MolCRAFT: Structure-Based Drug Design in Continuous Parameter Space

Yanru Qu [*1 2] **Keyue Qiu** [*1 3] **Yuxuan Song** [*1 3] **Jingjing Gong** [1]
Jiawei Han [2] **Mingyue Zheng** [4] **Hao Zhou** [1] **Wei-Ying Ma** [1]

Abstract

Generative models for structure-based drug design (SBDD) have shown promising results in recent years. Existing works mainly focus on how to generate molecules with higher binding affinity, ignoring the feasibility prerequisites for generated 3D poses and resulting in *false positives*. We conduct thorough studies on key factors of ill-conformational problems when applying autoregressive methods and diffusion to SBDD, including mode collapse and hybrid continuous-discrete space. In this paper, we introduce MolCRAFT, the first SBDD model that operates in the continuous parameter space, together with a novel noise reduced sampling strategy. Empirical results show that our model consistently achieves superior performance in binding affinity with more stable 3D structure, demonstrating our ability to accurately model interatomic interactions. To our best knowledge, MolCRAFT is the first to achieve reference-level Vina Scores (-6.59 kcal/mol) with comparable molecular size, outperforming other strong baselines by a wide margin (-0.84 kcal/mol). Code is available at https://github.com/AlgoMole/MolCRAFT.

1. Introduction

Structure-based drug design (SBDD) advances drug discovery by leveraging 3D structures of biological targets, thereby facilitating efficient and rational design of molecules within a certain chemical space of interests (Wang et al., 2022; Isert et al., 2023). Recently, generative models for molecules have emerged as a promising direction, which could streamline SBDD by directly proposing desired molecules, eliminating the need for exhaustive blind search in the vast space (Walters, 2019; Luo et al., 2021). Recent progress in SBDD can be divided into two categories, *i.e.* auto-regressive models (Luo et al., 2021; Peng et al., 2022; Zhang et al., 2023) as next-token prediction for text generation, and diffusion models (Guan et al., 2022; 2023) as for image generation.

The essential criteria for drug-like candidate molecules are outlined as follows: (i) *high affinity* towards specific binding sites (*a.k.a*, protein pockets), where a higher affinity indicates better performance, (ii) *satisfactory drug-like properties*, such as synthesizability and drug-likeness scores, which often serve as thresholds for filtering out unfavorable compounds (Ursu et al., 2011; Tian et al., 2015), and (iii) *well-conformational 3D structure*, which needs special attention as there is a danger of generating unrealistic 3D conformations yet with deceptively high affinities.

However, current generative models focus primarily on (i) and (ii), whereas we observe that the generated molecules often fail to meet all criteria simultaneously, especially for (iii) conformational stability. This challenge manifests as the *False Positives* phenomenon (FP) in generative modeling of SBDD, where models yield molecules that reside outside the true molecular manifold yet appear to exhibit good binding affinity after redocking. Specifically, these molecules suffer from *distorted structure*, displaying problematically unusual topology, and *inferior binding mode*, whereby the generated poses fail to capture true interactions and may even violate biophysical constraints, and thus go through post-fixes and significant rearrangements from docking software. Such problems threaten to jeopardize reliable model assessment, ultimately hindering their application in SBDD (Sec. 2.1).

Both autoregressive and diffusion-based models exhibit challenges with generating accurate molecular conformations, yet these issues stem from distinct causes. In Sec. 2.2, we delve into the *mode collapse* issue faced by autoregressive methods. Empirically, they tend to repeatedly generate a limited number of specific (sub-)structures due to an unnatural atom ordering imposed during generation. On the other hand, the problem with diffusion-based models is attributed

[*]Equal contribution Work was done during Yanru Qu's internship at Tsinghua AIR. [1]Institute for AI Industry Research (AIR), Tsinghua University [2]University of Illinois Urbana-Champaign, USA [3]Department of Computer Science and Technology, Tsinghua University [4]Shanghai Institute of Materia Medica, Chinese Academy of Sciences. Correspondence to: Hao Zhou <zhouhao@air.tsinghua.edu.cn>, Jingjing Gong <gongjingjing@air.tsinghua.edu.cn>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

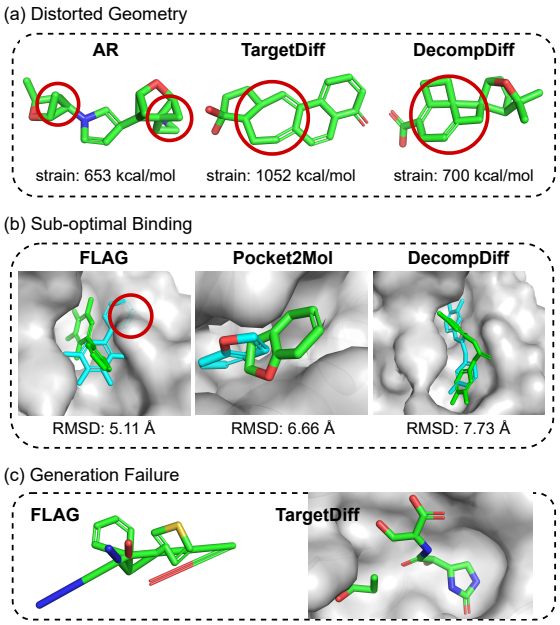


Figure 1: Typical failure modes. (**a**) Unusual 3-membered rings generated by AR, large fused rings with more than 7 atoms generated by diffusion models. (**b**) Examples of steric clashes by FLAG, and other ligand undergoing significant conformational rearrangements upon redocking (Before: blue. After: green). (**c**) Failures in generation process. Left: atoms mis-connected in autoregressive sampling. Right: incomplete molecules with multiple components.

to *denoising in hybrid yet highly twisted space*, which is a blend of discrete atomic types and continuous atomic coordinates. Different modalities need to be carefully handled in the hybrid space, and lack of consideration might result in severely strained and infeasible outputs (Sec. 2.3).

Notably, DecompDiff (Guan et al., 2023) proposes to inject the molecular inductive bias by manually decomposing ligands into arms and scaffolds priors before training, and utilizing validity guidance in sampling. However, it cannot fully address the ill-conformational problem, since the inductive bias is simply impossible to enumerate. As shown in Fig. 2, for common C-N and C-O bond with two modes of typical length distribution, nearly all SBDD models are struggling to fit this substructural pattern. More visualization results can be referred to in Fig. 8, 9, 10, Appendix D.

In order to capture the complicated data manifold for molecules, we take a shift to a unified continuous parameter space instead of a hybrid space, inspired by Graves et al. (2023). We propose MolCRAFT (Continuous paRAmeter space Facilitated molecular generaTion), which not only alleviates the mode collapse issue by non-autoregressive generation as in its diffusion counterparts, but also addresses the continuous-discrete gap by applying continuous noise and smooth transformation.

Our contributions can be summarized as follows:

- We investigate the challenges of current SBDD models, and identify key issues including the mode collapse of autoregressive methods, and the gap of continuous-discrete space when applying diffusion models.
- We propose MolCRAFT to address these two issues, which is a unified SE-(3) equivariant generative model, equipped with sampling in the parameter space that avoids further noise.
- We conduct comprehensive evaluation under controlled molecular sizes. Experiments show that our model generates high-affinity binders with feasible 3D poses. To our best knowledge, we are the first to achieve **reference-level Vina Scores** (-6.59 vs. -6.36 kcal/mol) with comparable molecule size, outperforming other strong baselines by a wide margin (-0.84 kcal/mol).

2. Challenges of Generative Models in SBDD

We provide an overview of current obstacles in pocket-based generation. We summarize common failures in Sec. 2.1, and then investigate the underlying problems, *i.e.* the *mode collapse* issue of autoregressive-based models in Sec. 2.2, and *hybrid denoising* issue of diffusion-based models in Sec. 2.3. Based on the aforementioned challenges, we propose to generate molecules in the continuous parameter space.

2.1. Failure Modes of Generated Molecules

As shown in Fig. 1, we divide undesired molecules in SBDD into three categories:

(a) **Distorted geometry**. We visualize the generated molecules at median strain energy (see Table 2), and models tend to produce either too many uncommon 3- or 4-membered rings, or extra-large rings with unstable structures, leading to much higher strain energy.

(b) **Inferior binding mode**. We observe a notable number of generated ligand conformations rearrange drastically after redocking, with some even violating biophysical constraints and producing steric clashes with the protein surface. This suggests that 3D SBDD models do not capture true interatomic interactions and rely on post-fixing via redocking as noted by Harris et al. (2023), which severely harms the credibility of generating molecules directly in 3D space.

(c) **Generation failure**. Autoregressive models tend to misplace an element and terminate prematurely, while diffusion models might generate incomplete molecules with disconnected parts, limiting sample efficiency.

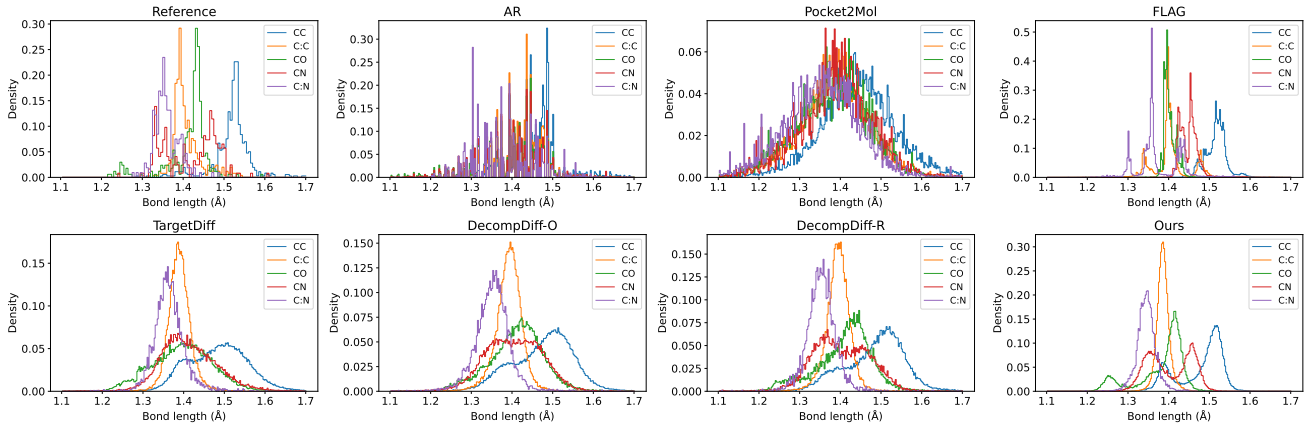


Figure 2: Bond length distribution of reference and generated molecules by autoregressive models (upper row) and non-autoregressive models (lower row) for top-5 frequent bond types.

The above problems hinder the applicability of SBDD models. In the following sections, we provide deeper understanding of the problematic methods underlying these failures.

2.2. Molecular Mode Collapse

The *mode collapse* issue focuses on the empirical performance of SBDD methods that tend to generate a limited number of specific (sub-)structures, where atom-based autoregressive models have displayed a particular preference for certain modes. We provide quantitative results from both the chemical and geometrical perspectives.

Chemical assessment is shown in Table 1. In order to measure *molecular distribution*, we report the percentage of unique samples (**Unique**) averaged on different pockets.[1] It can be seen that the ratio of unique molecules of AR (Luo et al., 2021) and Pocket2Mol (Peng et al., 2022) is considerably lower than other counterparts. Moreover, DecompDiff (Guan et al., 2023) is also found to generate repeated molecules, possibly due to its use of prior clusters At the *substructural* level, we report the percentage of molecules with certain types of rings defined by Jiang et al. (2024), with respect to all ring-structured molecules. Pocket2Mol displays a preference for more fused rings as also noted by Harris et al. (2023), while AR exhibits an obvious pattern in generating repeated three-membered rings.

Geometrically measured, as shown in Fig. 2, atom-based autoregressive methods model the bond lengths for different bond types similarly, where reference distribution is multimodal and varies across different types, while Pocket2Mol only captures a single mode, and for AR different bond lengths are distributed in a very similar fashion.

[1]Here we remove all post-filters from autoregressive models that avoid generating duplicate or invalid molecules, in order to faithfully demonstrate their performances. In all other experiments, we stick to the original implementation.

Table 1: Percentage (%) of molecular modes in terms of distribution and substructures. *Note:* Fused refers to 80 specific rings, 3-Ring denotes three-membered rings, and so on. Highly deviated values are highlighted in ***bold Italic***.

	Unique	Fused	3-Ring	4-Ring	5-Ring	6-Ring
Reference	-	30.0	4.0	0.0	49.0	84.0
Train	-	21.6	3.8	0.6	56.1	90.9
AR	***36.2***	39.7	***50.8***	0.8	35.8	71.9
Pocket2Mol	***73.7***	***52.0***	0.3	0.1	38.0	88.6
FLAG	99.7	42.4	3.1	0.0	39.9	84.7
TargetDiff	99.6	37.8	0.0	***7.3***	57.0	76.1
Decomp-O	***61.6***	13.1	9.0	***11.4***	64.0	83.3
Decomp-R	***50.3***	28.1	5.4	***8.3***	51.5	65.6
Ours	97.7	30.9	0.0	0.6	47.0	85.1

FLAG (Zhang et al., 2023) generates fragment-by-fragment, which avoids collapsing by explicitly incorporating optimal and diverse substructures. But it suffers from more severe error accumulation, resulting in significant steric clashes and undesirable Vina Score (see Sec. 5.2). Generally speaking, autoregressive models are still trapped in sub-optimal performance. Intuitively, such limitations could be attributed to an unnatural atom ordering imposed during generation.

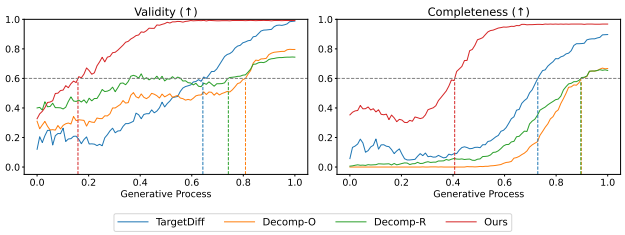


Figure 3: Percentage of valid, complete molecules in the trajectories during generative process.

2.3. Hybrid Continuous-Discrete Space

Diffusion-based models, on the other hand, successfully alleviate mode collapse problem via non-autoregressive generation in terms of substructural distribution (see Fig. 2). However, the inconsistency between different modalities has long troubled molecular generation models, as suggested by MolDiff (Peng et al., 2023) and EquiFM (Song et al., 2024b), where a careful design of either different noise levels or different probability paths is required.

A key insight is that the hybrid continuous-discrete space poses challenges to accurately capture the complicated data manifold for molecules, where the sample space in diffusion models is exposed to high variance, and the intermediate noisy latent is very likely to go outside the manifold. Inspired by GeoBFN (Song et al., 2024a), we propose to operate within the fully continuous pamarater space, which enables considerably lower input variance and a smooth transformation towards the target distribution.

To further illustrate the difference between continuous-discrete diffusion and our fully continuous MolCRAFT, we sample 10 molecules for each of the 100 test proteins, and plot the curves of the ratio of valid, complete molecules against different timesteps during sampling. As shown in Fig. 3, continuous-discrete diffusions heavily rely on the latter steps, passing a certain validity and completeness threshold in the final 60%-90% stage where noise scales are lower, while MolCRAFT approaches target distribution far earlier (in the first 20%-40% steps), thereby possessing greater capacity to progressively refine and adjust the generated feasible structures, resulting in better conformations.

3. Preliminary

In this section, we briefly overview Bayesian Flow Networks (BFN) (Graves et al., 2023) in comparison with diffusion models for SBDD. For its detailed formulation and mathematical details, we refer readers to Appendix A.

3.1. Problem Definition

Structure-based Drug Design (SBDD) can be formulated as a conditional generation task. Given input protein binding site $\mathcal{P} = \{(\mathbf{x}_P^{(i)}, \mathbf{v}_P^{(i)})\}_{i=1}^{N_P}$, which contains N_P atoms with each $\mathbf{x}_P^{(i)} \in \mathbb{R}^3$ and $\mathbf{v}_P^{(i)} \in \mathbb{R}^{D_P}$ correspond to atom coordinates and atom features, respectively (*e.g.*, element types, backbone or side chain indicator). The output is a ligand molecule $\mathcal{M} = \{(\mathbf{x}_M^{(i)}, \mathbf{v}_M^{(i)})\}_{i=1}^{N_M}$, where $\mathbf{x}_M^{(i)} \in \mathbb{R}^3$ and $\mathbf{v}_M^{(i)} \in \mathbb{R}^{D_M}$, N_M is the number of atoms in molecule. For convenience, we denote $\mathbf{p} = [\mathbf{x}_P, \mathbf{v}_P], (\mathbf{x}_P \in \mathbb{R}^{N_P \times 3}, \mathbf{v}_P \in \mathbb{R}^{N_P \times D_P})$ and $\mathbf{m} = [\mathbf{x}_M, \mathbf{v}_M], (\mathbf{x}_M \in \mathbb{R}^{N_M \times 3}, \mathbf{v}_M \in \mathbb{R}^{N_M \times D_M})$ as the concatenation of all protein or ligand atoms.

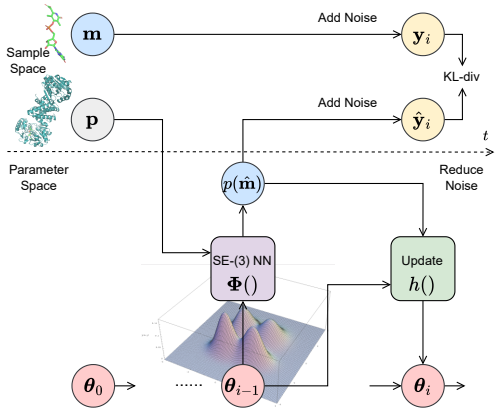


Figure 4: Overall Architecture.

3.2. Molecular Generation in Parameter Space

The overall architecture of MolCRAFT are shown in Fig. 4. The generative process is viewed as message exchanges between a sender and a receiver, where the sender is only visible in sample space, and the receiver makes the guess from its understanding of samples and parameters. In every round of communication, the sender selects a molecule datapoint $\mathbf{m}$, adds noise for timestep t_i according to *sender distribution* $p_S(\mathbf{y}_i \mid \mathbf{m}; \alpha_i)$, and sends the noisy latent $\mathbf{y}$ to receiver, resembling the forward diffusion process. Here α_i is a noise factor from the schedule $\beta(t_i)$.

The receiver, on the other hand, outputs the reconstructed molecule $\hat{\mathbf{m}}$ based on its previous knowledge of parameters $\boldsymbol{\theta}$, yielding *output distribution* p_O. With the sender's noisy factor α known, the receiver can also add noise to the estimated output and give the predicted noisy latent, arriving at *receiver distribution* p_R.

$$p_R(\mathbf{y}_i \mid \boldsymbol{\theta}_{i-1}, \mathbf{p}; t_i) = \mathop{\mathbb{E}}_{\hat{\mathbf{m}} \sim p_O} p_S(\mathbf{y}_i \mid \hat{\mathbf{m}}; \alpha_i), \quad (1)$$

$$\text{where} \quad p_O(\hat{\mathbf{m}} \mid \boldsymbol{\theta}_{i-1}, \mathbf{p}; t_i) = \boldsymbol{\Phi}(\boldsymbol{\theta}_{i-1}, \mathbf{p}, t_i). \quad (2)$$

$\boldsymbol{\Phi}$ is a neural network which is expected to reconstruct clean sample $\hat{\mathbf{m}}$ given parameters $\boldsymbol{\theta}_{i-1}$, pocket $\mathbf{p}$ and time t_i.

The key difference between BFN and diffusion lies in its introduction of parameters. Thanks to structured Bayesian updates defined via Bayesian inference, the receiver is able to maintain fully continuous parameters and perform closed-form update on its belief of parameters. *Bayesian update distribution* p_U stems from the Bayesian update function h,

$$p_U(\boldsymbol{\theta}_i \mid \boldsymbol{\theta}_{i-1}, \mathbf{m}, \mathbf{p}; \alpha_i) = \mathop{\mathbb{E}}_{\mathbf{y}_i' \sim p_S} \delta\Big(\boldsymbol{\theta}_i - h(\boldsymbol{\theta}_{i-1}, \mathbf{y}_i, \alpha_i)\Big), \quad (3)$$

where $\delta(\cdot)$ is Dirac delta distribution. The parameter space enables arbitrarily applying noise as long as the Bayesian update is tractable, and eliminates the need to invert a predefined forward process as in diffusion models.

According to the nice additive property of accuracy (Graves et al., 2023), the *Bayesian flow distribution* p_F could be obtained to achieve simulation-free training, once teacher forcing with $\mathbf{m}$ is applied:

$$p_F(\boldsymbol{\theta}_i \mid \mathbf{m}, \mathbf{p}; t_i) = \mathop{\mathbb{E}}_{\boldsymbol{\theta}_{1\ldots i-1} \sim p_U} p_U(\boldsymbol{\theta}_i \mid \boldsymbol{\theta}_{i-1}, \mathbf{m}, \mathbf{p}; \alpha_i) = p_U(\boldsymbol{\theta}_i \mid \boldsymbol{\theta}_0, \mathbf{m}, \mathbf{p}; \beta(t_i)) \tag{4}$$

Therefore, the training objective for n steps is to minimize:

$$L^n(\mathbf{m}, \mathbf{p}) = \mathop{\mathbb{E}}_{i \sim U(1,n)} \mathop{\mathbb{E}}_{\mathbf{y}_i \sim p_S, \boldsymbol{\theta}_{i-1} \sim p_F} D_{\mathrm{KL}}(p_S \parallel p_R). \tag{5}$$

4. Methodology

We introduce our proposed MolCRAFT in as follows: in Sec. 4.1, we demonstrate how to model continuous atom coordinates and discrete atom types within BFN framework, with the guarantee of SE-(3) equivariance for molecular data. Then in Sec. 4.2, we elaborate our novel sampling strategy tailored for the parameter space. Within the fully continuous and differentiable space, MolCRAFT is able to capture the global connection between different modalities, and sample efficiently with low variance.

4.1. Resolving Different Modalities in Parameter Space

This section demonstrates how to resolve continuous atom coordinates and discrete atom types in parameter space.

Unified parameter $\boldsymbol{\theta} \stackrel{\text{def}}{:=} [\boldsymbol{\theta}^x, \boldsymbol{\theta}^v]$ Following Hoogeboom et al. (2022), continuous atom coordinates $\mathbf{x}$ are characterized by Gaussian distribution $\mathcal{N}(\mathbf{x} \mid \boldsymbol{\mu}, \rho^{-1}\mathbf{I})$, and we set $\boldsymbol{\theta}^x = \{\boldsymbol{\mu}, \rho\}$, where $\boldsymbol{\mu}$ is learned and ρ is predefined by noise factor α. The Bayesian update function $\{\boldsymbol{\mu}_i, \rho_i\} \leftarrow h(\{\boldsymbol{\mu}_{i-1}, \rho_{i-1}\}, \mathbf{y}^x, \alpha_i)$ is defined as:

$$\rho_i = \rho_{i-1} + \alpha_i \tag{6}$$

$$\boldsymbol{\mu}_i = \frac{\boldsymbol{\mu}_{i-1}\rho_{i-1} + \mathbf{y}^x \alpha_i}{\rho_i} \tag{7}$$

For discrete atom types v, we use a categorical distribution $\boldsymbol{\theta}^v \in \mathbb{R}^{N_M \times K}$, and update it given α' via

$$h(\boldsymbol{\theta}^v_{i-1}, \mathbf{y}^v, \alpha'_i) \stackrel{\text{def}}{:=} \frac{e^{\mathbf{y}^v}\boldsymbol{\theta}^v_{i-1}}{\sum_{k=1}^K e^{\mathbf{y}^v_k}(\boldsymbol{\theta}^v_{i-1})_k} \tag{8}$$

For prior $\boldsymbol{\theta}_0$, we adopt standard Gaussian and uniform distribution respectively, following Graves et al. (2023).

Applying noise for different modalities Thanks to the continuous nature of parameters, we are able to apply the following continuous noise even for discrete atom types, instantiating the *sender distribution* p_S:

$$p_S(\mathbf{y}^x \mid \mathbf{x}_M; \alpha) = \mathcal{N}(\mathbf{y}^x \mid \mathbf{x}_M, \alpha^{-1}\mathbf{I}) \tag{9}$$

$$p_S(\mathbf{y}^v \mid \mathbf{v}_M; \alpha') = \mathcal{N}\Big(\mathbf{y}^v \mid \alpha'(K\mathbf{e}_{\mathbf{v}_M} - \mathbf{1}), \alpha' K\mathbf{I}\Big) \tag{10}$$

where $\mathbf{e}_{\mathbf{v}_M} = [\mathbf{e}_{\mathbf{v}_M^{(1)}}, \ldots, \mathbf{e}_{\mathbf{v}_M^{(N_M)}}] \in \mathbb{R}^{N_M \times K}$, $\mathbf{e}_j \in \mathbb{R}^K$ is the projection from the class index j to the length-K one-hot vector, and K the number of atom types. Note that we could set different noise schedules for different modalities (α for coordinates and α' for types) for more efficient training of the joint noise prediction network.

Thereby for *receiver distribution* in Eq. 1,

$$p_R(\mathbf{y}^x \mid \boldsymbol{\theta}^x, \mathbf{p}; t) = \mathcal{N}(\mathbf{y}^x \mid \boldsymbol{\Phi}(\boldsymbol{\theta}^x, \mathbf{p}, t), \alpha^{-1}\mathbf{I}) \tag{11}$$

$$p_R(\mathbf{y}^v \mid \boldsymbol{\theta}^v, \mathbf{p}; t) = \Big[p_R\big((\mathbf{y}^v)^{(d)} \mid \cdot\big)\Big]_{d=1\ldots N}, \tag{12}$$

where $p_R\Big((\mathbf{y}^v)^{(d)} \mid \cdot\Big) = \sum_k p_O^v(k|\cdot) p_S^v\Big((\mathbf{y}^v)^{(d)} | k; \alpha\Big)$.

SE-(3) equivariance We introduce a fundamental inductive bias for SBDD to BFN, *i.e.* the density should be invariant to translation and rotation of protein-molecule complex (Satorras et al., 2021; Xu et al., 2021; Hoogeboom et al., 2022), in the following proposition (proof in Appendix B).

Proposition 4.1. *Denote the SE-(3) transformation as T_g, the likelihood is invariant* w.r.t. *T_g on the protein-ligand complex: $p_\phi(T_g(\mathbf{m}|\mathbf{p})) = p_\phi(\mathbf{m}|\mathbf{p})$ if we shift the Center of Mass (CoM) of protein atoms to zero and parameterize $\boldsymbol{\Phi}(\boldsymbol{\theta}, \mathbf{p}, t)$ with an SE-(3) equivariant network.*

4.2. Noise Reduced Sampling in Parameter Space

MolCRAFT addresses the high-variance discrete variable problem by maintaining a continuous probability mass function as beliefs of distributional parameters, which allows a smooth transformation towards the target distribution. This natural coherence with continuous coordinates gives us an advantage over continuous-discrete diffusion process.

During sampling, original BFN shifts the denoising process from sample space (recall diffusion $\mathbf{y}_{i-1} \to \mathbf{y}_i$) to parameter space $(\boldsymbol{\theta}_{i-1}, \mathbf{y}_i) \to \boldsymbol{\theta}_i$ via Bayesian update function h, where the information flows in this direction:

$$\boldsymbol{\theta}_{i-1} \xrightarrow{\boldsymbol{\Phi}} \hat{\mathbf{m}} \xrightarrow{p_S} \mathbf{y}_i \xrightarrow{p_U} \boldsymbol{\theta}_i, \tag{13}$$

where $p_U(\boldsymbol{\theta}_i \mid \boldsymbol{\theta}_{i-1}, \mathbf{m}, \mathbf{p}; \alpha_i)$ is defined in Eq. 3, and $\mathbf{m}$ is set to estimated $\hat{\mathbf{m}}$ drawn from p_O in Eq. 2.

It should be noted that the existing generative process of BFN, as well as that of diffusion models, performs continuous atom coordinates and discrete atom type sampling at each timestep. This risks introducing too much noise, and might end up generating incomplete molecules. To alleviate

such a problem, we design an empirically effective sampling strategy, which operates within the parameter space, and thus avoids introducing further noise from sampling discrete variables. The graphical description becomes:

$$\boldsymbol{\theta}_{i-1} \xrightarrow{\boldsymbol{\Phi}} \hat{\mathbf{m}} \xrightarrow{p_F} \boldsymbol{\theta}_i \tag{14}$$

Specifically, denoting $\gamma(t) \overset{\text{def}}{:=} \frac{\beta(t)}{1-\beta(t)}$, we update the parameter via Eq. 4, which simplifies to:

$$p_F(\boldsymbol{\mu} \mid \hat{\mathbf{x}}, \mathbf{p}; t) = \mathcal{N}\Big(\boldsymbol{\mu} \mid \gamma(t)\hat{\mathbf{x}}, \gamma(t)(1-\gamma(t))\mathbf{I}\Big) \tag{15}$$

$$p_F(\boldsymbol{\theta}^v \mid \hat{\mathbf{v}}, \mathbf{p}; t) = \mathop{\mathbb{E}}_{\mathcal{N}\left(\mathbf{y}^v \mid \beta(t)(K\mathbf{e}_{\hat{\mathbf{v}}}-\mathbf{1}), \beta(t)K\mathbf{I}\right)} \delta(\boldsymbol{\theta}^v - \text{softmax}(\mathbf{y}^v)) \tag{16}$$

We use the estimated $\hat{\mathbf{m}} = [\hat{\mathbf{x}}, \hat{\mathbf{v}}]$ (note that $\hat{\mathbf{v}}$ directly takes the continuous output categorical values without sampling) to directly update parameter for the next step, bypassing the sampling of noisy data needed for Bayesian update $\boldsymbol{\theta}_i = h(\boldsymbol{\theta}_{i-1}, \mathbf{y}, \alpha)$. The whole generative process happens in the parameter space except for the final step, which enjoys the advantage of lower variance and accelerates the overall generation path towards the complicated structure of molecules, with greatly improved sample quality at significantly fewer sampling steps, as shown in Fig. 7. Details of sampling are described in Algorithm 2.

5. Experiments

5.1. Experimental Setup

Dataset We use the CrossDocked dataset (Francoeur et al., 2020a) for training and testing, which originally contains 22.5 million protein-ligand pairs, and after the RMSD-based filtering and 30% sequence identity split by Luo et al. (2021), results in 100,000 training pairs and 100 test proteins. For each test protein, we sample 100 molecules for evaluation.

Baselines For autoregressive sampling-based models, we choose atom-based models AR (Luo et al., 2021), Pocket2Mol (Peng et al., 2022) and fragment-based model FLAG (Zhang et al., 2023). For diffusion-based models, we consider TargetDiff (Guan et al., 2022) and two variants of DecompDiff (Guan et al., 2023). Decomp-R uses the prior estimated from reference molecules in the test set, while Decomp-O selects the optimal prior from the reference prior and pocket prior, where the pocket prior center is predicted by AlphaSpace2 (Katigbak et al., 2020) and ligand atom number by a neural classifier.

Evaluation We conduct a comprehensive evaluation of SBDD models on all 100 proteins in test set, including:

- **Binding Affinity**. We employ AutoDock Vina (Trott & Olson, 2010) to measure binding affinity as it is a common practice (Luo et al., 2021; Peng et al., 2022; Guan et al., 2022; 2023), and report **Vina Score**, a direct score of generated pose, **Vina Min**, which scores the optimized pose after a local minimization of energy, and **Vina Dock**, the best possible score after re-docking, a global grid-based search optimization process. Therefore, it is highly favorable if Vina Score is close to Vina Min and Vina Dock, suggesting that the generated poses capture the 3D interaction well.
- **Conformation Stability**. We measure the stability for *ligand-only* and *binding complex* conformation. For *ligand-only*, we use the Jensen-Shannon divergence (**JSD**) between reference and generated distributions of bond length, bond angle and torsion angle at substructure level, and for a more global view, we employ **Strain Energy** to evaluate the rationality of generated ligand conformation. For *binding complex*, we adopt Steric Clashes (**Clash**) to detect possible clashes in protein-ligand complex, following Harris et al. (2023). We further propose to evaluate symmetry-corrected **RMSD** between the generated ligand atoms and Vina redocked poses as the metric of binding mode consistency, where poses with an RMSD below 2Å is generally regarded as chemically meaningful (Alhossary et al., 2015; Hassan et al., 2017; McNutt et al., 2021).
- **Drug-like Properties**. Drug-likeliness (**QED**), synthetic accessibility (**SA**), and diversity (**Div**) are adopted as molecular property metrics.
- **Overall**. To evaluate the overall quality of generated molecules, we calculate the **Binding Feasibility** as the ratio of molecules with reasonable affinity (Vina Score $<$ -2.49 kcal/mol) and stable conformation (Strain Energy $<$ 836 kcal/mol, RMSD $<$ 2Å) simultaneously, where the threshold values are set to the 95 percentile of the reference molecules. We also report **Success Rate** (Vina Dock $<$ -8.18, QED $>$ 0.25, SA $>$ 0.59) following Long et al. (2022) and Guan et al. (2022).
- **Sample Efficiency**. In order to make a practical comparison among non-autoregressive methods, we report the average **Time** and **Generation Success**, with the latter defined as the ratio of valid and complete molecules versus the intended number of samples.

5.2. Main Results

Our main findings are listed as below:

- MolCRAFT resembles and even surpasses the reference set in terms of binding affinity and overall feasibility, showing that we effectively learn the binding dynamics from protein-ligand complex distribution.

Table 2: Summary of different properties of reference and generated molecules under different sizes. (↑) / (↓) indicates larger / smaller is better. Top 2 results are highlighted with **bold text** and underlined text. *Note:* SE is short for Strain Energy, Div for Diversity, BF for Binding Feasibility, and SR for Success Rate. Baselines are either evaluated based on publicly available codebase (Decomp-R) or officially released samples (others).

Methods	Binding Affinity						Conformation Stability					Drug-like Properties			Overall		
							Ligand			Complex							
	Vina Score (↓)		Vina Min (↓)		Vina Dock (↓)		SE (↓)			Clash (↓)	RMSD (↑)	SA (↑)	QED (↑)	Div (↑)	BF (↑)	SR (↑)	Size
	Avg.	Med.	Avg.	Med.	Avg.	Med.	25%	50%	75%	Avg.	% < 2 Å	Avg.	Avg.	Avg.	(%)	(%)	Avg.
Reference	-6.36	-6.46	-6.71	-6.49	-7.45	-7.26	34	107	196	5.51	34.0	0.73	0.48	-	29.0	25.0	22.8
AR	-5.75	-5.64	-6.18	-5.88	-6.75	-6.62	259	595	2286	**4.49**	31.1	0.64	0.51	0.70	17.3	7.1	17.7
Pocket2Mol	-5.14	-4.70	**-6.42**	-5.82	**-7.15**	**-6.79**	102	189	374	6.24	30.8	**0.76**	**0.57**	0.69	24.6	**24.4**	17.7
Ours-small	**-5.96**	**-5.89**	-6.33	**-6.04**	-6.98	-6.63	**44**	**103**	**274**	4.88	**38.6**	0.74	0.52	**0.74**	**33.3**	17.4	17.8
FLAG[2]	16.48	4.53	1.21	-4.04	-5.63	-6.61	143	396	1164	40.83	8.2	**0.70**	0.49	0.70	3.8	14.1	21.5
TargetDiff	-5.47	-6.30	-6.64	-6.83	-7.80	-7.91	369	1243	13871	10.84	29.4	0.58	0.48	0.72	14.4	10.5	24.2
Decomp-R	-5.19	-5.27	-6.03	-6.00	-7.03	-7.16	115	421	1424	8.16	22.7	0.66	**0.51**	**0.73**	14.6	14.9	21.2
Ours	**-6.59**	**-7.04**	**-7.27**	**-7.26**	**-7.92**	**-8.01**	**83**	**195**	**510**	**7.09**	**41.8**	0.69	0.50	0.72	**33.8**	**26.8**	22.7
Decomp-O	-5.67	-6.04	-7.04	-7.09	-8.39	-8.43	379	983	4133	14.63	23.9	0.61	0.45	**0.68**	11.1	24.5	29.4
Ours-large	**-6.61**	**-8.14**	**-8.14**	**-8.42**	**-9.25**	**-9.20**	**171**	**333**	**1110**	**10.73**	**42.7**	**0.62**	**0.46**	0.61	**31.1**	**36.1**	29.4

- Non-autoregressive molecule generation could benefit from modeling in continuous parameter space, demonstrated by our performance in capturing diverse substructural modes and greatly improved conformation.
- Reliable evaluation of SBDD ought to take molecule sizes into account. To achieve fair comparison, controlled experiment regarding molecule size is needed.

Binding Affinity We report Vina metrics in Table 2.

I. Our model consistently outperforms other strong baselines in affinities, achieving a reference-level Vina Score of -6.59 kcal/mol. As Vina Score directly scores the pose and Vina Min only optimizes locally, they directly measure the generated pose quality. To the best of our knowledge, MolCRAFT is the first to achieve reference-level affinity scores without significant rearrangements via redocking, showing our superiority in learning binding interactions.

II. Vina Dock can potentially be hacked by generating larger molecules. Intuitively, larger molecules have more chances of forming interactions with protein surfaces. With the largest molecule sizes, Decomp-O achieves the second-best Vina Dock (-8.39 kcal/mol), far better than reference molecules. Further investigation reveals that Decomp-O gains an advantage by producing considerably larger out-of-distribution (OOD) molecules and thereby brings up the highest possible affinity post-docking. For a fair comparison, we report variants of DecompDiff and MolCRAFT stratified by size, and with the same number of atoms as Decomp-O, our model consistently achieves SOTA affinities, underscoring its robustness across different molecular sizes.

Conformation Stability We report the *substructural* level's average Jensen-Shannon divergence (JSD) between reference and generated bond length, angle and torsion angle distributions in Table 3 (detailed results for different bond/angle/torsion types in Appendix D). At the *global structure* level, we report strain energy for ligand-only conformational stability, and measure clashes in the binding complex, together with RMSD between generated and redocked poses in Table 2.

Table 3: Summary of molecular conformation results. (↓) indicates smaller is better. Top 2 results are highlighted with **bold text** and underlined text. *Note:* JSD is calculated between distributions estimated from generated and reference molecules, we report the mean of all JSD values here.

Methods	Length (↓) Avg. JSD	Angle (↓) Avg. JSD	Torsion (↓) Avg. JSD
AR	0.554	0.507	0.552
Pocket2Mol	0.485	0.482	0.459
FLAG	0.511	0.406	**0.270**
TargetDiff	0.382	0.435	0.400
Decomp-O	0.359	0.414	0.358
Decomp-R	0.348	0.412	0.317
Ours	**0.319**	**0.379**	0.300

I. Our model excels in modeling diverse local modes, and ranks first in bond length and angle distributions. Moreover, Fig. 2 shows MolCRAFT is the only model that captures two distinct modes for multi-modal C-C, C-N and C-O bond, justifying our choice of modeling in the joint continuous parameter space. More results are in Fig. 8, 9 and 10.

II. Injecting substructural inductive bias helps to capture more modes. Fragment-based model FLAG displays the best torsion angle distribution, and prior-enhanced DecompDiff also exhibits relatively competitive performances in modeling molecular geometries, whereas other autoregressive models collapse into certain modes as in Fig. 2.

III. For ligand-only stability, we greatly improve upon

[2]Though relatively worse than reported in FLAG paper, the evaluation is based on released samples and confirmed by authors.

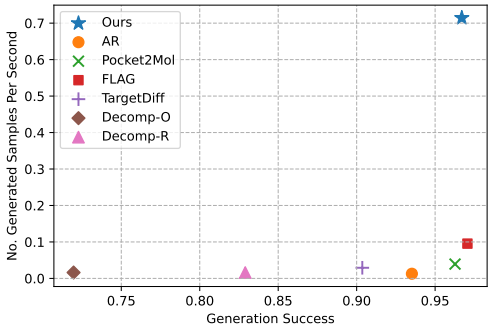


Figure 5: Sample efficiency, where Generation Success means the generated molecules are both valid and complete.

the strained conformations, even surpassing autoregressive methods. According to Table 2, our model is at least an order of magnitude better than diffusion-based counterparts, and is close to reference. While autoregressive methods generally display better strain energy, MolCRAFT still achieves superior performance under comparable molecule sizes.

IV. Our binding complex contains fewer clashes and remains consistent after redocking. We achieve few steric clashes, and has the best RMSD performance, which means 46% of our molecules already resemble accurate docking pose even without force field optimization or redocking, rendering it reliable for generating molecules in 3D space. The reason why we achieve even better RMSD than reference can be explained by a distribution shift. In dataset construction[3], the training set contains 52.4% docked molecules, while the test set only contains 37.0% docked ones, which aligns with the fact that for reference there are only 34.0% with RMSD < 2Å. This accounts for why MolCRAFT has more consistent and high-affinity binders, which effectively captures the training set distribution and learns the binding dynamics.

Overall We report the overall feasible rate and success rate in Table 2. MolCRAFT achieves the best among all, demonstrating our competency in generating molecules with high affinity and stable conformation. Our method captures the interatomic interactions in 3D space, and proposes desirable molecules without relying on post-fixed docking poses. This further validates our choice of learning in the continuous parameter space.

Sampling performance We compare the generation speed (average time for generating 100 samples) and generation success in Figure 5. We achieve SOTA sampling performance in both dimensions, generating more complete (96.7%) molecules at 30× **speedup**. While it takes on average 3428s and 6189s for TargetDiff and DecompDiff to generate 100 samples respectively, our model only uses 141s, thanks to our improved sampling strategy (see Sec. 5.3).

[3]There are two kinds of 3D ligand poses in the dataset, i.e. Vina minimized poses in the given receptor, and Vina docked poses. https://github.com/gnina/models/tree/master/data/CrossDocked2020

5.3. Ablation Study of Sampling Strategy

Considering that we propose the first-of-its-kind SBDD model that operates in the fully continuous parameter space, and present a noise-reduced sampling approach adapted to the space, we conduct ablation study that validates our design, showing a performance boost from Vina Score/Min of -5.42/-6.30 kcal/mol to -6.51/-7.13 kcal/mol.

We test different sampling strategies with different steps for the same checkpoint, and sample 10 molecules each for 100 test proteins. We plot the curves of QED, SA, Completeness (↑) and Vina Score (↓) in Figure 7, Appendix D.2. As the sampling step increases to training steps, we found the original sampling strategy exhibits first enhanced then slightly decreased sample quality, possibly because the update of parameters is smoothed or oversmoothed by finer partitioned noise factor α, whereas the noise reduced strategy displays this tendency far earlier and generates the best quality of molecules with fewer sampling steps, indicating its high efficiency. Considering the overall sample quality, we decide to use 100 sampling steps for our model, which is 10× faster than sampling at original 1000 training steps.

6. Related Work

Target-Aware Molecule Generation Trained on protein-ligand complex data, target-aware methods directly model the interaction between protein pockets and ligands. Early attempts are based on 1D SMILES or 2D molecular graph generation (Bjerrum & Threlfall, 2017; Gómez-Bombarelli et al., 2018; Segler et al., 2018) and fail to consider spatial information. Recent works focus on 3D molecule generation, and there are mainly two fashions: (1) *Autoregressive methods*. For atom-based methods, LiGAN (Masuda et al., 2020) and AR (Luo et al., 2021) adopt an atomic density grid view of molecules, the former predicting a voxelized density grid and performing optimization to reconstruct atom types and coordinates, the latter assigning atomic probability to each voxel and utilizes MCMC to generate atom-by-atom. GraphBP (Liu et al., 2022) uses normalizing flow and encodes the context to preserve 3D geometric equivariance, and Pocket2Mol (Peng et al., 2022) further adds bond generation for more realistic molecular structure. For fragment-based methods (Powers et al., 2022; Zhang & Liu, 2023; Zhang et al., 2023), molecules are decomposed into chemically meaningful motifs rather than seperated atom point cloud, and generated via motif assembling. (2) *Diffusion-based methods* have recently been proposed, aiming to overcome the problem of sampling ef-

ficiency and unnatural ordering brought by autoregressive fashion (Schneuing et al., 2022; Guan et al., 2022; 2023). But these methods still suffer from false positive problems.

7. Conclusion

In this paper, we first investigate the challenges of current generative models in SBDD, *i.e.*, distorted structures and sub-optimal binding modes. Based on the observations concerning mode collapse and hybrid space, we propose MolCRAFT, an SE-(3) equivariant generative model operating in the continuous parameter space with a noise reduced sampling strategy, which yields higher quality molecules.

Acknowledgements

This work is supported by the National Science and Technology Major Project (2022ZD0117502), Natural Science Foundation of China (62376133) and Guoqiang Research Institute General Project, Tsinghua University (No. 2021GQG1012). Jiawei Han's work is supported by U.S. National Science Foundation IIS-19-56151, and the Molecule Maker Lab Institute: An AI Research Institutes program supported by U.S. NSF under Award No. 2019897. The authors would like to thank Xiangyu Li for his valuable comments on this work.

Impact Statement

This paper is aimed to facilitate in-silico rational drug design. Potential society consequences include mal-intended usage of toxic compound discovery, which needs support from professional wet labs and thus expensive to reach. Therefore we do not possess a negative vision that this might lead to serious ethical consequences, though we are aware of such a possibility.

References

Alhossary, A., Handoko, S. D., Mu, Y., and Kwoh, C.-K. Fast, accurate, and reliable molecular docking with quickvina 2. *Bioinformatics*, 31(13):2214–2216, 2015.

Ba, J. L., Kiros, J. R., and Hinton, G. E. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

Bjerrum, E. J. and Threlfall, R. Molecular generation with recurrent neural networks (rnns). *arXiv preprint arXiv:1705.04612*, 2017.

Francoeur, P. G., Masuda, T., Sunseri, J., Jia, A., Iovanisci, R. B., Snyder, I., and Koes, D. R. 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, 2020a. doi: 10.1021/acs.jcim.0c00411. URL https://doi.org/10.1021/acs.jcim.0c00411. PMID: 32865404.

Francoeur, P. G., Masuda, T., Sunseri, J., Jia, A., Iovanisci, R. B., Snyder, I., and Koes, D. R. 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, 2020b.

Gómez-Bombarelli, R., Wei, J. N., Duvenaud, D., Hernández-Lobato, J. M., Sánchez-Lengeling, B., Sheberla, D., Aguilera-Iparraguirre, J., Hirzel, T. D., Adams, R. P., and Aspuru-Guzik, A. Automatic chemical design using a data-driven continuous representation of molecules. *ACS central science*, 4(2):268–276, 2018.

Graves, A., Srivastava, R. K., Atkinson, T., and Gomez, F. Bayesian flow networks. *arXiv preprint arXiv:2308.07037*, 2023.

Guan, J., Qian, W. W., Ma, W.-Y., Ma, J., and Peng, J. Energy-inspired molecular conformation optimization. In *international conference on learning representations*, 2021.

Guan, J., Qian, W. W., Peng, X., Su, Y., Peng, J., and Ma, J. 3d equivariant diffusion for target-aware molecule generation and affinity prediction. In *The Eleventh International Conference on Learning Representations*, 2022.

Guan, J., Zhou, X., Yang, Y., Bao, Y., Peng, J., Ma, J., Liu, Q., Wang, L., and Gu, Q. DecompDiff: Diffusion models with decomposed priors for structure-based drug design. In Krause, A., Brunskill, E., Cho, K., Engelhardt, B., Sabato, S., and Scarlett, J. (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 11827–11846. PMLR, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/guan23a.html.

Harris, C., Didi, K., Jamasb, A. R., Joshi, C. K., Mathis, S. V., Lio, P., and Blundell, T. Benchmarking generated poses: How rational is structure-based drug design with generative models? *arXiv preprint arXiv:2308.07413*, 2023.

Hassan, N. M., Alhossary, A. A., Mu, Y., and Kwoh, C.-K. Protein-ligand blind docking using quickvina-w with inter-process spatio-temporal integration. *Scientific reports*, 7(1):15451, 2017.

Ho, J., Jain, A., and Abbeel, P. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Hoogeboom, E., Satorras, V. G., Vignac, C., and Welling, M. Equivariant diffusion for molecule generation in 3d. In *International conference on machine learning*, pp. 8867–8887. PMLR, 2022.

Isert, C., Atz, K., and Schneider, G. Structure-based drug design with geometric deep learning. *Current Opinion in Structural Biology*, 79:102548, April 2023. ISSN 0959440X. doi: 10.1016/j.sbi.2023.102548. URL https://linkinghub.elsevier.com/retrieve/pii/S0959440X23000222.

Jiang, Y., Zhang, G., You, J., Zhang, H., Yao, R., Xie, H., Zhang, L., Xia, Z., Dai, M., Wu, Y., et al. Pocketflow is a data-and-knowledge-driven structure-based molecular generative model. *Nature Machine Intelligence*, pp. 1–12, 2024.

Katigbak, J., Li, H., Rooklin, D., and Zhang, Y. Alphaspace 2.0: Representing concave biomolecular surfaces using beta-clusters. *Journal of Chemical Information and Modeling*, 60(3):1494–1508, 2020. doi: 10.1021/acs.jcim.9b00652. URL https://doi.org/10.1021/acs.jcim.9b00652. PMID: 31995373.

Liu, M., Luo, Y., Uchino, K., Maruhashi, K., and Ji, S. Generating 3D Molecules for Target Protein Binding, May 2022. URL http://arxiv.org/abs/2204.09410. arXiv:2204.09410 [cs, q-bio].

Long, S., Zhou, Y., Dai, X., and Zhou, H. Zero-shot 3d drug design by sketching and generating. In *NeurIPS*, 2022.

Luo, S., Guan, J., Ma, J., and Peng, J. A 3D Generative Model for Structure-Based Drug Design. *Advances in Neural Information Processing Systems*, 34: 6229–6239, 2021. URL http://arxiv.org/abs/2203.10446.

Masuda, T., Ragoza, M., and Koes, D. R. Generating 3D Molecular Structures Conditional on a Receptor Binding Site with Deep Generative Models, November 2020. URL http://arxiv.org/abs/2010.14442. arXiv:2010.14442 [physics, q-bio].

McNutt, A. T., Francoeur, P., Aggarwal, R., Masuda, T., Meli, R., Ragoza, M., Sunseri, J., and Koes, D. R. Gnina 1.0: molecular docking with deep learning. *Journal of cheminformatics*, 13(1):1–20, 2021.

Peng, X., Luo, S., Guan, J., Xie, Q., Peng, J., and Ma, J. Pocket2Mol: Efficient molecular sampling based on 3D protein pockets. In Chaudhuri, K., Jegelka, S., Song, L., Szepesvari, C., Niu, G., and Sabato, S. (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 17644–17655. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/peng22b.html.

Peng, X., Guan, J., Liu, Q., and Ma, J. Moldiff: Addressing the atom-bond inconsistency problem in 3d molecule diffusion generation. In *International Conference on Machine Learning*, pp. 27611–27629. PMLR, 2023.

Powers, A. S., Yu, H. H., Suriana, P. A., and Dror, R. O. Fragment-based ligand generation guided by geometric deep learning on protein-ligand structures. In *ICLR2022 Machine Learning for Drug Discovery*, 2022. URL https://openreview.net/forum?id=192L9cr-8HU.

Satorras, V. G., Hoogeboom, E., and Welling, M. E (n) equivariant graph neural networks. In *International conference on machine learning*, pp. 9323–9332. PMLR, 2021.

Schneuing, A., Du, Y., Harris, C., Jamasb, A., Igashov, I., Du, W., Blundell, T., Lió, P., Gomes, C., Welling, M., Bronstein, M., and Correia, B. Structure-based Drug Design with Equivariant Diffusion Models, October 2022. URL http://arxiv.org/abs/2210.13695. arXiv:2210.13695 [cs, q-bio].

Segler, M. H., Kogej, T., Tyrchan, C., and Waller, M. P. Generating focused molecule libraries for drug discovery with recurrent neural networks. *ACS central science*, 4 (1):120–131, 2018.

Song, Y., Gong, J., Qu, Y., Zhou, H., Zheng, M., Liu, J., and Ma, W.-Y. Unified generative modeling of 3d molecules via bayesian flow networks. *arXiv preprint arXiv:2403.15441*, 2024a.

Song, Y., Gong, J., Xu, M., Cao, Z., Lan, Y., Ermon, S., Zhou, H., and Ma, W.-Y. Equivariant flow matching with hybrid probability transport for 3d molecule generation. *Advances in Neural Information Processing Systems*, 36, 2024b.

Tian, S., Wang, J., Li, Y., Li, D., Xu, L., and Hou, T. The application of in silico drug-likeness predictions in pharmaceutical research. *Advanced drug delivery reviews*, 86: 2–10, 2015.

Trott, O. and Olson, A. J. Autodock vina: Improving the speed and accuracy of docking with a new scoring function, efficient optimization, and multithreading. *Journal of Computational Chemistry*, 31 (2):455–461, 2010. doi: https://doi.org/10.1002/jcc.21334. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/jcc.21334.

Ursu, O., Rayan, A., Goldblum, A., and Oprea, T. I. Understanding drug-likeness. *Wiley Interdisciplinary Reviews: Computational Molecular Science*, 1(5):760–781, 2011.

Walters, W. P. Virtual chemical libraries. *Journal of Medicinal Chemistry*, 62(3):1116–1124, 2019. doi: 10.1021/acs.jmedchem.8b01048. URL https://doi.org/10.1021/acs.jmedchem.8b01048. PMID: 30148631.

Wang, M., Wang, Z., Sun, H., Wang, J., Shen, C., Weng, G., Chai, X., Li, H., Cao, D., and Hou, T. Deep learning approaches for de novo drug design: An overview. *Current Opinion in Structural Biology*, 72:135–144, February 2022. ISSN 0959440X. doi: 10.1016/j.sbi.2021.10.001. URL https://linkinghub.elsevier.com/retrieve/pii/S0959440X21001433.

Xu, M., Yu, L., Song, Y., Shi, C., Ermon, S., and Tang, J. Geodiff: A geometric diffusion model for molecular conformation generation. In *International Conference on Learning Representations*, 2021.

Zhang, Z. and Liu, Q. Learning Subpocket Prototypes for Generalizable Structure-based Drug Design, May 2023. URL http://arxiv.org/abs/2305.13997. arXiv:2305.13997 [cs, q-bio].

Zhang, Z., Min, Y., Zheng, S., and Liu, Q. Molecule Generation For Target Protein Binding with Structural Motifs. 2023.

A. Detailed Formulation of BFN

Introducing parameters to diffusion process Classic diffusion process consists of a *forward process* which gradually applies noise to the data $\mathbf{y}_i \sim p(\mathbf{y}_i|\mathbf{m};\alpha_i)$ till finally standard Gaussian $\mathbf{y}_0 \sim \mathcal{N}(\mathbf{0},\mathbf{I})$, and a *reverse process* which starts from Gaussian noise $\mathbf{y}_0$ and iteratively denoises $\mathbf{y}_i \sim p(\mathbf{y}_i|\mathbf{y}_{i-1},\mathbf{p};t)$ to produce a sample $\mathbf{m} \sim p(\mathbf{m}|\mathbf{y}_n,\mathbf{p};n)$. Thus the key designs of diffusion are about how to apply noise given $\mathbf{m}$, and how to denoise with $(\mathbf{y},\mathbf{p})$.

The *Variational Lower Bound* for diffusion models (Ho et al., 2020) is:

$$-\log p_\theta(\mathbf{m}|\mathbf{p}) \leq \mathcal{L}_{\text{VLB}} = D_{\text{KL}}\Big(q(\mathbf{y}_{0\ldots n}|\mathbf{m},\mathbf{p})\|p_\phi(\mathbf{y}_{0\ldots n}|\mathbf{p})\Big) \tag{17}$$

Through adding noise to $\mathbf{m}$ and denoising from $\mathbf{y}_0$ and $\mathbf{p}$, diffusion transports a simple prior distribution $p(\mathbf{y}_0)$ to the desired data distribution $p(\mathbf{m}|\mathbf{p})$, and generates target-aware molecules in a non-autoregressive fashion. To transfer the generative process from sample space to parameter space, latent variables $\boldsymbol{\theta}_{0\ldots n}$ are further introduced to characterize the distribution of $\mathbf{y}_{1\ldots n}$ (Graves et al., 2023):

$$\mathcal{L}_{\text{VLB}} = D_{\text{KL}}\Big(q(\mathbf{y}_{1\ldots n},\boldsymbol{\theta}_{0\ldots n}|\mathbf{m},\mathbf{p})\|p_\phi(\mathbf{y}_{1\ldots n},\boldsymbol{\theta}_{0\ldots n}|\mathbf{p})\Big) = n \mathop{\mathbb{E}}_{i\sim U(1,n)} \mathop{\mathbb{E}}_{\mathbf{y}_i,\boldsymbol{\theta}_{i-1}\sim q} D_{\text{KL}}\Big(q(\mathbf{y}_i|\mathbf{m})\|p_\phi(\mathbf{y}_i|\boldsymbol{\theta}_{i-1},\mathbf{p})\Big). \tag{18}$$

With Eq. 18 pulling close p_ϕ to q, p_ϕ is finally able to alternatively generates $\mathbf{y}_i$ and $\boldsymbol{\theta}_i$, *i.e.*, a generative process driven from the parameter space. Here for BFN, the generative process is characterized by *receiver distribution* (Eq. 1) with the help of *output distribution* (Eq. 2), and the noise adding process by *sender distribution*:

$$q(\mathbf{y}_i \mid \mathbf{m}) = p_S(\mathbf{y}_i \mid \mathbf{m};\alpha_i) \tag{19}$$

Training objective BFN can be trained by minimizing the KL-divergence between noisy sample distributions. BFN allows training in discrete time and continuous time, and for efficiency we adopt the n-step discrete loss. Since the atom coordinates and the noise are Gaussian, the loss can be written analytically as follows:

$$\begin{aligned} L_x^n &= D_{\text{KL}}\Big(\mathcal{N}(\mathbf{x},\alpha_i^{-1}I) \parallel \mathcal{N}(\hat{\mathbf{x}}(\boldsymbol{\theta}_{i-1},\mathbf{p},t),\alpha_i^{-1}I)\Big) \\ &= \frac{\alpha_i}{2}\Big\|\mathbf{x}-\hat{\mathbf{x}}(\boldsymbol{\theta}_{i-1},\mathbf{p},t)\Big\|^2 \end{aligned} \tag{20}$$

Similarly, atom type loss can also be derived by taking KL-divergence between Gaussians, yielding:

$$\begin{aligned} L_v^n = \ln\mathcal{N}\Big(\mathbf{y}^v \mid \alpha_i(K\mathbf{e}_\mathbf{v}-\mathbf{1}),\alpha_i KI\Big) - \\ \sum_{d=1}^{N_M}\ln\Big(\sum_{k=1}^{K} p_O(k|\boldsymbol{\theta};t)\mathcal{N}\big(\cdot^{(d)}|\alpha_i(K\mathbf{e}_k-\mathbf{1}),\alpha_i KI\big)\Big) \end{aligned} \tag{21}$$

Algorithm 1 Discrete-Time Loss

Require: $\mathbf{x}_M \in \mathbb{R}^{3N_M}, \mathbf{v}_M \in \mathbb{R}^{N_M K}, \mathbf{p} \in \mathbb{R}^{N_P(3+D_P)}, \sigma_1, \beta_1 \in \mathbb{R}^+$

$i \sim U(1,n), t \leftarrow \frac{i-1}{n}$
$\boldsymbol{\mu} \sim p_F^x(\boldsymbol{\mu}|\mathbf{x}_M,\mathbf{p};t,\beta(t)=\sigma_1^{-2t}-1)$
$\boldsymbol{\theta}^v \sim p_F^v(\boldsymbol{\theta}^v|\mathbf{v}_M,\mathbf{p};t,\beta(t)=t^2\beta_1)$
$\hat{\mathbf{x}},\hat{\mathbf{v}} \leftarrow p_O(\boldsymbol{\mu},\boldsymbol{\theta}^v,\mathbf{p},t)$
$L_x^n \leftarrow \frac{(1-\sigma_1^{2/n})}{2\sigma_1^{2i/n}}\|\mathbf{x}_M-\hat{\mathbf{x}}\|^2$
$\alpha \leftarrow \beta_1(\frac{2i-1}{n^2})$
$\mathbf{y}^v \sim p_S^v(\mathbf{y}^v|\mathbf{v}_M,\alpha))$
$L_v^n \leftarrow \ln p_S^v(\mathbf{y}^v|\mathbf{v}_M;\alpha) - \ln p_R^v(\mathbf{y}^v|\hat{\mathbf{v}};\alpha,t)$
return $L^n(\mathbf{m},\mathbf{p}) = L_x^n + L_v^n$

Algorithm 2 Sampling

function update($\hat{\mathbf{x}} \in \mathbb{R}^{3N}, \hat{\mathbf{v}} \in \mathbb{R}^{NK}, \beta(t), \beta'(t), t \in \mathbb{R}^+$)
  $\gamma \leftarrow \frac{\beta(t)}{1-\beta(t)}$
  $\boldsymbol{\mu} \sim \mathcal{N}(\gamma\hat{\mathbf{x}}, \gamma(1-\gamma)\mathbf{I})$
  $\mathbf{y}^v \sim \mathcal{N}(\mathbf{y}^v | \beta'(t)(K\mathbf{e}_{\hat{\mathbf{v}}} - \mathbf{1}), \beta'(t)K\mathbf{I})$
  $\boldsymbol{\theta}^v \leftarrow [\text{softmax}((\mathbf{y}^v)^{(d)})]_{d=1\dots N_M}$
  return $\boldsymbol{\mu}, \boldsymbol{\theta}^v$
end function
Require: Network $\boldsymbol{\Phi}$, $\mathbf{p} \in \mathbb{R}^{N_P(3+D_P)}, N, n_M, K \in \mathbb{N}^+, \sigma_1, \beta_1 \in \mathbb{R}^+$
$\boldsymbol{\mu} \leftarrow \mathbf{0}, \rho \leftarrow 1, \boldsymbol{\theta}^v \leftarrow [\frac{1}{K}]_{n_M \times K}$
for $i = 1$ to N **do**
  $t \leftarrow \frac{i-1}{n}$
  $\hat{\mathbf{x}}, \hat{\mathbf{v}} \leftarrow p_O(\boldsymbol{\mu}, \boldsymbol{\theta}^v, \mathbf{p}, t)$
  $\boldsymbol{\mu}, \boldsymbol{\theta}^v \leftarrow$ update($\hat{\mathbf{x}}, \hat{\mathbf{v}}, \sigma_1, \beta_1, t$)
end for
$\hat{\mathbf{x}}, p_O^v(\hat{\mathbf{v}}|\boldsymbol{\theta}^v, \mathbf{p}; 1) \leftarrow p_O(\boldsymbol{\mu}, \boldsymbol{\theta}^v, \mathbf{p}, 1)$
$\hat{\mathbf{v}} \sim p_O^v(\hat{\mathbf{v}}|\boldsymbol{\theta}^v, \mathbf{p}; 1)$
return $[\hat{\mathbf{x}}, \hat{\mathbf{v}}]$

B. Proof of SE-(3) Invariant Objective and SE-(3) Equivariant Sampling Process

Density estimation and distribution learning on the 3D molecules should take translational and rotational invariance of the protein-ligand complex into consideration, *a.k.a*, the Special Euclidean group (SE-(3)) in 3D space (Satorras et al., 2021; Xu et al., 2021; Hoogeboom et al., 2022). Denote T_g as the group of SE-(3) transformation, $T_g(\mathbf{x}) = \mathbf{R}\mathbf{x} + \mathbf{b}$, where $\mathbf{R} \in \mathbb{R}^{3\times 3}$ is the rotation matrix, and $\mathbf{b} \in \mathbb{R}^3$ is the translation vector.

Following Guan et al. (2022), we move the center of protein to zero, *i.e.*, $\tilde{\mathbf{p}} = [\tilde{\mathbf{x}}_P, \mathbf{v}_P], \tilde{\mathbf{x}}_P = \mathbf{Q}\mathbf{x}_P$, $\mathbf{Q} = \mathbf{I}_3 \otimes (\mathbf{I}_{N_P} - \frac{1}{N_P}\mathbf{1}_{N_P}\mathbf{1}_{N_P}^\top)$ and shift molecule $\mathbf{m}$ and variable $\boldsymbol{\mu}$ the same way. In another word, $\tilde{\mathbf{p}}, \tilde{\mathbf{m}}, \tilde{\boldsymbol{\mu}}$ are only defined with zero gravity of $\mathbf{p}$, namely zero Center-of-Mass (CoM) space. That is, for any T_g applied to the protein and molecule/variable complex, we always have $T_g(\tilde{\mathbf{m}}, \tilde{\mathbf{p}}) = \mathbf{R}(\tilde{\mathbf{m}}, \tilde{\mathbf{p}}), T_g(\tilde{\boldsymbol{\mu}}, \tilde{\mathbf{p}}) = \mathbf{R}(\tilde{\boldsymbol{\mu}}, \tilde{\mathbf{p}})$. Thus translational invariance is naturally satisfied on $\tilde{\mathbf{p}}, \tilde{\mathbf{m}}, \tilde{\boldsymbol{\mu}}$ by definition. For convenience, in the following discussion, we mention $\tilde{\mathbf{p}}, \tilde{\mathbf{m}}, \tilde{\boldsymbol{\mu}}$ as $\mathbf{p}, \mathbf{m}, \boldsymbol{\mu}$.

Then we slightly rewrite the key steps in Algorithm 1, and for convenience we omit $\boldsymbol{\theta}^v$, $\mathbf{v}_M$, $\mathbf{v}_P$ and t since these variables are not in the 3D space.

$$\begin{aligned}
\boldsymbol{\mu} &= \gamma\mathbf{x}_M + \gamma(1-\gamma)\boldsymbol{\epsilon} \\
\hat{\mathbf{x}} &= \boldsymbol{\Phi}(\boldsymbol{\mu}, \mathbf{x}_P) \\
L_x^n(\mathbf{x}_M, \mathbf{x}_P) &= \text{const}\|\mathbf{x}_M - \hat{\mathbf{x}}\|^2
\end{aligned}$$

Since ϵ is sampled from isotropic Gaussian, $\epsilon = \mathbf{R}\epsilon'$ is from the same distribution. If we apply T_g to the $\boldsymbol{\mu}, \mathbf{x}_P$, since $\boldsymbol{\Phi}(\boldsymbol{\mu}, \mathbf{x}_p)$ is an SE-(3) equivariant graph network, the output of network $\boldsymbol{\Phi}$ will be

$$\boldsymbol{\Phi}(T_g(\boldsymbol{\mu}, \mathbf{x}_P)) = \boldsymbol{\Phi}(\mathbf{R}(\boldsymbol{\mu}, \mathbf{x}_P)) = \mathbf{R}(\boldsymbol{\Phi}(\boldsymbol{\mu}, \mathbf{x}_P)) = \mathbf{R}(\hat{\mathbf{x}}) = T_g(\hat{\mathbf{x}}) \tag{22}$$

Thus the loss of T_g-transformed complex will be

$$\begin{aligned}
\|T_g(\mathbf{x}_M) - T_g(\hat{\mathbf{x}})\|^2 &= \|\mathbf{R}\mathbf{x}_M + \mathbf{b} - \mathbf{R}\hat{\mathbf{x}} - \mathbf{b}\|^2 = \|\mathbf{R}\mathbf{x}_M - \mathbf{R}\hat{\mathbf{x}}\|^2 \\
&= (\mathbf{x}_M - \hat{\mathbf{x}})^\top \mathbf{R}^\top \mathbf{R}(\mathbf{x}_M - \hat{\mathbf{x}}) = (\mathbf{x}_M - \hat{\mathbf{x}})^\top (\mathbf{x}_M - \hat{\mathbf{x}}) = \|\mathbf{x}_M - \hat{\mathbf{x}}\|^2
\end{aligned} \tag{23}$$

Thus the objective is SE-(3) invariant to $\mathbf{p}, \mathbf{m}, \boldsymbol{\mu}$.

Before discuss the sampling process in Algorithm 2, we slightly rewrite the key steps, and for convenience, we omit

$\boldsymbol{\theta}^v, \mathbf{v}_M, \mathbf{v}_P$ and t since these variables are not in the 3D space.

$$\begin{aligned}
\boldsymbol{\mu}_0 &= \mathbf{0}, \rho_0 = 1 \\
\mathbf{x}_i &= \boldsymbol{\Phi}(\boldsymbol{\mu}_{i-1}, \mathbf{x}_P) \\
\mathbf{y}_i^x &= \mathbf{x}_i + \alpha_i \boldsymbol{\epsilon} \\
\rho_i &= \rho_{i-1} + \alpha_i \\
\boldsymbol{\mu}_i &= \frac{\rho_{i-1}\boldsymbol{\mu}_{i-1} + \alpha_i \mathbf{y}_i^x}{\rho_i}.
\end{aligned}$$

At step 1, since $\boldsymbol{\Phi}(\boldsymbol{\mu}, \mathbf{x}_P)$ is an SE-(3) equivariant graph network,

$$T_g(\mathbf{x}_1) = T_g(\boldsymbol{\Phi}(\boldsymbol{\mu}_0, \mathbf{x}_P)) = \boldsymbol{\Phi}(T_g(\boldsymbol{\mu}_0, \mathbf{x}_P)) = \boldsymbol{\Phi}(\boldsymbol{\mu}_0, T_g(\mathbf{x}_P)), \tag{24}$$

thus $\mathbf{x}_1$ is SE-(3) Equivariant to $\boldsymbol{\mu}_0, \mathbf{x}_P$.

For every step i, if we assume $T_g(\mathbf{x}_i) = T_g(\boldsymbol{\Phi}(\boldsymbol{\mu}_{i-1}, \mathbf{x}_P)) = \boldsymbol{\Phi}(T_g(\boldsymbol{\mu}_{i-1}, \mathbf{x}_P))$, *i.e.*, $\mathbf{x}_i$ is SE-(3) equivariant to $\boldsymbol{\mu}_{i-1}, \mathbf{x}_P$, we have (1) $\mathbf{y}_i^x$ is SE-(3) equivariant to $\mathbf{x}_i$ since this update is simple addition and $\boldsymbol{\epsilon}$ from isotropic Gaussian, (2) $\boldsymbol{\mu}_i$ is SE-(3) equivariant to $\boldsymbol{\mu}_{i-1}, \mathbf{y}_i^x$, thus $\boldsymbol{\mu}_i$ is SE-(3) equivariant to $\mathbf{x}_i, \mathbf{x}_P$. And $\mathbf{x}_{i+1}$ is is SE-(3) equivariant to $\boldsymbol{\mu}_i, \mathbf{x}_P$, thus $\mathbf{x}_{i+1}$ is SE-(3) equivariant to $\mathbf{x}_i, \mathbf{x}_P$.

With mathematical induction, we have $\mathbf{x}_i$ SE-(3) equivariant to $\boldsymbol{\mu}_0, \mathbf{x}_P$, thus the final sample $\mathbf{x}_N$ is also SE-(3) equivariant to $\boldsymbol{\mu}_0, \mathbf{x}_P$. The sampling process in Algorithm 2 is SE-(3) equivariant to $\mathbf{x}_P$.

C. Implementation Details

C.1. Parameterization with SE-(3) Equivariant Network

We model the interaction between ligand molecule atoms and protein pocket atoms with an SE-(3) equivariant network, PosNet3D (Guan et al., 2021), as our backbone $\boldsymbol{\Phi}$ in Eq. 2.

A protein-molecule graph is firstly constructed through k-nearest neighbor search of the atom coordinates, $G = \langle V, E \rangle$. For each layer, the atom hidden states $\mathbf{h}^l$ and coordinates $\mathbf{x}^l$ are updated alternately as follows:

$$\mathbf{h}_i^{l+1} = \mathbf{h}_i^l + \sum_{j \in N_G(i)} \phi_h\Big(d_{ij}, \mathbf{h}_i^l, \mathbf{h}_j^l, \mathbf{e}_{ij}, t\Big) \tag{25}$$

$$\Delta \mathbf{x}_i = \sum_{j \in N_G(i)} \Big(\mathbf{x}_j^l - \mathbf{x}_i^l\Big) \phi_x\Big(d_{ij}, \mathbf{h}_i^{l+1}, \mathbf{h}_j^{l+1}, \mathbf{e}_{ij}, t\Big)$$

$$\mathbf{x}_i^{l+1} = \mathbf{x}_i^l + \Delta \mathbf{x}_i \cdot 1_{mol} \tag{26}$$

where $N_G(i)$ denotes the neighborhood of atom i in G, $(\mathbf{h}_i, \mathbf{x}_i)$ and $(\mathbf{h}_j, \mathbf{x}_j)$ denote atom i and j, d_{ij}^l is the euclidean distance between atoms i and j, $\mathbf{e}_{ij}$ indicates the connection is between protein atoms, ligand atoms, or protein atom and ligand atom, 1_{mol} is to indicate only ligand atoms are updated, ϕ_h and ϕ_x are attention blocks which take $\mathbf{h}_i^l$ as query and $[\mathbf{h}_i^l, \mathbf{h}_j^l, \mathbf{e}_{ij}]$ as keys and values.

For the first layer, $\mathbf{x}^0 = [\boldsymbol{\mu}, \mathbf{x}_P]$, $\mathbf{h}^0 = \text{linear}(\boldsymbol{\theta}^v, \mathbf{v}_P, t)$. For the last layer, $\boldsymbol{\Phi}$ directly outputs an estimation $\hat{\mathbf{x}} = \boldsymbol{\Phi}^x$. And for discrete variable $\mathbf{v}^{(d)}$, $\boldsymbol{\Phi}$ takes softmax over the network output as distribution $\hat{\mathbf{v}}^{(d)} = \text{softmax}\Big((\boldsymbol{\Phi}^v)^{(d)}\Big)$.

C.2. Featurization

For each protein atom, we represent it by several features, including a one-hot element indicator (H, C, N, O, S, Se) to identify the element type, a one-hot amino acid type indicator (20 dimension) to indicate the amino acid type, a one-dimension flag to indicate if the atom is a backbone atom, and a one-hot arm/scaffold region indicator to determine if the atom belongs to an arm or scaffold region based on its distance from the arm prior center.

The features for the ligand atom include a one-hot element indicator (C, N, O, F, P, S, Cl) to represent the element type, and a one-hot arm/scaffold indicator to differentiate between aromatic and non-aromatic atoms.

Two graphs are dynamically built for message passing in the protein-ligand complex, a k-nearest neighbors graph for ligand atoms and protein atoms, and a fully-connected graph for ligand atoms. The edge features in the k-nearest neighbors graph are the outer products of distance embedding (obtained by expanding the distance using radial basis functions) and edge type (a 4-dim one-hot vector indicating the type of edge). The ligand graph represents ligand bonds with a one-hot bond type vector (non-bond, single, double, triple, aromatic).

C.3. Model Hyperparameters

For the SE-(3) equivariant network, we experiment with kNN graphs with 32-nearest neighbor search to construct graph, 9 layers with hidden dimension of 128, 16 head attention, ReLU activation with Layer Normalization (Ba et al., 2016).

For the noise schedules, we use $\beta_1 = 1.5$ for atom types, $\sigma_1 = 0.03$ for atom coordinates, and train the model with discrete time loss of 1000 training steps.

For training, we use Adam optimizer with learning rate 0.005, batch size of 8, and exponential moving average of model parameters with a factor of 0.999. The training will converge within 15 epochs on a single RTX 3090, taking around 24 hours. For sampling, we take 100 sample steps with noise-reduced sampling strategy.

D. Additional Experimental Results

D.1. Full Evaluation Results

Binding Interaction In addition to our main results in Table 2, we provide evaluation in terms of key interactions in Table 4, i.e., No. Hydrogen Bond Donors (HB Donors), No. Hydrogen Bond Acceptors (HB Acceptors), van-der Waals contacts (vdWs) and Hydrophobic interactions as described in PoseCheck (Harris et al., 2023). It can be seen that under different molecule sizes, ours is consistently the best in forming hydrogen bonds, indicating that MolCRAFT finely captures key protein-ligand interactions.

Table 4: Number of key interactions for SBDD models under different molecule sizes. Top-1 values highlighted in **bold** text.

		HB Donors (Avg.)	HB Acceptor (Avg.)	vdWs (Avg.)	Hydrophobic (Avg.)	Molecule Size
	Reference	0.87	1.42	6.61	5.06	22.8
Smaller-size	AR	0.51	0.90	5.54	3.78	17.7
	Pocket2Mol	0.32	0.63	5.25	**4.53**	17.7
	FLAG	0.28	0.30	5.85	3.76	16.7
	Ours-small	**0.62**	**1.09**	**6.24**	4.42	17.8
Reference-size	TargetDiff	0.63	0.98	**7.92**	**5.43**	24.2
	Decomp-R	0.56	0.99	6.70	4.37	21.2
	Ours	**0.71**	**1.25**	7.38	5.07	22.7
Larger-size	Decomp-O	0.52	0.87	**9.14**	**6.84**	29.4
	Ours-large	**0.75**	**1.38**	8.64	6.07	29.4

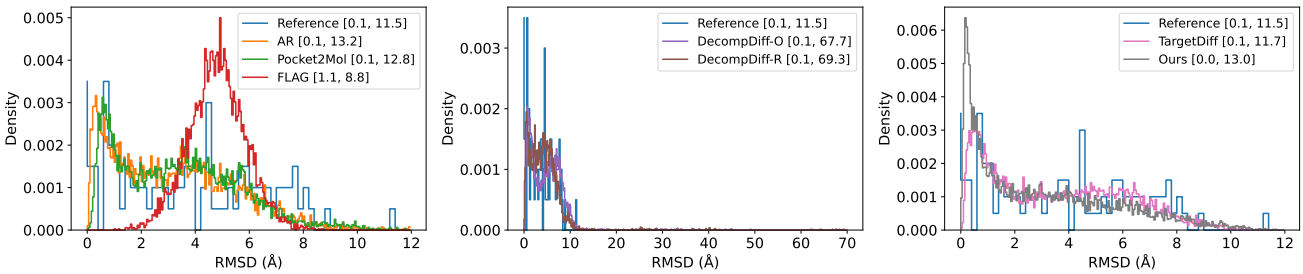


Figure 6: RMSD distribution of generated molecules compared with reference molecules. *Note:* values in [] are RMSD ranges for each model, which generally lie in [0, 13] except for significant outliers of DecompDiff.

Comformation Stability Besides the substructural analyis in Table 3, we present the full evaluation results for bond length, bond angle and torsion angle distributions of different types are presented in Table 5, 6, and 7, and further visualize the distributions in Fig. 8, 9, and 10, in order to look into details of 3D local structures.

As shown in Table 7, FLAG (Zhang et al., 2023) displays the best torsion angle distribution, owing to its fragment-based sampling strategy. However, FLAG has been observed to generate 0.2% severe outlier molecules that could not be parsed in JSD calculation, casting doubt on its local structures associated with fragment linkers.

Another perspective is given in Fig. 2, where we find that atom-based autoregressive models can hardly distinguish different bond types, yet diffusion models and our model show clear pattern similar to reference, which demonstrates another drawback of autoregressive models for SBDD. Since FLAG is a fragment-based autoregressive model, it inherits natural bonds by its nature. AR is a voxel-based model, thus its curves are not smooth as expected.

Next, we move on for a closer inspection into conformation stability of binding complex. According to Table 2, our model has the best RMSD of binding poses, which means 46% of our generated molecules already have accurate docking pose even without force field optimization or redocking, rendering our results more reliable for drug discovery. A notable phenomenon is the extremely low rate of FLAG, thus we investigate the overall distributions in Fig. 6. From this figure, we find AR (Luo et al., 2021), Pocket2Mol (Peng et al., 2022), TargetDiff and our model have closer distribution and value ranges to reference, yet FLAG has an obvious mode around 5, which explains the extremely low RMSD of FLAG.

For the reason why our model has better RMSD than reference, a possible explanation is that in CrossDocked (Francoeur et al., 2020b) training set, 52.4% poses are obtained via Vina Dock, while in the test set, only 37% molecules undergo redocking. This ratio accounts for the reference contains 34% RMSD below 2Å, and ours (46.1%) is approaching the limit of training set (52.4%).

Besides RMSD, we also report steric clashes (Harris et al., 2023) in protein-ligand complex conformation. As shown in Table 3, we achieve considerably fewer steric clashes. It could be seen that DecompDiff does surpass TargetDiff with the help of better priors, but it also suffers from the distribution shift of molecular size (29.4, larger than reference 22.8) introduced by manually chosen priors.

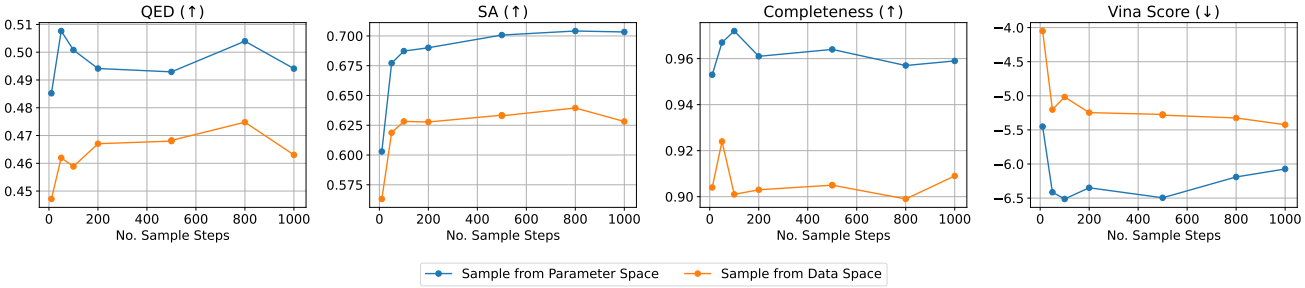


Figure 7: Ablation study on our proposed noise reduced sampling strategy under different sampling steps. Higher QED, SA, Completeness and lower Vina Score indicate better performance. The model is trained with 1000 steps.

D.2. Ablation Studies

As described in Fig. 5, our method samples at significantly faster speed. The reason of our considerable speed up mainly contributes to our improved sampling strategy from the parameter space, discussed in Sec. 4.2. For this reason, TargetDiff and DecompDiff requires 1000 steps for sampling, yet our model requires much fewer sampling steps to achieve comparable results. Therefore, we conduct ablation studies on sampling strategy and sampling steps so as to validate our design.

We test different sampling strategies with different steps for the same checkpoint, and sample 10 molecules each for 100 test proteins. We plot the curves of QED, SA, Completeness (↑) and Vina Score (↓) in Figure 7. As the sampling step increases to training steps, we found the original sampling strategy exhibits first enhanced then slightly decreased sample quality, possibly because the update of parameters is smoothed / oversmoothed by finer partitioned noise factor α, whereas the noise reduced strategy displays this tendency far earlier and generates the best quality of molecules with fewer sampling steps, indicating its high efficiency. Considering the overall sample quality, we decide to use 100 sampling steps for our model.

Table 5: Jensen-Shannon divergence of top-8 frequent bond length distributions between the reference and the generated molecules (↓ is better). No connected line, "=", and ":" represent single, double, and aromatic bonds. Top 2 results are highlighted with **bold text** and underlined text, respectively.

Bond	AR	Pocket2Mol	FLAG	TargetDiff	Decomp-O	Decomp-R	Ours
CC	0.610	0.494	**0.231**	0.369	0.359	0.326	0.290
C:C	0.450	0.414	0.366	0.263	0.251	**0.238**	0.330
CO	0.490	0.452	0.556	0.421	0.375	0.346	**0.339**
CN	0.472	0.422	0.529	0.362	0.342	0.337	**0.292**
C:N	0.551	0.484	0.470	**0.235**	0.269	**0.235**	0.242
OP	0.676	0.523	0.690	0.441	0.435	0.450	**0.347**
C=O	0.556	0.510	0.638	0.461	0.368	0.391	**0.342**
O=P	0.626	0.581	0.609	0.506	0.472	0.458	**0.369**
Avg.	0.554	0.485	0.511	0.382	0.359	0.348	**0.319**

Table 6: Jensen-Shannon divergence of top-8 frequent bond angle distributions between the reference and the generated molecules (↓ is better). Top 2 results are highlighted with **bold text** and underlined text.

Angle	AR	Pocket2Mol	FLAG	TargetDiff	Decomp-O	Decomp-R	Ours
CCC	0.372	0.380	**0.231**	0.345	0.358	0.337	0.280
C:C:C	0.572	0.480	0.199	0.283	0.266	0.255	**0.172**
CCO	0.477	0.475	**0.318**	0.440	0.403	0.394	0.319
C:C:N	0.537	0.506	0.465	0.454	**0.429**	**0.429**	0.446
CCN	0.447	0.443	0.388	0.437	0.404	0.419	**0.377**
CNC	0.535	**0.498**	0.510	0.521	**0.498**	0.504	0.499
COC	0.496	0.494	0.607	0.502	0.484	0.492	**0.460**
C:N:C	0.619	0.580	0.526	0.495	0.473	**0.462**	0.475
Avg.	0.507	0.482	0.406	0.435	0.414	0.412	**0.379**

Table 7: Jensen-Shannon divergence of top-8 frequent torsion angle distributions between the reference and the generated molecules (↓ is better). Top 2 results are highlighted with **bold text** and underlined text.

Torsion Angle	AR	Pocket2Mol	TargetDiff	Decomp-O	Decomp-R	Ours
CCCC	0.378	0.320	0.312	0.349	0.348	**0.286**
C:C:C:C	0.704	0.514	0.348	0.264	0.230	**0.130**
CCOC	0.419	0.401	**0.390**	0.392	0.391	0.393
CCCO	0.431	0.405	0.403	0.402	0.403	**0.396**
CCNC	0.430	0.437	0.423	0.403	**0.398**	0.401
C:C:N:C	0.664	0.504	0.386	0.285	**0.197**	0.212
C:C:C:N	0.663	0.512	0.441	0.388	0.316	**0.281**
C:N:C:N	0.742	0.535	0.476	0.366	**0.247**	0.303
CCCN	0.495	0.549	0.512	0.501	0.525	**0.493**
Avg.	0.547	0.464	0.410	0.372	0.340	**0.322**

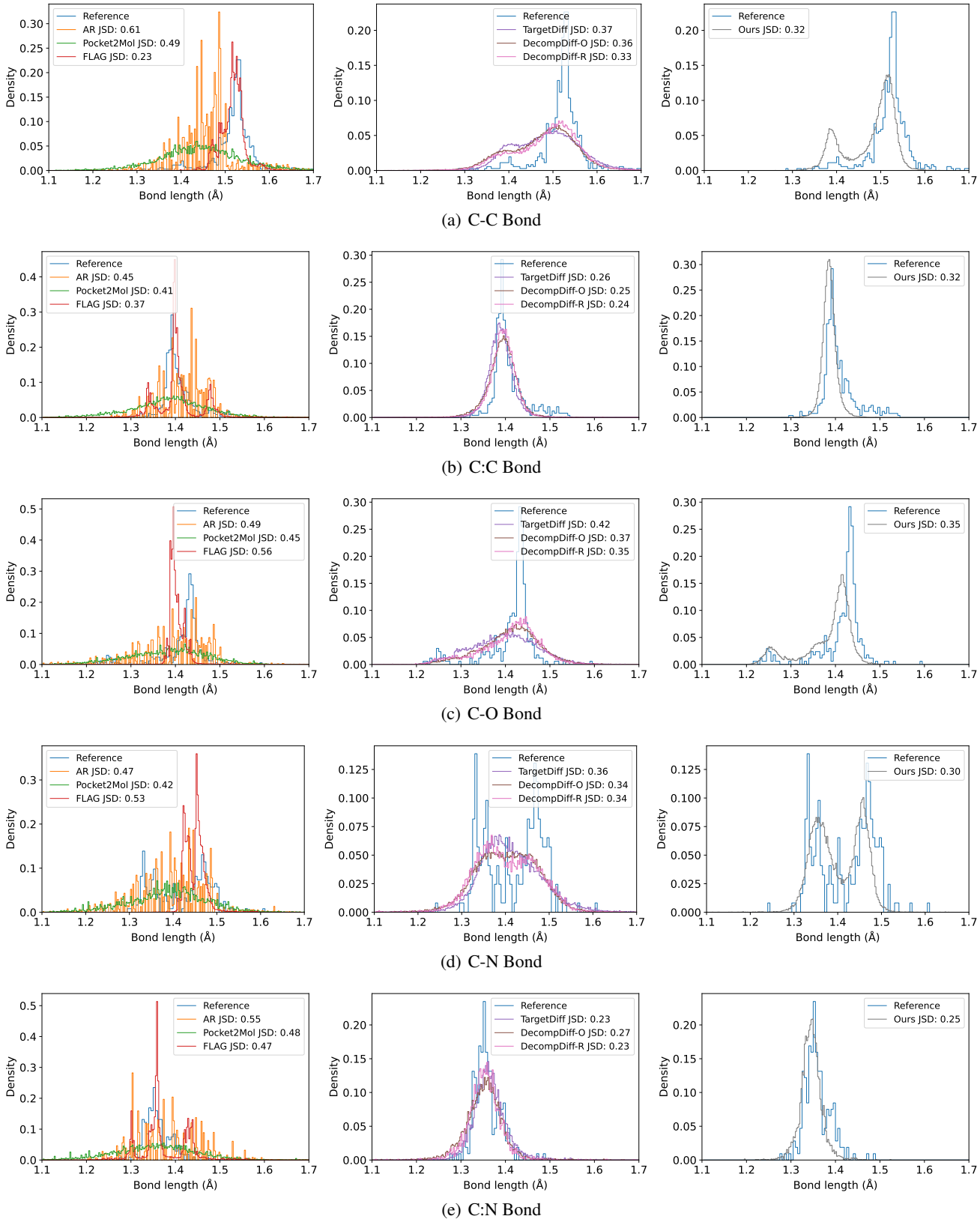
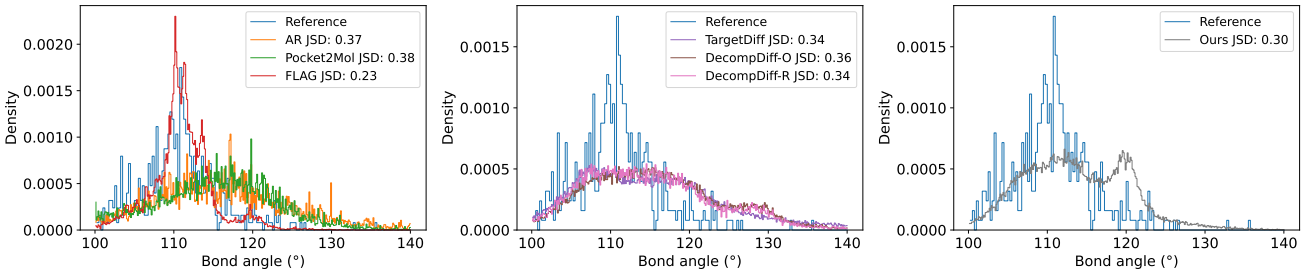
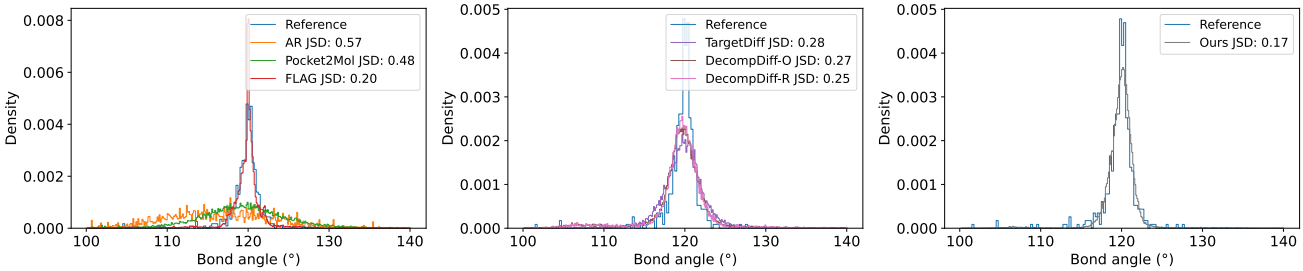


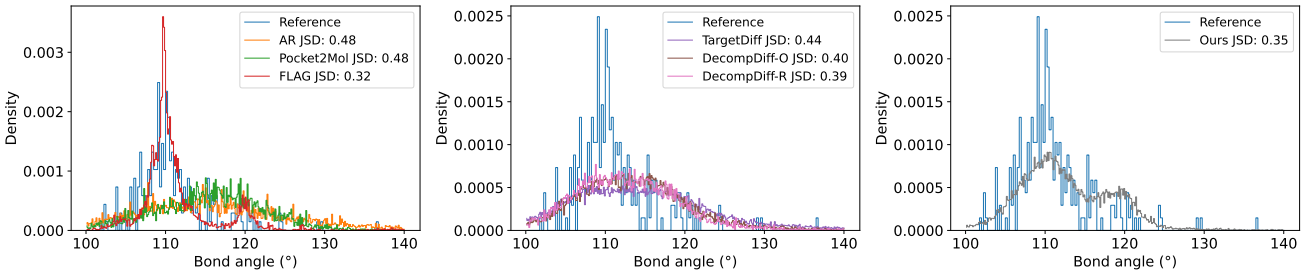
Figure 8: Bond length distribution of generated molecules compared with reference molecules.



(a) C-C-C Bond Angle

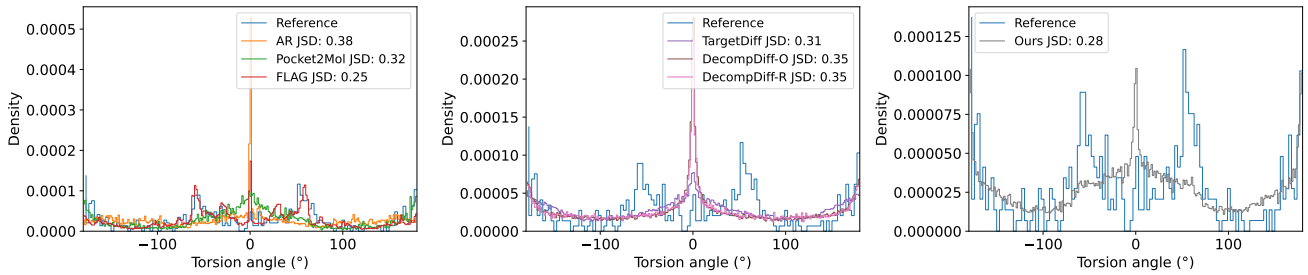


(b) C:C:C Bond Angle

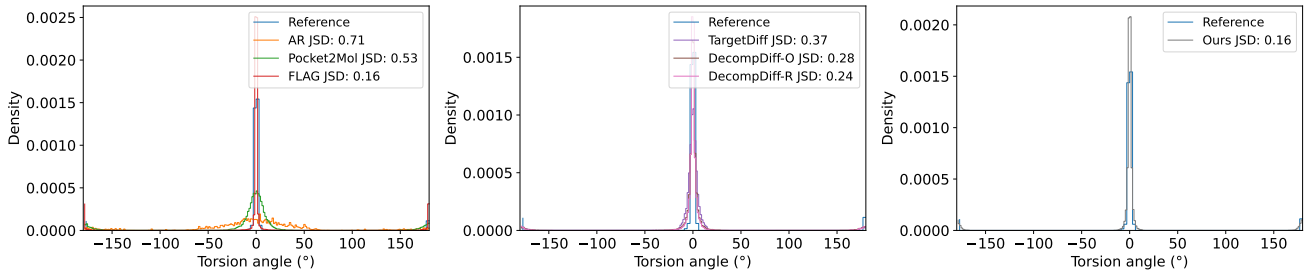


(c) C-C-O Bond Angle

Figure 9: Bond angle distribution of generated molecules compared with reference molecules.



(a) C-C-C-C Torsion Angle



(b) C:C:C:C Torsion Angle

Figure 10: Torsion angle distribution of generated molecules compared with reference molecules.

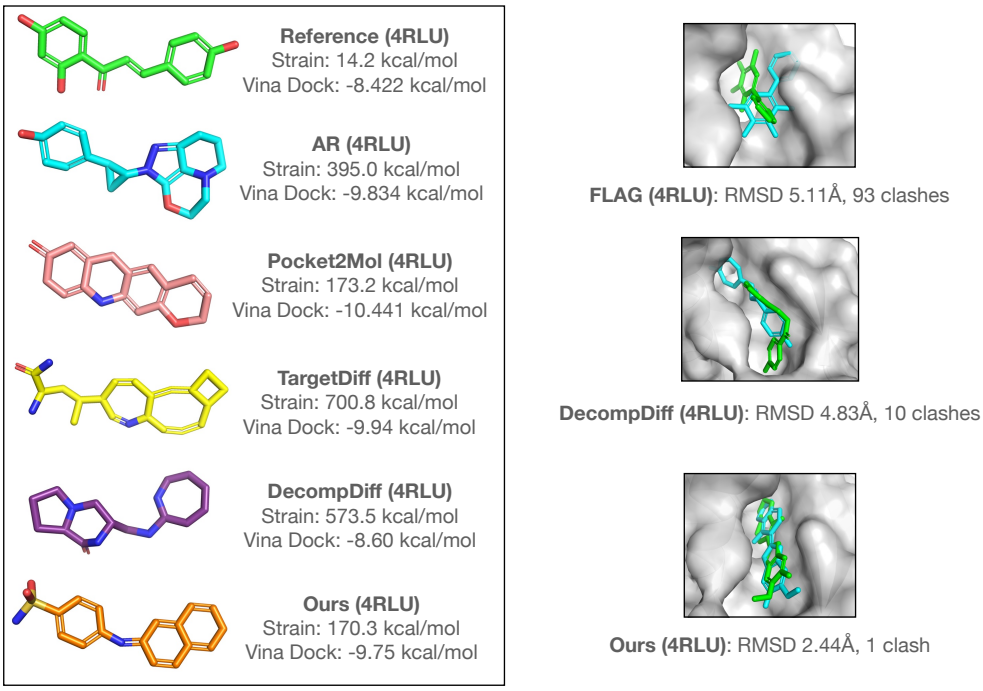


Figure 11: Visualization of molecules for a randomly chosen test protein (PDB ID: 4RLU), representative in the median values of strain energy (Strain) and RMSD.