Université de Montréal

Deep Geometric Probabilistic Models

par

Minkai Xu

Département d'informatique et de recherche opérationnelle
Faculté des arts et des sciences

Mémoire présenté en vue de l'obtention du grade de
Maître ès sciences (M.Sc.)
en informatique

October 25, 2022

Université de Montréal

Faculté des arts et des sciences

Ce mémoire intitulé

Deep Geometric Probabilistic Models

présenté par

Minkai Xu

a été évalué par un jury composé des personnes suivantes :

Miklos Csuros

(président-rapporteur)

Jian Tang

(directeur de recherche)

Golnoosh Farnadi

(membre du jury)

Résumé

La géométrie moléculaire, également connue sous le nom de conformation, est la représentation la plus intrinsèque et la plus informative des molécules. Cependant, prédire des conformations stables à partir de graphes moléculaires reste un problème difficile et fondamental en chimie et en biologie computationnelles. Les méthodes expérimentales et computationelles traditionnelles sont généralement coûteuses et chronophages. Récemment, nous avons assisté à des progrès considérables dans l'utilisation de l'apprentissage automatique, en particulier des modèles génératifs, pour accélérer cette procédure. Cependant, les approches actuelles basées sur les données n'ont généralement pas la capacité de modéliser des distributions complexes et ne tiennent pas compte de caractéristiques géométriques importantes. Dans cette thèse, nous cherchons à construire des modèles génératifs basés sur des principes pour la génération de conformation moléculaire qui peuvent surmonter les problèmes ci-dessus. Plus précisément, nous avons proposé des modèles de diffusion basés sur les flux, sur l'énergie et de débruitage pour la génération de structures moléculaires. Cependant, il n'est pas trivial d'appliquer ces modèles à cette tâche où la vraisemblance des géométries devrait avoir la propriété importante d'invariance par rotation par de translation. Inspirés par les progrès récents de l'apprentissage des représentations géométriques, nous fournissons à la fois une justification théorique et une mise en œuvre pratique sur la manière d'imposer cette propriété aux modèles. Des expériences approfondies sur des jeux de données de référence démontrent l'efficacité de nos approches proposées par rapport aux méthodes de référence existantes.

Mots-clés: Génération de conformation moléculaire, modèles génératifs profonds, flux continu de normalisation, modèles basés sur l'énergie, modèles probabilistes de diffusion

Abstract

Molecular geometry, also known as conformation, is the most intrinsic and informative representation of molecules. However, predicting stable conformations from molecular graphs remains a challenging and fundamental problem in computational chemistry and biology. Traditional experimental and computational methods are usually expensive and time-consuming. Recently, we have witnessed considerable progress in using machine learning, especially generative models, to accelerate this procedure. However, current data-driven approaches usually lack the capacity for modeling complex distributions and fail to take important geometric features into account. In this thesis, we seek to build principled generative models for molecular conformation generation that can overcome the above problems. Specifically, we proposed flow-based, energy-based, and denoising diffusion models for molecular structure generation. However, it's nontrivial to apply these models to this task where the likelihood of the geometries should have the important property of rotational and translation invariance. Inspired by the recent progress of geometric representation learning, we provide both theoretical justification and practical implementation about how to impose this property into the models. Extensive experiments on common benchmark datasets demonstrate the effectiveness of our proposed approaches over existing baseline methods.

Keywords: Molecular conformation generation, deep generative models, continuous normalizing flow, energy-based models, diffusion probabilistic models

Contents

List of tables

List of figures

Acknowledgement

There are a lot of people I want to thank. They gave me their unconditional and endless support and encouragement for my studies and lives, and helped me in various aspects of my career in the past several years.

Firstly, I'd especially like to thank my supervisor, Jian Tang, who provides continuous guidance and support throughout since my undergraduate. Jian is a great mentor — he always gave me the freedom to pursue my research interests, and also provide insightful feedback and supervision for my research. I am grateful to have such opportunity to work with him for nearly 4 years.

I'd also like to thank all of my collaborators, without whom all my research outcomes would be impossible. Specifically, I'd like to thank Stefano Ermon for his constructive advise. Rafael Gómez-Bombarelli for his helpful discussions. Lei Li, Mingxuan Wang, and Hao Zhou for hosting me at ByteDance AI Lab, and provides precious feedback on my research. Weinan Zhang for supervising me during my undergraduate. Yoshua Bengio for detailed guidance on several projects. Chence Shi for our long-term close collaboration and discussions. Wujie Wang and Chen Cai for the projects we've done together. Lantao Yu and Yang Song for the collaborations. Yuxuan Song and Zhiming Zhou for mentoring me in my early years.

There are also many people who were supportive to my life outside of research. I'd like to thank my girlfriend Shiyi Cao for her kind and careful company. Meng Qu, Zhaocheng Zhu, and Jie Fu for always help my life at Montreal and get me familiar with the lab quickly. Zuobai Zhang, Huiyu Cai, Jiarui Lu, and Dinghuai Zhang for all the happy hours we had enjoyed together.

Lastly, I would like to thank my parents for their forever support and love of my whole life, including this academic journey.

Chapter 1

Introduction

Recently, we have witnessed huge success of machine learning for molecular modeling, and especially, with deep learning (Goodfellow et al., 2016). Various deep learning models have shown effective in a variety of applications, such molecular property prediction (Gilmer et al., 2017), molecule generation (Jin et al., 2018; Shi et al., 2020b), and retrosynthesis prediction (Dai et al., 2019; Shi et al., 2020a). However, these models typically treat molecules as discrete graphs, where atoms are annotated as nodes and covalent chemical bonds as edges. Despite the empirical effectiveness, a more intrinsic and natural way to represent molecule is the *3D geometry*, also known as *molecular conformations*, where atoms are characterized by their 3D Cartesian coordinates. Molecular geometry contains much more critical information beyond graph representations like bond lengths and angles, which can directly determine many vital biological and physical properties, such as charge distribution and therapeutic interactions with proteins (Thomas et al., 2018; Jing et al., 2021). Indeed, recently a handful of studies have shown the effectiveness of using molecular conformations for molecular modeling tasks, *e.g.*, property prediction (Kearnes et al., 2016) and energy modeling (Behler and Parrinello, 2007; Rupp et al., 2012; Smith et al., 2017).

Though conformations play a key role in so many computational chemistry and biology applications, obtaining stable 3D structures of these molecules is still a challenging task. Traditional experimental methods by expensive crystallography are very slow and costly, and how to predict valid low-energy conformations efficiently and accurately has become an emerging and active research topic in modern computational chemistry. With decades of studies, current computational approaches are typically based on molecular dynamics (MD) or Markov chain Monte Carlo (MCMC) (De Vivo et al., 2016), which propose conformations by running simulations through expensive quantum calculations or approximate empirical potential (Ballard et al., 2015). Despite the progress that has been achieved, these methods are still either time-consuming (Ballard et al., 2015) or less accurate, making the performance not satisfying enough.

Recently, there are more and more efforts devoted to developing machine learning approaches, especially with deep generative model (Goodfellow et al., 2014; Kingma and Dhariwal, 2018; Song and Ermon, 2019), to accelerate this process (Mansimov et al., 2019; Simm and Hernández-Lobato, 2020). The problem is typically formulated as a conditional generative task, which aims to learn the conditional distribution $p(\boldsymbol{R}|\mathcal{G})$ of stable conformations $\boldsymbol{R}$ given the molecular graph representation $\mathcal{G}$. Such a model can be trained with just a collection of molecules with available stable conformations. Specifically, Mansimov et al. (2019) proposed to use Variational auto-encoders (VAE) (Kingma and Welling, 2013) models to directly predict the atomic coordinates. It first uses a message passing neural network (Gilmer et al., 2017) to extract atom representations from the molecular graph, then further generate the positions based on these atom embeddings. However, this method suffers the limitation of failing to impose the roto-translational invariance of molecular geometries. To overcome this problem, Simm and Hernández-Lobato (2020) propose to predict the distances between atoms instead of the 3D coordinates, and then generate molecular conformations by solving the distance geometry algorithm (Liberti et al., 2014). Since atomic pairwise distances will not be affected by rotation and translation, such approaches can effectively enjoy the roto-translational invariance of molecular conformations and therefore achieved promising results.

Despite the huge progress we have achieved, the current performance is still not satisfying enough and there remains a significant space for further improvement. This is mainly because of several challenges of the task. Firstly, the distribution of conformations is highly complex and multi-modal that each molecule usually has multiple diverse and stable conformations, which require a strong density modeling capacity of the probabilistic models. Secondly, the molecular conformations are complex geometries, which contain lots of essential geometric information, *e.g.*, pairwise relative direction tensors. However, naive graph neural networks are incapable of taking this important information into consideration.

In this thesis, we study how to tackle the above challenges and specifically propose two principled novel probabilistic models for molecular conformations. First, the previous methods (Mansimov et al., 2019; Simm and Hernández-Lobato, 2020) are mainly based on VAE models, which typically show inferior distribution modeling performance compared with more recent generative models (Vahdat et al., 2021). To this end, in this thesis, we explore introducing more advanced generative models to improve the generation capacity, including flow-based models (Dinh et al., 2017), energy-based models (Du and Mordatch, 2019), and diffusion models (Ho et al., 2020). However, how to build these models for molecular conformation generation is a non-trivial problem, where the learned likelihood is required to be roto-translationally invariant. Besides, the models also need to be parameterized with more advanced neural networks to incorporate the high-order features within the molecular geometries. In our work, we provide both theoretical justification for imposing invariance properties into the generative models, and practical solutions for parameterizing the models by the recent progress of geometric representation learning.

To facilitate further developments in this area, we also provide a suite of benchmark metrics to evaluate the quality of generated molecular conformations. The comprehensive benchmarks can measure both the quality and diversity of generated samples. We conduct extensive experiments to compare our proposed methods with competitive baseline approaches on the benchmarks. Results show that our model can significantly outperform existing data-driven methods, which demonstrates the effectiveness of our proposed approach.

1.1. Contributions

The main contributions of the thesis are summarized as follows:

- In this thesis, we propose two works (both have been accepted as publications at machine learning conferences) about deep generative models for molecular confirmation generation. We provide prologues of the two articles in Chap. 2 and Chap. 4 respectively, and elaborate the article details in Chap. 3 and Chap. 5. In the two works, we propose novel flow-based, energy-based, and denoising diffusion models for molecular geometry generation. We provide both the theoretical foundation and practical implementation of all these methods, combining the recent progress of both deep generative models and geometric representation learning.
- We provide several standardized benchmarks for evaluating the molecular confirmation generation models. We adopt the recently released dataset GEOM (Axelrod and Gomez-Bombarelli, 2020) and provide the benchmark setup compatible with several major deep learning frameworks that can evaluate both the quality and diversity of generated confirmations.
- We conduct comprehensive experiments to evaluate our methods as well as related state-of-the-art baselines on the benchmark. Results show that our method can consistently achieve better performance with a significant improvement, which demonstrates the effectiveness of our proposed approaches.

Chapter 2

Prologue to First Article

2.1. Article Details

Learning Neural Generative Dynamics for Molecular Conformation Generation. Minkai Xu*, Shitong Luo*, Yoshua Bengio, Jian Peng, Jian Tang. *9th International Conference on Learning Representations, 2021.*

2.2. Personal Contribution

(*) denotes co-first authorship. I came up with the idea of combining flow-based and energy-based model to design a new class of generative models, and specifically use this model for molecular conformation generation. Shitong Luo implemented the prototype of the model. Shitong Luo and I iterated on the model to improve the performance step by step. Jian Tang provide supervision for the whole project, and we also receive important feedback from Yoshua Bengio and Jian Peng during the process. I wrote up the first version of the paper. Shitong Luo and Jian Tang significantly helped improve its presentation, while Yoshua Bengio and Jian Peng also helped to polish the paper.

Chapter 3

Geometric Neural Generative Dynamics Models

3.1. Introduction

Recently, we have witnessed the success of graph-based representations for molecular modeling in a variety of tasks such as property prediction (Gilmer et al., 2017) and molecule generation (You et al., 2018; Shi et al., 2020b). However, a more natural and intrinsic representation of a molecule is its 3D structure, commonly known as the molecular geometry or *conformation*, which represents each atom by its 3D coordinate. The conformation of a molecule determines its biological and physical properties such as charge distribution, steric constraints, as well as interactions with other molecules. Furthermore, large molecules tend to comprise a number of rotatable bonds, which may induce flexible conformation changes and a large number of feasible conformations in nature. Generating valid and stable conformations of a given molecule remains very challenging. Experimentally, such structures are determined by expensive and time-consuming crystallography. Computational approaches based on Markov chain Monte Carlo (MCMC) or molecular dynamics (MD) (De Vivo et al., 2016) are computationally expensive, especially for large molecules (Ballard et al., 2015).

Machine learning methods have recently shown great potential for molecular conformation generation by training on a large collection of data to model the probability distribution of potential conformations $\boldsymbol{R}$ based on the molecular graph $\mathcal{G}$, *i.e.*, $p(\boldsymbol{R}|\mathcal{G})$. For example, Mansimov et al. (2019) proposed a Conditional Variational Graph Autoencoders (CVGAE) for molecular conformation generation. A graph neural network (Gilmer et al., 2017) is first applied to the molecular graph to get the atom representations, based on which 3D coordinates are further generated. One limitation of such an approach is that by directly generating the 3D coordinates of atoms it fails to model the rotational and translational invariance of molecular conformations. To address this issue, instead of generating the 3D coordinates directly, Simm and Hernández-Lobato (2020) recently proposed to first model the molecule's distance geometry (*i.e.*, the distances between atoms)—which are rotationally and translationally invariant—and then generate the molecular conformation based on the distance geometry through a post-processing algorithm (Liberti et al., 2014). Similar to Mansimov et al. (2019), a few layers of graph neural networks are applied to the molecular graph to learn the representations of different *edges*, which are further used to generate the distances of different edges independently. This approach is capable of more often generating valid molecular conformations.

Although these new approaches have made tremendous progress, the problem remains very challenging and far from solved. First, each molecule may have multiple stable conformations around a number of states which are thermodynamically stable. In other words, the distribution $p(\boldsymbol{R}|\mathcal{G})$ is very complex and multi-modal. Models with high capacity are required to model such complex distributions. Second, existing approaches usually apply a few layers of graph neural networks to learn the representations of nodes (or edges) and then generate the 3D coordinates (or distances) based on their representations independently. Such approaches are necessarily limited to capturing a single mode of $p(\boldsymbol{R}|\mathcal{G})$ (since the coordinates or distances are sampled independently) and are incapable of modeling multimodal joint distributions and the form of the graph neural net computation makes it difficult to capture long-range dependencies between atoms, especially in large molecules.

Inspired by the recent progress with deep generative models, this paper proposes a novel and principled probabilistic framework for molecular geometry generation, which addresses the above two limitations. Our framework combines the advantages of normalizing flows (Dinh et al., 2014) and energy-based approaches (LeCun et al., 2006), which have a strong model capacity for modeling complex distributions, are flexible to model long-range dependency between atoms, and enjoy efficient sampling and training procedures. Similar to the work of Simm and Hernández-Lobato (2020), we also first learn the distribution of distances $\boldsymbol{d}$ given the graph $\mathcal{G}$, *i.e.*, $p(\boldsymbol{d}|\mathcal{G})$, and define another distribution of conformations $\boldsymbol{R}$ given the distances $\boldsymbol{d}$, *i.e.*, $p(\boldsymbol{R}|\boldsymbol{d},\mathcal{G})$. Specifically, we propose a novel Conditional Graph Continuous Flow (CGCF) for distance geometry ($\boldsymbol{d}$) generation conditioned on the molecular graph $\mathcal{G}$. Given a molecular graph $\mathcal{G}$, CGCF defines an invertible mapping between a base distribution (*e.g.*, a multivariate normal distribution) and the molecular distance geometry, using a virtually infinite number of graph transformation layers on atoms represented by a Neural Ordinary Differential Equations architecture (Chen et al., 2018). Such an approach enjoys very high flexibility to model complex distributions of distance geometry. Once the molecular distance geometry $\boldsymbol{d}$ is generated, we further generate the 3D coordinates $\boldsymbol{R}$ by searching from the probability $p(\boldsymbol{R}|\boldsymbol{d},\mathcal{G})$.

Though the CGCF has a high capacity for modeling complex distributions, the distances of different edges are still independently updated in the transformations, which limits its capacity for modeling long-range dependency between atoms in the sampling process. Therefore, we further propose another unnormalized probability function, *i.e.*, an energy-based model (EBM) (Hinton and Salakhutdinov, 2006; LeCun et al., 2006; Ngiam et al., 2011), which acts as a tilting term of the flow-based distribution and directly models the joint distribution of $\boldsymbol{R}$. Specifically, the EBM trains an energy function $E(\boldsymbol{R}, \mathcal{G})$, which is approximated by a neural network. The flow- and energy-based models are combined in a novel way for joint training and mutual enhancement. First, energy-based methods are usually difficult to train due to the slow sampling process. In addition, the distribution of conformations is usually highly multi-modal, and the sampling procedures based on Gibbs sampling or Langevin Dynamics (Bengio et al., 2013a,b) tend to get trapped around modes, making it difficult to mix between different modes (Bengio et al., 2013a). Here we use the flow-based model as a proposal distribution for the energy model, which is capable to generate diverse samples for training energy models. Second, the flow-based model lacks the capacity to explicitly model the long-range dependencies between atoms, which we find can however be effectively modeled by an energy function $E(\boldsymbol{R}, \mathcal{G})$. Our sampling process can be therefore viewed as a *two-stage dynamic* system, where we first take the flow-based model to quickly synthesize realistic conformations and then used the learned energy $E(\boldsymbol{R}, \mathcal{G})$ to refine the generated conformations through Langevin Dynamics.

We conduct comprehensive experiments on several recently proposed benchmarks, including GEOM-QM9, GEOM-Drugs (Axelrod and Gomez-Bombarelli, 2020) and ISO17 (Simm and Hernández-Lobato, 2020). Numerical evaluations show that our proposed framework consistently outperforms the previous state-of-the-art (GraphDG) on both conformation generation and distance modeling tasks, with a clear margin.

3.2. Problem Definition and Preliminaries

3.2.1. Problem Definition

Notations. Following existing work (Simm and Hernández-Lobato, 2020), each molecule is represented as an undirected graph $\mathcal{G} = \langle \mathcal{V}, \mathcal{E} \rangle$, where $\mathcal{V}$ is the set of nodes representing atoms and $\mathcal{E}$ is the set of edges representing inter-atomic bonds. Each node v in $\mathcal{V}$ is labeled with atomic properties such as element type. The edge in $\mathcal{E}$ connecting u and v is denoted as e_{uv}, and is labeled with its bond type. We also follow the previous work (Simm and Hernández-Lobato, 2020) to expand the molecular graph with auxiliary bonds, which is elaborated in Appendix 3.6.2. For the molecular 3D representation, each atom in $\mathcal{V}$ is assigned with a 3D position vector $\boldsymbol{r} \in \mathbb{R}^3$. We denote $d_{uv} = \|\boldsymbol{r}_u - \boldsymbol{r}_v\|_2$ as the Euclidean distance between the u^{th} and v^{th} atom. Therefore, we can represent all the positions $\{\boldsymbol{r}_v\}_{v \in \mathcal{V}}$

as a matrix $\boldsymbol{R} \in \mathbb{R}^{|\mathcal{V}| \times 3}$ and all the distances between connected nodes $\{d_{uv}\}_{e_{uv} \in \mathcal{E}}$ as a vector $\boldsymbol{d} \in \mathbb{R}^{|\mathcal{E}|}$.

Problem Definition. The problem of *molecular conformation generation* is defined as a conditional generation process. More specifically, our goal is to model the conditional distribution of atomic positions $\boldsymbol{R}$ given the molecular graph $\mathcal{G}$, *i.e.*, $p(\boldsymbol{R}|\mathcal{G})$.

3.2.2. Preliminaries

Continuous Normalizing Flow. A normalizing flow (Dinh et al., 2014; Rezende and Mohamed, 2015) defines a series of invertible deterministic transformations from an initial known distribution $p(z)$ to a more complicated one $p(x)$. Recently, normalizing flows have been generalized from discrete number of layers to continuous (Chen et al., 2018; Grathwohl et al., 2018) by defining the transformation f_θ as a continuous-time dynamic $\frac{\partial z(t)}{\partial t} = f_\theta(z(t), t)$. Formally, with the latent variable $z(t_0) \sim p(z)$ at the start time, the continuous normalizing flow (CNF) defines the transformation $x = z(t_0) + \int_{t_0}^{t_1} f_\theta(z(t),t) dt$. Then the exact density for $p_\theta(x)$ can be computed by:

$$\log p_\theta(x) = \log p(z(t_0)) - \int_{t_0}^{t_1} \operatorname{Tr}\left(\frac{\partial f_\theta}{\partial z(t)}\right) dt \tag{3.2.1}$$

where $z(t_0)$ can be obtained by inverting the continuous dynamic $z(t_0) = x + \int_{t_1}^{t_0} f_\theta(z(t), t) dt$. A black-box ordinary differential equation (ODE) solver can be applied to estimate the outputs and inputs gradients and optimize the CNF model (Chen et al., 2018; Grathwohl et al., 2018).

Energy-based Models. Energy-based models (EBMs) (Dayan et al., 1995; Hinton and Salakhutdinov, 2006; LeCun et al., 2006) use a scalar parametric energy function $E_\phi(x)$ to fit the data distribution. Formally, the energy function induces a density function with the Boltzmann distribution $p_\phi(x) = \exp(-E_\phi(x))/Z(\phi)$, where $Z = \int \exp(-E_\phi(x))\, dx$ denotes the partition function. EBM can be learned with Noise contrastive estimation (NCE) (Gutmann and Hyvärinen, 2010) by treating the normalizing constant as a free parameter. Given the training examples from both the dataset and a noise distribution $q(x)$, ϕ can be estimated by maximizing the following objective function:

$$J(\phi) = \mathbb{E}_{p_{\text{data}}}\Big[\log \frac{p_\phi(x)}{p_\phi(x) + q(x)}\Big] + \mathbb{E}_q\Big[\log \frac{q(x)}{p_\phi(x) + q(x)}\Big], \tag{3.2.2}$$

which turns the estimation of EBM into a discriminative learning problem. Sampling from E_ϕ can be done with a variety of methods such as Markov chain Monte Carlo (MCMC) or Gibbs sampling (Hinton and Salakhutdinov, 2006), possibly accelerated using Langevin dynamics (Du and Mordatch, 2019; Song et al., 2020b), which leverages the gradient of the

EBM to conduct sampling:

$$x_k = x_{k-1} - \frac{\epsilon}{2}\nabla_x E_\phi\left(x_{k-1}\right) + \sqrt{\epsilon}\omega, \omega \sim \mathcal{N}(0, \mathcal{I}), \tag{3.2.3}$$

where ϵ refers to the step size. x_0 are the samples drawn from a random initial distribution and we take the x_K with K Langevin dynamics steps as the generated samples of the stationary distribution.

3.3. Method

3.3.1. Overview

We first present a high-level description of our model. Directly learning a generative model on Cartesian coordinates heavily depends on the (arbitrary) rotation and translation (Mansimov et al., 2019). Therefore, in this paper we take the atomic pairwise distances as intermediate variables to generate conformations, which are invariant to rotation and translation. More precisely, the cornerstone of our method is to factorize the conditional distribution $p_\theta(\boldsymbol{R}|\mathcal{G})$ into the following formulation:

$$p_\theta(\boldsymbol{R}|\mathcal{G}) = \int p(\boldsymbol{R}|\boldsymbol{d},\mathcal{G}) \cdot p_\theta(\boldsymbol{d}|\mathcal{G})\, \mathrm{d}\boldsymbol{d}, \tag{3.3.1}$$

where $p_\theta(\boldsymbol{d}|\mathcal{G})$ models the distribution of inter-atomic distances given the graph $\mathcal{G}$ and $p(\boldsymbol{R}|\boldsymbol{d},\mathcal{G})$ models the distribution of conformations given the distances $\boldsymbol{d}$. In particular, the conditional generative model $p_\theta(\boldsymbol{d}|\mathcal{G})$ is parameterized as a conditional graph continuous flow, which can be seen as a continuous dynamics system to transform the random initial noise to meaningful distances. This flow model enables us to capture the long-range dependencies between atoms in the hidden space during the dynamic steps.

Though CGCF can capture the dependency between atoms in the hidden space, the distances of different edges are still independently updated in the transformations, which limits the capacity of modeling the dependency between atoms in the sampling process. Therefore we further propose to correct $p_\theta(\boldsymbol{R}|\mathcal{G})$ with an energy-based tilting term $E_\phi(\boldsymbol{R}, \mathcal{G})$:

$$p_{\theta,\phi}(\boldsymbol{R}|\mathcal{G}) \propto p_\theta(\boldsymbol{R}|\mathcal{G}) \cdot \exp(-E_\phi(\boldsymbol{R}, \mathcal{G})). \tag{3.3.2}$$

The tilting term is directly defined on the joint distribution of $\boldsymbol{R}$ and $\mathcal{G}$, which explicitly captures the long-range interaction directly in observation space. The tilted distribution $p_{\theta,\phi}(\boldsymbol{R}|\mathcal{G})$ can be used to provide refinement or optimization for the conformations generated from $p_\theta(\boldsymbol{R}|\mathcal{G})$. This energy function is also designed to be invariant to rotation and translation.

In the following parts, we will firstly describe our flow-based generative model $p_\theta(\boldsymbol{R}|\mathcal{G})$ in Section 3.3.2 and elaborate the energy-based tilting model $E_\phi(\boldsymbol{R},\mathcal{G})$ in Section 3.3.3. Then we

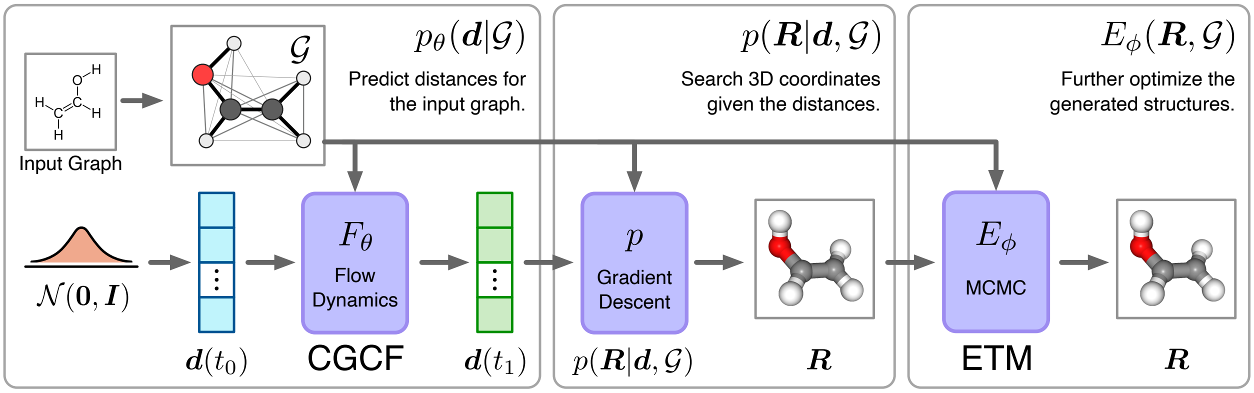
Input Graph
$\mathcal{G}$
$p_\theta(\boldsymbol{d}|\mathcal{G})$
Predict distances for the input graph.
$p(\boldsymbol{R}|\boldsymbol{d},\mathcal{G})$
Search 3D coordinates given the distances.
$E_\phi(\boldsymbol{R},\mathcal{G})$
Further optimize the generated structures.
$\mathcal{N}(\mathbf{0},\boldsymbol{I})$
$\boldsymbol{d}(t_0)$
F_θ
Flow Dynamics
CGCF
$\boldsymbol{d}(t_1)$
p
Gradient Descent
$p(\boldsymbol{R}|\boldsymbol{d},\mathcal{G})$
$\boldsymbol{R}$
E_ϕ
MCMC
ETM
$\boldsymbol{R}$

Fig. 3.1. Illustration of the proposed framework. Given the molecular graph, we 1) first draw latent variables from a Gaussian prior, and transform them to the desired distance matrix through the Conditional Graph Continuous Flow (CGCF); 2) search the possible 3D coordinates according to the generated distances and 3) further optimize the generated conformation via a MCMC procedure with the Energy-based Tilting Model (ETM).

introduce the two-stage sampling process with both deterministic and stochastic dynamics in Section 3.3.4. An illustration of the whole framework is given in Fig. 3.1.

3.3.2. Flow-based Generative Model

Conditional Graph Continuous Flows $p_\theta(\boldsymbol{d}|\mathcal{G})$. We parameterize the conditional distribution of distances $p_\theta(\boldsymbol{d}|\mathcal{G})$ with the continuous normalizing flow, named **C**onditional **G**raph **C**ontinuous **F**low (CGCF). CGCF defines the distribution through the following dynamics system:

$$\boldsymbol{d} = F_\theta(\boldsymbol{d}(t_0), \mathcal{G}) = \boldsymbol{d}(t_0) + \int_{t_0}^{t_1} f_\theta(\boldsymbol{d}(t), t; \mathcal{G})\mathrm{d}t, \quad \boldsymbol{d}(t_0) \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I}) \tag{3.3.3}$$

where the dynamic f_θ is implemented by Message Passing Neural Networks (MPNN) (Gilmer et al., 2017), which is a widely used architecture for representation learning on molecular graphs. MPNN takes node attributes, edge attributes and the bonds lengths $\boldsymbol{d}(t)$ as input to compute the node and edge embeddings. Each message passing layer updates the node embeddings by aggregating the information from neighboring nodes according to its hidden vectors of respective nodes and edges. Final features are fed into a neural network to compute the value of the dynamic f_θ for all distances independently. As $t_1 \to \infty$, our dynamic can have an infinite number of steps and is capable to model long-range dependencies. The invertibility of F_θ allows us to not only conduct fast sampling, but also easily optimize the parameter set θ by minimizing the exact negative log-likelihood:

$$\mathcal{L}_{\text{mle}}(\boldsymbol{d}, \mathcal{G}; \theta) = -\mathbb{E}_{p_{\text{data}}} \log p_\theta(\boldsymbol{d}|\mathcal{G}) = -\mathbb{E}_{p_{\text{data}}} \left[\log p(\boldsymbol{d}(t_0)) + \int_{t_0}^{t_1} \operatorname{Tr}\left(\frac{\partial f_{\theta,G}}{\partial \boldsymbol{d}(t)} \right) dt \right]. \tag{3.3.4}$$

Closed-form $p(\boldsymbol{R}|\boldsymbol{d},\mathcal{G})$. The generated pair-wise distances can be converted into 3D structures through postprocessing methods such as the classic Euclidean Distance Geometry (EDG) algorithm. In this paper, we adopt an alternative way by defining the conformations as a conditional distribution:

$$p(\boldsymbol{R}|\boldsymbol{d}, \mathcal{G}) = \frac{1}{Z} \exp\Big\{ - \sum_{e_{uv} \in \mathcal{E}} \alpha_{uv} \Big(\|\boldsymbol{r}_u - \boldsymbol{r}_v\|_2 - d_{uv} \Big)^2 \Big\}, \tag{3.3.5}$$

where Z is the partition function to normalize the probability and $\{\alpha_{uv}\}$ are parameters that control the variance of desired Cartesian coordinates, which can be either learned or manually designed according to the graph structure $\mathcal{G}$. With the probabilistic formulation, we can conduct either sampling via MCMC or searching the local optimum with optimization methods. This simple function is fast to calculate, making the generation procedure very efficient with a negligible computational cost.

Compared with the conventional EDG algorithm adopted in GraphDG (Simm and Hernández-Lobato, 2020), our probabilistic solution enjoys following advantages: 1) $p(\boldsymbol{R}|\boldsymbol{d}, \mathcal{G})$ enables the calculation for the likelihood $p_\theta(\boldsymbol{R}|\mathcal{G})$ of Eq. 3.3.1 by approximation methods, and thus can be further combined with the energy-based tilting term $E_\phi(\boldsymbol{R},\mathcal{G})$ to induce a superior distribution; 2) GraphDG suffers the drawback that when invalid sets of distances are generated, EDG will fail to construct 3D structure. By contrast, our method can always be successful to generate conformations by sampling from the distribution $p(\boldsymbol{R}|\boldsymbol{d},\mathcal{G})$.

3.3.3. Energy-based Tilting Model

The last part of our framework is the **E**nergy-based **T**iling **M**odel (ETM) $E_\phi(\boldsymbol{R},\mathcal{G})$, which helps model the long-range interactions between atoms explicitly in the observation space. $E_\phi(\boldsymbol{R},\mathcal{G})$ takes the form of SchNet (Schütt et al., 2017), which is widely used to model the potential-energy surfaces and energy-conserving force fields for molecules. The continuous-filter convolutional layers in SchNet allow each atom to aggregate the representations of all single, pairwise, and higher-order interactions between the atoms through non-linear functions. The final atomic representations are pooled to a single vector and then passed into a network to produce the scalar output.

Typically the EBMs can be learned by maximum likelihood, which usually requires the lengthy MCMC procedure and is time-consuming for training. In this work, we learn the ETM by Noise Contrastive Estimation (Gutmann and Hyvärinen, 2010), which is much more efficient. In practice, the noise distribution is required to be close to data distribution, otherwise the classification problem would be too easy and would not guide E_ϕ to learn much about the modality of the data. We propose to take the pre-trained CGCF to serve as a strong noise distribution, leading to the following discriminative learning objective for the

ETM[2]:

$$\mathcal{L}_{\text{nce}}(\boldsymbol{R},\mathcal{G};\phi) = -\,\mathbb{E}_{p_{\text{data}}}\Big[\log \frac{1}{1+\exp(E_\phi(\boldsymbol{R},\mathcal{G}))}\Big] - \mathbb{E}_{p_\theta}\Big[\log \frac{1}{1+\exp(-E_\phi(\boldsymbol{R},\mathcal{G}))}\Big]. \tag{3.3.6}$$

3.3.4. Sampling

We employ a two-stage dynamic system to synthesize a possible conformation given the molecular graph representation $\mathcal{G}$. In the first stage, we first draw a latent variable $\hat{z}$ from the Gaussian prior $\mathcal{N}(0,I)$, and then pass it through the continuous deterministic dynamics model F_θ defined in Eq. 3.3.3 to get $\hat{\boldsymbol{d}}_0 = F_\theta(\hat{z}_0,G)$. Then an optimization procedure such as stochastic gradient descent is employed to search the realistic conformations $\boldsymbol{R}$ with local maximum probability of $p(\boldsymbol{R}|\boldsymbol{d},\mathcal{G})$ (defined in Eq. 3.3.5). By doing this, an initial conformation $\boldsymbol{R}^{(0)}$ can be generated. In the second stage, we further refine the initial conformation $\boldsymbol{R}^{(0)}$ with the energy-based model defined in Eq. 3.3.2 with K steps of Langevin dynamics:

$$\begin{aligned} &\boldsymbol{R}_k = \boldsymbol{R}_{k-1} - \frac{\epsilon}{2}\nabla_{\boldsymbol{R}} E_{\theta,\phi}\left(\boldsymbol{R}|\mathcal{G}\right) + \sqrt{\epsilon}\omega, \omega \sim \mathcal{N}(0,\mathcal{I}), \\ &\text{where } E_{\theta,\phi}(\boldsymbol{R}|\mathcal{G}) = -\log p_{\theta,\phi}(\boldsymbol{R}|\mathcal{G}) = E_\phi(\boldsymbol{R},\mathcal{G}) - \log \int p(\boldsymbol{R}|\boldsymbol{d},\mathcal{G})p_\theta(\boldsymbol{d}|\mathcal{G})\mathrm{d}\boldsymbol{d}. \end{aligned} \tag{3.3.7}$$

where ϵ denotes the step size. The second integration term in $E_{\theta,\phi}$ can be estimated through approximate methods. In practice, we use Monte Carlo Integration to conduct the approximation, which is simple yet effective with just a few distance samples from the CGCF model $p_\theta(\boldsymbol{d}|\mathcal{G})$.

3.4. Experiments

3.4.1. Experiment Setup

Evaluation Tasks. To evaluate the performance of proposed model, we conduct experiments by comparing with the counterparts on: (1) **Conformation Generation** evaluates the model's capacity to learn the distribution of conformations by measuring the diversity and accuracy of generated samples (section 3.4.2); (2) **Distribution over distances** is first proposed in Simm and Hernández-Lobato (2020), which concentrate on the distance geometry of generated conformations (section 3.4.2).

Benchmarks. We use the recent proposed GEOM-QM9 and GEOM-Drugs (Axelrod and Gomez-Bombarelli, 2020) datasets for conformation generation task and ISO17 dataset (Simm and Hernández-Lobato, 2020) for distances modeling task. The choice of different datasets is because of their distinct properties. Specifically, GEOM datasets consist of stable conformations, which is suitable to evaluate the conformation generation task. By contrast, ISO17 contains snapshots of molecular dynamics simulations, where the structures are not

[2] Detailed derivations of the training loss can be found in Appendix 3.6.6.

equilibrium conformations but can reflect the density around the equilibrium state. Therefore, it is more suitable for the assessment of similarity between the model distribution and the data distribution around equilibrium states.

More specifically, **GEOM-QM9** is an extension to the QM9 (Ramakrishnan et al., 2014) dataset: it contains multiple conformations for most molecules while the original QM9 only contains one. This dataset is limited to 9 heavy atoms (29 total atoms), with small molecular mass and few rotatable bonds. We randomly draw 50000 conformation-molecule pairs from GEOM-QM9 to be the training set, and take another 17813 conformations covering 150 molecular graphs as the test set. **GEOM-Drugs** dataset consists of much larger drug molecules, up to a maximum of 181 atoms (91 heavy atoms). It also contains multiple conformations for each molecule, with a larger variance in structures, *e.g.*, there are the 6.5 rotatable bonds in average. We randomly take 50000 conformation-molecule pairs from GEOM-Drugs as the training set, and another 9161 conformations (covering 100 molecular graphs) as the test split. **ISO17** dataset is also built upon QM9 datasets, which consists of 197 molecules, each with 5000 conformations. Following Simm and Hernández-Lobato (2020), we also split ISO17 into the training set with 167 molecules and the test set with another 30 molecules.

Baselines. We compared our proposed method with the following state-of-the-art conformation generation methods. **CVGAE** (Mansimov et al., 2019) uses a conditional version of VAE to directly generate the 3D coordinates of atoms given the molecular graph. **GraphDG** (Simm and Hernández-Lobato, 2020) also employs the conditional VAE framework. Instead of directly modeling the 3D structure, they propose to learn the distribution over distances. Then the distances are converted into conformations with an EDG algorithm. Furthermore, we also take **RDKit** (Riniker and Landrum, 2015) as a baseline model, which is a classical EDG approach built upon extensive calculation collections in computational chemistry.

3.4.2. Conformation Generation

In this section, we evaluate the ability of the proposed method to model the equilibrium conformations. We focus on both the *diversity* and *accuracy* of the generated samples. More specifically, diversity measures the model's capacity to generate multi-modal conformations, which is essential for discovering new conformations, while accuracy concentrates on the similarity between generated conformations and the equilibrium conformations.

Evaluation. For numerical evaluations, we follow previous work (Hawkins, 2017; Mansimov et al., 2019) to calculate the Root-Mean-Square Deviation (RMSD) of the heavy atoms between generated samples and reference ones. Precisely, given the generated conformation $\boldsymbol{R}$ and the reference $\boldsymbol{R}^*$, we obtain $\hat{\boldsymbol{R}}$ by translating and rotating $\boldsymbol{R}^*$ to minimize the

Tableau 3.1. Comparison of different methods on the COV and MAT scores. Top 4 rows: deep generative models for molecular conformation generation. Bottom 5 rows: different methods that involve an additional rule-based force field to further optimize the generated structures.

Dataset	GEOM-QM9				GEOM-Drugs			
Metric	COV* (%)		MAT (Å)		COV* (%)		MAT (Å)	
	Mean	Median	Mean	Median	Mean	Median	Mean	Median
CVGAE	8.52	5.62	0.7810	0.7811	0.00	0.00	2.5225	2.4680
GraphDG	55.09	56.47	0.4649	0.4298	7.76	0.00	1.9840	2.0108
CGCF	69.60	70.64	0.3915	0.3986	49.92	41.07	1.2698	1.3064
CGCF + ETM	**72.43**	**74.38**	**0.3807**	**0.3955**	**53.29**	**47.06**	**1.2392**	**1.2480**
RDKit	**79.94**	**87.20**	0.3238	**0.3195**	65.43	70.00	1.0962	1.0877
CVGAE + FF	63.10	60.95	0.3939	0.4297	83.08	95.21	0.9829	0.9177
GraphDG + FF	70.67	70.82	0.4168	0.3609	84.68	93.94	0.9129	0.9090
CGCF + FF	**73.52**	**72.75**	0.3131	0.3251	92.28	98.15	**0.7740**	**0.7338**
CGCF + ETM + FF	**73.54**	72.58	**0.3088**	**0.3210**	**92.41**	**98.57**	**0.7737**	0.7616

* For the reported COV score, the threshold δ is set as 0.5Å for QM9 and 1.25Å for Drugs. More results of COV scores with different threshold δ are given in Appendix 3.6.8.

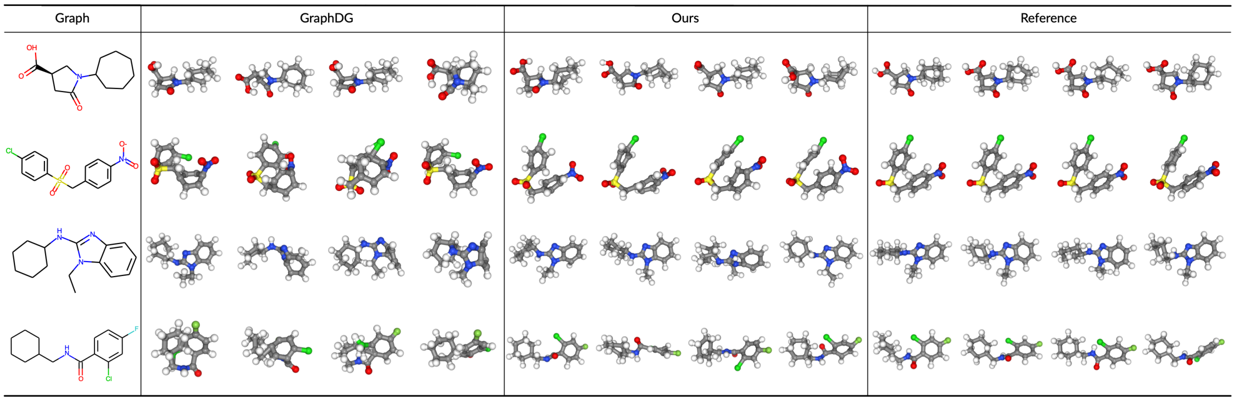

Fig. 3.2. Visualization of generated conformations from the state-of-the-art baseline (GraphDG), our method and the ground-truth, based on four random molecular graphs from the test set of GEOM-Drugs. C, O, H, S and Cl are colored gray, red, white, yellow and green respectively.

following predefined RMSD metric:

$$\mathrm{RMSD}(\boldsymbol{R}, \hat{\boldsymbol{R}}) = \left(\frac{1}{n}\sum_{i=1}^{n}\|\boldsymbol{R}_i - \hat{\boldsymbol{R}}_i\|^2\right)^{\frac{1}{2}}, \tag{3.4.1}$$

where n is the number of heavy atoms. Then the smallest distance is taken as the evaluation metric. Built upon the RMSD metric, we define **Cov**erage (COV) and **Mat**ching (MAT) score to measure the diversity and quality respectively. Intuitively, COV measures the fraction of conformations in the reference set that are matched by at least one conformation in the generated set. For each conformation in the generated set, its neighbors in the reference set within a given RMSD threshold δ are marked as matched:

$$\mathrm{COV}(\mathbb{S}_g(\mathcal{G}), \mathbb{S}_r(\mathcal{G})) = \frac{1}{|\mathbb{S}_r|}\left|\left\{\boldsymbol{R} \in \mathbb{S}_r \middle| \mathrm{RMSD}(\boldsymbol{R}, \boldsymbol{R}') < \delta, \exists \boldsymbol{R}' \in \mathbb{S}_g\right\}\right|, \tag{3.4.2}$$

where $\mathbb{S}_g(\mathcal{G})$ denotes the generated conformations set for molecular graph $\mathcal{G}$, and $\mathbb{S}_r(\mathcal{G})$ denotes the reference set. In practice, the number of samples in the generated set is two times of the reference set. Typically, a higher COV score means the a better diversity performance. The COV score is able to evaluate whether the generated conformations are diverse enough to cover the ground-truth.

While COV is effective to measure the diversity and detect the mode-collapse case, it is still possible for the model to achieve high COV with a high threshold tolerance. Here we define the MAT score as a complement to measure the quality of generated samples. For each conformation in the reference set, the RMSD distance to its nearest neighbor in the generated set is computed and averaged:

$$\mathrm{MAT}(\mathbb{S}_g(\mathcal{G}), \mathbb{S}_r(\mathcal{G})) = \frac{1}{|\mathbb{S}_r|}\sum_{\boldsymbol{R}' \in \mathbb{S}_r} \min_{\boldsymbol{R} \in \mathbb{S}_g} \mathrm{RMSD}(\boldsymbol{R}, \boldsymbol{R}'). \tag{3.4.3}$$

This metric concentrate on the accuracy of generated conformations. More realistic generated samples lead to a lower matching score.

Results. Tab. 3.1 shows that compared with the existing state-of-the-art baselines, our CGCF model can already achieve superior performance on all four metrics (top 4 rows). As a CNF-based model, CGCF holds much the higher generative capacity for both diversity and quality compared than VAE approaches. The results are further improved when combined with ETM to explicitly incorporate the long-range correlations. We visualize several representative examples in Fig. 3.2, and leave more examples in Appendix 3.6.7. A meaningful observation is that though competitive over other neural models, the rule-based RDKit method occasionally shows better performance than our model, which indicates that RDKit can generate more realistic structures. We argue that this is because after generating the initial coordinates, RDKit involves additional hand-designed molecular force field (FF) energy functions (Rappé et al., 1992; Halgren, 1996a) to find the stable conformations with local minimal energy. By contrast, instead of finding the local minimums, our deep generative models aim to model and sample from the potential distribution of structures. To yield a better comparison, we further test our model by taking the generated structures as initial states and utilize the Merck Molecular Force Field (MMFF) (Halgren, 1996a) to find the local stable points. A more precise description of about the MMFF Force Field algorithms in RDKit is given in Appendix 3.6.9. This postprocessing procedure is also employed in the previous work (Mansimov et al., 2019). Additional results in Tab. 3.1 verify our conjecture that FF plays a vital role in generating more realistic structures, and demonstrate the capacity of our method to generate high-quality initial coordinates.

Tableau 3.2. Comparison of distances density modeling with different methods. We compare the marginal distribution of single ($p(d_{uv}|\mathcal{G})$), pair ($p(d_{uv},d_{ij}|\mathcal{G})$) and all ($p(\boldsymbol{d}|\mathcal{G})$) edges between C and O atoms. Molecular graphs $\mathcal{G}$ are taken from the test set of ISO17. We take two metrics into consideration: 1) **median** MMD between the ground truth and generated ones, and 2) **mean** ranking (1 to 3) based on the MMD metric.

	Single		Pair		All	
	Mean	Median	Mean	Median	Mean	Median
RDKit	3.4513	3.1602	3.8452	3.6287	4.0866	3.7519
CVGAE	4.1789	4.1762	4.9184	5.1856	5.9747	5.9928
GraphDG	0.7645	0.2346	0.8920	0.3287	1.1949	0.5485
CGCF	**0.4490**	**0.1786**	**0.5509**	**0.2734**	**0.8703**	**0.4447**
CGCF + ETM	0.5703	0.2411	0.6901	0.3482	1.0706	0.5411

3.4.3. Distributions Over Distances

Tough primarily designed for 3D coordinates, we also following Simm and Hernández-Lobato (2020) to evaluate the generated distributions of pairwise distance, which can be viewed as a representative element of the model capacity to model the inter-atomic interactions.

Evaluation. Let $p(d_{uv}|\mathcal{G})$ denote the conditional distribution of distances on each edge e_{uv} given a molecular graph $\mathcal{G}$. The set of distances are computed from the generated conformations $\boldsymbol{R}$. We calculate maximum mean discrepancy (MMD) (Gretton et al., 2012) to compare the generated distributions and the ground-truth distributions. Specifically, we evaluate the distribution of individual distances $p(d_{uv}|\mathcal{G})$, pair distances $p(d_{uv}, d_{ij}|\mathcal{G})$ and all distances $p(\boldsymbol{d}|\mathcal{G})$. For this benchmark, the number of samples in the generated set is the same as the reference set.

Results. The results of MMD are summarized in Tab. 3.2. The statistics show that RDKit suffers the worst performance, which is because it just aims to generate the most stable structures as illustrated in Section 3.4.2. For CGCF, the generated samples are significantly closer to the ground-truth distribution than baseline methods, where we consistently achieve the best numerical results. Besides, we notice that ETM will slightly hurt the performance in this task. However, one should note that this phenomenon is natural because typically ETM will sharpen the generated distribution towards the stable conformations with local minimal energy. By contrast, the ISO17 dataset consists of snapshots of molecular dynamics where the structures are not equilibrium conformations but samples from the density around the equilibrium state. Therefore, ETM will slightly hurt the results. This phenomenon is also consistent with the observations for RDKit. Instead of generating unbiased samples from the underlying distribution, RDKit will only generate the stable ones with local minimal energy by involving the hand-designed molecular force field (Simm and Hernández-Lobato,

Tableau 3.3. Conformation Diversity. Mean and Std represent the corresponding mean and standard deviation of pairwise RMSD between the generated conformations per molecule.

	RDKit	CVGAE	GraphDG	CGCF	CGCF +ETM
Mean	0.083	0.207	0.249	0.810	0.741
Std	0.054	0.187	0.104	0.223	0.206

2020). And as shown in the results, though highly competitive in Tab. 3.1, RDKit also suffers much weaker results in Tab. 3.2. The marginal distributions $P(d_{uv}|\mathcal{G})$ for pairwise distances in visualized in Appendix 3.6.11, which further demonstrate the superior capacity of our proposed method.

We also follow Mansimov et al. (2019) to calculate the diversity of conformations generated by all compared methods, which is measured by calculating the mean and standard deviation of the pairwise RMSD between each pair of generated conformations per molecule. The results shown in Tab. 3.3 demonstrate that while our method can achieve the lowest MMD, it does not collapse to generating extremely similar conformations. Besides, we observe that ETM will slightly hurt the diversity of CGCF, which verifies our statement that ETM will sharpen the generated distribution towards the stable conformations with local minimal energy.

3.5. Conclusion and Future Work

In this paper, we propose a novel probabilistic framework for molecular conformation generation. Our generative model combines the advantage of both flow-based and energy-based models, which is capable of modeling the complex multi-modal geometric distribution and highly branched atomic correlations. Experimental results show that our method outperforms all previous state-of-the-art baselines on the standard benchmarks. Future work includes applying our framework on much larger datasets and extending it to more challenging structures (*e.g.*, proteins).

3.6. Appendix

3.6.1. Related Works

Conformation Generation. There have been results showing deep learning speeding up molecular dynamics simulation by learning efficient alternatives to quantum mechanics-based energy calculations (Schütt et al., 2017; Smith et al., 2017). However, though accelerated by neural networks, these approaches are still time-consuming due to the lengthy MCMC process. Recently, Gebauer et al. (2019) and Hoffmann and Noé (2019) propose to directly generate 3D structures with deep generative models. However, these models can hardly capture graph- or bond-based structure, which is typically complex and highly branched. Some other works (Lemke and Peter, 2019; AlQuraishi, 2019; Ingraham et al., 2019; Noé et al., 2019; Senior et al., 2020) also focus on learning models to directly generate 3D structure, but focus on the protein folding problem. Unfortunately, proteins are linear structures while general molecules are highly branched, making these methods not naturally transferable to general molecular conformation generation tasks.

Energy-based Generative Model. There has been a long history for energy-based generative models. Xie et al. (2016) proposes to train an energy-based model parameterized by modern deep neural network and learned it by Langevin based MLE. The model is called generative ConvNet since it can be derived from the discriminative ConvNet. In particular, this paper is the first to formulate modern ConvNet-parametrized EBM as exponential tilting of a reference distribution, and connect it to discriminative ConvNet classifier. More recently, Du and Mordatch (2019) implemented the deep EBMs with ConvNet as energy function and achieved impressive results on image generation.

Different from the previous works, we concentrate on molecular geometry generation, and propose a novel and principled probabilistic framework to address the domain-specific problems. More specifically, we first predict the atomic distances through the continuous normalizing flow, and then convert them to the desired 3D conformation and optimize it with the energy-based model. This procedure enables us to keep the rotational and translational invariance property. Besides, to the best of our knowledge, we are the first one to combine neural ODE with EBMs. We take the ODE model to improve the training of EBM, and combine both to conduct the two-stage sampling dynamics.

3.6.2. Data Preprocess

Inspired by classic molecular distance geometry (Crippen et al., 1988), we also generate the confirmations by firstly predicting all the pairwise distances, which enjoys the invariant property to rotation and translation. Since the bonds existing in the molecular graph are not sufficient to characterize a conformation, we pre-process the graphs by extending *auxiliary* edges. Specifically, the atoms that are 2 or 3 hops away are connected with *virtual bonds*, labeled differently from the real bonds of the original graph. These extra edges contribute to reducing the degrees of freedom in the 3D coordinates, with the edges between second neighbors helping to fix the angles between atoms, and those between third neighbors fixing dihedral angles.

3.6.3. Network Architecture

In this section, we elaborate on the network architecture details of CGCF and ETM.

3.6.3.1. Continuous Graph Flow. In CGCF, the dynamic function f_θ defined in Eq. 3.3.3 is instanced with a message passing neural networks. Given the node attributes, edge attributes and intermediate edge lengths as input, we first embed them into the feature space through feedforward networks:

$$\begin{aligned} \boldsymbol{h}_v^{(0)} &= \text{NodeEmbedding}(v), \quad v \in \mathcal{V}, \\ \boldsymbol{h}_{e_{uv}} &= \text{EdgeEmbedding}(e_{uv}, d_{uv}(t_0)), \quad e_{uv} \in \mathcal{E}. \end{aligned} \tag{3.6.1}$$

Then, the node and edge features along are passed sequentially into L layers message passing networks with the graph structure $\mathcal{G}$:

$$\boldsymbol{h}_v^{(\ell)} = \text{MLP}\left(\boldsymbol{h}_v^{(\ell-1)} + \sum_{u \in N_{\mathcal{G}}(v)} \sigma(\boldsymbol{h}_u^{(\ell-1)} + \boldsymbol{h}_{e_{uv}})\right), \quad \ell = 1 \ldots L, \tag{3.6.2}$$

where $N_{\mathcal{G}}(v)$ denotes the first neighbors in the graph $\mathcal{G}$ and σ is the activation function. After L message passing layers, we use the final hidden representation $h^{(L)}$ as the node representations. Then for each bond, the corresponding node features are aggregated along with the edge feature to be fed into a neural network to compute the value of the dynamic f_θ:

$$\frac{\partial d_{uv}}{\partial t} = \text{NN}(\boldsymbol{h}_u, \boldsymbol{h}_v, \boldsymbol{h}_{e_{uv}}, t). \tag{3.6.3}$$

3.6.3.2. Energy-based Tilting Model. The ETM is implemented with SchNet. It takes both the graph and conformation information as input and output a scalar to indicate the energy level. Let the atoms are described by a tuple of features $\mathbf{X}^l = (\mathbf{x}_1^l, \ldots, \mathbf{x}_n^l)$, where n denote the number of atoms and l denote the layer. Then given the positions $\boldsymbol{R}$, the node embeddings are updated by the convolution with all surrounding atoms:

$$\mathbf{x}_i^{l+1} = \left(X^l * W^l\right)_i = \sum_{j=0}^{n_{\text{atoms}}} \mathbf{x}_j^l \circ W^l \left(\mathbf{r}_j - \mathbf{r}_i\right), \tag{3.6.4}$$

where "o" represents the element-wise multiplication. It is straightforward that the above function enables to include translational and rotational invariance by computing pairwise distances instead of using relative positions. After L convolutional layers, we perform a sum-pooling operator over the node embeddings to calculate the global embedding for the whole molecular structure. Then the global embedding is fed into a feedforward network to compute the scalar of the energy level.

3.6.4. Two-stage Dynamic System for Sampling

3.6.5. Implementation Details

Our model is implemented in PyTorch (Paszke et al., 2017). The MPNN in CGCF is implemented with 3 layers, and the embedding dimension is set as 128. And the SchNet

Algorithm 1 Sampling Procedure of the Proposed Method

```
Input: molecular graph $\mathcal{G}$, CGCF model with parameter $\theta$, ETM with parameter $\phi$, the number of optimization steps for $p(\boldsymbol{R}|\boldsymbol{d},\mathcal{G})$ $M$ and its step size $r$, the number of MCMC steps for $E_{\theta,\phi}$ $N$ and its step size $\epsilon$
Output: molecular conformation $\boldsymbol{R}$
Sample $\boldsymbol{d}(t_0) \sim \mathcal{N}(0,\mathcal{I})$
$\boldsymbol{d} = F_\theta(\boldsymbol{d}(t_0),\mathcal{G})$
for $m = 1,...,M$ do
    $\boldsymbol{R}_m = \boldsymbol{R}_{m-1} + r\nabla_{\boldsymbol{R}} \log p(\boldsymbol{R}|\boldsymbol{d},\mathcal{G})$
end for
for $n = 1,...,N$ do
    $\boldsymbol{R}_n = \boldsymbol{R}_{n-1} - \frac{\epsilon}{2}\nabla_{\boldsymbol{R}} E_{\theta,\phi}(\boldsymbol{R}|\mathcal{G}) + \sqrt{\epsilon}\omega, \omega \sim \mathcal{N}(0,\mathcal{I}),$
end for
```

in ETM is implemented with 6 layers with the embedding dimension set as 128. We train our CGCF with a batch size of 128 and a learning rate of 0.001 until convergence. After obtaining the CGCF, we train the ETM with a batch size of 384 and a learning rate of 0.001 until convergence. For all experimental settings, we use Adam (Kingma and Ba, 2014) to optimize our model.

3.6.6. Detailed Derivations of Energy-based Model

Here we present the detailed derivations of the training objective function of Energy-based Tilting Model (ETM) in Eq. 3.3.6:

$$\begin{aligned}
\mathcal{L}_{\text{nce}}(\boldsymbol{R},\mathcal{G};\phi) &= -\mathbb{E}_{p_{\text{data}}}\Big[\log \frac{p_{\theta,\phi}(\boldsymbol{R}|\mathcal{G})}{p_{\theta,\phi}(\boldsymbol{R}|\mathcal{G}) + p_\theta(\boldsymbol{R}|\mathcal{G})}\Big] - \mathbb{E}_{p_\theta}\Big[\log \frac{p_\theta(\boldsymbol{R}|\mathcal{G})}{p_{\theta,\phi}(\boldsymbol{R}|\mathcal{G}) + p_\theta(\boldsymbol{R}|\mathcal{G})}\Big] \\
&= -\mathbb{E}_{p_{\text{data}}}\Big[\log \frac{p_\theta(\boldsymbol{R}|\mathcal{G})\exp(-E_\phi(\boldsymbol{R},\mathcal{G}))}{p_\theta(\boldsymbol{R}|\mathcal{G})\exp(-E_\phi(\boldsymbol{R},\mathcal{G})) + p_\theta(\boldsymbol{R}|\mathcal{G})}\Big] \\
&\quad - \mathbb{E}_{p_\theta}\Big[\log \frac{p_\theta(\boldsymbol{R}|\mathcal{G})}{p_\theta(\boldsymbol{R}|\mathcal{G})\exp(-E_\phi(\boldsymbol{R},\mathcal{G})) + p_\theta(\boldsymbol{R}|\mathcal{G})}\Big] \\
&= -\mathbb{E}_{p_{\text{data}}}\Big[\log \frac{1}{1+\exp(E_\phi(\boldsymbol{R},\mathcal{G}))}\Big] - \mathbb{E}_{p_\theta}\Big[\log \frac{1}{1+\exp(-E_\phi(\boldsymbol{R},\mathcal{G}))}\Big].
\end{aligned} \tag{3.6.5}$$

3.6.7. More Generated Samples

We present more visualizations of generated 3D structures in Fig. 3.3, which are generated from our model (CGCF + ETM) learned on both GEOM-QM9 and GEOM-Drugs datasets. The visualizations demonstrate that our proposed framework holds the high capacity to model the chemical structures in the 3D coordinates.

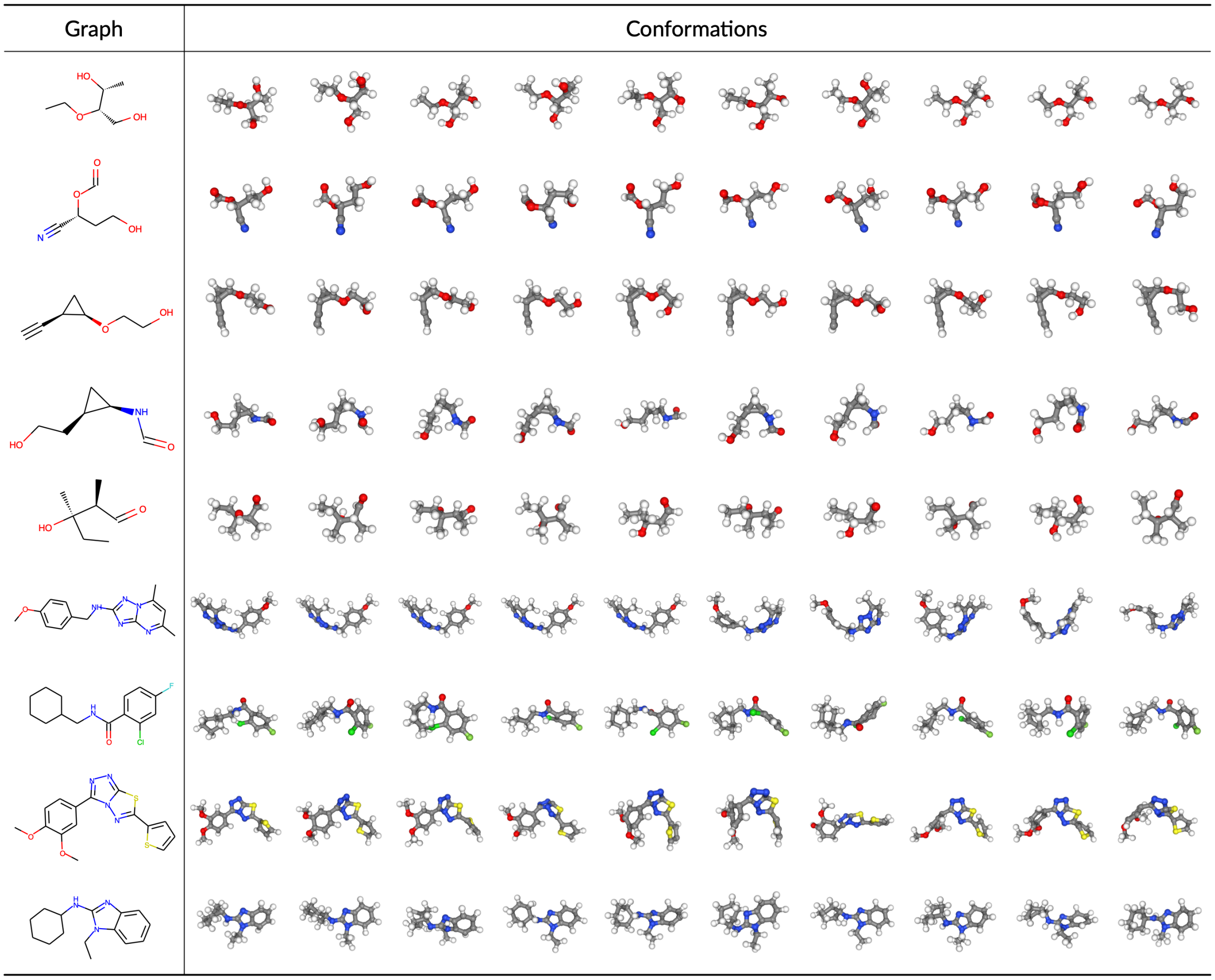

Fig. 3.3. Visualizations of generated graphs from our proposed method. In each row, we show multiple generated conformations for one molecular graph. For the top 5 rows, the graphs are chosen from the small molecules in GEOM-QM9 test dataset; and for the bottom 4 rows, graphs are chosen from the larger molecules in GEOM-Drugs test dataset. C, O, H, S and CI are colored gray, red, white, yellow and green respectively.

3.6.8. More Results of Coverage Score

We give more results of the coverage (COV) score with different threshold δ in Fig. 3.4. As shown in the figure, our proposed method can consistently outperform the previous state-of-the-art baselines CVGAE and GraphDG, which demonstrate the effectiveness of our model.

3.6.9. Implementation for MMFF

In this section, we give a more precise description of the MMFF Force Field implementation in the RDKit toolkit (Riniker and Landrum, 2015).

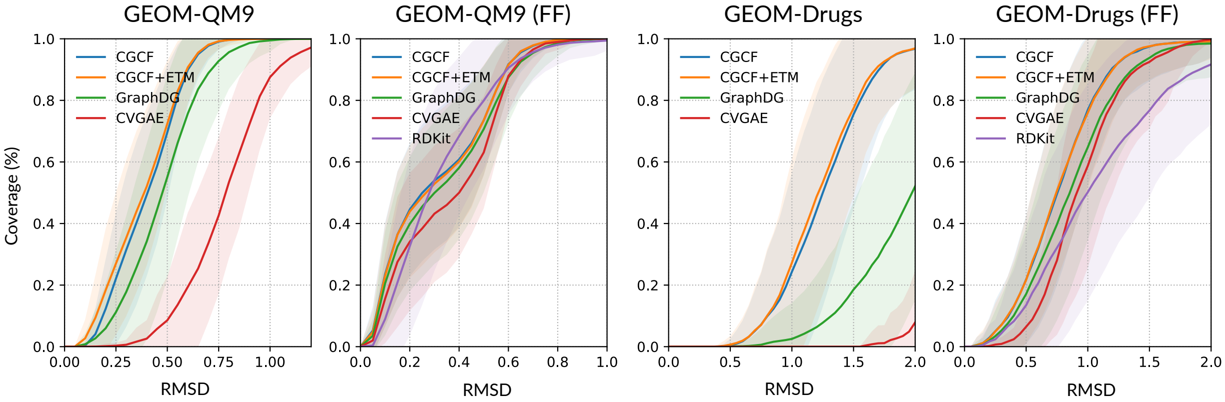

Fig. 3.4. Curves of the averaged coverage score with different RMSD thresholds on GEOM-QM9 (left two) and GEOM-Drugs (right two) datasets. The first and third curves are results of only the generative models, while the other two are results when further optimized with rule-based force fields.

In MMFF, the energy expression is constituted by seven terms: bond stretching, angle bending, stretch-bend, out-of-plane bending, torsional, van der Waals and electrostatic. The detailed functional form of individual terms can be found in the original literature (Halgren, 1996a). To build the force field for a given molecular system, the first step is to assign correct types to each atom. At the second step, atom-centered partial charges are computed according to the MMFF charge model (Halgren, 1996b). Then, all bonded and non-bonded interactions in the molecular system under study, depending on its structure and connectivity, are loaded into the energy expression. Optionally, external restraining terms can be added to the MMFF energy expression, with the purpose of constraining selected internal coordinates during geometry optimizations. Once all bonded and non-bonded interactions, plus optional restraints, have been loaded into the MMFF energy expression, potential gradients of the system under study can be computed to minimize the energy.

3.6.10. More Evaluations for Conformation Generation

Junk Rate. The COV and MAT score in Section 3.4.2 do not appear to explicitly measure the generated false samples. Here we additionally define **Junk** rate measurement. Intuitively, JUNK measures the fraction of generated conformations that are far away from all the conformations in the reference set. For each conformation in the generated set, it will be marked as a false sample if its RMSD to all the conformations of reference set are above a given threshold δ:

$$\mathrm{JUNK}(\mathbb{S}_g(\mathcal{G}), \mathbb{S}_r(\mathcal{G})) = \frac{1}{|\mathbb{S}_g|}\left|\left\{\boldsymbol{R} \in \mathbb{S}_g \middle| \mathrm{RMSD}(\boldsymbol{R}, \boldsymbol{R}') > \delta, \forall \boldsymbol{R}' \in \mathbb{S}_r\right\}\right|, \tag{3.6.6}$$

Typically, a lower JUNK rate means better generated quality. The results are shown in Tab. 3.4. As shown in the table, our CGCF model can already outperform the existing state-of-the-art baselines with an obvious margin. The results are further improved when combined with ETM to explicitly incorporate the long-range correlations.

Tableau 3.4. Comparison of different methods on the JUNK scores. Top 4 rows: deep generative models for molecular conformation generation. Bottom 5 rows: different methods that involve an additional rule-based force field to further optimize the generated structures.

Dataset	GEOM-QM9		GEOM-Drugs	
Metric	JUNK* (%)		JUNK* (%)	
	Mean	Median	Mean	Median
CVGAE	71.59	100.00	100.00	100.00
GraphDG	61.25	66.26	97.83	100.00
CGCF	55.24	57.24	77.82	90.00
CGCF + ETM	**52.15**	**54.23**	**75.81**	**88.64**
RDKit	17.07	5.90	45.51	45.94
CVGAE + FF	62.92	71.21	72.01	78.44
GraphDG + FF	45.53	46.35	55.50	61.54
CGCF + FF	43.01	46.69	37.48	36.63
CGCF + ETM + FF	**41.63**	**43.97**	**36.16**	**33.05**

[*] For the reported JUNK score, the threshold δ is set as 0.5Å for QM9 and 1.25Å for Drugs.

3.6.11. Distance Distribution Visualization

In Fig. 3.5, we plot the marginal distributions $p(d_{uv}|\mathcal{G})$ for all pairwise distances between C and O atoms of a molecular graph in the ISO17 test set. As shown in the figure, though primarily designed for 3D structure generation, our method can make much better estimation of the distances than GraphDG, which is the state-of-the-art model for molecular geometry prediction. As a representative element of the pairwise property between atoms, the inter-atomic distances demonstrate the capacity of our model to capture the inter-atomic interactions.

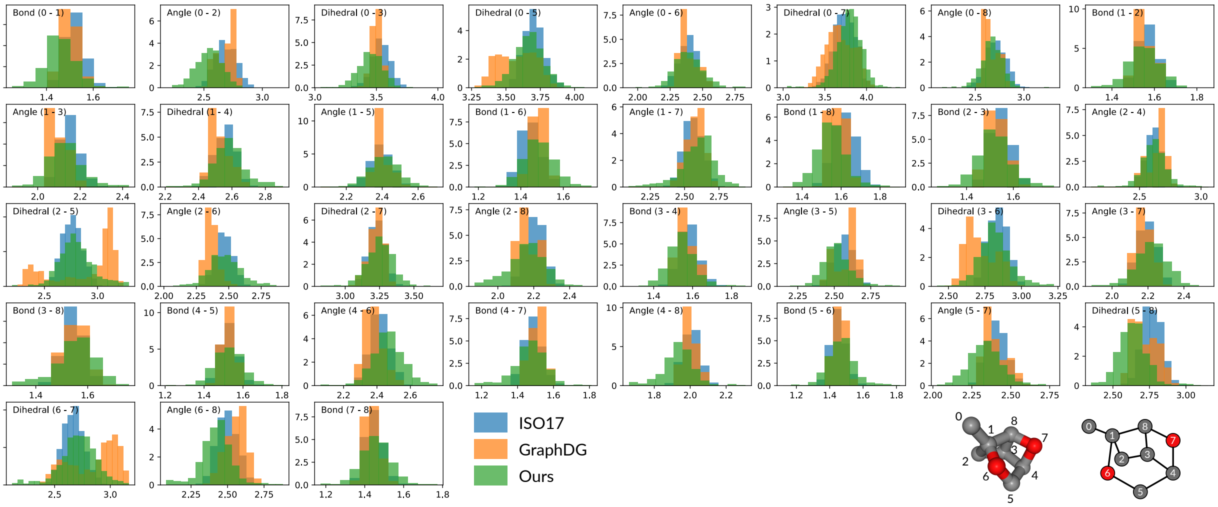

Fig. 3.5. Marginal distributions $p(d_{uv}|\mathcal{G})$ of ground-truth and generated conformations between C and O atoms given a molecular graph from the test set of ISO17. In each subplot, the annotation $(u - v)$ indicates the atoms connected by the corresponding bond d_{uv}. We concentrate on the heavy atoms (C and O) and omit the H atoms for clarity.

Chapter 4

Prologue to Second Article

4.1. Article Details

GeoDiff: a Geometric Diffusion Model for Molecular Conformation Generation. Minkai Xu, Lantao Yu, Yang Song, Chence Shi, Stefano Ermon, Jian Tang. *10th International Conference on Learning Representations, 2022.*

4.2. Personal Contribution

I came up with the idea of introducing denoising diffusion generative models for general 3-dimensional geometry generation, and specifically tackle the molecular conformation generation problem for experimental part. I received lots of important suggestions and discussions from Lantao Yu, Yang Song, Chence Shi, Stefano Ermon, and Jian Tang. I wrote up the code base for the proposed model, further improved model to achieve better performances, and run experiments on widely-adopted benchmarks. I wrote the majority of the paper, and Chence helped to render several figures in the paper. Jian Tang significantly contributed to improving the writing, and the others have also helped polishing the paper.

Chapter 5

Geometric Denoising Diffusion Generative Models

5.1. Introduction

Graph representation learning has achieved huge success for molecule modeling in various tasks ranging from property prediction (Gilmer et al., 2017; Duvenaud et al., 2015) to molecule generation (Jin et al., 2018; Shi et al., 2020b), where typically a molecule is represented as an atom-bond graph. Despite its effectiveness in various applications, a more intrinsic and informative representation for molecules is the 3D *geometry*, also known as *conformation*, where atoms are represented as their Cartesian coordinates. The 3D structures determine the biological and physical properties of molecules and hence play a key role in many applications such as computational drug and material design (Thomas et al., 2018; Gebauer et al., 2021; Jing et al., 2021; Batzner et al., 2021). Unfortunately, how to predict stable molecular conformation remains a challenging problem. Traditional methods based on molecular dynamics (MD) or Markov chain Monte Carlo (MCMC) are very computationally expensive, especially for large molecules (Hawkins, 2017).

Recently, significant progress has been made with machine learning approaches, especially with deep generative models. For example, Simm and Hernández-Lobato (2020); Xu et al. (2021b) studied predicting atomic distances with variational autoencoders (VAEs) (Kingma and Welling, 2013) and flow-based models (Dinh et al., 2017) respectively. Shi et al. (2021) proposed to use denoising score matching (Song and Ermon, 2019, 2020) to estimate the gradient fields over atomic distances, through which the gradient fields over atomic coordinates can be calculated. Ganea et al. (2021) studied generating conformations by predicting both bond lengths and angles. As molecular conformations are roto-translational invariant, these approaches circumvent directly modeling atomic coordinates by leveraging intermediate geometric variables such as atomic distances, bond and torsion angles, which are roto-translational invariant. As a result, they are able to achieve very compelling performance. However, as all these approaches seek to indirectly model the intermediate geometric variables, they have inherent limitations in either training or inference process (see Sec. 5.2 for a detailed description). Therefore, an ideal solution would still be directly modeling the atomic coordinates and at the same time taking the roto-translational invariance property into account.

In this paper, we propose such a solution called GeoDiff, a principled probabilistic framework based on denoising diffusion models (Sohl-Dickstein et al., 2015). Our approach is inspired by the *diffusion process* in nonequilibrium thermodynamics (De Groot and Mazur, 2013). We view atoms as particles in a thermodynamic system, which gradually diffuse from the original states to a noisy distribution in contact with a heat bath. At each time step, stochastic noises are added to the atomic positions. Our high-level idea is learning to reverse the diffusion process, which recovers the target geometric distribution from the noisy distribution. In particular, inspired by recent progress of denoising diffusion models on image generation (Ho et al., 2020; Song et al., 2020a), we view the noisy geometries at different timesteps as latent variables, and formulate both the forward diffusion and reverse denoising process as Markov chains. Our goal is to learn the transition kernels such that the reverse process can recover realistic conformations from the chaotic positions sampled from a noise distribution. However, extending existing methods to geometric generation is highly non-trivial: a direct application of diffusion models on the conformation generation task lead to poor generation quality. As mentioned above, molecular conformations are roto-translational invariant, *i.e.*, the estimated (conditional) likelihood should be unaffected by translational and rotational transformations (Köhler et al., 2020). To this end, we first theoretically show that a Markov process starting from an roto-translational *invariant* prior distribution and evolving with roto-translational *equivariant* Markov kernels can induce an roto-translational *invariant* density function. We further provide practical parameterization to define a roto-translational *invariant* prior distribution and a Markov kernel imposing the equivariance constraints. In addition, we derive a weighted variational lower bound of the conditional likelihood of molecular conformations, which also enjoys the roto-translational invariance and can be efficiently optimized.

A unique strength of GeoDiff is that it directly acts on the atomic coordinates and entirely bypasses the usage of intermediate elements for both training and inference. This general formulation enjoys several crucial advantages. First, the model can be naturally trained end-to-end without involving any sophisticated techniques like bilevel programming (Xu et al., 2021b), which benefits from small optimization variances. Besides, instead of solving geometries from bond lengths or angles, the one-stage sampling fashion avoids accumulating any intermediate error, and therefore leads to more accurate predicted structures. Moreover, GeoDiff enjoys a high model capacity to approximate the complex distribution of conformations. Thus, the model can better estimate the highly multi-modal distribution and generate structures with high quality and diversity.

We conduct comprehensive experiments on multiple benchmarks, including conformation generation and property prediction tasks. Numerical results show that GeoDiff consistently outperforms existing state-of-the-art machine learning approaches, and by a large margin on the more challenging large molecules. The significantly superior performance demonstrate the high capacity to model the complex distribution of molecular conformations and generate both diverse and accurate molecules.

5.2. Related Work

Recently, various deep generative models have been proposed for conformation generation. Among them, CVGAE (Mansimov et al., 2019) first proposed a VAE model to directly generate 3D atomic coordinates, which fails to preserve the roto-translation equivariance property of conformations and suffers from poor performance. To address this problem, the majority of subsequent models are based on intermediate geometric elements such as atomic distances and torsion angles. A favorable property of these elements is the roto-translational invariance, (*e.g.* atomic distances does not change when rotating the molecule), which has been shown to be an important inductive bias for molecular geometry modeling (Köhler et al., 2020). However, such a decomposition suffers from several drawbacks for either training or sampling. For example, GraphDG (Simm and Hernández-Lobato, 2020) and CGCF (Xu et al., 2021a) proposed to predict the interatomic distance matrix by VAE and Flow respectively, and then solve the geometry through the Distance Geometry (DG) technique (Liberti et al., 2014), which searches reasonable coordinates that matches with the predicted distances. ConfVAE further improves this pipeline by designing an end-to-end framework via bilevel optimization (Xu et al., 2021b). However, all these approaches suffer from the accumulated error problem, meaning that the noise in the predicted distances will misguide the coordinate searching process and lead to inaccurate or even erroneous structures. To overcome this problem, ConfGF (Shi et al., 2021; Luo et al., 2021) proposed to learn the gradient of the log-likelihood *w.r.t* coordinates. However, in practice the model is still aided by intermediate geometric elements, in that it first estimates the gradient *w.r.t* interatomic distances via denoising score matching (DSM) (Song and Ermon, 2019, 2020), and then derives the gradient of coordinates using the chain rule. The problem is, by learning the distance gradient via DSM, the model is fed with perturbed distance matrices, which may violate the triangular inequality or even contain negative values. As a consequence, the model is actually learned over invalid distance matrices but tested with valid ones calculated from coordinates, making it suffer from serious out-of-distribution (Hendrycks and Gimpel, 2016) problem. Most recently, another concurrent work (Ganea et al., 2021) proposed a highly *systematic* (rule-based) pipeline named GeoMol, which learns to predict a minimal set of geometric quantities (*i.e.* length and angles) and then reconstruct the local and global structures of the conformation in a sophisticated procedure. Besides, there has also been efforts to use reinforcement learning for conformation search Gogineni et al. (2020). Nevertheless, this method relies on rigid rotor approximation and can only model the torsion angles, and thus fundamentally differs from other approaches.

5.3. Preliminaries

5.3.1. Notations and Problem Definition

Notations. In this paper each molecule with n atoms is represented as an undirected graph $\mathcal{G} = \langle \mathcal{V}, \mathcal{E} \rangle$, where $\mathcal{V} = \{v_i\}_{i=1}^n$ is the set of vertices representing atoms and $\mathcal{E} = \{e_{ij} \mid (i,j) \subseteq |\mathcal{V}| \times |\mathcal{V}|\}$ is the set of edges representing inter-atomic bonds. Each node $v_i \in \mathcal{V}$ describes the atomic attributes, *e.g.*, the element type. Each edge $e_{ij} \in \mathcal{E}$ describes the corresponding connection between v_i and v_j, and is labeled with its chemical type. In addition, we also assign the unconnected edges with a *virtual* type. For the geometry, each atom in $\mathcal{V}$ is embedded by a coordinate vector $\boldsymbol{c} \in \mathbb{R}^3$ into the 3-dimensional space, and the full set of positions (*i.e.*, the conformation) can be represented as a matrix $\mathcal{C} = [\boldsymbol{c}_1, \boldsymbol{c}_2, \cdots, \boldsymbol{c}_n] \in \mathbb{R}^{n \times 3}$.

Problem Definition. The task of *molecular conformation generation* is a conditional generative problem, where we are interested in generating stable conformations for a provided graph $\mathcal{G}$. Given multiple graphs $\mathcal{G}$, and for each $\mathcal{G}$ given its conformations $\mathcal{C}$ as *i.i.d* samples from an underlying Boltzmann distribution (Noé et al., 2019), our goal is learning a generative model $p_\theta(\mathcal{C}|\mathcal{G})$, which is easy to draw samples from, to approximate the Boltzmann function.

5.3.2. Equivariance

Equivariance is ubiquitous in machine learning for atomic systems, *e.g.*, the vectors of atomic dipoles or forces should rotate accordingly *w.r.t.* the conformation coordinates (Thomas et al., 2018; Weiler et al., 2018; Fuchs et al., 2020; Miller et al., 2020; Simm et al., 2021; Batzner et al., 2021). It has shown effectiveness to integrate such inductive bias into model parameterization for modeling 3D geometry, which is critical for the generalization capacity (Köhler et al., 2020; Satorras et al., 2021a). Formally, a function $\mathcal{F} : \mathcal{X} \to \mathcal{Y}$ is equivariant *w.r.t* a group G if:

$$\mathcal{F} \circ T_g(x) = S_g \circ \mathcal{F}(x), \tag{5.3.1}$$

where T_g and S_g are transformations for an element $g \in G$, acting on the vector spaces $\mathcal{X}$ and $\mathcal{Y}$, respectively. In this work, we consider the SE(3) group, *i.e.*, the group of rotation,

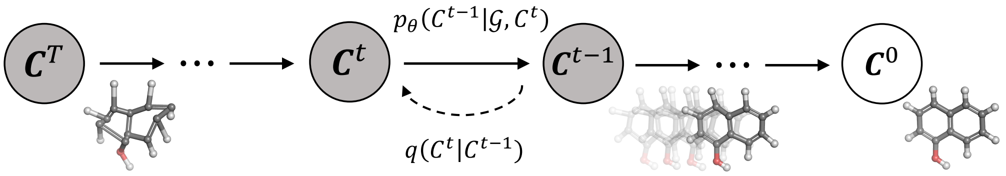

Fig. 5.1. Illustration of the diffusion and reverse process of GeoDiff. For diffusion process, noise from fixed posterior distributions $q(\mathcal{C}^t|\mathcal{C}^{t-1})$ is gradually added until the conformation is destroyed. Symmetrically, for generative process, an initial state $\mathcal{C}^T$ is sampled from standard Gaussian distribution, and the conformation is progressively refined via the Markov kernels $p_\theta(\mathcal{C}^{t-1}|\mathcal{G},\mathcal{C}^t)$.

translation in 3D space. This requires the estimated likelihood unaffected with translational and rotational transformations, and we will elaborate on how our method satisfy this property in Sec. 5.4.

5.4. GeoDiff Method

In this section, we elaborate on the proposed equivariant diffusion framework. We first present a high level description of our 3D diffusion formulation in Sec. 5.4.1, based on recent progress of denoising diffusion models (Sohl-Dickstein et al., 2015; Ho et al., 2020). Then we emphasize several non-trivial challenges of building diffusion models for geometry generation scenario, and show how we technically tackle these issues. Specifically, in Sec. 5.4.2, we present how we parameterize $p_\theta(\mathcal{C}|\mathcal{G})$ so that the conditional likelihood is roto-translational invariant, and in Sec. 5.4.3, we introduce our surgery of the training objective to make the optimization also invariant of translation and rotation. Finally, we briefly show how to draw samples from our model in Sec. 5.4.4.

5.4.1. Formulation

Let $\mathcal{C}^0$ denotes the ground truth conformations and let $\mathcal{C}^t$ for $t = 1, \cdots, T$ be a sequence of latent variables with the same dimension, where t is the index for diffusion steps. Then a diffusion probabilistic model (Sohl-Dickstein et al., 2015) can be described as a latent variable model with two processes: the forward *diffusion* process, and the reverse *generative* process. Intuitively, the *diffusion process* progressively injects small noises to the data $\mathcal{C}^0$, while the *generative process* learns to revert the diffusion process by gradually eliminating the noise to recover the ground truth. We provide a high-level schematic of the processes in Fig. 5.1.

Diffusion process. Following the physical insight, we model the particles $\mathcal{C}$ as an evolving thermodynamic system. With time going by, the equilibrium conformation $\mathcal{C}^0$ will gradually diffuse to the next chaotic states $\mathcal{C}^t$, and finally converge into a white noise distribution after T iterations. Different from typical latent variable models, in diffusion model this *forward process* is defined as a fixed (rather than trainable) posterior distribution $q(\mathcal{C}^{1:T}|\mathcal{C}^0)$. Specifically, we define it as a Markov chain according to a fixed variance schedule $\beta_1, \ldots ,\beta_T$:

$$q(\mathcal{C}^{1:T}|\mathcal{C}^0) = \prod_{t=1}^{T} q(\mathcal{C}^t|\mathcal{C}^{t-1}), \quad q(\mathcal{C}^t|\mathcal{C}^{t-1}) = \mathcal{N}(\mathcal{C}^t; \sqrt{1-\beta_t}\mathcal{C}^{t-1},\beta_t I). \tag{5.4.1}$$

Note that, in this work we do not impose specific (invariance) requirement upon the diffusion process, as long as it can efficiently draw noisy samples for training the generative process $p_\theta(\mathcal{C}^0)$.

Let $\alpha_t = 1-\beta_t$ and $\bar{\alpha}_t = \prod_{s=1}^{t} \alpha_s$, a special property of the forward process is that $q(\mathcal{C}^t|\mathcal{C}^0)$ of arbitrary timestep t can be calculated in closed form $q(\mathcal{C}^t|\mathcal{C}^0) = \mathcal{N}(\mathcal{C}^t; \sqrt{\bar{\alpha}_t}\mathcal{C}^0,(1-\bar{\alpha}_t)I)$[2]. This indicates with sufficiently large T, the whole forward process will convert $\mathcal{C}^0$ to whitened isotropic Gaussian, and thus it is natural to set $p(\mathcal{C}^T)$ as a standard Gaussian distribution.

Reverse Process. Our goal is learning to recover conformations $\mathcal{C}^0$ from the white noise $\mathcal{C}^T$, given specified molecular graphs $\mathcal{G}$. We consider this generative procedure as a reverse dynamics of the above diffusion process, starting from the noisy particles $\mathcal{C}^T \sim p(\mathcal{C}^T)$. We formulate this reverse dynamics as a conditional Markov chain with learnable transitions:

$$p_\theta(\mathcal{C}^{0:T-1}|\mathcal{G}, \mathcal{C}^T) = \prod_{t=1}^{T} p_\theta(\mathcal{C}^{t-1}|\mathcal{G},\mathcal{C}^t), \quad p_\theta(\mathcal{C}^{t-1}|\mathcal{G},\mathcal{C}^t) = \mathcal{N}(\mathcal{C}^{t-1}; \mu_\theta(\mathcal{G}, \mathcal{C}^t,t), \sigma_t^2 I). \tag{5.4.2}$$

Herein μ_θ are parameterized neural networks to estimate the means, and σ_t can be any user-defined variance. The initial distribution $p(\mathcal{C}^T)$ is set as a standard Gaussian. Given a graph $\mathcal{G}$, its 3D structure is generated by first drawing chaotic particles $\mathcal{C}^T$ from $p(\mathcal{C}^T)$, and then iteratively refined through the reverse Markov kernels $p_\theta(\mathcal{C}^{t-1}|\mathcal{G},\mathcal{C}^t)$.

Having formulated the *reverse* dynamics, the marginal likelihood can be calculated by $p_\theta(\mathcal{C}^0|\mathcal{G}) = \int p(\mathcal{C}^T)p_\theta(\mathcal{C}^{0:T-1}|\mathcal{G},\mathcal{C}^T)\mathrm{d}\boldsymbol{\mathcal{C}}^{1:T}$. Herein a non-trivial problem is that the likelihood should be invariant *w.r.t* translation and rotation, which has proved to be a critical inductive bias for 3D object generation (Köhler et al., 2020; Satorras et al., 2021a). In the following subsections, we will elaborate on how we parameterize the Markov kernels $p_\theta(\mathcal{C}^{t-1}|\mathcal{G},\mathcal{C}^t)$ to achieve this desired property, and also how to maximize this likelihood by taking the invariance into account.

5.4.2. Equivariant Reverse Generative Process

Instead of directly leveraging existing methods, we consider building the density $p_\theta(\mathcal{C}^0)$ that is invariant to rotation and translation transformations. Intuitively, this requires the

[2] Detailed derivations are provided in the Appendix 5.7.1.

likelihood to be unaffected by translations and rotations. Formally, let T_g be some roto-translational transformations of a group element $g \in \mathrm{SE}(3)$, then we have the following statement:

Proposition 1. *Let $p(x_T)$ be an SE(3)-invariant density function, i.e., $p(x_T) = p(T_g(x_T))$. If Markov transitions $p(x_{t-1}|x_t)$ are SE(3)-equivariant, i.e., $p(x_{t-1}|x_t) = p(T_g(x_{t-1})|T_g(x_t))$, then we have that the density $p_\theta(x_0) = \int p(x_T)p_\theta(x_{0:T-1}|x_T)\mathrm{d}\boldsymbol{x}_{1:T}$ is also SE(3)-invariant.*

This proposition indicates that the dynamics starting from an invariant standard density along an equivariant Gaussian Markov kernel can result in an invariant density. Now we provide a practical implementation of GeoDiff based on the recent *denoising diffusion* framework (Ho et al., 2020).

Invariant Initial Density $p(\mathcal{C}^T)$. We first introduce the invariant distribution $p(\mathcal{C}^T)$, which will also be employed in the equivariant Markov chain. We borrow the idea from Köhler et al. (2020) to consider systems with zero center of mass (CoM), termed CoM-free systems. We define $p(\mathcal{C}^T)$ as a "CoM-free standard density" $\hat{\rho}(\mathcal{C})$, built upon an isotropic normal density $\rho(\mathcal{C})$: for evaluating the likelihood $\hat{\rho}(\mathcal{C})$ we can firstly translate $\mathcal{C}$ to zero CoM and then calculate $\rho(\mathcal{C})$, and for sampling from $\hat{\rho}(\mathcal{C})$ we can first sample from $\rho(\mathcal{C})$ and then move the CoM to zero.

We provide a formal theoretical analysis of $\hat{\rho}(\mathcal{C})$ in Appendix 5.7.1. Intuitively, the isotropic Gaussian is manifestly invariant to rotations around the zero CoM. And by considering CoM-free system, moving the particles to zero CoM can always ensure the translational invariance. Consequently, $\hat{\rho}(\mathcal{C})$ is constructed as a roto-transitional invariant density.

Equivariant Markov Kernels $p(\mathcal{C}^{t-1}|\mathcal{G}, \mathcal{C}^t)$. Similar to the prior density, we also consider equipping all intermediate structures $\mathcal{C}^t$ as CoM-free systems. Specifically, given mean $\mu_\theta(\mathcal{G}, \mathcal{C}^t, t)$ and variance σ_t, the likelihood of $\mathcal{C}^{t-1}$ will be calculated by $\hat{\rho}(\frac{\mathcal{C}^{t-1}-\mu_\theta(\mathcal{G},\mathcal{C}^t,t)}{\sigma_t})$. The CoM-free Gaussian ensures the translation invariance in the Markov kernels. Consequently, to achieve the equivariant property defined in Proposition 1, we focus on the rotation equivariance.

Then in general, the key requirement is to ensure the means $\mu_\theta(\mathcal{G}, \mathcal{C}^t, t)$ to be roto-translation equivariant *w.r.t* $\mathcal{C}^t$. Following Ho et al. (2020), we consider the following parameterization of μ_θ:

$$\mu_\theta(\mathcal{C}^t, t) = \frac{1}{\sqrt{\alpha_t}}\left(\mathcal{C}^t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(\mathcal{G}, \mathcal{C}^t, t)\right), \tag{5.4.3}$$

where ϵ_θ are neural networks with trainable parameters θ. Intuitively, the model ϵ_θ learns to predict the noise necessary to decorrupt the conformations. This is analogous to the physical force fields (Schütt et al., 2017; Zhang et al., 2018; Hu et al., 2021; Shuaibi et al., 2021), which also gradually push particles towards convergence around the equilibrium states.

Now the problem is transformed to constructing ϵ_θ to be roto-translational equivariant. We draw inspirations from recent equivariant networks (Thomas et al., 2018; Satorras et al., 2021b) to design an equivariant convolutional layer, named graph field network (GFN). In the l-th layer, GFN takes node embeddings $\mathbf{h}^l \in \mathbb{R}^{n\times b}$ (b denotes the feature dimension) and corresponding coordinate embeddings $\mathbf{x}^l \in \mathbb{R}^{n\times 3}$ as inputs, and outputs $\mathbf{h}^{l+1}$ and $\mathbf{x}^{l+1}$ as follows:

$$\mathbf{m}_{ij} = \Phi_m\left(\mathbf{h}_i^l, \mathbf{h}_j^l, \|\mathbf{x}_i^l - \mathbf{x}_j^l\|^2, e_{ij}; \theta_m\right) \tag{5.4.4}$$

$$\mathbf{h}_i^{l+1} = \Phi_h\Big(\mathbf{h}_i^l, \sum_{j\in\mathcal{N}(i)} \mathbf{m}_{ij}; \theta_h\Big) \tag{5.4.5}$$

$$\mathbf{x}_i^{l+1} = \sum_{j\in\mathcal{N}(i)} \frac{1}{d_{ij}} \left(\mathbf{c}_i - \mathbf{c}_j\right) \Phi_x\left(\mathbf{m}_{ij}; \theta_x\right) \tag{5.4.6}$$

where Φ are feed-forward networks and d_{ij} denotes interatomic distances. $\mathcal{N}(i)$ denotes the neighborhood of i^{th} node, including both connected atoms and other ones within a radius threshold τ, which enables the model to explicitly capture long-range interactions and support molecular graphs with disconnected components. Initial embeddings $\mathbf{h}^0$ are combinations of atom and timestep embeddings, and $\mathbf{x}^0$ are atomic coordinates. The main difference between proposed GFN and other GNNs lies in equation 5.4.6, where $\mathbf{x}$ is updated as a combination of radial directions weighted by $\Phi_x : \mathbb{R}^b \to \mathbb{R}$. Such vector field $\mathbf{x}^L$ enjoys the roto-translation equivariance property. Formally, we have:

Proposition 2. *Parameterizing $\epsilon_\theta(\mathcal{G},\mathcal{C},t)$ as a composition of L GFN layers, and take the $\mathbf{x}^L$ after L updates as the output. Then the noise vector field ϵ_θ is SE(3) equivariant w.r.t the 3D system $\mathcal{C}$.*

Intuitively, given $\mathbf{h}^l$ already invariant and $\mathbf{x}^l$ equivariant, the message embedding $\mathbf{m}$ will also be invariant since it only depends on invariant features. Since $\mathbf{x}$ is updated with the relative differences $\mathbf{c}_i - \mathbf{c}_j$ weighted by invariant features, it will be translation-invariant and rotation-equivariant. Then inductively, composing ϵ_θ with L GFN layers enables equivariance with $\mathcal{C}^t$. We provide the formal proof of equivariance properties in Appendix 5.7.1.

5.4.3. Improved Training Objective

Having formulated the generative process and the model parameterization, now we consider the practical training objective for the reverse dynamics. Since directly optimizing the exact log-likelihood is intractable, we instead maximize the usual variational lower bound (ELBO)[3]:

$$\begin{aligned}\mathbb{E}\left[\log p_\theta(\mathcal{C}^0|\mathcal{G})\right] &= \mathbb{E}\Big[\log \mathbb{E}_{q(\mathcal{C}^{1:T}|\mathcal{C}^0)} \frac{p_\theta(\mathcal{C}^{0:T}|\mathcal{G})}{q(\mathcal{C}^{1:T}|\mathcal{C}^0)}\Big] \\ &\geq -\mathbb{E}_q\Big[\sum_{t=1}^{T} D_{\mathrm{KL}}(q(\mathcal{C}^{t-1}|\mathcal{C}^t,\mathcal{C}^0)\|p_\theta(\mathcal{C}^{t-1}|\mathcal{C}^t,\mathcal{G}))\Big] := -\mathcal{L}_{\mathrm{ELBO}}\end{aligned} \tag{5.4.7}$$

[3] The detailed derivations and full proofs are provided in Appendix 5.7.1.

where $q(\mathcal{C}^{t-1}|\mathcal{C}^t,\mathcal{C}^0)$ is analytically tractable as $\mathcal{N}(\frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t}\mathcal{C}^0 + \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}\mathcal{C}^t, \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t)^3$. Most recently, Ho et al. (2020) showed that under the parameterization in equation 5.4.3, the ELBO of the diffusion model can be further simplified by calculating the KL divergences between Gaussians as weighted $\mathcal{L}_2$ distances between the means ϵ_θ and ϵ^3. Formally, we have:

Proposition 3. *(Ho et al., 2020) Under the parameterization in equation 5.4.3, we have:*

$$\mathcal{L}_{\text{ELBO}} = \sum_{t=1}^{T} \gamma_t \mathbb{E}_{\{\mathcal{C}^0,\mathcal{G}\}\sim q(\mathcal{C}^0,\mathcal{G}),\epsilon\sim\mathcal{N}(0,I)} \left[\|\epsilon - \epsilon_\theta(\mathcal{G},\mathcal{C}^t,t)\|_2^2 \right] \tag{5.4.8}$$

where $\mathcal{C}^t = \sqrt{\bar{\alpha}_t}\mathcal{C}^0 + \sqrt{1-\bar{\alpha}_t}\epsilon$*. The weights* $\gamma_t = \frac{\beta_t}{2\alpha_t(1-\bar{\alpha}_{t-1})}$ *for* $t > 1$*, and* $\gamma_1 = \frac{1}{2\alpha_1}$*.*

The intuition of this objective is to independently sample chaotic conformations of different timesteps from $q(\mathcal{C}^{t-1}|\mathcal{C}^t,\mathcal{C}^0)$, and use ϵ_θ to model the noise vector ϵ. To yield a better empirical performance, Ho et al. (2020) suggests to set all weights γ_t as 1, which is in line with the the objectives of recent noise conditional score networks (Song and Ermon, 2019, 2020).

As ϵ_θ is designed to be equivariant, it is natural to require its supervision signal ϵ to be equivariant with $\mathcal{C}^t$. Note that once this is achieved, the ELBO will also become invariant. However, the ϵ in the forward diffusion process is not imposed with such equivariance, violating the above properties. Here we propose two approaches to obtain the modified noise vector $\hat{\epsilon}$, which, after replacing ϵ in the $\mathcal{L}_2$ distance calculation in equation 5.4.8, achieves the desired equivariance:

Alignment approach. Considering the fact that ϵ can be calculated by $\frac{\mathcal{C}^t - \sqrt{\bar{\alpha}_t}\mathcal{C}^0}{\sqrt{1-\bar{\alpha}_t}}$, we can first rotate and translate $\mathcal{C}^0$ to $\hat{\mathcal{C}}^0$ by aligning *w.r.t* $\mathcal{C}^t$, and then compute $\hat{\epsilon}$ as $\frac{\mathcal{C}^t - \sqrt{\bar{\alpha}_t}\hat{\mathcal{C}}^0}{\sqrt{1-\bar{\alpha}_t}}$. Since the aligned conformation $\hat{\mathcal{C}}^0$ is equivariant with $\mathcal{C}^t$, the processed $\hat{\epsilon}$ will also enjoy the equivariance. Specifically, the alignment is implemented by first translating $\mathcal{C}^0$ to the same CoM of $\mathcal{C}^t$ and then solve the optimal rotation matrix by Kabsch alignment algorithm (Kabsch, 1976).

Chain-rule approach. Another meaningful observation is that by reparameterizing the Gaussian distribution $q(\mathcal{C}^t|\mathcal{C}^0)$ as $\mathcal{C}^t = \sqrt{\bar{\alpha}_t}\mathcal{C}^0 + \sqrt{1-\bar{\alpha}_t}\epsilon$, ϵ can be viewed as a weighted score function $\sqrt{1-\bar{\alpha}_t}\nabla_{\mathcal{C}^t} q(\mathcal{C}^t|\mathcal{C}^0)$. Shi et al. (2021) recently shows that generally this score function $\nabla_{\mathcal{C}^t} q(\mathcal{C}^t|\cdot)$ can be designed to be equivariant by decomposing it into $\partial_{\mathcal{C}^t}\mathbf{d}^t\, \nabla_{\mathbf{d}^t} q(\mathcal{C}^t|\cdot)$ with the chain rule, where $\mathbf{d}^t$ can be any invariant features of the structures $\mathcal{C}^t$ such as the inter-atomic distances. We refer readers to Shi et al. (2021) for more details. The insight is that as gradient of invariant variables *w.r.t* equivariant variables, the partial derivative $\partial_{\mathcal{C}^t}\mathbf{d}^t$ will always be equivalent with $\mathcal{C}^t$. In this work, under the common assumption that $\mathbf{d}$ also follows a Gaussian distribution (Kingma and Welling, 2013), our practical implementation is to first approximately calculate $\nabla_{\mathbf{d}^t} q(\mathcal{C}^t|\mathcal{C}^0)$ as $\frac{\mathbf{d}^t-\sqrt{\bar{\alpha}_t}\mathbf{d}^0}{1-\bar{\alpha}_t}$, and then compute the modified noise vector $\hat{\epsilon}$ as $\sqrt{1-\bar{\alpha}_t}\,\partial_{\mathcal{C}^t}\mathbf{d}^t(\frac{\mathbf{d}^t-\sqrt{\bar{\alpha}_t}\mathbf{d}^0}{1-\bar{\alpha}_t}) = \frac{\partial_{\mathcal{C}^t}\mathbf{d}^t\cdot(\mathbf{d}^t-\sqrt{\bar{\alpha}_t}\mathbf{d}^0)}{\sqrt{1-\bar{\alpha}_t}}$.

5.4.4. Sampling

With a learned reverse dynamics $\epsilon_\theta(\mathcal{G},\mathcal{C}^t,t)$, the transition means $\mu_\theta(\mathcal{G},\mathcal{C}^t,t)$ can be calculated by equation 5.4.3. Thus, given a graph $\mathcal{G}$, its geometry $\mathcal{C}^0$ is generated by first sampling chaotic particles $\mathcal{C}^T \sim p(\mathcal{C}^T)$, and then progressively sample $\mathcal{C}^{t-1} \sim p_\theta(\mathcal{C}^{t-1}|\mathcal{G},\mathcal{C}^t)$ for $t = T,T-1,\cdots,1$. This process is Markovian, which gradually shifts the previous noisy positions towards equilibrium states. We provide the pseudo code of the whole sampling process in Algorithm 2.

Algorithm 2 Sampling Algorithm of GeoDiff.

```
Input: the molecular graph $\mathcal{G}$, the learned reverse model $\epsilon_\theta$.
Output: the molecular conformation $\mathcal{C}$.
Sample $\mathcal{C}^T \sim p(\mathcal{C}^T) = \mathcal{N}(0,I)$
for $s = T,T-1,\cdots,1$ do
    Shift $\mathcal{C}^s$ to zero CoM
    Compute $\mu_\theta(\mathcal{C}^s,\mathcal{G},s)$ from $\epsilon_\theta(\mathcal{C}^s,\mathcal{G},s)$ using equation 5.4.3
    Sample $\mathcal{C}^{s-1} \sim \mathcal{N}(\mathcal{C}^{s-1}; \mu_\theta(\mathcal{C}^s,\mathcal{G},s), \sigma_t^2 I)$
end for
return $\mathcal{C}^0$ as $\mathcal{C}$
```

5.5. Experiment

In this section, we empirically evaluate GeoDiff on the task of equilibrium conformation generation for both small and drug-like molecules. Following existing work (Shi et al., 2021; Ganea et al., 2021), we test the proposed method as well as the competitive baselines on two standard benchmarks: **Conformation Generation** (Sec. 5.5.2) and **Property Prediction** (Sec. 5.5.3). We first present the general experiment setups, and then describe task-specific evaluation protocols and discuss the results in each section. The implementation details are provided in Appendix 5.7.3.

5.5.1. Experiment Setup

Datasets. Following prior works (Xu et al., 2021a,b), we also use the recent GEOM-QM9 (Ramakrishnan et al., 2014) and GEOM-Drugs (Axelrod and Gomez-Bombarelli, 2020) datasets. The former one contains small molecules while the latter one are medium-sized organic compounds. We borrow the data split produced by Shi et al. (2021). For both datasets, the training split consists of 40,000 molecules with 5 conformations for each, resulting in 200,000 conformations in total. The valid split share the same size as training split. The test split contains 200 distinct molecules, with 22,408 conformations for QM9 and 14,324 ones for Drugs.

Baselines. We compare GeoDiff with 6 recent or established state-of-the-art baselines. For the ML approaches, we test the following models with highest reported performance: CVGAE (Mansimov et al., 2019), GraphDG (Simm and Hernández-Lobato, 2020), CGCF (Xu et al., 2021a), ConfVAE (Xu et al., 2021b) and ConfGF (Shi et al., 2021). We also test the classic RDKit (Riniker and Landrum, 2015) method, which is arguably the most popular open-source software for conformation generation. We refer readers to Sec. 5.2 for a detailed discussion of these models.

5.5.2. Conformation Generation

Evaluation metrics. The task aims to measure both quality and diversity of generated conformations by different models. We follow Ganea et al. (2021) to evaluate 4 metrics built upon root-mean-square deviation (RMSD), which is defined as the normalized Frobenius norm of two atomic coordinates matrices, after alignment by Kabsch algorithm (Kabsch, 1976). Formally, let S_g and S_r denote the sets of generated and reference conformers respectively, then the **Cov**erage and **Mat**ching metrics (Xu et al., 2021a) following the conventional *Recall* measurement can be defined as:

$$\text{COV-R}(S_g, S_r) = \frac{1}{|S_r|}\Big|\Big\{\mathcal{C} \in S_r \,|\, \text{RMSD}(\mathcal{C}, \hat{\mathcal{C}}) \le \delta, \hat{\mathcal{C}} \in S_g\Big\}\Big|, \tag{5.5.1}$$

$$\text{MAT-R}(S_g, S_r) = \frac{1}{|S_r|}\sum_{\mathcal{C}\in S_r} \min_{\hat{\mathcal{C}}\in S_g} \text{RMSD}(\mathcal{C}, \hat{\mathcal{C}}), \tag{5.5.2}$$

where δ is a pre-defined threshold. The other two metrics COV-P and MAT-P inspired by *Precision* can be defined similarly but with the generated and reference sets exchanged. In practice, S_g is set as twice of the size of S_r for each molecule. Intuitively, the COV scores measure the percentage of structures in one set covered by another set, where covering means the RMSD between two conformations is within a certain threshold δ. By contrast, the MAT scores measure the average RMSD of conformers in one set with its closest neighbor in another set. In general, higher COV rates or lower MAT score suggest that more realistic conformations are generated. Besides, the *Precision* metrics depend more on the quality, while the *Recall* metrics concentrate more on the diversity. Either metrics can be more appealing considering the specific scenario. Following previous works (Xu et al., 2021a; Ganea et al., 2021), δ is set as 0.5Å and 1.25Å for QM9 and Drugs datasets respectively.

Results & discussion. The results are summarized in Tab. 5.1 and Tab. 5.5 (left in Appendix. 5.7.4). As noted in Sec. 5.4.3, GeoDiff can be trained with two types of modified ELBO, named *alignment* and *chain-rule* approaches. We denote models learned by these two objectives as GeoDiff-A and GeoDiff-C respectively. As shown in the tables, GeoDiff consistently outperform the state-of-the-art ML models on all datasets and metrics, especially by a significant margin for more challenging large molecules (Drugs

Tableau 5.1. Results on the **GEOM-Drugs** dataset, without FF optimization.

	COV-R (%) ↑		MAT-R (Å) ↓		COV-P (%) ↑		MAT-P (Å) ↓	
Models	Mean	Median	Mean	Median	Mean	Median	Mean	Median
CVGAE	0.00	0.00	3.0702	2.9937	-	-	-	-
GraphDG	8.27	0.00	1.9722	1.9845	2.08	0.00	2.4340	2.4100
CGCF	53.96	57.06	1.2487	1.2247	21.68	13.72	1.8571	1.8066
ConfVAE	55.20	59.43	1.2380	1.1417	22.96	14.05	1.8287	1.8159
GeoMol	67.16	71.71	1.0875	1.0586	-	-	-	-
ConfGF	62.15	70.93	1.1629	1.1596	23.42	15.52	1.7219	1.6863
GeoDiff-A	88.36	96.09	0.8704	0.8628	60.14	61.25	1.1864	1.1391
GeoDiff-C	**89.13**	**97.88**	**0.8629**	**0.8529**	**61.47**	**64.55**	**1.1712**	**1.1232**

* The COV-R and MAT-R results of CVGAE, GraphDG, CGCF, and ConfGF are borrowed from Shi et al. (2021). The results of GeoMol are borrowed from a most recent study Zhu et al. (2022). Other results are obtained by our own experiments. The results of all models for the GEOM-QM9 dataset (summarized in Tab. 5.5) are collected in the same way.

dataset). The results demonstrate the superior capacity of GeoDiff to model the multi modal distribution, and generative both accurate and diverse conformations. We also notice that in general GeoDiff-C performs slightly better than GeoDiff-A, which suggests that *chain-rule approach* leads to a better optimization procedure. We thus take GeoDiff-C as the representative in the following comparisons. We visualize samples generated by different models in Fig. 5.2 to provide a qualitative comparison, where GeoDiff is shown to capture better both local and global structures.

On the more challenging Drugs dataset, we further test RDKit. As shown in Tab. 5.2, our observation is in line with previous studies (Shi et al., 2021) that the state-of-the-art ML models (shown in Tab. 5.1) perform better on COV-R and MAT-R. However, for the new *Precision*-based metrics we found that ML models are still not comparable. This indicates that ML models tend to explore more possible representatives while RDKit concentrates on a few most common ones, prioritizes quality over diversity. Previous works (Mansimov et al., 2019; Xu et al., 2021b) suggest that this is because RDKit involves an additional empirical force field (FF) (Halgren, 1996b) to optimize the structure, and we follow them to also combine GeoDiff with FF to yield a more fair comparison. Results in Tab. 5.2 demonstrate that GeoDiff +FF can keep the superior diversity (*Recall* metrics) while also enjoy significantly improved accuracy ((*Precision* metrics)).

5.5.3. Property Prediction

Evaluation metrics. This task estimates the molecular *ensemble properties* (Axelrod and Gomez-Bombarelli, 2020) over a set of generated conformations. This can provide an

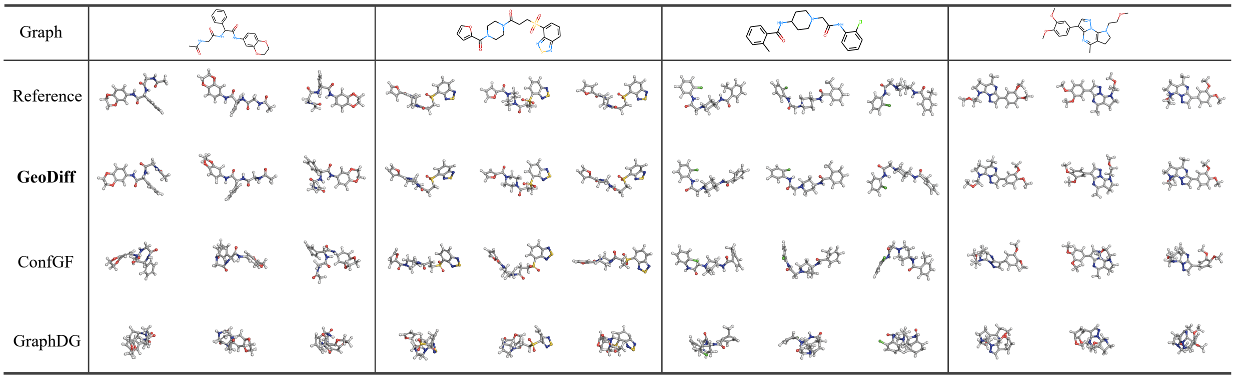

Fig. 5.2. Examples of generated structures from Drugs dataset. For every model, we show the conformation best-aligned with the ground truth. More examples are provided in Appendix 5.7.5.

Tableau 5.2. Results on the **GEOM-Drugs** dataset, with FF optimization.

Models	COV-R (%) ↑		MAT-R (Å) ↓		COV-P (%) ↑		MAT-P (Å) ↓	
	Mean	Median	Mean	Median	Mean	Median	Mean	Median
RDKit	60.91	65.70	1.2026	1.1252	72.22	88.72	1.0976	0.9539
GeoDiff + FF	**92.27**	**100.00**	**0.7618**	**0.7340**	**84.51**	**95.86**	**0.9834**	**0.9221**

direct assessment on the quality of generated samples. In specific, we follow Shi et al. (2021) to extract a split from GEOM-QM9 covering 30 molecules, and generate 50 samples for each. Then we use the chemical toolkit Psi4 (Smith et al., 2020) to calculate each conformer's energy E and HOMO-LUMO gap ϵ, and compare the average energy $\overline{E}$, lowest energy E_{min}, average gap $\overline{\Delta\epsilon}$, minimum gap $\Delta\epsilon_{\text{min}}$, and maximum gap $\Delta\epsilon_{\text{max}}$ with the ground truth.

Results & discussions. The mean absolute errors (MAE) between calculated properties and the ground truth are reported in Tab. 5.3. CVGAE is excluded due to the poor performance, which is also reported in Simm and Hernández-Lobato (2020); Shi et al. (2021). The properties are highly sensitive to geometric structure, and thus the superior performance demonstrate that GeoDiff can consistently predict more accurate conformations across different molecules.

Tableau 5.3. MAE of predicted ensemble properties in eV.

Method	$\overline{E}$	E_{min}	$\overline{\Delta\epsilon}$	$\Delta\epsilon_{\text{min}}$	$\Delta\epsilon_{\text{max}}$
RDKit	0.9233	0.6585	0.3698	0.8021	0.2359
GraphDG	9.1027	0.8882	1.7973	4.1743	0.4776
CGCF	28.9661	2.8410	2.8356	10.6361	0.5954
ConfVAE	8.2080	0.6100	1.6080	3.9111	0.2429
ConfGF	2.7886	0.1765	0.4688	2.1843	**0.1433**
GeoDiff	**0.25974**	**0.1551**	**0.3091**	**0.7033**	0.1909

5.6. Conclusion

We propose GeoDiff, a novel probabilistic model for generating molecular conformations. GeoDiff marries denoising diffusion models with geometric representations, where we parameterize the reverse generative dynamics as a Markov chain, and novelly impose roto-translational invariance into the density with equivariant Markov kernels. We derive a tractable invariant objective from the variational lower bound to optimize the likelihood. Comprehensive experiments over multiple tasks demonstrate that GeoDiff is competitive with the existing state-of-the-art models. Future work includes further improving or accelerating the model with other recent progress of diffusion models, and extending our method to other challenging structures such as proteins.

5.7. Appendix

5.7.1. Proofs

5.7.1.1. Properties of the Diffusion Model. We include proofs for several key properties of the probabilistic diffusion model here to be self-contained. For more detailed discussions, please refer to Ho et al. (2020). Let $\{\beta_0,...,\beta_T\}$ be a sequence of variances, and $\alpha_t = 1-\beta_t$ and $\bar{\alpha}_t = \prod_{s=1}^{t} \alpha_s$. The two following properties are crucial for deriving the final tractable objective in equation 5.4.8.

Property 1. *Tractable marginal of the forward process:*

$$q(\mathcal{C}^t|\mathcal{C}^0) = \int q(\mathcal{C}^{1:t}|\mathcal{C}^0)\, d\mathcal{C}^{1:(t-1)} = \mathcal{N}(\mathcal{C}^t;\ \sqrt{\bar{\alpha}_t}\mathcal{C}^0, (1-\bar{\alpha}_t)I).$$

Démonstration. Let ϵ_i's be independent standard Gaussian random variables. Then, by definition of the Markov kernels $q(\mathcal{C}^t|\mathcal{C}^{t-1})$ in equation 5.4.1, we have

$$\begin{aligned}
\mathcal{C}^t &= \sqrt{\alpha_t}\mathcal{C}^{t-1} + \sqrt{\beta_t}\epsilon_t \\
&= \sqrt{\alpha_t\alpha_{t-1}}\mathcal{C}^{t-2} + \sqrt{\alpha_t\beta_{t-1}}\epsilon_{t-1} + \sqrt{\beta_t}\epsilon_t \\
&= \sqrt{\alpha_t\alpha_{t-1}\alpha_{t-1}}\mathcal{C}^{t-3} + \sqrt{\alpha_t\alpha_{t-1}\beta_{t-2}}\epsilon_{t-2} + \sqrt{\alpha_t\beta_{t-1}}\epsilon_{t-1} + \sqrt{\beta_t}\epsilon_t \\
&= \cdots \\
&= \sqrt{\bar{\alpha}_t}\mathcal{C}^0 + \sqrt{\alpha_t\alpha_{t-1}\cdots\alpha_2\beta_1}\epsilon_1 + \cdots + \sqrt{\alpha_t\beta_{t-1}}\epsilon_{t-1} + \sqrt{\beta_t}\epsilon_t
\end{aligned} \tag{5.7.1}$$

Therefore $q(\mathcal{C}^t|\mathcal{C}^0)$ is still Gaussian, and the mean of $\mathcal{C}^t$ is $\sqrt{\bar{\alpha}_t}\mathcal{C}^0$, and the variance matrix is $(\alpha_t\alpha_{t-1}\cdots\alpha_2\beta_1 + \cdots + \alpha_t\beta_{t-1} + \beta_t)I = (1-\bar{\alpha}_t)I$. Then we have:

$$q(\mathcal{C}^t|\mathcal{C}^0) = \mathcal{N}(\mathcal{C}^t;\ \sqrt{\bar{\alpha}_t}\mathcal{C}^0,\ (1-\bar{\alpha}_t)I).$$

This property provides convenient closed-form evaluation of $\mathcal{C}^t$ knowing $\mathcal{C}^0$:

$$\mathcal{C}^t = \sqrt{\bar{\alpha}_t}\mathcal{C}^0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon},$$

where $\boldsymbol{\epsilon} \sim \mathcal{N}(0, I)$.

Besides, it is worth noting that,

$$q(\mathcal{C}^T|\mathcal{C}^0) = \mathcal{N}(\mathcal{C}^T;\ \sqrt{\bar{\alpha}_T}\mathcal{C}^0,\ (1-\bar{\alpha}_T)I),$$

where $\bar{\alpha}_T = \prod_{t=1}^{T}(1-\beta_t)$ approaches zero with large T, which indicates the diffusion process can finally converge into a whitened noisy distribution. □

Property 2. *Tractable posterior of the forward process:*

$$q(\mathcal{C}^{t-1}|\mathcal{C}^t,\mathcal{C}^0) = \mathcal{N}(\mathcal{C}^{t-1};\ \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t}\mathcal{C}^0 + \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}\mathcal{C}^t, \frac{(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}\beta_t I).$$

Démonstration. Let $\tilde{\beta}_t = \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t$, then we can derive the posterior by Bayes rule:

$$\begin{aligned} q(\mathcal{C}^{t-1}|\mathcal{C}^t,\mathcal{C}^0) &= \frac{q(\mathcal{C}^t|\mathcal{C}^{t-1})\ q(\mathcal{C}^{t-1}|\mathcal{C}^0)}{q(\mathcal{C}^t|\mathcal{C}^0)} \\ &= \frac{\mathcal{N}(\mathcal{C}^t;\sqrt{\alpha_t}\mathcal{C}^{t-1},\beta_t I)\ \mathcal{N}(\mathcal{C}^{t-1};\sqrt{\bar{\alpha}_{t-1}}\mathcal{C}^0,(1-\bar{\alpha}_{t-1})I)}{\mathcal{N}(\mathcal{C}^t;\sqrt{\bar{\alpha}_t}\mathcal{C}^0,(1-\bar{\alpha}_t)I)} \\ &= (2\pi\beta_t)^{-\frac{d}{2}}(2\pi(1-\bar{\alpha}_{t-1}))^{-\frac{d}{2}}(2\pi(1-\bar{\alpha}_t))^{\frac{d}{2}}\times \\ &\quad \exp\left(-\frac{\|\mathcal{C}^t-\sqrt{\alpha_t}\mathcal{C}^{t-1}\|^2}{2\beta_t} - \frac{\|\mathcal{C}^{t-1}-\sqrt{\bar{\alpha}_{t-1}}\mathcal{C}^0\|^2}{2(1-\bar{\alpha}_{t-1})} + \frac{\|\mathcal{C}^t-\sqrt{\bar{\alpha}_t}\mathcal{C}^0\|^2}{2(1-\bar{\alpha}_t)}\right) \\ &= (2\pi\tilde{\beta}_t)^{-\frac{d}{2}}\exp\left(-\frac{1}{2\tilde{\beta}_t}\left\|\mathcal{C}^{t-1} - \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t}\mathcal{C}^0 - \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}\mathcal{C}^t\right\|^2\right) \end{aligned} \tag{5.7.2}$$

Then we have the posterior $q(\mathcal{C}^{t-1}|\mathcal{C}^t,\mathcal{C}^0)$ as the given form. □

5.7.1.2. Proof of Proposition 1. Let T_g be some roto-translational transformations of a group element $g \in$ SE(3), and let $p(x_T)$ be a density which is SE(3)-invariant, *i.e.*, $p(x_T) = p(T_g(x_T))$. If the Markov transitions $p(x_{t-1}|x_t)$ are SE(3)-equivariant, *i.e.*, $p(x_{t-1}|x_t) = p(T_g(x_{t-1})|T_g(x_t))$, then we have that the density $p_\theta(x_0) = \int p(x_T)p_\theta(x_{0:T-1}|x_T)\mathrm{d}\boldsymbol{x}_{1:T}$ is also SE(3)-invariant.

Démonstration.

$$\begin{aligned} p_\theta(T_g(x_0)) &= \int p(T_g(x_T))p_\theta(T_g(x_{0:T-1})|T_g(x_T))\mathrm{d}\boldsymbol{x}_{1:T} \\ &= \int p(T_g(x_T))\Pi_{t=1}^{T}p_\theta(T_g(x_{t-1})|T_g(x_t))\mathrm{d}\boldsymbol{x}_{1:T} \\ &= \int p(x_T)\Pi_{t=1}^{T}p_\theta(T_g(x_{t-1})|T_g(x_t))\mathrm{d}\boldsymbol{x}_{1:T} \quad \text{(invariant prior } p(x_T)\text{)} \\ &= \int p(x_T)\Pi_{t=1}^{T}p_\theta(x_{t-1}|x_t)\mathrm{d}\boldsymbol{x}_{1:T} \quad \text{(equivariant kernels } p(x_{t-1}|x_t)\text{)} \\ &= \int p(x_T)p_\theta(x_{0:T-1}|x_T)\mathrm{d}\boldsymbol{x}_{1:T} \\ &= p_\theta(x_0) \end{aligned} \tag{5.7.3}$$

□

5.7.1.3. Proof of Proposition 2. In this section we prove that the output $\mathbf{x}$ of GFN defined in equation 5.4.4, 5.4.5 and 5.4.6 is translationally invariant and rotationally equivariant with the input $\mathcal{C}$. Let $g \in \mathbb{R}^3$ denote any translation transformations and orthogonal matrices $R \in \mathbb{R}^{3\times 3}$ denote any rotation transformations. let $R\mathbf{x}$ be shorthand for $(R\mathbf{x}_1, \cdots, R\mathbf{x}_N)$. Formally, we aim to prove that the model satisfies:

$$R\mathbf{x}^{l+1}, \mathbf{h}^{l+1} = \mathrm{GFN}(R\mathbf{x}^l, R\mathcal{C} + g, \mathbf{h}^l). \tag{5.7.4}$$

This equation indicates that, given $\mathbf{x}^l$ already rotationally equivalent with $\mathcal{C}$, and $\mathbf{h}^l$ already invariant, then such property can propagate through a single GFN layer to $\mathbf{x}^{l+1}$ and $\mathbf{h}^{l+1}$.

Démonstration. Firstly, given that $\mathbf{h}^l$ already invariant to SE(3) transformations, we have that the messages $\mathbf{m}_{ij}$ calculated from equation 5.4.4 will also be invariant. This is because it sorely relies on the distance between two atoms, which are manifestly invariant to rotations $\|R\mathbf{x}_i^l - R\mathbf{x}_j^l\|^2 = (\mathbf{x}_i^l - \mathbf{x}_j^l)^\top R^\top R(\mathbf{x}_i^l - \mathbf{x}_j^l) = (\mathbf{x}_i^l - \mathbf{x}_j^l)^\top I(\mathbf{x}_i^l - \mathbf{x}_j^l) = \|\mathbf{x}_i^l - \mathbf{x}_j^l\|^2$. Formally, the invariance of messages in equation 5.4.4 can be written as:

$$\mathbf{m}_{i,j} = \Phi_m\left(\mathbf{h}_i^l, \mathbf{h}_j^l, \left\|R\mathbf{x}_i^l - R\mathbf{x}_j^l\right\|^2, e_{ij}\right) = \Phi_m\left(\mathbf{h}_i^l, \mathbf{h}_j^l, \left\|\mathbf{x}_i^l - \mathbf{x}_j^l\right\|^2, e_{ij}\right). \tag{5.7.5}$$

And similarly, the $\mathbf{h}^{t+1}$ updated from equation 5.4.5 will also be invariant.

Next, we prove that the vector $\mathbf{x}$ updated from equation 5.4.6 preserves rotational equivariance and translational invariance. Given $\mathbf{m}_{ij}$ already invariant as proven above, we have that:

$$\sum_{j \in \mathcal{N}(i)} \frac{1}{d_{ij}} \left(R\mathbf{c}_i + g - R\mathbf{c}_j - g\right) \Phi_x\left(\mathbf{m}_{i,j}\right) = R \sum_{j \in \mathcal{N}(i)} \frac{1}{d_{ij}} \left(\mathbf{c}_i - \mathbf{c}_j\right) \Phi_x\left(\mathbf{m}_{i,j}\right) = R\mathbf{x}_i^{l+1}. \tag{5.7.6}$$

Therefore, we have that rotating and translating $\mathbf{c}$ results in the same rotation and no translation on $\mathbf{x}^{l+1}$ by updating through equation 5.4.6.

Thus we can conclude that the property defined in equation 5.7.4 is satisfied. □

Having proved the equivariance property of a single GFN layer, then inductively, we can draw conclusion that a composition of L GFN layers will also preserve the same equivariance.

5.7.1.4. Proof of Proposition 3. We first derive the variational lower bound (ELBO) objective in equation 5.4.7. The ELBO can be calculated as follows:

$$\begin{aligned}
\mathbb{E}\log p_\theta(\mathcal{C}^0|\mathcal{G}) &= \mathbb{E}\log \mathbb{E}_{q(\mathcal{C}^{1:T}|\mathcal{C}^0)}\Big[\frac{p_\theta(\mathcal{C}^{0:T-1}|\mathcal{G},\mathcal{C}^T)\times p(\mathcal{C}^T)}{q(\mathcal{C}^{1:T}|\mathcal{C}^0)}\Big] \\
&\geq \mathbb{E}_q \log \frac{p_\theta(\mathcal{C}^{0:T-1}|\mathcal{G},\mathcal{C}^T)\times p(\mathcal{C}^T)}{q(\mathcal{C}^{1:T}|\mathcal{C}^0)} \\
&= \mathbb{E}_q\Big[\log p(\mathcal{C}^T) - \sum_{t=1}^{T}\log\frac{p_\theta(\mathcal{C}^{t-1}|\mathcal{G},\mathcal{C}^t)}{q(\mathcal{C}^t|\mathcal{C}^{t-1})}\Big] \\
&= \mathbb{E}_q\Big[\log p(\mathcal{C}^T) - \log\frac{p_\theta(\mathcal{C}^0|\mathcal{G},\mathcal{C}^1)}{q(\mathcal{C}^1|\mathcal{C}^0)} - \sum_{t=2}^{T}\Big(\log\frac{p_\theta(\mathcal{C}^{t-1}|\mathcal{G},\mathcal{C}^t)}{q(\mathcal{C}^{t-1}|\mathcal{C}^t,\mathcal{C}^0)} + \log\frac{q(\mathcal{C}^{t-1}|\mathcal{C}^0)}{q(\mathcal{C}^t|\mathcal{C}^0)}\Big)\Big] \\
&= \mathbb{E}_q\Big[\log\frac{p(\mathcal{C}^T)}{q(\mathcal{C}^T|\mathcal{C}^0)} - \log p_\theta(\mathcal{C}^0|\mathcal{G},\mathcal{C}^1) - \sum_{t=2}^{T}\log\frac{p_\theta(\mathcal{C}^{t-1}|\mathcal{G},\mathcal{C}^t)}{q(\mathcal{C}^{t-1}|\mathcal{C}^t,\mathcal{C}^0)}\Big] \\
&= -\mathbb{E}_q\Big[\mathrm{KL}\left(q(\mathcal{C}^T|\mathcal{C}^0)\|p(\mathcal{C}^T)\right) + \sum_{t=2}^{T}\mathrm{KL}\left(q(\mathcal{C}^{t-1}|\mathcal{C}^t,\mathcal{C}^0)\|p_\theta(\mathcal{C}^{t-1}|\mathcal{G},\mathcal{C}^t)\right) - \log p_\theta(\mathcal{C}^0|\mathcal{G},\mathcal{C}^1)\Big].
\end{aligned} \tag{5.7.7}$$

It can be noted that the first term $\mathrm{KL}\left(q(\mathcal{C}^T|\mathcal{C}^0)\|p(\mathcal{C}^T)\right)$ is a constant, which can be omitted in the objective. Furthermore, for brevity, we also merge the final term $\log p_\theta(\mathcal{C}^0|\mathcal{G},\mathcal{C}^1)$ into the second term (sum over KL divergences), and finally derive that $\mathcal{L}_{\text{ELBO}} = \sum_{t=1}^{T} D_{\text{KL}}(q(\mathcal{C}^{t-1}|\mathcal{C}^t,\mathcal{C}^0)\|p_\theta(\mathcal{C}^{t-1}|\mathcal{G},\mathcal{C}^t))$ as in equation 5.4.7.

Now we consider how to compute the KL divergences as the proposition 3. Since both $q(\mathcal{C}^{t-1}|\mathcal{C}^t,\mathcal{C}^0)$ and $p_\theta(\mathcal{C}^{t-1}|\mathcal{G},\mathcal{C}^t)$ are Gaussian share the same covariance matrix $\tilde{\beta}_t I$, the KL divergence between them can be calculated by the squared ℓ_2 distance between their means weighed by a certain weights $\frac{1}{2\tilde{\beta}_t}$. By the expression of $q(\mathcal{C}^t|\mathcal{C}^0)$, we have the reparameterization that $\mathcal{C}^t = \sqrt{\bar{\alpha}_t}\mathcal{C}^0 + \sqrt{1-\bar{\alpha}_t}\epsilon$. Then we can derive:

$$\begin{aligned}
&\mathbb{E}_q\ \mathrm{KL}\left(q(\mathcal{C}^{t-1}|\mathcal{C}^t,\mathcal{C}^0)\|p_\theta(\mathcal{G},\mathcal{C}^{t-1}|\mathcal{C}^t)\right) \\
&= \frac{1}{2\tilde{\beta}_t}\mathbb{E}_{\mathcal{C}^0}\left\|\frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t}\mathcal{C}^0 + \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}\mathcal{C}^t - \frac{1}{\sqrt{\alpha_t}}\left(\mathcal{C}^t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(\mathcal{C}^t,\mathcal{G},t)\right)\right\|^2 \\
&= \frac{1}{2\tilde{\beta}_t}\mathbb{E}_{\mathcal{C}^0,\epsilon}\left\|\frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t}\cdot\frac{\mathcal{C}^t-\sqrt{1-\bar{\alpha}_t}\epsilon}{\sqrt{\bar{\alpha}_t}} + \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}\mathcal{C}^t - \frac{1}{\sqrt{\alpha_t}}\left(\mathcal{C}^t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(\mathcal{C}^t,\mathcal{G},t)\right)\right\|^2 \\
&= \frac{1}{2\tilde{\beta}_t}\cdot\frac{\beta_t^2}{\alpha_t(1-\bar{\alpha}_t)}\mathbb{E}_{\mathcal{C}^0,\epsilon}\left\|0\cdot\mathcal{C}^t + \epsilon - \epsilon_\theta(\mathcal{C}^t,\mathcal{G},t)\right\|^2 \\
&= \frac{\beta_t^2}{2\frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t\alpha_t(1-\bar{\alpha}_t)}\mathbb{E}_{\mathcal{C}^0,\epsilon}\left\|\epsilon - \epsilon_\theta(\mathcal{C}^t,\mathcal{G},t)\right\|^2 \\
&= \gamma_t\mathbb{E}_{\mathcal{C}^0,\epsilon}\left\|\epsilon - \epsilon_\theta(\mathcal{C}^t,t)\right\|^2,
\end{aligned} \tag{5.7.8}$$

where γ_t represent the wights $\frac{\beta_t}{2\alpha_t(1-\bar{\alpha}_{t-1})}$. And we finish the proof.

5.7.1.5. Analysis of the invariant density in Sec. 5.4.2. Given a geometric system $x \in \mathbb{R}^{N \cdot 3}$, we obtain the CoM-free $\hat{x}$ by subtracting its CoM. This can be considered as a linear transformation:

$$\hat{x} = Qx, \text{ where } Q = I_3 \otimes \left(I_N - \frac{1}{N} \mathbf{1}_N \mathbf{1}_N^T \right) \tag{5.7.9}$$

where I_k denotes the $k \times k$ identity matrix and $\mathbf{1}_k$ denotes the k-dimensional vector filled with ones. It can be noted that Q is a symmetric projection operator, *i.e.*, $Q^2 = Q$ and $Q^T = Q$. And we also have that $\text{rank}[Q] = (N-1) \cdot 3$. Furthermore, let U represent the space of CoM-free systems, we can easily have that $Qy = y$ for any $y \in U$ since the CoM of y is already zero.

Formally, let $n = N \cdot 3$ and set $\mathbb{R}^n$ with an isotropic normal distribution $\rho = \mathcal{N}(0, I_n)$, then the CoM-free density can be formally written as $\hat{\rho} = \mathcal{N}(0, QI_nQ^T) = \mathcal{N}(0, QQ^T)$. Thus, sampling from $\hat{\rho}$ can be trivially achieved by sampling from ρ and then projecting with Q. And $\hat{\rho}(y)$ can be calculated by $\rho(y)$ since for any $y \in U$ we have $\|y\|_2^2 = \|Qy\|_2^2$, and thus $\rho(y) = \hat{\rho}(y)$.

And in this paper, with the SE(3)-equivariant Markov kernels of the reverse process, any CoM-free system will transit to another CoM-free system. And thus we can induce a well-defined Markov chain on the subspace spanned by Q.

5.7.2. Other related work

Protein structure generation. There has also been many recent works working on protein structure folding. For example, Boltzmann generators Noé et al. (2019) use flow-based models to generate the structure of protein main chains. AlQuraishi (2019) uses recurrent networks to model the amino acid sequences. Ingraham et al. (2019) proposed neural networks to learn an energy simulator to infer the protein structures. Most recently, AlphaFold Senior et al. (2020); Jumper et al. (2021) has significantly improved the performance of protein structure generation. Nevertheless, proteins are mainly linear backbone structures while general molecules are highly branched with various rings, making protein folding approaches unsuitable for our setting.

Point cloud generation. Recently, some other works (Luo and Hu, 2021; Chibane et al., 2020) has also been proposed for 3D structure generation with diffusion-based models, but focus on the point cloud problem. Unfortunately, in general, point clouds are not considered as graphs with various atom and bond information, and equivariance is also not widely considered, making these methods fundamentally different from our model.

5.7.3. Experiment details

In this section, we introduce the details of our experiments. In practice, the means ϵ_θ are parameterized as compositions of both typical invariant MPNNs (Schütt et al., 2017) and the proposed equivariant GFNs in Sec. 5.4.2. As a default setup, the MPNNs for parameterizing the means ϵ_θ are all implemented with 4 layers, and the hidden embedding dimension is set as 128. After the MPNNs, we can obtain the informative invariant atom embeddings, which we denote as $\mathbf{h}^0$. Then the embeddings $\mathbf{h}^0$ are fed into equivariant layers and updated with equation 5.4.4, equation 5.4.5, and equation 5.4.6 to obtain the equivariant output. For the training of GeoDiff, we train the model on a single Tesla V100 GPU with a learning rate of 0.001 until convergence and Adam (Kingma and Welling, 2013) as the optimizer. The practical training time is around 48 hours. The other hyper-parameters of GeoDiff are summarized in Tab. 5.4, including highest variance level β_T, lowest variance level β_T, the variance schedule, number of diffusion timesteps T, radius threshold for determining the neighbor of atoms τ, batch size, and number of training iterations.

Tableau 5.4. Additional hyperparameters of our GeoDiff.

Task	β_1	β_T	β scheduler	T	τ	Batch Size	Train Iter.
QM9	1e-7	2e-3	sigmoid	5000	10Å	64	1M
Drugs	1e-7	2e-3	sigmoid	5000	10Å	32	1M

5.7.4. Additional experiments

5.7.4.1. Results for GEOM-QM9. The results on the GEOM-QM9 dataset are reported in Tab. 5.5.

5.7.4.2. Ablation study with fewer diffusion steps. We also test our method with fewer diffusion steps. Specifically, we test the setting with $T = 1000$, $\beta_1 =$1e-7 and $\beta_T =$9e-3. The results on the more challenging Drugs dataset are shown in Tab. 5.6. Compared with the results in Tab. 5.1, we can observe that when setting the diffusion steps as 1000, though slightly weaker than the performance with 5000 decoding steps, the model can already outperforms all existing baselines. Note that, the most competitive baseline ConfGF (Shi et al., 2021) also requires 5000 sampling steps, which indicates that our model can achieve better performance with fewer computational costs compared with the state-of-the-art method.

5.7.5. More Visualizations

We provide more visualization of generated structures in Fig. 5.3. The molecules are chosen from the test split of GEOM-Drugs dataset.

Tableau 5.5. Results on the **GEOM-QM9** dataset, without FF optimization.

	COV-R (%) ↑		MAT-R (Å) ↓		COV-P (%) ↑		MAT-P (Å) ↓	
Models	Mean	Median	Mean	Median	Mean	Median	Mean	Median
CVGAE	0.09	0.00	1.6713	1.6088	-	-	-	-
GraphDG	73.33	84.21	0.4245	0.3973	43.90	35.33	0.5809	0.5823
CGCF	78.05	82.48	0.4219	0.3900	36.49	33.57	0.6615	0.6427
ConfVAE	77.84	88.20	0.4154	0.3739	38.02	34.67	0.6215	0.6091
GeoMol	71.26	72.00	0.3731	0.3731	-	-	-	-
ConfGF	88.49	94.31	0.2673	0.2685	46.43	43.41	0.5224	0.5124
GeoDiff-A	**90.54**	**94.61**	0.2104	0.2021	52.35	50.10	0.4539	0.4399
GeoDiff-C	90.07	93.39	**0.2090**	**0.1988**	**52.79**	**50.29**	**0.4448**	**0.4267**

Tableau 5.6. Additional results on the **GEOM-Drugs** dataset, without FF optimization.

	COV-R (%) ↑		MAT-R (Å) ↓		COV-P (%) ↑		MAT-P (Å) ↓	
Models	Mean	Median	Mean	Median	Mean	Median	Mean	Median
GeoDiff (T=1000)	82.96	96.29	0.9525	0.9334	48.27	46.03	1.3205	1.2724

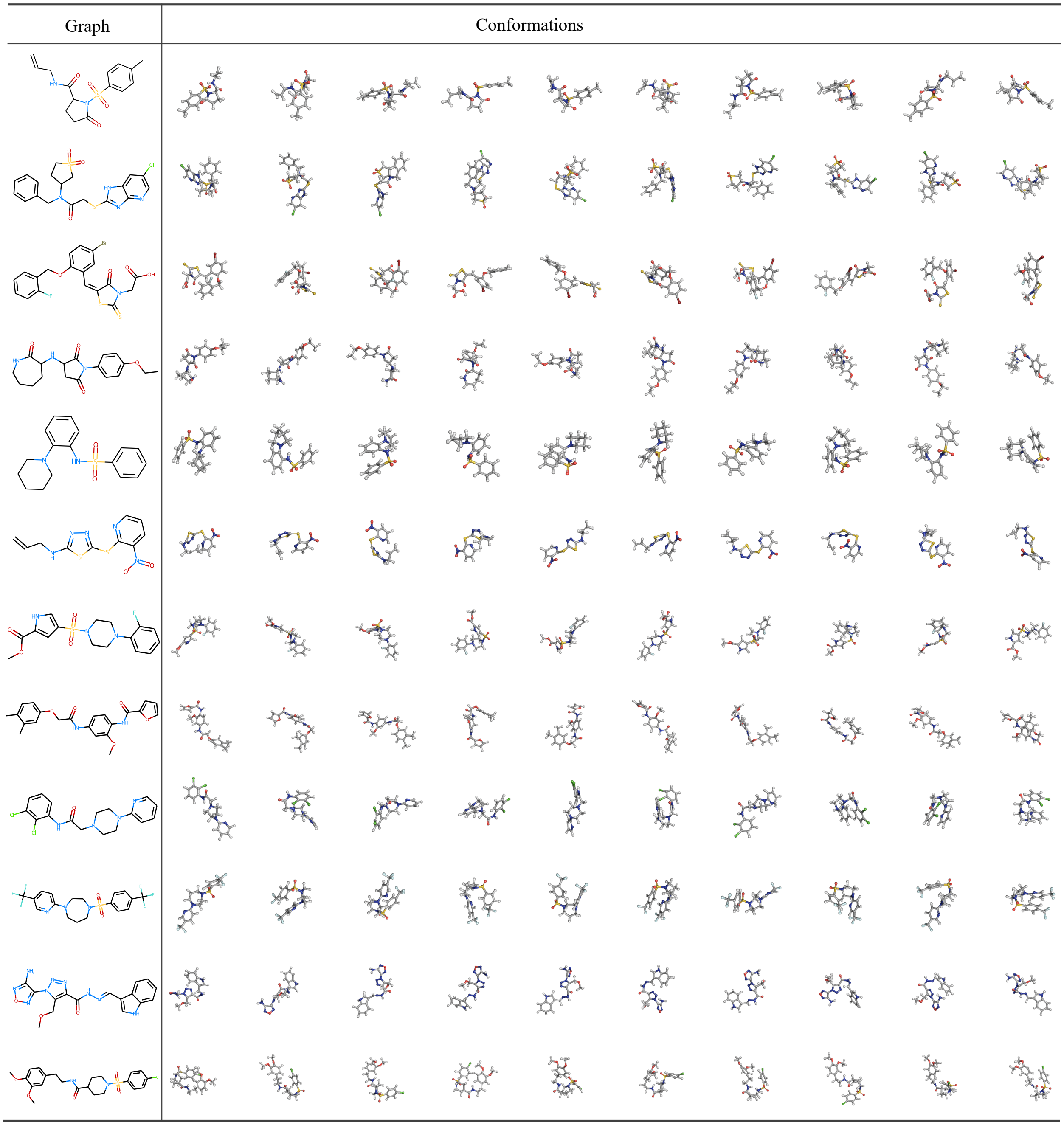

Fig. 5.3. Visualization of drug-like conformations generated by GeoDiff.

Chapter 6

Conclusion

In this thesis, we propose CGCF, ETM, and GeoDiff, three principled probabilistic models for molecular conformation generation. Our methods novelly combine the progress of geometrical representation learning and deep generative models. Specifically, we introduced flow-based, energy-based, and denoising diffusion generative models to this scenario, and keep the property of roto-translational invariance by parameterizing the models with equivariant graph neural networks. Our generative model can extract informative geometric features from the complex conformations, and enjoy a high capacity for modeling multi-modal distributions. Comprehensive experiments demonstrate that our methods can achieve consistent improvement over previous baselines on several benchmarks. Future work includes exploring other probabilistic models in the context of geometry generation and extending our method to other more challenging structures such as proteins and catalysts.

Références bibliographiques

AlQuraishi, M. (2019). End-to-end differentiable learning of protein structure. *Cell systems*, 8(4):292–301.

Axelrod, S. and Gomez-Bombarelli, R. (2020). Geom: Energy-annotated molecular conformations for property prediction and molecular generation. *arXiv preprint arXiv:2006.05531*.

Ballard, A. J., Martiniani, S., Stevenson, J. D., Somani, S., and Wales, D. J. (2015). Exploiting the potential energy landscape to sample free energy. *Wiley Interdisciplinary Reviews: Computational Molecular Science*, 5(3):273–289.

Batzner, S., Smidt, T. E., Sun, L., Mailoa, J. P., Kornbluth, M., Molinari, N., and Kozinsky, B. (2021). Se (3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials. *arXiv preprint arXiv:2101.03164*.

Behler, J. and Parrinello, M. (2007). Generalized neural-network representation of high-dimensional potential-energy surfaces. *Physical review letters*, 98(14):146401.

Bengio, Y., Mesnil, G., Dauphin, Y., and Rifai, S. (2013a). Better mixing via deep representations. In *International conference on machine learning*, pages 552–560.

Bengio, Y., Yao, L., Alain, G., and Vincent, P. (2013b). Generalized denoising auto-encoders as generative models. In *Advances in neural information processing systems*, pages 899–907.

Chen, R. T., Rubanova, Y., Bettencourt, J., and Duvenaud, D. K. (2018). Neural ordinary differential equations. In *Advances in neural information processing systems*, pages 6571–6583.

Chibane, J., Alldieck, T., and Pons-Moll, G. (2020). Implicit functions in feature space for 3d shape reconstruction and completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6970–6981.

Crippen, G. M., Havel, T. F., et al. (1988). *Distance geometry and molecular conformation*, volume 74. Research Studies Press Taunton.

Dai, H., Li, C., Coley, C., Dai, B., and Song, L. (2019). Retrosynthesis prediction with conditional graph logic network. *Advances in Neural Information Processing Systems*, 32.

Dayan, P., Hinton, G. E., Neal, R. M., and Zemel, R. S. (1995). The helmholtz machine. *Neural computation*, 7(5):889–904.

De Groot, S. R. and Mazur, P. (2013). *Non-equilibrium thermodynamics*. Courier Corporation.

De Vivo, M., Masetti, M., Bottegoni, G., and Cavalli, A. (2016). Role of molecular dynamics and related methods in drug discovery. *Journal of medicinal chemistry*, 59(9):4035–4061.

Dinh, L., Krueger, D., and Bengio, Y. (2014). Nice: Non-linear independent components estimation. *arXiv preprint arXiv:1410.8516*.

Dinh, L., Sohl-Dickstein, J., and Bengio, S. (2017). Density estimation using Real NVP. In *ICLR*.

Du, Y. and Mordatch, I. (2019). Implicit generation and generalization in energy-based models. *arXiv preprint arXiv:1903.08689*.

Duvenaud, D. K., Maclaurin, D., Iparraguirre, J., Bombarell, R., Hirzel, T., Aspuru-Guzik, A., and Adams, R. P. (2015). Convolutional networks on graphs for learning molecular fingerprints. In *Advances in neural information processing systems*, pages 2224–2232.

Fuchs, F., Worrall, D., Fischer, V., and Welling, M. (2020). Se(3)-transformers: 3d roto-translation equivariant attention networks. *NeurIPS*.

Ganea, O.-E., Pattanaik, L., Coley, C. W., Barzilay, R., Jensen, K. F., Green, W. H., and Jaakkola, T. S. (2021). Geomol: Torsional geometric generation of molecular 3d conformer ensembles. *arXiv preprint arXiv:2106.07802*.

Gebauer, N., Gastegger, M., and Schütt, K. (2019). Symmetry-adapted generation of 3d point sets for the targeted discovery of molecules. In *Advances in Neural Information Processing Systems*, pages 7566–7578.

Gebauer, N. W., Gastegger, M., Hessmann, S. S., Müller, K.-R., and Schütt, K. T. (2021). Inverse design of 3d molecular structures with conditional generative neural networks. *arXiv preprint arXiv:2109.04824*.

Gilmer, J., Schoenholz, S. S., Riley, P. F., Vinyals, O., and Dahl, G. E. (2017). Neural message passing for quantum chemistry. *arXiv preprint arXiv:1704.01212*, pages 1263–1272.

Gogineni, T., Xu, Z., Punzalan, E., Jiang, R., Kammeraad, J. A., Tewari, A., and Zimmerman, P. (2020). Torsionnet: A reinforcement learning approach to sequential conformer search. *ArXiv*, abs/2006.07078.

Goodfellow, I., Bengio, Y., Courville, A., and Bengio, Y. (2016). *Deep learning*, volume 1. MIT Press.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680.

Grathwohl, W., Chen, R. T., Bettencourt, J., Sutskever, I., and Duvenaud, D. (2018). Ffjord: Free-form continuous dynamics for scalable reversible generative models. *arXiv preprint arXiv:1810.01367*.

Gretton, A., Borgwardt, K. M., Rasch, M. J., Schölkopf, B., and Smola, A. (2012). A kernel two-sample test. *The Journal of Machine Learning Research*, 13(1):723–773.

Gutmann, M. and Hyvärinen, A. (2010). Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pages 297–304.

Halgren, T. A. (1996a). Merck molecular force field. i. basis, form, scope, parameterization, and performance of mmff94. *Journal of computational chemistry*, 17(5-6):490–519.

Halgren, T. A. (1996b). Merck molecular force field. v. extension of mmff94 using experimental data, additional computational data, and empirical rules. *Journal of Computational Chemistry*, 17(5-6):616–641.

Hawkins, P. C. (2017). Conformation generation: the state of the art. *Journal of Chemical Information and Modeling*, 57(8):1747–1756.

Hendrycks, D. and Gimpel, K. (2016). A baseline for detecting misclassified and out-of-distribution examples in neural networks. *arXiv preprint arXiv:1610.02136*.

Hinton, G. E. and Salakhutdinov, R. R. (2006). Reducing the dimensionality of data with neural networks. *science*, 313(5786):504–507.

Ho, J., Jain, A., and Abbeel, P. (2020). Denoising diffusion probabilistic models. *arXiv preprint arXiv:2006.11239*.

Hoffmann, M. and Noé, F. (2019). Generating valid euclidean distance matrices. *arXiv preprint arXiv:1910.03131*.

Hu, W., Shuaibi, M., Das, A., Goyal, S., Sriram, A., Leskovec, J., Parikh, D., and Zitnick, L. (2021). Forcenet: A graph neural network for large-scale quantum chemistry simulation.

Ingraham, J., Riesselman, A. J., Sander, C., and Marks, D. S. (2019). Learning protein structure with a differentiable simulator. In *International Conference on Learning Representations*.

Jin, W., Barzilay, R., and Jaakkola, T. (2018). Junction tree variational autoencoder for molecular graph generation. *arXiv preprint arXiv:1802.04364*.

Jing, B., Eismann, S., Suriana, P., Townshend, R. J. L., and Dror, R. (2021). Learning from protein structure with geometric vector perceptrons. In *International Conference on Learning Representations*.

Jumper, J., Evans, R., Pritzel, A., Green, T., Figurnov, M., Ronneberger, O., Tunyasuvunakool, K., Bates, R., Žídek, A., Potapenko, A., et al. (2021). Highly accurate protein structure prediction with alphafold. *Nature*, 596(7873):583–589.

Kabsch, W. (1976). A solution for the best rotation to relate two sets of vectors. *Acta Crystallographica Section A: Crystal Physics, Diffraction, Theoretical and General Crystallography*, 32(5):922–923.

Kearnes, S., McCloskey, K., Berndl, M., Pande, V., and Riley, P. (2016). Molecular graph convolutions: moving beyond fingerprints. *Journal of computer-aided molecular design*, 30(8):595–608.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kingma, D. P. and Dhariwal, P. (2018). Glow: Generative flow with invertible 1x1 convolutions. In *Advances in neural information processing systems*, pages 10215–10224.

Kingma, D. P. and Welling, M. (2013). Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

Köhler, J., Klein, L., and Noe, F. (2020). Equivariant flows: Exact likelihood generative learning for symmetric densities. In *Proceedings of the 37th International Conference on Machine Learning*.

LeCun, Y., Chopra, S., Hadsell, R., Ranzato, M., and Huang, F. (2006). A tutorial on energy-based learning. *Predicting structured data*, 1(0).

Lemke, T. and Peter, C. (2019). Encodermap: Dimensionality reduction and generation of molecule conformations. *Journal of chemical theory and computation*, 15(2):1209–1215.

Liberti, L., Lavor, C., Maculan, N., and Mucherino, A. (2014). Euclidean distance geometry and applications. *SIAM review*, 56(1):3–69.

Luo, S. and Hu, W. (2021). Diffusion probabilistic models for 3d point cloud generation. *ArXiv*, abs/2103.01458.

Luo, S., Shi, C., Xu, M., and Tang, J. (2021). Predicting molecular conformation via dynamic graph score matching. *Advances in Neural Information Processing Systems*, 34.

Mansimov, E., Mahmood, O., Kang, S., and Cho, K. (2019). Molecular geometry prediction using a deep generative graph neural network. *arXiv preprint arXiv:1904.00314*.

Miller, B., Geiger, M., Smidt, T., and Noé, F. (2020). Relevance of rotationally equivariant convolutions for predicting molecular properties. *ArXiv*, abs/2008.08461.

Ngiam, J., Chen, Z., Koh, P. W., and Ng, A. Y. (2011). Learning deep energy models. In *Proceedings of the 28th international conference on machine learning (ICML-11)*, pages 1105–1112.

Noé, F., Olsson, S., Köhler, J., and Wu, H. (2019). Boltzmann generators: Sampling equilibrium states of many-body systems with deep learning. *Science*, 365(6457):eaaw1147.

Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., and Lerer, A. (2017). Automatic differentiation in pytorch.

Ramakrishnan, R., Dral, P. O., Rupp, M., and Von Lilienfeld, O. A. (2014). Quantum chemistry structures and properties of 134 kilo molecules. *Scientific data*, 1(1):1–7.

Rappé, A. K., Casewit, C. J., Colwell, K., Goddard III, W. A., and Skiff, W. M. (1992). Uff, a full periodic table force field for molecular mechanics and molecular dynamics simulations. *Journal of the American chemical society*, 114(25):10024–10035.

Rezende, D. J. and Mohamed, S. (2015). Variational inference with normalizing flows. *arXiv preprint arXiv:1505.05770*.

Riniker, S. and Landrum, G. A. (2015). Better informed distance geometry: using what we know to improve conformation generation. *Journal of chemical information and modeling*, 55(12):2562–2574.

Rupp, M., Tkatchenko, A., Müller, K.-R., and Von Lilienfeld, O. A. (2012). Fast and accurate modeling of molecular atomization energies with machine learning. *Physical review letters*, 108(5):058301.

Satorras, V. G., Hoogeboom, E., Fuchs, F. B., Posner, I., and Welling, M. (2021a). E (n) equivariant normalizing flows for molecule generation in 3d. *arXiv preprint arXiv:2105.09016*.

Satorras, V. G., Hoogeboom, E., and Welling, M. (2021b). E(n) equivariant graph neural networks.

Schütt, K., Kindermans, P.-J., Felix, H. E. S., Chmiela, S., Tkatchenko, A., and Müller, K.-R. (2017). Schnet: A continuous-filter convolutional neural network for modeling quantum interactions. In *Advances in neural information processing systems*, pages 991–1001. Curran Associates, Inc.

Senior, A. W., Evans, R., Jumper, J., Kirkpatrick, J., Sifre, L., Green, T., Qin, C., Žídek, A., Nelson, A. W., Bridgland, A., et al. (2020). Improved protein structure prediction using potentials from deep learning. *Nature*, 577(7792):706–710.

Shi, C., Luo, S., Xu, M., and Tang, J. (2021). Learning gradient fields for molecular conformation generation. *ArXiv*.

Shi, C., Xu, M., Guo, H., Zhang, M., and Tang, J. (2020a). A graph to graphs framework for retrosynthesis prediction. In *International Conference on Machine Learning*, pages 8818–8827. PMLR.

Shi, C., Xu, M., Zhu, Z., Zhang, W., Zhang, M., and Tang, J. (2020b). Graphaf: a flow-based autoregressive model for molecular graph generation. *arXiv preprint arXiv:2001.09382*.

Shuaibi, M., Kolluru, A., Das, A., Grover, A., Sriram, A., Ulissi, Z., and Zitnick, C. L. (2021). Rotation invariant graph neural networks using spin convolutions. *arXiv preprint arXiv:2106.09575*.

Simm, G. N. and Hernández-Lobato, J. M. (2020). A generative model for molecular distance geometry. In III, H. D. and Singh, A., editors, *International Conference on Machine Learning*, volume 119, pages 8949–8958. PMLR.

Simm, G. N. C., Pinsler, R., Csányi, G., and Hernández-Lobato, J. M. (2021). Symmetry-aware actor-critic for 3d molecular design. In *International Conference on Learning Representations*.

Smith, D. G. A., Burns, L., Simmonett, A., Parrish, R., Schieber, M. C., Galvelis, R., Kraus, P., Kruse, H., Remigio, R. D., Alenaizan, A., James, A. M., Lehtola, S., Misiewicz, J. P., et al. (2020). Psi4 1.4: Open-source software for high-throughput quantum chemistry. *The Journal of chemical physics*.

Smith, J. S., Isayev, O., and Roitberg, A. E. (2017). Ani-1: an extensible neural network potential with dft accuracy at force field computational cost. *Chemical science*, 8(4):3192–3203.

Sohl-Dickstein, J., Weiss, E. A., Maheswaranathan, N., and Ganguli, S. (2015). Deep unsupervised learning using nonequilibrium thermodynamics. *arXiv preprint arXiv:1503.03585*.

Song, J., Meng, C., and Ermon, S. (2020a). Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*.

Song, Y. and Ermon, S. (2019). Generative modeling by estimating gradients of the data distribution. In *Advances in Neural Information Processing Systems*, volume 32, pages 11918–11930. Curran Associates, Inc.

Song, Y. and Ermon, S. (2020). Improved techniques for training score-based generative models. *NeurIPS*.

Song, Y., Ye, Q., Xu, M., and Liu, T.-Y. (2020b). Discriminator contrastive divergence: Semi-amortized generative modeling by exploring energy of the discriminator. *arXiv preprint arXiv:2004.01704*.

Thomas, N., Smidt, T., Kearnes, S. M., Yang, L., Li, L., Kohlhoff, K., and Riley, P. (2018). Tensor field networks: Rotation- and translation-equivariant neural networks for 3d point clouds. *ArXiv*.

Vahdat, A., Kreis, K., and Kautz, J. (2021). Score-based generative modeling in latent space. In *Neural Information Processing Systems (NeurIPS)*.

Weiler, M., Geiger, M., Welling, M., Boomsma, W., and Cohen, T. (2018). 3d steerable cnns: Learning rotationally equivariant features in volumetric data. In *NeurIPS*.

Xie, J., Lu, Y., Zhu, S.-C., and Wu, Y. (2016). A theory of generative convnet. In *International Conference on Machine Learning*, pages 2635–2644.

Xu, M., Luo, S., Bengio, Y., Peng, J., and Tang, J. (2021a). Learning neural generative dynamics for molecular conformation generation. In *International Conference on Learning Representations*.

Xu, M., Wang, W., Luo, S., Shi, C., Bengio, Y., Gomez-Bombarelli, R., and Tang, J. (2021b). An end-to-end framework for molecular conformation generation via bilevel programming. *arXiv preprint arXiv:2105.07246*.

You, J., Liu, B., Ying, Z., Pande, V., and Leskovec, J. (2018). Graph convolutional policy network for goal-directed molecular graph generation. In *Advances in neural information processing systems*, pages 6410–6421.

Zhang, L., Han, J., Wang, H., Car, R., and E, W. (2018). Deep Potential Molecular Dynamics: A Scalable Model with the Accuracy of Quantum Mechanics. *Physical Review Letters*, 120(14):143001.

Zhu, J., Xia, Y., Liu, C., Wu, L., Xie, S., Wang, T., Wang, Y., Zhou, W., Qin, T., Li, H., et al. (2022). Direct molecular conformation generation. *arXiv preprint arXiv:2202.01356*.