# GRAPHVF: CONTROLLABLE PROTEIN-SPECIFIC 3D MOLECULE GENERATION WITH VARIATIONAL FLOW

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Designing molecules that bind to specific target proteins is a fundamental task in drug discovery. Recent generative models leveraging geometrical constraints imposed by proteins and molecules have shown great potential in generating protein-specific 3D molecules. Nevertheless, these methods fail to generate 3D molecules with 2D skeletal curtailments, which encode pharmacophoric patterns essential to drug potency and synthesizability. To cope with this challenge, we propose GraphVF, which integrates geometrical and skeletal restraints into a variational flow framework, where the former is captured through a normalizing flow transformation and the latter is encoded by an amortized factorized Gaussian. We empirically verify that our method achieves state-of-the-art performance on protein-specific 3D molecule generation in terms of binding affinity and some other drug properties. In particular, it represents the first *controllable* geometry-aware, protein-specific molecule generation method, which enables creating binding 3D molecules with specified chemical sub-structures or drug properties.

## 1 INTRODUCTION

The *de novo* design of synthetically feasible drug-like molecules that bind to specific protein pockets is a crucial yet very challenging task in drug discovery. To cope with such challenge, there has been a recent surge of interest in leveraging deep generative models to effectively searching the chemical space for molecules with desired properties. These machine learning models typically encode the chemical structures of molecules into a low-dimensional space, which then can be optimized and sampled to generate potential 2D or 3D molecule candidates (Jin et al., 2018; Shi et al., 2020; Zhu et al., 2022; Hoogeboom et al., 2022). Along this research line, a more promising direction has also been explored recently: generating 3D molecules that bind to given proteins.

Such binding 3D molecule generation is fundamentally important because binding in fact mainly facilitates the functionalities of drugs. Fortunately, leveraging autoregressive models to generate drug molecules (i.e., ligands) directly based on the 3D geometry of the binding pocket have shown promising potential (Luo et al., 2021; Peng et al., 2022; Liu et al., 2022). These methods explicitly capture the fine-grained atomic interactions in the 3D space, and produce ligand poses that can directly fit into the given binding pocket. Nevertheless, two critical issues remains unsolved for these existing geometric approaches: 1) effective encoding and sufficient preservation of pharmacophoric structural patterns in the ligand candidates, and 2) controllable ligand generation that aims at specified drug properties or sub-structures. The former prevents generating ligands that seem geometrically plausible, yet structurally invalid or pharmacophorically impotent; the later dominates the synthesibility and the practical usefulness of the drugs. We further elaborate them next.

In practice, it is extremely valuable to keep track of the pharmacophoric patterns in the existing ligands, which indeed determines a ligand's bio-chemical activities and binding affinity to a large extent (Wermuth et al., 1998). Consider, for example, the molecules of serotonin (a benign neurotransmiter) and N,N-Dimethyltryptamine (DMT, a famous hallucinogen). As can be seen in Figure 1a, serotonin and DMT share a large common bulk of their structures, which both possess an indole and an ethylamine group, but differ enormously in their neural activities. In fact, the extra Methyl groups in DMT's $\mathrm{NHMe_2}$ are pharmacophoric, inducing an attractive charge interaction with Asp-231 (Gomez-Jeria & Robles-Navarro, 2015). This pharmacophoric feature gives rise to

DMT's binding affinity with the $5-\mathrm{HT}_{2\mathrm{A}}$ binding site and produces hallucination. Such observations suggest that effectively enforcing pharmacophoric patterns in ligands is critical for binding.

Equally important, controlling molecular properties like solubility, polarizability and heat capacity are instrumental to drug quality. This is to make sure that the synthesized drug molecules have good exposure, *i.e.* absorption/distribution/metabolism/excretion (ADME) *in vivo*, and thus, sufficient efficacy in clinical trials (Egan, 2010). It is worth noting that, although recent diffusion models like EDM (Hoogeboom et al., 2022) have been popular for their capability to perform controlled generation on these properties, performing such control while being pertinent to a given pocket structure for binding remains under-explored by previous works.

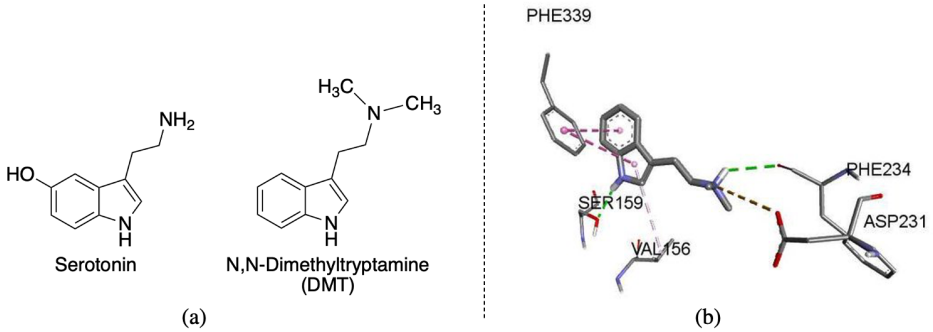


Figure 1: (a) Comparison between Serotonin and DMT structure; (b) Binding pose of DMT with $5-\mathrm{HT}_{2\mathrm{A}}$, pay special attention to the interaction between $\mathrm{NHMe}_2$ and Asp-231.

To address the aforementioned two issues, we propose GraphVF, a protein-aware molecule generation framework that integrates both geometrical and skeletal constraints, aiming at controlling over the structure and property of the generated ligands. To attain this goal, we leverage flow-based architecture that combines amortized variational inference (Zhang et al., 2018) and autoregressive normalizing-flow generation. In specific, global structure of the drug ligand is organized as a junction tree (Jin et al., 2018), and fine-grained geometrical context of the protein receptor is encoded via a valence-aware E(3)-GNN. These two constraints are integrated into a variational flow architecture, where the former enforces the variational distribution globally, while the latter administers the flow transformations autoregressively.

We show empirically that, GraphVF generates drug molecules with high binding affinity to the receptor proteins, with or without the aid of reference ligands, outperforming state-of-the-art methods in terms of binding affinity and some other drug properties. More importantly, GraphVF exposes a clean-cut interface for imposing customized constraints, which is extremely useful in practice for controlling the sub-structure and bio-chemical property of generated drug ligands. To specify what our proposed model can actually do, we comprehensively compare GraphVF with several representative models for molecule generation in Table 1.

Our main contributions are summarized as follows.

- We devise a novel variational flow-based framework to integrate geometrical and skeletal restraints for protein-specific 3D molecule generation.
- We propose the first controllable protein-specific molecule generation method, enabling generating 3D molecules with specified chemical sub-structures or bio-chemical properties.
- We empirically demonstrate our method's superior performance to state-of-the-art approaches on generating binding 3D molecules.

## 2 Related Work

**Non-Protein Specific Molecule Generation** Different generative techniques have been applied to the task of molecular generation, including Variational Autoencoders (VAEs) (Kingma & Welling, 2013), Diffusion Models (Sohl-Dickstein et al., 2015), Normalizing Flows (NFs) (Dinh et al., 2016),

Table 1: Comparison among representative molecular generative methods.

| Name | Generative Model | Moleculer Encoding | | | Controlled Generation | | |
|---|---|---|---|---|---|---|---|
| | | Atom | Bond | 3D Coord. | Ligand Struc. | Receptor Conf. | Generic Prop. |
| EDM | Diffusion | ✓ | - | ✓ | - | - | ✓ |
| DMCG | VAE | ✓ | ✓ | ✓ | ✓ | - | - |
| JT-VAE | VAE | ✓ | ✓ | - | ✓ | - | - |
| GraphAF | Autoregressive Flow | ✓ | ✓ | - | - | - | - |
| GraphBP | Autoregressive Flow | ✓ | - | ✓ | - | ✓ | - |
| Pocket2Mol | Spatial Autoregression | ✓ | ✓ | ✓ | - | ✓ | - |
| GraphVF | Variational Flow | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |

and Autoregressive Models (Van Oord et al., 2016). The line of work is usually context-free, aiming to produce high-quality molecules from scratch, or to render reasonable 3D conformations of given molecules. For example, JT-VAE (Jin et al., 2018) generates molecular graphs with the guidance of a tree-structured scaffold over chemical substructures. GraphAF (Shi et al., 2020) uses a flow-based model to generate atoms and bonds in an autoregressive manner. DMCG (Zhu et al., 2022) and EDM (Hoogeboom et al., 2022) leverage equivariant diffusion or iterative sampling and de-noising to generate 3D conformations from 2D structures. Unlike these methods, our approach aims at generating molecules that bind to given 3D protein pockets.

**3D Molecule Generation for Target Protein Binding** With the wide availability of large-scale datasets (Francoeur et al., 2020; Li et al., 2021) for target protein binding, recent works have been able to generate drug ligands directly based on the 3D geometry of the binding pockets. For example, Pocket2Mol (Peng et al., 2022) leverages a spatial-autoregressive model; it directly models the *p.d.f.* for atom occurrence in the 3D space as a Gaussian mixture (GMM), and then iteratively places the atoms from the learned distribution until there is no room for new atoms. GraphBP (Liu et al., 2022), an autoregressive model, retains good model capacity via normalizing flow; variables are randomly sampled from a compact latent space, before they are projected into the chemical space by an arbitrarily complex flow transformation. Despise their promising potential, these methods ignore the topological organization of the drug ligand itself, as well as the structural patterns and pharmacophoric features embodied in it. As a result, existing methods tend to generate ligands that seem geometrically plausible, yet structurally invalid or pharmacophorically impotent. Our approach here aims to address this problem. Also, our method enables controllable molecule generation, facilitating generating drug ligand candidates with specified chemical sub-structures or drug properties.

## 3 PRELIMINARIES

### 3.1 AUTOREGRESSIVE FLOW MODELS

Based on the prior distribution $p_Z$, we can define a flow model (Dinh et al., 2014; Rezende & Mohamed, 2015; Weng, 2018) as an invertible parameterized function $f_\theta : \mathbf{z} \in \mathbb{R}^D \rightarrow \mathbf{x} \in \mathbb{R}^D$ , which maps the latent variable $\mathbf{z} \sim p_Z$ to the data variable $\mathbf{x}$. Then we could calculate the log-likelihood of a data point $\mathbf{x}$ by

$$\log p_X(\mathbf{x}) = \log p_Z\left(f_\theta^{-1}(\mathbf{x})\right) + \log \left| \det \frac{\partial f_\theta^{-1}(\mathbf{x})}{\partial \mathbf{x}} \right|, \tag{1}$$

and we hope that we could calculate the Jacobian determinant easily. Let $\mathbf{x}_i$ be the $i$-th dimension of $\mathbf{x}$. Autoregressive flow model (Papamakarios et al., 2017) is a kind of flow model which is formulated to be autoregressive. In case that for every $i$, $\mathbf{x}_i$ is conditioned on $\mathbf{x}_{1...i-1}$, $f_\theta^{-1}$ is defined as the following transformation function:

$$\mathbf{x}_i = \sigma_i(\mathbf{x}_{1...i-1}) \odot \mathbf{z}_i + \mu_i(\mathbf{x}_{1...i-1}), \quad i = 1...D, \tag{2}$$

where $\odot$ denotes element-wise multiplication, $\sigma_i(\cdot) \in \mathbb{R}$ and $\mu_i(\cdot) \in \mathbb{R}$ are functions of $\mathbf{x}_{1...i-1}$. Thus, we can easily calculate:

$$\mathbf{z}_i = \frac{\mathbf{x}_i - \mu_i}{\sigma_i}, \quad \det \frac{\partial f_\theta^{-1}(\mathbf{x})}{\partial \mathbf{x}} = \prod_{i=1}^{D} \frac{1}{\sigma_i}. \tag{3}$$

### 3.2 Problem Formulation and Notations

Given a protein pocket, our task is to generate a ligand molecule that binds to the protein pocket. The target protein pocket is represented by $\mathcal{P} = \{a_i^{pro}, \mathbf{x}_i^{pro}\}, i = 1...N$, where $N$ denotes the number of atoms in the pocket. For the $i$-th atom in the protein, $a_i^{pro}$ represents its element and $\mathbf{x}_i^{pro}$ represents its coordinates in 3D space. We represent a ligand molecule as $\mathcal{L} = \{a_i^{lig}, \mathbf{x}_i^{lig}, \mathbf{b}_i^{lig}\}, i = 1...n$, where $n$ is the number of atoms in the ligand, and for the $i$-th atom in the ligand, $a_i^{lig}$, $\mathbf{x}_i^{lig}$, and $\mathbf{b}_i^{lig}$ represents its element, 3D coordinates, and bonding relationships to the previous atoms in the ligand, respectively. For every $i$, $\mathbf{x}_i$ is a real vector and $|\mathbf{x}_i| = 3$, $\mathbf{b}_i^{lig}$ is an integer vector and $|\mathbf{b}_i^{lig}| = i - 1$. All bond types are represented by $1...b$, and $0$ indicates no bond or no bonding information is provided. Additionally, we represent the 2D skeleton of a ligand molecule as $\mathcal{R} = \{a_i^{lig}, \mathbf{b}_i^{lig}\}, i = 1...n$.

## 4 The Proposed Method

In this section, we first introduce how we encode both the ligand scaffold and the protein-ligand geometry information, and then detail how these two pieces of knowledge are fused into a variational flow framework for binding geometric molecule generation.

### 4.1 Ligand Scaffolds Encoding

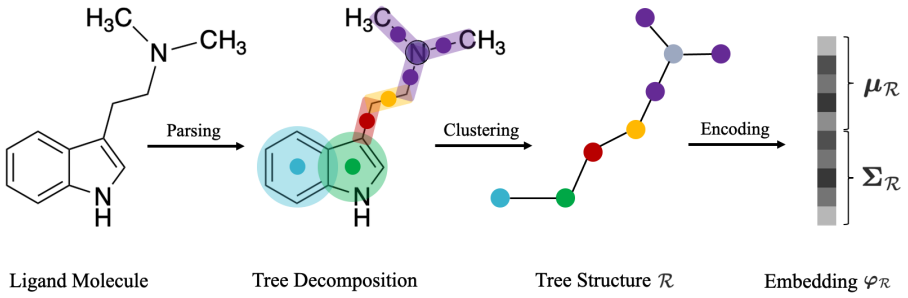


Figure 2: A ligand molecule (e.g. DMT) is encoded for its structure patterns through sub-structure parsing, graph clustering and tree encoding. By definition, the blue and green nodes belong to canonical type (I), the grey node belong to canonical type (III), and other nodes belong to canonical type (II).

Inspired by (Jin et al., 2018), structure patterns of the reference drug ligand are extracted in a fragment-driven approach (Jin et al., 2018). In most cases, a reasonable molecule is a compilation of canonical chemical sub-structures like rings and functional groups. Thus, encoding molecule structures on the granularity of fragments (rather than atoms) can preserve semantics on the sub-structure level to the greatest extent. Figure 2 illustrates the overall procedure for fragment-based molecule encoding, using DMT as an example. Following JT-VAE (Jin et al., 2018), the molecule scaffolds are first parsed into a compilation of occluded canonical sub-structures: (I) rings, (II) non-ring covalent atom pairs, and (III) intersection atoms of 3+ sub-structures. Next, the resulting scaffolds are further clustered into a tree structure $\mathcal{R}$, which is then encoded into a molecular embedding $\boldsymbol{\varphi}_{\mathcal{R}}$ via a Gated Recurrent Unit (Chung et al., 2014) adapted for tree message passing, as follows.

$$\mathbf{s}_{ij} = \sum\nolimits_{k \in N(i) \setminus j} \mathbf{m}_{ki}. \tag{4}$$

$$\mathbf{z}_{ij} = \sigma(\mathbf{W}^z \mathbf{x}_i + \mathbf{U}^z \mathbf{s}_{ij} + \mathbf{b}^z). \tag{5}$$

$$\mathbf{r}_{ki} = \sigma(\mathbf{W}^r \mathbf{x}_i + \mathbf{U}^r \mathbf{m}_{ki} + \mathbf{b}^r). \tag{6}$$

$$\widetilde{\mathbf{m}}_{ij} = \tanh(\mathbf{W}\mathbf{x}_i + \mathbf{U} \sum_{k \in N(i) \setminus j} \mathbf{r}_{ki} \odot \mathbf{m}_{ki}). \tag{7}$$

$$\mathbf{m}_{ij} = (1 - \mathbf{z}_{ij}) \odot \mathbf{s}_{ij} + \mathbf{z}_{ij} \odot \widetilde{\mathbf{m}}_{ij}. \tag{8}$$

The resulting embedding $\boldsymbol{\varphi}_{\mathcal{R}}$, which encodes the sub-structures of ligands, is then split into halves as $(\boldsymbol{\mu}_{\mathcal{R}}, \boldsymbol{\Sigma}_{\mathcal{R}})$ and parameterizes the latent distribution of the variational flow, which will be discussed in detail in Section 4.3.

### 4.2 Geometry Graph Encoding

Equivariant graph neural networks like SchNet (Schutt et al., 2017) and EGNN (Satorras et al., 2021) have become a routine component in this **receptor-based** line of work, which are essential for encoding molecular features with roto-translational equivariance. Atoms around the binding pocket are organized into a *k*NN/radius graph, based on their euclidean distance in the 3D space. This graph organization is appropriate for modeling non-covalent interactions (hydrophobic, ionic, hydrogen bond, *etc.*) between receptor atoms and ligand atoms. Nonetheless, it is still a worthwhile effort to distinguish among different types of covalent (single, double, triplet, aromatic, etc.) and non-covalent ones.

To explicitly incorporate bond information during geometry graph encoding, we devise **Echnet**, a *SE(3)-Invariant* graph neural network. As an enhanced version to *Schnet* (Schutt et al., 2017), *Echnet* encodes relative distance together with other *SE(3)-Invariant* edge features by concatenating their representations and feeding them into message passing layers, as formulated as follows:

$$\boldsymbol{h}_i^{(0,lig^k)} = \text{Emb}(a_i^{lig}), i = 1...k-1. \tag{9}$$

$$\boldsymbol{h}_i^{(l,lig^k)} = \boldsymbol{h}_i^{(l-1,lig^k)} + \sum_{(i',i)\in\mathcal{E}} \boldsymbol{h}_{i'}^{(l-1,lig^k)} \odot \mathbf{MLP}_G^l(\boldsymbol{e}_{i',i}), l \geq 1. \tag{10}$$

$$\text{where} \quad \boldsymbol{e}_{i',i} = concat\{\text{Erbf}(dist(i',i)), \text{Emb}(\mathbf{b}_i^{lig}[i'])\}. \tag{11}$$

Here $\mathbf{b}_i^{lig}[i']$ denotes the $i'$-th coordinate in the vector $\mathbf{b}_i^{lig}$, representing the bonding relationship between the $i$-th atom and the $i'$-th atom in the ligand. **Emb** denotes ordinary embeddings for $a_k^{lig}$ and $\mathbf{b}_i^{lig}[i']$, while **Erbf** represents the embedding of $d^k$ with radial basis functions (Liu et al., 2022).

### 4.3 Variational Flow

The two types of encoding discussed in Section 4.1 and 4.2 are seamlessly incorporated in a variational flow framework, fulfilling the following special needs for binding ligand generation:

1) Integration of both binding pocket geometry and ligand structural patterns;
2) Generation of molecules with high binding affinity, even without reference ligands;
3) Controllable generation interface for customized bio-chemical constraints.

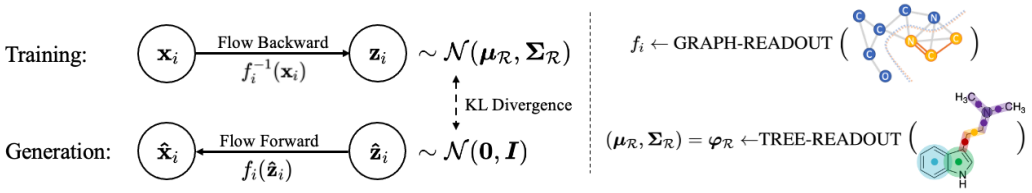


Figure 3: Framework for variational flow based generation. 3D conformational graph is encoded autoregressively during flow transformation, whereas the 2D structure backbone is encoded globally as amortized variational distribution.

We illustrate this variational flow framework in Figure 3, and discuss its *training* and *generation* processes in detail next.

**Training:** Consider we are now in the $i^{th}$ step for training. $\mathcal{G}_{i-1}$ is an intermediary 3D geometry, which consists of atoms determined in the previous $i-1$ steps, and depicts the fine-grained 3D geometric dependencies inside the binding pocket. $\mathcal{R}$ is the complete 2D scaffold of a reference drug ligand, which gives a holistic sense of the ligand structure, and is shared across all steps.

$\mathbf{x}_i$ is the ground truth label for the bond/atom to be generated in step $i$. The autoregressive flow transformation $f_i$ is parameterized via geometric graph encoding of $\mathcal{G}_{i-1}$. The latent distribution $p_Z = \mathcal{N}(\boldsymbol{\mu}_\mathcal{R}, \boldsymbol{\Sigma}_\mathcal{R})$ is parameterized via tree message-passing along the junction tree of $\mathcal{R}$.

**Generation:** In the $i^{th}$ step for generation, $f_i$ is parameterized like-wise, but the latent distribution $p_{\hat{Z}}$ (from which $\mathbf{z}_i$ is sampled) can be flexibly parameterized, according to different generation settings: (1) $\mathcal{N}(0, I)$ for regular pocket-based generation, and (2) $\mathcal{N}(\boldsymbol{\mu}_\mathcal{P}, \boldsymbol{\Sigma}_\mathcal{P})$ for controlled generation on given property $\mathcal{P}$, which is encoded by taking the average over the $(\boldsymbol{\mu}_\mathcal{R}, \boldsymbol{\Sigma}_\mathcal{R})$s of ligands with the specific property $\mathcal{P}$. $\hat{\mathbf{x}}_i = f_i(\hat{\mathbf{z}}_i)$ is the predicted atom/chemical bond, which concludes the $i^{th}$ generation step.

**Optimization:** To achieve robust performance across the above generation settings, variational flow is formalized into a bi-level optimization task during each training step $i$:

$$\text{maximize} \quad p_X(\mathbf{x}_i)\,, \tag{12}$$

$$\text{subject to} \quad D_{\mathrm{KL}}(\hat{X}_i||X_i) = D_{\mathrm{KL}}(\hat{Z}||Z) < \tau, \tag{13}$$

where $Z$ and $\hat{Z}$ are the latent distributions respectively for training and generation, and $X_i = f_i(Z)$ and $\hat{X}_i = f_i(\hat{Z})$ are the chemical space distributions during training and generation, respectively. The discrepancy between ground-truth $X_i$ and generated $\hat{X}_i$ is regularized with Kullback-Leibler (KL) divergence. Since $f_i$ is an invertible function, this is equivalent to regularizing the KL divergence between $Z$ and $\hat{Z}$. And thus, we can derive the practical form of loss function during training step $i$ as:

$$\mathcal{L}_\rangle = \mathcal{L}_{\text{flow}} + \beta\mathcal{L}_{\mathrm{KL}} \tag{14}$$

$$= \frac{1}{2}(\mathbf{z}_i - \boldsymbol{\mu}_\mathcal{R})^T \boldsymbol{\Sigma}_\mathcal{R}^{-1}(\mathbf{z}_i - \boldsymbol{\mu}_\mathcal{R}) - \log\left|\det\frac{\partial f_i^{-1}(\mathbf{x}_i)}{\partial \mathbf{x}_i}\right| \tag{15}$$

$$+ \beta \cdot D_{\mathrm{KL}}(\mathcal{N}(\boldsymbol{\mu}_\mathcal{R}, \boldsymbol{\Sigma}_\mathcal{R})||\mathcal{N}(\mathbf{0}, \boldsymbol{I})), \tag{16}$$

where equation 15 is the flow loss term, derived from equation 1, and equation 16 is the KL loss term. Note that in equation 15, the first term stands for the log likelihood of the amortized Gaussian $\log p_Z(\mathbf{z}_0)$.

It is worth noting that, in equation 14, $\beta$ is a sensitivity hyper-parameter that trades off between 2D and 3D context, preventing model deterioration and boosting model robustness. For example, decreasing $\beta$ would encourage the diversity of the amortized distribution and allow for greater $\boldsymbol{\phi}_\mathcal{R}$ expressiveness. However, this also makes the flow transformation more hinged on reference $\phi_R$, and prevents the model from scaling across diverse generative settings. By contrast, increasing $\beta$ would limit the expressiveness of the latent distribution, and degrade the model to a purely geometric approach that provides no explicit control over other desired qualities. In other words, $\beta$ can be fine-tuned to find a good trade-off for different end applications.

### 4.4 New Ligand Generation

We formalize the procedure of new ligand generation as a sequential decision-making process. At each step, we generate all information of one atom in the ligand, as illustrated in Figure 4. To be specific, at the $k$-th step we perform the following operations in sequential order.

Firstly, we construct a graph $\mathcal{G}^k$ based on $\mathcal{P}$ and $\mathcal{L}^{k-1}$, where $\mathcal{L}^{k-1} = \{a_i^{lig}, \mathbf{x}_i^{lig}, \mathbf{b}_i^{lig}\}, i = 1...k-1$ represents the generated part of the ligand. One atom is mapped to one node in $\mathcal{G}^k$, and an edge is created between two nodes if the distance between them is less than a certain constant $c$. In $\mathcal{G}^k$, atomic elements serve as node features, while bonding relationships and relative distances are used as edge features. We create different node features for protein elements and ligand elements. We then encode the protein and generated part of the ligand molecule and obtain node representations, using the $\mathbf{Echnet}$ (see Section 4.2) as encoder:

$$\boldsymbol{h}_i^{lig^k}, \boldsymbol{h}_j^{pro^k} = \mathbf{Echnet}(\mathcal{G}^k), i = 1...k-1, j = 1...N, \tag{17}$$

Secondly, for every ligand atom, we predict whether it could serve as a *focal atom* (Liu et al., 2022).

$$focal_i^k = \mathbf{MLP}_f(\boldsymbol{h}_i^{lig^k}), k > 1, \tag{18}$$

$$start_j^1 = \mathbf{MLP}_s(\boldsymbol{h}_j^{pro^1}). \tag{19}$$

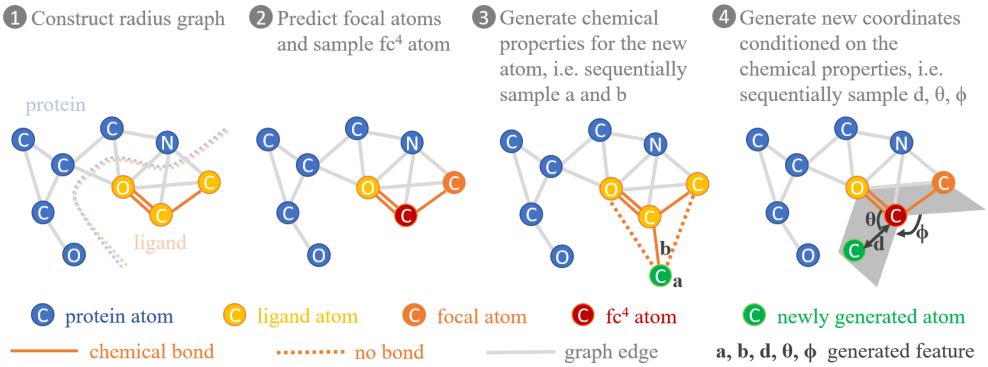


Figure 4: An illustration of one generation step when $k = 4$. Details are discussed in section 4.4.

An atom in the ligand is regarded as a focal atom if and only if one or more bonds could be attached to it. The whole generation process would terminate when no atom could serve as a focal atom. After that, one atom is sampled from all focal atoms as $fc^k$. As for $k = 1$, we use another classifier to predict that, for every atom in the target protein, whether it could serve as the start point of generation. Then we sample one of the possible start points as $fc^1$.

Based on $fc^k$, the closest atom to $fc^k$ (denoted as $fc^{k_1}$) and the second closest atom to $fc^k$ (denoted as $fc^{k_2}$), we construct a local spherical coordinate system, in which we could write coordinates of any $atom$ as $(d^k, \theta^k, \phi^k)$. $d^k$ is the distance between $fc^k$ and $atom$, $\theta^k$ is the angle formed by line $(fc^k, fc^{k_1})$ and line $(fc^k, atom)$, and $\phi^k$ is the torsion angle formed by plane $(fc^k, fc^{k_1}, fc^{k_2})$ and plane $(fc^k, fc^{k_1}, atom)$. Notice that $(d^k, \theta^k, \phi^k)$ is *SE(3)-Invariant*.

Finally, we use the variational flow model, as discussed in Section 4.3, to generate $a_k^{lig}$, $\mathbf{b}_k^{lig}$, $d^k$, $\theta^k$, $\phi^k$. We sequentially generate these features so that generating each feature is conditioned on all previously generated features, in order to better capture the underlying dependencies (Liu et al., 2022). $z_a^k, z_b^k, z_d^k, z_\theta^k, z_\phi^k$, however, are independently sampled from certain prior distribution and used as latent variables in the variational flow. Under the setting for unconditional ligand generation, we set the prior distribution as $\mathcal{N}(\mathbf{0}, \boldsymbol{I})$. To be specific, we first generate $a_k^{lig}$ and $\mathbf{b}_k^{lig}$ in the following steps:

$$\mu_{ele_x}^k, \sigma_{ele_x}^k = \mathbf{MLP}_{ele_x}(\boldsymbol{h}_{fc^k}^{lig^k}), \tag{20}$$

$$a_k^{lig} = argmax_x\{\sigma_{ele_x}^k z_a^k + \mu_{ele_x}^k\}, \tag{21}$$

$$\boldsymbol{h}_i^{lig_a^k} = \boldsymbol{h}_i^{lig^k} \odot \text{Emb}(a_k^{lig}), i = 1...k-1, \tag{22}$$

$$\mu_{bond_x}^k[i], \sigma_{bond_x}^k[i] = \mathbf{MLP}_{bond_x}(\boldsymbol{h}_i^{lig_a^k}), i = 1...k-1, \tag{23}$$

$$\mathbf{b}_k^{lig}[i] = argmax_x\{\sigma_{bond_x}^k[i] z_b^k + \mu_{bond_x}^k[i]\}, i = 1...k-1, \tag{24}$$

where $\odot$ demotes element-wise multiplication, $ele$ is a list of possible atom types and $bond$ is a list of possible bonding relationships (including no bond).

Similarly, we generate $d^k, \theta^k, \phi^k$ in a sequential order:

$$\boldsymbol{h}_i^{lig_b^k} = \boldsymbol{h}_i^{lig_a^k} \odot \text{Emb}(\mathbf{b}_k^{lig}[i]), i = 1...k-1, \tag{25}$$

$$\mu_d^k, \sigma_d^k = \mathbf{MLP}_d(\boldsymbol{h}_{fc^k}^{lig_b^k}), d^k = \sigma_d^k z_d^k + \mu_d^k, \tag{26}$$

$$\boldsymbol{h}_i^{lig_d^k} = \boldsymbol{h}_i^{lig_b^k} \odot \text{Erbf}(d^k), i = 1...k-1, \tag{27}$$

$$\mu_\theta^k, \sigma_\theta^k = \mathbf{MLP}_\theta(\boldsymbol{h}_{fc^k}^{lig_d^k}, \boldsymbol{h}_{fc^{k_1}}^{lig_d^k}), \theta^k = \sigma_\theta^k z_\theta^k + \mu_\theta^k, \tag{28}$$

$$\boldsymbol{h}_i^{lig_\theta^k} = \boldsymbol{h}_i^{lig_d^k} \odot \text{Ecbf}(d^k, \theta^k), i = 1...k-1, \tag{29}$$

$$\mu_\phi^k, \sigma_\phi^k = \mathbf{MLP}_\phi(\boldsymbol{h}_{fc^k}^{lig_\theta^k}, \boldsymbol{h}_{fc^{k_1}}^{lig_\theta^k}, \boldsymbol{h}_{fc^{k_2}}^{lig_\theta^k}), \tag{30}$$

$$\phi^k = \sigma_\phi^k z_\phi^k + \mu_\phi^k, \tag{31}$$

where Ecbf denotes the embedding of $\phi^k$ with circular basis functions (Liu et al., 2022). Note that we can directly derive $\mathbf{x}_k^{lig}$ from $\mathbf{x}_{fc^k}^{lig}$, $\mathbf{x}_{fc^{k_1}}^{lig}$, $\mathbf{x}_{fc^{k_2}}^{lig}$, $d^k$, $\theta^k$, $\phi^k$.

# 5 EXPERIMENTS

| Method | High Affinity(%)↑ |
|---|---|
| GraphBP | 13.4 |
| Pocket2Mol | 27.2 |
| GraphVF (ours) | **31.1** |
| GraphVF (w/o 2D encoder) | 26.3 |

Table 2: Performance of different methods on 3D molecular generation based on protein pockets. Higher values indicate better results. Best results are in **bold**.

## 5.1 3D MOLECULAR GENERATION CONDITIONED ON PROTEIN POCKET

**Dataset.** We use the benchmarking CrossDocked dataset (Francoeur et al., 2020), which contains 22.5 million protein-ligand pairs, to evaluate the generation performance of GraphVF. For fair comparison, we follow Pocket2Mol (Peng et al., 2022) to prepare and split the data.

**Setup.** Following GraphBP (Liu et al., 2022) and Pocket2Mol, we randomly sample 100 molecules for every protein pocket in the generation stage. The quality of generated molecules is evaluated by High Affinity, which estimates the percentage of generated molecules that have higher *CNNAfinity* calculated by the *Gnina* program (McNutt et al., 2021). We choose GraphBP and Pocket2Mol as our baselines, which reprent the state-of-art models for binding molecule generation. For GraphBP and GraphVF, we trained them on the dataset for 40 epochs with the same hyperparameters. For Pocket2Mol (Peng et al., 2022), we obtain the pre-trained model from their authors, and then compute the scores using *Gnina*.

**Results.** The comparison results are presented in Table 5. We can see that our GraphVF outperforms the two state-of-the-art baselines in terms of binding affinity. Meanwhile, the last setting in Table 5 indicates that GraphVF significantly benefits from the 2D encoder, namely the molecular sub-structure constraint. Without the constraint, the High Affinity value for GraphVF drops drastically from 31.1 to 26.3.

## 5.2 CONTROLLABLE GENERATION FOR SPECIFIED CHEMICAL SUB-STRUCTURES

Our pretrained framework could be used to encourage the desired sub-structures to be generated without losing diversity. We carry out case studies on generation of molecules containing the following motifs: oxhydryl, peptide bond, 6-member carbon ring, and 5-member ring containing element S. For each motif, we collect $\mu_{\tilde{\mathcal{R}}}$ and $\sigma_{\tilde{\mathcal{R}}}$ among 500 randomly sampled reference ligand molecules that contain the motif as a sub-structure. Then we set the mean of $\mu_{\tilde{\mathcal{R}}}$ and $\sigma_{\tilde{\mathcal{R}}}$ as the prior distribution during flow generation. Finally, we calculate the rate of the generated molecules that contain

the desired sub-structures on the test set, which is compared with the results of directly sampling from prior distribution $\mathcal{N}(\mathbf{0}, \boldsymbol{I})$.

The experimental results are summarized in Table 3. With the prior distributions collected from molecules that contains certain desired sub-structures, our model is more likely to generate ligand molecules with those sub-structures.

Table 3: Controllable Generation for Specified Chemical Sub-structures.

| **Rate of desired sub-structure(%)** | **w/ latent $\mathcal{P}$** | **w/o latent $\mathcal{P}$** |
|---|---|---|
| oxhydryl | **51.7** | 42.8 |
| peptide bond | **6.4** | 1.5 |
| 6-member carbon ring | **14.5** | 0.3 |
| 5-member ring containing element S | **29.7** | 0.4 |

### 5.3 Controllable Molecular Generation for Specified Drug Properties

Our framework can also be explicitly controlled to generate drug-like molecules with desired properties. To support this claim, we perform case studies under two classical pharmaceutic settings:

1) **Antibiotic Discovery** (Stokes et al., 2020) This task aims to identify molecules that inhibit the growth of *E. coli*, a bacterium canonically used for testing antibiotic activity.
2) **SARS Inhibition** (Tokars & Mesecar, 2021). This task is to identify molecules that inhibit the 3CL protease of SARS-CoV, the pathogen to a respiratory pandemic during the 2000s.

For antibiotic discovery (likewise for SARS Inhibition), the inhibition scores of all the reference ligands in the CrossDocked test set are evaluated via a pretrained ensemble model (Yang et al., 2019). We select the top 5% among them with the highest inhibition scores, denoted as $\{\mathcal{R}_a\}_{a\in I}$, where $I$ is the index set of all the selected ligands. The latent distribution for the desired property $\mathcal{P}$ is defined as $\mathcal{N}(\boldsymbol{\mu}_{\mathcal{P}}, \boldsymbol{\Sigma}_{\mathcal{P}})$, which can be naturally parameterized as:

$$(\boldsymbol{\mu}_{\mathcal{P}}, \boldsymbol{\Sigma}_{\mathcal{P}}) = \frac{1}{|I|}\sum_{a\in I}(\boldsymbol{\mu}_{\mathcal{R}_a}, \boldsymbol{\Sigma}_{\mathcal{R}_a}). \tag{32}$$

Intuitively, this definition of $(\boldsymbol{\mu}_{\mathcal{P}}, \boldsymbol{\Sigma}_{\mathcal{P}})$ contains inductive bias for the desired property $\mathcal{P}$, and ligands that are sampled from this distribution should be more likely to to possess property $\mathcal{P}$.

Results for the two case studies are presented in Table 4, which clearly shows that the latent $\mathcal{P}$ is effective in terms of manipulating desired properties of the generated molecules.

Table 4: Controllable Generation for Antibiotic Discovery and SARS Inhibition.

| **Avg. Inhibition(%)** | **w/ latent $\mathcal{P}$** | **w/o latent $\mathcal{P}$** |
|---|---|---|
| Antibiotic | **3.26** | 1.03 |
| SARS | **28.3** | 11.5 |

## 6 CONCLUDING REMARKS

We proposed GraphVF, a novel variational flow-based framework for controllable binding 3D molecule generation. We empirically demonstrated that, through effectively integrating 2D structure semantics and 3D pocket geometry, GraphVF obtained superior performance to the state-of-the-art strategies for pocket-based 3D molecule generation. We also experimentally showed that, GraphVF can effectively generate binding molecules with desired ligand sub-structures and bio-chemical properties.

We here demonstrate that domain constraints can be effectively leveraged by deep generative models to improve the qualities of molecule design and fulfill the needs for controllable molecule generation. Our studies here shed light on the potential of generating binding ligands with sophisticated domain knowledge and finer-grained control over a variety of bio-chemical properties.

## REFERENCES


Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*, 2014.

Laurent Dinh, David Krueger, and Yoshua Bengio. Nice: Non-linear independent components estimation. *arXiv preprint arXiv:1410.8516*, 2014.

Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real nvp. *arXiv preprint arXiv:1605.08803*, 2016.

William J Egan. Predicting adme properties in drug discovery. *Drug Design: Structure-and Ligand-Based Approaches*, pp. 165–177, 2010.

Paul G Francoeur, Tomohide Masuda, Jocelyn Sunseri, Andrew Jia, Richard B Iovanisci, Ian Snyder, and David R Koes. Three-dimensional convolutional neural networks and a cross-docked data set for structure-based drug design. *Journal of chemical information and modeling*, 60(9):4200–4215, 2020.

Juan S Gomez-Jeria and Andres Robles-Navarro. A note on the docking of some hallucinogens to the 5-ht2a receptor. *Journal of Computational Methods in Molecular Design*, 5(1):45–57, 2015.

Emiel Hoogeboom, Victor Garcia Satorras, Clement Vignac, and Max Welling. Equivariant diffusion for molecule generation in 3d. In *International Conference on Machine Learning*, pp. 8867–8887. PMLR, 2022.

Wengong Jin, Regina Barzilay, and Tommi Jaakkola. Junction tree variational autoencoder for molecular graph generation. In *International conference on machine learning*, pp. 2323–2332. PMLR, 2018.

Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *2nd International Conference on Learning Representations*, 2013.

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

Meng Liu, Youzhi Luo, Kanji Uchino, Koji Maruhashi, and Shuiwang Ji. Generating 3d molecules for target protein binding. *arXiv preprint arXiv:2204.09410*, 2022.

Shitong Luo, Jiaqi Guan, Jianzhu Ma, and Jian Peng. A 3d generative model for structure-based drug design. *Advances in Neural Information Processing Systems*, 34:6229–6239, 2021.

A. T. McNutt, P. Francoeur, R. Aggarwal, T. Masuda, R. Meli, M. Ragoza, J. Sunseri, and D. R. Koes. Gnina 1.0: Molecular docking with deep learning. *Journal of cheminformatics*, 2021.

George Papamakarios, Theo Pavlakou, and Iain Murray. Masked autoregressive flow for density estimation. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pp. 2335–2344, 2017.

Xingang Peng, Shitong Luo, Jiaqi Guan, Qi Xie, Jian Peng, and Jianzhu Ma. Pocket2mol: Efficient molecular sampling based on 3d protein pockets. *arXiv preprint arXiv:2205.07249*, 2022.

Danilo Rezende and Shakir Mohamed. Variational inference with normalizing flows. In *International Conference on Machine Learning*, pp. 1530–1538, 2015.

Victor Garcia Satorras, Emiel Hoogeboom, and Max Welling. E (n) equivariant graph neural networks. In *International conference on machine learning*, pp. 9323–9332. PMLR, 2021.

K. T. Schutt, P.-J. Kindermans, H. E. Sauceda, S. Chmiela, A. Tkatchenko, and K.-R. Muller. Schnet: A continuous-filter convolutional neural network for modeling quantum interactions. *arXiv preprint arXiv:1706.08566*, 2017.

Chence Shi, Minkai Xu, Zhaocheng Zhu, Weinan Zhang, Ming Zhang, and Jian Tang. Graphaf: a flow-based autoregressive model for molecular graph generation. *arXiv preprint arXiv:2001.09382*, 2020.

Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, pp. 2256–2265. PMLR, 2015.

Jonathan M Stokes, Kevin Yang, Kyle Swanson, Wengong Jin, Andres Cubillos-Ruiz, Nina M Donghia, Craig R MacNair, Shawn French, Lindsey A Carfrae, Zohar Bloom-Ackermann, et al. A deep learning approach to antibiotic discovery. *Cell*, 180(4):688–702, 2020.

Valerie Tokars and Andrew Mesecar. Qfret-based primary biochemical high throughput screening assay to identify inhibitors of the sars coronavirus 3c-like protease (3clpro), 2021.

Aaron Van Oord, Nal Kalchbrenner, and Koray Kavukcuoglu. Pixel recurrent neural networks. In *International Conference on Machine Learning*, pp. 1747–1756, 2016.

Lilian Weng. Flow-based deep generative models. *lilianweng. github. io/lil-log*, 2018.

C. G. Wermuth, C. R. Ganellin, P. Lindberg, and L. A. Mitscher. Glossary of terms used in medicinal chemistry (iupac recommendations 1998). *Pure and Applied Chemistry*, 70(5):1129–1143, 1998. doi: doi:10.1351/pac199870051129. URL https://doi.org/10.1351/pac199870051129.

Kevin Yang, Kyle Swanson, Wengong Jin, Connor Coley, Philipp Eiden, Hua Gao, Angel Guzman-Perez, Timothy Hopper, Brian Kelley, Miriam Mathea, et al. Analyzing learned molecular representations for property prediction. *Journal of chemical information and modeling*, 59(8):3370–3388, 2019.

Cheng Zhang, Judith Butepage, Hedvig Kjellstrom, and Stephan Mandt. Advances in variational inference. *IEEE transactions on pattern analysis and machine intelligence*, 41(8):2008–2026, 2018.

Jinhua Zhu, Yingce Xia, Chang Liu, Lijun Wu, Shufang Xie, Tong Wang, Yusong Wang, Wengang Zhou, Tao Qin, Houqiang Li, et al. Direct molecular conformation generation. *arXiv preprint arXiv:2202.01356*, 2022.