TORSIONAL DIFFUSION FOR MOLECULAR CONFORMER GENERATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Diffusion-based generative models generate samples by mapping noise to data via the reversal of a diffusion process which typically consists of the addition of independent Gaussian noise to every data coordinate. This diffusion process is, however, not well suited to the fundamental task of molecular conformer generation where the degrees of freedom differentiating conformers lie mostly in torsion angles. We, therefore, propose Torsional Diffusion that generates conformers by leveraging the definition of a diffusion process over the space $\mathbb{T}^m$, a high dimensional torus representing torsion angles, and a $SE(3)$ equivariant model capable of accurately predicting the score over this process. Empirically, we demonstrate that our model outperforms state-of-the-art methods in terms of both diversity and accuracy of generated conformers, reducing the minimum RMSD by respectively 27% and 9%. When compared to Gaussian diffusion models, Torsional Diffusion enables significantly more accurate generation while performing two orders of magnitude fewer inference time-steps.

1 INTRODUCTION

Molecules are identified by their molecular graph, i.e., a set of atoms and the covalent bonds between them. However, it is the set of structures that the graph realizes when embedded in 3D space, called *conformers*, that determine many of its properties. Molecular conformer generation—predicting an ensemble or distribution over 3D conformers for a given molecular graph—is, therefore, a fundamental problem in computational chemistry. Existing approaches consist of methods that sample from the underlying potential energy surface, which are accurate but slow; or approaches leveraging chemical heuristics, which are fast but less accurate.

Deep generative models have been explored for molecular conformer generation in the hopes of combining high accuracy with fast sampling. GeoMol (Ganea et al., 2021) recently demonstrated competitive performance with a message-passing neural network and a custom parameterization and assembly procedure. Diffusion generative models (Ho et al., 2020; Song et al., 2021), which learn to reverse a stochastic process transforming the data distribution into noise, have also shown promise on this task (Shi et al., 2021; Luo et al., 2021; Xu et al., 2021b). In particular, GeoDiff (Xu et al., 2021b) uses an SE(3) equivariant score model to reverse a diffusion process which adds independent Gaussian noise to each atomic coordinate in Euclidean space. Sampling thus consists of denoising a point cloud where atoms are in random initial positions irrespective of the molecular graph. A large number ($T = 5000$) of such denoising steps are needed to accurately generate a conformer from such a point cloud.

We argue that this approach of diffusing Euclidean coordinates is ill-suited for molecular conformer generation, where bond lengths and angles can be determined very quickly and relatively accurately from the graph alone, and the difference between possible conformers lies largely in the torsion angles (Axelrod & Gomez-Bombarelli, 2020). Instead, we consider each conformer to be parameterized by torsion angles, and learn to reverse a diffusion that occurs only over these *torsion angle* coordinates. This has the effect of significantly reducing the dimensionality of the sample space; the molecules in GEOM-DRUGS, a common conformer generation dataset (Axelrod & Gomez-Bombarelli, 2020), have, on average, $n = 46.2$ atoms, corresponding to a $3n$-dimensional Euclidean space, but only $m = 8.65$ torsion angles of rotatable bonds.

However, m torsion angle coordinates define not a Euclidean space, but rather an m-dimensional torus $\mathbb{T}^m$. Thus, we first formulate the forward diffusion, score-matching, reverse diffusion and denoising procedures over the torus $\mathbb{T}^m$. The theoretical extension of diffusion modeling to non-Euclidean manifolds was developed very recently by De Bortoli et al. (2022). We build upon their work to present, to the best of our knowledge, the first extension of diffusion models to a real-world non-Euclidean domain.

Learning a neural network score model over the input space $\mathbb{T}^m$ presents its own challenges. The dimensionality of this space varies between molecular graphs, and all the information about the molecular graph would have to be made available to the score model. Additionally, there is no canonical way to define the torsion angle coordinate about each bond. To circumvent these difficulties, we instead formulate a *torsion update* about a particular bond as a *geometric* (i.e., SE(3)-equivariant) property of a 3D point cloud, and use 3D-equivariant networks to directly predict these properties from a point cloud $\mathbb{R}^{3n}$ representation of the conformer.

By combining diffusion over a torus with a novel equivariant score model over point clouds, we achieve state-of-the-art results on the standard GEOM-DRUGS dataset (Axelrod & Gomez-Bombarelli, 2020). Moreover, we generate samples using only 20 denoising steps—two orders of magnitude fewer than the Euclidean diffusion approach employed by GeoDiff.

2 Related Work

Molecular conformer generation is a fundamental problem that has been tackled with a wide range of approaches. Ab initio metadynamics methods like DFT are considered gold-standards but are too computationally demanding for most applications. Methods like CREST (Pracht et al., 2020) use various heuristics to reduce the number of energy evaluations, but still require an average of 90 core hours (Axelrod & Gomez-Bombarelli, 2020) to sample a drugs-like molecule conformations. In order to make large molecular screening possible, several rule-based methods such as RDKit ETKDG (Riniker & Landrum, 2015) and OMEGA (Hawkins et al., 2010) have been developed offering significantly faster but less accurate conformer generation. Machine learning approaches, such as GeoMol (Ganea et al., 2021), have been applied, but so far have only obtained results worse or comparable to state-of-the-art rule-based software.

Torsion angle molecular dynamics (TAMD) refers to the range of molecular dynamics methods (Ryckaert et al., 1977; Stein et al., 1997; Chen et al., 2005) using torsion angles instead of Cartesian coordinates as degrees of freedom. This significantly simplifies the potential energy landscape allowing for longer-time steps in numerical integration. However, TAMD methods have the disadvantage that the Lagrange equations of motions with torsion angles are more complex than Newton's in Euclidean space and naive approaches have a complexity cubic in the size of the system (Mazur & Abagyan, 1989; Mazur et al., 1991).

3 Torsional Diffusion

In Torsional Diffusion, we assume that the bond lengths and angles (collectively *local structures*) are already predicted to high accuracy by standard methods such as RDKit, and that generating the torsion angles around rotatable bonds is the main difficulty of conformer generation. We therefore develop a diffusion-based generative model over torsion angles. During inference, our model takes as input a conformer with local structures predicted by RDKit and random torsion angles, and successively updates the torsions to generate a new conformer. During training, the model is presented with ground-truth conformers with their torsion angles randomly perturbed (to varying scales) and learns to reverse these perturbations.

In Section 3.1, we formulate the constrained diffusion process over conformers. We find that the torsions are most naturally coordinates on a high-dimensional torus, so in Section 3.2 we describe how the continuous-time diffusion model framework transfers to data distributions on the torus. However, instead of defining a score model with inputs on the torus, we build a neural network whose *inputs* are conformers viewed as point clouds, but whose *outputs* for each bond are scores on the torus. In Section 3.3 we discuss the implications of this juxtaposition in terms of the symmetries

required of the model, and in Section 3.4 we describe our architecture which satisfies these symmetries. Finally, in Section 3.5, we discuss the distributional shift that results from naively *training* on conformers with ground truth local structures but *generating* conformers starting from RDKit local structures, and our strategy for resolving this distributional shift.

Throughout the exposition, we consider a bond *rotatable* if severing the bond creates two connected components of G, each of which has at least two atoms. Notably, this definition excludes torsion angles in rings / cycles, which are constrained geometrically and cannot be diffused independently. It includes, however, double bonds which are usually considered constrained by *cis*/*trans* isomerism. It also includes bonds to methyl, hydroxyl, or amine groups, whose torsions affect only the placements of hydrogens.

3.1 CONFORMER DIFFUSION

A molecule is a graph $G = (\mathcal{V}, \mathcal{E}) \in \mathcal{G}$ with atoms $v \in \mathcal{V}$ and (undirected) bonds $e \in \mathcal{E}$. A conformer of a molecule is an assignment $\mathcal{V} \mapsto \mathbb{R}^3$ of each atom to a point in 3D-space, defined up to global rototranslation. For notational convenience, we suppose there is an ordering of nodes such that we can regard such a mapping as a vector in $\mathbb{R}^{3n}$ where $n = |\mathcal{V}|$; because our formulation and models never make use of this ordering, they are manifestly permutation equivariant. Then a conformer $c \in \mathcal{C}_G$ can be regarded as a set of vectors in $\mathbb{R}^{3n}$ equivalent under the group action of $SE(3)$: $c = \{g(\mathbf{x}) \mid g \in SE(3)\}$ for some $\mathbf{x} \in \mathbb{R}^{3n}$.

For a given G, a conformer $c \in \mathcal{C}_G$ can be defined in terms of *intrinsic coordinates*: chirality tags $\mathbf{z} \in \{\text{D}, \text{L}\}^k$ where k is the number of chiral centers, a set $L \in \mathcal{L}_G$ of local structures, i.e., bond lengths and angles (which lie on a manifold given by geometric constraints), and some parameterization of the *torsions* around each bond. The chirality $\mathbf{z}$ is given as part of the identification of the molecule, and we assume L can be accurately predicted by standard methods; therefore we are concerned with conformer diffusion of the *torsions* only, conditioned on fixed $G, \mathbf{z}, L$. Typically, torsions are specified in terms of dihedral angles; i.e., the torsion around bond BC is given by the (oriented) dihedral angle between planes ABC, BCD, where A, D are respectively neighbors of B, C. However, this requires a choice of neighbors A, B, and furthermore changes sign when considering the bond in the reverse direction CB. We argue that requiring such a choice of neighbors and directions forces an unnatural asymmetry into the learning task, even if the choice is itself learned as in Ganea et al. (2021).

We instead propose a more natural and geometric parameterization of torsion by considering $\mathcal{C}_G$ as a space acted on by the group $SO(2)^m$, where m is the number of freely rotatable bonds. The group $SO(2)^m$ acts upon $\mathcal{C}_G$ in the following manner. Suppose there is an ordering of rotable bonds $\mathcal{E}_{\text{rot}} = (e_1, \ldots, e_m)$; again, we never make direct use of this ordering so our diffusion and model remain permutation equivariant. Then the group element g with an element corresponding to rotation θ in the ith position, and all other entries equal to the identity, acts upon a conformer c by modifying the torsion angle around bond e_i by θ in a manner which is defined independently of edge direction. Formally, if (a, b) are the bonded atoms and x_a, x_b are their positions, then

$$c' = g(c) \iff \exists \mathbf{x} \in c, \mathbf{x}' \in c'. \quad \mathbf{x}'_{\mathcal{V}(a)} = \mathbf{x}_{\mathcal{V}(a)} \quad \mathbf{x}'_{\mathcal{V}(b)} = R\left(\frac{x_b - x_a}{||x_b - x_a||}\theta, x_b\right)\mathbf{x}_{\mathcal{V}(b)} \tag{1}$$

where $\mathbf{x}_{\mathcal{V}(a)}$ is the positions of atoms on the same side of the bond as a, and $R(\boldsymbol{\theta}, x_b) \in SE(3)$ is the rotation by Euler vector $\boldsymbol{\theta}$ about x_b. This definition does not depend on choice of edge direction since c is defined only up to global action of $SE(3)$. More informally, rotating bond $e_i = (a, b)$ by θ means a *relative* rotation of $\mathbf{x}_{\mathcal{V}(a)}$ and $\mathbf{x}_{\mathcal{V}(b)}$ where the difference in rotation vectors $\boldsymbol{\theta}_b - \boldsymbol{\theta}_a$ is θ times the unit vector in the direction $x_b - x_a$. This definition is clearly indifferent to edge direction and is illustrated visually in Figure 1. The action of a generic action in $SO(2)^m$ is then given by sequentially applying the torsion updates for each rotatable bond; they are independent and thus respect the commutative group structure.

Having defined this group action, if we assume that for given $G, \mathbf{z}, L$ that there is a *canonical* conformer $c^*_G(\mathbf{z}, L) \in \mathcal{C}_G$, then for any $c \in \mathcal{C}_G$ we can identify the group action $g \in SO(2)^m$ that takes c^*_G to c with c itself, i.e., there is an isopmorphism $g \leftrightarrow g(c^*_G(\mathbf{z}, L))$. We can thus *lift* the space of conformers for given $G, \mathbf{z}, L$ to the group $SO(2)$, which is itself isomorphic to the

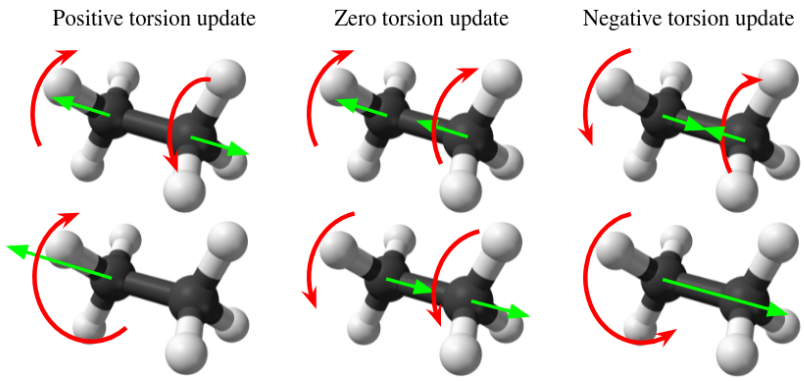

Figure 1: Torsion updates around a single bond. Rotations are shown in red and Euler vectors in green. Two equivalent torsion updates (among many possible) are shown for a positive, null, and negative torsion update, emphasizing the importance of *relative* rotations only.

m-torus: $SO(2)^m \cong \mathbb{T}^m \cong [-\pi, \pi)^m$. The coordinates on the torus then provide the sought-after parameterization of torsion, which we denote $T \in \mathbb{T}^m$. The assembly of 3D coordinates from intrinsic coordinates $\mathbf{z}, L, T$ is thus a bijection: $F_G : \{\mathrm{D}, \mathrm{L}\}^k \times \mathcal{L} \times \mathbb{T}^m \leftrightarrow \mathcal{C}_G$. With this parameterization, for any fixed $\mathbf{z}, L$, a diffusion over $\mathbb{T}^m$ maps to a diffusion on the corresponding subset of $\mathcal{C}_G$.

It may appear that the choice of $c^*_G(\mathbf{z}, L)$ is important but ambiguous. However, a change in the choice of $c^*_G(\mathbf{z}, L)$ merely corresponds to a shift in origin on the m-torus, translating the original and diffused data distributions but not otherwise changing the diffusion process. Importantly, the training and sampling routines also do not depend on the choice of coordinate origins, since we construct our model to operate *directly on conformation space* $\mathcal{C}_G$. Thus, the canonical conformer $c^*_G(\mathbf{z}, L)$ may be regarded as merely a formal and notational device to assist the definition of a coordinate system (and hence diffusion) over the space of intrinsic coordinates, but itself without any practical implications.

3.2 Score Modelling on $\mathbb{T}^m$

We now turn to formulating diffusion modelling on the m-torus. In Euclidean diffusion models, the data distribution $\mathbf{x}(0) \in \mathbb{R}^d$ is the initial distribution for a diffusion process

$$d\mathbf{x} = \mathbf{f}(\mathbf{x}, t)\ dt + g(t)\ d\mathbf{w} \quad t \in (0, 1) \tag{2}$$

which transforms $\mathbf{x}(0)$ into (approximately) a simple Gaussian $\mathbf{x}(1)$. A neural network trained to model the score $\nabla_{\mathbf{x}} \log p_t(\mathbf{x})$ enables sampling from the reverse diffusion,

$$d\mathbf{x} = \mathbf{f}(\mathbf{x}, t)\ dt - g^2(t) \nabla_{\mathbf{x}} \log p(\mathbf{x}, t)\ dt + g(t)\ d\bar{\mathbf{w}} \tag{3}$$

which transforms samples from the simple Gaussian $\mathbf{x}(1)$ into the data distribution $\mathbf{x}(0)$ (Song et al., 2021; Anderson, 1982). Access to the *diffusion kernel* $p(\mathbf{x}(t) \mid \mathbf{x}(0))$ for all times t is sufficient to train a score model $s_\theta(\mathbf{x}, t)$ via denoising score matching; and for sampling the reverse diffusion via the Euler-Maruyama solver. We refer to Song et al. (2021) for further details.

The continuous diffusion model formulation can also be applied with relatively few modifications to compact Reimannian manifolds (De Bortoli et al., 2022). If $d\mathbf{w}$ is redefined as Brownian motion on the manifold, then equation 3 continues to hold, where the score is now an element of the tangent space $\nabla_{\mathbf{x}} \log p(\mathbf{x}, t) \in T_{\mathbf{x}} M$. That is, sampling from a *geodesic random walk* that discretizes the reverse SDE correctly recovers the original data distribution $\mathbf{x}(0)$. While a number of subtleties exist for diffusion on general manifolds, we focus on $\mathbb{T}^m$ and refer interested readers to De Bortoli et al. (2022) for the general case.

A common choice of diffusion process is the rescaled Brownian motion given by $\mathbf{f}(\mathbf{x}, t) = 0, g(t) = \sqrt{\frac{d}{dt}\sigma^2(t)}$ where $\sigma^2(t)$ is the variance of the heat kernel $p(\mathbf{x}(t) \mid \mathbf{x}(0))$ and is the main parameter of

the diffusion. Specifically, we use an exponential diffusion $\sigma^2(t) = \sigma^2_{\text{min}}(\sigma^2_{\text{max}}/\sigma^2_{\text{min}})^t$ as in Song & Ermon (2019). We transfer this SDE to $\mathbb{T}^m$ by redefining $d\mathbf{w}$ to be the Brownian motion on the torus $\mathbb{T}^m$ viewed as the quotient space $\mathbb{R}^m/2\pi\mathbb{Z}^m$. Hence, the heat kernel on $\mathbb{T}^m$ is a simple wrapping of the heat kernel on $\mathbb{R}^m$; that is, for any $\mathbf{x}(t), \mathbf{x}(0) \in [-\pi, \pi)^m$, we have

$$p(\mathbf{x}(t) \mid \mathbf{x}(t)) \propto \sum_{\mathbf{d}\in\mathbb{Z}^m}^{\infty} \exp\left(-\frac{||\mathbf{x}(t) - \mathbf{x}(0) + 2\pi\mathbf{d}||^2}{2\sigma^2(t)}\right) \tag{4}$$

Because the space is compact, for sufficiently large $\sigma(1)$, the prior distribution $\mathbf{x}(1)$ approaches a *uniform* distribution (De Bortoli et al., 2022) over $\mathbb{T}^m$.

For training via denoising score matching, we compute the scores of this kernel via a numerical approximation and match in the tangent spaces $T_{\mathbf{x}}M$, which are isomorphic to $\mathbb{R}^m$. For sampling, we first sample from a uniform prior over the torsions. Then, the Euler-Maruyama solver is generalized to Riemannian manifolds in terms of a geodesic random walk (De Bortoli et al., 2022), which in the case of the torus simplifies as the wrapping of the Euler-Maruyama random walk on $\mathbb{R}^m$.

3.3 Score Equivariance

We now desire a score model which maps from $\mathcal{C}_G$ to the tangent space of $\mathbb{T}^m$, which is isomorphic to $\mathbb{R}^m$. We therefore use an $SE(3)$-equivariant score model (Geiger et al., 2020) conditioned on the input graph $s_G : \mathbb{R}^{3n} \mapsto \mathbb{R}^m$, which can be viewed as a function over $\mathcal{C}_G$ since for any $c \in \mathcal{C}_G$ and any $\mathbf{x}, \mathbf{x}' \in c$, we have $s_G(\mathbf{x}) = s_G(\mathbf{x}')$.

An additional symmetry arises from the fact that the underlying physical energy is invariant (or extremely nearly so) under *parity inversion* (Quack, 2002); thus our learned density should respect $p_G(-c) = p_G(c)$ where $-c = \{-\mathbf{x} \mid \mathbf{x} \in c\}$. In terms of intrinsic coordinates, the chirality and local structures transform under parity inversion as $\mathbf{z} \mapsto -\mathbf{z}, L \mapsto L$. To see how the torsions T transform, suppose that the *canonical conformer* function is exactly equivariant under parity inversion: $c^*_G(-\mathbf{z}, L) = -c^*_G(\mathbf{z}, L)$. We also need the following proposition:

Proposition 1. *For $g \in SO(2)^n$, $c \in \mathcal{C}_G$, and with the action of g on $\mathcal{C}_G$ defined previously, we have $-g(c) = (-g)(-c)$.*

Proof. It suffices to consider an action on a single bond as in equation 1. $c' = -g(c)$ implies

$$\exists \mathbf{x} \in c, \mathbf{x}' \in c'. \quad \mathbf{x}'_{\mathcal{V}(a)} = -\mathbf{x}_{\mathcal{V}(a)} \quad \mathbf{x}'_{\mathcal{V}(b)} = -\left[R\left(\frac{x_b - x_a}{||x_b - x_a||}\theta, x_b\right)\mathbf{x}_{\mathcal{V}(b)}\right] \tag{5}$$

On the other hand, $c' = (-g)(-c)$ implies

$$\exists \mathbf{x} \in c, \mathbf{x}' \in c'. \quad \mathbf{x}'_{\mathcal{V}(a)} = -\mathbf{x}_{\mathcal{V}(a)} \quad \mathbf{x}'_{\mathcal{V}(b)} = R\left(\frac{x_b - x_a}{||x_b - x_a||}\theta, -x_b\right)\left(-\mathbf{x}_{\mathcal{V}(b)}\right) \tag{6}$$

□

Since the two conditions are the same, we have $-g(c) = (-g)(-c)$. In particular, $-g(c^*_G(\mathbf{z}, L)) = (-g)(c^*_G(-\mathbf{z}, L))$. However, given the assembly function F_G defined previously, this is just $-F_G(\mathbf{z}, L, T) = F_G(-\mathbf{z}, L, -T)$. Thus we see that $T \mapsto -T$ under parity inversion.

Since an invariant density corresponds to an *equivariant* score, we need $s_G(F_G(T, L, \mathbf{z})) = -s_G(F_G(-T, L, -\mathbf{z}))$, which implies $s_G(\mathbf{x}) = -s_G(-\mathbf{x})$. Thus, the score model must be *invariant* under $SE(3)$ but *equivariant* under parity inversion of the input point cloud—i.e., it must output a set of *pseudoscalars*. While there exist a number of GNN architectures which are $SE(3)$ equivariant (Jing et al., 2021; Satorras et al., 2021), they are unable to produce pseudoscalar outputs and hence cannot satisfy the desired symmetry. The problem formulations therefore calls for the use of equivariant tensor-product based networks (Thomas et al., 2018; Geiger et al., 2020).

3.4 Score Network Architecture

Overview To perform the torsion score prediction under these symmetry constraints we design a novel architecture formed by three components: an embedding layer, a series of K interaction layers

and a pseudotorque layer. The pseudotorque layer produces pseudoscalar torsion scores τ_g for every rotatable bond. Following the notation from Thomas et al. (2018), we represent the node representations as $V_{acm}^{(k,l,p)}$ a dictionary with keys the layer k, rotation order l and parity p that contains tensors with shapes $[|\mathcal{V}|, n_l, 2l+1]$ corresponding to the indices of the node, channel and representation respectively. We use the `e3nn` library (Geiger et al., 2020) to implement our architecture.

Embedding layer In the embedding layer, we build a radius graph $(\mathcal{V}, \mathcal{E}_{r_{\max}})$ around each atom on top of the original molecular graph and generate initial scalar embeddings for nodes $V_a^{(0,0,0)}$ and edges e_{ab} combining chemical properties, sinusoidal embeddings of time $\phi(t)$ (Vaswani et al., 2017) and, for the edges, a radial basis function representation of their length $\mu(r_{ab})$ (Schütt et al., 2017):

$$\mathcal{E}_{r_{\max}} = \mathcal{E} \sqcup \{(a,b) \mid r_{ab} < r_{\max}\}$$
$$e_{ab} = \Upsilon^{(e)}(fe_{ab}||\mu(r_{ab})||\phi(t)) \quad \forall (a,b) \in \mathcal{E}_{r_{\max}}$$
$$V_a^{(0,0,0)} = \Upsilon^{(v)}(f_a||\phi(t)) \quad \forall a \in \mathcal{V}$$

where $\Upsilon^{(e)}$ and $\Upsilon^{(v)}$ are learnable two-layers MLPs, r_{ab} is the Euclidean distance between atoms a and b, $r_{\max} = 5\mathring{A}$ is the distance cutoff, f_a are the chemical features of atom a, f_{ab} are the chemical features of bond (a,b) if it was part of $\mathcal{E}$ and 0 otherwise.

Interaction layers The interaction layers are based on E(3)NN (Geiger et al., 2020) convolutional layers. At each layer, for every pair of nodes in the graph, we construct messages using tensor products of the current irreducible representation of each node with the spherical harmonic representations of the normalized edge vector. These messages are themselves irreducible representations, which are weighted channel-wise by a scalar function of the current scalar representations of the two nodes and the edge and aggregated with Clebsch-Gordan coefficients.

At every layer k, for every node a and rotation order l_o:

$$V_{acm_o}^{(k,l_o,p_o)} = \sum_{l_f,l_i,p_i} \sum_{m_f,m_i} C_{(l_i,m_i)(l_f,m_f)}^{(l_o,m_o)} \frac{1}{|\mathcal{N}_a|} \sum_{b \in \mathcal{N}_a} \psi_{abc}^{(k,l_o,l_f,l_i,p_i)} \; Y_{m_f}^{(l_f)}(\hat{r}_{ab}) \; V_{bcm_i}^{(k-1,l_i,p_i)}$$
$$\text{with } \psi_{abc}^{(k,l_o,l_f,l_i,p_i)} = \Psi_c^{(k,l_o,l_f,l_i,p_i)}(e_{ab}||V_a^{(k-1,0,1)}||V_b^{(k-1,0,1)})$$

where the outer sum is over values of l_f, l_i, p_i such that $|l_i - l_f| \leq l_o \leq l_i + l_f$ and $(-1)^{l_f} p_i = p_o$, C indicates the Clebsch-Gordan coefficients (Thomas et al., 2018), $\mathcal{N}_a = \{b \mid (a,b) \in \mathcal{E}_{\max}\}$ the neighborhood of a and Y the spherical harmonics. The rotational order of the nodes representations l_o and l_i are restricted to be at most 1, while the one of spherical harmonics of the edges (l_f) at most 2. All the learnable weights are contained in Ψ, a dictionary of MLPs that compute per-channel weights based on the edge embeddings and scalar features of the outgoing and incoming node.

Pseudotorque layer The final part of our architecture is a pseudotorque layer that predicts a pseudoscalar score τ_g for each rotatable bond g from the per-node outputs of the interaction layers. For every rotatable bond, we construct a tensor-valued filter, centered on the bond, from the tensor product of the spherical harmonics with a $l = 2$ representation of the *bond axis*. Since the parity of the $l = 2$ spherical harmonic is even, this representation does not require a choice of bond direction. The filter is then used to convolve with the representations of every neighbor on a radius graph, and the products which produce pseudoscalars are passed through odd-function (i.e., with $\tanh$ nonlinearity and no bias) dense layers to produce a single prediction.

For all rotatable bonds $g = (g_0, g_1) \in \mathcal{E}_{\text{rot}}$ and $b \in \mathcal{V}$, let r_{gb} and $\hat{r}_{gb}$ be the magnitude and direction of the vector connecting the center of bond g and b.

$$\mathcal{E}_\tau = \{(g,b) \mid g \in \mathcal{E}_r, b \in \mathcal{V}, r_{gb} < r_{\max}\} \qquad e_{gb} = \Upsilon^{(\tau)}(\mu(r_{gb}))$$
$$T_{gbcm_o}^{(l_o,p_o)} = \sum_{m_g,m_r,l_r:p_o=(-1)^{l_r}} C_{(2,m_g)(l_r,m_r)}^{(l_o,m_o)} Y_{m_f}^{(2)}(\hat{r}_g) \; Y_{m_r}^{(l_r)}(\hat{r}_{gb})$$
$$\tau_g = \sum_{l,p_f,p_i:p_f p_i=-1} \sum_{m_o,m_i} C_{(l,m_f)(l,m_i)}^{(0,0)} \frac{1}{|\mathcal{N}_g|} \sum_{b \in \mathcal{N}_g} \gamma_{gb}^{(l,p_i)} \; T_{gbcm_f}^{(l,p_f)} \; V_{bcm_i}^{(K,l,p_i)}$$

Algorithm 1: Conformer matching

Input: true conformers of G $[c_1, ...c_K]$
Output: approximate conformers for training $[\hat{c}_1, ...\hat{c}_K]$
generate local structures $[\hat{L}_1, ...\hat{L}_K]$ with RDKit;
for (i, j) ***in*** $[1, K] \times [1, K]$ **do**
  c_{temp} = von_Mises_matching(c_i, $\hat{L}_j$);
  cost[i,j] = RMSD(c_i, c_{temp});
assignment = linear_sum_assignment(cost);
for $i \leftarrow 1$ **to** K **do**
  j = assignment[i];
  $\hat{c}_i$ = differential_evolution(c_i, $\hat{L}_j$, RMSD);

$$\text{with } \gamma_{gb}^{(l,p_i)} = \Gamma^{(l,p_i)}(e_{gb}||V_b^{(K,0,0)}||V_{g_0}^{(K,0,0)} + V_{g_1}^{(K,0,0)})$$

where $\Upsilon^{(\tau)}$ and $\Gamma^{(l_o,l_i,l_f,p_o,p_i)}$ are MLPs with learnable parameters and $\mathcal{N}_g = \{b \mid (g, b) \in \mathcal{E}_\tau\}$.

3.5 Sampling and Conformer Matching

Sampling conformers At inference time, in order to use the Torsional Diffusion on a molecular graph G with chirality $\mathbf{z}$, we first need to access a estimate sample $\hat{L}$ containing a set of local structures from $\mathcal{L}_G$. This estimate can be done with high accuracy with rule-based methods, therefore we use RDKit ETKDG (Riniker & Landrum, 2015) to get random conformation whose local structure we refer as $\hat{L}$. We then sample uniformly random initial conformations in the toroidal space defined by this local structure $\hat{L}$ by randomly changing each of its torsion angles by $U[-\pi, \pi]$. Random conformers from the trained model are then sampled by performing 20 steps of reverse diffusion on each of these conformations.

Conformer matching At training time, if we directly diffuse the ground truth conformers, the model would have access to the exact local structures when learning to match the denoising scores. Our experiments show that this disparity between training and inference of having exact or estimated local structures (L versus $\hat{L}$) causes a distributional shift that hurts the model at inference time. We bridge this shift at training time by a preprocessing procedure that we refer to as *conformer matching*.

The essence of this procedure is that we substitute each ground truth conformer c with a synthetic conformer $\hat{c}$ with the same chirality $\mathbf{z}$, but with local structures $\hat{L}$ sampled by RDKit —thus available at testing time— and made as similar as possible to c. That is, we use RDKit to generate $\hat{L}$ and change torsion angles $\hat{T}$ to minimize RMSD$(c, \hat{c})$. Naively, we could sample $\hat{L} \sim L \mid G, \mathbf{z}$ from RDKit independently for each ground truth conformer. However, this nullifies any possible dependence between L and T that could serve as a potential training signal. Instead, we recognize that the distributional shift induces a domain adaptation problem that can be solved by matching the RDKit and ground truth distributions in an optimal sense as follows.

For a molecule with K conformers, we first generate K random local structure estimates from RDKit. To match with the ground truth local structures, we compute the cost of matching each true conformer with each estimate (so a K^2 cost matrix), where the cost is the best RMSD that can be achieved by modifying the torsions of the RDKit conformer to match the ground truth one. In practice, we compute an upper bound to this optimal RMSD using the fast von Mises torsion matching procedure proposed by Stärk et al. (2022). We find an optimal matching of true conformers c to local structure estimates $\hat{L}$ by solving the linear sum assignment problem over the approximate cost matrix using Jonker-Volgenant algorithm (Crouse, 2016). Finally, for each matched pair we find obtain the optimal $\hat{c}$ by running a differential evolution optimisation procedure over the torsion angles (Méndez-Lucio et al., 2021). The complete assignment resulting from the linear sum solution guarantees that there is no distributional shift in the local structures seen during training and inference. This procedure is summarised in Algorithm 1.

Table 1: Performance of various methods on the GEOM-Drugs dataset test-set. Note that GeoDiff uses a different set of random splits.

	Cov-R		AMR-R		Cov-P		AMR-P	
Model	Mean	Med	Mean	Med	Mean	Med	Mean	Med
RDKit ETKDG	68.78	76.04	1.042	0.982	71.06	88.24	1.036	0.943
OMEGA	81.64	97.25	0.851	0.771	77.18	**96.15**	0.951	0.854
GraphDG	10.37	0.00	1.950	1.933	3.98	0.00	2.420	2.420
CGCF	54.35	56.74	1.248	1.224	24.48	15.00	1.837	1.829
GeoMol	82.43	95.10	0.862	0.837	78.52	94.4	0.933	0.856
GeoDiff	89.13	97.88	0.863	0.853	61.47	64.55	1.171	1.123
Torsional Diffusion (ours)	**94.09**	**100.00**	**0.619**	**0.605**	**80.78**	90.95	**0.850**	**0.812**

4 EXPERIMENTS

Dataset & evaluation To test our method on the GEOM-DRUGS dataset (Axelrod & Gomez-Bombarelli, 2020) which is composed of a total of 304k molecules each with an associated set of conformers obtained using CREST. In order to provide a fair comparison with previous methods, we follow the filtering and splitting from Ganea et al. (2021) and use the same evaluation metrics: Average Minimum RMSD (AMR) and Coverage (COV) both for Recall (R) —how many ground truth conformers are correctly predicted— and Precision (P) —how many of the predicted structures are of high quality.

Baselines We compare our performance with a wide variety of existing methods. RDKit ETKDG (Riniker & Landrum, 2015) is the most popular open-source method. OMEGA (Hawkins et al., 2010; Hawkins & Nicholls, 2012) is a rule-based commercial package in continuous development. GraphDG (Simm & Hernández-Lobato, 2019), CGCF (Xu et al., 2021a), GeoMol (Ganea et al., 2021) and GeoDiff (Xu et al., 2021b) are recent machine learning approaches that have achieved competitive or state-of-the-art performances.

Results & discussion The results presented in Table 1 show that our method outperforms all previous methods on 7 out of the 8 evaluation metrics reducing by 27% the average minimum recall RMSD and by 9% the precision RMSD of the previous state-of-the-art method. This highlights the strength of the presented method and the high diversity obtainable with diffusion models (resulting in significantly better recall performances). The advantage provided by the Torsional Diffusion is evident when comparing it to the other method based on diffusion models, GeoDiff. GeoDiff requires 5k inference steps, and therefore 5k score model evaluations, to obtain the results in Table 1. On the other hand, Torsional Diffusion is able to significantly outperform it on all metrics, with only 20 steps by restricting the diffusion process in the subspace where most of the molecule's flexibility lies.

5 CONCLUSION

We presented Torsional Diffusion, a novel method based on score-based diffusion models, to generate molecular conformers. We define a diffusion process of the high dimensional torus representing the space of possible torsion angles and the associated score-matching and reverse diffusion. Then, we present a novel SE(3)-equivariant model to predict the scores in the torsion angle distributions and a preprocessing technique to bridge the inference distributional shift. Empirically, we obtain state-of-the-art performances on all diversity metrics and competitive precision results. Moreover, compared to previous diffusion-based techniques on Euclidean spaces, we are able to generate conformers with two orders of magnitude fewer time-steps.

REFERENCES

Brian DO Anderson. Reverse-time diffusion equation models. *Stochastic Processes and their Applications*, 12(3):313–326, 1982.

Simon Axelrod and Rafael Gomez-Bombarelli. Geom: Energy-annotated molecular conformations for property prediction and molecular generation. *arXiv preprint arXiv:2006.05531*, 2020.

Jianhan Chen, Wonpil Im, and Charles L Brooks III. Application of torsion angle molecular dynamics for efficient sampling of protein conformations. *Journal of computational chemistry*, 26(15): 1565–1578, 2005.

David F Crouse. On implementing 2d rectangular assignment algorithms. *IEEE Transactions on Aerospace and Electronic Systems*, 52(4):1679–1696, 2016.

Valentin De Bortoli, Emile Mathieu, Michael Hutchinson, James Thornton, Yee Whye Teh, and Arnaud Doucet. Riemannian score-based generative modeling. *arXiv preprint arXiv:2202.02763*, 2022.

Octavian Ganea, Lagnajit Pattanaik, Connor Coley, Regina Barzilay, Klavs Jensen, William Green, and Tommi Jaakkola. Geomol: Torsional geometric generation of molecular 3d conformer ensembles. *Advances in Neural Information Processing Systems*, 34, 2021.

Mario Geiger, Tess Smidt, Alby M., Benjamin Kurt Miller, Wouter Boomsma, Bradley Dice, Kostiantyn Lapchevskyi, Maurice Weiler, Michał Tyszkiewicz, Simon Batzner, Martin Uhrin, Jes Frellsen, Nuri Jung, Sophia Sanborn, Josh Rackers, and Michael Bailey. Euclidean neural networks: e3nn, 2020. URL `https://doi.org/10.5281/zenodo.5292912`.

Paul CD Hawkins and Anthony Nicholls. Conformer generation with omega: learning from the data set and the analysis of failures. *Journal of chemical information and modeling*, 52(11): 2919–2936, 2012.

Paul CD Hawkins, A Geoffrey Skillman, Gregory L Warren, Benjamin A Ellingson, and Matthew T Stahl. Conformer generation with omega: algorithm and validation using high quality structures from the protein databank and cambridge structural database. *Journal of chemical information and modeling*, 50(4):572–584, 2010.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 2020.

Bowen Jing, Stephan Eismann, Patricia Suriana, Raphael John Lamarre Townshend, and Ron Dror. Learning from protein structure with geometric vector perceptrons. In *International Conference on Learning Representations*, 2021.

Shitong Luo, Chence Shi, Minkai Xu, and Jian Tang. Predicting molecular conformation via dynamic graph score matching. *Advances in Neural Information Processing Systems*, 34, 2021.

AK Mazur, VE Dorofeev, and RA Abagyan. Derivation and testing of explicit equations of motion for polymers described by internal coordinates. *Journal of Computational Physics*, 92(2):261–272, 1991.

Alexey K Mazur and Ruben A Abagyan. New methodology for computer-aided modelling of biomolecular structure and dynamics 1. non-cyclic structures. *Journal of Biomolecular Structure and Dynamics*, 6(4):815–832, 1989.

Oscar Méndez-Lucio, Mazen Ahmad, Ehecatl Antonio del Rio-Chanona, and Jörg Kurt Wegner. A geometric deep learning approach to predict binding conformations of bioactive molecules. *Nature Machine Intelligence*, 3(12):1033–1039, 2021.

Philipp Pracht, Fabian Bohle, and Stefan Grimme. Automated exploration of the low-energy chemical space with fast quantum chemical methods. *Physical Chemistry Chemical Physics*, 22(14): 7169–7192, 2020.

Martin Quack. How important is parity violation for molecular and biomolecular chirality? *Angewandte Chemie International Edition*, 41(24):4618–4630, 2002.

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.

Jean-Paul Ryckaert, Giovanni Ciccotti, and Herman JC Berendsen. Numerical integration of the cartesian equations of motion of a system with constraints: molecular dynamics of n-alkanes. *Journal of computational physics*, 23(3):327–341, 1977.

Vıctor Garcia Satorras, Emiel Hoogeboom, and Max Welling. E (n) equivariant graph neural networks. In *International Conference on Machine Learning*, pp. 9323–9332. PMLR, 2021.

Kristof Schütt, Pieter-Jan Kindermans, Huziel Enoc Sauceda Felix, Stefan Chmiela, Alexandre Tkatchenko, and Klaus-Robert Müller. Schnet: A continuous-filter convolutional neural network for modeling quantum interactions. *Advances in neural information processing systems*, 30, 2017.

Chence Shi, Shitong Luo, Minkai Xu, and Jian Tang. Learning gradient fields for molecular conformation generation. In *International Conference on Machine Learning*, pp. 9558–9568. PMLR, 2021.

Gregor NC Simm and José Miguel Hernández-Lobato. A generative model for molecular distance geometry. *arXiv preprint arXiv:1909.11459*, 2019.

Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in Neural Information Processing Systems*, 2019.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *International Conference on Learning Representations*, 2021.

Hannes Stärk, Octavian-Eugen Ganea, Lagnajit Pattanaik, Regina Barzilay, and Tommi Jaakkola. Equibind: Geometric deep learning for drug binding structure prediction. *arXiv preprint arXiv:2202.05146*, 2022.

Evan G Stein, Luke M Rice, and Axel T Brünger. Torsion-angle molecular dynamics as a new efficient tool for nmr structure calculation. *Journal of Magnetic Resonance*, 124(1):154–164, 1997.

Nathaniel Thomas, Tess Smidt, Steven Kearnes, Lusann Yang, Li Li, Kai Kohlhoff, and Patrick Riley. Tensor field networks: Rotation-and translation-equivariant neural networks for 3d point clouds. *arXiv preprint arXiv:1802.08219*, 2018.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Minkai Xu, Shitong Luo, Yoshua Bengio, Jian Peng, and Jian Tang. Learning neural generative dynamics for molecular conformation generation. *arXiv preprint arXiv:2102.10240*, 2021a.

Minkai Xu, Lantao Yu, Yang Song, Chence Shi, Stefano Ermon, and Jian Tang. Geodiff: A geometric diffusion model for molecular conformation generation. In *International Conference on Learning Representations*, 2021b.