CoarsenConf: Equivariant Coarsening with Aggregated Attention for Molecular Conformer Generation

Danny Reidenbach
UC Berkeley
dreidenbach@berkeley.edu

Aditi S. Krishnapriyan
UC Berkeley
aditik1@berkeley.edu

Abstract

Molecular conformer generation (MCG) is an important task in cheminformatics and drug discovery. The ability to efficiently generate low-energy 3D structures can avoid expensive quantum mechanical simulations, leading to accelerated virtual screenings and enhanced structural exploration. Several generative models have been developed for MCG, but many struggle to consistently produce high-quality conformers. To address these issues, we introduce CoarsenConf, which coarse-grains molecular graphs based on torsional angles and integrates them into an SE(3)-equivariant hierarchical variational autoencoder. Through equivariant coarse-graining, we aggregate the fine-grained atomic coordinates of subgraphs connected via rotatable bonds, creating a variable-length coarse-grained latent representation. Our model uses a novel aggregated attention mechanism to restore fine-grained coordinates from the coarse-grained latent representation, enabling efficient generation of accurate conformers. Furthermore, we evaluate the chemical and biochemical quality of our generated conformers on multiple downstream applications, including property prediction and oracle-based protein docking. Overall, CoarsenConf generates more accurate conformers compared to prior models.

1 Introduction

Molecular conformer generation (MCG) is a fundamental task in computational chemistry. The objective is to predict stable low-energy 3D molecular structures, known as conformers. Accurate molecular conformations are crucial for various applications that depend on precise spatial and geometric qualities, including drug discovery and protein docking. For MCG, traditional physics-based methods present a trade-off between speed and accuracy. Quantum mechanical methods are more accurate but computationally slow, while stochastic cheminformatics-based methods like RDKit EKTDG [Riniker and Landrum, 2015] provide more efficient but less accurate results. As the difficulty of computing low-energy structures increases with the number of atoms and rotatable bonds in a molecule, there has been interest in developing machine learning (ML) methods to generate accurate conformer predictions efficiently.

We present CoarsenConf, a novel conditional hierarchical VAE. CoarsenConf learns a coarse-grained (CG) or subgraph-level latent distribution for SE(3)-equivariant conformer generation. CoarsenConf aggregates information from fine-grained (FG) atomic coordinates to create a flexible subgraph-level representation, improving the accuracy of conformer generation. Unlike prior MCG methods, CoarsenConf generates low-energy conformers with the ability to model atomic coordinates (FG and CG), distances, and torsion angles directly via variable-length coarse-graining. To our knowledge, this is the first method to use coarse-graining in the context of MCG. CoarsenConf is the first model capable of handling variable-length coarse-to-fine generation using an *Aggregated Attention* strategy. CoarsenConf employs a single flexible variable-length node-level latent representation that can

NeurIPS 2023 Generative AI and Biology Workshop.

uniquely represent molecules of any size with any number of coarse-grained nodes. Furthermore, variable-length coarse-graining circumvents having to train separate generative models for each number of CG beads to represent the same molecular dataset accurately (100+ models for MCG), which is a limitation of fixed-length methods [Yang and Gomez-Bombarelli, 2023].

We predominantly outperform prior methods on GEOM-QM9 and GEOM-DRUGS for RMSD precision and property prediction benchmarks [Axelrod and Gómez-Bombarelli, 2022]. We also produce a lower overall RMSD distribution across all conformers, achieving this with an order of magnitude less training time compared to prior methods. We evaluate CoarsenConf on multiple downstream applications to assess the chemical and biochemical quality of our generated conformers, including oracle-based protein docking [Huang et al., 2021] (the affinity of generated conformers to bind to specific protein pockets) under both flexible and rigid conformational energy minimizations. Despite lacking prior knowledge about the protein or the downstream task, CoarsenConf generates significantly better binding ligand conformers for known protein binding sites when compared to prior MCG methods for both oracle scenarios. We include an extensive discussion on our design decisions and further relevant work in Appendix §B. We discuss prominent 2D and 3D equivariant autoregressive molecular generation, MCG methods, protein docking, and structure-based drug discovery (SBDD) techniques, as well as our formal definition of SE(3)-equivariance.

2 Methods

Learning Framework. CoarsenConf is a conditional generative model that learns $p(X|\mathcal{R})$ where X is the low-energy 3D conformation, and $\mathcal{R}$ is the RDKit approximate conformation. Specifically, we optimize $p(X|\mathcal{R})$ by maximizing its variational lower bound with an approximate conditional posterior distribution $q_\phi(z|X, \mathcal{R})$ and learned conditional prior $p_\psi(z|\mathcal{R})$:

$$\log p(X|\mathcal{R}) \geq \underbrace{\mathbb{E}_{q_\phi(z|X,\mathcal{R})} \log p_\theta(X|\mathcal{R}, z)}_{\mathcal{L}_{\text{reconstruction}}} + \underbrace{\mathbb{E}_{q_\phi(z|X,\mathcal{R})} \log \frac{p_\psi(z|\mathcal{R})}{q_\phi(z|X,\mathcal{R})}}_{\mathcal{L}_{\text{latent regularization}}} + \mathcal{L}_{\text{auxiliary}}, \tag{1}$$

where $q_\phi(z|X, \mathcal{R})$ is the hierarchical equivariant encoder model, $p_\theta(X|\mathcal{R}, z)$ is the equivariant decoder model to recover X from $\mathcal{R}$ and z, and $p_\psi(z|\mathcal{R})$ is the learned prior distribution. The reconstruction loss, $\mathcal{L}_{\text{recon.}}$, is implemented as MSE$(\mathcal{A}(X_{true}, X_{model}), X_{model})$, where $\mathcal{A}$ is the Kabsch alignment function that provides an optimal rotation matrix and translation vector to minimize the mean squared error (MSE) [Kabsch, 1993]. The second term, $\mathcal{L}_{\text{reg.}}$, can be viewed as a regularization over the latent space and is implemented as $\beta D_{KL}(q_\phi(z|X, \mathcal{R}) \parallel p_\psi(z|\mathcal{R}))$ [Higgins et al., 2017]. More details on the geometric auxiliary loss function are in Appendix §C.

Encoder Architecture. CoarsenConf's encoder, shown in Appendix Fig. 4(I), operates over SE(3)-invariant atom features $h \in R^{n \times D}$, and SE(3)-equivariant atomistic coordinates $x \in R^{n \times 3}$. A single encoder layer is composed of three modules: fine-grained, pooling, and coarse-grained. Full equations for each module can be found in Appendix §E.1, §E.2, §E.3, respectively. The encoder module takes in x and h from the ground truth and RDKit conformer and creates coarse-grained latent representations for each, Z and $\tilde{Z} \in R^{N \times F \times 3}$, where N is the number of CG beads, and F is the latent dimensions. Z then undergoes standard VAE parameterization defined in Appendix §F.

We sample from the learned posterior (training) and learned prior (inference) to get $Z = \mu + \epsilon\sigma$, where ϵ is noise sampled from a standard Gaussian distribution as the input to the decoder. Given Z is still in CG space, we need to perform variable-length backmapping to convert back to FG space so that we can further refine the atom coordinates to generate the low energy conformer. The variable-length aspect is crucial because every molecule can be coarsened into a different number of beads, and there is no explicit limit to the number of atoms a single bead can represent. Unlike CGVAE [Wang et al., 2022], which requires training a separate model for each choice in CG granularity N, CoarsenConf is capable of reconstructing FG coordinates from any N (illustrated in Appendix Fig. 4(III) and Fig. 3).

Decoder Architecture: Channel Selection. CGVAE defines the process of channel selection (CS) as selecting the top-k latent channels, where k is the number of atoms in a CG bead of interest. Instead of discarding all learned information in the remaining $F - k$ channels in the latent representation, we use a novel aggregated attention mechanism. This mechanism learns the optimal mixing of channels to reconstruct the FG coordinates, and is illustrated in Appendix Fig. 5. The attention operation allows us to actively query our latent representation for the number of atoms we need, and draw upon similarities to the learned RDKit approximation that has been distilled into the latent space through the hierarchical encoding process. Channel selection translates the CG latent tensor $Z \in R^{N \times F \times 3}$ into FG coordinates $x_{CS} \in R^{n \times 3}$.

Decoder Architecture: Coordinate Refinement. Once channel selection is complete, we have effectively translated the variable-length CG representation back into the desired FG form. From here, we explore two methods of decoding that use the same underlying architecture. x_{CS} can be passed in a single step through the decoder or can be grouped into its corresponding CG beads, but left in FG coordinates to do a bead-wise autoregressive (AR) generation of final coordinates (Fig. 4(IV)). CoarsenConf is the first MCG method to explore AR generation. Unlike prior 3D AR methods, CoarsenConf does not use a pre-calculated molecular fragment vocabulary, and instead conditions directly on a learned mixture of previously generated 3D coordinates and invariant atom features.

The decoder architecture is similar to the EGNN-based FG module in the encoder, but has one key difference. Instead of learning raw atom coordinates, we learn to predict the difference between the RDKit reference and ground truth conformations. As the goal of MCG is to model the conditional distribution $p(X|\mathcal{R})$, we simplify the learning objective by setting $X = \mathcal{R} + \Delta X$, and learn the optimal distortion ΔX from the RDKit approximation. The simplification follows, as we have ensured $\mathcal{L}_{\text{recon.}}$ is no worse than that of the RDKit approximation, which is trivial to obtain, compared to the cost of model training and inference.

See Appendix §G for formal discussions of the decoder architecture and message-passing equations.

3 Experiments

We evaluate MCG on RMSD spatial accuracy (§3.1), property prediction (§3.1), and biochemical quality through flexible (§M.1) and rigid (§M.2) oracle-based protein docking. We include the following models for comparison with CoarsenConf, which are previous MCG methods that use the same train/test split: Torsional Diffusion (TD) [Jing et al., 2022], GeoMol (GM) [Ganea et al., 2021], and when possible, GeoDiff (GD) [Xu et al., 2022]. For our model configuration and compute resource breakdown, see Appendix §H. The details of CoarsenConf's initial RDKit structures, as well as how CoarsenConf learns to avoid the distribution shift found in TD, can be found in Appendix §I.

3.1 GEOM Benchmarks: 3D Coordinate RMSD

We use the GEOM dataset [Axelrod and Gómez-Bombarelli, 2022], consisting of QM9 (average 11 atoms) and DRUGS (average 44 atoms), to train and evaluate our model. We use the same train/val/test splits from Ganea et al. [2021] (QM9: 106586/13323/1000 and DRUGS: 243473/30433/1000).

Problem setup. We report the average minimum RMSD (AMR) between ground truth and generated conformers, and Coverage for Recall and Precision. Coverage is defined as the percentage of conformers with a minimum error under a specified AMR threshold. Recall matches each ground truth conformer to its closest generated structure, and Precision measures the overall spatial accuracy of the generated conformers. Following Jing et al. [2022], we generate two times the number of ground truth conformers for each molecule. More formally, for $K = 2L$, let $\{C_l^*\}_{l \in [1,L]}$ and $\{C_k\}_{k \in [1,K]}$ respectively be the sets of ground truth and generated conformers:

$$\begin{aligned} \text{COV-Precision} &:= \frac{1}{K} \left| \{k \in [1..K] : \min_{l \in [1..L]} \text{RMSD}(C_k, C_l^*) < \delta\} \right|, \\ \text{AMR-Precision} &:= \frac{1}{K} \sum_{k \in [1..K]} \min_{l \in [1..L]} \text{RMSD}(C_k, C_l^*), \end{aligned} \tag{2}$$

where δ is the coverage threshold. The recall metrics are obtained by swapping ground truth and generated conformers. We also report the full RMSD error distributions, as the AMR only gives a small snapshot into overall error behavior. For an in-depth discussion on the advantages and limitations of these metrics can be found in Appendix §J.

Results. In Tab. 1, we outperform all models on QM9, and yield competitive results with TD on DRUGS when using an optimal transport (OT) loss (see Appendix §K- §L for more details). CoarsenConf also achieves the lowest overall error distribution, as seen in Fig. 1. CoarsenConf-OT uses an OT loss with the same decoder architecture as in Appendix Fig. 4, but is no longer autoregressive (see Appendix Eq. 11 for a formal definition). We also see in Appendix Fig. 7 that the recall is heavily dependent on the sampling budget, as performance gaps shrink from ~50% to

Table 1: Quality of ML generated conformer ensembles for the GEOM-QM9 ($\delta = 0.5$Å) and GEOM-DRUGS ($\delta = 0.75$Å) test set in terms of Coverage (%) and Average RMSD (Å) Precision. Bolded results are the best, and the underlined results are second best. See Appendix §K- §L for more details.

	QM9-Precision				DRUGS-Precision			
	Coverage ↑		AMR ↓		Coverage ↑		AMR ↓	
Method	Mean	Med	Mean	Med	Mean	Med	Mean	Med
GeoDiff	-	-	-	-	23.7	13.0	1.131	1.083
GeoMol	75.9	**100.0**	0.262	0.233	40.5	33.5	0.919	0.842
Torsional Diffusion ($\ell = 2$)	78.4	**100.0**	0.222	0.197	**52.1**	**53.7**	**0.770**	0.720
CoarsenConf-OT	**80.2**	**100.0**	**0.149**	**0.107**	52.0	52.1	0.836	**0.694**

~5%. CoarsenConf-OT was trained for 15 hours (2 epochs) compared to TD's 11 days, both on a single A6000. Furthermore, when limited to the same equivariance, CoarsenConf-OT performs predominantly better (Appendix Tab. 6). As CoarsenConf (autoregressive, no OT loss) results in a lower overall DRUGS RMSD distribution (Fig. 1), we use it for the remaining downstream tasks.

3.2 GEOM Benchmarks: Property Prediction

Problem setup. We generate and relax min(2L, 32) conformers (L ground truth) for 100 molecules from GEOM-DRUGS using GFN2-xTB with the BFGS optimizer. We then predict various properties, including Energy (E), HOMO-LUMO Gap ($\Delta\epsilon$), minimum Energy (E_{min}) in kcal/mol, and dipole moment (μ) in debye via xTB [Bannwarth et al., 2019]. The mean absolute error of the generated ensemble properties compared to ground truth is reported.

Results. Tab. 2 demonstrates CoarsenConf's ability to generate the lowest energy structures with the most accurate chemical properties. For further discussions, please see Appendix §J.

3.3 Flexible Oracle-based Protein Docking

We evaluate MCG models, pretrained on GEOM-DRUGS, using nine protein docking oracle functions provided by the Therapeutics Data Commons (TDC) [Huang et al., 2021].

Problem setup. Starting with a known 2D ligand[1] molecule, protein, and desired 3D protein binding pocket, we measure conformer quality by comparing the predicted binding affinity of generated conformers of each MCG method. See §M.1 for full details.

Results. CoarsenConf significantly outperforms prior MCG methods on the TDC oracle-based affinity prediction task (Tab. 3). CoarsenConf generates the best ligand conformers for 8/9 tested proteins, with improvements of up to 53% compared to the next best method. CoarsenConf is 1.46 kcal/mol better than all methods when averaged over all 9 proteins, which corresponds to a 14.4% improvement on average compared to the next best method.

[1] No ground truth 3D structures. All ligand SMILES taken from Protein Data Bank: https://www.rcsb.org/

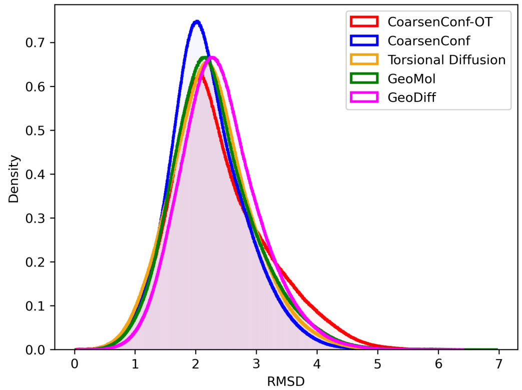


Table 2: Property prediction: Mean absolute error of generated vs. ground truth ensemble properties for E, HOMO-LUMO gap $\Delta\epsilon$, E_{min} (kcal/mol), and dipole moment μ (debye).

	E	μ	$\Delta\epsilon$	E_{min}
GeoMol	28.80	1.475	4.186	0.267
Torsional Diffusion	16.75	1.333	2.908	0.096
CoarsenConf	**12.41**	**1.250**	**2.522**	**0.049**

Figure 1: GEOM-DRUGS test set RMSD error distributions for each ML model.

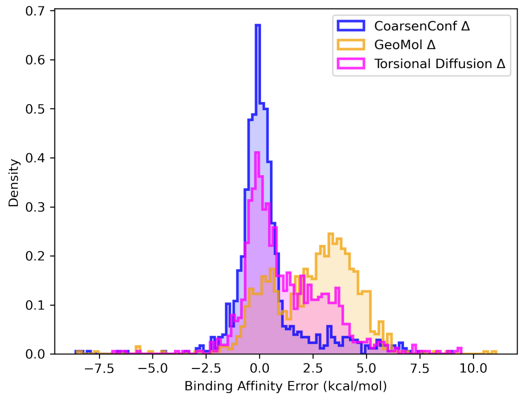


Figure 2: Binding affinity (↓ is better) error distributions for 100k conformer-protein complexes in the CrossDocked dataset. The error is the difference in binding affinity between the generated and ground truth energy minimized 3D ligands.

Table 4: Binding affinity error distribution statistics in kcal/mol (more negative is better).

Method	Mean	Min
GeoMol	2.476	-8.523
Torsional Diffusion	1.178	-6.876
CoarsenConf	**0.368**	**-8.602**

3.4 Rigid Oracle-based Protein Docking

We evaluate MCG models on rigid oracle-based protein docking. We use the 166000 protein-ligand complexes from the CrossDocked [Francoeur et al., 2020] training set (Appendix §M for details).

Problem setup. Similar to the flexible docking task (§M.1), we can generate conformers of known ligands for known protein pockets, but now have them only undergo a rigid pocket-specific energy minimization before predicting the binding affinity. See §M.2 for more details.

Results. We report the results for 100,000 unique conformer-protein interactions; note that there is a large cost to run the binding affinity prediction (see Appendix §M for more details). We also emphasize that the presented evaluation is not to be confused with actual docking solutions, as a low-energy conformer is not always guaranteed to be the best binding pose. Instead, we employ an unbiased procedure to present empirical evidence for how CoarsenConf can generate input structures to Vina that significantly outperform prior MCG models in achieving the best binding affinities.

Fig. 8 further demonstrates CoarsenConf's superior performance on orders of magnitude more protein complexes than the prior flexible oracle task. CoarsenConf decreases the average error by 56% compared to TD, and is the only method not to exhibit bimodal behavior with error greater than zero. Overall CoarsenConf best approximates the ground truth ligand conformers of CrossDocked and generates the best structures for Vina's rigid energy relaxation and binding affinity prediction.

4 Conclusion

We present CoarsenConf, a novel approach for robust molecular conformer generation that combines an SE(3)-equivariant hierarchical VAE with geometric coarse-graining techniques for accurate conformer generation. By utilizing easy-to-obtain approximate conformations, our model effectively learns the optimal distortion to generate low-energy conformers. CoarsenConf possesses unrestricted degrees of freedom, as it can adjust atomic coordinates, distances, and torsion angles freely. CoarsenConf's CG procedure can also be tailored to handle even larger systems, whereas prior methods are restricted to full FG or torsion angle space. Our experiments demonstrate the effectiveness of CoarsenConf compared to existing methods. Our study also extends recent 3D molecule-protein benchmarks to conformer generation, providing valuable insights into robust generation and downstream biologically relevant tasks.

Table 3: Quality of best generated conformer for known protein ligands for all 9 proteins from the TDC library. Quality is measured by free energy change (kcal/mol) of the binding process with AutoDock Vina's flexible docking simulation (↓ is better).

	Best Protein-Conformer Binding Affinity (↓ is better)								
Method	3PBL	2RGP	1IEP	3EML	3NY8	4RLU	4UNN	5M04	7L11
RDKit + MMFF	-8.26	-11.42	-10.75	-9.26	-9.69	-8.72	-9.73	-9.53	-9.19
GeoMol	-8.23	-11.49	-11.16	-9.39	**-11.66**	-8.85	-10.28	-9.31	-9.29
Torsional Diffusion	-8.53	-11.34	-10.76	-9.25	-10.32	-8.96	-10.65	-9.61	-9.10
CoarsenConf	**-8.81**	**-12.93**	**-16.43**	**-9.82**	-11.26	**-9.54**	**-11.62**	**-14.00**	**-9.43**

References

Keir Adams and Connor W. Coley. Equivariant shape-conditioned generation of 3d molecules for ligand-based drug design. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=4MbGnp4iPQ.

Dylan M. Anstine and Olexandr Isayev. Generative models as an emerging paradigm in the chemical sciences. *Journal of the American Chemical Society*, 145(16):8736–8750, 2023. doi: 10.1021/jacs.2c13467. URL https://doi.org/10.1021/jacs.2c13467. PMID: 37052978.

Marloes Arts, Victor Garcia Satorras, Chin-Wei Huang, Daniel Zuegner, Marco Federici, Cecilia Clementi, Frank Noé, Robert Pinsler, and Rianne van den Berg. Two for one: Diffusion models and force fields for coarse-grained molecular dynamics, 2023.

Simon Axelrod and Rafael Gómez-Bombarelli. Geom, energy-annotated molecular conformations for property prediction and molecular generation. *Scientific Data*, 9(1):185, 2022. doi: 10.1038/s41597-022-01288-4. URL https://doi.org/10.1038/s41597-022-01288-4.

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, 03 2019. doi: 10.1021/acs.jctc.8b01176. URL https://doi.org/10.1021/acs.jctc.8b01176.

Benson Chen, Xiang Fu, Regina Barzilay, and Tommi S. Jaakkola. Fragment-based sequential translation for molecular optimization. In *NeurIPS 2021 AI for Science Workshop*, 2021. URL https://openreview.net/forum?id=E_Slr0JVvuC.

Shriram Chennakesavalu, David J Toomer, and Grant M Rotskoff. Ensuring thermodynamic consistency with invertible coarse-graining. *The Journal of Chemical Physics*, 158(12), 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.

Congyue Deng, Or Litany, Yueqi Duan, Adrien Poulenard, Andrea Tagliasacchi, and Leonidas J. Guibas. Vector neurons: A general framework for so(3)-equivariant networks. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 12180–12189, 2021.

Jerome Eberhardt, Diogo Santos-Martins, Andreas F. Tillack, and Stefano Forli. Autodock vina 1.2.0: New docking methods, expanded force field, and python bindings. *Journal of Chemical Information and Modeling*, 61(8):3891–3898, 08 2021. doi: 10.1021/acs.jcim.1c00203. URL https://doi.org/10.1021/acs.jcim.1c00203.

Rémi Flamary, Nicolas Courty, Alexandre Gramfort, Mokhtar Z. Alaya, Aurélie Boisbunon, Stanislas Chambon, Laetitia Chapel, Adrien Corenflos, Kilian Fatras, Nemo Fournier, Léo Gautheron, Nathalie T.H. Gayraud, Hicham Janati, Alain Rakotomamonjy, Ievgen Redko, Antoine Rolet, Antony Schutz, Vivien Seguy, Danica J. Sutherland, Romain Tavenard, Alexander Tong, and Titouan Vayer. Pot: Python optimal transport. *Journal of Machine Learning Research*, 22(78):1–8, 2021. URL http://jmlr.org/papers/v22/20-451.html.

Paul G. Francoeur, Tomohide Masuda, Jocelyn Sunseri, Andrew Jia, Richard B. Iovanisci, Ian Snyder, and David R. Koes. Three-dimensional convolutional neural networks and a cross-docked data set for structure-based drug design. *Journal of Chemical Information and Modeling*, 60(9):4200–4215, 09 2020. doi: 10.1021/acs.jcim.0c00411. URL https://doi.org/10.1021/acs.jcim.0c00411.

Octavian Ganea, Lagnajit Pattanaik, Connor Coley, Regina Barzilay, Klavs Jensen, William Green, and Tommi Jaakkola. Geomol: Torsional geometric generation of molecular 3d conformer ensembles. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 13757–13769. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/725215ed82ab6306919b485b81ff9615-Paper.pdf.

Paraskevi Gkeka, Gabriel Stoltz, Amir Barati Farimani, Zineb Belkacemi, Michele Ceriotti, John D. Chodera, Aaron R. Dinner, Andrew L. Ferguson, Jean-Bernard Maillet, Hervé Minoux, Christine Peter, Fabio Pietrucci, Ana Silveira, Alexandre Tkatchenko, Zofia Trstanova, Rafal Wiewiora, and Tony Lelièvre. Machine learning force fields and coarse-grained variables in molecular dynamics: Application to materials and biological systems. *Journal of Chemical Theory and Computation*, 16:4757–4775, 2020. doi: 10.1021/acs.jctc.0c00355. URL `https://doi.org/10.1021/acs.jctc.0c00355`.

Jiaqi Guan, Wesley Wei Qian, Xingang Peng, Yufeng Su, Jian Peng, and Jianzhu Ma. 3d equivariant diffusion for target-aware molecule generation and affinity prediction. *arXiv preprint arXiv:2303.03543*, 2023.

Jiaqi Han, Yu Rong, Tingyang Xu, and Wen bing Huang. Geometrically equivariant graph neural networks: A survey. *ArXiv*, abs/2202.07230, 2022.

Paul C. D. 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, 04 2010. doi: 10.1021/ci100031x. URL `https://doi.org/10.1021/ci100031x`.

Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-VAE: Learning basic visual concepts with a constrained variational framework. In *International Conference on Learning Representations*, 2017. URL `https://openreview.net/forum?id=Sy2fzU9gl`.

Kexin Huang, Tianfan Fu, Wenhao Gao, Yue Zhao, Yusuf Roohani, Jure Leskovec, Connor W Coley, Cao Xiao, Jimeng Sun, and Marinka Zitnik. Therapeutics data commons: Machine learning datasets and tasks for drug discovery and development. *Proceedings of Neural Information Processing Systems, NeurIPS Datasets and Benchmarks*, 2021.

Yinan Huang, Xing Peng, Jianzhu Ma, and Muhan Zhang. 3dlinker: An e(3) equivariant variational autoencoder for molecular linker design. In *International Conference on Machine Learning*, 2022.

Brooke E. Husic, Nicholas E. Charron, Dominik Lemm, Jiang Wang, Adrià Pérez, Maciej Majewski, Andreas Krämer, Yaoyi Chen, Simon Olsson, Gianni de Fabritiis, Frank Noé, and Cecilia Clementi. Coarse graining molecular dynamics with graph neural networks. *The Journal of Chemical Physics*, 153(19), 11 2020. ISSN 0021-9606. doi: 10.1063/5.0026133. URL `https://doi.org/10.1063/5.0026133`. 194101.

Jaehyeok Jin, Alexander J. Pak, Aleksander E. P. Durumeric, Timothy D. Loose, and Gregory A. Voth. Bottom-up coarse-graining: Principles and perspectives. *Journal of Chemical Theory and Computation*, 18(10):5759–5791, 10 2022a. doi: 10.1021/acs.jctc.2c00643. URL `https://doi.org/10.1021/acs.jctc.2c00643`.

Wengong Jin, Regina Barzilay, and T. Jaakkola. Hierarchical generation of molecular graphs using structural motifs. In *International Conference on Machine Learning*, 2020.

Wengong Jin, Regina Barzilay, and Tommi Jaakkola. Antibody-antigen docking and design via hierarchical structure refinement. In *International Conference on Machine Learning*, pages 10217–10227. PMLR, 2022b.

Bowen Jing, Gabriele Corso, Regina Barzilay, and Tommi S. Jaakkola. Torsional diffusion for molecular conformer generation. In *ICLR2022 Machine Learning for Drug Discovery*, 2022. URL `https://openreview.net/forum?id=D9IxPlXPJJS`.

John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, Alex Bridgland, Clemens Meyer, Simon A. A. Kohl, Andrew J. Ballard, Andrew Cowie, Bernardino Romera-Paredes, Stanislav Nikolov, Rishub Jain, Jonas Adler, Trevor Back, Stig Petersen, David Reiman, Ellen Clancy, Michal Zielinski, Martin Steinegger, Michalina Pacholska, Tamas Berghammer, Sebastian Bodenstein, David Silver, Oriol Vinyals, Andrew W. Senior, Koray Kavukcuoglu, Pushmeet Kohli, and Demis Hassabis. Highly accurate protein structure prediction with alphafold. *Nature*, 596(7873):583–589, 2021. doi: 10.1038/s41586-021-03819-2. URL `https://doi.org/10.1038/s41586-021-03819-2`.

W. Kabsch. Automatic processing of rotation diffraction data from crystals of initially unknown symmetry and cell constants. *Journal of Applied Crystallography*, 26(6):795–800, Dec 1993. doi: 10.1107/S0021889893005588. URL `https://doi.org/10.1107/S0021889893005588`.

Sebastian Kmiecik, Dominik Gront, Michal Kolinski, Lukasz Wieteska, Aleksandra Elzbieta Dawid, and Andrzej Kolinski. Coarse-grained protein models and their applications. *Chemical Reviews*, 116(14):7898–7936, 07 2016. doi: 10.1021/acs.chemrev.6b00163. URL `https://doi.org/10.1021/acs.chemrev.6b00163`.

David Ryan Koes, Matthew P. Baumgartner, and Carlos J. Camacho. Lessons learned in empirical scoring with smina from the csar 2011 benchmarking exercise. *Journal of Chemical Information and Modeling*, 53(8):1893–1904, 08 2013. doi: 10.1021/ci300604z. URL `https://doi.org/10.1021/ci300604z`.

Shitong Luo, Chence Shi, Minkai Xu, and Jian Tang. Predicting molecular conformation via dynamic graph score matching. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 19784–19795. Curran Associates, Inc., 2021. URL `https://proceedings.neurips.cc/paper_files/paper/2021/file/a45a1d12ee0fb7f1f872ab91da18f899-Paper.pdf`.

Elman Mansimov, Omar Mahmood, Seokho Kang, and Kyunghyun Cho. Molecular geometry prediction using a deep generative graph neural network. *Scientific reports*, 9(1):20381, 2019.

Xingang Peng, Shitong Luo, Jiaqi Guan, Qi Xie, Jian Peng, and Jianzhu Ma. Pocket2mol: Efficient molecular sampling based on 3d protein pockets. In *International Conference on Machine Learning*, pages 17644–17655. PMLR, 2022.

Philipp Pracht, Fabian Bohle, and Stefan Grimme. Automated exploration of the low-energy chemical space with fast quantum chemical methods. *Phys. Chem. Chem. Phys.*, 22:7169–7192, 2020. doi: 10.1039/C9CP06869D. URL `http://dx.doi.org/10.1039/C9CP06869D`.

Danny Reidenbach, Micha Livne, Rajesh K. Ilango, Michelle Lynn Gill, and Johnny Israeli. Improving small molecule generation using mutual information machine. In *ICLR 2023 - Machine Learning for Drug Discovery workshop*, 2023. URL `https://openreview.net/forum?id=iOJlwUTUyrN`.

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, 12 2015. doi: 10.1021/acs.jcim.5b00654. URL `https://doi.org/10.1021/acs.jcim.5b00654`.

Victor Garcia Satorras, Emiel Hoogeboom, and Max Welling. E(n) equivariant graph neural networks. In *International Conference on Machine Learning*, 2021.

Kristof T. Schütt, Oliver T. Unke, and Michael Gastegger. Equivariant message passing for the prediction of tensorial properties and molecular spectra. In *International Conference on Machine Learning*, 2021.

Chence Shi, Minkai Xu, Zhaocheng Zhu, Weinan Zhang, Ming Zhang, and Jian Tang. Graphaf: a flow-based autoregressive model for molecular graph generation. In *International Conference on Learning Representations*, 2020. URL `https://openreview.net/forum?id=S1esMkHYPr`.

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

Gregor Simm and Jose Miguel Hernandez-Lobato. A generative model for molecular distance geometry. In *Proceedings of the 37th International Conference on Machine Learning*, pages 8949–8958. PMLR, 2020.

Evan Walter Clark Spotte-Smith, Samuel M. Blau, Xiaowei Xie, Hetal D. Patel, Mingjian Wen, Brandon Wood, Shyam Dwaraknath, and Kristin Aslaug Persson. Quantum chemical calculations of lithium-ion battery electrolyte and interphase species. *Scientific Data*, 8(1):203, 2021. doi: 10.1038/s41597-021-00986-9. URL https://doi.org/10.1038/s41597-021-00986-9.

Hannes Stärk, Octavian Ganea, Lagnajit Pattanaik, Regina Barzilay, and Tommi Jaakkola. Equibind: Geometric deep learning for drug binding structure prediction. In *International Conference on Machine Learning*, pages 20503–20521. PMLR, 2022.

Michael G. Taylor, Daniel J. Burrill, Jan Janssen, Enrique R. Batista, Danny Perez, and Ping Yang. Architector for high-throughput cross-periodic table 3d complex building. *Nature Communications*, 14(1):2786, 2023. doi: 10.1038/s41467-023-38169-2. URL https://doi.org/10.1038/s41467-023-38169-2.

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.

Wujie Wang, Minkai Xu, Chen Cai, Benjamin Kurt Miller, Tess E. Smidt, Yusu Wang, Jian Tang, and Rafael Gomez-Bombarelli. Generative coarse-graining of molecular conformations. In *International Conference on Machine Learning*, 2022.

Ronald J. Williams and David Zipser. A Learning Algorithm for Continually Running Fully Recurrent Neural Networks. *Neural Computation*, 1(2):270–280, 06 1989. ISSN 0899-7667. doi: 10.1162/neco.1989.1.2.270. URL https://doi.org/10.1162/neco.1989.1.2.270.

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. URL https://openreview.net/forum?id=pAbm1qfheGk.

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*, pages 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. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=PzcvxEMzvQC.

Soojung Yang and Rafael Gomez-Bombarelli. Chemically transferable generative backmapping of coarse-grained proteins. *ArXiv*, abs/2303.01569, 2023.

Gengmo Zhou, Zhifeng Gao, Qiankun Ding, Hang Zheng, Hongteng Xu, Zhewei Wei, Linfeng Zhang, and Guolin Ke. Uni-mol: A universal 3d molecular representation learning framework. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=6K2RM6wVqKu.

Jinhua Zhu, Yingce Xia, Chang Liu, Lijun Wu, Shufang Xie, Yusong Wang, Tong Wang, Tao Qin, Wengang Zhou, Houqiang Li, Haiguang Liu, and Tie-Yan Liu. Direct molecular conformation generation. *Transactions on Machine Learning Research*, 2022. URL https://openreview.net/forum?id=lCPOHiztuw.

A Figures

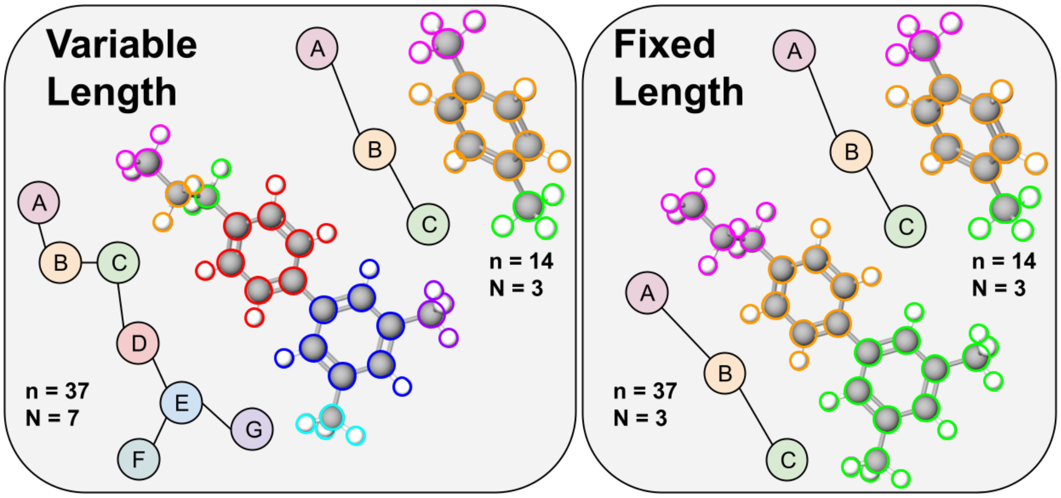


Figure 3: **Learning flexible coarse-grained representations.** CoarsenConf is the first model to employ variable-length (on left) coarse-graining. Each input molecule (n fine-grained (FG) atoms) can be represented by a different number of coarse-grained (CG) nodes N, thus accommodating diverse molecular sizes. In contrast, prior approaches rely on fixed-length (on right) coarse-graining, thereby forcing all molecules to possess the same number of CG nodes. Variable-length coarse-graining enhances the model's ability to create better learned representations across molecules of different sizes and geometries. The molecules on the left are coarsened along torsional angles.

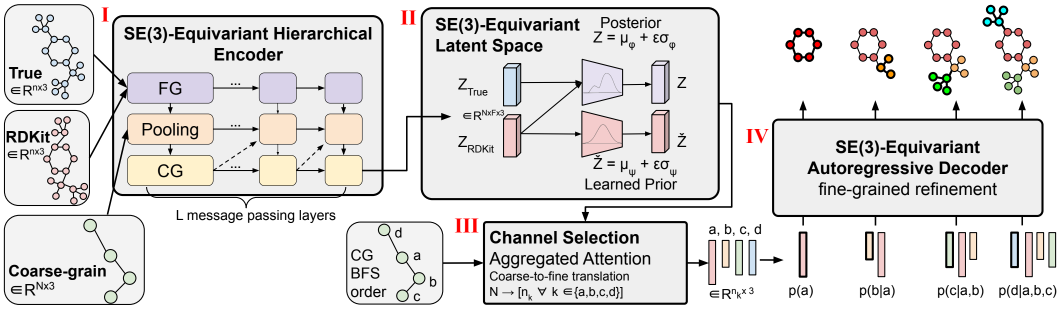


Figure 4: **CoarsenConf architecture. (I)** The encoder $q_\phi(z|X, \mathcal{R})$ takes fine-grained (FG) ground truth conformer X, RDKit approximate conformer $\mathcal{R}$, and coarse-grained (CG) conformer $\mathcal{C}$ as inputs (derived from X and the predefined CG strategy), and outputs a variable-length equivariant CG representation via equivariant message passing and point convolutions. **(II)** Equivariant MLPs are applied to learn the mean and log variance of both the posterior and prior distributions. **(III)** The posterior (training) or prior (inference) is sampled and fed into the Channel Selection module, where an attention layer is used to learn the optimal mapping from CG to FG structure. **(IV)** Given the FG latent vector and the RDKit approximation, the decoder $p_\theta(X|\mathcal{R}, z)$ learns to recover the low-energy FG structure through autoregressive equivariant message passing. The entire model can be trained end-to-end by optimizing the KL divergence of latent distributions and reconstruction error.

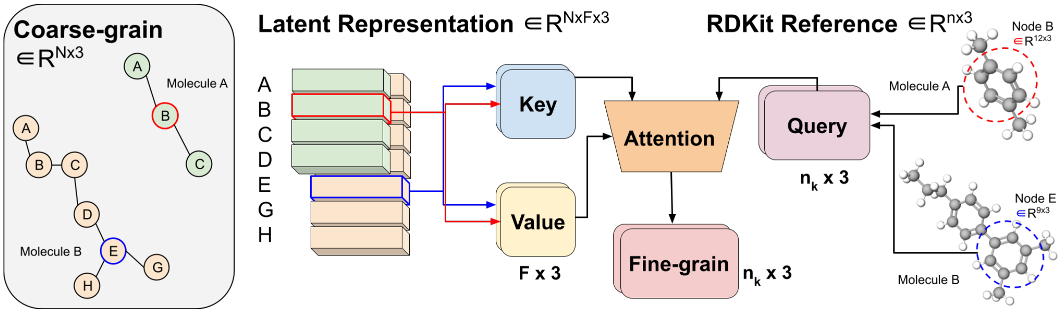


Figure 5: **Variable-length coarse-to-fine backmapping via Aggregated Attention.** The highlighted latent beads of two independent molecules are attended to by the respective fine-grained queries in a batched manner (see red and blue), to generate FG coordinates in the desired shape (matching input queries on right). The single-head attention operation uses the latent vectors of each CG bead $Z_k \in R^{F \times 3}$ for each molecule as the keys and values, with an embedding dimension of 3 to match the x, y, z coordinates. The query vectors are the FG subset of the respective RDKit conformers, corresponding to each CG bead $\in R^{n_k \times 3}$. We know a priori how many FG atoms correspond to a certain CG bead (n_k). Aggregated Attention learns the optimal blending of CG features for FG reconstruction by aggregating 3D segments of FG information to form our latent query.

B Related Work

Existing Challenges. Existing generative MCG ML models can be broadly classified based on two primary criteria: (1) the choice of the architecture to model the distribution of low-energy conformers and (2) the manner in which they incorporate geometric information to model 3D conformers. A majority of these methods utilize a single geometric representation, such as distance matrices, atom coordinates, or torsional angles, and pair them with various probabilistic modeling techniques, including variational autoencoders (VAEs) and diffusion models. A recurring limitation of these prior approaches is that they are restricted to modeling only one specific aspect of geometric information *i.e.*, coordinates or angles. By restricting themselves to either a pure coordinate or angle space, they often fail to fully leverage the full geometric information inherent to the problem. Consequently, in this work, we introduce a more flexible molecular latent representation that seamlessly integrates multiple desired geometric modalities, thereby enhancing the overall accuracy and versatility of conformer generation.

To create this flexible representation, we use a process known as coarse-graining to distill molecular information into a simplified, coarse-grained (CG) representation based on specific coarsening criteria. This is analogous to previous 2D fragment-based generative techniques [Chen et al., 2021]. In these scenarios, the fragmentation of molecules into prominent substructures led to significant enhancements in representation learning and generative tasks. Coarse-graining has seen success in related ML applications like molecular dynamics and protein modeling [Husic et al., 2020, Kmiecik et al., 2016]. However, these ML approaches have primarily explored rigid coarsening criteria, such as distance-based nearest neighbor methods, which represent all inputs with a fixed granularity or number of beads (*i.e.* CG nodes, N). ML models that use fixed-length coarse-graining often necessitate N be fixed for all input molecules [Yang and Gomez-Bombarelli, 2023]. This approach may not be suitable for all scenarios, especially when navigating multi-modal datasets of drug-like molecules of varying sizes (multiple N), which better reflects real-world conditions.

To address the limitations of fixed-length CG representations, we introduce *Aggregated Attention* for variable-length coarse-graining (see Fig. 3). This methodology allows a single latent representation to accommodate molecules with different numbers of fine-grained (FG) atoms and CG beads. The inherent flexibility of the attention mechanism allows input molecules to be fragmented along torsion angles, enabling the modeling of interatomic distances, 3D atom coordinates, and torsion angles in an equivariant manner, regardless of the molecule's shape or size. Through Aggregated Attention, we also harness information from the entire learned representation, unlike preceding approaches that restrict the backmapping (from CG back to FG) to a subset [Wang et al., 2022]. The adaptability of our learned variable-length representations enables more accurate generation.

Classical Methods for Conformer Generation. A molecular conformer refers to the collection of 3D structures that are energetically favorable and correspond to local minima of the potential energy surface. CREST [Pracht et al., 2020] uses semi-empirical tight-binding density functional theory (DFT) for energy calculations, which, while computationally less expensive than ab-initio quantum mechanical (QM) methods, still requires approximately 90 core hours per drug-like molecule [Axelrod and Gómez-Bombarelli, 2022]. Though CREST was used to generate the “ground truth” GEOM dataset, it is too slow for downstream applications such as high-throughput virtual screening.

Cheminformatics methods, such as RDKit ETKDG, are commonly used to quickly generate approximate low-energy conformations of molecules. These methods are less accurate than QM methods due to the sparse coverage of the conformational space resulting from stochastic sampling. Additionally, force field optimizations are inherently less accurate than the above QM methods. RDKit ETKDG employs a genetic algorithm for Distance Geometry optimization that can be enhanced with a molecular mechanics force field optimization (MMFF).

Deep Learning Methods for Conformer Generation. Several probabilistic deep learning methods for MCG have been developed [Anstine and Isayev, 2023], such as variational autoencoders in CVGAE [Mansimov et al., 2019] and ConfVAE [Xu et al., 2021b], normalizing flows in CGCF [Xu et al., 2021a], score-based generative models in ConfGF [Shi et al., 2021] and DGSM [Luo et al., 2021], and diffusion models in GeoDiff [Xu et al., 2022] and Torsional Diffusion [Jing et al., 2022]. GraphDG [Simm and Hernandez-Lobato, 2020] forgoes modeling coordinates and angles, relying solely on distance geometry. DMCG [Zhu et al., 2022] and Uni-Mol [Zhou et al., 2023] present examples of effective large models, the first mimicking the architecture of AlphaFold [Jumper et al., 2021] and the second using large-scale SE(3)-equivariant transformer pre-training.

Molecular Coarse-graining. Molecular coarse-graining refers to the simplification of a molecule representation by grouping the fine-grained (FG) atoms in the original structure into individual coarse-grained (CG) beads with a rule-based mapping.[2] Coarse-graining has been widely utilized in protein design [Kmiecik et al., 2016] and molecular dynamics [Gkeka et al., 2020], and analogously fragment-level or subgraph-level generation has proven to be highly valuable in diverse 2D molecule design tasks [Chen et al., 2021]. Breaking down generative problems into smaller pieces can be applied to several 3D molecule tasks. For instance, CGVAE [Wang et al., 2022] learns a latent distribution to back map or restore FG coordinates from a fixed number of CG beads effectively. We note that various coarse-graining strategies exist [Jin et al., 2022a, Arts et al., 2023, Husic et al., 2020, Chennakesavalu et al., 2023], and many require the ability to represent inputs with a non-fixed granularity. To handle this, CoarsenConf uses a flexible variable-length CG representation that is compatible with all coarse-graining techniques.

Autoregressive Molecule Generation. Autoregressive models provide control over the generative process by enabling direct conditioning on prior information, allowing for a more precise and targeted generation of output. Autoregressive generation has shown success in 2D molecule tasks using SMILE-based methods, as seen in MolMIM [Reidenbach et al., 2023], as well as graph-based atom-wise and subgraph-level techniques, as shown in GraphAF [Shi et al., 2020] and HierVAE [Jin et al., 2020]. Similarly, 3DLinker [Huang et al., 2022] and SQUID [Adams and Coley, 2023] showcase the usefulness of 3D autoregressive molecule generation and their ability to leverage conditional information in both atom-wise and subgraph-level settings for 3D linkage and shape-conditioned generative tasks respectively. We note that, unlike prior methods [Adams and Coley, 2023], CoarsenConf does not require a predefined fragment vocabulary. HERN [Jin et al., 2022b] further demonstrates the power of hierarchical equivariant autoregressive methods in the task of computational 3D antibody design. Similarly, Pocket2Mol [Peng et al., 2022] uses autoregressive sampling for structure-based drug design.

Protein Docking and Structure-based Drug Design. Protein docking is a key downstream use case for generating optimal 3D molecule structures. Recent research has prominently explored two distinct directions within this field. The first is blind docking, where the goal is to locate the pocket and generate the optimal ligand to bind [Corso et al., 2022]. The second is structure-based drug design (SBDD), where optimal 3D ligands are generated by conditioning on a specific protein pocket. Specifically, the SBDD task focuses on the ability to generate ligands that achieve a low AutoDock Vina score for the CrossDocked2020 [Francoeur et al., 2020] dataset. AutoDock Vina [Eberhardt

[2] We use the terms CG graph nodes and beads interchangeably.

et al., 2021] is a widely used molecular docking software that predicts the binding affinity of ligands (drug-like molecules) to target proteins. Autodock Vina takes in the 3D structures of the ligand, target protein, and binding pocket and considers various factors such as van der Waals interactions, electrostatic interactions, and hydrogen bonding between the ligand and target protein to predict the binding affinity. We demonstrate how SBDD can be adapted to construct comprehensive MCG benchmarks. In this framework, we evaluate the generative abilities of MCG models by measuring the binding affinities of generated comforters and comparing them to the provided ground truth ligand conformers for a wide array of protein-ligand complexes.

Equivariance. Let $\mathcal{X}$ and $\mathcal{Y}$ be the input and output vector spaces, respectively, which possess a set of transformations G: $G \times \mathcal{X} \to \mathcal{X}$ and $G \times \mathcal{Y} \to \mathcal{Y}$. The function $\phi : \mathcal{X} \to \mathcal{Y}$ is called equivariant with respect to G if, when we apply any transformation to the input, the output also changes via the same transformation or under a certain predictable behavior, *i.e.*,

Definition 1 *The function $\phi : \mathcal{X} \mapsto \mathcal{Y}$ is G-equivariant if it commutes with any transformation in G,*

$$\phi(\rho_{\mathcal{X}}(g)x) = \rho_{\mathcal{Y}}(g)\phi(x), \forall g \in G, \tag{3}$$

where $\rho_{\mathcal{X}}$ and $\rho_{\mathcal{Y}}$ are the group representations in the input and output space, respectively. Specifically, ϕ is called invariant if $\rho_{\mathcal{Y}}$ is the identity.

By enforcing SE(3)-equivariance in our probabilistic model, $p(\boldsymbol{X}|\mathcal{R})$ remains unchanged for any rototranslation of the approximate conformer $\mathcal{R}$. CoarsenConf's architecture is inspired by recent equivariant graph neural network architectures, such as EGNN [Satorras et al., 2021] and PaiNN [Schütt et al., 2021], as well as Vector Neuron multi-layer perceptron (VN-MLP) [Deng et al., 2021].

C Loss function

As described in §2, CoarsenConf optimizes the following loss function:

$$\text{MSE}(\mathcal{A}(X, X_{true})) + \beta_1 D_{KL}(q_\phi(z|X,\mathcal{R}) \parallel p_\psi(z|\mathcal{R})) + \beta_2 \frac{1}{|\mathcal{E}^*|} \sum_{(i,j)\in\mathcal{E}^*} ||r_{ij} - r_{ij}^{true}||^2, \tag{4}$$

where $\mathcal{A}$ is the Kabsch alignment function [Kabsch, 1993], $\mathcal{E}^*$ are all the 1 and 2-hop edges in the molecular graph, with r_{ij} corresponding to the distance between atoms i and j. We note that both β_1 and β_2 play a crucial role in the optimization. β_1 has to be set low enough ($1e{-}3$) to allow the optimization to focus on the MSE when the differences between the model-based X and the ground truth are very close, due to the RDKit distortion parameterization.

For the QM9 experiments, β_1 is annealed starting from $1e{-}6$ to $1e{-}1$, increasing by a factor of 10 each epoch. β_2 controls the distance auxiliary loss and also had to be similarly annealed. We found that when $\beta_2 = 0$, CoarsenConf still learned to improve upon the aligned MSE loss by 50%, as compared to RDKit. Our error analysis showed that the resulting molecules either had extremely low distance error with high MSE, or vice-versa. Therefore, when the learning objective is unconstrained, our model learns to violate distance constraints by placing atoms in low-error but unphysical positions.

For QM9, by slowly annealing the distance loss, we allow our model to reach a metaphysical unstable transition state where distances are violated, but the aligned coordinate error is better. We then force the model to respect distance constraints. In the case of DRUGS, we found that this transition state was too difficult for the model to escape from, and we report the results using $\beta_2 = 0.5$ in Tab. 1. In Appendix §L, we further explore this idea and experiment with different annealing schedules for DRUGS. We note that as CoarsenConf learns the torsion angles in an unsupervised manner because of the chosen CG strategy, we leave explicit angle optimization to future work.

D Coarse-graining

Coarse-graining Procedure. We first define a rotatable bond as any single bond between two non-terminal atoms, excluding amides and conjugated double bonds, where the torsion angle is the angle of rotation around the central bond. Formally, the torsion angle τ_{abcd} is defined about bond $(b, c) \in \mathcal{E}$ where (a, b) are a choice of reference neighbors s.t $a \in \mathcal{N}(b) \setminus c$ and $d \in \mathcal{N}(c) \setminus b$.

We coarsen molecules into a single fragment or bead for each connected component, resulting from severing all rotatable bonds. This choice in CG procedure implicitly forces the model to learn over torsion angles, as well as atomic coordinates and inter-atomic distances. We found that using a more physically constrained definition of torsional angles, as defined by Ganea et al. [2021], in the CG procedure led to a significant increase in performance compared to that used in Jing et al. [2022]. This is because the latter allows rotations around double and triple bonds, while the former does not. An example of the coarse-graining procedure is in Fig. 3. For formal definitions, see Appendix §D.

Following Wang et al. [2022], we represent fine-grained (FG) molecular conformers as $x = \{x_i\}_{i=1}^{n} \in \mathbb{R}^{n\times 3}$. Similarly, the coarse-grained (CG) conformers are represented by $X = \{X_I\}_{I=1}^{N} \in \mathbb{R}^{N\times 3}$ where $N < n$. Let $[n]$ and $[N]$ denote the set $\{1, 2, ..., n\}$ and $\{1, 2, ..., N\}$ respectively. The CG operation can be defined as an assignment $m : [n] \rightarrow [N]$, which maps each FG atom i in $[n]$ to CG bead $I \in [N]$, i.e., bead I is composed of the set of atoms $C_I = (k \in n \mid m(k) = I)$. X_I is initialized at the center of mass = $\frac{1}{|C_I|} \sum_{j \in C_I} x_j$.

We note that CoarsenConf coarsens input molecules by first severing all torsion angles τ_{abcd}, with k torsion angles resulting in $k + 1$ connected components or CG beads. This allows us, on average, to represent QM9 molecules with three beads and large drug molecules $(n > 100)$ with 29 beads. We opted for a torsion angle-based strategy as it allows for unsupervised control over torsion angles, as well as the ability to rotate each subgraph independently. The CG strategy can be altered for various applications going forward.

E Encoder Equations

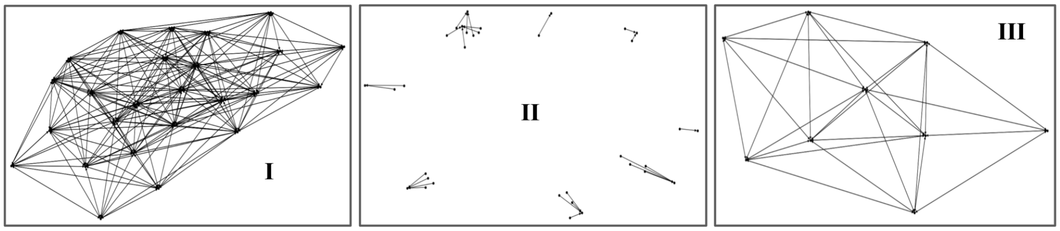


Figure 6: **Encoder module message passing structure. (I)** Fine-grained graph with auxiliary 4Å distance cut off. **(II)** Pooling graph with nodes for each atom and coarse-grained bead. Each group of nodes represents the formation of a CG bead. There is a single directional edge from each atom to its corresponding bead. **(III)** Coarse-grained graph with auxiliary 4Å distance cut off, using the learned representation from the pooling graph. CoarsenConf reduces the input from (I) to (III), drastically reducing the complexity of the problem.

E.1 Fine-grain Module

We describe the encoder, shown in Fig. 4(I). The model operates over SE(3)-invariant atom features $h \in R^{n\times D}$, and SE(3)-equivariant atomistic coordinates $x \in R^{n\times 3}$. A single encoder layer is composed of three modules: fine-grained, pooling, and coarse-grained. Full equations for each module can be found in Appendix §E.1, §E.2, §E.3, respectively.

The fine-grained module is a graph-matching message-passing architecture. It differs from Stärk et al. [2022] by not having internal closed-form distance regularization and exclusively using unidirectional attention. It aims to effectively match the approximate conformer and ground truth by updating attention from the former to the latter.

The FG module is responsible for processing the FG atom coordinates and invariant features. More formally, the FG is defined as follows:

$$\begin{aligned}
\boldsymbol{m}_{j\to i} &= \phi^e(\boldsymbol{h}_i^{(t)}, \boldsymbol{h}_j^{(t)}, \|\boldsymbol{x}_i^{(t)} - \boldsymbol{x}_j^{(t)}\|^2, \boldsymbol{f}_{j\to i}), \forall (I, J) \in \mathcal{E} \cup \mathcal{E}', \\
\boldsymbol{u}_{j'\to i} &= a_{j'\to i} \boldsymbol{W} \boldsymbol{h}_{j'}^{(t)}, \forall i \in \mathcal{V}, j' \in \mathcal{V}', \\
\boldsymbol{m}_i &= \frac{1}{|\mathcal{N}(i)|} \sum_{j \in \mathcal{N}(i)} \boldsymbol{m}_{j\to i}, \forall i \in \mathcal{V} \cup \mathcal{V}', \\
\boldsymbol{u}_i &= \sum_{j' \in \mathcal{V}'} \boldsymbol{u}_{j'\to i}, \forall i \in \mathcal{V}, \quad \text{and} \quad \boldsymbol{u}'_i = 0, \\
\boldsymbol{x}_i^{(t+1)} &= \eta_x \cdot \boldsymbol{x}_i^{(0)} + (1 - \eta_x) \cdot \boldsymbol{x}_i^{(t)} + \sum_{j \in \mathcal{N}(i)} (\boldsymbol{x}_i^{(t)} - \boldsymbol{x}_j^{(t)}) \phi^x(\boldsymbol{m}_{j\to i}), \\
\boldsymbol{h}_i^{(t+1)} &= (1 - \eta_h) \cdot \boldsymbol{h}_i^{(t)} + \eta_h \cdot \phi^h(\boldsymbol{h}_i^{(t)}, \boldsymbol{m}_i, \boldsymbol{u}_i, \boldsymbol{f}_i), \forall i \in \mathcal{V} \cup \mathcal{V}',
\end{aligned} \tag{5}$$

where f represents the original invariant node features $h^{t=0}$, $a_{j\to i}$ are SE(3)-invariant attention coefficients derived from h embeddings, $\mathcal{N}(i)$ are the graph neighbors of node i, and W is a parameter matrix. $(\mathcal{V}, \mathcal{E})$ and $(\mathcal{V}', \mathcal{E}')$ refer to the low-energy and RDKit approximation molecular graphs, respectively. The various ϕ functions are modeled using shallow MLPs, with ϕ^x outputting a scalar and ϕ^e and ϕ^h returning a D-dimensional vector. η_x and η_h are weighted update parameters for the FG coordinates x and invariant features h respectively. We note that attention flows in a single direction from the RDKit approximation to the ground truth to prevent leakage in the parameterization of the learned prior distribution.

E.2 Pooling Module

The pooling module takes in the updated representations (h and x) of both the ground truth molecule and the RDKit reference from the FG module. The pooling module is similar to the FG module, except it no longer uses attention and operates over a pooling graph. Given a molecule with n atoms and N CG beads, the pooling graph consists of $n + N$ nodes. There is a single directional edge from all atoms to their respective beads. This allows message passing to propagate information through the predefined coarsening strategy.

The pooling module is responsible for learning the coordinates and invariant features of each coarse-grained bead by pooling FG information in a graph-matching framework. More formally, the pooling module is defined as follows:

$$
\begin{aligned}
\boldsymbol{m}_{j\to I} &= \phi^e(\boldsymbol{H}_I^{(t)}, \boldsymbol{h}_j^{(t)}, \|\boldsymbol{X}_I^{(t)} - \boldsymbol{x}_j^{(t)}\|^2, \boldsymbol{f}_{j\to I}), \forall (I, J) \in \mathcal{E} \cup \mathcal{E}', \\
\boldsymbol{m}_I &= \frac{1}{|\mathcal{N}(I)|} \sum_{j \in \mathcal{N}(I)} \boldsymbol{m}_{j\to I}, \forall I \in \mathcal{V} \cup \mathcal{V}', \\
\boldsymbol{X}_I^{(t+1)} &= \eta_X \cdot \boldsymbol{X}_I^{(0)} + (1 - \eta_X) \cdot \boldsymbol{X}_I^{(t)} + \sum_{j \in \mathcal{N}(I)} (\boldsymbol{X}_I^{(t)} - \boldsymbol{x}_j^{(t)}) \phi^x(\boldsymbol{m}_{j\to I}), \\
\boldsymbol{H}_I^{(t+1)} &= (1 - \eta_H) \cdot \boldsymbol{H}_I^{(t)} + \eta_H \cdot \phi^h(\boldsymbol{H}_I^{(t)}, \boldsymbol{m}_I, \boldsymbol{f}_I), \forall I \in \mathcal{V} \cup \mathcal{V}',
\end{aligned}
\tag{6}
$$

where capital letters refer to the CG representation of the pooling graph. The pooling module mimics the FG module without attention on a pooling graph, as seen in Fig. 6(II). The pooling graph contains a single node for each atom and CG bead, with a single edge from each FG atom to its corresponding bead. It is used to learn the appropriate representations of the CG information. As the pooling graph only contains edges from fine-to-coarse nodes, the fine-grain coordinates and features remain unchanged. The pooling graph at layer t uses the invariant feature H from the CG module of layer $t - 1$ to propagate information forward through the neural network. The main function of the pooling module is to act as a buffer between the FG and CG spaces. As a result, we found integrating the updated CG representation useful for building a better transition from FG to CG space.

E.3 Coarse-grain Module

The coarse-grained module uses the updated CG representations ($H \in R^{N\times D}$ and $X \in R^{N\times 3}$) from the pooling module to learn equivariant CG features (Z and $\tilde{Z} \in R^{N\times F\times 3}$) for the ground truth molecule and the RDKit reference. F is fixed as a hyperparameter for latent space size. N is allowed to be variable-length to handle molecules resulting from any coarsening procedure. The CG features are learned using a graph-matching point convolution [Thomas et al., 2018] with similar unidirectional attention as the FG module. Prior to the main message-passing operations, the input features undergo equivariant mixing [Huang et al., 2022] to further distill geometric information into the learned CG representation.

The CG module is responsible for taking the pooled CG representation from the pooling module and learning a node-level equivariant latent representation. We note that we use simple scalar and vector operations to mix equivariant and invariant features without relying on computationally expensive higher-order tensor products. In the first step, invariant CG features H and equivariant features $\boldsymbol{v} \in \mathbb{R}^{F\times 3}$ are transformed and mixed to construct new expressive intermediate features $H', H'', \boldsymbol{v}'$ by,

$$H'_I = \phi_1(h_I^{(t)}, \|\text{VN-MLP}_1(\boldsymbol{v}_{\boldsymbol{I}}^{(\boldsymbol{t})})\|) \in \mathbb{R}^D, \tag{7a}$$

$$H''_I = \phi_2(h_I^{(t)}, \|\text{VN-MLP}_2(\boldsymbol{v}_{\boldsymbol{I}}^{(\boldsymbol{t})})\|) \in \mathbb{R}^F, \tag{7b}$$

$$\boldsymbol{v}'_I = \text{diag}\{\phi_3(H_I^{(t)})\} \cdot \text{VN-MLP}_3(\boldsymbol{v}_{\boldsymbol{I}}^{(\boldsymbol{t})}) \in \mathbb{R}^{F\times 3}. \tag{7c}$$

Next, a point convolution [Thomas et al., 2018, Schütt et al., 2021, Huang et al., 2022] is applied to linearly transform the mixed features $H', H'', \boldsymbol{v}'$ into messages:

$$\begin{aligned}
\boldsymbol{m}^{\boldsymbol{H}}_{I \leftarrow J} &= \mathrm{Ker}_1(\|\boldsymbol{r}_{I,J}\|) \odot H'_J, && \text{(8a)}\\
\boldsymbol{m}^{\boldsymbol{v}}_{I \leftarrow J} &= \mathrm{diag}\left\{\mathrm{Ker}_2(\|\boldsymbol{r}_{I,J}\|)\right\} \cdot \boldsymbol{v}'_J + \left(\mathrm{Ker}_3(\|\boldsymbol{r}_{I,J}\|) \odot H''_J\right) \cdot \boldsymbol{r}^{\top}_{I,J}, && \text{(8b)}\\
\boldsymbol{u}_{J' \to I} &= a_{J' \to I} \boldsymbol{W} \boldsymbol{H}^{(t)}_{J'}, \forall I \in \mathcal{V}, J' \in \mathcal{V}', && \text{(8c)}\\
\boldsymbol{u}_I &= \sum_{J' \in \mathcal{V}'} \boldsymbol{u}_{J' \to I}, \forall I \in \mathcal{V}, \quad \text{and} \quad \boldsymbol{u}'_I = 0, && \text{(8d)}\\
\boldsymbol{H}^{t+1}_I &= (1 - \eta_H) \cdot H^{\ell}_I + \eta_H \cdot \mathrm{MLP}(H^{\ell}_I, \sum_{J \in N(I)} m^H_{I \leftarrow J}, u_I), \forall I \in \mathcal{V} \cup \mathcal{V}', && \text{(8e)}\\
\boldsymbol{v}^{t+1}_I &= (1 - \eta_v) \cdot v^{\ell}_I + \eta_v \cdot \text{VN-MLP}_4(\boldsymbol{v}^{\ell}_I, \sum_{J \in N(I)} \boldsymbol{m}^{\boldsymbol{v}}_{I \leftarrow J}), \forall I \in \mathcal{V} \cup \mathcal{V}', && \text{(8f)}
\end{aligned}$$

where each Ker refers to a learned RBF kernel, r_{IJ} is the difference between X_I and X_J, and $a_{J \to I}$ are SE(3)-invariant attention coefficients derived from the learned invariant features H. η_H and η_v control the mixing of the learned invariant and equivariant representations.

We note that for $t > 0$, the H_I from the CG module are used in the next layer's pooling module, creating a cyclic dependency to learn an information-rich CG representation. This is shown by the dashed lines in Fig. 4(I). The cyclic flow of information grounds the learned CG representation to the innate FG structure. All equivariant CG features v are initialized as zero and are slowly built up through each message passing layer. As point convolutions and VN operations are strictly SO(3)-equivariant, we subtract the molecule's centroid from the atomic coordinates prior to encoding, making it effectively SE(3)-equivariant.

The modules in each encoder layer communicate with the respective module of the previous layer. This hierarchical message-passing scheme results in an informative and geometrically grounded final CG latent representation. We note that the pooling module of layer ℓ uses the updated invariant features H from the CG module of layer $\ell - 1$, as shown by the dashed lines in Fig. 4(I).

F Equivariant Latent Space.

As Z holds a mixture of equivariant spatial information, we maintain equivariance through the reparametrization trick of the VAE (Fig. 4(II)). Specifically, we define the posterior and prior means ($\boldsymbol{\mu}_\phi$, $\boldsymbol{\mu}_\psi$) and standard deviations ($\boldsymbol{\sigma}_\phi$, $\boldsymbol{\sigma}_\psi$), as follows:

$$\begin{aligned}
\text{Posterior}: \boldsymbol{\mu}_\phi &= \text{VN-MLP}(Z, \tilde{Z}), \quad & \log(\boldsymbol{\sigma}^{\mathbf{2}}_\phi) &= \mathrm{MLP}(Z, \tilde{Z}),\\
\text{Prior}: \boldsymbol{\mu}_\psi &= \text{VN-MLP}(\tilde{Z}), \quad & \log(\boldsymbol{\sigma}^{\mathbf{2}}_\psi) &= \mathrm{MLP}(\tilde{Z}).
\end{aligned} \tag{9}$$

We use an invariant MLP to learn the variance and apply it to the x, y, and z directions to enforce equivariance. We note that the conditional posterior is parameterized with both the ground truth and RDKit approximation, whereas the learned conditional prior only uses the RDKit.

G Decoder Architecture

We sample from the learned posterior (training) and learned prior (inference) to get $Z = \mu + \epsilon\sigma$, where ϵ is noise sampled from a standard Gaussian distribution as the input to the decoder. We note the role of the decoder is two-fold. The first is to convert the latent coarsened representation back into FG space through a process we call channel selection. The second is to refine the fine-grain representation autoregressively to generate the final low-energy coordinates.

Channel Selection. To explicitly handle all choices of coarse-graining techniques, our model performs variable-length backmapping. This aspect is crucial because every molecule can be coarsened into a different number of beads, and there is no explicit limit to the number of atoms a single bead can represent. Unlike CGVAE [Wang et al., 2022], which requires training a separate model for each choice in granularity N, CoarsenConf is capable of reconstructing FG coordinates from any N (illustrated in Fig. 4(III)).

CGVAE defines the process of channel selection as selecting the top k latent channels, where k is the number of atoms in a CG bead of interest. Instead of discarding all learned information in the remaining $F - k$ channels in the latent representation, we use a novel aggregated attention mechanism. This mechanism learns the optimal mixing of channels to reconstruct the FG coordinates and is illustrated in Fig. 5. The attention operation allows us to actively query our latent representation for the number of atoms we need, and draw upon similarities to the learned RDKit approximation that has been distilled into the latent space through the encoding process. Channel selection translates the CG latent tensor $Z \in R^{N\times F\times 3}$ into FG coordinates $x_{cs} \in R^{n\times 3}$.

Coordinate Refinement. Once channel selection is complete, we have effectively translated the variable-length CG representation back into the desired FG form. From here, x_{cs} is grouped into its corresponding CG beads but left in FG coordinates to do a bead-wise autoregressive generation of final low-energy coordinates (Fig. 4(IV)). As there is no intrinsic ordering of subgraphs, we use a breadth-first search that prioritizes larger subgraphs with large out-degrees. In other words, we generate a linear order that focuses on the largest, most connected subgraphs and works outward. We believe that by focusing on the most central component first, which occupies the most 3D volume, we can reduce the propagation of error that is typically observed in autoregressive approaches. We stress that by coarse-graining by torsion angle connectivity, our model learns the optimal torsion angles in an unsupervised manner, as the conditional input to the decoder is not aligned. CoarsenConf ensures each next generated subgraph is rotated properly to achieve a low coordinate and distance error.

Learning the Optimal Distortion. The decoder architecture is similar to the EGNN-based FG layer in the encoder. However, it differs in two important ways. First, we mix the conditional coordinates with the invariant atom features using a similar procedure as in the CG layer instead of typical graph matching. Second, we learn to predict the difference between the RDKit reference and ground truth conformations. This provides an upper error bound and enables us to leverage easy-to-obtain approximations more effectively.

More formally, a single decoder layer is defined as follows:

$$\boldsymbol{\mu}^{(t)} = \frac{1}{|\mathcal{V}_{prev}|}\sum_{k\in\mathcal{V}_{prev}} x_k, \tag{10a}$$

$$\tilde{\boldsymbol{h}}_i = \phi^m(\boldsymbol{h}_i^{(t)}, \boldsymbol{x}_i^{(t)}, \boldsymbol{\mu}^{(t)}, \|\boldsymbol{x}_i^{(t)} - \boldsymbol{\mu}^{(t)}\|^2), \forall i \in \mathcal{V}_{cur}, \tag{10b}$$

$$\boldsymbol{m}_{j\to i} = \phi^e(\tilde{\boldsymbol{h}}_i^{(t)}, \tilde{\boldsymbol{h}}_j^{(t)}, \|\boldsymbol{x}_i^{(t)} - \boldsymbol{x}_j^{(t)}\|^2, \|\boldsymbol{x}_i^{(t)} - \boldsymbol{x}_{ref,j}^{(t)}\|^2, \|\boldsymbol{x}_i^{(t)} - \boldsymbol{x}_{ref,i}^{(t)}\|^2), \forall (i,j) \in \mathcal{E}_{cur}, \tag{10c}$$

$$\boldsymbol{m}_i = \frac{1}{|\mathcal{N}(i)|}\sum_{j\in\mathcal{N}(i)} \boldsymbol{m}_{j\to i}, \forall i \in \mathcal{V}_{cur}, \tag{10d}$$

$$\boldsymbol{u}_{j'\to i} = a_{j'\to i}\boldsymbol{W}\boldsymbol{h}_{j'}^{(t)}, \forall i \in \mathcal{V}_{cur}, j' \in \mathcal{V}_{prev}, \tag{10e}$$

$$\boldsymbol{u}_i = \sum_{j'\in\mathcal{V}_{prev}} \boldsymbol{u}_{j'\to i}, \forall i \in \mathcal{V}_{cur}, \tag{10f}$$

$$\boldsymbol{x}_i^{(t+1)} = \boldsymbol{x}_{ref,i}^{(t)} + \sum_{j\in\mathcal{N}(i)} (\boldsymbol{x}_i^{(t)} - \boldsymbol{x}_j^{(t)})\phi^x(\boldsymbol{m}_{j\to i}), \forall i \in \mathcal{V}_{cur}, \tag{10g}$$

$$\boldsymbol{h}_i^{(t+1)} = (1-\beta)\cdot \boldsymbol{h}_i^{(t)} + \beta \cdot \phi^h(\tilde{\boldsymbol{h}}_i^{(t)}, \boldsymbol{m}_i, \boldsymbol{u}_i, \boldsymbol{f}_i), \forall i \in \mathcal{V}_{cur}, \tag{10h}$$

where ($\mathcal{V}_{cur}$, $\mathcal{E}_{cur}$) and ($\mathcal{V}_{prev}$, $\mathcal{E}_{prev}$) refer to the subgraph currently being generated and the set of all previously generated subgraphs, *i.e.*, the current state of the molecule. ϕ^m, ϕ^e, ϕ^x, and ϕ^h refer to separate shallow MLPs for the feature mixing, edge message calculation, coordinate update, and invariant feature update, respectively. Eq. 10(a-b) creates a mixed feature for each atom consisting of the current FG invariant feature and 3D position vectors (h and x), and the previous centroid μ and respective centroid distances. Eq. 10(c-d) defines the message passing operation that uses the aforementioned mixed features $\tilde{h}$ and a series of important distances between the model-based conformer and RDKit reference.Eq. 10(e-f) apply the same unidirectional attention updates seen in the encoder architecture. Eq. 10(g-h) update the position and feature vector for each atom using the above messages and attention coefficients, with f representing the original invariant node features $h^{\ell=0}$ and β a weighted update parameter. We emphasize that Eq. 10(g) formulates the overall objective as learning the optimal distortion of the RDKit reference to achieve the low-energy position *i.e.*, $x^* = x_{ref} + \Delta x$. The CG autoregressive strategy allows CoarsenConf to handle extremely large molecules efficiently, as the max number of time steps is equal to the max number of CG beads. CoarsenConf is trained using teacher forcing [Williams and Zipser, 1989], which enables an explicit mixing of low-energy coordinates with the current FG positions from channel selection Eq. 10(a-b).

H Model Configuration

Model. We present the model configuration that was used to generate the results in §3.1 - §M.2. Overall, the default model has 1.9M parameters: 1.6M for the encoder and 300K for the decoder. We note that as CoarsenConf uses graph matching, half the encoder parameters are used for each of the two inputs representing the same molecule in different spatial orientations. For both the encoder and decoder, we use five message-passing layers, a learning rate of $1e{-}3$ with an 80% step reduction after each epoch, and a latent space channel dimension (F) of 32. All other architectural parameters, such as feature mixing ratios or nonlinearities, were set following similar architectures [Huang et al., 2022, Deng et al., 2021, Stärk et al., 2022]. We present further ablations in Appendix §L. We note that the ability to share weights between the inputs as well as between each layer in the encoder is left as a hyperparameter. This could allow the encoder to see a 2x or 5x reduction in model size, respectively.

Compute. The QM9 model was trained and validated for five epochs in 15 hours using a single 40GB A100 GPU. We used a batch size of 600, where a single input refers to two graphs: the ground truth and RDKit approximate conformer. The DRUGs model was trained and validated for five epochs in 50 hours using distributed data-parallel (DDP) with 4 40GB A100 GPUs with a batch size of 300 on each GPU. For DRUGs, the GPU utilization was, on average, 66% as few batches contain very large molecules. In the future, lower run times can be achieved if the large molecules are more intuitively spaced out in each batch.

We note DDP has a negative effect on overall model benchmark performance due to the gradient synchronization but was used due to compute constraints. Without DDP, we expect the training time to take around 7 days, which is on par with Torsional Diffusion (4-11 days). We demonstrated that CoarsenConf achieves as good or better results than prior methods with less data and time, and these results can be further optimized in future work. We provide evidence of the negative effects of DDP in Appendix §L.

Optimal Transport reduces compute requirements. The optimal transport (OT) models were trained on 2 epochs on a single A6000 GPU for 8 and 15 hours total for QM9 and DRUGS, respectively. For OT details, see Appendix Eq. 11. Here, both models use the first 5 ground truth conformers. In real-world applications like polymer design, the availability of data is frequently limited and accompanied by a scarcity of conformers for each molecule. The current datasets, QM9 and DRUGS, do not mimic this setting very well. For example, on average, QM9 has 15 conformers per molecule, and DRUGS has 104 per molecule—both datasets have significantly more conformers than in an experimental drug design setting. Given this, rather than training on the first 30 conformers as done in Torsional Diffusion, we train on the first five (typically those with the largest Boltzmann weight) for QM9 and DRUGS, respectively.

I RDKit Approximate Conformer

Generating Approximate Conformers. For CoarsenConf's initial conditional approximations, we only use RDKit + MMFF when it can converge (~90% and ~40% convergence for QM9 and DRUGS, respectively). We emphasize that RDKit only throws an error when MMFF is not possible but often returns structures with a non-zero return code, which signifies incomplete and potentially inaccurate optimizations. Therefore, in generating the RDKit structures for training and evaluation, we filter for MMFF converged structures. We default to the base EKTDG-produced structures when either the optimization cannot converge, or MMFF does not yield enough unique conformers. CoarsenConf ultimately offers a solution that can effectively learn from traditional cheminformatics methods. This aspect of MMFF convergence has not been discussed in prior ML for MCG methods, and we leave it to future cheminformatics research to learn the causes and implications of incomplete optimizations.

Eliminating distribution shift with explicit conditioning. Both CoarsenConf and TD optimize $p(X|\mathcal{R})$ but utilize the RDKit approximations $\mathcal{R}$ in different ways. TD learns to update the torsion angles of $\mathcal{R}$, while CoarsenConf leverages CG information to inform geometric updates (coordinates, distances, and torsion angles) to translate $\mathcal{R}$ to X. Unlike TD, which uses a preprocessing optimization procedure to generate substitute ground truth conformers that mimic $p(\mathcal{R})$, CoarsenConf directly learns from both X and $\mathcal{R}$ through its hierarchical graph matching procedure. This directly addresses the distributional shift problem. We hypothesize that this, along with our angle-based CG strategy, leads to our observed improvements. Overall, CoarsenConf provides a comprehensive framework for accurate conformer generation that can be directly applied for downstream tasks such as oracle-based protein docking.

J GEOM Benchmark Discussion

Advantages and limitations of RMSD-based metrics. 3D coordinate-based metrics are commonly used to evaluate ML methods because these models are typically trained using spatial error-based loss functions (*i.e.*, MSE). However, for domain scientists, these metrics can be somewhat challenging to interpret. Low spatial error (measured by RMSD) is not directly informative of free energy, the primary quantity of interest to scientists [Spotte-Smith et al., 2021, Taylor et al., 2023]. Additionally, current spatial benchmarks can be categorized into two distinct types: precision and recall. Each of these metrics comes with its own advantages and limitations.

(1) Precision measures the generation accuracy. It tells us if each generated conformer is close to *any* one of the given ground truth structures, but it does not tell us if we have generated the lowest energy structure, which is the most important at a standardized temperature of 0 K. At industrial temperatures, the full distribution of generated conformers is more important than the ability to generate a single ground truth conformer (for the full RMSD error distribution, see Fig. 1).

(2) Recall compares each ground truth to its closest generated conformer. However, in many applications, we are only concerned with obtaining the lowest energy conformer, not all feasible ones. Furthermore, Recall is severely biased by the number of generated conformers for each molecule. Results worsen by up to ~60% for all models when we move from the previously set standard sampling budget for each molecule of $2L$ to min(L/32, 1), where L is the number of ground truth conformers (see Appendix Fig. 7). As we sample more molecules, we greatly influence the chance of reducing the AMR-Recall. Because of this dependency on the number of samples, we focus on the Precision metrics, which were consistent across all tested sample size budgets. We note that L is 104 on average for GEOM-DRUGS [Mansimov et al., 2019], so even L/32 is a reasonable number.

xTB energy and property prediction. We note the issues surrounding the RMSD metrics have always existed, and prior MCG methods have introduced energy-based benchmarks that we describe and report in Tab. 2. We note these energies are calculated with xTB, and thus are not very accurate compared to density functional theory (DFT), as it is limited by the level of theory used to produce the energies further discussed in Axelrod and Gómez-Bombarelli [2022]. Therefore, since current benchmarks mainly focus on gauging the effectiveness of the machine learning objective and less on the chemical feasibility and downstream use of the generated conformers, we use oracle-based protein docking-based to evaluate conformer quality on downstream tasks. These evaluations are highly informative, as molecular docking is a crucial step in the drug discovery process, as it helps researchers identify potential drug candidates and understand how they interact with their target proteins. The combination of RMSD, xTB energy, and downstream docking tasks presents a more comprehensive evaluation of generated conformers.

K QM9 Experimental Details

Both CoarsenConf and CoarsenConf-OT were trained on 5 conformers per ground truth molecule, compared to Torsional Diffusion's 30. We hypothesize that since CoarsenConf uses a one-to-one loss function, we are able to maintain high recall, whereas the OT model finds an optimal matching that focuses on precision. By adding more ground truth conformers, we hypothesize our model can better cover the true conformer space, improving recall, as the OT setting would not be as biased toward precision.

Table 5: Quality of generated conformer ensembles for the GEOM-QM9 test set ($\delta = 0.5$Å) in terms of Coverage (%) and Average RMSD (Å). Torsional Diffusion (TD) was benchmarked using its evaluation code and available generated molecules, per their public instructions. Note that CoarsenConf (5 epochs) was restricted to using 41% of the data used by TD (250 epochs) to exemplify a low-compute and data-constrained setting. OMEGA results were taken from Jing et al. [2022] (we were unable to run the coverage normalization).

	Recall				Precision			
	Coverage ↑		AR ↓		Coverage ↑		AR ↓	
Method	Mean	Med	Mean	Med	Mean	Med	Mean	Med
OMEGA	85.5	100.0	0.177	0.126	82.9	100.0	0.224	0.186
RDKit + MMFF	75.2	100.0	0.219	0.173	82.1	100.0	0.157	0.119
GeoMol	79.4	100.0	0.219	0.191	75.9	100.0	0.262	0.233
Torsional Diffusion	82.2	100.0	0.179	0.148	78.4	100.0	0.222	0.197
CoarsenConf	76.9	100.0	0.246	0.211	80.2	100.0	0.227	0.186
CoarsenConf-OT	56.1	50.0	0.361	0.345	80.2	100.0	0.149	0.108

L DRUGS Extended Benchmarks

Evaluation Details. All models in Tab. 1 were benchmarked with Torsional Diffusion's (TD) evaluation code and retrained if generated molecules were not public (using their public instructions). We note that TD uses higher-order tensor products to maintain equivariance ($\ell = 2$). In contrast, GeoMol, GeoDiff, and CoarsenConf use scalar-vector operations that are theoretically analogous to $\ell = 1$. CoarsenConf-OT uses an optimal transport (OT) loss with the same decoder architecture as in Fig. 4, but is no longer autoregressive. GeoDiff's code would not load, so we were able to evaluate the GeoDiff generated DRUGS molecules from the Torsional Diffusion authors' evaluation on the same test set.

Table 6: DRUGS-Precision equivariance ablations. OMEGA [Hawkins et al., 2010] results were taken from Jing et al. [2022]. All others were re-benchmarked using Torsional Diffusion's code with an error normalized Coverage score to prevent the masking out of method failures. This enforces that each method is fairly evaluated on the entire test set as now Coverage truly represents the percentage of the test set that meets the threshold criteria. OMEGA requires a commercial license, so we were unable to test the results ourselves, thus taking results from TD. As a non-ML method, we also assume OMEGA has no failures as each molecule in the test set is valid, which could artificially inflate the observed coverage scores.

	Coverage ↑		AMR ↓	
Method	Mean	Med	Mean	Med
RDKit	37.9	29.9	0.988	0.878
RDKit + MMFF	52.3	52.1	0.840	0.715
OMEGA	53.4	54.6	0.841	0.762
GeoDiff	23.7	13.0	1.131	1.083
GeoMol	40.5	33.5	0.919	0.842
Torsional Diffusion ($\ell = 1$)	48.9	50.0	0.804	0.758
Torsional Diffusion ($\ell = 2$)	52.1	53.7	0.770	0.720
CoarsenConf	43.8	35.5	0.914	0.829
CoarsenConf-OT	52.0	52.1	0.836	0.694

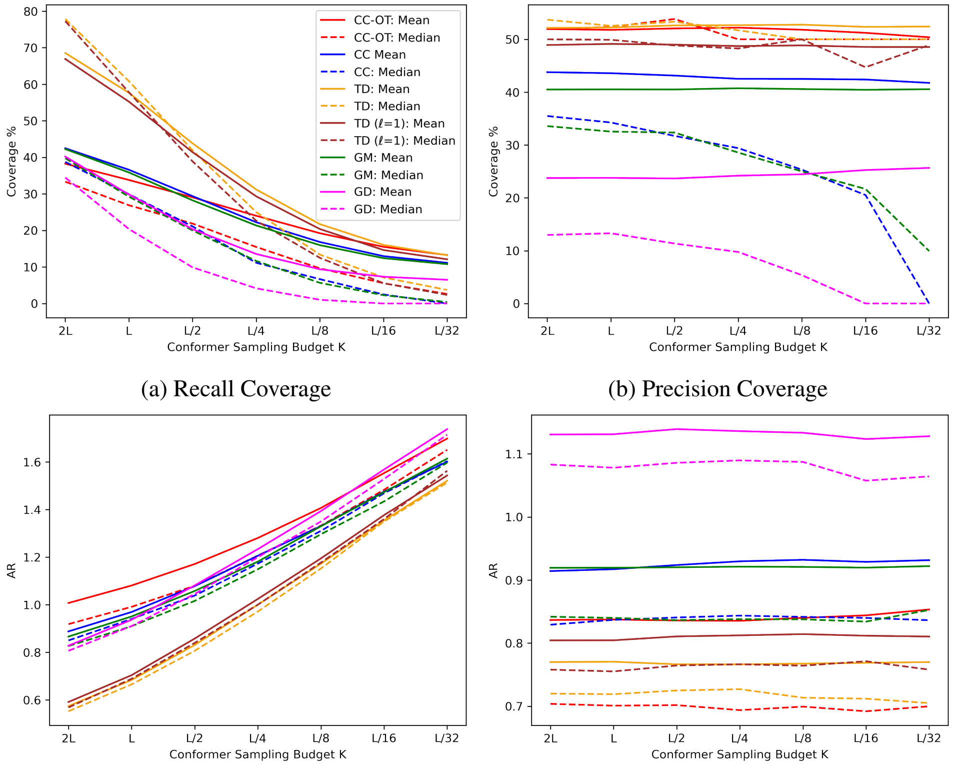
We copy the results from Tab. 1 and provide additional results, including TD for rotation order $\ell = 1$, OMEGA [Hawkins et al., 2010], and RDKit. This allows for a closer comparison to the scalar and vector operations that CoarsenConf employs to maintain equivariance. Using a lower rotation order results in slightly worse results in nearly all categories. We further discuss the implications of the choice in equivariant representation in Appendix §N.

Optimal Transport. In practice, our model generates a set of conformers, $\{\mathcal{C}_k\}_{k\in[1..K]}$, that needs to match a variable-length set of low-energy ground truth conformers, $\{\mathcal{C}^*_l\}_{l\in[1..L]}$. In our case, the number L of true conformers, or the matching between generated and true conformers is not known upfront. For these reasons, we introduce an optimal transport-based, minimization-only, loss function [Ganea et al., 2021]:

$$\mathcal{L}_{OT} = \min_{\mathbf{T} \in \mathcal{Q}_{K,L}} \sum_{k,l} T_{kl} \mathcal{L}(\mathcal{C}_k, \mathcal{C}_l^*),$$
$$\mathcal{L}(\mathcal{C}_k, \mathcal{C}_l^*) = \text{MSE}(\mathcal{C}_k, \mathcal{C}_l^*) + \text{distance error}(\mathcal{C}_k, \mathcal{C}_l^*), \tag{11}$$

where $\mathbf{T}$ is the ***transport plan*** satisfying $\mathcal{Q}_{K,L} = \{\mathbf{T} \in \mathbb{R}_+^{K \times L} : \mathbf{T}\mathbf{1}_L = \frac{1}{K}\mathbf{1}_K, \mathbf{T}^T\mathbf{1}_K = \frac{1}{L}\mathbf{1}_L\}$. The minimization w.r.t. T is computed quickly using the Earth Mover Distance and the POT library [Flamary et al., 2021]. As the OT loss focuses more on finding the optimal mapping from generated conformers to ground truth reference, we removed the autoregressive decoding path of CoarsenConf and replaced it with a single pass with the same decoder architecture. The underlying loss function, which is tasked to minimize MSE coordinate error, and interatomic distance error is the same in both (Eq. 4), the autoregressive (AR) and non-AR OT-based loss functions. The OT version additionally finds the optimal mapping between the generated and ground truth structures, which better aligns with the AMR and Coverage benchmarks.

Hyperparameter Ablations. We experimented with increasing the latent channels (F) from 32 to 64 and 128, and introducing a step-wise distance loss and KL regularization annealing schedule, as done in the QM9 experiments. Both these experiments resulted in slightly worse performance when limited to 2 conformers per training molecule. We hypothesize that due to the DRUGs molecules being much larger than those in QM9, more training may be necessary, and a more sensitive annealing schedule may be required.



(a) Recall Coverage

(b) Precision Coverage

(c) Recall AMR

(d) Precision AMR

Figure 7: GEOM-DRUGS evaluation as a function of number of generated conformers. GEOM-DRUGS has 104 conformers per molecule on average. Recall is heavily dependent on the sampling budget. Precision is mostly stable. Lower AMR and higher coverage is better, but coverage is set by an arbitrary threshold, which in this case is 0.75Å. Results show CoarsenConf (CC), Torsional Diffusion (TD), GeoMol (GM), and GeoDiff (GD).

GEOM-DRUGS Recall Results. Fig. 7 demonstrates extensive Precision and Recall results for a wide range of tested sampling budgets for GEOM-DRUGS. We see that only Precision is stable across nearly all values. Due to the extreme sensitivity of the Recall metric and little difference in model performance for reasonable sampling budgets, we focus on Precision for QM9 and DRUGS. We also note that while CoarsenConf-OT saw worse recall results for QM9, this was not the case for DRUGS. In the case of DRUGS, CoarsenConf-OT achieves the learning objective of instilling force field optimizations as the lower error bound and does so with very little training and inference time.

M Oracle-based Protein Docking

M.1 Flexible Oracle-based Protein Docking

We evaluate MCG models, pretrained on GEOM-DRUGS, using nine protein docking oracle functions provided by the Therapeutics Data Commons (TDC) [Huang et al., 2021].

Problem setup. Starting with a known 2D ligand[3] molecule, protein, and desired 3D protein binding pocket, we measure conformer quality by comparing the predicted binding affinity of generated conformers of each MCG method. TDC's protein docking oracle functions take in a ligand SMILES string, generate a 3D conformer, and try multiple poses, before ultimately returning the best binding affinity via Autodock Vina's flexible docking simulation. We augment TDC with the ability to query ML models pretrained on GEOM-DRUGS, instead of the built-in RDKit + MMFF approach for MCG. For each evaluated MCG method, we generate 50 conformers for each of the nine ligands and report the best (lowest) binding affinity. Given that the ligand and protein identity and the protein pocket are fixed, this task measures the quality of 3D conformer coordinates through their binding efficacy to the specified pocket. We note that this task is indicative of real-world simulation workflows.

Results. CoarsenConf significantly outperforms prior MCG methods on the TDC oracle-based affinity prediction task (Tab. 7). CoarsenConf generates the best ligand conformers for 8/9 tested proteins, with improvements of up to 53% compared to the next best method. CoarsenConf is 1.46 kcal/mol better than all methods when averaged over all 9 proteins, which corresponds to a 14.4% improvement on average compared to the next best method.

M.2 Rigid Oracle-based Protein Docking

We evaluate MCG models on rigid oracle-based protein docking. We use the 166000 protein-ligand complexes from the CrossDocked [Francoeur et al., 2020] training set (Appendix §M for more details).

Problem setup. Similar to the flexible docking task (§M.1), we can generate conformers of known ligands for known protein pockets, but now have them only undergo a rigid pocket-specific energy minimization before predicting the binding affinity. To handle the fact that MCG generates ligand structures in a vacuum and has no information about the target protein, we assume the centroid of the ground truth ligand is accurate, and translate each generated structure to match the center.

We use AutoDock Vina [Eberhardt et al., 2021] and its local BFGS energy optimization (similar to that done with xTB), *i.e.* a relaxation of all tested structures (including the ground truth), following the SBDD benchmarking framework Guan et al. [2023]. We report the difference between the generated

[3] No ground truth 3D structures. All ligand SMILES taken from Protein Data Bank: https://www.rcsb.org/

Table 7: Quality of best generated conformer for known protein ligands for all 9 proteins from the TDC library. Quality is measured by free energy change (kcal/mol) of the binding process with AutoDock Vina's flexible docking simulation (↓ is better).

	Best Protein-Conformer Binding Affinity (↓ is better)								
Method	3PBL	2RGP	1IEP	3EML	3NY8	4RLU	4UNN	5M04	7L11
RDKit + MMFF	-8.26	-11.42	-10.75	-9.26	-9.69	-8.72	-9.73	-9.53	-9.19
GeoMol	-8.23	-11.49	-11.16	-9.39	**-11.66**	-8.85	-10.28	-9.31	-9.29
Torsional Diffusion	-8.53	-11.34	-10.76	-9.25	-10.32	-8.96	-10.65	-9.61	-9.10
CoarsenConf	**-8.81**	**-12.93**	**-16.43**	**-9.82**	-11.26	**-9.54**	**-11.62**	**-14.00**	**-9.43**

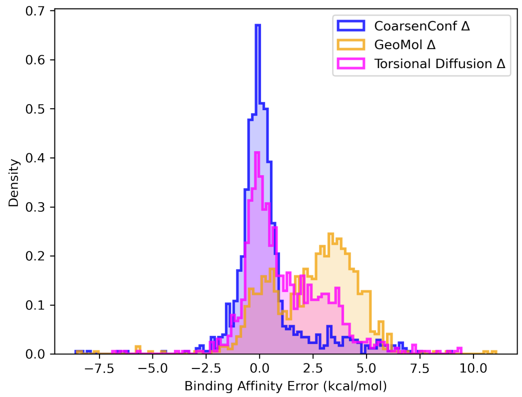


Table 8: Binding affinity error distribution statistics in kcal/mol (more negative is better).

Method	Mean	Min
GeoMol	2.476	-8.523
Torsional Diffusion	1.178	-6.876
CoarsenConf	**0.368**	**-8.602**

Figure 8: Binding affinity (↓ is better) error distributions for 100k conformer-protein complexes in the CrossDocked dataset. The error is the difference in binding affinity between the generated and ground truth energy minimized 3D ligands.

structure's minimum binding affinity and that of the ground truth 3D ligand. Unlike the docking simulation, Vina's energy minimization does not directly adjust the torsion angles or internal degrees of freedom within the ligand. Instead, it explores different atomic positions and orientations of the entire ligand molecule within the binding site of the protein to find energetically favorable binding poses. By further isolating the MCG-generated structures, this task better evaluates the generative capacity of MCG models. While the original SBDD task [Peng et al., 2022, Guan et al., 2023] reports the Vina minimization score as its main metric, it requires the 3D ligand to be generated from scratch. Here, we use the SBDD framework to isolate the generated ligand conformer as the only source of variability to evaluate the biological and structural quality of MCG models in an unbiased fashion.

Results. We report the results for 100,000 unique conformer-protein interactions; note that there is a large cost to run the binding affinity prediction (see Appendix §M for more details). We also emphasize that the presented evaluation is not to be confused with actual docking solutions, as a low-energy conformer is not always guaranteed to be the best binding pose. Instead, we employ an unbiased procedure to present empirical evidence for how CoarsenConf can generate input structures to Vina that significantly outperform prior MCG models in achieving the best binding affinities.

Fig. 8 further demonstrates CoarsenConf's superior performance on orders of magnitude more protein complexes than the prior flexible oracle task. CoarsenConf decreases the average error by 56% compared to TD, and is the only method not to exhibit bimodal behavior with error greater than zero. We hypothesize that the success of MCG methods in matching ground truth structures is influenced by the complexity of protein pockets. In simpler terms, open pockets better facilitate Vina's optimization, but the initial position generated by MCG remains crucial. Furthermore, if the initial MCG-generated structure did not matter, the distributions for each MCG model would be identical. We also note that the lowest energy conformer is not always the optimal ligand structure for binding, but in our experiments, it yields the best input for the Vina-based oracles. Overall CoarsenConf best approximates the ground truth ligand conformers of CrossDocked and generates the best structures for Vina's rigid energy relaxation and binding affinity prediction.

Dataset Details. We utilize the oracle-based protein docking task as molecules with higher affinity (more negative) have more potential for higher bioactivity, which is significant for real-world drug discovery. We use the CrossDocked2020 trainset consisting of 166000 protein-ligand interactions (2,358 unique proteins and 11,735 unique ligands) and its associated benchmarks, as it has been heavily used in Structure-based drug discovery as defined by Peng et al. [2022], Guan et al. [2023].

The CrossDocked2020 dataset is derived from PDBBind but uses smina [Koes et al., 2013], a derivative of AutoDock Vina with more explicit scoring control, to generate the protein-conditioned ligand structures to yield ground truth data. We note that based on the raw data, 2.2 billion conformer-protein interactions are possible, but we filtered out any ground-truth example that AutoDock Vina failed to score. Furthermore, in the TDC oracle-based task, each ligand is known to fit well in the given protein. CrossDocked2020, on the other hand, consists of various ligand-protein interactions, not all of which are optimal, making the overall task more difficult.

We note that while it takes on the orders of hours to generate 1.2M conformers (100 conformers per molecule), it takes on the orders of ~weeks to months to score each conformer for up to the 2,358 unique proteins for each evaluated method (evaluation time is 100x the time to score the ground truth data as we generate 100 conformers per molecule). As a result, we report the results for the first 100,000 conformer-protein interactions.

N Limitations

As demonstrated in §3.1-§M.2, CoarsenConf significantly improves the accuracy and reduces the overall data usage and runtime for conformer generation. However, CoarsenConf also has some limitations that we will discuss in this section.

Autoregressive generation. While CoarsenConf improves accuracy with reduced training time and overall data, autoregressive generation is the main bottleneck in inference time. We linearize the input molecule based on spatially significant subgraphs and then process each one autoregressively. For a model with k torsion angles, we need $k + 1$ passes through our decoder. Coarse-graining is an effective strategy to reduce the number of decoder passes compared to traditional atom-wise autoregressive modeling. For example, for a given molecule, the number of torsion angles (which we use to coarse-grain) is significantly less than the number of atoms. Our choice of coarse-graining strategy allows us to break the problem into more manageable subunits, making autoregressive modeling a useful strategy, as it provides greater flexibility and control by allowing conditional dependence. CoarsenConf is a good example of the trade-offs that exist between generative flexibility and speed. We target this limitation by introducing a non-autoregressive version with an optimal transport loss. We see this improves the overall GEOM results, at the slight cost of a higher right tail of the error distribution.

Optimal Transport. While CoarsenConf-OT trained with a non-autoregressive decoder with an optimal transport loss significantly outperforms prior methods and accomplishes the goal of effectively learning from traditional cheminformatics methods, the recall results still have room for improvement, especially for QM9. While CoarsenConf (no OT) achieves competitive results, we believe that continuing to focus on how to better integrate physics and cheminformatics into machine learning will be crucial for improving downstream performance. Due to the above concerns, we chose to evaluate our non-OT model on the property prediction and protein docking tasks, as wanted to use our best model denoted by the lowest overall RMSD error distribution.

Approximate structure error. The success of learning the optimal distortion between low-energy and RDKit approximate structure depends on having reasonable approximations. While CoarsenConf relaxes the rigid local structure assumption of Torsional Diffusion in a way that leverages the torsional flexibility in molecular structures, it still depends on an approximate structure. This is a non-issue in some instances, as RDKit does well. In more experimental cases for larger systems, the RDKit errors may be too significant to overcome. We emphasize that the underlying framework of CoarsenConf is adjustable and can learn from scratch, not only the distortion from approximate RDKit structures. In some cases, this may be more appropriate if the approximations have particularly high error. We leave to future work to explore the balance between the approximation error and the inductive bias of learning from approximate structures, as well as methods to maintain flexibility while avoiding the issues of conditioning on out-of-distribution poor approximations.

Equivariance. As CoarsenConf uses the EGNN [Satorras et al., 2021] framework as its equivariant backbone and thus only scalar and vector operations, there is no simple way to incorporate higher-order tensors. As the value of using higher-order tensors is still actively being explored, and in some cases, the costs outweigh the benefits, we used simple scalar and vector operations and avoided expensive tensor products. We leave exploring the use of higher-order equivariant representations to future work, as it is still an ongoing research effort [Han et al., 2022].