Eastwood, M.P., Dror, R.O., Seeliger, M.A., Shaw, D.E., 2011. How does a drug molecule find its target binding site? Journal of the American Chemical Society 133, 9181–9183.
[78] Shi, C., Luo, S., Xu, M., Tang, J., 2021. Learning gradient fields for molecular conformation generation, in: International Conference on Machine Learning, PMLR. pp. 9558–9568.

[79] Smith, J.S., Isayev, O., Roitberg, A.E., 2017. Ani-1: an extensible neural network potential with dft accuracy at force field computational cost. Chemical science 8, 3192–3203.

[80] Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N., Ganguli, S., 2015. Deep unsupervised learning using nonequilibrium thermodynamics, in: International Conference on Machine Learning, PMLR. pp. 2256–2265.

[81] Song, Y., Ermon, S., 2019. Generative modeling by estimating gradients of the data distribution. Advances in Neural Information Processing Systems 32.

[82] Song, Y., Garg, S., Shi, J., Ermon, S., 2020a. Sliced score matching: A scalable approach to density and score estimation, in: Uncertainty in Artificial Intelligence, PMLR. pp. 574–584.

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

[84] Suomivuori, C.M., Gamiz-Hernandez, A.P., Sundholm, D., Kaila, V.R., 2017. Energetics and dynamics of a light-driven sodium-pumping rhodopsin. Proceedings of the National Academy of Sciences 114, 7043–7048.

[85] Thomas, N., Smidt, T., Kearnes, S., Yang, L., Li, L., Kohlhoff, K., Riley, P., 2018. Tensor field networks: Rotation-and translation-equivariant neural networks for 3d point clouds. arXiv preprint arXiv:1802.08219.

[86] Tuckerman, M., 2010. Statistical mechanics: theory and molecular simulation. Oxford university press.

[87] Tuckerman, M.E., Martyna, G.J., 2000. Understanding modern molecular dynamics: Techniques and applications.

[88] Unke, O.T., Chmiela, S., Sauceda, H.E., Gastegger, M., Poltavsky, I., Schutt, K.T., Tkatchenko, A., Muller, K.R., 2021. Machine learning force fields. Chemical Reviews 121, 10142–10186.

[89] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I., 2017. Attention is all you need. Advances in neural information processing systems 30.

[90] Vincent, P., 2011. A connection between score matching and denoising autoencoders. Neural computation 23, 1661–1674.

[91] Wang, H., Zhang, L., Han, J., Weinan, E., 2018. Deepmd-kit: A deep learning package for many-body potential energy representation and molecular dynamics. Computer Physics Communications 228, 178–184.

[92] Wang, J., Arantes, P.R., Bhattacharjee, A., Hsu, R.V., Pawnikar, S., Huang, Y.m.M., Palermo, G., Miao, Y., 2021. Gaussian accelerated molecular dynamics: Principles and applications. Wiley Interdisciplinary Reviews: Computational Molecular Science 11, e1521.

[93] Wu, F., Zhang, Q., Radev, D., Cui, J., Zhang, W., Xing, H., Zhang, N., Chen, H., 2021. 3d-transformer: Molecular representation with transformer in 3d space. arXiv preprint arXiv:2110.01191.

[94] Xie, T., Fu, X., Ganea, O.E., Barzilay, R., Jaakkola, T., 2021. Crystal diffusion variational autoencoder for periodic material generation. arXiv preprint arXiv:2110.06197.

[95] Xu, M., Luo, S., Bengio, Y., Peng, J., Tang, J., 2021a. Learning neural generative dynamics for molecular conformation generation. arXiv preprint arXiv:2102.10240.

[96] Xu, M., Wang, W., Luo, S., Shi, C., Bengio, Y., Gomez-Bombarelli, R., Tang, J., 2021b. An end-to-end framework for molecular conformation generation via bilevel programming, in: International Conference on Machine Learning, PMLR. pp. 11537–11547.
[97] Xu, M., Yu, L., Song, Y., Shi, C., Ermon, S., Tang, J., 2022. Geodiff: a geometric diffusion model for molecular conformation generation. arXiv preprint arXiv:2203.02923.

[98] Zhang, L., Han, J., Wang, H., Car, R., Weinan, E., 2018. Deep potential molecular dynamics: a scalable model with the accuracy of quantum mechanics. Physical review letters 120, 143001.

[99] Zhang, Y., Hu, C., Jiang, B., 2021. Accelerating atomistic simulations with piecewise machine-learned ab initio potentials at a classical force field-like cost. Physical Chemistry Chemical Physics 23, 1815–1821.
Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper’s contributions and scope? [Yes] Yes
   (b) Did you describe the limitations of your work? [Yes] Yes
   (c) Did you discuss any potential negative societal impacts of your work? [N/A]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes] Yes

2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes] Yes
   (b) Did you include complete proofs of all theoretical results? [Yes] Yes

3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] Yes
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] Yes
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] Yes
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] Yes

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes] Yes
   (b) Did you mention the license of the assets? [Yes] Yes
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes] Yes
   (d) Did you discuss whether and how consent was obtained from people whose data you’re using/curating? [Yes] Yes
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [Yes] Yes

5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]
A More about Molecular Dynamics Simulations

Force Field for MD. The following equation defines the classic force field for MD simulations as:

\[
U = \sum_{i} \frac{c_{i}}{2} (l_{i} - l_{0,i})^{2} + \sum_{i} \frac{c_{\alpha}}{2} (\alpha_{i} - \alpha_{0,i})^{2} + \sum_{i,j,k} \left\{ \frac{M}{2} U_{ik} [1 + \cos (c_{ik} \cdot \theta_{ik} - \theta_{0,ik})] \right\} + \sum_{i,j} \varepsilon_{ij} \left[ \left( \frac{x_{0,ij}}{x_{ij}} \right)^{12} - 2 \left( \frac{x_{0,ij}}{x_{ij}} \right)^{6} \right] + \sum_{i,j,k} \frac{q_i q_j}{\Delta \varepsilon_0 \varepsilon_r x_{ij}}.
\]

where \(l\), \(\alpha\), and \(\theta\) correspond to the bond lengths, angles, and dihedral angles, respectively. \(l_0\) and \(\alpha_0\) are reference values. \(c_i\) and \(c_\alpha\) are force constants. \(c_{ik}\) is a parameter describing the multiplicity for the \(k\)-th term of the series. \(\theta_{0,ik}\) is the corresponding phase angle. \(U_{ik}\) is the energy barrier. \(x_{ij}\) is the distance between considered atoms. \(\varepsilon_{ij}\) defines the depth of the energy well. \(x_{0,ij}\) is the minimum distance that equals the sum of the van der Waals radii of the two interaction atoms. \(q_i\) and \(q_j\) are the partial changes of a pair of atoms. \(\varepsilon_0\) stands for the permittivity of free space, and \(\varepsilon_r\) is the relative permittivity, which takes a value of 1 in vacuum.

The first three terms in Eq. [16] represent intramolecular interactions of the atoms. They depict variations in potential energy as a function of bond stretching, bending, and torsions between atoms directly involved in bonding relationships. They are represented as the summations over bond lengths (\(l\)), angles (\(\theta\)), and dihedral angles (\(\alpha\)), respectively. The last two terms stand for van der Waals interactions, modeled using the Lennard-Jones potential, and electrostatic interactions, modeled using Coulomb’s law. These atomic forces are denoted as non-bonded items because they are caused by interactions between atoms that are not bonded.

Integrator. Then numerical methods are utilized to advance the trajectory over small time increments \(\Delta t\) with the assistance of some integrator. An integrator advances the trajectory over small time increments \(\Delta t\) as:

\[
\mathbf{x}(t_0) \rightarrow \mathbf{x}(t_0 + \Delta t) \rightarrow \mathbf{x}(t_0 + 2\Delta t) \rightarrow \cdots \rightarrow \mathbf{x}(t_0 + T\Delta t),
\]

where \(T\) is usually taken around \(10^4\) picoseconds (ps) to \(10^7\) ps. Various strategies have been invented to iteratively update atomic positions including the central difference (Verlet, leap-frog, velocity Verlet, Beeman algorithm) [18, 2], the predictor-corrector [39], and the symplectic integrators [87, 72].

MD trajectory iterations. With sampled forces, we can now advance to update the velocities of each atom and compute their coordinates of the next timeframe. As an example of an integrator, the velocity-Verlet [50] in Eq. [18] is a simple and widely used algorithm in MD software such as AMBER [11]. The coordinates and velocities are updated simultaneously as:

\[
\mathbf{x}_{i}^{(t+1)} = \mathbf{x}_{i}^{(t)} + \mathbf{v}_{i}^{(t)} \cdot \Delta t + \frac{1}{2} \left( \frac{\mathbf{x}_{i}^{(t)}}{m_i} \right) (\Delta t)^2,
\]

\[
\mathbf{v}_{i}^{(t+1)} = \mathbf{v}_{i}^{(t)} + \frac{1}{2} \left[ \frac{\mathbf{x}_{i}^{(t)}}{m_i} + \frac{\mathbf{x}_{i}^{(t+1)}}{m_i} \right] \Delta t.
\]

Gaussian accelerated MD GaMD [55, 63] is a robust computational method for simultaneous unconstrained enhanced sampling and free energy calculations of biomolecules, which greatly accelerates MD by orders of magnitude [22]. It works by adding a harmonic boost potential to reduce energy barriers. When the system potential \(U(x)\) is lower than a threshold energy \(\bar{U}\), a boost potential \(\Delta U\) is injected as:

\[
\Delta U = \frac{1}{2} \eta_U \left( \bar{U} - U(x) \right)^2, U(x) < \bar{U},
\]

where \(\eta_U\) is the harmonic force constant. The modified system potential is given by \(U^* = U + \Delta U\). Otherwise, when the system potential is above the threshold energy, the boost potential is set to zero.
Remarkably, the boost potential $\Delta U$ exhibits a Gaussian distribution, allowing for accurate reweighting of the simulations using cumulant expansion to the second order and recovery of the original free energy landscapes even for large biomolecules [54, 63]. Moreover, three enhanced sampling principles are imposed to the boost potential in GaMD. Among them, for the sake of ensuring accurate reweighting [53], the standard deviation of $\Delta U$ ought to be adequately small, i.e., narrow distribution:

$$\sigma_{\Delta U} = \sqrt{\left( \frac{\partial \Delta U}{\partial U} \bigg|_{U = U_{\text{avg}}} \right)^2 \sigma_U^2} = \eta (\bar{U} - U_{\text{avg}}) \sigma_U \leq \sigma_0,$$

where $U_{\text{avg}}$ and $\sigma_U$ are the average and standard deviation of the system potential energies, $\sigma_{\Delta U}$ is the standard deviation of $\Delta U$ with $\sigma_0$ as a user-specific upper limit (e.g. $10k_BT$) for accurate reweighting [51]. This also indicates that our choice of $\eta$ ought to be small enough with some upper bound.

**B More about ScoreMD**

**B.1 Reverse Process in ScoreMD**

**Score Matching Objective.** Notably, other score matching objectives, such as sliced score matching [82] and finite-difference score matching [62] can also be applied here rather than denoising score matching in Eq. 8.

**Transition kernel.** In our case, the transition kernel has a closed form as:

$$p_{0s} \left( x^{(t+1)}(s) | x^{(t+1)}(0) \right) = N \left( x^{(t+1)}(s); x^{(t+1)}(0), \frac{1}{2 \log \sigma_a(t)} (\sigma_{a(t)}^2 - 1) I \right).$$

**B.2 Sampling in ScoreMD**

**SDE solvers.** Numerical solvers provide approximate computation from SDEs. Many general-purpose numerical methods, such as Euler-Maruyama and stochastic Runge-Kutta methods [67], are applicable to the reverse-time SDE for sample generation.

**Predictor-corrector samplers.** In addition to SDE solvers, we can also employ score-based MCMC approaches such as Langevin MCMC [64] or HMC [57] to sample from directly, and correct the solution of a numerical SDE solver. We refer readers to Song et al. [83] for more details.

**Algorithm 1** Sampling Algorithm with Predictor-Corrector.

**Require:** $N_P$: Number of discretization steps for the reverse-time SDE.
**Require:** $N_C$: Number of corrector steps.

Initialize $x^{(t+1)}(N_P) \leftarrow x^{(t)}$

for $i = N_P - 1$ to 0 do

$x^{(t+1)}(i) \leftarrow \text{Predictor} \left( x^{(t+1)}(i + 1) \right)$

for $j = 1$ to $N_C - 1$ do

$x^{(t+1)}(i) \leftarrow \text{Corrector} \left( x^{(t+1)}(i) \right)$

end for

end for

return $x^{(t+1)}$

**C Equivariance Proof for EGT**

In this part, we provide a strict proof that EGT achieves $E(n)$ equivariance on $x^{(t)}$. More formally, for any orthogonal matrix $Q \in \mathbb{R}^{n \times n}$ and any translation matrix $o \in \mathbb{R}^{n \times 3}$, the model should satisfy:

$$Qx^{(t)} + o, Qu^{(t)} + o, h^{(t)} + o = EGL \left( Qx^{(t)} + o, Qu^{(t)} + o, h^{(t)} + o \right).$$

(23)
As assumed in the preliminary, $h^{(t),0}$ is invariant to E(n) transformation. In other words, we do not encode any information about the absolute position or orientation of $x^{(t),0}$ into $h^{(t),0}$. Moreover, the spherical representations $e_{SBF}^{a}$ and $e_{SBF}^{b}$ will be invariant too. This is because the distance between two particles and the three angles are invariant to translation as

$$
\|x^{(t),l}_a + o - (x^{(t),l}_b + o)\|^2 = \|x^{(t),l}_a - x^{(t),l}_b\|^2,
$$

and they are invariant to rotations as

$$
\|Qx^{(t),l}_a - Qx^{(t),l}_b\|^2 = (x^{(t),l}_a - x^{(t),l}_b)^T Q^T Q (x^{(t),l}_a - x^{(t),l}_b) = (x^{(t),l}_a - x^{(t),l}_b)^T I (x^{(t),l}_a - x^{(t),l}_b) = \|x^{(t),l}_a - x^{(t),l}_b\|^2.
$$

As for angles, the intersection angles take the form of $\arccos$ while $h$. Similarly, in this subsection, we introduce details of hyper-parameters used in our experiments.

D.1 Implementation Details

The invariance of the dihedral angle can be proven in the same way as the intersection angles. Finally, it leads to the result that all query, key, value vectors, the attention scores, and feature embeddings become invariant. Now we aim to show:

$$Qv^{(t),l+1}_i = f_v (h^{(t),l}_i) Qv^{(t),l}_i + \sum_{j=1}^{N} \phi (a_{ij}) (Qx^{(t),l}_{ij} + o - (Qx^{(t),l}_{ij} + o)).
$$

The right hand of this equation can be written as:

$$
Qf_v \left( h^{(t),l}_i \right) v^{(t),l}_i + \sum_{j=1}^{N} \phi (a_{ij}) \left( x^{(t),l}_{ij} + o - (x^{(t),l}_{ij} + o) \right)
$$

$$= Qf_v \left( h^{(t),l}_i \right) v^{(t),l}_i + Q \sum_{j=1}^{N} \phi (a_{ij}) \left( x^{(t),l}_{ij} - x^{(t),l}_{ij} \right)
$$

$$= Q \left( f_v \left( h^{(t),l}_i \right) v^{(t),l}_i + \sum_{j=1}^{N} \phi (a_{ij}) \left( x^{(t),l}_{ij} - x^{(t),l}_{ij} \right) \right)
$$

$$= Qv^{(t),l+1}_i.
$$

Obviously, it is easy to show that $Qx^{(t),l+1}_i + o = Qx^{(t),l}_i + o + Qv^{(t),l+1}_i$, and we omit this derivation.

D Experiment

D.1 Implementation Details

In this subsection, we introduce details of hyper-parameters used in our experiments.
**Training details.** All models are implemented in Pytorch [66], and are optimized with an Adam [34] optimizer with the initial learning rate of 5e-4 and weight decay of 1e-10 on a single A100 GPU. A ReduceLROnPlateau scheduler is utilized to adjust the learning rate according to the training ARMSE with a lower bound of 1e-7 and a patience of 5 epochs. The random seed is 1. The hidden dimension is 64. All baseline models are evaluated with 4 layers. For RF, TFN, EGNN, GMN, SCFNN, and ScoreMD, the batch size is 200. The batch size is reduced to 100 for SE(3)-Transformer due to the memory explosion error. We train models for 200 epochs and test their performance each 5 epochs. Similar to [78, 27], for graph-based baselines, we augment the original molecular graphs with 2-hop neighbors, and concatenate the hop index with atom number of the connected atoms as well as the edge type indicator as the edge feature. For GMN, we use the configurations in the paper that bonds without commonly-connected atoms are selected as sticks, and the rest of atoms as isolated particles. Notably, unlike Huang et al. [27], we keep all atoms including the hydrogen atoms for a more accurate description of the microscopic system.

**ScoreMD architecture.** The score function, EGT, has 6 EGLs, and each layer has 8 attention heads. We adopt ReLU as the activation function and a dropout rate of 0.1. The input embedding size is 128 and the hidden size for feed-forward network (FFN) is 2048. The numbers of the degree, root, and the order of the radial basis are all 2. The interaction cutoff is 1.6 Å. We use the 2-norm of velocity concatenated with the atom number as the node feature, which is invariant to geometric transformations. As for the generative process, we exploit the ODE sampler instead of the predictor-corrector sampler because the former is much faster than the latter. The tolerances of the absolute error and the relative error are both 1e-5. We tune several key hyper-parameters via grid search based on the validation dataset (see Table 2).

| Name   | Description                                      | Range                      |
|--------|--------------------------------------------------|----------------------------|
| $\sigma_s$ | The standard derivation when the acceleration is above the threshold. | [1e-3, 1e-2, 1e-1, 1]      |
| $\hat{a}$  | The acceleration threshold.                      | [1e-2, 1e-1, 1, 2, 5, 10]  |
| $\eta_{\sigma}$ | The harmonic acceleration constant.             | [1e-2, 1e-1, 1, 10]       |
| $\epsilon$ | The smallest time step for numerical stability in ODE sampler. | [1e-1, 0.2, 0.4, 0.8, 0.9, 0.99] |

**D.2 Additional Results**

Table 3 records the training loss of different models in MD17. It demonstrates the common phenomenon that models suffers a lot from error propagation. Though they achieves really low training losses, the error increases significantly as the time continues.

| Methods | Aspirin | Benzene | Ethanol | Malonaldehyde | Naphthalene | Salicylic | Toluene | Uracil |
|---------|---------|---------|---------|---------------|-------------|-----------|---------|--------|
| Naive   | 0.934   | 92.988  | 1.071   | 1.120         | 0.205       | 0.180     | 1.136   | 0.163  |
| TFN     | **0.139** | **0.073** | **0.200** | **0.140**     | **0.132**   | **0.156** | **0.122** | **0.128** |
| RF      | 2.444   | 0.581   | 2.072   | 1.796         | 1.977       | 2.648     | 2.217   | 2.103  |
| SE(3)-Tr. | 0.971   | 0.329   | 1.290   | 1.092         | 1.043       | 0.981     | 1.134   | 0.933  |
| EGNN    | 0.152   | **0.073** | 0.202   | 0.143         | 0.141       | 0.162     | 0.156   | 0.155  |
| GMN     | 0.189   | 0.080   | 0.222   | 0.179         | 0.188       | 0.200     | 0.192   | 0.184  |
| SCFNN   | 35.569  | 14.683  | 8.481   | 12.226        | 10.512      | 16.344    | 20.887  | 19.002 |
| ScoreMD | 0.686   | 1.023   | 1.153   | 2.305         | 0.620       | 0.633     | 0.653   | 0.945  |

**D.3 MD Trajectory of Benzene**

There we envision the MD trajectory of benzene with a total time span of 5K fts (see Fig. 5). It can be found that the inner structure of benzene is very stable but the trajectory fails to consider the PBC. Thus, the prediction error of benzene is much larger than other molecules.
D.4 Error Propagation Curves

Fig D.4 plots the changing trend of different models in the C7O2H10 dataset. It is worth noticing that the errors of SE(3)-Transformer, RF, TFN, and SCFNN explode acutely, and that of EGNN, GMN, and ScoreMD rise more smoothly.

Figure 5: The MD trajectory of benzene in MD17.

D.5 Generated MD Trajectory Samples

We present visualizations of samples in the generated trajectories from different approaches in Fig 7. For each method, we sample each 800 fts. The first row is the ground truth trajectory. It can be seen that even if EGNN, GMN, ScoreMD achieve low ARMSEs, their variance between different frames are tiny. Particularly, SCFNN is caught by a fixed point and its prediction keeps unchanged. That is, its output is the same as its input. On the other hand, other models including TFN, RF, and SE(3)-Transformer adjust the conformations dramatically, the generated conformations are mostly illegal. In other words, they produce molecular structures that break the underlying law of biological geometry.

Figure 6: Error propagation curves of the five target molecules for each model in the C7O2H10 dataset. The red line denotes the naive baseline for different target isomers.
Figure 7: **Examples of the generated MD trajectory for ISO_1004 generated in C7O2H10.** This first line is the ground truth trajectory of this compound.