Generating Molecular Conformer Fields

Yuyang Wang **Ahmed A. Elhag*** **Navdeep Jaitly**
Joshua M. Susskind **Miguel Angel Bautista**
Apple
{yuyang_wang4, aa_elhag, jsusskind, njaitly, mbautistamartin}@apple.com

Abstract

In this paper we tackle the problem of generating conformers of a molecule in 3D space given its molecular graph. We parameterize these conformers as continuous functions that map elements from the molecular graph to points in 3D space. We then formulate the problem of learning to generate conformers as learning a distribution over these functions using a diffusion generative model, called Molecular Conformer Fields (MCF). To model the field of graphs, we build score network on Transformer-based architecture. Our approach is simple and scalable, and obtains results that are comparable or better than the previous state-of-the-art while making no assumptions about the explicit structure of molecules (*e.g.* modeling torsional angles). MCF represents an advance in extending diffusion models to handle complex scientific problems in a conceptually simple, scalable and effective manner.

1 Introduction

In this paper we tackle the problem of Molecular Conformer Generation, *i.e.* predicting the diverse low-energy three-dimensional conformers of molecules, relying solely on their molecular graphs as illustrated in Fig. 1. Molecular Conformer Generation is a fundamental problem in computational drug discovery and chemo-informatics (Batzner et al., 2022). Conventional methods like rule-based OMEGA (Hawkins et al., 2010) or stochastic Markov chain Monte Carlo (MCMC) (Wilson et al., 1991) and molecular dynamics (MD) (Grebner et al., 2011) can suffer from sampling inefficiency and fail to generate representative conformers for complex molecules (Hawkins, 2017). In the domain of learning-based approaches, several works have looked at conformer generation problems through the lens of probabilistic modeling, using either normalizing flows (Xu et al., 2021a) or diffusion models (Xu et al., 2022; Jing et al., 2022). For example, the quality of conformers from GeoMol (Ganea et al., 2021) and Torsional Diffusion (Jing et al., 2022) depends on the local substructure prediction model which is not differentiable. On the other hand, recent works have proposed domain-agnostic approaches for generative modeling of data in function space (Du et al., 2021; Dupont et al., 2022b,a; Zhuang et al., 2023) obtaining great performance (see Sect. A.1 for additional related work). As an example, in Zhuang et al. (2023) the authors use a diffusion model to learn a distribution over fields f, showing great results on different data domains like images (*i.e.* $f : \mathbb{R}^2 \to \mathbb{R}^3$) or 3D geometry (*i.e.* $f : \mathbb{R}^3 \to \mathbb{R}^1$), where the domain of the function $\mathbb{R}^n$ is fixed across functions. However, dealing with fields defined on different domains (*e.g.* different molecular graphs, as in molecular conformer generation) still remains an open problem.

To address these issues, we present Molecular Conformer Fields (MCF), an approach to learn generative models of molecular conformers. We interpret conformers as fields/functions (we use both terms exchangeably) on graphs that map elements in the graph $\mathcal{G}_i$ to points in $\mathbb{R}^3$, $f_i : \mathcal{G}_i \to \mathbb{R}^3$, which we define as *conformer fields*. Our contributions can be summarized as follows:

*Work was completed while A.A. was an intern with Apple.

Neural Information Processing Systems (NeurIPS) 2023 Workshop on Generative AI and Biology.

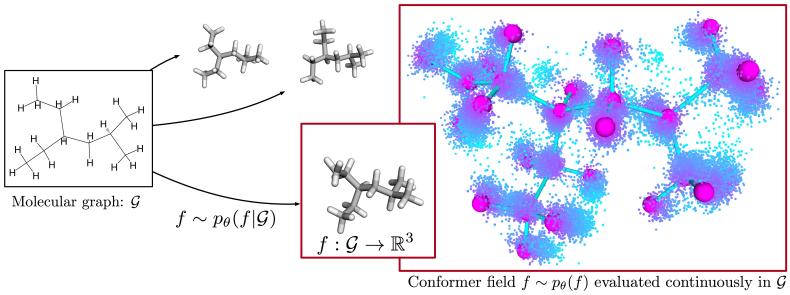


Figure 1: We formulate the molecular conformer generation problem as learning a distribution of functions over graphs. Each conformer is a field (*e.g.* a function) f that can map elements from a particular molecular graph $\mathcal{G}$ to points in 3D space, $f : \mathcal{G} \to \mathbb{R}^3$. We call these *conformer fields* as they can be continuously evaluated for any point in $\mathcal{G}$. We visualize an example of conformer field where points are colored based on the distance to their closest atom (see Appendix A.7 for details).

- We provide a simple yet effective approach for molecule conformer generation that outperforms previous approaches on standard benchmarks.
- Our approach directly predicts the 3D position of atoms as opposed to torsional angles or other domain-specific variables, providing a simple and scalable training recipe.
- Unlike most conformer generation models based on graph neural networks, we demonstrate the effectiveness of Transformer-based encode-decoder model to this problem.
- We provide an extensive ablation study to understand what are the factors that are critical for performance in molecular conformer generation.

2 Method

MCF is a diffusion generative model that captures distributions over conformer fields. We are given observations in the form of an empirical distribution $f_0 \sim q(f_0)$ over fields where a field $f_0 : \mathcal{G} \to \mathbb{R}^3$ maps vertices $v \in \mathcal{G}$ on a molecular graph $\mathcal{G}$ to 3D space $\mathbb{R}^3$. As a result, latent variables $f_{1:T}$ are also functions on graphs that can be continuously evaluated.

2.1 Conformers as Functions on Graphs

Following the setting in previous work (Xu et al., 2022; Ganea et al., 2021; Jing et al., 2022) a 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 $\mathcal{E} = \{e_{ij} | (i,j) \subseteq |\mathcal{V}| \times |\mathcal{V}|\}$ is the set of edges representing inter-atomic bonds. In this paper, we parameterize a molecule's conformer as a function $f : \mathcal{G} \to \mathbb{R}^3$ that takes atoms in the molecular graph $\mathcal{G}$ and maps them to 3D space, we call this function a *conformer field*. The training set is composed of conformer fields $f_i : \mathcal{G}_i \to \mathbb{R}^3$, that each maps atoms of a different molecule $\mathcal{G}_i$ to a 3D point. We then formulate the task of conformer generation as learning a prior over a training set of conformer fields. We drop the subscript i in the remainder of the text for notation simplicity.

We learn a denoising diffusion generative model (Ho et al., 2020) over conformer fields f. In particular, given conformer field samples $f_0 \sim q(f_0)$ the forward process takes the form of a Markov Chain with progressively increasing Gaussian noise: $q(f_{1:T}|f_0) = \prod_{t=1}^T q(f_t|f_{t-1}), \quad q(f_t|f_{t-1}) := \mathcal{N}(f_{t-1}; \sqrt{\bar{\alpha}_t} f_0, (1-\bar{\alpha}_t)\mathbf{I})$. We train MCF using the denoising objective function in (Ho et al., 2020): $\mathbb{E}_{t\sim[0,T], f_0\sim q(f_0), \epsilon\sim\mathcal{N}(0,\mathbf{I})} \left[\|\epsilon - \epsilon_\theta(\sqrt{\bar{\alpha}_t} f_0 + \sqrt{1-\bar{\alpha}_t}\epsilon, t)\|^2 \right]$. We discuss the benefits of building generative model over conformer fields in Appendix A.7.

2.2 Diffusion Generative Model on Conformer Fields

To tackle the problem of learning a diffusion generative model over conformer fields we extend the recipe in (Zhuang et al., 2023), generalizing from fields defined in ambient Euclidean space to functions on graphs (*e.g.* conformer fields). In order to do this, we compute the k leading eigenvectors

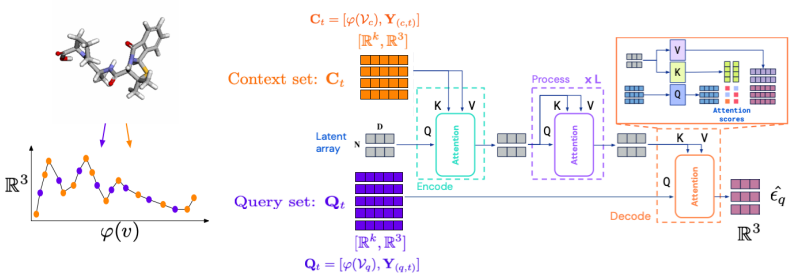


Figure 2: Interaction between context and query pairs in the PerceiverIO architecture. Context pairs $\mathbf{C}_t$ attend to a latent array of learnable parameters via cross attention. The latent array then goes through several self attention blocks. Finally, the query pairs $\mathbf{Q}_t$ cross-attend to the latent array to produce the final noise prediction $\hat{\epsilon_q}$.

of the normalized graph Laplacian $\Delta_{\mathcal{G}}$ (Maskey et al., 2022; Sharp et al., 2022) as positional encoding for points in the graph. The eigen-decomposition of the normalized graph Laplacian can be computed efficiently using sparse eigen-problem solvers (Hernandez et al., 2009) and only needs to be computed once before training. We use the term $\varphi(v) = \sqrt{n}[\varphi_1(v), \varphi_2(v), \ldots, \varphi_k(v)] \in \mathbb{R}^k$ to denote the normalized Laplacian eigenvector representation of a vertex $v \in \mathcal{G}$.

We adopt an explicit field parametrization where a field is characterized by uniformly sampling a set of vertex-signal pairs $\{(\varphi(v_c), \boldsymbol{y}_{(c,0)})\}$, $v_c \in \mathcal{G}, \boldsymbol{y}_{(c,0)} \in \mathbb{R}^3$, which is denoted as *context set*. We row-wise stack the context set and refer to the resulting matrix via $\mathbf{C}_0 = [\varphi(\mathcal{V}_c),\ \mathbf{Y}_{(c,0)}]$. Here, $\varphi(\mathcal{V}_c)$ denotes the Laplacian eigenvector representation context vertices and $\mathbf{Y}_{(c,0)}$ denotes the 3D position of context vertices at time $t = 0$. We define the forward process for the context set by diffusing the 3D positions and keeping Laplacian eigenvectors fixed:

$$\mathbf{C}_t = [\varphi(\mathcal{V}_c), \mathbf{Y}_{(c,t)} = \sqrt{\bar{\alpha}_t}\mathbf{Y}_{(c,0)} + \sqrt{1-\bar{\alpha}_t}\epsilon_c], \tag{1}$$

where $\epsilon_c \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ is a noise vector of the appropriate size. We now turn to the task of formulating a score network for fields. The score network needs to take as input the context set (*i.e.* the field parametrization), and needs to accept being evaluated continuously in $\mathcal{G}$. We do this by sampling a *query set* of vertex-signal pairs $\{\varphi(v_q), \boldsymbol{y}_{(q,0)}\}$. Equivalently to the context set, we row-wise stack query pairs and denote the resulting matrix as $\mathbf{Q}_0 = [\varphi(\mathcal{V}_q),\ \mathbf{Y}_{(q,0)}]$. Note that the forward diffusion process is equivalently defined for both context and query sets:

$$\mathbf{Q}_t = [\varphi(\mathcal{V}_q), \mathbf{Y}_{(q,t)} = \sqrt{\bar{\alpha}_t}\mathbf{Y}_{(q,0)} + \sqrt{1-\bar{\alpha}_t}\epsilon_q], \tag{2}$$

where $\epsilon_q \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ is a noise vector of the appropriate size. The underlying field is solely defined by the context set, and the query set are the function evaluations to be de-noised. The resulting *score field* model is formulated as follows, $\hat{\epsilon_q} = \epsilon_\theta(\mathbf{C}_t, t, \mathbf{Q}_t)$. The detailed algorithms can be found in the Appendix A.5.

2.3 Score Field Network ϵ_θ

In MCF, the score field's design space covers all architectures that can process irregularly sampled data, such as Transformers (Vaswani et al., 2017) and their corresponding Graph counterparts (Maskey et al., 2022; Sharp et al., 2022; He et al., 2022; Dwivedi et al., 2020). The score field network ϵ_θ is primarily implemented using PerceiverIO (Jaegle et al., 2022), an effective transformer encoder-decoder architecture that efficiently manage large numbers of elements in the context and query sets. Fig. 2 demonstrates how these sets are used within the PerceiverIO architecture.

3 Experiments

We use two popular datasets: GEOM-QM9 (Ruddigkeit et al., 2012) and GEOM-DRUGS (Ruddigkeit et al., 2012). We follow the same splitting and experimental settings as described in GeoMol (Ganea

et al., 2021) and Torsional Diff. Jing et al. (2022). we report the same metrics as previous works to compare the generated and ground truth conformer ensembles: Average Minimum RMSD (AMR) and Coverage. In the Appendix A.6.1 and Appendix A.6.2 we provide additional experiments that validate the design choices for the score network architecture, as well as empirically validating the chemical properties of generated conformer ensembles in Appendix A.6.3.

3.1 GEOM-QM9

We report results in Tab. 1, showing that MCF outperforms previous approaches by a substantial margin. In addition, it is important to note that MCF is a general approach for learning functions on graphs that does not make any assumptions about the intrinsic geometric factors important in conformers like torsional angles. This makes MCF simpler to implement and applicable to other settings in which intrinsic geometric factors are not known or expensive to compute.

	Recall				Precision			
	Coverage ↑		AMR ↓		Coverage ↑		AMR ↓	
	mean	median	mean	median	mean	median	mean	median
CGCF (Xu et al., 2021a)	69.47	96.15	0.425	0.374	38.20	33.33	0.711	0.695
GeoDiff (Xu et al., 2022)	76.50	**100.00**	0.297	0.229	50.00	33.50	0.524	0.510
GeoMol (Ganea et al., 2021)	91.50	**100.00**	0.225	0.193	87.60	**100.00**	0.270	0.241
Torsional Diff. (Jing et al., 2022)	92.80	**100.00**	0.178	0.147	92.70	**100.00**	0.221	0.195
MCF (ours)	**96.57**	**100.00**	**0.107**	**0.072**	**94.60**	**100.00**	**0.130**	**0.084**

Table 1: Molecule conformer generation results on GEOM-QM9. MCF obtains better results than the state-of-the-art Torsional Diffusion (Jing et al., 2022), without making any explicit assumptions about the geometric structure of molecules (*i.e.* without modeling torsional angles).

3.2 GEOM-DRUGS

Results on Tab. 2 where we see how MCF outperforms previous approaches by substantial a margin while being comparable to Torsional Diffusion (Jing et al., 2022). In particular, we see that MCF tends to do better in Recall than Torsional Diffusion, while the trend is reversed for Precision. However, the gap by which MCF outperforms Torsional Diffusion in Recall is larger than the one opposite trend for Precision. Again, it is important to note that MCF does not make any assumptions about the intrinsic geometric factors important in conformers like torsional angles and thus is simpler to train (*e.g.* it does not require a local substructure prediction model). We include detailed analysis of MCF results compared with Torsional Diff. in Appendix A.6.4. We also include experiments of transferring MCF trained on GEOM-DRUGS to GEOM-XL to the Appendix A.6.5.

	Recall				Precision			
	Coverage ↑		AMR ↓		Coverage ↑		AMR ↓	
	mean	median	mean	median	mean	median	mean	median
GeoDiff (Xu et al., 2022)	42.10	37.80	0.835	0.809	24.90	14.50	1.136	1.090
GeoMol (Ganea et al., 2021)	44.60	41.40	0.875	0.834	43.00	36.40	0.928	0.841
Torsional Diff. (Jing et al., 2022)	72.70	80.00	0.582	0.565	55.20	**56.90**	**0.778**	**0.729**
MCF (ours)	**78.15**	**86.03**	**0.526**	**0.501**	**55.86**	55.13	0.779	0.734

Table 2: Molecule conformer generation results on GEOM-DRUGS.

4 Conclusions

In this paper we introduced MCF, where we formulate the problem of molecular conformer generation as learning a distribution over continuous fields on molecular graphs. MCF is formulated specifically as a diffusion generative model over fields, which we show obtains comparable or better performance relative to state-of-the-art methods when evaluated on molecular generation across different benchmarks. Notably, MCF achieves these results without explicitly modeling geometric properties of molecules like torsional angles, which makes it simpler to understand and scale. MCF presents an exciting avenue for future research on scaling conformer generation to proteins and other macro molecular structures.

References

Ryan Abbott, Michael S Albergo, Denis Boyda, Kyle Cranmer, Daniel C Hackett, Gurtej Kanwar, Sébastien Racanière, Danilo J Rezende, Fernando Romero-López, Phiala E Shanahan, et al. Gauge-equivariant flow models for sampling in lattice field theories with pseudofermions. *Physical Review D*, 106(7):074506, 2022.

Ryan Abbott, Michael S Albergo, Aleksandar Botev, Denis Boyda, Kyle Cranmer, Daniel C Hackett, Gurtej Kanwar, Alexander GDG Matthews, Sébastien Racanière, Ali Razavi, et al. Normalizing flows for lattice gauge theory in arbitrary space-time dimension. *arXiv preprint arXiv:2305.02402*, 2023.

Yutong Bai, Jieru Mei, Alan L Yuille, and Cihang Xie. Are transformers more robust than cnns? *Advances in neural information processing systems*, 34:26831–26843, 2021.

Christoph Bannwarth, Sebastian Ehlert, and Stefan Grimme. Gfn2-xtb—an accurate and broadly parametrized self-consistent tight-binding quantum chemical method with multipole electrostatics and density-dependent dispersion contributions. *Journal of chemical theory and computation*, 15 (3):1652–1671, 2019.

Simon Batzner, Albert Musaelian, Lixin Sun, Mario Geiger, Jonathan P Mailoa, Mordechai Kornbluth, Nicola Molinari, Tess E Smidt, and Boris Kozinsky. E (3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials. *Nature communications*, 13(1):2453, 2022.

David Berthelot, Arnaud Autef, Jierui Lin, Dian Ang Yap, Shuangfei Zhai, Siyuan Hu, Daniel Zheng, Walter Talbot, and Eric Gu. Tract: Denoising diffusion models with transitive closure time-distillation. *arXiv preprint arXiv:2303.04248*, 2023.

Gabriele Corso, Hannes Stärk, Bowen Jing, Regina Barzilay, and Tommi Jaakkola. Diffdock: Diffusion steps, twists, and turns for molecular docking. *arXiv preprint arXiv:2210.01776*, 2022.

Tri Dao, Dan Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. Flashattention: Fast and memory-efficient exact attention with io-awareness. *Advances in Neural Information Processing Systems*, 35:16344–16359, 2022.

Y. Du, K. Collins, J. Tenenbaum, and V. Sitzmann. Learning signal-agnostic manifolds of neural fields. In *NeurIPS*, 2021.

E. Dupont, H. Kim, S. Eslami, D. Rezende, and D. Rosenbaum. From data to functa: Your data point is a function and you should treat it like one. In *ICML*, 2022a.

E. Dupont, Y. Teh, and A. Doucet. Generative models as distributions of functions. In *AISTATS*, 2022b.

Vijay Prakash Dwivedi, Chaitanya K Joshi, Thomas Laurent, Yoshua Bengio, and Xavier Bresson. Benchmarking graph neural networks. 2020.

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*, 34:13757–13769, 2021.

Daniele Grattarola and Pierre Vandergheynst. Generalised implicit neural representations. *arXiv preprint arXiv:2205.15674*, 2022.

Christoph Grebner, Johannes Becker, Svetlana Stepanenko, and Bernd Engels. Efficiency of tabu-search-based conformational search algorithms. *Journal of Computational Chemistry*, 32(10): 2245–2253, 2011.

D. Ha, A. Dai, and Q. Le. Hypernetworks. In *ICLR*, 2017.

Paul CD Hawkins. Conformation generation: the state of the art. *Journal of chemical information and modeling*, 57(8):1747–1756, 2017.

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.

Xiaoxin He, Bryan Hooi, Thomas Laurent, Adam Perold, Yann LeCun, and Xavier Bresson. A generalization of vit/mlp-mixer to graphs. *arXiv preprint arXiv:2212.13350*, 2022.

V Hernandez, JE Roman, A Tomas, and V Vidal. A survey of software for sparse eigenvalue problems. *Universitat Politecnica De Valencia, SLEPs technical report STR-6*, 2009.

J. Ho, A. Jain, and P. Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020.

Emiel Hoogeboom, Vıctor Garcia Satorras, Clément Vignac, and Max Welling. Equivariant diffusion for molecule generation in 3d. In *International conference on machine learning*, pp. 8867–8887. PMLR, 2022.

A. Jaegle, S. Borgeaud, J. Alayrac, et al. Perceiver io: A general architecture for structured inputs & outputs. In *ICLR*, 2022.

Bowen Jing, Gabriele Corso, Jeffrey Chang, Regina Barzilay, and Tommi Jaakkola. Torsional diffusion for molecular conformer generation. *Advances in Neural Information Processing Systems*, 35:24240–24253, 2022.

John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, et al. Highly accurate protein structure prediction with alphafold. *Nature*, 596(7873):583–589, 2021.

Gurtej Kanwar, Michael S Albergo, Denis Boyda, Kyle Cranmer, Daniel C Hackett, Sébastien Racaniere, Danilo Jimenez Rezende, and Phiala E Shanahan. Equivariant flow-based sampling for lattice gauge theory. *Physical Review Letters*, 125(12):121601, 2020.

D. Kingma and J. Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015.

Lukas Koestler, Daniel Grittner, Michael Moeller, Daniel Cremers, and Zorah Lähner. Intrinsic neural fields: Learning functions on manifolds. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part II*, pp. 622–639. Springer, 2022.

Jonas Köhler, Leon Klein, and Frank Noé. Equivariant flows: exact likelihood generative learning for symmetric densities. In *International conference on machine learning*, pp. 5361–5370. PMLR, 2020.

Juho Lee, Yoonho Lee, Jungtaek Kim, Adam Kosiorek, Seungjin Choi, and Yee Whye Teh. Set transformer: A framework for attention-based permutation-invariant neural networks. In *International conference on machine learning*, pp. 3744–3753. PMLR, 2019.

Derek Lim, Joshua Robinson, Lingxiao Zhao, Tess Smidt, Suvrit Sra, Haggai Maron, and Stefanie Jegelka. Sign and basis invariant networks for spectral graph representation learning. *arXiv preprint arXiv:2202.13013*, 2022.

Sohir Maskey, Ali Parviz, Maximilian Thiessen, Hannes Stärk, Ylli Sadikaj, and Haggai Maron. Generalized laplacian positional encoding for graph representation learning. *arXiv preprint arXiv:2210.15956*, 2022.

M Peter Nightingale and Cyrus J Umrigar. *Quantum Monte Carlo methods in physics and chemistry*. Number 525. Springer Science & Business Media, 1998.

J. Park, P. Florence, J. Straub, R. Newcombe, and S. Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *CVPR*, 2019.

Lars Ruddigkeit, Ruud Van Deursen, Lorenz C Blum, and Jean-Louis Reymond. Enumeration of 166 billion organic small molecules in the chemical universe database gdb-17. *Journal of chemical information and modeling*, 52(11):2864–2875, 2012.

Alvaro Sanchez-Gonzalez, Jonathan Godwin, Tobias Pfaff, Rex Ying, Jure Leskovec, and Peter Battaglia. Learning to simulate complex physics with graph networks. In *International conference on machine learning*, pp. 8459–8468. PMLR, 2020.

Nicholas Sharp, Souhaib Attaiki, Keenan Crane, and Maks Ovsjanikov. Diffusionnet: Discretization agnostic learning on surfaces. *ACM Transactions on Graphics (TOG)*, 41(3):1–16, 2022.

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.

J. Song, C. Meng, and S. Ermon. Denoising diffusion implicit models. In *ICLR*, 2021.

Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. 2023.

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.

Joseph L Watson, David Juergens, Nathaniel R Bennett, Brian L Trippe, Jason Yim, Helen E Eisenach, Woody Ahern, Andrew J Borst, Robert J Ragotte, Lukas F Milles, et al. De novo design of protein structure and function with rfdiffusion. *Nature*, pp. 1–3, 2023.

Stephen R Wilson, Weili Cui, Jules W Moskowitz, and Kevin E Schmidt. Applications of simulated annealing to the conformational analysis of flexible molecules. *Journal of computational chemistry*, 12(3):342–349, 1991.

Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? *arXiv preprint arXiv:1810.00826*, 2018.

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, 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. In *International Conference on Machine Learning*, pp. 11537–11547. PMLR, 2021b.

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.

Sheheryar Zaidi, Michael Schaarschmidt, James Martens, Hyunjik Kim, Yee Whye Teh, Alvaro Sanchez-Gonzalez, Peter Battaglia, Razvan Pascanu, and Jonathan Godwin. Pre-training via denoising for molecular property prediction. *arXiv preprint arXiv:2206.00133*, 2022.

Peiye Zhuang, Samira Abnar, Jiatao Gu, Alex Schwing, Josh Susskind, and Miguel Angel Bautista. Diffusion probabilistic fields. In *ICLR*, 2023.

A Appendix

A.1 Related Work

Recent works have tackled the problem of conformer generation using learning-based generative models. Simm & Hernández-Lobato (2019) and Xu et al. (2021b) develop two-stage methods which first generate interatomic distances following VAE framework and then predict conformers based on the distances. In Xu et al. (2021a), a normalizing flow approach is proposed as an alternative to VAEs. To avoid the accumulative errors from two-stage generation, Shi et al. (2021) implement score-based generative model to directly model the gradient of logarithm density of atomic coordinates. In GeoDiff (Xu et al., 2022), a diffusion model is used which focuses on crafting equivariant forward and backward processes with equivariant graph neural networks. In GeoMol (Ganea et al., 2021), the authors propose a regression objective coupled with an Optimal Transport loss to predict the torsional angles of bonds that assemble substructures of a molecule. Following this, Torsional Diffusion (Jing et al., 2022) proposed a diffusion model on the torsional angles of the bonds rather than a regression model used in Ganea et al. (2021). We further discuss the equivariance in conformer generation in Appendix A.3.

Our approach extends recent efforts in generative models for continuous functions in Euclidean space (Zhuang et al., 2023; Dupont et al., 2022b,a; Du et al., 2021), to functions defined over graphs (*e.g.* chemical structure of molecules). The term Implicit Neural Representation (INR) is used in these works to denote a parameterization of a single function (*e.g.* a single image in 2D) using a neural network that maps the function's inputs (*i.e.* pixel coordinates) to its outputs (*i.e.* RGB values). Different approaches have been proposed to learn distributions over fields in Euclidean space; GASP (Dupont et al., 2022b) leverages a GAN whose generator produces field data whereas a point cloud discriminator operates on discretized data and aims to differentiate real and generated functions. Two-stage approaches (Dupont et al., 2022a; Du et al., 2021) adopt a latent field parameterization (Park et al., 2019) where functions are parameterized via a hyper-network (Ha et al., 2017) and a generative model is learnt on the latent or INR representations. In addition, our approach also relates to recent work focusing on fitting a function (*e.g.* learning an INR) on a manifold using an intrinsic coordinate system (Koestler et al., 2022; Grattarola & Vandergheynst, 2022), and generalizes it to the problem of learning a probabilistic model over multiple functions defined on different manifolds/graphs. Intrinsic coordinate systems have also been used in Graph Transformers to tackle supervised learning tasks (Maskey et al., 2022; Sharp et al., 2022; He et al., 2022; Dwivedi et al., 2020).

Recent strides in the domain of protein folding dynamics have witnessed revolutionary progress, with modern methodologies capable of predicting crystallized 3D structures solely from amino-acid sequences using auto-regressive models like AlphaFold (Jumper et al., 2021). However, transferring these approaches seamlessly to general molecular data is fraught with challenges. Molecules present a unique set of complexities, manifested in their highly branched graphs, varying bond types, and chiral information, aspects that make the direct application of protein folding strategies to molecular data a challenging endeavor.

A.2 Limitations and Future Work

While MCF shows competitive performance in molecular conformer generation, it does encounter limitations and potential improvements for future explorations. One limitation is that our proposed method is computationally expensive. Extensive computations first stem from the Transformer-based (Vaswani et al., 2017) score network. In MCF, we use a PerceiverIO (Jaegle et al., 2022) as score network, an efficient transformer that allows for sub-quadratic compute. Other efficient transformer architectures and tricks like (Dao et al., 2022) can be used to improve training efficiency. The other factor is computational cost during inference. In MCF, we iterate 1000 timesteps to sample a conformer following DDPM (Ho et al., 2020). Efficient sampling like DDIM (Song et al., 2021), which can be seamlessly incorporated in MCF, as well as distillation approaches like (Song et al., 2023; Berthelot et al., 2023).

Another limitation could be the fact that MCF does not explicitly model the equivariance of the molecular system. Though experiments in the paper have demonstrated that equivariance may not be necessary to achieve competitive performance on conformer generation. MCF may not perform as well as conformer generation when applied to problems with limited data or related to sequential

problems like molecular dynamics (MD) simulations. In future work, we plan to extend MCF to conditional inference. For example, molecular docking can be formulated as conformer generation problem conditioned on proteins (Corso et al., 2022). Also, current framework can be expanded to *de novo* drug designs where no molecule information is provided (Hoogeboom et al., 2022). Besides, scaling up our model to large molecules, like proteins, can be of great interest. MCF by nature provides the flexibility to generate from partially observed sample, which can be suitable for designing proteins with known functional motifs (Watson et al., 2023).

A.3 Equivariance in Conformer Generation

Equivariance has become an important topic of study in generative models (Abbott et al., 2023, 2022; Kanwar et al., 2020). In particular, enforcing equivariance as an explicit inductive bias in neural networks can often lead to improved generalization (Köhler et al., 2020) by constraining the space of functions that can be represented by a model. On the other hand, recent literature shows that models that can learn these symmetries from data rather than explicitly enforcing them (*e.g.* transformers vs CNNs) tend to perform better as they are more amenable to optimization (Bai et al., 2021).

Equivariance also plays an interesting role in conformer generation. On one hand, it is important when training likelihood models of conformers, as the likelihood of a conformer is invariant to roto-translations (Köhler et al., 2020). On the other hand, when training models to generate conformers given a molecular graph, explicitly baking roto-translation equivariance might not be as necessary. This is because the intrinsic structure of the conformer encodes far more information about its properties than the extrinsic coordinate system (eg. rotation and translation) in which the conformer is generated (Ruddigkeit et al., 2012). In addition, recent approaches for learning simulations on graphs (Sanchez-Gonzalez et al., 2020) or pre-training models for molecular prediction tasks (Zaidi et al., 2022) have relied on non-equivariant architectures.

We follow this trend and empirically show that explicitly enforcing roto-translation equivariance is not a strong requirement for generalization. Furthermore, we also show that approaches that do not explicitly enforce roto-translation equivariance (like ours) can match or outperform approaches that do.

A.4 Implementation details

In this section we describe implementation details for all our experiments. We also provide hyper-parameters and settings for the implementation of the score field network ϵ_θ and compute used for each experiment in the paper.

A.4.1 Score Field Network implementation details

The time-step t is incorporated into the score computation by concatenating a positional embedding representation of t to the context and query sets. The specific PerceiverIO settings used in all quantitatively evaluated experiments are presented in Tab. 3. Practically, the MCF network consists of 6 transformer blocks, each containing 1 cross-attention layer and 2 self-attention layers. Each of these layers has 4 attention heads. An Adam (Kingma & Ba, 2015) optimizer is employed during training with a learning rate of $1e-4$. We use EMA with a decay of 0.9999. A modified version of the publicly available repository is used for PerceiverIO [2]. Since molecules have different number of atoms, we set the number of context and query sets as the number of atoms during training and inference.

A.4.2 Compute

Each model was trained on an machine with 8 Nvidia A100 GPUs, we trained models for 500 epochs.

A.5 Sampling and Inference Algorithm

For training, we uniformly sample context and query sets from $f_0 \sim \text{Uniform}(q(f_0))$ and only corrupt their signal using the forward process in Eq. 1 and Eq. 2. We train the score field network ϵ_θ to denoise the signal portion of the query set, given the context set. During sampling, to generate

[2] https://huggingface.co/docs/transformers/model_doc/perceiver

Hyper-parameter	GEOM-QM9	GEOM-DRUGS	Ablation GEOM-QM9
#eigenfuncs (k)	28	32	28
#freq pos. embed t	64	64	64
#latents	512	512	512
#dim latents	512	256	256
#blocks	6	6	6
#dec blocks	2	2	2
#self attends per block	2	2	2
#self attention heads	4	4	4
#cross attention heads	4	4	4
batch size	64	64	64
lr	$1e-4$	$1e-4$	$1e-4$
epochs	250	100	250

Table 3: Hyperparameters and settings for MCF on different datasets.

a conformer fields $f_0 \sim p_\theta(f_0)$ we first define a query set $\mathbf{Q}_T = [\varphi(\mathcal{V}_q),\ \mathbf{Y}_{(q,T)} \sim \mathcal{N}(\mathbf{0},\ \mathbf{I})]$ of random values to be de-noised. We set the context set to be a random subset of the query set. We use the context set to denoise the query set and follow ancestral sampling as in the vanilla DDPM (Ho et al., 2020). Note that during inference the eigen-function representation $\varphi(v)$ of the context and query sets does not change, only their corresponding signal value (*e.g.* their 3D position). Alg. 1 and Alg. 2 show the sampling and inference algorithms of our proposed MCF respectively.

Algorithm 1 Training

$\Delta_{\mathcal{G}}\varphi_i = \varphi_i\lambda_i$ // Compute Laplacian eigenvectors
repeat
  $(\mathbf{C}_0, \mathbf{Q}_0) \sim \text{Uniform}(q(f_0))$
  $t \sim \text{Uniform}(\{1, \dots, T\})$
  $\epsilon_c \sim \mathcal{N}(\mathbf{0}, \mathbf{I}),\ \epsilon_q \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
  $\mathbf{C}_t = [\varphi(\mathcal{V}_c), \sqrt{\bar{\alpha}_t}\mathbf{Y}_{(c,0)} + \sqrt{1-\bar{\alpha}_t}\epsilon_c]$
  $\mathbf{Q}_t = [\varphi(\mathcal{V}_q), \sqrt{\bar{\alpha}_t}\mathbf{Y}_{(q,0)} + \sqrt{1-\bar{\alpha}_t}\epsilon_q]$
  Take gradient descent step on $\nabla_\theta \left\|\epsilon_q - \epsilon_\theta(\mathbf{C}_t, t, \mathbf{Q}_t)\right\|^2$
until converged

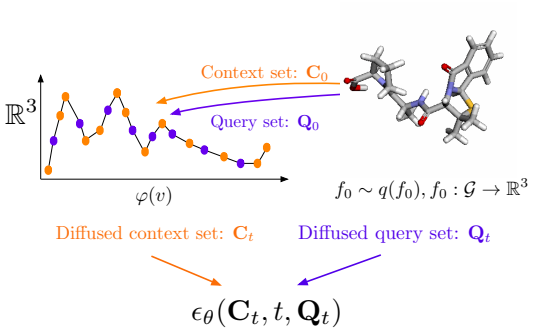


Figure 3: **Left:** MCF training algorithm. **Right**: Visual depiction of a training iteration for a conformer field. See Sect. 2 for definitions.

Algorithm 2 Sampling

$\Delta_{\mathcal{G}}\varphi_i = \varphi_i\lambda_i$ // LBO eigen-decomposition
$\mathbf{Q}_T = [\varphi(\mathcal{V}_q), \mathbf{Y}_{(q,t)} \sim \mathcal{N}(\mathbf{0}_q, \mathbf{I}_q)]$
$\mathbf{C}_T \subseteq \mathbf{Q}_T$ ▷ Random subset
for $t = T, \dots, 1$ **do**
  $\boldsymbol{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ if $t > 1$, else $\boldsymbol{z} = \mathbf{0}$
  $\mathbf{Y}_{(q,t-1)} = \frac{1}{\sqrt{\alpha_t}}\left(\mathbf{Y}_{(q,t)} - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(\mathbf{C}_t, t, \mathbf{Q}_t)\right) + \sigma_t\boldsymbol{z}$
  $\mathbf{Q}_{t-1} = [\varphi(\mathcal{V}_q), \mathbf{Y}_{(q,t-1)}]$
  $\mathbf{C}_{t-1} \subseteq \mathbf{Q}_{t-1}$ ▷ Same subset as in step 2
end for
return f_0 evaluated at coordinates $\varphi(\mathcal{V}_q)$

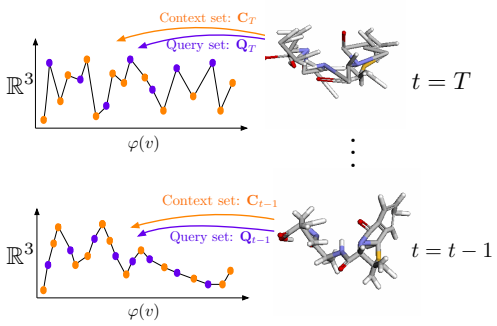


Figure 4: **Left:** MCF sampling algorithm. **Right**: Visual depiction of the sampling process of a conformer field.

A.6 Additional experiments

In this section we include additional experiments ablating architecture choices, as well as prediction the ensemble properties of generated conformers.

A.6.1 Ablation experiments

In this section we provide an ablation study over the key design choices of MCF. We run all our ablation experiments on the GEOM-QM9 dataset following the settings in GeoMol (Ganea et al., 2021) and Torsional Diffusion (Jing et al., 2022) and described in Sect. 3.2. In particular we study: (i) how does performance behave as a function of the number of Laplacian eigenvectors used in $\varphi(v)$. (ii) How does the model perform without atom features (*e.g.* how predictable conformers are given only the graph topology, without using atom features). Results in Tab. 4 show that the graph topology $\mathcal{G}$ encodes a surprising amount of information for sampling reasonable conformers in GEOM-QM9, as shown in row 2. In addition, we show how performance of MCF changes as a function of the number of eigen-functions k. Interestingly, with as few as $k = 2$ eigen-functions MCF is able to generate consistent accurate conformer.

		Recall				Precision			
		Coverage ↑		AMR ↓		Coverage ↑		AMR ↓	
k	Atom Features	mean	median	mean	median	mean	median	mean	median
28	YES	94.86	100.00	0.125	0.081	91.49	100.00	0.175	0.122
28	NO	90.70	100.00	0.187	0.124	79.82	93.86	0.295	0.213
16	YES	**94.87**	100.00	**0.139**	0.093	**87.54**	100.00	0.220	0.151
8	YES	94.28	100.00	0.162	0.109	84.27	100.00	0.261	0.208
4	YES	94.57	100.00	0.145	0.093	86.83	100.00	0.225	0.151
2	YES	93.15	100.00	0.152	**0.088**	86.97	100.00	**0.211**	**0.138**

Table 4: Experiments on GEOM-QM9 with different numbers of eigenvectors. In these experiments, we use a smaller model than what we have in Table 1

A.6.2 Architectural choices

To further investigate the design choices of architecture in proposed MCF, we include additional experiments on GEOM-QM9 as shown in Tab. 5. To investigate the effectiveness of using Laplacian eigenvectors as positional embedding, we leverage SignNet (Lim et al., 2022) as the positional embedding, which explicitly models symmetries in eigenvectors. Using SignNet does not benefit the performance when compared with the standard MCF. Though adding edge attributes in SignNet achieves better performance than SignNet alone, the performance is still not rival. Also, it's worth mentioning that SignNet includes graph neural networks (Xu et al., 2018) and Set Transformer (Lee et al., 2019) which makes training less efficient.

In addition, we also report results using a vanilla Transformer encoder-decoder (Vaswani et al., 2017) as the backbone instead of PerceiverIO (Jaegle et al., 2022). The Transformer contains 6 encoder layers and 6 decoder layers with 4 attention heads. Other model hyperparameters follow the same setting as listed in Tab. 3. It is indicated that Transformer is performing worse than PerceiverIO on conformer generation, which validates the design choice of architecture in MCF.

		Precision				Recall			
		Coverage ↑		AMR ↓		Coverage ↑		AMR ↓	
		mean	median	mean	median	mean	median	mean	median
MCF	SignNet	94.1	100.0	0.153	0.098	87.5	100.0	0.222	0.152
MCF	SignNet (edge attr.)	95.3	100.0	0.143	0.091	90.2	100.0	0.197	0.135
MCF	Transformer	94.3	100.0	0.159	0.111	90.7	100.0	0.202	0.136
MCF	PerceiverIO	**96.57**	**100.00**	**0.107**	**0.072**	**94.60**	**100.00**	**0.130**	**0.084**

Table 5: Molecule conformer generation results with different network architectures on GEOM-QM9.

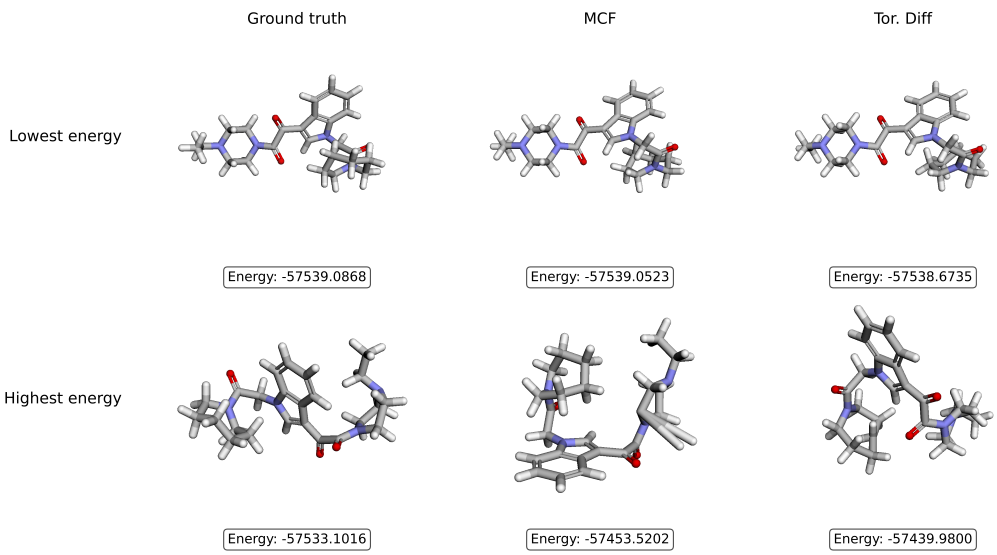


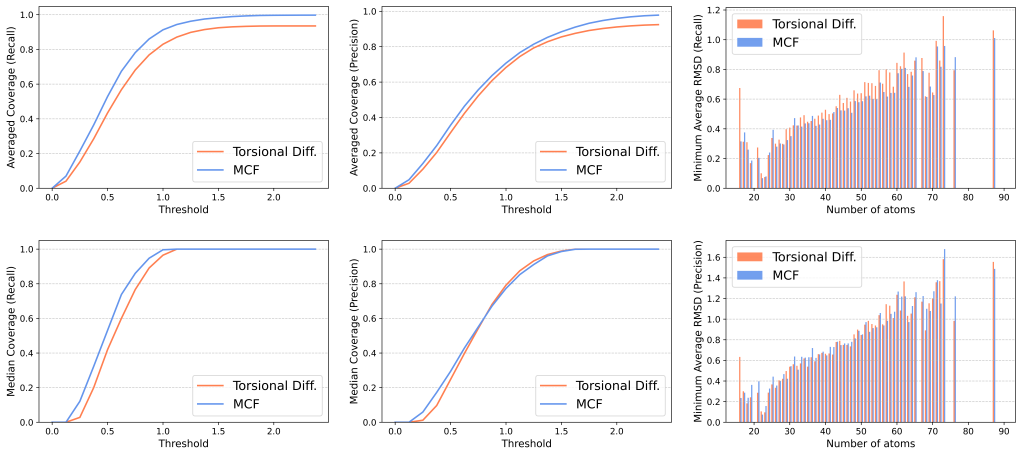
Figure 5: Comparison of conformers from ground truth, MCF, and Torsional Diffusion with energy calculated with xTB (in kcal/mol).

A.6.3 Ensemble property prediction

To fully assess the quality of generated conformers we also compute chemical property resemblance between the synthesized and the authentic ground truth ensembles. We select a random group of 100 molecules from the GEOM-DRUGS and produce a minimum of $2K$ and a maximum of 32 conformers for each molecule following (Jing et al., 2022). Subsequently, we undertake a comparison of the Boltzmann-weighted attributes of the created and the true ensembles. To elaborate, we calculate the following characteristics using xTB (as documented by (Bannwarth et al., 2019)): energy (E), dipole moment (μ), the gap between HOMO and LUMO ($\Delta\epsilon$), and the lowest possible energy, denoted as E_{min}. Since we don't have the access to the exact subset of DRUGS used in Jing et al. (2022), we randomly pick three subsets and report the averaged and standard deviation. The results are listed in Tab. 6. Our model achieves the lowest error on E_{min} when compared with other baselines, which demonstrates that MCF is succeeds at generating stable conformers that are very close to the ground states. This could root from the fact that MCF doesn't rely on rule-based cheminfomatics methods and the model learns to better model stable conformers from data. Besides, MCF achieves competitive performance on μ and $\Delta\epsilon$. However, the error of E is high compared to the rest of approaches, meaning that though MCF performs well in generating samples close to ground states, it may also generate conformers with high energy that are not plausible in the dataset. In future work we plan to further study the reasons for this result. Fig. 5 show examples of generated conformers from our MCF and Torsional Diffusion when compared with ground truth. MCF generates conformers where the ground state is closer to the ground truth than Torsional Diffusion. However, both methods can generate conformers when higher energy than ground truth data.

	E	μ	$\Delta\epsilon$	E_{min}
OMEGA	0.68	0.66	0.68	0.69
GeoDiff	0.31	0.35	0.89	0.39
GeoMol	0.42	0.34	0.59	0.40
Torsional Diff.	**0.22**	0.35	**0.54**	0.13
MCF (ours)	0.71±0.06	**0.31±0.05**	0.60±0.06	**0.04±0.00**

Table 6: Median averaged errors of ensemble properties between sampled and generated conformers (E, $\Delta\epsilon$, E_{min} in kcal/mol, and μ in debye).



(a) (b) (c)

Figure 6: (a) Recall coverage metric as a function of the threshold distance. MCF outperforms Torsional Diffusion across the full spectrum of thresholds. (b) Precision coverage as a function of the threshold distance. We see that MCF outperforms Torsional Diffusion for median coverage at low thresholds, which shows that MCF better captures that fine intrinsic geometric structure of conformers. (c) Minimum averaged RMSD (lower is better) of recall and precision as a function of the number of atoms in molecules.

A.6.4 Analysis of Results on GEOM-DRUGS

In Fig. 6 we show a breakdown of the performance on GEOM-DRUGS of MCF vs. Torsional diffusion (Jing et al., 2022) as a function of the threshold distance, as well as a function of the number of atoms in molecules. MCF outperforms Torsional Diffusion across the full spectrum of thresholds. When looking at Precision metrics MCF outperforms Torsional Diffusion in median coverage at lower thresholds, which shows that MCF better captures the fine intrinsic geometric structure of conformers. We also include experiments of transferring MCF trained on GEOM-DRUGS to GEOM-XL to the Appendix A.6.5.

A.6.5 Generalization to GEOM-XL

We now turn to the task of evaluating how well a model trained on GEOM-DRUGS transfers to unseen molecules with large numbers of atoms. As proposed in Torsional Diff. (Jing et al., 2022) we use the GEOM-XL dataset, which is a subset of GEOM-MoleculeNet that contains all species with more than 100 atoms, which is a total of 102 molecules. Note that this evaluation not only tests the capacity of models to generalize to molecules with large number of atoms but also serves as an out-of-distribution generalization experiment.

In Tab. 7 we report AMR for both precision and recall and compare with GeoDiff (Xu et al., 2022), GeoMol (Ganea et al., 2021) and Torsional Diffusion (Jing et al., 2022). In particular, when taking the numbers directly from (Jing et al., 2022), Torsional Diffusion performs best closely followed by MCF. However, in running the checkpoint provided by Torsional Diffusion and following their validation process we found that 25 molecules failed to be generated, this is due to the fact that Torsional Diffusion generates torsional angles conditioned on the molecular graph $\mathcal{G}$ and the local structures obtained from RDKit. In some cases, RDKit can fail to find local structures and Torsional Diffusion cannot generate conformers in this case. As a result, when evaluating both approaches in exactly the same molecules MCF are Torsional Diffusion are extremely close in terms of generalization performance, with Torsional Diffusion leading by a very narrow margin.

A.7 Continuous conformers

Molecules are ubiquitously represented as graphs $\mathcal{G}$ where vertices $\mathcal{V}$ of the graph represents atoms and their properties and edges represents bonds that encode interactions between atoms. These bonds

	AMR-P ↓		AMR-R ↓		# mols
	mean	median	mean	median	
GeoDiff	2.92	2.62	3.35	3.15	102
GeoMol	2.47	2.39	3.30	3.14	102
Torsional Diff. (Jing et al., 2022)	**2.05**	**1.86**	**2.94**	**2.78**	102
MCF (ours)	2.23	1.92	3.26	2.89	102
Torsional Diff. (our eval)	**1.93**	1.86	**2.84**	**2.71**	77
MCF (ours)	1.97	**1.85**	2.98	2.72	77

Table 7: Generalization results on GEOM-XL. MCF obtains comparable results to Torsional Diffusion Jing et al. (2022), without making any explicit assumptions about the geometric structure of molecules.

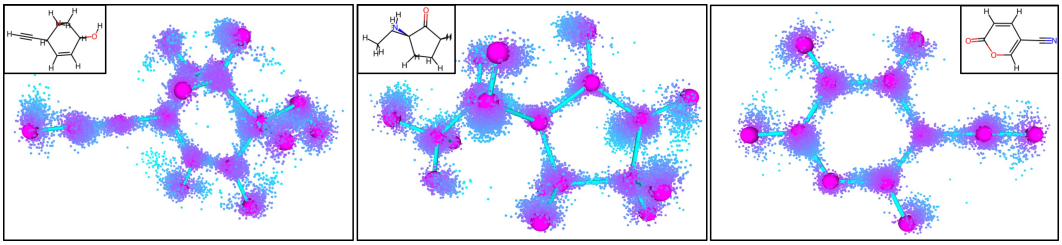


Figure 7: Continuously evaluating generated conformer fields for different molecules in GEOM-QM9.

encode electron interactions between atoms but also serve to describe the nature and characteristics of such interactions (*i.e.* covalent bonds, ionic bonds, metallic bonds, etc,). In particular, the molecule's conformer is generated only for discrete atoms (*e.g.* vertices in $\mathcal{G}$). However, since MCF encodes continuous conformer fields it can be *continuously evaluated* in $\mathcal{G}$. For example, we can evaluate the sampled conformer fields for points along bonds. In order to do this, for a point p in a bond connecting atoms (v_i, v_j) we linearly interpolate the Laplacian eigenvector representation of it's endpoints $\varphi(p) = \alpha\varphi(v_i) + (1-\alpha)\varphi(v_j)$, we then feed this interpolated Laplacian eigenvector into the model to sample its 3D position in the conformer field. We visualize results in Fig. 1 and 7. We generated this visualizations an MCF model trained on GEOM-QM9 without atom features. Note that while MCF is never trained on points along molecular bonds, it manages to generate plausible 3D positions for such points.

This flexibility to evaluate conformers continuously in $\mathcal{G}$ opens a realm of possibilities. For example, at this subatomic scales, the paths that electrons take are not well-defined tracks, but rather regions of space where they are most likely to be found, represented by probability density functions. These electron clouds form the bonds between atoms, and their shapes can vary quite a bit depending on the bond type, being sometimes symmetrical and other times quite complex and diffuse. MCF enables the possibility of using additional training data from Quantum Monte Carlo methods (Nightingale & Umrigar, 1998) to capture the probability density of electron clouds.

A.8 Additional visualization

In the supplementary material we provide videos showing the iterative inference process of MCF on different molecules in GEOM-QM9, GEOM-DRUGS and GEOM-XL. Finally in Fig. 8 we show GT and generated conformers from MCF for a molecule in GEOM-DRUGS.

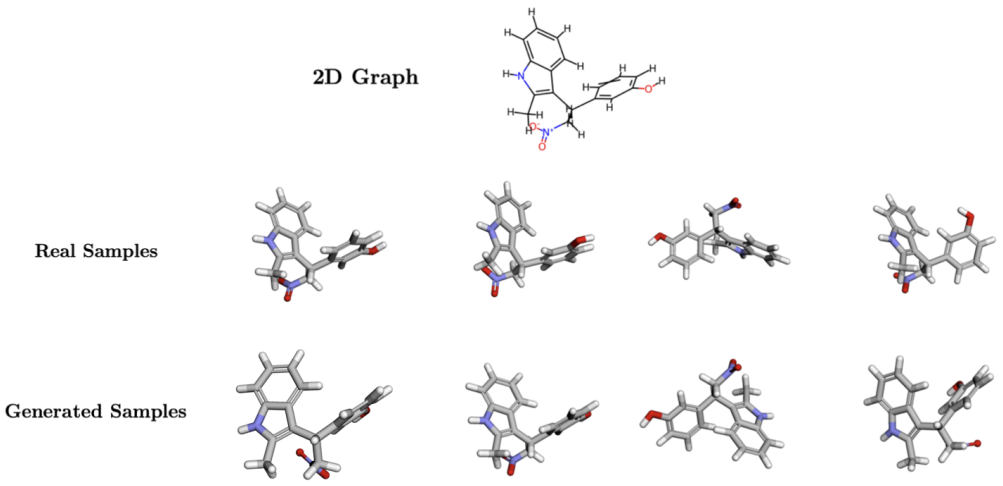


Figure 8: GT and generated conformers from MCF for a molecule in GEOM-DRUGS.