Regularized Molecular Conformation Fields

Anonymous Author(s)
Affiliation
Address
email

Abstract

Predicting energetically favorable 3-dimensional conformations of organic molecules from molecular graph plays a fundamental role in computer-aided drug discovery research. However, effectively exploring the high-dimensional conformation space to identify (meta)stable conformers is anything but trivial. In this work, we introduce RMCF, a novel framework to generate a diverse set of low-energy molecular conformations through sampling from a regularized molecular conformation field. We develop a data-driven molecular segmentation algorithm to automatically partition each molecule into several structural building blocks to reduce the modeling degrees of freedom. Then, we employ a Markov Random Field to learn the joint probability distribution of fragment configurations and inter-fragment dihedral angles, which enables us to sample from different low-energy regions of a conformation space. Our model constantly outperforms state-of-the-art models for the conformation generation task on the GEOM-Drugs dataset. We attribute the success of RMCF to modeling in a regularized feature space and learning a global fragment configuration distribution for effective sampling. The proposed method could be generalized to deal with larger biomolecular systems.

1 Introduction

The spatial arrangement of atoms within a molecule, also known as molecular conformation, determines the molecular physico-chemical properties, which plays an essential role in downstream computer-aided drug discovery tasks. However, the high-dimensional conformation space spanned by all atomic degrees of freedom (DoF) makes it a great challenge to identify the local minima of the associated *potential energy surface* (PES) of a molecule. Recently, many machine learning (ML) models have achieved remarkable success in molecular conformation generation tasks [Mansimov et al., 2019, Simm and Hernandez-Lobato, 2020, Xu et al., 2021a, Ganea et al., 2021, Xu et al., 2022], exhibiting orders of magnitude faster conformation prediction speed than traditional computational simulation approaches. This makes ML generative models a powerful tool for high-throughput screening in drug discovery, especially large molecules.

To fully exploit the power of ML in molecular conformation generation, we need to tackle a few key challenges. First, a molecule exhibits invariance under SE(3) transformations (i.e., translation and rotation) in 3-dimensional (3D) Euclidean space. In other words, each molecular conformation is uniquely defined up to rigid motion. Hence a single molecule can adopt an infinite set of possible poses. Second, molecules exhibit a variety of dynamics under ambient conditions (e.g., bond rotation, vibration, etc.), leading to a possibly complex PES landscape of high dimension. This makes it particularly challenging for ML models to identify local minima within this space to generate energetically favorable conformations effectively.

Some recently developed ML models tried to solve these problems from different perspectives. For instance, GraphDG [Simm and Hernandez-Lobato, 2020] and CGCF [Xu et al., 2021a] used distance geometry as invariant features for molecular representation learning. The drawbacks of

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

such representations are that these invariant variables are potentially redundant and exhibit mutual dependency, which may cause numerical instability during training. One can also design equivariant networks to hard-code the invariance/equivariance condition into the model [Satorras et al., 2021, Xu et al., 2022]. These models circumvent using intermediate invariant features, but instead directly learn from the spatial arrangement of atoms. However, some studies [Cohen et al., 2018, Li et al., 2021] have suggested that the specialized equivariant layers may cause some loss in the expressiveness of the neural network. On the other hand, GeoMol [Ganea et al., 2021] predicts a local structure for each atom, followed by assembling these atom-based building blocks to form the molecular conformation. These models have difficulty dealing with cyclic graphs (e.g., benzyl group) and may lead to unreasonable conformation predictions.

In the meantime, we realize that the high-dimensional PES could be described in a reduced basis by considering only a few significant DoF that contribute to the low-energy conformations. Specifically, consider the well-known example of the potential energy diagram of ethane molecule as shown in Figure 1. While the molecule exhibits a total of 18 DoF, we could capture the most significant dynamics using a single variable, i.e., the rotation w.r.t. the carbon-carbon bond, quantified by the H-C-C-H dihedral angle. Other DoF in the ethane molecule, e.g., the stretching and bending of other bonds, do not have a significant impact on the potential energy landscape, such that they can be treated as small perturbations to the molecular conformation within a potential well. In other words, to serve the purpose of effectively sampling local minima for low-energy conformations, we could reduce the dimension of the conformation space by tracking only a few significant DoF, as long as they adequately shape the corresponding energy landscape (e.g., the "ω" shape in Figure 1). It then follows naturally to generate new conformers by sampling from the low-dimensional conformation space, which is also physically meaningful. In fact, a similar approach has been widely adopted by the molecular docking community, i.e., incremental construction [Meng et al., 2011], where a ligand molecule is partitioned into multiple fragments connected by rotatable covalent bonds. One may then gradually add ligand fragments with appropriate orientations to fit the ligand into the receptor pocket.

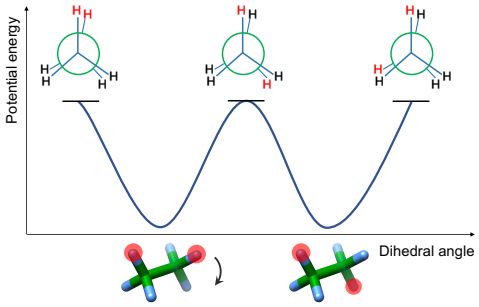

Figure 1: Schematic of the potential energy diagram of an ethane molecule. The upper panel shows the Newman projection of three degenerate eclipsed conformations, the lower panel shows two energetically-favorable staggered conformations. The H-C-C-H dihedral angle alone is sufficient to describe the potential energy change.

This paper presents Regularized Molecular Conformation Fields (RMCF), a novel framework for 3D molecular conformation generation. The novelty of this work lies in (1) developing a data-driven molecular segmentation algorithm to partition each molecule into fragments with low internal flexibility, which serves as a regularization term for our framework, and (2) employing a Markov Random Field (MRF [Murphy, 2012, Cotta et al., 2020]) to capture the joint probability distribution of fragment configurations and inter-fragment dihedral angles. Our work is partly inspired by the Ising model [Cipra, 1987] for simulating quantum spin systems, where we make an analogy between the spin state (i.e., spin up or down) and the configuration of each molecular fragment (e.g., chair or boat conformation of cyclohexane). The introduction of molecular fragments effectively reduces the dimension of the conformation space, where we only keep the most significant components (i.e., fragment configuration and inter-fragment dihedral angle) for conformation generation. This molecular segmentation serves as a regularization term for our framework, which is data-driven and has the effect of reducing feature dimensions. Then, we employ MRF as a generalized setting of the Ising model (with more spin states and more coupling terms) to capture the relationship between adjacent fragments and model the uncertainty of conformations. By estimating the parameters of the MRF, we can obtain a potential energy surface associated with different conformations. Therefore, we can later sample from the low-energy region of PES, which is implicitly learned by the MRF, to obtain energetically favorable molecular conformations.

We demonstrate the effectiveness of RMCF using the GEOM-QM9 and GEOM-Drugs dataset[Axelrod and Gomez-Bombarelli, 2022], where results show that RMCF significantly outperforms state-of-the-art ML models. Specifically, our model can generate a diverse set of low-energy conformations

located at distinct local minima of the underlying PES. We attribute the success of our model to the automatic construction of low-dimensional conformation space and learning a joint probability distribution using MRF to achieve effective sampling of new conformations. We argue that capturing the governing dynamics of molecular systems could significantly improve the performance of learning models for generative purposes.

2 Related Work

Recently, various machine learning models have been proposed for molecular conformation generation. CVGAE [Mansimov et al., 2019] first used a variational autoencoder (VAE) model to generate conformation with atomic coordinates. The model suffers from multi-modality problems due to invariance under SE(3) transformations. Some studies tried to address this problem using two main types of approaches. The first type of approach use intermediate invariant features, such as atomic distance, to encode the system, then leverage geometric algorithms to solve a set of atomic coordinates that match the invariant quantities. For example, GraphDG [Simm and Hernandez-Lobato, 2020] and CGCF [Xu et al., 2021a] proposed to predict the distance matrix by VAE and Flow, respectively, and solve the geometry through the Distance Geometry (DG) method [Liberti et al., 2014]. However, these invariant features are potentially redundant and could exhibit mutual dependency, which makes these methods numerically unstable and could predict unreasonable conformations. Some studies suggest that inconsistencies between training and test are responsible for the poor performance of the models [Shi et al., 2021, Xu et al., 2021b]. ConfGF [Shi et al., 2021] passed the gradient of loss to coordinates and ConfVAE [Xu et al., 2021b] used a bilevel optimization to alleviate the inconsistencies. GeoMol [Ganea et al., 2021] defined another set of invariants, which are bond lengths, bond angles, and dihedral angles. Zhu et al. [2022] proposed a model invariant to roto-translation of coordinates of conformations and permutation of symmetric atoms in molecules. Geomol generated multiple conformations by adding noise to the input, which did not learn a global energy function from which to sample the low-energy conformations.

Another type of approach applies equivariant networks or kernels, such that after the input is rotated and translated, the output will be transformed accordingly. Satorras et al. [2021] proposed an equivariant normalizing flow, E(n)-flow, and Xu et al. [2022] proposed GeoDiff, which is a diffusion model [Song and Ermon, 2019] with equivariant Markov kernels. Guan et al. [2021] proposed a variant of SE(3)-equivariant neural networks to learn the gradient fields of an implicit conformational energy landscape. Equivariant methods circumvent intermediate invariants and model the coordinates directly, but require the design of specialized equivariant layers, which some studies have suggested would lose the expressive power of the network [Li et al., 2021].

In addtion, some software for conformation generation are widely used in biochemical research. For instance, RDKit [Riniker and Landrum, 2015] is a popular open-source software which generates conformations using ETKDG distance geometry, and OMEGA is a commercial software which assembles the fragments with knowledge-based rules to generate conformations.

3 RMCF: Generating in Regularized Conformation Space

In this section, we present the regularized molecular conformation field (RMCF) in detail. Intuitively, RMCF partitions the original molecular graph into multiple molecular fragments, where adjacent fragments in the graph are connected with associated dihedral angles. In such a case, the molecular conformation generation problem can be decomposed into a) generating the conformation of molecular fragments, and b) determining the specific dihedral angles between connected fragments. The RMCF learns the potential energy landscape of molecular conformations by modeling a joint probability distribution of fragment conformation and dihedral angles, from which we can perform inference to draw diverse samples.

3.1 Overview of RMCF

We start with an overview of the our proposed RMCF model. Generally, as shown in Figure 2, given a molecular graph G, to obtain a generated 3D conformation R, RMCF mainly takes three steps:

$$G \xrightarrow{1} \mathcal{G} \xrightarrow{2} \mathcal{X} \xrightarrow{3} R$$

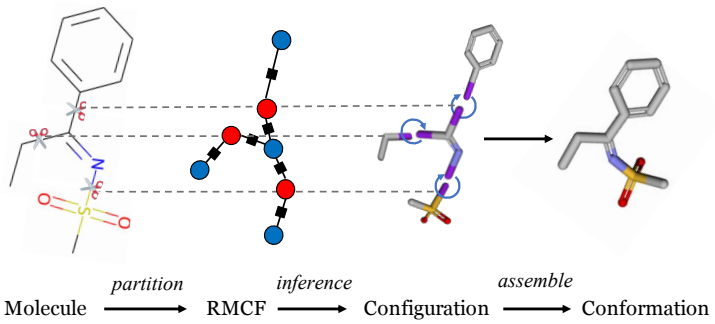

Figure 2: The workflow of RMCF. Starting from a 2D molecular graph, we partition the molecule into fragments with least intra-fragment DoF. Blue and red circles denote fragment and dihedral angle configurations, respectively, while the black squares denotes the interaction between neighboring configurations. We then use MRF to model the joint probability distribution of fragment and dihedral configurations. The last step is to assemble the predicted molecular conformation according to the predicted dihedral angles and fragment conformations.

1. $G \longrightarrow \mathcal{G}$ (Section 3.2): The main idea of our proposed model is to build a conformation distribution within a regularized molecular conformation space, which is achieved by partitioning the molecular graph G into multiple molecular fragments F connected by dihedral angles D. Notably, the graph partition should satisfy the least intra-fragment DoF principle (introduced in Section 3.2), which is crucial for reducing the dimension of the molecular conformation space.

   The resulting graph after partitioning (including F and D) could be jointly modeled with a Markov random field (also called undirected graphical model). Since it is regularized with reduced total DoF, we name our model Regularized Molecular Conformation Fields.

   Formally, a regularized conformation field $\mathcal{G} = (F, D, E)$ is an undirected graph formed by a collection of fragment vertices $F = (f_1, f_2, \cdots, f_{N_F})$, a collection of dihedral vertices $D = (d_1, d_2, \cdots, d_{N_D})$, and a collection of edges $E = (e_1, e_2, \cdots, e_{N_E}) \subset F \times D$ (edges between fragments and dihedral angles in the graph), where N_F, N_D, N_E are the size of F, D, E, respectively [Wainwright et al., 2008]. Each edge consists of a pair of vertices $f \in F$ and $d \in D$. We associate with each vertex $f \in F$ a random variable X_f taking values from a set of fragment conformations $V(f)$ (see Section 3.2), each vertex $d \in D$ a random variable X_d, representing the value of dihedral angles.

2. $\mathcal{G} \longrightarrow \mathcal{X}$ (Section 3.3 and 3.4): Given $\mathcal{G}$, the conformation generation problem turns into generating the conformation of molecular fragments and determining the dihedral angles between them, namely inferring a configuration $\mathcal{X}$ from $\mathcal{G}$. A configuration $\mathcal{X} = (\mathcal{X}_f, \mathcal{X}_d)$ of the RMCF contains a set of fragment conformation $\mathcal{X}_f = \{x_{f_i} | i = 1, 2, \cdots, N_F\}$ and a set of dihedral angle values $\mathcal{X}_d = \{x_{d_i} | i = 1, 2, \cdots, N_D\}$, which could be assembled into the 3D conformation of a molecule. $\mathcal{X}$ could be obtained by maximizing the probability modeled by RMCF:

$$P(x_{f_1}, \cdots, x_{f_{N_F}}, x_{d_1}, \cdots, x_{d_{N_d}}) = \frac{1}{Z} \left\{ \prod_{f_i \in F} \psi(x_{f_i}) \cdot \prod_{d_i \in D} \psi(x_{d_i}) \cdot \prod_{(f_i, d_j) \in E} \psi(x_{f_i}, x_{d_j}) \right\} \quad (1)$$

   where Z is the normalizing factor, ψ is the compatibility functions, which will be further described in Section 3.3.

3. $\mathcal{X} \longrightarrow R$ (Section 3.5): Given the configuration $\mathcal{X}$, we assemble the predicted fragment conformation and dihedral angle values into the corresponding molecular conformation R.

3.2 Graph Construction with Least DoF Principle

In this section, we elaborate more about the molecular graph partition with least DoF principle used in our proposed RMCF. This is a crucial step to our model since it builds the foundation of the following probabilistic modeling. The entire graph construction process has three main components, namely a) constructing the fragment library with BRICS, b) obtaining fragment conformation vocabulary with clustering, and c) yielding final fragment partition by minimizing the intra-fragement DoF.

Specifically,We use BRICS, a decomposition method based on molecular functional groups, to pre-cut all molecular graphs and their 3D conformations and collect each fragment conformation into a set $\mathcal{C}(f)$[1]. A sufficient number (i.e., 1,000) of fragment conformations r are sampled from $\mathcal{C}(f)$ for the following clustering steps.

Then, we use the K-Medoids clustering algorithm and use the root-mean-square deviation (RMSD) of two fragment conformations as the distance metric between two fragments, which takes rotation and translation invariance into consideration. We adjust the number of cluster centers k from 1 to 10 and use the Silhouette Coefficient[Rousseeuw, 1987] to measure the goodness of these clusters and choose the optimal hyper-parameter k with clusters having the highest Silhouette Coefficient value. Cluster centroid elements are collected into conformation vocabulary $V(f)$.

Although BRICS presents a solution to partition the molecular graph, it cannot guarantee a small DoF within the fragments. Therefore, we define a metric to quantify the *intra*-fragment DoF and design an algorithm to re-fragment the molecules to minimize this quantity. We define the fragment DoF as the maximum RMSD between all pairs of fragment conformations in $V(f)$, and the total DoF of a partitioned molecule as the mean of its fragment DoF. The best partition strategy $\mathcal{P}^*$ should have the least DoF:

$$\mathcal{P}^* = \arg\min_{\mathcal{P}} \frac{1}{|\mathcal{P}|} \sum_{f \in \mathcal{P}} \mathrm{DoF}(f) = \arg\min_{\mathcal{P}} \frac{1}{|\mathcal{P}|} \sum_{f \in \mathcal{P}} \max_{r_i, r_j \in V(f)} \mathrm{RMSD}(r_i, r_j) \tag{2}$$

We could employ graph dynamic programming to search for the optimal solution and the detailed algorithm is in Appendix D.

After partition, we can get the fragment set $F = (f_1, f_2, \cdots, f_{N_F})$ and insert the dihedral vertices between connected fragments to establish the RMCF $\mathcal{G}$ on the molecular graph G as shown in the Figure 1

3.3 Approximated Training of RMCF

Training Objective Training in RMCF is intractable, especially when it contains loops [Murphy, 2012]. It is difficult to estimate the normalization factor Z in the objective (Eq. 1),which means we have to marginalize all possible conformations. We approximate the likelihood through *piece-wise training*, which has been proven efficient in training graphical models [Sutton and McCallum, 2009, 2012, Lin et al., 2016, Qu et al., 2022]. The key idea is to train each node and edge independently.

$$\begin{aligned}\mathcal{L} &= -\log P(x_{f_1}, \cdots, x_{f_{N_F}}, x_{d_1}, \cdots, x_{d_{N_d}}) \\ &\approx \sum_{f_i \in F} \underbrace{\boxed{-\log \frac{\psi(x_{f_i})}{Z_f}}}_{\triangleq \mathcal{L}_n(f)} + \sum_{d_i \in D} \underbrace{\boxed{-\log \frac{\psi(x_{d_i})}{Z_d}}}_{\triangleq \mathcal{L}_n(d)} + \sum_{(f_i, d_j) \in E} \underbrace{\boxed{-\log \frac{\psi(x_{f_i}, x_{d_j})}{Z_e}}}_{\triangleq \mathcal{L}_e(f,d)}\end{aligned} \tag{3}$$

where $\mathcal{L}_n$ and $\mathcal{L}_e$ denote the node-wise and edge-wise negative log-likelihood. Especially, $\mathcal{L}_e(f, d)$ measures *how probable a fragment f and a dihedral d appear at the same time*.

More concretely, for a pair of nodes in the field $e = (f, d) \in \mathcal{G}$, let $|f|$ and $|g|$ denote the cardinality of the state space, $|\mathbf{h}|$ denotes the hidden dimension, y_f and y_d are the ground truth of fragment conformation f and dihedral angle value d , which will introduced calculation in the next paragraph. We calculate the representations from the graph neural networks: Let $\mathbf{h}_f^0 = \mathrm{emb}(f)$ and $\mathbf{h}_d^0 = \mathrm{emb}(d)$ denote the fragment embedding and the dihedral embedding (which is a "dummy" embedding) as the network input, we compute the contextualized fragment representations $\mathbf{h}_f^{\ell+1}$ and dihedral representations $\mathbf{h}_d^{\ell+1}$ by:

$$\mathbf{h}_f^{\ell+1} = g\left(\mathbf{h}_f^{\ell}, \left\{\mathbf{h}_d^{\ell}\right\}_{d \in \mathcal{N}_f}\right), \quad \mathbf{h}_d^{\ell+1} = g\left(\mathbf{h}_d^{\ell}, \left\{\mathbf{h}_f^{\ell}\right\}_{f \in \mathcal{N}_d}\right) \tag{4}$$

where ℓ denotes the layer index, g denotes the massage-passing function of GNN, $\mathcal{N}_f$ and $\mathcal{N}_d$ denotes the neighborhoods of fragment and dihedral respectively. The node-wise negative log-likelihood is defined as:

$$\mathcal{L}_n(f) = -\log \mathrm{softmax}\left(\mathrm{out}(\mathbf{h}_f)\right)[y_f], \quad \mathcal{L}_n(d) = -\log \mathrm{softmax}\left(\mathrm{out}(\mathbf{h}_d)\right)[y_d] \tag{5}$$

[1]On the base of BRICS, we further cut the links of rings and side chains to prevent the combinatorial explosion

where $\mathrm{out}(\cdot) : \mathbb{R}^{|\mathbf{h}|} \to \mathbb{R}^{|f|}$ is an operator parameterized by a linear layer. The edge-wise negative log-likelihood is defined as:

$$\mathcal{L}_e(f, d) = -\log \mathrm{softmax}\left(\mathbf{E}_f \mathbf{W}_e \mathbf{E}_d^{\mathrm{T}}\right)[y_f, y_d] \tag{6}$$

where $\mathbf{E}_f \in \mathbb{R}^{|f| \times |\mathbf{h}|}$ and $\mathbf{E}_d \in \mathbb{R}^{|d| \times |\mathbf{h}|}$ are the representations of a set of node states, $\mathbf{W}_e \in \mathbb{R}^{|\mathbf{h}| \times |\mathbf{h}|}$ is a matrix carrying the global information between two nodes, parameterized by a neural network $\mathbf{W}_e = \mathrm{MLP}(\mathbf{h}_f, \mathbf{h}_d)$ [Sun et al., 2019].

Obtaining ground truth The oracle configuration $\mathcal{Y} = (\mathcal{Y}_f, \mathcal{Y}_d)$ contains a set of ground truth fragment conformations indexes $\mathcal{Y}_f = \{y_{f_i} | i = 1, 2, \cdots, N_F\}$ and a set of dihedral angle indexes $\mathcal{Y}_d = \{y_{d_i} | i = 1, 2, \cdots, N_D\}$ denote the choices of fragments conformations and dihedral angles, respectively. For a given fragment f, we pick the configuration s whose corresponding fragment conformation r_s in vocabulary $V(f)$, has the minimal RMSD to the actual fragment conformation r, as the ground truth y^f.

$$y_f = \arg\min_s \mathrm{RMSD}(r, r_s), r_s \in V(f) \tag{7}$$

The dihedral angle φ is defined as the angle between half-planes of two connected 3D fragments. Since the dihedral angle depends on the set of the atoms involved in the calculation, for the same type of fragments, we should keep the same set of atoms to calculate.

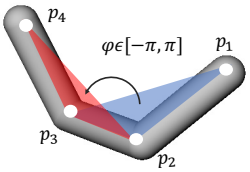

Figure 3: Dihedral angle

For a system where two fragments contain four consecutively-bonded atoms, two half-planes intersect on a rotatable bond. The angle between them is the *dihedral angle*. If the connected points are sequentially numbered and located at positions $\mathbf{p}_1$, $\mathbf{p}_2$, $\mathbf{p}_3$, $\mathbf{p}_4$ and the corresponding bond vectors are defined as $\mathbf{u}_1 = \mathbf{p}_2 - \mathbf{p}_1$,$\mathbf{u}_2 = \mathbf{p}_3 - \mathbf{p}_2$,$\mathbf{u}_3 = \mathbf{p}_4 - \mathbf{p}_3$. Then we have:

$$\varphi(\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3) = \mathrm{atan2}\left(|\mathbf{u}_2| \, \mathbf{u}_1 \cdot (\mathbf{u}_2 \times \mathbf{u}_3), (\mathbf{u}_1 \times \mathbf{u}_2) \cdot (\mathbf{u}_2 \times \mathbf{u}_3)\right) \tag{8}$$

In practice, we quantize the dihedral angles φ into a number of evenly divided bins which has interval length L to discretize the continuous angle values

$$y_d = \lfloor \frac{180\varphi}{\pi L} \rfloor \tag{9}$$

3.4 Inference & Sampling

In this section, we give a brief overview of how we perform inference and sampling in RMCF.

Inference If a RMCF is acyclic (linear- or tree-structured), the max-a-posterior decoding can be done via dynamic programming [Forney, 1973]. Cases are a bit complicated when loops occur in the RMCF. We adopt a more generalized algorithm, namely loopy belief propagation (LBP) [Murphy et al., 2013], to overcome such obstacles. We refer the reader to textbooks for more details about the LBP algorithm [Murphy, 2012]. It is worth noting that LBP is used only during inference time. We do not perform any inference during training due to the issue of training stability [Qu et al., 2022].

Sampling Markov Chain Monte Carlo is popular in sampling from graphical models [Andrieu et al., 2003]. We employ a simple variant of it, i.e., Gibbs sampling [George and McCulloch, 1993], to draw samples from RMCF. Specifically, we resample each fragment or dihedral individually, keeping all other nodes fixed in each iteration [Murphy, 2012]. The detailed algorithm can be found in Appendix E.

Clustering Once the Markov chain achieves detailed balance, we randomly draw $10,000$ samples from it. However, these samples can be governed by a few conformations, which means most of them are similar. We measure the distance between two configurations $\mathcal{X}^{(1)}$ and $\mathcal{X}^{(2)}$ by:

$$\mathrm{d}(\mathcal{X}^{(1)}, \mathcal{X}^{(2)}) = \|\mathcal{X}_d^{(1)} - \mathcal{X}_d^{(2)}\|_2^2 + \mathbb{H}(\mathcal{X}_f^{(1)}, \mathcal{X}_f^{(2)}) \tag{10}$$

where $\|\cdot\|_2$ is the Euclidean norm measuring the difference between dihedrals, $\mathbb{H}$ is the Hamming distance [Hamming, 1950] counting the number of different fragment configurations. We further run K-means clustering [MacQueen et al., 1967] based on the pair-wise distance to partition all samples into a fixed number of clusters.

3.5 Assembling

Once we have predicted the dihedral angle $\hat{x}_d$ between two fragments, we need to compute the transformation matrix $\mathbf{T}$ to align the two fragments in arbitrary position and make their dihedral angle equal to the given value φ and repeat these steps until the whole conformation is assembled.

Given two fragments with six points, located at $\mathbf{p}_1$, $\mathbf{p}_2$, $\mathbf{p}_3$, $\mathbf{p}_4$, $\mathbf{p}_5$, $\mathbf{p}_6$, the bond vectors are $\mathbf{u}_1 = \mathbf{p}_2 - \mathbf{p}_1$, $\mathbf{u}_2 = \mathbf{p}_3 - \mathbf{p}_2$, $\mathbf{v}_1 = \mathbf{p}_5 - \mathbf{p}_4$, $\mathbf{v}_2 = \mathbf{p}_6 - \mathbf{p}_5$. $\mathbf{u}_2$ and $\mathbf{v}_1$ are the same bond after assembling, so we have $|\mathbf{u}_2| = |\mathbf{v}_1|$.

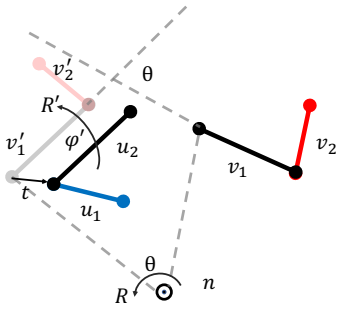

Figure 4: Transformations for Assembling

First we need to rotate the second fragment around some axis so that $\mathbf{u}_2$ and $\mathbf{v}_1$ will be parallel. This axis is the unit normal vector $\mathbf{n}$ towards the plane of $\mathbf{u}_2$ and $\mathbf{v}_1$, and the rotate angle is equal to the cross angle θ.

$$\cos\theta = \frac{\mathbf{u}_2 \cdot \mathbf{v}_1}{|\mathbf{u}_2|\,|\mathbf{v}_1|}, \quad \mathbf{n} = \frac{\mathbf{u}_2 \times \mathbf{v}_1}{|\mathbf{u}_2 \times \mathbf{v}_1|} \tag{11}$$

By using the Rodrigues' rotation formula, we can obtain the rotation matrix with unit normal vector $\mathbf{n}$ and rotation angle θ. Next, we rotate the second fragment around $\mathbf{u}_2$ to match with the target dihedral angle. Finally we align the anchor points by calculating the final transformation matrix $\mathbf{T}$ as a composite matrix of two rotations and one translation.

$$\mathbf{R}(\mathbf{n},\theta) = I\cos\theta + (1-\cos\theta)\begin{pmatrix} n_x \\ n_y \\ n_z \end{pmatrix}(n_x, n_y, n_z) + \sin\theta \begin{pmatrix} 0 & -n_z & n_y \\ n_z & 0 & -n_x \\ -n_y & n_x & 0 \end{pmatrix} \tag{12}$$

$$\mathbf{R}' = \mathbf{R}(\mathbf{u}_2, \frac{\pi L \hat{x}_d}{180} - \varphi(\mathbf{u}_1, \mathbf{u}_2, \mathbf{R}(n,\theta)\mathbf{v}_2)), \quad \mathbf{t} = (\mathbf{p}_1 - \mathbf{R}'\mathbf{R}\mathbf{p}_4)^{\mathbf{T}} \tag{13}$$

$$\mathbf{T} = \begin{pmatrix} \mathbf{R}'\mathbf{R} & \mathbf{t} \\ \mathbf{0}^{\mathrm{T}} & 1 \end{pmatrix} \tag{14}$$

4 Experiment

We now demonstrate the effectiveness of RMCF on the conformation generation task for drug-like molecules.

4.1 Dataset and Baseline

Following previous work on conformation generation, we benchmark our model performance using the GEOM-QM9 and GEOM-Drugs dataset, which contains small and mid-sized organic molecules with high quality conformations. Considering the small size of molecules in QM9, we put the model performance in Appendix 4, and mainly focus on discussing GEOM-Drugs results hereafter. We use the same test set as that in GeoDiff, where the remaining molecules are used for training and validation with a 9:1 ratio. The final training/validation/test set contains 271,539/30,171/1,034 molecules, respectively.

Our molecular segmentation algorithm eventually returned a vocabulary of 9,081 types of 2D fragments and 30,408 types of 3D fragments for the GEOM-Drugs dataset. As for the angular discretization of dihedral angles, the 360 degree interval is evenly divided into 72 bins. We adopt the Message-Passing Neural Network (MPNN) [Gilmer et al., 2017] as the framework for implementing our graph neural network. Other implementation details are provided in Appendix A.

4.2 Evaluation Metrics

To quantitatively measure the quality and diversity of generated molecular conformations, we use the same evaluation metrics as defined in Ganea et al. [2021] and Xu et al. [2022]. Let S_g and S_r

denote the set of generated and reference conformations, respectively. The coverage score (COV) and matching score (MAT) following the conventional Recall measurement can be defined as:

$$\begin{aligned} \text{COV-R}\left(S_g, S_r\right) &= \frac{1}{|S_r|}\left|\left\{\mathcal{C} \in S_r \mid \text{RMSD}(\mathcal{C}, \hat{\mathcal{C}}) \leq \delta, \hat{\mathcal{C}} \in S_g\right\}\right| \\ \text{MAT-R}\left(S_g, S_r\right) &= \frac{1}{|S_r|}\sum_{\mathcal{C} \in S_r} \min_{\hat{\mathcal{C}} \in S_g} \text{RMSD}(\mathcal{C}, \hat{\mathcal{C}}) \end{aligned} \tag{15}$$

where the threshold δ in coverage score is set as 1.25Å for the GEOM-Drugs dataset in our work. The corresponding prediction precision metrics, i.e., COV-P and MAT-P, are defined in a similar manner, where the set of generated and reference conformations are swapped in the above definition. We set S_g as twice the size of S_r for each molecule for fair comparison with previous work [Ganea et al., 2021, Xu et al., 2022]. The precision-associated metrics focus more on generating accurate conformations that match those in the reference dataset, while the recall-associated metrics emphasize the structural diversity of generated conformations.

Table 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	3.070	2.994	-	-	-	-
GraphDG	8.27	0.00	1.972	1.985	2.08	0.00	2.434	2.410
CGCF	53.96	57.06	1.249	1.225	21.68	13.72	1.857	1.807
ConfVAE	55.20	59.43	1.238	1.142	22.96	14.05	1.829	1.816
GeoMol	82.43	95.10	0.862	0.837	78.52	94.40	0.933	0.856
ConfGF	62.15	70.93	1.163	1.160	23.42	15.52	1.722	1.686
GeoDiff	89.13	97.88	0.863	0.853	61.47	64.55	1.171	1.123
Direct	**96.69**	**100.0**	**0.722**	0.724	-	-	-	-
RDKit	60.19	64.28	1.219	1.133	69.23	87.63	1.113	0.963
OMEGA	81.64	97.25	0.851	0.771	77.18	96.15	0.951	0.854
RMCF-R	82.25	90.77	0.839	0.789	**83.02**	**98.50**	**0.812**	**0.722**
RMCF-C	87.12	96.26	0.749	**0.709**	82.01	95.91	0.835	0.754

4.3 Results and Discussions

The main results are presented in Table 1. We generated conformations by sampling on the RMCF with two strategies: (1) directly sample a specific number of conformations (i.e., S_g), and (2) first generate a sufficient number of conformations (we generate 10,000 conformations in this work), then cluster them into S_g clusters and sample one conformation from each cluster. The model performance associated with the above two strategies are named RMCF-R and RMCF-C, respectively. As shown in Table 1, RMCF significantly outperforms all other models in all metrics except for the COV-R metric, where our model performance is comparable to that of GeoDiff. In particular, for COV-P and MAT-R metrics, our model shows a significant performance gain compared to others, which means RMCF can generate more accurate and high quality conformations. In addition, we find that the clustering-after-sampling strategy leads to a boost in the structural diversity, while keeping a good quality (i.e., energetically favorable conformations) of the generated conformations.

Since we employ a discretized treatment for the continuous dihedral angle as well as the fragment 3D conformation, the model performance will inevitably have some discrepancy between the predicted and ground truth conformations. To that end, we investigate the upper and lower bound of our model performance to have a better understanding of its predicting power, as shown in Table 2. Specifically, we discretize the ground truth conformations using the molecular segmentation algorithm to obtain the fragment configurations ("gold $\mathcal{X}_f$") and inter-fragment dihedral angles ("gold $\mathcal{X}_d$"), then evaluate the corresponding metrics. On the other hand, we randomly sample some $\mathcal{X}_f$ and $\mathcal{X}_d$, whose evaluation results should indicate the performance lower bound using our fragment representation. Surprisingly, we find that although using randomly sampled $\mathcal{X}_f$ and $\mathcal{X}_d$, we can still outperform some previous models. This is an advantage of using molecular fragment as building blocks for conformation generation, since we bypass the need to generate many insignificant variables which may degrade

the model performance. We also learn that having accurate dihedral angle predictions (i.e., $\mathcal{X}_d$) is more significant than good fragment configuration predictions (i.e., $\mathcal{X}_f$). These experimental results support our claim that capturing a few significant DoF within a molecule is adequate for generating a diverse set of low-energy conformations, while other DoF could be safely ignored during modeling.

Table 2: The empirical upper and lower bound of RMCF performance on the **GEOM-Drugs** dataset.

RMCF Settings	COV-R (%) ↑		MAT-R (Å) ↓		COV-P (%) ↑		MAT-P (Å) ↓	
	Mean	Median	Mean	Median	Mean	Median	Mean	Median
gold $\mathcal{X}_f$,gold $\mathcal{X}_d$	99.16	100.0	0.302	0.242	99.37	100.0	0.290	0.235
rand $\mathcal{X}_f$,gold $\mathcal{X}_d$	96.74	100.0	0.511	0.464	88.93	100.0	0.634	0.561
gold $\mathcal{X}_f$,rand $\mathcal{X}_d$	66.21	77.42	1.125	1.092	40.30	34.25	1.429	1.397
rand $\mathcal{X}_f$,rand $\mathcal{X}_d$	63.60	72.22	1.156	1.135	37.68	31.25	1.463	1.429

* $\mathcal{X}_f$ denotes the fragment configuration set, $\mathcal{X}_d$ denotes the inter-fragment dihedral angle set
* "gold" refers to the ground truth distribution, and "rand" refers to a randomly sampled distribution.

At last, we showcase the generated conformations of two example molecules in Figure 6. For each molecule, we take the first three predicted conformations, and align the non-rigid parts for visualization purposes. From Figure 6(a), we observe a diverse set of conformations mainly driven by the rotation of two single bonds without causing too much steric effect. For the amide bond regions, the model correctly predicts the corresponding dihedral angles to form planar conjugate systems. Meanwhile, for Figure 6(b) we see a large planar conjugate system on the left side and a cyclooctane ring on the right. Again, the model accurately captures the conjugate system and only makes variations in the cyclooctane conformations. Interestingly, the segmentation algorithm tends to only preserve the cyclic groups (e.g., benzyl and furan groups), and even cuts through the amide bonds and other acyclic conjugate systems. This behavior is a direct consequence of balancing the vocabulary size and the variety of local chemical environment.

We show that our model performs reasonable segmentation of drug-like molecules into small functional groups with limited internal DoF, and could correctly predict planar conjugate systems. Therefore, we believe that our generative process is essentially sampling from the local-minima of the learned potential energy surface by RMCF, where those non-essential DoF which contribute to local structural perturbations are omitted in our modeling framework. We argue that both the data-driven segmentation and MRF modeling of the joint probability distribution are essential for the success of our model.

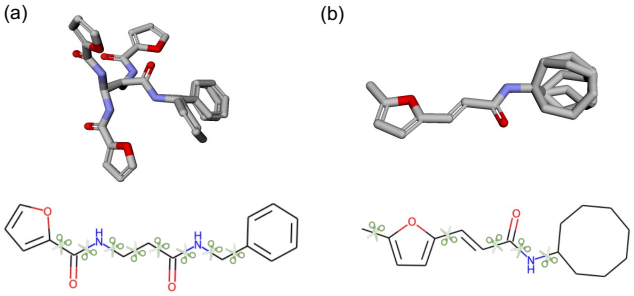

Figure 5: The first three generated conformations of two example molecules. The upper panel shows the 3D atomic arrangement, where the non-rigid fragments are aligned to help visualization. The lower panel shows where the segmentation has been made for each molecule, as indicated by the scissors.

5 Conclusion

We introduce RMCF, a novel framework for 3D molecular conformation generation. Our model is physics-motivated, with the central idea to effectively model the joint probability distribution of governing dynamical modes in a reduced conformation space to achieve energetically favorable conformation generation. Experimental results show that RMCF outperforms state-of-the-art models on the GEOM-Drugs dataset to predict a diverse set of conformations located at distinct local minima of the corresponding molecular potential energy surface. Our methodology can be naturally extended to larger biomolecular systems, e.g., proteins, whose conformation prediction is a significant topic in the biological research community. We will address this challenge in our future work.

References

Elman Mansimov, Omar Mahmood, Seokho Kang, and Kyunghyun Cho. Molecular geometry prediction using a deep generative graph neural network. arXiv preprint arXiv:1904.00314, 2019.

Gregor Simm and Jose Miguel Hernandez-Lobato. A generative model for molecular distance geometry. In Hal Daumé III and Aarti Singh, editors, Proceedings of the 37th International Conference on Machine Learning, volume 119, pages 8949–8958. PMLR, 2020.

Minkai Xu, Shitong Luo, Yoshua Bengio, Jian Peng, and Jian Tang. Learning neural generative dynamics for molecular conformation generation. In International Conference on Learning Representations, 2021a.

Octavian-Eugen Ganea, Lagnajit Pattanaik, Connor W Coley, Regina Barzilay, Klavs F Jensen, William H Green, and Tommi S Jaakkola. Geomol: Torsional geometric generation of molecular 3d conformer ensembles. arXiv preprint arXiv:2106.07802, 2021.

Minkai Xu, Lantao Yu, Yang Song, Chence Shi, Stefano Ermon, and Jian Tang. Geodiff: A geometric diffusion model for molecular conformation generation. arXiv preprint arXiv:2203.02923, 2022.

Victor Garcia Satorras, Emiel Hoogeboom, Fabian B Fuchs, Ingmar Posner, and Max Welling. E (n) equivariant normalizing flows for molecule generation in 3d. arXiv preprint arXiv:2105.09016, 2021.

Taco S Cohen, Mario Geiger, Jonas Köhler, and Max Welling. Spherical cnns. arXiv preprint arXiv:1801.10130, 2018.

Xianzhi Li, Ruihui Li, Guangyong Chen, Chi-Wing Fu, Daniel Cohen-Or, and Pheng-Ann Heng. A rotation-invariant framework for deep point cloud analysis. IEEE Transactions on Visualization and Computer Graphics, 2021.

Xuan-Yu Meng, Hong-Xing Zhang, Mihaly Mezei, and Meng Cui. Molecular docking: a powerful approach for structure-based drug discovery. Current computer-aided drug design, 7(2):146–157, 2011.

Kevin P Murphy. Machine learning: a probabilistic perspective. MIT press, 2012.

Leonardo Cotta, Carlos HC Teixeira, Ananthram Swami, and Bruno Ribeiro. Unsupervised joint k-node graph representations with compositional energy-based models. Advances in Neural Information Processing Systems, 33:17536–17547, 2020.

Barry A Cipra. An introduction to the ising model. The American Mathematical Monthly, 94(10): 937–959, 1987.

Simon Axelrod and Rafael Gomez-Bombarelli. Geom, energy-annotated molecular conformations for property prediction and molecular generation. Scientific Data, 9(1):1–14, 2022.

Leo Liberti, Carlile Lavor, Nelson Maculan, and Antonio Mucherino. Euclidean distance geometry and applications. SIAM review, 56(1):3–69, 2014.

Chence Shi, Shitong Luo, Minkai Xu, and Jian Tang. Learning gradient fields for molecular conformation generation. ArXiv, 2021.

Minkai Xu, Wujie Wang, Shitong Luo, Chence Shi, Yoshua Bengio, Rafael Gomez-Bombarelli, and Jian Tang. An end-to-end framework for molecular conformation generation via bilevel programming. arXiv preprint arXiv:2105.07246, 2021b.

Jinhua Zhu, Yingce Xia, Chang Liu, Lijun Wu, Shufang Xie, Tong Wang, Yusong Wang, Wengang Zhou, Tao Qin, Houqiang Li, et al. Direct molecular conformation generation. arXiv preprint arXiv:2202.01356, 2022.

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

Jiaqi Guan, Wesley Wei Qian, Wei-Ying Ma, Jianzhu Ma, Jian Peng, et al. Energy-inspired molecular conformation optimization. In International Conference on Learning Representations, 2021.

Sereina Riniker and Gregory A. Landrum. Better informed distance geometry: Using what we know to improve conformation generation. Journal of Chemical Information and Modeling, 55(12): 2562–2574, 2015.

Martin J Wainwright, Michael I Jordan, et al. Graphical models, exponential families, and variational inference. Foundations and Trends® in Machine Learning, 1(1–2):1–305, 2008.

Peter J Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. Journal of computational and applied mathematics, 20:53–65, 1987.

Charles Sutton and Andrew McCallum. Piecewise training for structured prediction. Machine learning, 77(2):165–194, 2009.

Charles Sutton and Andrew McCallum. Piecewise training for undirected models. arXiv preprint arXiv:1207.1409, 2012.

Guosheng Lin, Chunhua Shen, Anton Van Den Hengel, and Ian Reid. Efficient piecewise training of deep structured models for semantic segmentation. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 3194–3203, 2016.

Meng Qu, Huiyu Cai, and Jian Tang. Neural structured prediction for inductive node classification. ICLR, 2022.

Zhiqing Sun, Zhuohan Li, Haoqing Wang, Di He, Zi Lin, and Zhihong Deng. Fast structured decoding for sequence models. Advances in Neural Information Processing Systems, 32, 2019.

G David Forney. The viterbi algorithm. Proceedings of the IEEE, 61(3):268–278, 1973.

Kevin Murphy, Yair Weiss, and Michael I Jordan. Loopy belief propagation for approximate inference: An empirical study. arXiv preprint arXiv:1301.6725, 2013.

Christophe Andrieu, Nando De Freitas, Arnaud Doucet, and Michael I Jordan. An introduction to mcmc for machine learning. Machine learning, 50(1):5–43, 2003.

Edward I George and Robert E McCulloch. Variable selection via gibbs sampling. Journal of the American Statistical Association, 88(423):881–889, 1993.

Richard W Hamming. Error detecting and error correcting codes. The Bell system technical journal, 29(2):147–160, 1950.

James MacQueen et al. Some methods for classification and analysis of multivariate observations. In Proceedings of the fifth Berkeley symposium on mathematical statistics and probability, volume 1, pages 281–297. Oakland, CA, USA, 1967.

Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. Neural message passing for quantum chemistry. In Proceedings of the 34th International Conference on Machine Learning-Volume 70, pages 1263–1272. JMLR. org, 2017.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [No]
   (c) Did you discuss any potential negative societal impacts of your work? [No]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...

(a) Did you state the full set of assumptions of all theoretical results? [No] without theoretical results

(b) Did you include complete proofs of all theoretical results? [No] without theoretical results

3. If you ran experiments...

(a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes]

(b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Appendix and Section 4

(c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [No] The results are stable

(d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] See Appendix

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...

(a) If your work uses existing assets, did you cite the creators? [No] No existing assets

(b) Did you mention the license of the assets? [No] No assets

(c) Did you include any new assets either in the supplemental material or as a URL? [No] No assets

(d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [No] Open-source data

(e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [No] The data without personally identifiable information or offensive content

5. If you used crowdsourcing or conducted research with human subjects...

(a) Did you include the full text of instructions given to participants and screenshots, if applicable? [No]

(b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [No]

(c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [No]

A Hyperparamters

Table 3: Hyperparameter choices of RMCF and training phase

Hyperparameters	Values
Message Passing Neural Network	
Number of MPNN Layers N	6
Dimension of Embeddings d_{embed}	320
Dimension of Hiddens d_h	320
Dimension of Feed-forward Layers d_{ff}	1280
Dropout Rate P_{drop}	0.1
Markov Random Fields Layer	
Number of States $\|f\|$ or $\|g\|$	72
Dimension of Hiddens $\mathbb{D}$	320
Training	
Batch Size	256
Learning Rate	5×10^{-4}
Max Training Steps	1.2×10^{6}
Optimizer	Adam
Learning Rate Scheduler	Linear

We conducted our experiments on 8 A100 GPUs and took about 80 hours to train RMCF on GEOM-Drugs.

B Visualizations

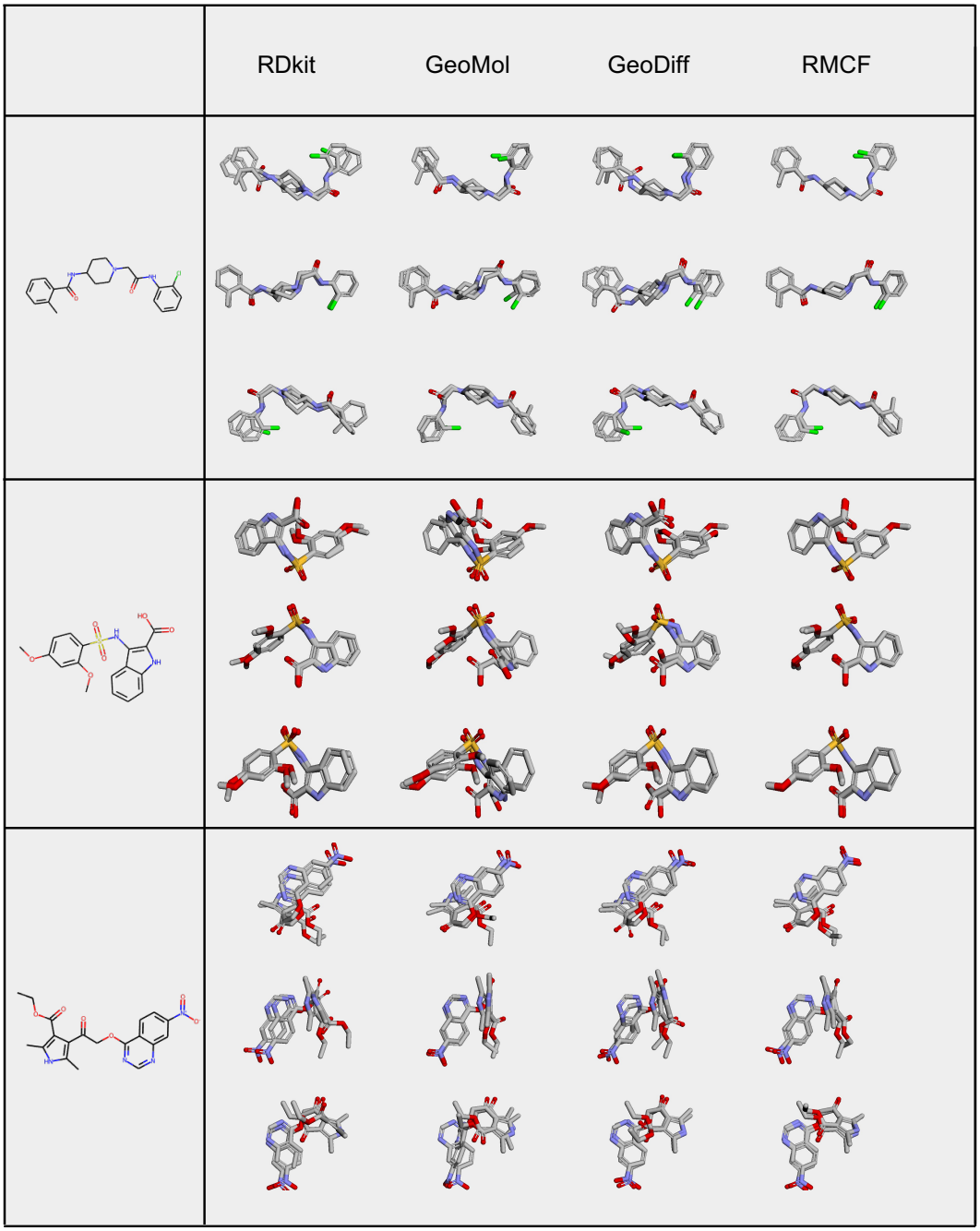

Figure 6: Examples of generated molecules from GEOM-Drugs dataset. For every model and molecule, we show three ground truths and the best-aligned conformations. Our model RMCF fits best with the references.

C QM9 Results

Table 4: Results on the **GEOM-QM9** dataset, without FF optimization.

Models	COV-R (%) ↑		MAT-R (Å) ↓		COV-P (%) ↑		MAT-P (Å) ↓	
	Mean	Median	Mean	Median	Mean	Median	Mean	Median
CVGAE	0.09	0.00	1.671	1.609	-	-	-	-
GraphDG	73.33	84.21	0.425	0.397	43.90	35.33	0.581	0.582
CGCF	78.05	82.48	0.422	0.390	36.49	33.57	0.662	0.643
ConfVAE	77.84	88.20	0.415	0.374	38.02	34.67	0.622	0.609
GeoMol	**91.52**	**100.0**	0.225	**0.193**	**86.71**	**100.0**	**0.270**	**0.241**
ConfGF	88.49	94.31	0.267	0.269	46.43	43.41	0.522	0.512
GeoDiff	90.07	93.39	**0.209**	0.199	52.79	50.29	0.445	0.440
RMCF-R	55.86	56.67	0.433	0.414	**84.86**	**96.51**	**0.260**	**0.227**
RMCF-C	76.22	85.50	0.320	0.289	83.10	91.44	0.283	0.261

D Graph Dynamic Programming Algorithm

Algorithm 1 Graph Dynamic Programming for Searching the Best Partition

Require: the fragments type collection $\mathcal{F}$, the molecular graph G_m
Ensure: the best molecular graph partition $\mathcal{P}^*$

```
function DoF(f)
    return max_{r_i, r_j ∈ V(f)} RMSD(r_i, r_j)
end function
function PARTITION(G)
    s ← 100, P ← {}
    for f ∈ F do
        if IsSubstructure(G, f) then
            s_m ← 0, z ← 0, P_m ← {f}
            F_r ← Remove(G, f)
            for f ∈ F_r do
                dof, P_f ← PARTITION(f)
                s_m ← s + dof * |P_f|, z ← z + |P_f|, P_m ← P_m ∪ P_f
            end for
            s_m ← s_m / z
        end if
        if s_m < s then
            s ← s_m, P ← P_m
        end if
        if P = {} then
            return DoF(G), {G}
        end if
    end for
end function
dof, P* ← PARTITION(G_m)
return P*
```

E Sampling Algorithm

Algorithm 2 Sampling Molecular Conformations From RMCF

Require: an RMCF $\mathcal{G}$, the number of conformations N_C, the number of sampling iterations N_S
Ensure: a set S_C containing N_C conformations

```
S ← {}
initialize a configuration 𝒳 randomly
for i ∈ [1, 2, ··· , N_S] do
    for n ∈ F ∪ D do
        U ← {u | (n, u) ∈ E}                                   ▷ x_u is fixed in this iteration
        P ← softmax{ Σ_{u∈U} E_u[x_u] W_un E_n^T + out(h_n) }   ▷ the value can be pre-computed
        x_n ~ Cat(|n|, P)
    end for
    add 𝒳 to S
end for
compute the pairwise distance K with Eq.10
S_C ← run clustering with k = N_C over S based on K
return S_C
```