DECOMPDIFF: Diffusion Models with Decomposed Priors for Structure-Based Drug Design

Jiaqi Guan [*1 2] **Xiangxin Zhou** [*3 4 2] **Yuwei Yang** [2] **Yu Bao** [2]
Jian Peng [1] **Jianzhu Ma** [5] **Qiang Liu** [3 4] **Liang Wang** [3 4] **Quanquan Gu** [2]

Abstract

Designing 3D ligands within a target binding site is a fundamental task in drug discovery. Existing structured-based drug design methods treat all ligand atoms equally, which ignores different roles of atoms in the ligand for drug design and can be less efficient for exploring the large drug-like molecule space. In this paper, inspired by the convention in pharmaceutical practice, we decompose the ligand molecule into two parts, namely arms and scaffold, and propose a new diffusion model, DECOMPDIFF, with decomposed priors over arms and scaffold. In order to facilitate the decomposed generation and improve the properties of the generated molecules, we incorporate both bond diffusion in the model and additional validity guidance in the sampling phase. Extensive experiments on CrossDocked2020 show that our approach achieves state-of-the-art performance in generating high-affinity molecules while maintaining proper molecular properties and conformational stability, with up to -8.39 Avg. Vina Dock score and 24.5% Success Rate. The code is provided at `https://github.com/bytedance/DecompDiff`

1 Introduction

Modern deep learning is revolutionizing many subfields in drug discovery, among which structure-based drug design (SBDD) (Anderson, 2003) is an important yet challenging one. Aiming at generating 3D ligand molecules conditioned on a target binding site, SBDD requires models to generate drug-like molecules with stable 3D structures and high binding affinities to the target. Recently, deep generative models have been successfully employed to achieve this goal. For example, autoregressive models (Luo & Ji, 2021; Liu et al., 2022b; Peng et al., 2022) have achieved promising performance in SBDD tasks, which generate 3D molecules in the target binding site by adding atoms and bonds iteratively. However, autoregressive models suffer from error accumulation and require a generation order, which is nontrivial for molecular graphs. In order to overcome the limitation of autoregressive models, recent works (Guan et al., 2023; Schneuing et al., 2022; Lin et al., 2022) use diffusion models (Ho et al., 2020) to approximate the distribution of atom types and positions from a standard Gaussian prior, and use a post-processing algorithm to assign bonds between atoms. These diffusion model-based methods can model local and global interactions between atoms simultaneously and achieve better performance than autoregressive models. Despite the state-of-the-art performance, existing diffusion model-based approaches neglect bonds in the modeling process, which may lead to unreasonable molecular structures. Moreover, diffusion model-based approaches treat the ligand molecule as a whole and learn the overall correspondence between the target binding site and the ligand. However, atoms within the same ligand can be designed for different functions. Therefore, treating all ligand atoms equally may not be the best way for SBDD, especially considering the tremendous drug-like space (Virshup et al., 2013) to explore and the limited amount of high quality target-ligand complexes (Berman et al., 2000) for training. This motivates us to study how to properly incorporate function-related prior knowledge into diffusion model-based SBDD methods.

In fact, decomposing ligands into smaller functional regions is a common practice in conventional drug design. As shown

[*]Equal contribution [1]Department of Computer Science, University of Illinois Urbana-Champaign, USA [2]ByteDance Research (Work was done during Jiaqi's and Xiangxin's internship at ByteDance) [3]School of Artificial Intelligence, University of Chinese Academy of Sciences [4]Center for Research on Intelligent Perception and Computing (CRIPAC), State Key Laboratory of Multimodal Artificial Intelligence Systems (MAIS), Institute of Automation, Chinese Academy of Sciences (CASIA) [5]Institute for AI Industry Research, Tsinghua University, Beijing, China. Correspondence to: Xiangxin Zhou <zhouxiangxin1998@gmail.com>, Quanquan Gu <quanquan.gu@bytedance.com>.

Proceedings of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023. Copyright 2023 by the author(s).

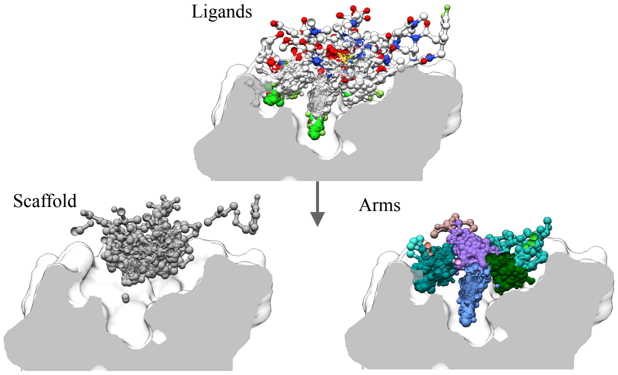

Figure 1. Ligand molecules can be decomposed into arms and scaffold. Using MDM2 as an example, small molecule ligands are collected and displayed in the upper panel (colors represent different atom types). The ligand atoms are separated into arms and scaffold based on their distance to the protein surface. Arms (lower right) form direct contact with the target, while scaffold (lower left) connects the arms together. Arm atoms are further clustered based on their positions, and the cluster (colored atom groups) show strong shape complementarity with local subpockets.

in Figure 1, ligands can be decomposed into scaffold and arms. The arms are responsible for interacting with the target to achieve high binding affinity, while the scaffold is responsible for positioning the arms into the desired binding regions. In lead optimization, a scaffold is identified first, and then a series of analogs are developed by altering the arms for further activity optimization (Wermuth, 2011); in scaffold hopping practice, arms (or fragments) are placed on the target surface first, then a scaffold (or linker) is placed to connect the arms (Schneider et al., 1999). Inspired by the convention in traditional drug design, we aim to incorporate decomposed molecules, i.e., arms and scaffold, into diffusion models.

In this paper, we propose DECOMPDIFF, a new diffusion model with data-dependent decomposed priors for SBDD. The decomposed priors respect the natural decomposition of a ligand into arms and scaffold when it interacts with a target. We also introduce a diffusion process on bonds to incorporate the bond generation in an end-to-end fashion instead of post-processing. To facilitate the generation process, we develop additional validity guidance in the sampling phase, such as promoting the connection between the scaffold and arms and avoiding a geometric clash between the generated molecule and the target. We highlight our main contributions as follows:

- We propose a diffusion model with decomposed priors for structure-based drug design, which incorporates the natural decomposition of a ligand molecule into function-related regions.
- We consider both atom and bond diffusion processes in the model to simultaneously generate atoms and bonds for improving drug-likeness and synthesizability.
- We design and incorporate several guidance terms in the decomposed generation process to improve the molecular validity.
- Putting all the above pieces together, our method can generate ligand molecules with a -8.39 Avg. Vina Dock score and 24.5% Success Rate, achieving the new SOTA on the CrossDocked2020 benchmark.

2 Related Work

Structure-Based Drug Design SBDD aims to generate 3D molecules in the presence of a target binding site. Early attempts use molecular docking for indirect consideration of the target (Yang et al., 2021; Li et al., 2021), which optimize ligand in the 3D space with docking score as a reward. Instead of considering every aspect of the target binding site, Long et al. (2022); Adams & Coley (2022) proposed to only use the shape information for ligand generation. More recent work directly model the correspondence between targets and ligands using target-ligand complexes. Ragoza et al. (2022b) represented molecules as atomic density grids and used conditional variational autoencoders to learn the 3D ligand distributions. Luo et al. (2021); Liu et al. (2022a); Peng et al. (2022) employed autoregressive models to generate atoms (and bonds) step-by-step. Recently, diffusion models start to play a role in SBDD (Schneuing et al., 2022; Guan et al., 2023), and existing approaches denoise atom types and positions sampled from a Gaussian prior. Unlike the previous approaches, which treat all the ligand atoms as a whole, our method decomposes ligand into arms and scaffold, and incorporate related prior knowledge into diffusion models for better molecular generation.

Decomposed Molecular Generation The key idea of decomposed molecular generation is to divide a ligand into subregions and generate each part separately. This approach can significantly reduce the search space for drug-like molecules by constraining the search effort in more confined regions. Scaffold hopping (or linker design) aims to generate a scaffold/linker to connect the existing arms/fragments. Yang et al. (2020) developed a linker design algorithm to connect two SMILES strings. DeLinker (Imrie et al., 2020), DEVELOP (Imrie et al., 2021) and 3DLinker (Huang et al., 2022) extend the application to connect two 3D fragments using autoregressive models. DiffLinker (Igashov et al., 2022) further relieve the constraints on the number of fragments to be connected. On the contrary, lead optimization starts from a scaffold structure and optimizes it by adding arms/fragments. Arús-Pous et al. (2020) developed a SMILES-based scaffold decoration method. Lim et al. (2020) and Li et al. (2019) achieved this goal on 2D molecular graphs, and Imrie et al. (2021) took consideration of the 3D scaffold information when generating 2D optimized ligands. However, the aforementioned methods all need a reliable initial structure for molecular design. In contrast, our method retains the benefits of molecular decomposition and can be applied to *de novo* drug design to generate molecules from scratch.

Diffusion Models Diffusion models (Sohl-Dickstein et al., 2015; Ho et al., 2020; Song & Ermon, 2019), which generate samples by iteratively denoising data points sampled from a prior distribution, have shown promising results in generating images (Dhariwal & Nichol, 2021; Nichol et al., 2021; Ramesh et al., 2022), texts (Li et al., 2022), and speech (Kong et al., 2021). Considering that the initial diffusion model aims to learn the data distribution, it is often necessary to modify it for various realistic controlled generation scenarios. Some researchers introduce classifier guidance (Dhariwal & Nichol, 2021), or classifier-free guidance (Ho & Salimans, 2022) for introducing controllable goals. Lee et al. (2021) leveraged conditional information as a non-standard data-dependent adaptive prior for improving conditional denoising diffusion models. Vignac et al. (2022) further showed that the prior distribution closer to data distribution can lead to superior performance. These works strongly support our design of the prior distribution for the diffusion model. It is worth noting that we design priors for more complex 3D structure data, which is less studied in previous work.

3 Method

In this section, we present DECOMPDIFF, which injects informative priors of the decomposed arms and scaffolds into a diffusion model for SBDD. We first define the SBDD task and introduce the standard diffusion model in Sec. 3.1. Then, we show how to introduce decomposed priors into the diffusion model in Sec. 3.2. In Sec. 3.3, we present bond diffusion and the network used for 3D molecular graph generation. Finally, in Sec. 3.4, we describe an effective guided sampling approach to improve the validity and quality of the generated molecules.

3.1 Preliminaries

In SBDD, we are provided with a protein binding site, which can be represented as a set of N_P atoms $\mathcal{P} = \{(\boldsymbol{x}_P^{(i)}, \boldsymbol{v}_P^{(i)})\}_{i=1}^{N_P}$. The goal is to generate ligand molecules that can bind with the protein. Similarly, the ligand molecule can be represented as a set of N_M atoms $\mathcal{M} = \{(\boldsymbol{x}_M^{(i)}, \boldsymbol{v}_M^{(i)})\}_{i=1}^{N_M}$. Here $\boldsymbol{x} \in \mathbb{R}^3$ and $\boldsymbol{v} \in \mathbb{R}^d$ denote the position and type of the atom respectively. The number of atoms N_M can be sampled from an empirical distribution (Hoogeboom et al., 2022; Guan et al., 2023) or predicted by a neural network (Lin et al., 2022), and is not involved in the diffusion process. By denoting the ligand molecule as $M = [\mathbf{x}, \mathbf{v}]$ for brevity, where $\mathbf{x} \in \mathbb{R}^{N_M \times 3}$ and $\mathbf{v} \in \mathbb{R}^{N_M \times d}$, the SBDD task can be formulated as modeling the conditional distribution $p(M|\mathcal{P})$.

Considering this task in the Denoising Diffusion Probabilistic Model (DDPM) framework, a small Gaussian noise is gradually injected into data as a Markov chain, leading to the following forward diffusion process:

$$q(M_{1:T}|M_0, \mathcal{P}) = \prod_{t=1}^{T} q(M_t|M_{t-1}, \mathcal{P}), \tag{1}$$

where the data $M_0 \sim p(M_0|\mathcal{P})$ and $M_1, M_2, \cdots, M_T$ is a sequence of latent variables induced by the diffusion process. The reverse process, also known as the generative process, learns to recover data by iteratively denoising with a neural network parameterized by θ as follows:

$$p_\theta(M_{0:T-1}) = \prod_{t=1}^{T} p_\theta(M_{t-1}|M_t, \mathcal{P}). \tag{2}$$

The distribution of $p(M_T)$ induced by the forward diffusion process is Gaussian and works as the prior distribution during sampling.

We can derive a tractable variational lower bound on the log-likelihood of data M_0, also known as evidence lower bound (ELBO), due to the property of standard Gaussian distribution. To align with the concept of loss, we can formulate it as follows:

$$-\log p(M_0|\mathcal{P}) \leq L_0 + \sum_{t=1}^{T-1} L_t + L_T, \tag{3}$$

where $L_0 = -\mathbb{E}[\log p_\theta(M_0|M_1, \mathcal{P})]$ denotes the data likelihood, $L_T = D_{\mathrm{KL}}(q(M_T|M_0, \mathcal{P}) \| p_\theta(M_T))$ is the Kullback-Leibler (KL) divergence between the final distribution induced by the diffusion process $q(M_T|M_0, \mathcal{P})$ and the prior distribution $p_\theta(M_T)$, and $L_t = -D_{\mathrm{KL}}(q(M_t|M_{t+1}, M_0, \mathcal{P}) \| p_\theta(M_t|M_{t+1}, \mathcal{P}))$.

3.2 Diffusion Model with Decomposed Priors

Motivated by the natural decomposition of the ligand molecules, we decompose a ligand into fragments $\mathcal{K}$, consisting of arms $\mathcal{A}$ and scaffold $\mathcal{S}$ ($|\mathcal{A}| \geq 1, |\mathcal{S}| \leq 1, K = |\mathcal{K}| = |\mathcal{A}| + |\mathcal{S}|$). From the perspective of machine learning, fragmentation provides natural clusters of ligand atoms. Intuitively, if we could leverage these natural clusters as informative prior, approximating p_θ would be easier than a non-informative Gaussian prior. Following the optimal prior hypothesis (Vignac et al., 2022), we estimate the prior of each cluster with a Gaussian distribution, which is obtained by Maximum Likelihood Estimation (MLE) on the positions of its cluster members. Specifically, given a protein binding site, ligand atoms exhibit a multimodal distribution spatially. In the training phase, the prior can be obtained from reference ligands as described above. In the test phase where reference ligands are not always available or ideal,

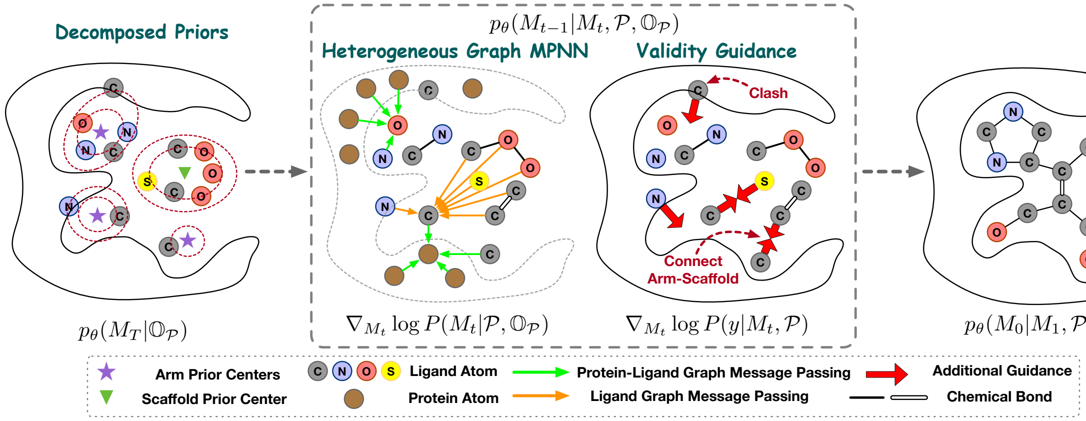
Decomposed Priors
$p_\theta(M_{t-1}|M_t, \mathcal{P}, \mathbb{O}_\mathcal{P})$
Heterogeneous Graph MPNN
Validity Guidance
Clash
Connect
Arm-Scaffold
$p_\theta(M_T|\mathbb{O}_\mathcal{P})$
$\nabla_{M_t} \log P(M_t|\mathcal{P}, \mathbb{O}_\mathcal{P})$
$\nabla_{M_t} \log P(y|M_t, \mathcal{P})$
$p_\theta(M_0|M_1, \mathcal{P}, \mathbb{O}_\mathcal{P})$
Arm Prior Centers
Scaffold Prior Center
Ligand Atom
Protein Atom
Protein-Ligand Graph Message Passing
Ligand Graph Message Passing
Additional Guidance
Chemical Bond

Figure 2. Overview of the sampling process of DecompDiff. (a) The initial atoms are sampled from informative decomposed priors. (b) An equivariant network on heterogeneous graphs denoises atom coordinates, atom types and bond types simultaneously. (c) The validity guidance alleviates the protein-ligand clash problem and encourages arms and scaffold to connect.

the prior can be obtained by human experts or rule-based algorithms. We provide more details for obtaining priors and fragmentation in Appendix A and Appendix B.

Following the above description, we can obtain a set of data-dependent priors $\mathbb{O}_\mathcal{P} = \{\boldsymbol{\mu}^{\mathcal{P}}_{1:K}, \boldsymbol{\Sigma}^{\mathcal{P}}_{1:K}, \boldsymbol{H}_\mathcal{P}\}$, where $\boldsymbol{\mu}_k \in \mathbb{R}^3$ is the prior center, $\boldsymbol{\Sigma}_k \in \mathbb{R}^{3\times 3}$ is the prior covariance matrix and $\boldsymbol{H}_\mathcal{P} = \{\eta^\mathcal{P} \in \{0,1\}^{N_M \times K} | \sum_{k=1}^{K} \eta^{\mathcal{P}}_{ik} = 1\}$ is the prior-atom mapping conditioned on the protein $\mathcal{P}$, i.e., $\eta_{ik} = 1$ indicates that the i-th molecule atom corresponds to the k-th prior. Note that an atom can only be assigned with one single prior. Next, we will describe how the diffusion process, generative process and ELBO will be adjusted accordingly when including the decomposed priors.

Our goal is to model the conditional molecular distribution $p_\theta(M|\mathcal{P}) = p_\theta(M|\mathcal{P}, \mathbb{O}_\mathcal{P})$. Since our informative prior is defined in the 3D coordinate space, we only consider the diffusion and generative process of ligand atom positions $\mathbf{x}$ and omit the ligand atom types $\mathbf{v}$ for clarity in the following derivation. The condition $\mathcal{P}$ in $\mathbb{O}_\mathcal{P}$ is also omitted.

To distinguish the difference induced by the decomposed priors from the standard ones, we highlight the critical differences of equations in blue. To achieve SE(3)-equivariance, we apply a center shifting operation (Xu et al., 2022) on every atom according to its corresponding prior center, denoted as $\tilde{\mathbf{x}}^{(i)}_{t,k} = \mathbf{x}^{(i)}_t - \boldsymbol{\mu}_k$. We follow the definitions and notations related to the noise schedule $\alpha_t, \beta_t, \bar{\alpha}_t, \tilde{\beta}_t$ in Ho et al. (2020). With data-dependent prior as the additional context, the diffusion and generative process in Equations (1) and (2) can be extended as follows:

$$q(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathcal{P}) = \prod_{i=1}^{N_M}\sum_{k=1}^{K} \eta_{ik} \mathcal{N}(\tilde{\mathbf{x}}^{(i)}_{t,k}; \tilde{\mathbf{x}}^{(i)}_{t-1,k}, \beta_t \boldsymbol{\Sigma}_k) \tag{4}$$

$$p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0, \mathcal{P}) = \prod_{i=1}^{N_M}\sum_{k=1}^{K} \eta_{ik} \mathcal{N}(\tilde{\mathbf{x}}^{(i)}_{t-1,k}; \tilde{\boldsymbol{\mu}}_t(\tilde{\mathbf{x}}^{(i)}_{t,k}, \tilde{\mathbf{x}}^{(i)}_{0,k}), \tilde{\beta}_t \boldsymbol{\Sigma}_k) \tag{5}$$

where $\tilde{\boldsymbol{\mu}}_t(\tilde{\mathbf{x}}^{(i)}_{t,k}, \tilde{\mathbf{x}}^{(i)}_{0,k}) = \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}\tilde{\mathbf{x}}^{(i)}_{t,k} + \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t}\tilde{\mathbf{x}}^{(i)}_{0,k}$.

The prior distribution induced by the forward diffusion process can be derived as follows:

$$p(\mathbf{x}_T|\mathcal{P}) = \prod_{i=1}^{N_M}\sum_{k=1}^{K} \eta_{ik} \mathcal{N}(\mathbf{x}^{(i)}_T; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k). \tag{6}$$

The evidence lower bound (ELBO) is still traceable as in Equation (3) where L_0, L_t, and L_T are as follows:

$$\begin{aligned} L_0 = -\mathbb{E}_{\mathbf{x}_0,\mathbf{x}_1}\Bigg[\sum_{i=1}^{N_M}\sum_{k=1}^{K}\eta_{ik}\Bigg[\frac{1}{2\tilde{\beta}_0}\left\|\mathbf{x}^{(i)}_0 - \hat{\mathbf{x}}^{(i)}_{0,1}\right\|^2_{\boldsymbol{\Sigma}_k^{-1}} \\ + \frac{1}{2}\ln\det\left(\boldsymbol{\Sigma}_k^{-1}\right)\Bigg]\Bigg] + C_0, \end{aligned} \tag{7}$$

$$L_t = \mathbb{E}_{\mathbf{x}_0,\mathbf{x}_t}\left[\gamma_t\sum_{i=1}^{N_M}\sum_{k=1}^{K}\eta_{ik}\left\|\mathbf{x}^{(i)}_0 - \hat{\mathbf{x}}^{(i)}_{0,t+1}\right\|^2_{\boldsymbol{\Sigma}_k^{-1}}\right] + C_t, \tag{8}$$

$$L_T = \mathbb{E}_{\mathbf{x}_0}\left[\frac{\bar{\alpha}_T}{2}\sum_{i=1}^{N_M}\sum_{k=1}^{K}\eta_{ik}\left\|\tilde{\mathbf{x}}^{(i)}_{0,k}\right\|^2_{\boldsymbol{\Sigma}_k^{-1}}\right] + C_T, \tag{9}$$

where C_0, C_t, and C_T are constant terms, and $\gamma_t = \frac{\bar{\alpha}_{t-1}\beta_t^2}{2(1-\bar{\alpha}_t)^2\tilde{\beta}_t}$. $\hat{\mathbf{x}}^{(i)}_{0,t} = f_{\boldsymbol{\theta}}(\mathbf{x}^{(i)}_t, t, \mathcal{P})$ is implemented by a SE(3)-equivariant neural network parameterized by $\boldsymbol{\theta}$. Note that the input of the neural network $\mathbf{x}^{(i)}_t = \sqrt{\bar{\alpha}_t}\sum_{k=1}^{K}\tilde{\mathbf{x}}^{(i)}_{0,k} + \sqrt{1-\bar{\alpha}_t}\epsilon_k + \boldsymbol{\mu}_k$ is different from that under the standard prior.

Proposition 3.1. *Let* $-\text{ELBO}_{\text{decomp}}(\boldsymbol{\theta})$ *and* $-\text{ELBO}_{\text{standard}}(\boldsymbol{\theta})$ *denote the* $-$ELBO *losses under the decomposed prior and the standard Gaussian prior respectively. Suppose that* $f_{\boldsymbol{\theta}}$ *is a simple graph neural network with an equivariant linear layer. If the decomposed prior aligns with data distribution, we have* $\min_{\boldsymbol{\theta}} -\text{ELBO}_{\text{decomp}}(\boldsymbol{\theta}) \leq \min_{\boldsymbol{\theta}} -\text{ELBO}_{\text{standard}}(\boldsymbol{\theta})$.

Proposition 3.1 provides insights into why our priors can induce better results. Please see Appendix C for its proof.

3.3 Bond Diffusion and Model Architecture

Introducing Bond Diffusion Existing diffusion models for 3D molecule generation (Hoogeboom et al., 2022; Guan et al., 2023; Schneuing et al., 2022; Lin et al., 2022) only consider generating atom coordinates and atom types with neural networks, while adding bonds by a post-processing algorithm, such as that implemented in OpenBabel (O'Boyle et al., 2011). Ideally, this paradigm can work well if the atom coordinates are predicted accurately. However, since the distributions of bond distances are very sharp and a small error may lead to totally different molecules, adding bonds based on imperfect atom coordinates with a post-processing algorithm is not always reliable.

To address this problem, we develop a new diffusion framework for 3D molecular *graph* generation which also considers bonds in the dynamics. Specifically, we extend the molecular representation as $\mathcal{M} = \{(\boldsymbol{x}_i, \boldsymbol{v}_i, \boldsymbol{b}_{ij})\}_{i,j \in \{1,\dots,N_M\}}$. We apply the discrete diffusion (Hoogeboom et al., 2021) for bond types, similar to how we model the atom types. The forward diffusion becomes as follows:

$$q(M_t|M_{t-1}, \mathcal{P}) = \mathcal{N}(\mathbf{x}_t; \sqrt{1-\beta_t}\mathbf{x}_{t-1}, \beta_t \mathbf{I}) \\ \cdot\, \mathcal{C}(\mathbf{v}_t|(1-\beta_t)\mathbf{v}_{t-1} + \beta_t/K_a) \cdot \mathcal{C}(\mathbf{b}_t|(1-\beta_t)\mathbf{b}_{t-1} + \beta_t/K_b), \tag{10}$$

where K_a and K_b are the number of atom types and bond types respectively. Benefiting from decomposing the drug space, we do not have to build the bond dynamics on the fully connected graph of the ligand molecule, but instead inside of arms/scaffold and between arms and scaffold: $\mathbf{b} = \{b_{ij}|i \in \mathcal{K}_n, j \in \mathcal{K}_n\}_{n=1:|\mathcal{K}|} \cup \{b_{ij}|i \in \mathcal{A}_n, j \in \mathcal{S}\}_{n=1:|\mathcal{A}|}$

Equivariant Network with Nodes and Edges Update Inspired by recent progress in equivariant neural networks (Thomas et al., 2018; Fuchs et al., 2020; Satorras et al., 2021; Guan et al., 2022), we propose a new equivariant neural network to denoise 3D molecular graph. Specifically, we maintain both node-level and edge-level hidden states in the neural network to better reflect the bond diffusion, unlike the commonly used EGNN (Satorras et al., 2021) where only node-level representation is considered.

We first build a k-nearest neighbors (knn) graph $\mathcal{G}_K$ upon ligand atoms and protein atoms to model the protein-ligand interaction:

$$\Delta\mathbf{h}_{K,i} \leftarrow \sum_{j \in \mathcal{N}_K(i)} \phi_{m_K}(\mathbf{h}_i, \mathbf{h}_j, \|\mathbf{x}_i - \mathbf{x}_j\|, E_{ij}, t), \tag{11}$$

where $\mathbf{h}$ is the atom's hidden state, $\mathcal{N}_K(i)$ is the neighbors of i in $\mathcal{G}_K$, E_{ij} indicates the edge ij is a protein-protein, ligand-ligand or protein-ligand edge.

We also build a fully/partially connected ligand graph $\mathcal{G}_L$ upon ligand atoms to model the interaction inside the ligand:

$$\begin{aligned} \mathbf{m}_{ij} &\leftarrow \phi_d(\|\mathbf{x}_i - \mathbf{x}_j\|, \mathbf{e}_{ij}) \\ \Delta\mathbf{h}_{L,i} &\leftarrow \sum_{j \in \mathcal{N}_L(i)} \phi_{m_L}(\mathbf{h}_i, \mathbf{h}_j, \mathbf{m}_{ji}, t), \end{aligned} \tag{12}$$

where $\mathbf{e}$ is the bond's hidden state. Based on the messages aggregated from heterogeneous graphs, we update the atom's hidden state as Equation (13):

$$\mathbf{h}_i \leftarrow \mathbf{h}_i + \phi_h(\Delta\mathbf{h}_{K,i} + \Delta\mathbf{h}_{L,i}). \tag{13}$$

We update the bond's hidden state following a directional message passing (Yang et al., 2019; Gasteiger et al., 2020) schema as Equation (14):

$$\mathbf{e}_{ji} \leftarrow \sum_{k \in \mathcal{N}_L(j) \setminus \{i\}} \phi_e(\mathbf{h}_i, \mathbf{h}_j, \mathbf{h}_k, \mathbf{m}_{kj}, \mathbf{m}_{ji}, t). \tag{14}$$

Finally, the atom positions of ligand molecules are updated as Equation (15):

$$\begin{aligned} \Delta\mathbf{x}_{K,i} &\leftarrow \sum_{j \in \mathcal{N}_K(i)} (\mathbf{x}_j - \mathbf{x}_i)\phi_{x_K}(\mathbf{h}_i, \mathbf{h}_j, \|\mathbf{x}_i - \mathbf{x}_j\|, t) \\ \Delta\mathbf{x}_{L,i} &\leftarrow \sum_{j \in \mathcal{N}_L(i)} (\mathbf{x}_j - \mathbf{x}_i)\phi_{x_L}(\mathbf{h}_i, \mathbf{h}_j, \|\mathbf{x}_j - \mathbf{x}_i\|, \mathbf{m}_{ji}, t) \\ \mathbf{x}_i &\leftarrow \mathbf{x}_i + (\Delta\mathbf{x}_{K,i} + \Delta\mathbf{x}_{L,i}) \cdot \mathbb{1}_{\text{mol}}, \end{aligned} \tag{15}$$

where $\mathbb{1}_{\text{mol}}$ is the indicator of ligand atoms since we assume the protein atoms are fixed as the context.

We obtain the initial atom hidden state $\mathbf{h}^0$ and bond hidden state $\mathbf{e}^0$ by two embedding layers that encode atom, bond and decomposition information. The final hidden states $\mathbf{h}^L$ and $\mathbf{e}^L$ are fed into two MLPs to obtain the predicted atom type $\hat{\mathbf{v}}_i = \text{softmax}(\text{MLP}(\mathbf{h}_i^L))$ and bond type $\hat{\mathbf{b}}_{ij} = \text{softmax}(\text{MLP}(\mathbf{e}_{ij}^L + \mathbf{e}_{ji}^L))$. Since atom type and bond type are subject to categorical distributions, we can directly compute the KL divergence between their estimated posteriors and the truth posteriors as losses:

$$L_t^{(v)} = \sum_{k=1}^{K_a} \boldsymbol{c}(\mathbf{v}_t, \mathbf{v}_0)_k \log \frac{\boldsymbol{c}(\mathbf{v}_t, \mathbf{v}_0)_k}{\boldsymbol{c}(\mathbf{v}_t, \hat{\mathbf{v}}_0)_k}, \tag{16}$$

$$L_t^{(b)} = \sum_{k=1}^{K_b} \boldsymbol{c}(\mathbf{b}_t, \mathbf{b}_0)_k \log \frac{\boldsymbol{c}(\mathbf{b}_t, \mathbf{b}_0)_k}{\boldsymbol{c}(\mathbf{b}_t, \hat{\mathbf{b}}_0)_k}, \tag{17}$$

where $\boldsymbol{c}(\mathbf{v}_t, \mathbf{v}_0) = \boldsymbol{c}^\star / \sum_{k=1}^{K_a} c_k^\star$ and $\boldsymbol{c}^\star(\mathbf{v}_t, \mathbf{v}_0) = [\alpha_t \mathbf{v}_t + (1-\alpha_t)/K_a] \odot [\bar{\alpha}_{t-1}\mathbf{v}_0 + (1-\bar{\alpha}_{t-1})/K_a]$. $\boldsymbol{c}(\mathbf{b}_t, \mathbf{b}_0)$ is defined in the similar way. Combined with the decomposed atom position loss described in Sec. 3.2, the final loss is a weighted sum of atom position loss, atom type loss and bond type loss: $L = L_t^{(x)} + \gamma_a L_t^{(v)} + \gamma_b L_t^{(b)}$, where $L_t^{(x)}$ is from Equations (7) to (9). More implementation details can be found in Appendix E.

3.4 Validity Guidance

To further improve the validity and quality of generated molecules, we introduce additional drift terms to guide the Langevin dynamics during sampling. Inspired by classifier guidance for conditional sampling of DDPM (Dhariwal & Nichol, 2021), we design several guidance terms

$\nabla_{\boldsymbol{x}_t} \log P(y|\boldsymbol{x}_t)$ which are applied during the sampling phase to constrain the generated molecules in a specific domain y as follows:

$$\nabla_{\boldsymbol{x}_t} \log P(\boldsymbol{x}_t|y) = \nabla_{\boldsymbol{x}_t} \log P(\boldsymbol{x}_t) + \nabla_{\boldsymbol{x}_t} \log P(y|\boldsymbol{x}_t). \tag{18}$$

We consider the validity of generated molecules from two aspects: (a) the arms and scaffold should be connected to form a complete molecule; (b) there should not be a clash between the generated molecule and protein surface. We will describe the guidance design from these two aspects detailedly in the following and leave related derivation in Appendix D.

We assert the existence of connections between all arms and scaffold if the following inequality holds for $n = 1 : |\mathcal{A}|$,

$$\rho_{\min} \leq \min_{i \in \mathcal{A}_n, j \in \mathcal{S}} \|\boldsymbol{x}^{(i)} - \boldsymbol{x}^{(j)}\|_2 \leq \rho_{\max}, \tag{19}$$

where $\rho_{\min}$ and $\rho_{\max}$ are hyperparameters approximately representing the range of a bond length and set to 1.2Å and 1.9Å respectively in practice. The arms-scaffold drift can be derived as follows:

$$-\nabla_{\boldsymbol{x}_t} \sum_{n=1}^{|\mathcal{A}|} [\xi_2 \max(0, d_t^{(n)} - \rho_{\max}) + \xi_1 \max(0, \rho_{\min} - d_t^{(n)})], \tag{20}$$

where $d_t^{(n)} = \min_{i \in \mathcal{A}_n, j \in \mathcal{S}} \|\boldsymbol{x}_t^{(i)} - \boldsymbol{x}_t^{(j)}\|_2$ and $\xi_1, \xi_2 > 0$ are constant coefficients that control the strength of drift.

Too short distances or clashes between the atoms of generated molecules and proteins will induce abnormal Van der Waals forces. Following Sverrisson et al. (2021); Ganea et al. (2021), we choose $\{\boldsymbol{x} \in \mathbb{R}^3 : S(\boldsymbol{x}) = \gamma\}$ where $S(\boldsymbol{x}) = -\sigma \ln(\sum_{j=1}^{N_P} \exp(-\|\boldsymbol{x} - \boldsymbol{x}_P^{(j)}\|^2/\sigma))$ as the descriptor of the protein surface. Recall that $\{\boldsymbol{x}_P^{(j)}\}_{j=1}^{N_P}$ represents the set of positions of protein atoms. The clash drift can be derived as follows:

$$-\nabla_{\boldsymbol{x}_t} \xi_3 \sum_{i}^{N_M} \max(0, \gamma - S(\boldsymbol{x}_t^{(i)})), \tag{21}$$

where $\xi_3 > 0$ is the constant coefficient that controls the strength of drift.

4 Experiments

4.1 Experimental Setup

Dataset Following the previous work (Luo et al., 2021; Peng et al., 2022), we trained our model on the CrossDocked2020 dataset (Francoeur et al., 2020). We use the same dataset preprocessing and splitting procedure as Luo et al. (2021), where the 22.5 million docked binding complexes are first refined to only keep high-quality docking poses (RMSD between the docked pose and the ground truth < 1Å) and diverse proteins (sequence identity $< 30\%$), and then $100,000$ complexes are selected for training and 100 novel proteins are selected as references for testing.

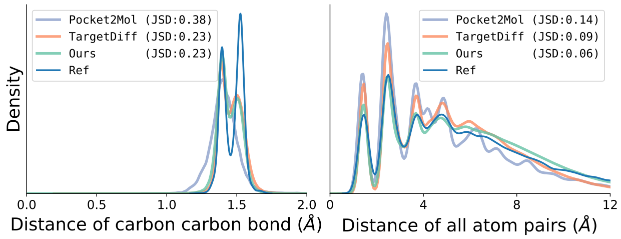

Figure 3. Comparing the distribution for distances of carbon-carbon pairs (left) and all-atom (right) for reference molecules in the test set and model-generated molecules. Jensen-Shannon divergence (JSD) between two distributions is reported.

Table 1. Jensen-Shannon divergence between bond distance distributions of the reference molecules and the generated molecules, and lower values indicate better performances. "-", "=", and ":" represent single, double, and aromatic bonds, respectively. We highlight the best two results with **bold text** and underlined text, respectively.

Bond	liGAN	GraphBP	AR	Pocket2 Mol	Target Diff	Ours
C−C	0.601	0.368	0.609	0.496	0.369	**0.359**
C=C	0.665	0.530	0.620	0.561	**0.505**	0.537
C−N	0.634	0.456	0.474	0.416	0.363	**0.344**
C=N	0.749	0.693	0.635	0.629	**0.550**	0.584
C−O	0.656	0.467	0.492	0.454	0.421	**0.376**
C=O	0.661	0.471	0.558	0.516	0.461	**0.374**
C:C	0.497	0.407	0.451	0.416	0.263	**0.251**
C:N	0.638	0.689	0.552	0.487	**0.235**	0.269

Baselines We compare our model with various representative baselines: **liGAN** (Ragoza et al., 2022a) is a conditional VAE model which uses 3D CNN to encode and generate voxelized atomic density. **AR** (Luo et al., 2021), **Pocket2Mol** (Peng et al., 2022) and **GraphBP** (Liu et al., 2022b) are GNN-based methods that generate 3D molecules atom by atom in an autoregressive manner. **TargetDiff** (Guan et al., 2023) is a diffusion-based method which generates atom coordinates and atom types in a non-autoregressive way, but the prior distribution is a standard Gaussian and bonds are generated with a post-processing algorithm.

Evaluation We evaluate the generated molecules from two perspectives: **molecular conformation** and **target binding affinity and molecular properties**. In terms of molecular conformation, we compute the Jensen-Shannon divergences (JSD) in atom/bond distance distributions between the reference molecules and the generated molecules. We employ AutoDock Vina (Eberhardt et al., 2021) to estimate the target binding affinity, following the same setup as Luo et al. (2021); Ragoza et al. (2022a). We collect all generated molecules across 100 test proteins and report

Table 2. Jensen-Shannon divergence between bond distance distributions of the reference molecules and the generated molecules, and lower values indicate better performances. We highlight the best two results with **bold text** and underlined text, respectively.

Angle	liGAN	GraphBP	AR	Pocket2Mol	TargetDiff	Ours
CCC	0.598	0.424	0.340	0.323	0.328	**0.314**
CCO	0.637	0.354	0.442	0.401	0.385	**0.324**
CNC	0.604	0.469	0.419	**0.237**	0.367	0.297
OPO	0.512	0.684	0.367	0.274	0.303	**0.217**
NCC	0.621	0.372	0.392	0.351	0.354	**0.294**
CC=O	0.636	0.377	0.476	0.353	0.356	**0.259**
COC	0.606	0.482	0.459	**0.317**	0.389	0.339

the mean and median of affinity-related metrics (*Vina Score*, *Vina Min*, *Vina Dock*, and *High Affinity*) and property-related metrics (drug-likeness *QED* (Bickerton et al., 2012), synthesizability *SA* (Ertl & Schuffenhauer, 2009), and *diversity*). Vina Score directly estimates the binding affinity based on the generated 3D molecules, Vina Min conducts a local structure minimization before estimation, Vina Dock involves a re-docking process and reflects the best possible binding affinity, and High Affinity measures the percentage of how many generated molecules binds better than the reference molecule per test protein. Following Yang et al. (2021); Long et al. (2022), we further report the percentage of molecules which pass certain criteria (QED > 0.25, SA > 0.59, Vina Dock < -8.18) as *Success Rate* to comprehensively evaluate the target binding affinity and the molecular properties, given the fact that practical drug design also requires the generated molecules to be drug-like, synthesizable, and maintain high binding affinity simultaneously (Jin et al., 2020; Xie et al., 2021). The thresholds used for QED and SA are computed as the 10th percentile of molecules in the DrugCentral database (Ursu et al., 2016), which are all pharmaceutical or under clinical trials.

4.2 Main Results

First, we compare our model and the representative methods in terms of molecular conformation. We compute different bond distance and bond angle distributions of the generated molecules and compare them against the corresponding reference empirical distributions in Tables 1 and 2, and plot the carbon-carbon bond distance distribution and all-atom pairwise distance distribution of the generated molecules in Figure 3. We see in Figure 3 that DecompDiff achieves the lowest JSD of 0.23 and 0.06 to reference in the carbon-carbon bond distance distribution and all-atom pairwise distance distribution of the generated molecules, indicating it captures real atomic distances well. Such performance is better than Pocket2Mol and TargetDiff, two strong baselines, while Pocket2Mol and TargetDiff have a biased estimation for carbon-carbon bond distribution and the long-range (especially $10 - 12$ Å) atomic distance. A similar observation can also be found in Tables 1 and 2. Our model has a comparable performance with TargetDiff and is better than all other baselines by a clear margin, showing the strong potential of our proposed method for generating stable molecular conformations directly.

Then we evaluate the effectiveness of our model in terms of binding affinity and molecular properties. We can see in Table 3 that our DecompDiff outperforms baselines by a large margin in affinity-related metrics. Specifically, DecompDiff surpasses the strong baseline TargetDiff by 6% and 12% in Avg. and Med. High Affinity, and around 0.40 in Vina Min and Vina Dock. All these gains clearly indicate that our decomposed priors are helpful for improving the potential of diffusion models for generating molecules with better target binding affinity.

We also see there is a trade-off between the property-related metrics (QED, SA) and affinity-related metrics. Like TargetDiff, our model falls behind the SOTA autoregressive model Pocket2Mol in QED and SA scores. Nevertheless, it is worth mentioning that such property-related metrics are often applied as rough screening metrics in real drug discovery scenarios as long as they fall into a reasonable range. Instead of directly comparing them in numerical value, a recommended evaluation strategy is using the Success Rate (Jin et al., 2020; Xie et al., 2021) to reflect the molecular properties comprehensively. In such a context, our DecompDiff achieves a 24.5% Success Rate, which is comparable to Pocket2Mol and clearly outperforms TargetDiff. We also show some visualization results in Appendix G to compare TargetDiff and DecompDiff.

4.3 Ablation Studies

Since our model is composed of multiple novel designs, including decomposed priors, bond diffusion and additional guidance, we perform comprehensive ablation studies to verify our hypothesis on the effects of each design.

Effect of Decomposition and Prior Our primary hypothesis is that decomposing the drug space with prior knowledge can improve the training and sampling efficiency, and thus boosting the molecular generation performance. To verify it, we compare our model with TargetDiff (Guan et al., 2023) under different number of diffusion steps. We first train our model and TargetDiff with $\{200, 400, 600, 800, 1000\}$ diffusion steps and evaluate the generated molecules by sampling the same number of steps as training on 16 selected pockets with clear decomposition. As shown in Figure 4, our model can achieve better validation loss under each setting. Under the same setting, the validation loss can be viewed as a surrogate of negative Evidence Lower Bound (ELBO) and lower validation loss means the model can better approximate the data distribution. The fact that the model trained with fewer diffusion steps achieves lower val-

Table 3. Summary of different properties of reference molecules and molecules generated by our model and other baselines. (↑) / (↓) denotes a larger / smaller number is better. Top 2 results are highlighted with **bold text** and underlined text, respectively.

Methods	Vina Score (↓)		Vina Min (↓)		Vina Dock (↓)		High Affinity (↑)		QED (↑)		SA (↑)		Diversity (↑)		Success Rate (↑)
	Avg.	Med.	Avg.	Med.	Avg.	Med.	Avg.	Med.	Avg.	Med.	Avg.	Med.	Avg.	Med.	Avg.
Reference	-6.36	-6.46	-6.71	-6.49	-7.45	-7.26	-	-	0.48	0.47	0.73	0.74	-	-	25.0%
liGAN	-	-	-	-	-6.33	-6.20	21.1%	11.1%	0.39	0.39	0.59	0.57	0.66	0.67	3.9%
GraphBP	-	-	-	-	-4.80	-4.70	14.2%	6.7%	0.43	0.45	0.49	0.48	**0.79**	**0.78**	0.1%
AR	**-5.75**	-5.64	-6.18	-5.88	-6.75	-6.62	37.9%	31.0%	0.51	0.50	0.63	0.63	0.70	0.70	7.1%
Pocket2Mol	-5.14	-4.70	-6.42	-5.82	-7.15	-6.79	48.4%	51.0%	**0.56**	**0.57**	**0.74**	**0.75**	0.69	0.71	24.4%
TargetDiff	-5.47	**-6.30**	-6.64	-6.83	-7.80	-7.91	58.1%	59.1%	0.48	0.48	0.58	0.58	0.72	0.71	10.5%
DECOMPDIFF	-5.67	-6.04	**-7.04**	**-7.09**	**-8.39**	**-8.43**	**64.4%**	**71.0%**	0.45	0.43	0.61	0.60	0.68	0.68	**24.5%**

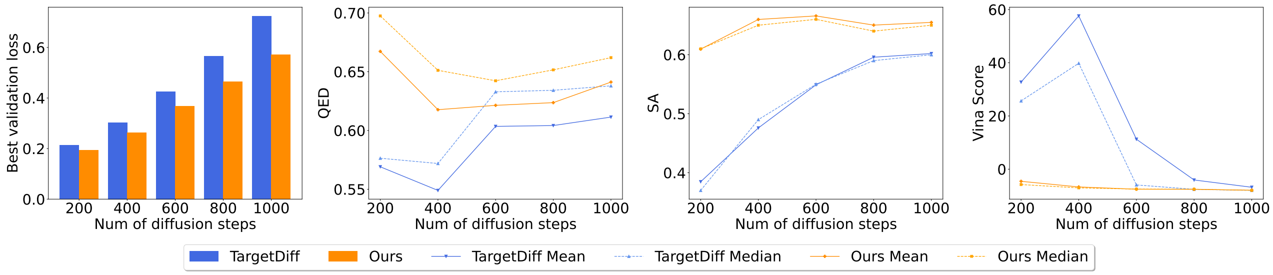

Figure 4. Ablation study on diffusion step number. We compare our models with TargetDiff (Guan et al., 2023) in terms of best validation loss, QED, SA, Vina Score under different diffusion step number settings.

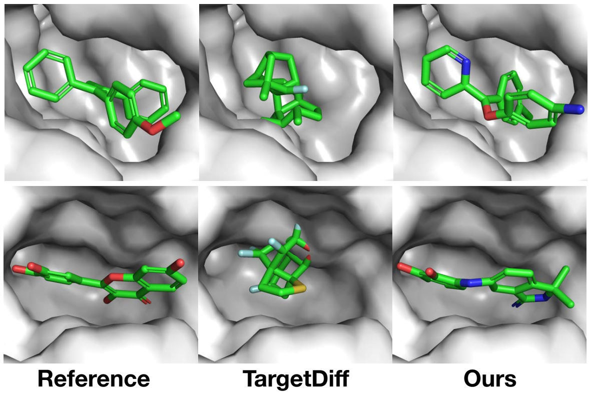

Figure 5. Visualization of reference binding molecules (left column), molecules generated by TargetDiff (Guan et al., 2023) (middle column), and our model (right column) with only 200 sampling steps on protein 4H3C (top row) and 2F2C (bottom row).

idation loss is because it fits noises better at fewer time steps with limited model capacity. Figure 4 also shows our model can generate high-quality ligand molecules (high QED and SA, low Vina) even with fewer sampling steps. Figure 5 shows examples of ligand molecules generated by sampling only 200 steps. With limited sampling steps, our model can already generate rational molecules, while TargetDiff tends to generate unrealistic local structures, such as messy rings.

Besides the training/sampling efficiency analysis, we also investigate the influence of different priors. We mainly explore two kinds of priors: *Ref Prior* is estimated from the reference molecule with a Gaussian distribution through maximum likelihood estimation. *Pocket Prior* utilizes the subpockets within the target binding site extracted by AlphaSpace2 (Rooklin et al., 2015) to estimate the prior center and a neural classifier to estimate the number of ligand atoms and prior standard deviation. See more details in Appendix A. In Table 4, we show the affinity-related metrics of an atom-only version (no bond diffusion) of our model with different priors. Compared with TargetDiff, a baseline without decomposition and informative prior, our model can generate better binding molecules with appropriate prior knowledge. Molecules generated using *Pocket Prior* achieve better results compared to *Ref Prior*, which could be caused by the fact that the reference molecule may not be ideal for the target and directly extract knowledge prior from the target binding site could be a better choice.

Table 4. The influence of decomposed prior.

Methods	Vina Score (↓)		Vina Min (↓)		Vina Dock (↓)	
	Avg.	Med.	Avg.	Med.	Avg.	Med.
TargetDiff	-5.47	-6.30	-6.64	-6.83	-7.80	-7.91
Ours - Ref Prior	**-6.11**	-6.16	-6.68	-6.58	-7.53	-7.59
Ours - Pocket Prior	-5.72	**-7.49**	**-7.66**	**-8.33**	**-9.08**	**-9.33**

Effect of Bond Diffusion Our motivation to include bond generation in the model is to reduce the ratio of unrealistic

Table 5. The influence of bond diffusion.

Methods	Vina Dock (↓)		QED (↑)		SA (↑)		Success (↑)
	Avg.	Med.	Avg.	Med.	Avg.	Med.	Avg.
Ours - Atom	**-7.53**	**-7.59**	0.48	0.48	0.61	0.60	11.21%
Ours - Bond	-7.10	-7.14	**0.51**	**0.51**	**0.66**	**0.65**	**15.38%**

Table 6. The influence of validity guidance in sampling.

Methods	Complete (↑)		Vina Score (↓)		Vina Min (↓)	
	Avg.	Med.	Avg.	Med.	Avg.	Med.
TargetDiff - Ref	0.91	0.96	-5.32	-5.99	-6.42	-6.51
Ours - No Drift	0.89	0.95	-4.75	-5.92	-6.17	-6.47
Ours - ArmSca Drift	0.94	0.98	-4.84	-5.99	-6.20	-6.50
Ours - Clash Drift	0.87	0.94	-5.79	-6.00	-6.58	-6.54
Ours - All Drift	**0.94**	**0.98**	**-6.11**	**-6.16**	**-6.68**	**-6.58**

2D structures caused by inaccurate bond prediction of the post-processing algorithm. We see in Table 5 that bond diffusion substantially improves QED, SA, and Success Rate, indicating simultaneously modeling atoms and bonds can generate more reasonable 2D structures.

Effect of Validity Guidance To show the effectiveness of validity guidance proposed in Sec. 3.4, we test different guidance drifts during the sampling phase in our decomposed atom diffusion model. As shown in Table 6, the arm-scaffold connection drift and the clash drift have a positive influence on the complete rate and Vina score respectively. Combining them (All Drift) achieves even better performance in both complete rate and Vina score.

5 Conclusions

In this work, we proposed DECOMPDIFF for SBDD by decomposing the drug space. The introduction of informative priors to the diffusion model improves the training and sampling efficiency and significantly improves the binding affinity measured by Vina. We also extend the diffusion model to simultaneously generate atom coordinates, atom types, and bond types, which is beneficial to generate more realistic molecules. Validity guidance during sampling is simple but effective under our decomposition framework.

Acknowledgements

We thank all the reviewers for their feedbacks through out the review cycles of the manuscript. This work is supported by National Natural Science Foundation of China (62206291, 62141608, 62236010), National Key R&D Program of China No. 2021YFF1201600, U.S. National Science Foundation under grant no. 2019897 and U.S. Department of Energy award DE-SC0018420.

References

Adams, K. and Coley, C. W. Equivariant shape-conditioned generation of 3d molecules for ligand-based drug design. *arXiv preprint arXiv:2210.04893*, 2022.

Anderson, A. C. The process of structure-based drug design. *Chemistry & biology*, 10(9):787–797, 2003.

Arús-Pous, J., Patronov, A., Bjerrum, E. J., Tyrchan, C., Reymond, J.-L., Chen, H., and Engkvist, O. Smiles-based deep generative scaffold decorator for de-novo drug design. *Journal of cheminformatics*, 12(1):1–18, 2020.

Berman, H. M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T. N., Weissig, H., Shindyalov, I. N., and Bourne, P. E. The protein data bank. *Nucleic acids research*, 28(1): 235–242, 2000.

Bickerton, G. R., Paolini, G. V., Besnard, J., Muresan, S., and Hopkins, A. L. Quantifying the chemical beauty of drugs. *Nature chemistry*, 4(2):90–98, 2012.

Degen, J., Wegscheid-Gerlach, C., Zaliani, A., and Rarey, M. On the art of compiling and using'drug-like'chemical fragment spaces. *ChemMedChem: Chemistry Enabling Drug Discovery*, 3(10):1503–1507, 2008.

Dhariwal, P. and Nichol, A. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34:8780–8794, 2021.

Eberhardt, J., Santos-Martins, D., Tillack, A. F., and Forli, S. Autodock vina 1.2. 0: New docking methods, expanded force field, and python bindings. *Journal of Chemical Information and Modeling*, 61(8):3891–3898, 2021.

Ertl, P. and Schuffenhauer, A. Estimation of synthetic accessibility score of drug-like molecules based on molecular complexity and fragment contributions. *Journal of cheminformatics*, 1(1):1–11, 2009.

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, 2020.

Fuchs, F., Worrall, D., Fischer, V., and Welling, M. Se (3)-transformers: 3d roto-translation equivariant attention networks. *Advances in Neural Information Processing Systems*, 33:1970–1981, 2020.

Ganea, O.-E., Huang, X., Bunne, C., Bian, Y., Barzilay, R., Jaakkola, T., and Krause, A. Independent se (3)-equivariant models for end-to-end rigid protein docking. *arXiv preprint arXiv:2111.07786*, 2021.

Gasteiger, J., Groß, J., and Günnemann, S. Directional message passing for molecular graphs. *arXiv preprint arXiv:2003.03123*, 2020.

Guan, J., Qian, W. W., Liu, Q., Ma, W.-Y., Ma, J., and Peng, J. Energy-inspired molecular conformation optimization. In *International Conference on Learning Representations*, 2022.

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. *arXiv preprint arXiv:2303.03543*, 2023.

Ho, J. and Salimans, T. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

Ho, J., Jain, A., and Abbeel, P. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.

Hoogeboom, E., Nielsen, D., Jaini, P., Forré, P., and Welling, M. Argmax flows and multinomial diffusion: Learning categorical distributions. *Advances in Neural Information Processing Systems*, 34:12454–12465, 2021.

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.

Huang, Y., Peng, X., Ma, J., and Zhang, M. 3dlinker: An e (3) equivariant variational autoencoder for molecular linker design. *arXiv preprint arXiv:2205.07309*, 2022.

Igashov, I., Stärk, H., Vignac, C., Satorras, V. G., Frossard, P., Welling, M., Bronstein, M., and Correia, B. Equivariant 3d-conditional diffusion models for molecular linker design. *arXiv preprint arXiv:2210.05274*, 2022.

Imrie, F., Bradley, A. R., van der Schaar, M., and Deane, C. M. Deep generative models for 3d linker design. *Journal of chemical information and modeling*, 60(4):1983–1995, 2020.

Imrie, F., Hadfield, T. E., Bradley, A. R., and Deane, C. M. Deep generative design with 3d pharmacophoric constraints. *Chemical science*, 12(43):14577–14589, 2021.

Jin, W., Barzilay, R., and Jaakkola, T. Multi-objective molecule generation using interpretable substructures. In *International conference on machine learning*, pp. 4849–4859. PMLR, 2020.

Katigbak, J., Li, H., Rooklin, D., and Zhang, Y. Alphaspace 2.0: Representing concave biomolecular surfaces using β-clusters. *Journal of chemical information and modeling*, 60(3):1494–1508, 2020.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Kong, Z., Ping, W., Huang, J., Zhao, K., and Catanzaro, B. Diffwave: A versatile diffusion model for audio synthesis. In *International Conference on Learning Representations*, 2021.

Lee, S.-g., Kim, H., Shin, C., Tan, X., Liu, C., Meng, Q., Qin, T., Chen, W., Yoon, S., and Liu, T.-Y. Priorgrad: Improving conditional denoising diffusion models with data-driven adaptive prior. *arXiv preprint arXiv:2106.06406*, 2021.

Li, X. L., Thickstun, J., Gulrajani, I., Liang, P., and Hashimoto, T. B. Diffusion-lm improves controllable text generation. *arXiv preprint arXiv:2205.14217*, 2022.

Li, Y., Hu, J., Wang, Y., Zhou, J., Zhang, L., and Liu, Z. Deepscaffold: a comprehensive tool for scaffold-based de novo drug discovery using deep learning. *Journal of chemical information and modeling*, 60(1):77–91, 2019.

Li, Y., Pei, J., and Lai, L. Structure-based de novo drug design using 3d deep generative models. *Chemical science*, 12(41):13664–13675, 2021.

Lim, J., Hwang, S.-Y., Moon, S., Kim, S., and Kim, W. Y. Scaffold-based molecular design with a graph generative model. *Chemical science*, 11(4):1153–1164, 2020.

Lin, H., Huang, Y., Liu, M., Li, X., Ji, S., and Li, S. Z. Diffbp: Generative diffusion of 3d molecules for target protein binding. *arXiv preprint arXiv:2211.11214*, 2022.

Liu, M., Luo, Y., Uchino, K., Maruhashi, K., and Ji, S. Generating 3d molecules for target protein binding. *arXiv preprint arXiv:2204.09410*, 2022a.

Liu, M., Luo, Y., Uchino, K., Maruhashi, K., and Ji, S. Generating 3d molecules for target protein binding. In *International Conference on Machine Learning*, 2022b.

Long, S., Zhou, Y., Dai, X., and Zhou, H. Zero-shot 3d drug design by sketching and generating. *arXiv preprint arXiv:2209.13865*, 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.

Luo, Y. and Ji, S. An autoregressive flow model for 3d molecular geometry generation from scratch. In *International Conference on Learning Representations*, 2021.

Nichol, A., Dhariwal, P., Ramesh, A., Shyam, P., Mishkin, P., McGrew, B., Sutskever, I., and Chen, M. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741*, 2021.

Nichol, A. Q. and Dhariwal, P. Improved denoising diffusion probabilistic models. In *International Conference on Machine Learning*, pp. 8162–8171. PMLR, 2021.

O'Boyle, N. M., Banck, M., James, C. A., Morley, C., Vandermeersch, T., and Hutchison, G. R. Open babel: An open chemical toolbox. *Journal of cheminformatics*, 3(1): 1–14, 2011.

Peng, X., Luo, S., Guan, J., Xie, Q., Peng, J., and Ma, J. Pocket2mol: Efficient molecular sampling based on 3d protein pockets. *arXiv preprint arXiv:2205.07249*, 2022.

Ragoza, M., Masuda, T., and Koes, D. R. Generating 3D molecules conditional on receptor binding sites with deep generative models. *Chem Sci*, 13:2701–2713, Feb 2022a. doi: 10.1039/D1SC05976A.

Ragoza, M., Masuda, T., and Koes, D. R. Generating 3d molecules conditional on receptor binding sites with deep generative models. *Chemical science*, 13(9):2701–2713, 2022b.

Ramesh, A., Dhariwal, P., Nichol, A., Chu, C., and Chen, M. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022.

Rooklin, D., Wang, C., Katigbak, J., Arora, P. S., and Zhang, Y. Alphaspace: fragment-centric topographical mapping to target protein–protein interaction interfaces. *Journal of chemical information and modeling*, 55(8):1585–1599, 2015.

Satorras, V. G., Hoogeboom, E., and Welling, M. E (n) equivariant graph neural networks. In *International Conference on Machine Learning*. PMLR, 2021.

Schneider, G., Neidhart, W., Giller, T., and Schmid, G. "scaffold-hopping" by topological pharmacophore search: a contribution to virtual screening. *Angewandte Chemie International Edition*, 38(19):2894–2896, 1999.

Schneuing, A., Du, Y., Harris, C., Jamasb, A., Igashov, I., Du, W., Blundell, T., Lió, P., Gomes, C., Welling, M., et al. Structure-based drug design with equivariant diffusion models. *arXiv preprint arXiv:2210.13695*, 2022.

Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N., and Ganguli, S. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, pp. 2256–2265. PMLR, 2015.

Song, Y. and Ermon, S. Generative modeling by estimating gradients of the data distribution. *Advances in Neural Information Processing Systems*, 32, 2019.

Sverrisson, F., Feydy, J., Correia, B. E., and Bronstein, M. M. Fast end-to-end learning on protein surfaces. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15272–15281, 2021.

Thomas, N., Smidt, T., Kearnes, S., Yang, L., Li, L., Kohlhoff, K., and Riley, P. Tensor field networks: Rotation-and translation-equivariant neural networks for 3d point clouds. *arXiv preprint arXiv:1802.08219*, 2018.

Ursu, O., Holmes, J., Knockel, J., Bologa, C. G., Yang, J. J., Mathias, S. L., Nelson, S. J., and Oprea, T. I. Drugcentral: online drug compendium. *Nucleic acids research*, pp. gkw993, 2016.

Vignac, C., Krawczuk, I., Siraudin, A., Wang, B., Cevher, V., and Frossard, P. Digress: Discrete denoising diffusion for graph generation. *arXiv preprint arXiv:2209.14734*, 2022.

Virshup, A. M., Contreras-García, J., Wipf, P., Yang, W., and Beratan, D. N. Stochastic voyages into uncharted chemical space produce a representative library of all possible drug-like compounds. *Journal of the American Chemical Society*, 135(19):7296–7303, 2013.

Wermuth, C. G. *The practice of medicinal chemistry*. Academic Press, 2011.

Xie, Y., Shi, C., Zhou, H., Yang, Y., Zhang, W., Yu, Y., and Li, L. Mars: Markov molecular sampling for multi-objective drug discovery. *arXiv preprint arXiv:2103.10432*, 2021.

Xu, M., Yu, L., Song, Y., Shi, C., Ermon, S., and Tang, J. Geodiff: A geometric diffusion model for molecular conformation generation. *arXiv preprint arXiv:2203.02923*, 2022.

Yang, K., Swanson, K., Jin, W., Coley, C., Eiden, P., Gao, H., Guzman-Perez, A., Hopper, T., Kelley, B., Mathea, M., et al. Analyzing learned molecular representations for property prediction. *Journal of chemical information and modeling*, 59(8):3370–3388, 2019.

Yang, Y., Zheng, S., Su, S., Zhao, C., Xu, J., and Chen, H. Syntalinker: automatic fragment linking with deep conditional transformer neural networks. *Chemical science*, 11 (31):8312–8322, 2020.

Yang, Y., Ouyang, S., Dang, M., Zheng, M., Li, L., and Zhou, H. Knowledge guided geometric editing for unsupervised drug design. 2021. URL `https://openreview.net/forum?id=91muTwt1_t5`.

A The Generation Details of Pocket Prior Centers

Given protein pockets, the arms are the main components of a ligand interacting with pockets, and the scaffold connects the arms to form a ligand. We obtain the prior centers of arms and scaffolds of ligands in the following steps:

1. **Searching alpha-atoms and beta-atoms**: Following Rooklin et al. (2015); Katigbak et al. (2020), we search the alpha-atoms and beta-atoms by `AlphaSpace2`[1] toolkit. An alpha-atom can be viewed as a virtual point, geometrically contacted with the local region of the protein surface (see Figure 6(a)). And a beta-atom is an alpha-atom group and assigned a beta-score to indicate their pocket ligandability (see Figure 6(b)). Notice that we only consider alpha-atoms and beta-atoms within 10Å of the reference ligand or the interested region.
2. **Clustering beta-atoms to beta-clusters**: We apply hierarchical clustering on beta-atoms to further group them into beta-clusters, based on the pairwise distances, as shown in Figure 6(d). A pair of beta-atoms whose distance is less than a certain distance (in practice, we use 5.5Å) are assigned to the same beta-cluster. The score of the beta-cluster is defined as the average beta-scores of the beta-atoms that belong to it.
3. **Assigning beta-clusters into arms and scaffolds**: We filter beta-clusters and determine the scaffold and arm priors with Algorithm 1. Specifically, we first find the region contains a proper number of beta-clusters with promising beta-scores. We then select centers of beta-clusters, according to their contribution to binding affinity and the positions holding scaffold-arm geometric connection, as centers of arm and scaffold priors, respectively.

Note that the distance for filtering beta-clusters is not Euclidean in 3D space. Instead, we define the distance by introducing a radius graph (called alpha-graph, see Figure 6(c)) where each alpha-atom is connected to the other alpha-atoms and beta-cluster centers. Then, we use the graph geodesic on the alpha-graph between two beta-cluster centers to define their distance (If two beta-cluster centers are unconnected, the distance is defined as $+\infty$).

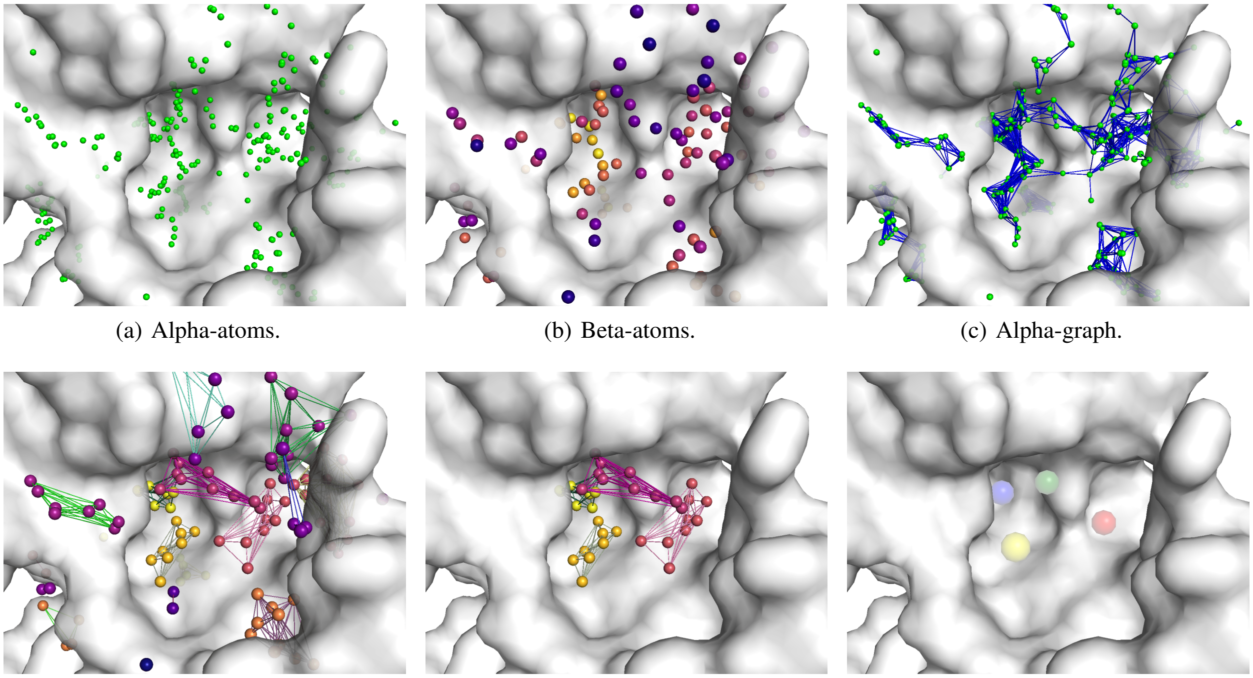
(a) Alpha-atoms.
(b) Beta-atoms.
(c) Alpha-graph.
(d) Beta-clusters.
(e) Filtered beta-clusters.
(f) Priors.

Figure 6. Illustration of prior centers generation process. (a) Alpha-atoms. (b) Beta-atoms with beta-scores. The closer the hue of a beta-atom to yellow, the better its beta-score is. (c) Alpha-graph. Here we only visualize the alpha-atoms and omit beta-cluster centers for clarity. (d) Beta-clusters. Each beta-cluster forms a fully-connected graph consisting of its beta-atoms. All beta-atoms in a beta-cluster are visualized in the same color, corresponding to the score. (e) Filtered beta-clusters, and corresponding (f) Priors. The scaffold prior is shown in yellow, and the arm priors are shown in other colors.

[1] `https://yzhang.hpc.nyu.edu/AlphaSpace2/`

Algorithm 1 Procedure of filtering beta-clusters and determining scaffold and arm priors.

```
Input: a list of beta-clusters clusterList, scores of the beta-clusters in the list score, maximum number of arms $N$, maximum distance between arms $\delta$, maximum distance between the scaffold and arms $\sigma$, and minimum size of interested connected components $M$.
Output: list of beta-clusters armClusterList which correspond to the arm priors, and beta-cluster scaffoldCluster which corresponds to the scaffold prior.
Compute pairwise distances of centers of beta-clusters in clusterList, and build a graph, in which two clusters are connected if their distance is less than $\delta$.
Remove all clusters with connected components constituting less than $M$ nodes from clusterList.
Generate a set of clusters tempClusterList with distances less than $\delta$ to the best cluster (the cluster with the highest score).
Use the first $N$ clusters in tempClusterList as arm priors armClusterList, and the others as candidates for scaffold priors scaffoldClusterList.
Search for the best scaffold cluster bestScaffoldCluster with minimum maximum distance to others in scaffoldClusterList.
Ensure the distances between bestScaffoldCluster and clusters in armClusterList are no more than $\sigma$, otherwise assign bestScaffoldCluster = NULL.
Assign finalClusterList = armClusterList, and append bestScaffoldCluster to finalClusterList, if bestScaffoldCluster $\neq$ NULL.
if len(finalClusterList) $> 2$ then
    # When there are at least 3 clusters, search for the centric one as the scaffold prior.
    Compute the pairwise distances of centers of clusters in finalClusterList, assign scaffoldCluster as the one with minimum maximum distances to others, and assign armClusterList as the others.
else
    # When there are only 2 clusters, the centric one can not be identified.
    Assign the cluster with the better score to the only element of armClusterList, and the other one to scaffoldCluster.
end if
```

B The Details of Arms-Scaffold Fragmentation

Here, we describe the algorithm for fragmenting a binding molecule into arms and scaffold, given the target protein. Such fragmentation is used for preparing our training data.

We first extract the target protein subpockets with AlphaSpace2 (Katigbak et al., 2020), which is a surface topographical mapping tool to identify the potential protein binding sites. We set the beta atoms' clustering distance as 1.6 Å, and the pockets' clustering distance as 6.0 Å. These extracted subpockets will be used as the potential arms/scaffold clustering center. We then use BRICS (Degen et al., 2008) to decompose ligand molecules into fragments. Our goal is to tag these fragments as arms or the scaffold. To achieve this, we perform clustering on these molecular fragments based on the following procedure:

- Perform the linear sum assignment to assign terminal fragments (only one connection site with other molecular parts) to subpockets extracted by AlphaSpace2.
- Take centroids of terminal fragments and remaining subpockets (may do not exist) as the *arm* clustering centers.
- Take the centroid of the fragment which is farthest from all existing arm centers as the *scaffold* clustering center.
- Perform nearest neighbor clustering, along which we always make sure the arm fragments are terminal.

Finally, we can extract arms and scaffold based on the clustering assignment. Figure 7 provides examples of our fragmented arms and scaffolds with their corresponding original ligands and protein pockets.

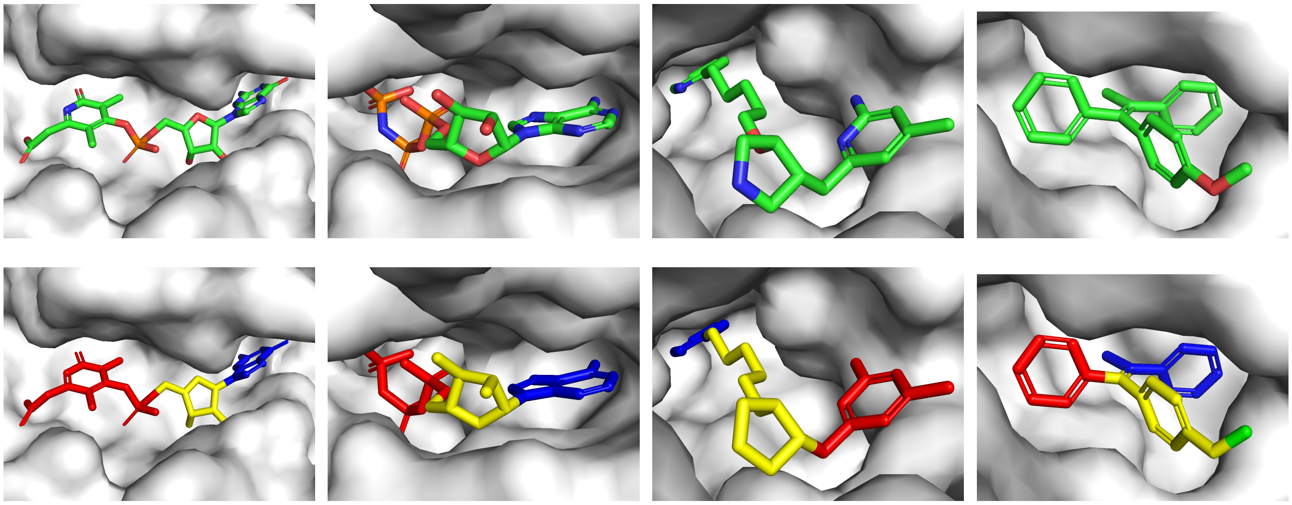

Figure 7. Examples of arms-scaffold fragmentation. Each ligand (top row) is fragmented into arms and a scaffold (bottom row). The scaffold is visualized in yellow. The arms are visualized in other colors.

C Theoretical Analysis

In this section, we will compare our decomposed prior with standard Gaussian prior and analyze its superiority theoretically. The decomposed prior manifests itself in two aspects: (a) It consists of multiple Gaussian distributions with different means rather than a single one; (b) The covariance matrix of each Gaussian distribution does have not to be a unit one. For simplicity, we will show the superiority of the decomposed prior theoretically from these two aspects respectively.

Recall that we denote $\tilde{\mathbf{x}}_{t,k}^{(i)} = \mathbf{x}_t^{(i)} - \boldsymbol{\mu}_k$. Additionally, for clarity in the proof, we denote $\tilde{\mathbf{x}}_t^{(i)} = \mathbf{x}_t^{(i)} - \boldsymbol{\mu}_k = \mathbf{x}_t^{(i)} - \boldsymbol{\mu}^{(i)}$ where k satisfies $\eta_{ik} = 1$. Similarly, for the standard prior, we denote $\breve{\mathbf{x}}_t^{(i)} = \mathbf{x}_t^{(i)} - \boldsymbol{\mu}$ here. $\mu_d^{(i)}$ and μ_d are the dth scalar element in $\boldsymbol{\mu}^{(i)}$ and $\boldsymbol{\mu}$ respectively.

We first show why the decomposed prior can approximate the data distribution better in aspect (a). Respecting the natural decomposition of a binding molecule, we make an assumption that $\mathbb{E}[\mathbf{x}_0^{(i)}] = \boldsymbol{\mu}_k \in \mathbb{R}^D, \mathbb{E}[\tilde{\mathbf{x}}_0^{(i)}\tilde{\mathbf{x}}_0^{(i)\top}] = \boldsymbol{I} \in \mathbb{R}^{D\times D}$ if $\eta_{ik} = 1$ and $\mathbb{E}[\tilde{\mathbf{x}}_0^{(j)}\tilde{\mathbf{x}}_0^{(i)\top}] = \mathbf{0}, \forall i \neq j$. Besides, we assume that the score network is a simple graph neural network with only one layer of linear transformation and summation as an aggregation function, i.e., $\hat{\mathbf{x}}_{0,t}^{(i)} = f_{\boldsymbol{\theta}}(\mathbf{x}_t, t) = \sum_{j\neq i}\theta(\mathbf{x}_t^{(j)} - \mathbf{x}_t^{(i)}) + \mathbf{x}_t^{(i)}$ and $\theta = \mathrm{diag}(\theta_1, \theta_2, \cdots, \theta_D) \in \mathbb{R}^{D\times D}$ are learnable parameters constraint to a diagonal matrix with freedom D. The diffusion process of the decomposed and standard cases are shown respectively as follows:

$$\tilde{\mathbf{x}}_t^{(i)} = \sqrt{\bar{\alpha}_t}\tilde{\mathbf{x}}_0^{(i)} + \sqrt{1-\bar{\alpha}_t}\epsilon \quad \text{where} \quad \mathbf{x}_t^{(i)} = \tilde{\mathbf{x}}_t^{(i)} + \boldsymbol{\mu}^{(i)} \tag{22}$$

$$\breve{\mathbf{x}}_t^{(i)} = \sqrt{\bar{\alpha}_t}\breve{\mathbf{x}}_0^{(i)} + \sqrt{1-\bar{\alpha}_t}\epsilon \quad \text{where} \quad \mathbf{x}_t^{(i)} = \breve{\mathbf{x}}_t^{(i)} + \boldsymbol{\mu} \tag{23}$$

where $\epsilon \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$.

Under the above assumption, the negative evidence lower bound (ELBO) based on the decomposed prior can be derived as:

$$\begin{aligned} -\text{ELBO}_{\text{decomp}} &= \mathbb{E}_{\mathbf{x}_0,\epsilon}\left[\sum_i\sum_k \eta_{ik}\left[\frac{\bar{\alpha}_T}{2}\left\|\tilde{\mathbf{x}}_{0,k}^{(i)}\right\|^2 + \sum_{t=1}^{T-1}\gamma_t\left\|\mathbf{x}_{0,k}^{(i)} - \hat{\mathbf{x}}_{0,t}^{(i)}\right\|^2\right]\right] + C \\ &= \mathbb{E}_{\mathbf{x}_0,\epsilon}\left[\sum_k\sum_{i:\eta_{ik}=1}\left[\frac{\bar{\alpha}_T}{2}\left\|\mathbf{x}_0^{(i)} - \boldsymbol{\mu}^{(i)}\right\|^2 + \sum_{t=1}^{T-1}\gamma_t\left\|\mathbf{x}_0^{(i)} - \hat{\mathbf{x}}_{0,t}^{(i)}\right\|^2\right]\right] + C \end{aligned} \tag{24}$$

Similarly, the negative ELBO based on the standard Gaussian prior can be derived as:

$$
\begin{aligned}
-\mathrm{ELBO}_{\text{standard}} &= \mathbb{E}_{\mathbf{x}_0,\epsilon}\left[\sum_i\sum_k \eta_{ik}\left[\frac{\bar{\alpha}_T}{2}\left\|\breve{\mathbf{x}}_0^{(i)}\right\|^2 + \sum_{t=1}^{T-1}\gamma_t\left\|\mathbf{x}_{0,k}^{(i)} - \hat{\mathbf{x}}_{0,t}^{(i)}\right\|^2\right]\right] + C \\
&= \mathbb{E}_{\mathbf{x}_0,\epsilon}\left[\sum_k\sum_{i:\eta_{ik}=1}\left[\frac{\bar{\alpha}_T}{2}\left\|\mathbf{x}_0^{(i)} - \boldsymbol{\mu}\right\|^2 + \sum_{t=1}^{T-1}\gamma_t\left\|\mathbf{x}_0^{(i)} - \hat{\mathbf{x}}_{0,t}^{(i)}\right\|^2\right]\right] + C
\end{aligned}
\tag{25}
$$

It is obvious that $\mathbb{E}_{\mathbf{x}_0}[\|\mathbf{x}_0 - \boldsymbol{\mu}^{(i)}\|^2] \leq \mathbb{E}_{\mathbf{x}_0}[\|\mathbf{x}_0 - \boldsymbol{\mu}\|^2]$. Thus we focus on proving the minimum of the second term $\mathbb{E}_{\mathbf{x}_0,\epsilon}[\sum_k\sum_{i:\eta_{ik}=1}\sum_{t=1}\gamma_t\|\mathbf{x}_0^{(i)} - \hat{\mathbf{x}}_{0,t}^{(i)}\|^2]$ based on the decomposed priors is less than that based on the standard priors. The main difference of this term is about $\hat{\mathbf{x}}_{0,t}^{(i)} = f_{\boldsymbol{\theta}}(\mathbf{x}_t, t)$ where $\mathbf{x}_t^{(i)} = \tilde{\mathbf{x}}_t^{(i)} + \boldsymbol{\mu}^{(i)}$ in the decomposed case and $\mathbf{x}_t^{(i)} = \breve{\mathbf{x}}_t^{(i)} + \boldsymbol{\mu}$ in the standard case as shown by Equation (22) and Equation (23).

Under a reasonable assumption that $\sum_i \boldsymbol{\mu}^{(i)} = \boldsymbol{\mu} = \mathbf{0}$, this term in the decomposed and standard case can be derived as Equation (27) and Equation (26), respectively.

$$
\begin{aligned}
&\mathbb{E}_{\mathbf{x}_0,\epsilon}\left[\sum_k\sum_{i:\eta_{ik}=1}\sum_{t=1}^{T-1}\gamma_t\left\|\mathbf{x}_0^{(i)} - \hat{\mathbf{x}}_{0,t}^{(i)}\right\|^2\right] \\
=&\mathbb{E}_{\mathbf{x}_0,\epsilon}\left[\sum_i\sum_{t=1}^{T-1}\gamma_t\left\|\mathbf{x}_0^{(i)} - \left[\theta\sum_{j\neq i}(\mathbf{x}_t^{(j)} - \mathbf{x}_t^{(i)}) + \mathbf{x}_t^{(i)}\right]\right\|^2\right] \\
=&\mathbb{E}_{\mathbf{x}_0,\epsilon}\left[\sum_i\sum_{t=1}^{T-1}\gamma_t\left\|\breve{\mathbf{x}}_0^{(i)} + \boldsymbol{\mu} - \theta\sum_{j\neq i}\left[(\breve{\mathbf{x}}_t^{(j)} + \boldsymbol{\mu}) - (\breve{\mathbf{x}}_t^{(i)} + \boldsymbol{\mu})\right] - \breve{\mathbf{x}}^{(i)} - \boldsymbol{\mu}\right\|^2\right] \\
=&\mathbb{E}_{\mathbf{x}_0,\epsilon}\left[\sum_i\sum_{t=1}^{T-1}\gamma_t\left\|\breve{\mathbf{x}}_0^{(i)} - \theta\sum_{j\neq i}\left[\breve{\mathbf{x}}_t^{(j)} - \breve{\mathbf{x}}_t^{(i)}\right] - \breve{\mathbf{x}}^{(i)}\right\|^2\right] \\
=&\mathbb{E}_{\mathbf{x}_0,\epsilon}\Bigg[\sum_i\sum_{t=1}^{T-1}\gamma_t\Bigg\|\breve{\mathbf{x}}_0^{(i)} - \left[\sqrt{\bar{\alpha}_t}\theta\sum_{j\neq i}\breve{\mathbf{x}}_0^{(j)} + \sqrt{1-\bar{\alpha}_t}\theta\sum_{j\neq i}\epsilon_t^{(j)} - (N-1)\sqrt{\bar{\alpha}_t}\theta\breve{\mathbf{x}}_0^{(i)} - (N-1)\sqrt{1-\bar{\alpha}_t}\theta\epsilon_t^{(i)}\right] \\
&\qquad\qquad - \left(\sqrt{\bar{\alpha}_t}\breve{\mathbf{x}}_0^{(i)} + \sqrt{1-\bar{\alpha}_t}\epsilon_t^{(i)}\right)\Bigg\|^2\Bigg] \\
=&\mathbb{E}_{\mathbf{x}_0,\epsilon}\Bigg[\sum_i\sum_{t=1}^{T-1}\gamma_t\Bigg\|(\boldsymbol{I} + (N-1)\sqrt{\bar{\alpha}_t}\theta - \sqrt{\bar{\alpha}_t}\boldsymbol{I})\breve{\mathbf{x}}_0^{(i)} + \sqrt{1-\bar{\alpha}_t}((N-1)\theta - \boldsymbol{I})\epsilon_t^{(i)} \\
&\qquad\qquad - \sqrt{\bar{\alpha}_t}\theta\sum_{j\neq i}\breve{\mathbf{x}}_0^{(j)} - \sqrt{1-\bar{\alpha}_t}\theta\sum_{j\neq i}\epsilon_t^{(j)}\Bigg\|^2\Bigg] \\
=&\sum_{t=1}^{T-1}\gamma_t\sum_d\Bigg[\left[N^2(N-1) + {\color{blue}\sqrt{\bar{\alpha}_t}[N + (N-1)^2]\sum_i\mu_d^{(i)2}}\right]{\color{red}\theta_d^2} \\
&\qquad + \left[2N(N-1)(\sqrt{\bar{\alpha}_t} - 1) + {\color{blue}\sqrt{\bar{\alpha}_t}(1-\sqrt{\bar{\alpha}_t})(2N-1)\sum_i\mu_d^{(i)2}}\right]{\color{red}\theta_d} \\
&\qquad + \left[N(1-\sqrt{\bar{\alpha}_t})^2 + N(1-\bar{\alpha}_t) + {\color{blue}(1-\sqrt{\bar{\alpha}_t})^2\sum_i\mu_d^{(i)2}}\right]\Bigg]
\end{aligned}
\tag{26}
$$

$$
\begin{aligned}
&\mathbb{E}_{\mathbf{x}_0,\epsilon}\left[\sum_{k}\sum_{i:\eta_{ik}=1}\sum_{t=1}^{T-1}\gamma_t\left\|\mathbf{x}_0^{(i)}-\hat{\mathbf{x}}_{0,t}^{(i)}\right\|^2\right]\\
=&\mathbb{E}_{\mathbf{x}_0,\epsilon}\left[\sum_{i}\sum_{t=1}^{T-1}\gamma_t\left\|\mathbf{x}_0^{(i)}-\left[\theta\sum_{j\neq i}(\mathbf{x}_t^{(j)}-\mathbf{x}_t^{(i)})+\mathbf{x}_t^{(i)}\right]\right\|^2\right]\\
=&\mathbb{E}_{\mathbf{x}_0,\epsilon}\left[\sum_{i}\sum_{t=1}^{T-1}\gamma_t\left\|\breve{\mathbf{x}}_0^{(i)}+\boldsymbol{\mu}^{(i)}-\theta\sum_{j\neq i}\left[(\breve{\mathbf{x}}_t^{(j)}+\boldsymbol{\mu}^{(i)})-(\breve{\mathbf{x}}_t^{(i)}+\boldsymbol{\mu}^{(j)})\right]-\breve{\mathbf{x}}^{(i)}-\boldsymbol{\mu}^{(i)}\right\|^2\right]\\
=&\mathbb{E}_{\mathbf{x}_0,\epsilon}\left[\sum_{i}\sum_{t=1}^{T-1}\gamma_t\left\|\tilde{\mathbf{x}}_0^{(i)}-\theta\sum_{j\neq i}\left[\tilde{\mathbf{x}}_t^{(j)}-\tilde{\mathbf{x}}_t^{(i)}\right]-\tilde{\mathbf{x}}^{(i)}\right\|^2\right]\\
=&\mathbb{E}_{\mathbf{x}_0,\epsilon}\Bigg[\sum_{i}\sum_{t=1}^{T-1}\gamma_t\Bigg\|\tilde{\mathbf{x}}_0^{(i)}-\left[\sqrt{\bar{\alpha}_t}\theta\sum_{j\neq i}\tilde{\mathbf{x}}_0^{(j)}+\sqrt{1-\bar{\alpha}_t}\theta\sum_{j\neq i}\epsilon_t^{(j)}-(N-1)\sqrt{\bar{\alpha}_t}\theta\tilde{\mathbf{x}}_0^{(i)}-(N-1)\sqrt{1-\bar{\alpha}_t}\theta\epsilon_t^{(i)}\right]\\
&\qquad\qquad-\left(\sqrt{\bar{\alpha}_t}\tilde{\mathbf{x}}_0^{(i)}+\sqrt{1-\bar{\alpha}_t}\epsilon_t^{(i)}\right)\Bigg\|^2\Bigg]\\
=&\mathbb{E}_{\mathbf{x}_0,\epsilon}\Bigg[\sum_{i}\sum_{t=1}^{T-1}\gamma_t\Bigg\|(\boldsymbol{I}+(N-1)\sqrt{\bar{\alpha}_t}\theta-\sqrt{\bar{\alpha}_t}\boldsymbol{I})\tilde{\mathbf{x}}_0^{(i)}+\sqrt{1-\bar{\alpha}_t}((N-1)\theta-\boldsymbol{I})\epsilon_t^{(i)}\\
&\qquad\qquad-\sqrt{\bar{\alpha}_t}\theta\sum_{j\neq i}\tilde{\mathbf{x}}_0^{(j)}-\sqrt{1-\bar{\alpha}_t}\theta\sum_{j\neq i}\epsilon_t^{(j)}\Bigg\|^2\Bigg]\\
=&\sum_{t=1}^{T-1}\gamma_t\sum_{d}\left[\left[N^2(N-1)\right]\theta_d^2+\left[2N(N-1)(\sqrt{\bar{\alpha}_t}-1)\right]\theta_d+\left[N(1-\sqrt{\bar{\alpha}_t})^2+N(1-\bar{\alpha}_t)\right]\right]
\end{aligned}
\tag{27}
$$

The difference between the final derivation of Equation (27) and Equation (26) are highlighted in blue. Thus the minimum along the dimension d over the parameter θ_d can be expressed in the same format as follows:

$$
\begin{aligned}
&N\sum_{t}\gamma_t[(1-\sqrt{\bar{\alpha}_t})^2+(1-\bar{\alpha}_t)]+\sum_{t}\gamma_t(1-\sqrt{\bar{\alpha}_t})^2\sum_{i}\mu_d^{(i)2}\\
&-\frac{\left[2N(N-1)\sum_t\gamma_t(\sqrt{\bar{\alpha}_t}-1)+(2N-1)\sum_t\gamma_t\sqrt{\bar{\alpha}_t}(1-\sqrt{\bar{\alpha}_t})\sum_i\mu_d^{(i)2}\right]^2}{4\left[N^2(N-1)\sum_t\gamma_t+[N+(N-1)^2]\sum_t\gamma_t\bar{\alpha}_t\sum_i\mu_d^{(i)2}\right]}
\end{aligned}
\tag{28}
$$

The above formula can be expressed in the form $f(x)=ax+b-\frac{(ex+f)^2}{cx+d}$ where $x=\sum_i\mu_d^{(i)2}$ and the constants a,b,c,d,e,f are all positive in our setting. $f(x)$ is strictly monotone increasing function on $[0,+\infty)$ when $ac>e^2$. Thus our proof is done when the following inequation holds:

$$
[N+(N-1)^2]\sum_{t}\gamma_t(1-\sqrt{\bar{\alpha}_t})^2\sum_{t}\gamma_t\bar{\alpha}_t>\left[(2N-1)\sum_{t}\gamma_t\sqrt{\bar{\alpha}_t}(1-\sqrt{\bar{\alpha}_t})\right]^2 \tag{29}
$$

Obviously, $N+(N-1)^2>(2N-1)^2$. According to Cauchy–Schwarz inequality $\left(\sum_i u_iv_i\right)^2\leq\left(\sum_i u_i\right)^2\left(\sum_i v_i\right)^2$, $\sum_t\gamma_t(1-\sqrt{\bar{\alpha}_t})^2\sum_t\bar{\alpha}_t=\left(\sum_t(\sqrt{\gamma_t}(1-\sqrt{\bar{\alpha}_t}))^2\right)\cdot\left(\sum_t(\sqrt{\gamma_t\bar{\alpha}_t})^2\right)\geq\left(\sum_t(\sqrt{\gamma_t}(1-\sqrt{\bar{\alpha}_t}))(\sqrt{\gamma_t\bar{\alpha}_t})\right)^2=\left(\sum_t\gamma_t\sqrt{\bar{\alpha}_t}(1-\sqrt{\bar{\alpha}_t})\right)^2$ also holds. Thus we reach the conclusion $\min_\theta-\text{ELBO}_{\text{decomp}}<\min_\theta-\text{ELBO}_{\text{standard}}$.

For the aspect (b), we refer to the proof in (Lee et al., 2021) which shows the tighter ELBO with prior $\mathcal{N}(\boldsymbol{\mu},\boldsymbol{\Sigma})$ than $\mathcal{N}(\mathbf{0},\boldsymbol{I})$ under some assumptions when $\boldsymbol{\Sigma}$ aligns with the covariance of data distribution. Because the decomposed priors used in the training phase are obtained by Maximum Likelihood Estimation (MLE), they are supposed to have better alignment with data distribution.

D Derivation of Validity Guidance

In this section, we will show the derivation of arms-scaffold drift and clash drift in detail.

The additional drift term that promotes the connection between the arms and scaffold is derived as follows:

$$
\begin{aligned}
&\nabla_{\boldsymbol{x}_t} \log P(\{\rho_{\min} \leq \min_{i\in\mathcal{A}_n, j\in\mathcal{S}} \|\boldsymbol{x}_t^{(i)} - \boldsymbol{x}_t^{(j)}\|_2 \leq \rho_{\max}, n = 1 : |\mathcal{A}|\}|\boldsymbol{x}_t) \\
=&\nabla_{\boldsymbol{x}_t} \sum_{n=1}^{|\mathcal{A}|} \log P(\{\rho_{\min} \leq \min_{i\in\mathcal{A}_n, j\in\mathcal{S}} \|\boldsymbol{x}_t^{(i)} - \boldsymbol{x}_t^{(j)}\|_2 \leq \rho_{\max}\}|\boldsymbol{x}_t) \\
=&\sum_{n=1}^{|\mathcal{A}|} \frac{\nabla_{\boldsymbol{x}_t} \left[P(\{-\min_{i\in\mathcal{A}_n, j\in\mathcal{S}} \|\boldsymbol{x}_t^{(i)} - \boldsymbol{x}_t^{(j)}\|_2 \leq -\rho_{\min}\}|\boldsymbol{x}_t) \cdot P(\{\min_{i\in\mathcal{A}_n, j\in\mathcal{S}} \|\boldsymbol{x}_t^{(i)} - \boldsymbol{x}_t^{(j)}\|_2 \leq \rho_{\max}\}|\boldsymbol{x}_t)\right]}{P(\{\rho_{\min} \leq \min_{i\in\mathcal{A}_n, j\in\mathcal{S}} \|\boldsymbol{x}_t^{(i)} - \boldsymbol{x}_t^{(j)}\|_2 \leq \rho_{\max}\}|\boldsymbol{x}_t)} \\
=&\sum_{n=1}^{|\mathcal{A}|} \zeta_1 \nabla_{\boldsymbol{x}_t} P(\{\min_{i\in\mathcal{A}_n, j\in\mathcal{S}} \|\boldsymbol{x}_t^{(i)} - \boldsymbol{x}_t^{(j)}\|_2 \leq \rho_{\max}\}|\boldsymbol{x}_t) + \zeta_2 \nabla_{\boldsymbol{x}_t} P(\{-\min_{i\in\mathcal{A}_n, j\in\mathcal{S}} \|\boldsymbol{x}_t^{(i)} - \boldsymbol{x}_t^{(j)}\|_2 \leq -\rho_{\min}\}|\boldsymbol{x}_t) \\
=&\sum_{n=1}^{|\mathcal{A}|} \zeta_1 \mathbb{E}[\nabla_{\boldsymbol{x}_t} \mathbb{I}(-\min_{i\in\mathcal{A}_n, j\in\mathcal{S}} \|\boldsymbol{x}_t^{(i)} - \boldsymbol{x}_t^{(j)}\|_2 \leq -\rho_{\min})|\boldsymbol{x}_t] + \zeta_2 \nabla_{\boldsymbol{x}_t} \mathbb{E}[\mathbb{I}(-\min_{i\in\mathcal{A}_n, j\in\mathcal{S}} \|\boldsymbol{x}_t^{(i)} - \boldsymbol{x}_t^{(j)}\|_2 \leq -\rho_{\min})|\boldsymbol{x}_t]
\end{aligned}
\tag{30}
$$

where $\zeta_1 = 1/P(\{-\min_{i\in\mathcal{A}_n, j\in\mathcal{S}} \|\boldsymbol{x}_t^{(i)} - \boldsymbol{x}_t^{(j)}\|_2 \leq -\rho_{\min}\}|\boldsymbol{x}_t)$ and $\zeta_2 = 1/P(\{\min_{i\in\mathcal{A}_n, j\in\mathcal{S}} \|\boldsymbol{x}_t^{(i)} - \boldsymbol{x}_t^{(j)}\|_2 \leq \rho_{\max}\}|\boldsymbol{x}_t)$.

The additional drift term that guides our model to generate molecules outside the protein surface is derived as follows:

$$
\begin{aligned}
&\nabla_{\boldsymbol{x}_t} \log P(\{S(\boldsymbol{x}_t^{(i)}) > \gamma, \forall i\}|\boldsymbol{x}_t) \\
=&\frac{\nabla_{\boldsymbol{x}_t} P(\{S(\boldsymbol{x}_t^{(i)}) > \gamma, \forall i\}|\boldsymbol{x}_t)}{P(\{S(\boldsymbol{x}_t^{(i)}) > \gamma, \forall i\}|\boldsymbol{x}_t)} \\
=&\zeta_3 \nabla_{\boldsymbol{x}_t} \mathbb{E}[\sum_{i=1}^{N_M} \mathbb{I}(-S(\boldsymbol{x}_t^{(i)}) < -\gamma)/N_M|\boldsymbol{x}_t]
\end{aligned}
\tag{31}
$$

where $\zeta_3 = 1/\mathbb{E}[\sum_{i=1}^{N_M} \mathbb{I}(-S(\boldsymbol{x}_t^{(i)}) < -\gamma)/N_M|\boldsymbol{x}_t]$.

Due to the discontinuity of the indicator function that is incompatible with the gradient operator, we use $\xi - \max(0, \xi - y)$ as a surrogate of $\mathbb{I}(y < \xi)$ in Equation (30) and Equation (31). Although $\zeta_1, \zeta_2, \zeta_3$ are dependent on $\boldsymbol{x}_t$, we find setting them as constant still works well. With these two approximations, we can derive the arms-scaffold drift and clash drift as Equation (20) and Equation (21) respectively.

E Implementation Details

E.1 Featurization

We represent each protein atom with the following features: one-hot element indicator (H, C, N, O, S, Se), one-hot amino acid type indicator (20 dimension), one-dim flag indicating whether the atom is a backbone atom, and one-hot arm/scaffold region indicator. If the distance between the protein atom and any arm prior center is within 10 Å, the protein atom will be labeled as belonging to an arm region and otherwise a scaffold region.

The ligand atom is represented with following features: one-hot element indicator (C, N, O, F, P, S, Cl) and one-hot arm/scaffold indicator. Note that the partition of arms and scaffold is predetermined. Thus, the decomposition feature only serves as the input of the model but will not be involved in the network's prediction.

We build two graphs for message passing in the protein-ligand complex: a k-nearest neighbors graph upon ligand atoms and protein atoms (we choose $k = 32$ in all experiments) and a fully-connected graph upon ligand atoms. As Sec. 3.3 mentioned, we only need to predict a subset of edges even though the message passing can be performed on all edges. In the knn graph, the edge features are the outer products of distance embedding and edge type. The distance embedding is obtained by expanding distance with radial basis functions located at 20 centers between 0 Åand 10 Å. The edge type is a

Table 7. Summary of different properties of reference molecules and molecules generated by multiple variants of our model and TargetDiff for comparison. (↑) / (↓) denotes a larger / smaller number is better.

Methods	Vina Score (↓)		Vina Min (↓)		Vina Dock (↓)		High Affinity (↑)		QED (↑)		SA (↑)		Success Rate (↑)
	Avg.	Med.	Avg.	Med.	Avg.	Med.	Avg.	Med.	Avg.	Med.	Avg.	Med.	Avg.
Reference	-6.36	-6.46	-6.71	-6.49	-7.45	-7.26	-	-	0.48	0.47	0.73	0.74	25.0%
TargetDiff	-5.47	-6.30	-6.64	-6.83	-7.80	-7.91	58.1%	59.1%	0.48	0.48	0.58	0.58	10.5%
DecompDiff- Atom - Ref Prior	-6.11	-6.16	-6.68	-6.58	-7.53	-7.59	59.8%	63.0%	0.48	0.48	0.61	0.61	11.2%
DecompDiff- Atom - Pocket Prior	-5.72	-7.49	-7.66	-8.33	-9.08	-9.32	79.0%	96.0%	0.41	0.38	0.52	0.52	11.4 %
DecompDiff- Atom - Opt Prior	-6.01	-7.34	-7.65	-8.01	-8.93	-9.02	74.5%	88.4%	0.37	0.34	0.50	0.50	5.8 %
DecompDiff- Bond - Ref Prior	-5.17	-5.25	-6.03	-5.98	-7.10	-7.14	48.9%	45.6%	0.51	0.51	0.66	0.65	15.4%
DecompDiff- Bond - Pocket Prior	-5.69	-6.06	-7.14	-7.19	-8.50	-8.55	69.2%	81.6%	0.41	0.39	0.59	0.59	19.8%
DecompDiff- Bond - Opt Prior	-5.67	-6.04	-7.04	-7.09	-8.39	-8.43	64.4%	71.0%	0.45	0.43	0.61	0.60	24.5%

4-dim one-hot vector indicating the edge is between ligand atoms, protein atoms, ligand-protein atoms or protein-ligand atoms. In the ligand graph, the ligand bond is represented with a one-hot bond type vector (non-bond, single, double, triple, aromatic), an additional feature indicating whether or not two ligand atoms are from the same arm/scaffold prior.

E.2 Model Details

Our neural network is mainly composed of three types of layers: atom update layer, bond update layer, and position update layer. In each layer, we apply graph attention to aggregate the message of each node/edge. The key/value/query embedding is obtained through a 2-layer MLP with LayerNorm and ReLU activation. Stacking these three layers as a block, our model consists of 6 blocks with `hidden_dim=128` and `n_heads=16`.

We set the number of diffusion steps as 1000. For this diffusion noise schedule, we choose to use a sigmoid β schedule with $\beta_1 = \texttt{1e-7}$ and $\beta_T = \texttt{2e-3}$ for atom coordinates, and a cosine β schedule suggested in (Nichol & Dhariwal, 2021) with $s = 0.01$ for atom types and bond types.

E.3 Training Details

The model is trained via gradient descent method Adam (Kingma & Ba, 2014) with `init_learning_rate=0.001`, `betas=(0.95, 0.999)`, `batch_size=4` and `clip_gradient_norm=8`. To balance the scales of different losses, we multiply a factor $\alpha = 100$ on the atom type loss and bond type loss. During the training phase, we add a small Gaussian noise with a standard deviation of 0.1 to protein atom coordinates as data augmentation. We also schedule to decay the learning rate exponentially with a factor of 0.6 and a minimum learning rate of 1e-6. The learning rate is decayed if there is no improvement for the validation loss in 10 consecutive evaluations. The evaluation is performed for every 2000 training steps. We trained our model on one NVIDIA GeForce GTX A100 GPU, and it could converge within 36 hours and 300k steps.

F Additional Results

F.1 Full Evaluation Results

In Table 7, we show the results of multiple variants of our models. All of them leverage decomposed priors and validity guidance to improve the sampling quality. The differences lie in whether to include bond diffusion and which kind of prior is used. *Ref Prior* is estimated from the reference molecule with a Gaussian distribution through maximum likelihood estimation. *Pocket Prior* estimates the prior center as illustrated in Appendix A and uses a neural classifier to estimate the number of ligand atoms and prior standard deviation. *Opt Prior* is a mixture of them which uses Ref Prior if the reference ligand could pass the Success criteria (QED > 0.25, SA > 0.59, Vina Dock < -8.18). It can be seen that bond diffusion consistently has a positive effect on QED and SA. Both bond diffusion and Opt prior have a better balance in the 2D structure rationality and binding affinity, which leads to a higher success rate.

F.2 Time Complexity

For the training efficiency, we summarize the running time per step and the total running time in Table 8. The increase in training time is mainly due to the introduction of bond diffusion, which makes the network more complex. The decomposed

Table 8. Training time of different models.

Model	Time(s) / Step	Total #Steps	Total Time (hrs)
AR	0.15	1.5 M	62.5
Pocket2Mol	0.55	475 K	72.6
TargetDiff	0.30	300 K	25.0
Ours	0.50	300 K	41.7

prior has a negligible impact on the training time.

For the sampling efficiency, AR, Pocket2Mol, GraphBP, and TargetDiff use 7785s, 2544s, 105s, and 3428s for generating 100 valid molecules on average separately. It takes DecompDiff 5570s / 6189s on average without / with validity guidance. Similarly, the decomposed prior has a negligible impact on the sampling time. Bond diffusion results in 1.62x sampling time compared to TargetDiff, and validity guidance makes the sampling time slightly increase by 10% further.

G Examples of Generated Ligands

In Figure 8, we visualize reference ligands and ligands generated by TargetDiff (Guan et al., 2023) and our model. As the visualization shows, the ligands generated by our model can occupy more space in the concave target binding site due to the design of decomposed prior. Bond diffusion and validity guidance can promote the quality of generated ligands. Thus the ligands generated by our model can achieve better Vina Scores while keeping reasonable QED and SA.

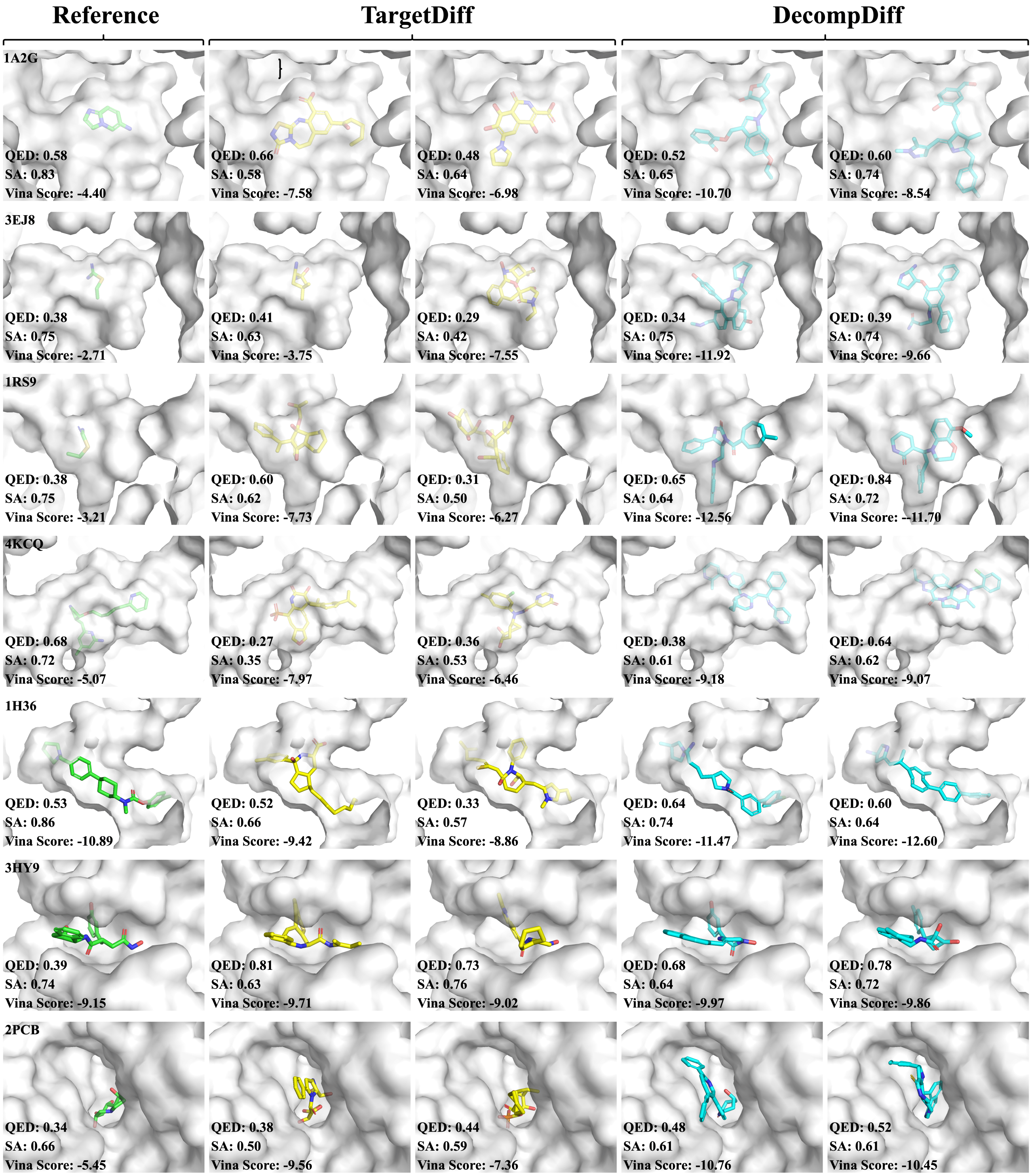

Figure 8. Examples of generated ligands. Carbon atoms in reference ligands, ligands generated by TargetDiff, and our model are visualized in green, yellow, and cyan respectively. Each row corresponds to a protein.