TAGMOL: Target Aware Gradient-guided Molecule Generation

Anonymous Authors[1]

Abstract

3D generative models have shown significant promise in *structure-based drug design (SBDD)*, particularly in discovering ligands tailored to specific target binding sites. Existing algorithms often focus primarily on ligand-target binding, characterized by binding affinity. Moreover, models trained solely on target-ligand distribution may fall short in addressing the broader objectives of drug discovery, such as the development of novel ligands with desired properties like drug-likeness, and synthesizability, underscoring the multi-faceted nature of the drug design process. To overcome these challenges, we decouple the problem into molecular generation and property prediction. The latter synergistically *guides* the diffusion sampling process, facilitating guided diffusion and resulting in the creation of meaningful molecules with the desired properties. We call this guided molecular generation process as TAGMOL. Through experiments on benchmark datasets, TAGMOL demonstrates superior performance compared to state-of-the-art baselines, achieving a 22% improvement in average Vina Score and yielding favorable outcomes in essential auxiliary properties. This establishes TAGMOL as a comprehensive framework for drug generation.

1. Introduction

The presence of molecular data featuring 3D spatial information has paved the way for the complex field of *structure-based drug design (SBDD)* (Anderson, 2003). The advent of generative AI for molecules has accelerated the rational drug-design—in contrast to the traditional Edisonian trial-and-error approach—with the goal of creating drug-like molecules in 3D space that effectively bind to specific targets. Specifically, deep generative models, such as those proposed by Luo et al. (2021); Liu et al. (2022); Peng et al. (2022), autoregressively generate atoms and bonds, while Zhang et al. (2023) generate motifs. Despite their progress, the performance of autoregressive models heavily relies on the order of generation, as they condition on previously generated atoms, which can lead to error propagation. Alternatively, diffusion models (Sohl-Dickstein et al., 2015; Ho et al., 2020) overcome this limitation by conditioning upon all the atoms simultaneously and there by efficiently generating realistic molecules that demonstrate stronger binding affinities with their intended targets.

The effectiveness of these generative models is heavily reliant on how well training datasets — consisting of protein-ligand complexes — align with desired properties such as binding affinity, drug-like topological features, synthetic accessibility (ease of synthesis), to name a few. For instance, CrossDocked 2020 (Francoeur et al., 2020), a widely-used training dataset for SBDD tasks, predominantly includes complexes with moderate binding affinities. Consequently, models trained solely on such datasets may yield sub-optimal molecules, intrinsically tying their success closely to the dataset's quality. Moreover, drug generation is a multi-faceted process that encompasses not only binding affinity but also a range of other desired properties. Compiling a comprehensive dataset that encompasses a broad range of desired properties poses significant challenges, particularly associated with the high computational costs of assessing a suite of properties. Additionally, refining datasets to meet specific quality constraints can drastically reduce the volume of usable training data. As the number of constraints grows, the likelihood of finding samples that satisfy all these criteria becomes increasingly difficult, further exacerbating the situation. Consequently, generative models trained on lower-quality, but high-volume, data may unintentionally capture suboptimal signals, leading to diminished performance in the context of drug discovery. This situation prompts an exploration into how, during the denoising phase of a generative model, we can effectively introduce desired signals while ensuring meaningful reconstruction.

To address these challenges, we introduce TAGMOL– (Target-Aware Gradient-guided Molecule Generation), wherein we decouple molecular generation and property prediction. We start by training a time-dependent *guide* model that predicts properties from inputs with noise levels similar to those in the base diffusion model. Crucially, we turn the challenge of using inferior quality data, i.e., the property of interest is well spread with the inclusion of sub-optimal values in the dataset, to our strength for robust *guide* training. Inspired by classifier guidance in diffusion models (Dhariwal & Nichol, 2021), we use the gradient of *guide* to direct the latent space during the diffusion sampling process, ensuring the reconstructed molecules possess the targeted properties. In the sampling phase, we harness the strengths of both the generative model and the *guide*. This interactive dynamic enables us to explore regions with superior properties while simultaneously denoising the latent space to generate diverse molecules. Our method marks a significant advancement in end-to-end molecular generation, reducing the reliance on post-optimization techniques. Furthermore, to address real-world scenarios where multiple property constraints exist, we train separate *guides* for each property, subsequently employing them to steer the diffusion process effectively.

Overall, the key contributions of our work are as follows.

- **Reformulation of the drug-discovery problem:** We reformulate the problem of drug generative modeling moving beyond the myopic lens of optimizing binding activity. The need to optimize other properties of interest, even when these signals are not adequately present in the train set, is explicitly coded into our problem formulation.
- **Algorithm design:** We design a novel generative process, called TAGMol, which jointly leverages the signals from two different components: an *SE(3) equivariant graph diffusion model* to mimic geometries of the train set, and a *multi-objective guide model*, empowered by an SE(3) invariant GNN, to steer the exploration region of diffusion sampling towards the property of interest by leveraging gradients.
- **Rigorous empirical evaluation:** We demonstrate that our model achieves 22% improvement in average Vina Score, all the while being guided by considerations of binding affinity and crucial pharmacological properties such as QED and SA.

2. Preliminary

In this section, we introduce the preliminary concepts central to our work and formulate the problem of gradient-guided molecular generation. All notations introduced in this section are summarized in Table 4 in the Appendix.

2.1. Problem Statement

From the standpoint of generative models, SBDD task involves the creation of ligand molecules with the ability to bind effectively to a specified protein binding site. A protein binding site is characterized by a set of atoms, denoted by $\mathcal{P} = \{(\mathbf{x}_i^{\mathcal{P}}, \mathbf{v}_i^{\mathcal{P}})\}_{i=1}^{N_{\mathcal{P}}}$, where $N_{\mathcal{P}}$ represents the number of protein atoms, $\mathbf{x}_i^{\mathcal{P}} \in \mathbb{R}^3$ represents the 3D coordinates of the protein atom, and $\mathbf{v}_i^{\mathcal{P}} \in \mathbb{R}^{N_f}$ represents protein atom features such as element types and amino acid types, with N_f representing the number of such features. We aim to jointly optimize binding affinity and desired pharmacological properties (denoted as $\mathbb{Y}$), by generating prospective ligands and molecules $\mathcal{M} = \{(\mathbf{x}_i^{\mathcal{M}}, \mathbf{v}_i^{\mathcal{M}})\}_{i=1}^{N_{\mathcal{M}}}$, for a given protein $\mathcal{P}$. Here $\mathbf{x}_i^{\mathcal{M}} \in \mathbb{R}^3$ and $\mathbf{v}_i^{\mathcal{M}} \in \mathbb{R}^K$ represents the atom coordinates and atom types of a ligand molecule, respectively, with K representing the number of such features to represent atom types. The variable $N_{\mathcal{M}}$ signifies the number of atoms in the ligand molecule, which can be sampled during inference utilizing either an empirical distribution (Hoogeboom et al., 2022; Guan et al., 2023a;b) or predicted through a neural network (Liu et al., 2022).

To simplify, in matrix representation the ligand molecule is denoted as $\mathbf{M} = [\mathbf{X}^{\mathcal{M}}, \mathbf{V}^{\mathcal{M}}]$, where $\mathbf{X}^{\mathcal{M}} \in \mathbb{R}^{N_{\mathcal{M}} \times 3}$ and $\mathbf{V}^{\mathcal{M}} \in \mathbb{R}^{N_{\mathcal{M}} \times K}$ and the protein is denoted as $\mathbf{P} = [\mathbf{X}^{\mathcal{P}}, \mathbf{V}^{\mathcal{P}}]$, where $\mathbf{X}^{\mathcal{P}} \in \mathbb{R}^{N_{\mathcal{P}} \times 3}$ and $\mathbf{V}^{\mathcal{P}} \in \mathbb{R}^{N_{\mathcal{P}} \times N_f}$.

2.2. Diffusion models for Target-Aware Generation

As delineated in previous works on diffusion-based target-aware molecular generations (Guan et al., 2023a;b), the process involves two phases: *noise injection*, also termed as forward diffusion, and *denoising* (backward diffusion).

Noise injection: This phase involves a gradual injection of Gaussian noise for co-ordinates and uniform noise for categorical data through a Markov chain. This noise addition is uniquely applied to the ligand molecule, excluding the protein in the diffusion process. In this context, the atom positions and types of the ligand molecule at time step t are represented as $\mathbf{X}_t^{\mathcal{M}}$ and $\mathbf{V}_t^{\mathcal{M}}$ respectively. The diffusion forward transition is defined by the following equations:

$$q(\mathbf{M}_t|\mathbf{M}_{t-1}, \mathbf{P}) = \mathcal{N}\left(\mathbf{X}_t^{\mathcal{M}}; \sqrt{1-\beta_t}\mathbf{X}_{t-1}^{\mathcal{M}}, \beta_t\mathbf{I}\right) \cdot \mathcal{C}\left(\mathbf{V}_t^{\mathcal{M}} | (1-\beta_t)\mathbf{V}_{t-1}^{\mathcal{M}} + \beta_t/K\right) \tag{1}$$

$$q(\mathbf{M}_t|\mathbf{M}_0, \mathbf{P}) = \mathcal{N}\left(\mathbf{X}_t^{\mathcal{M}}; \sqrt{\bar{\alpha}_t}\mathbf{X}_0^{\mathcal{M}}, (1-\bar{\alpha}_t)\mathbf{I}\right) \cdot \mathcal{C}\left(\mathbf{V}_t^{\mathcal{M}} | \bar{\alpha}_t\mathbf{V}_0^{\mathcal{M}} + (1-\bar{\alpha}_t)/K\right) \tag{2}$$

where $\mathcal{N}$ and $\mathcal{C}$ denotes the normal and categorical distribution respectively while $\beta_1, ..., \beta_T$ represents the variance schedules. The corresponding posteriors are analytically derived as follows:

$$q(\mathbf{M}_{t-1}|\mathbf{M}_0, \mathbf{M}_t, \mathbf{P}) = \mathcal{N}\left(\mathbf{X}_{t-1}^{\mathcal{M}}; \tilde{\mu}_t\left(\mathbf{X}_t^{\mathcal{M}}, \mathbf{X}_0^{\mathcal{M}}\right), \tilde{\beta}_t\mathbf{I}\right) \cdot \mathcal{C}\left(\mathbf{V}_{t-1}^{\mathcal{M}} | \tilde{\boldsymbol{c}}_t\left(\mathbf{V}_t^{\mathcal{M}}, \mathbf{V}_0^{\mathcal{M}}\right)\right) \tag{3}$$

where, $\tilde{\mu}_t\left(\mathbf{X}_t^{\mathcal{M}}, \mathbf{X}_0^{\mathcal{M}}\right) = \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t}\mathbf{X}_0^{\mathcal{M}} + \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}\mathbf{X}_t^{\mathcal{M}}$, $\tilde{\beta}_t = \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t$, $\alpha_t = 1 - \beta_t$, $\bar{\alpha}_t = \Pi_{s=1}^t \alpha_s$, $\tilde{c}_t\left(\mathbf{V}_t^{\mathcal{M}}, \mathbf{V}_0^{\mathcal{M}}\right) = \boldsymbol{c}^\star / \sum_{k=1}^K c_k^\star$, and $\boldsymbol{c}^\star\left(\mathbf{V}_t^{\mathcal{M}}, \mathbf{V}_0^{\mathcal{M}}\right) = [\alpha_t\mathbf{V}_t^{\mathcal{M}} + (1-\alpha_t)/K] \odot [\bar{\alpha}_{t-1}\mathbf{V}_0^{\mathcal{M}} + (1-\bar{\alpha}_{t-1})/K]$.

In practical applications, it is recognized that the schedules β_t for coordinates and categories may differ. However, for the sake of simplicity in this context, they are uniformly represented.

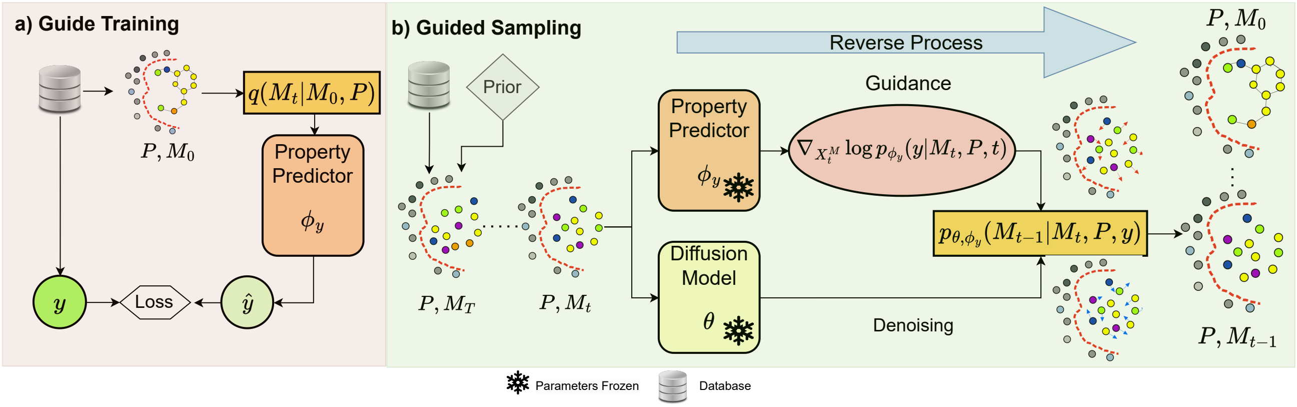

Figure 1. Overview of TAGMOL. (a) Training a property-oriented *guide* using existing data. (b) Utilizing the trained *guide* and *diffusion* model to steer the diffusion sampling process towards the optimal regions of the property of interest.

Denoising phase: In the generative process, a neural network parameterized by θ learns to recover $\mathbf{M}_0$ by iteratively denoising $\mathbf{M}_T$. During reverse process, $\mathbf{M}_0$ is approximated using θ and $\mathbf{M}_{t-1}$ by predicting $\widehat{\mathbf{M}}_{0|t} = [\widehat{\mathbf{X}}^{\mathcal{M}}_{0|t}, \widehat{\mathbf{V}}^{\mathcal{M}}_{0|t}]$ at time step t and $\mathbf{M}_{t-1}$ is sampled as follows:

$$\begin{aligned} p_\theta\left(\mathbf{M}_{t-1}|\mathbf{M}_t, \mathbf{P}\right) &= q\left(\mathbf{M}_{t-1}|\widehat{\mathbf{M}}_{0|t}, \mathbf{M}_t, \mathbf{P}\right) \\ &= \mathcal{N}\left(\mathbf{X}^{\mathcal{M}}_{t-1}; \tilde{\mu}_\theta\left(\mathbf{M}_t, \mathbf{P}, t\right), \tilde{\beta}_t \mathbf{I}\right) \\ &\quad \cdot \mathcal{C}\left(\mathbf{V}^{\mathcal{M}}_{t-1}|\tilde{c_\theta}\left(\mathbf{M}_t, \mathbf{P}, t\right)\right) \qquad (4) \\ &= \mathcal{N}\left(\mathbf{X}^{\mathcal{M}}_{t-1}; \tilde{\mu}_t\left(\mathbf{X}^{\mathcal{M}}_t, \hat{\mathbf{X}}^{\mathcal{M}}_{0|t}\right), \tilde{\beta}_t \mathbf{I}\right) \\ &\quad \cdot \mathcal{C}\left(\mathbf{V}^{\mathcal{M}}_{t-1}|\tilde{c_t}\left(\mathbf{V}^{\mathcal{M}}_t, \hat{\mathbf{V}}^{\mathcal{M}}_{0|t}\right)\right) \end{aligned}$$

Training: In line with the principles outlined in the variational auto-encoder (Kingma & Welling, 2013), the model undergoes training through the optimization of variational bound on the negative log-likelihood. Given that both $q\left(\mathbf{X}_{t-1}|\mathbf{M}_0, \mathbf{M}_t, \mathbf{P}\right)$ and $p_\theta\left(\mathbf{X}_{t-1}|\mathbf{M}_t, \mathbf{P}\right)$ are Gaussian distributions, the Kullback-Leibler (KL) divergence for the atom coordinates is expressed in a closed-form equation:

$$\begin{aligned} L^{(x)}_{t-1} &= \frac{1}{2\tilde{\beta}_t} \|\tilde{\mu}_t\left(\mathbf{X}^{\mathcal{M}}_t, \mathbf{X}^{\mathcal{M}}_0\right) - \tilde{\mu}_\theta\left([\mathbf{X}^{\mathcal{M}}_t, \mathbf{V}^{\mathcal{M}}_t], \mathbf{P}, t\right)\|^2 + C \\ &= \gamma_t \|\mathbf{X}^{\mathcal{M}}_0 - \widehat{\mathbf{X}}^{\mathcal{M}}_{0|t}\|^2 + C \end{aligned} \qquad (5)$$

where $\gamma_t = \frac{\bar{\alpha}_{t-1}\beta_t^2}{2\sigma_t^2(1-\bar{\alpha}_t)^2}$ and C is a constant. As recommended by (Ho et al., 2020) and (Guan et al., 2023a), training the model using an unweighted Mean Squared Error (MSE) loss, by setting $\gamma_t = 1$, leads to enhanced performance. Regarding the atom-type loss, the KL divergence of categorical distributions is computed in the following manner:

$$L^{(v)}_{t-1} = \sum_{k=1}^{K} \boldsymbol{c}\left(\mathbf{V}^{\mathcal{M}}_t, \mathbf{V}^{\mathcal{M}}_0\right)_k \log \frac{\boldsymbol{c}\left(\mathbf{V}^{\mathcal{M}}_t, \mathbf{V}^{\mathcal{M}}_0\right)_k}{\boldsymbol{c}\left(\mathbf{V}^{\mathcal{M}}_t, \widehat{\mathbf{V}}^{\mathcal{M}}_{0|t}\right)_k} \qquad (6)$$

The final loss is a weighted sum of atom coordinate loss and atom type loss, which is expressed as $L = L^{(x)}_{t-1} + \lambda L^{(v)}_{t-1}$.

3. TAGMOL

In this work, we improve upon *Conditional Diffusion models* with a particular emphasis on utilizing protein pockets as a conditioning factor for generating ligand molecules (Guan et al., 2023a;b). The effectiveness of such models in practical scenarios can be hindered if the conditioning signal is overlooked or weakened, an issue that becomes more pronounced with datasets of inferior quality. Models that are exclusively trained to maximize the likelihood of protein-ligand complexes in such datasets naturally inherit the same quality issues. To counteract this, TAGMOL employs a strategic approach to explicitly integrate additional conditioning signals learned over a set of binding and desirable pharmacological properties $\mathbb{Y}$ (Recall § 2.1) during the model's denoising phase. This tactic aligns with the method proposed by (Dhariwal & Nichol, 2021), which involves conditioning a pre-trained diffusion model using classifier gradients. In particular, for a property $\mathbf{y} \in \mathbb{Y}$, we train a regressor (or classifier as appropriate) $p_\phi(\mathbf{y}|\mathbf{M}_t, \mathbf{P}, t)$ on noisy molecule $\mathbf{M}_t$ and then use the gradients $\nabla_{\mathbf{X}^{\mathcal{M}}_t} p_\phi(\mathbf{y}|\mathbf{M}_t, \mathbf{P}, t)$ to guide the diffusion sampling process towards the desirable properties encoded in $\mathbf{y}$.

Fig 1 presents the pipeline of TAGMOL. TAGMOL consists of two key components. The first component is *Guide Training*, introduced in § 3.1, where we present our *SE(3) Invariant* GNN architecture and its modeling objectives, establishing the foundation for our approach. The goal of this step is to assess how well a molecule conforms to the desirable properties $\mathbb{Y}$. The second component, discussed in § 3.2, is *Guided Sampling*. Here, we elaborate on how a trained guide model efficiently steers the diffusion sampling process towards specific regions of interest, emphasizing desired properties. Our study goes further to demonstrate the concurrent optimization of multiple properties, as detailed in § 3.3. This multifaceted approach more holistically addresses the various aspects of molecular design, ultimately simplifying the efforts of chemists devote to the lead optimization process.

3.1. Property Guide

We approach the input space as a 3D point cloud system where we build a k-nearest neighbors (k-NN) graph $\mathcal{G}$ by representing ligand and protein atoms as nodes, and each atom is connected to its k-nearest neighbors. For properties such as binding affinity, including protein atoms within the graph is critical; however, the inclusion of protein atoms can be omitted for properties solely dependent on the ligand. We parameterize our guide model using an Invariant GNN ϕ_y, which is trained to predict property $\mathbf{y} \in \mathbb{Y}$ given a noisy input $\mathbf{M}_t$, protein $\mathbf{P}$ and time t. Later in the denoising phase, the gradients of ϕ_y are used to direct exploration in regions of interest. Given the graph representation $\mathcal{G}$ at diffusion time step t, we define the GNN convolution layer as follows:

$$\begin{aligned}
\mathbf{d}_{i,j} &= \mathcal{D}(\|\mathbf{x}_{i,t} - \mathbf{x}_{j,t}\|^2) \\
\mathbf{m}_{i,j} &= \phi^m_{\mathbf{y}}(\mathbf{h}^l_{i,t}, \mathbf{h}^l_{j,t}, \mathbf{d}_{i,j}, \mathbf{a}_{i,j}) \\
\mathbf{e}_{i,j} &= \phi^e_{\mathbf{y}}(\mathbf{m}_{i,j}) \\
\mathbf{m}_i &= \sum_{j \in \mathcal{N}(i)} \mathbf{e}_{i,j}\mathbf{m}_{i,j} \\
\mathbf{h}^{l+1}_{i,t} &= \phi^h_{\mathbf{y}}(\mathbf{h}^l_{i,t}, \mathbf{m}_i) \\
\mathbf{h}^0_{i,t} &= \begin{cases} \phi^p_{\mathbf{y}}(\mathbf{v}^{\mathcal{P}}_i) & \text{if } i \in \mathcal{P} \\ \phi^l_{\mathbf{y}}([\mathbf{v}^{\mathcal{M}}_{i,t}, \tau]) & \text{if } i \in \mathcal{M} \end{cases}
\end{aligned} \tag{7}$$

where $\mathbf{h}^l_{i,t} \in \mathbb{R}^d$ represents the SE(3)-invariant hidden representation of protein and ligand atoms after l layers; $\mathbf{x}_{i,t}, \mathbf{x}_{j,t} \in \mathbb{R}^3$ represents hidden representation of protein and ligand atoms. $\mathbf{d}_{i,j}$ represents the distance embedding and $\mathbf{a}_{i,j}$ represents the edge attributes in the graph. Invariance of $\mathbf{h}^l_{i,t}$ stems from the fact that we take the L2-distance between the atom representations. $\mathcal{N}(i)$ stands for the set of neighbors for atom i. τ represents the time embedding to make the model aware of noise at time step t. $\phi^m_{\mathbf{y}}$, $\phi^e_{\mathbf{y}}$, $\phi^h_{\mathbf{y}}$, $\phi^p_{\mathbf{y}}$ and $\phi^l_{\mathbf{y}}$ are Multi-Layer Perceptrons (MLP) where as $\mathcal{D}$ is a distance encoder.

Once we get the final hidden states $\mathbf{h}^L_{i,t}$, we predict the final property using an MLP layer as:

$$\hat{\mathbf{y}} = \phi^f_{\mathbf{y}}\left(\sum_{i \in \mathcal{M}} \mathbf{h}^L_{i,t}\right) \tag{8}$$

While equivariance is essential for generative models to maintain the consistency of the probability $p(\mathbf{M}_0|\mathbf{P})$ against *SE(3)* transformations in protein-ligand complexes, it is also vital that a scalar property predicted from *guide* remains independent of *SE(3)* transformations. Hence, the GNN in guide is SE(3) invariant. This modeling approach introduces a valuable inductive bias, which is particularly beneficial in the generation of 3D objects, such as molecules (Köhler et al., 2020; Satorras et al., 2021; Xu et al., 2022; Hoogeboom et al., 2022; Guan et al., 2023a;b).

In our approach, diverging from the classifier-guided methodology (Dhariwal & Nichol, 2021) in diffusion processes where the classifier directly estimates probabilities, we train our model to regress specific properties. The probability modeling is articulated through a normal distribution as:

$$p_{\phi_y}(\mathbf{y}|\mathbf{M}_t, \mathbf{P}) = \mathcal{N}(\mathbf{y}, \phi_{\mathbf{y}}(\mathbf{M}_t, \mathbf{P}, t), \mathbf{I}) \tag{9}$$

The training of these models is oriented toward minimizing the Negative log-likelihood (NLL), which effectively simplifies the final objective to minimizing the Root Mean Square Error (RMSE) between $\mathbf{y}$ and $\phi_{\mathbf{y}}(\mathbf{M}_t, \mathbf{P}, t)$.

$$\begin{aligned}
NLL &= -\mathbb{E}_{p(\mathbf{P}, \mathbf{M}_{0:T})} \sum_{t=0}^{T} \log(p_{\phi_y}(\mathbf{y}|\mathbf{M}_t, \mathbf{P}, t)) \\
&= \mathbb{E}_{p(\mathbf{P}, \mathbf{M}_{0:T})} \sum_{t=0}^{T} \frac{(\mathbf{y} - \phi_{\mathbf{y}}(\mathbf{M}_t, \mathbf{P}, t))^2}{2}
\end{aligned} \tag{10}$$

Although we designed our guide for regression tasks, it is versatile enough to be adapted for classification tasks or other suitable applications.

3.2. Single Objective Guidance

In this section, we first design the framework to obtain guidance from a single property. Subsequently, in § 3.3, we extend to multi-objective guidance.

Departing from the methodologies outlined in prior studies (Guan et al., 2023a;b), our generative model distinctively conditions on both the protein pocket and property to be guided. Remarkably, without retraining the diffusion model, we guide the diffusion sampling process by shifting coordinates. This approach empowers the model to consistently condition on optimal property values during the denoising phase. By steering the denoising process towards an optimal property, it significantly enhances the model's capability to generate molecules with desired characteristics.

As demonstrated in (Dhariwal & Nichol, 2021), efficiently sampling $\mathbf{M}_{t-1}$ for each denoising transition can be adequately achieved by

$$\begin{aligned}
p_{\theta,\phi_{\mathbf{y}}}(\mathbf{M}_{t-1}|\mathbf{M}_t, \mathbf{P}, \mathbf{y}) &= Z p_\theta(\mathbf{M}_{t-1}|\mathbf{M}_t, \mathbf{P}) \\
&\quad \cdot p_{\phi_{\mathbf{y}}}(\mathbf{y}|\mathbf{M}_{t-1}, \mathbf{P}, t-1)
\end{aligned} \tag{11}$$

where Z is a normalizing constant. However, direct sampling from this distribution is intractable. We approximate the sampling process via perturbation to a Gaussian distribution as per prior work (Sohl-Dickstein et al., 2015; Dhariwal & Nichol, 2021). Thus, we use perturbation to $p_\theta(\mathbf{X}_{t-1}|\mathbf{M}_t, \mathbf{P})$ and sample $\mathbf{X}^{\mathcal{M}}_{t-1}$ as:

$$\begin{aligned}&\mathbf{X}^{\mathcal{M}}_{t-1} \sim \mathcal{N}(\tilde{\mu}_\theta(\mathbf{M}_t, \mathbf{P}, t) + s\tilde{\beta}\nabla_{\mathbf{X}^{\mathcal{M}}_t} \log p_{\phi_{\mathbf{y}}}(\mathbf{y}|\mathbf{M}_t, \mathbf{P}, t), \tilde{\beta}\mathbf{I})\\ &\mathbf{V}^{\mathcal{M}}_{t-1} \sim \mathcal{C}\left(\tilde{\boldsymbol{c}}_\theta\left(\mathbf{M}_t, \mathbf{P}, t\right)\right)\end{aligned} \tag{12}$$

In this context, the parameter s, denoting the guidance strength, plays a crucial role in prioritizing $\mathbf{M}_t$ with optimal property by sampling it from the updated distribution $\propto\ p_\theta(\mathbf{M}_{t-1}|\mathbf{M}_t, \mathbf{P})\left(p_{\phi_{\mathbf{y}}}(\mathbf{y}|\mathbf{M}_{t-1}, \mathbf{P}, t)\right)^s$. Fine-tuning of s is essential during the optimization process to improve property prediction while maintaining the effectiveness of denoising within the diffusion model.

3.3. Multi Objective Guidance

We now generalize our objective to holistically enhance various desired properties, collectively denoted as $\mathbb{Y}$, in the molecules we generate. At each denoising step t, we condition on $\mathbb{Y}$ and sample $\mathbf{M}_{t-1}$ according to the probability distribution similar to Equation 11 as :

$$\begin{aligned}p_{\theta,\phi_{\mathbb{Y}}}(\mathbf{M}_{t-1}|\mathbf{M}_t, \mathbf{P}, \mathbb{Y}) = Z p_\theta(\mathbf{M}_{t-1}|\mathbf{M}_t, \mathbf{P})\\ \cdot\, p_{\phi_{\mathbb{Y}}}(\mathbb{Y}|\mathbf{M}_{t-1}, \mathbf{P}, t)\end{aligned} \tag{13}$$

Assuming all the properties $\mathbf{y} \in \mathbb{Y}$ are conditionally independent given $\mathbf{M}_{t-1}$ and $\mathbf{P}$, the above Equation 13 can be factorized as follows:

$$\begin{aligned}p_{\theta,\phi_{\mathbb{Y}}}(\mathbf{M}_{t-1}|\mathbf{M}_t, \mathbf{P}, \mathbb{Y}) = Z p_\theta(\mathbf{M}_{t-1}|\mathbf{M}_t, \mathbf{P})\\ \cdot \prod_{\mathbf{y}\in\mathbb{Y}} p_{\phi_{\mathbf{y}}}(\mathbf{y}|\mathbf{M}_{t-1}, \mathbf{P}, t)\end{aligned} \tag{14}$$

Thus we train a set of models $\phi_{\mathbb{Y}} = \{\phi_{\mathbf{y}} : \forall \mathbf{y} \in \mathbb{Y}\}$ independently and sample $\mathbf{M}_{t-1}$ using a similar approximated posterior distribution in Equation 12 as:

$$\begin{aligned}&\mathbf{X}^{\mathcal{M}}_{t-1} \sim \mathcal{N}(\tilde{\mu}_\theta(\mathbf{M}_t, \mathbf{P}, t) + \delta, \tilde{\beta}\mathbf{I})\\ &\mathbf{V}^{\mathcal{M}}_{t-1} \sim \mathcal{C}\left(\tilde{\boldsymbol{c}}_\theta\left(\mathbf{M}_t, \mathbf{P}, t\right)\right)\end{aligned} \tag{15}$$

where

$$\delta = \sum_{\mathbf{y}\in\mathbb{Y}} s_{\mathbf{y}}\tilde{\beta}\nabla_{\mathbf{X}^{\mathcal{M}}_t} \log p_{\phi_{\mathbf{y}}}(\mathbf{y}|\mathbf{M}_t, \mathbf{P}, t) \tag{16}$$

and $s_{\mathbf{y}}$ represents the guidance strength for property $\mathbf{y} \in \mathbb{Y}$.

4. Experiments

In this section, we benchmark TAGMOL against state-of-the-art molecular generative models and establish that gradient guidance leads to superior performance across an array of important metrics assessing binding and pharmacological properties.

4.1. Datasets

TAGMOL is trained and evaluated on the CrossDocked2020 dataset (Francoeur et al., 2020), consistent with the approaches outlined in Luo et al. (2021); Peng et al. (2022); Guan et al. (2023a;b). The dataset originally comprised of 22.5 million docked binding complexes, which undergo a rigorous refinement process. This refinement narrows down the dataset to better-quality docking poses, ensuring that the root-mean-square deviation (RMSD) between the docked pose and the ground truth is less than 1Å, and that the protein sequences exhibit less than 30% identity. From this refined pool, 100,000 protein-ligand pairs are selected for the training set. For the evaluation, we employ a separate set of 100 proteins that are distinct from those in the training dataset. To ensure a fair comparison with our baseline methods, we adhere to identical data splits for training our *guide* models and evaluating our overall method.

4.2. Baselines

We benchmark against state-of-the-art baselines in the realm of structure-based drug design (SBDD). This includes liGAN (Ragoza et al., 2022), which leverages a conditional variational autoencoder (CVAE) and is trained on a grid representation of atomic densities in protein-ligand structures. Additionally, we consider AR (Luo et al., 2021), and Pocket2Mol (Peng et al., 2022), GNN-based methods both of which employ autoregressive frameworks to generate 3D molecular atoms by conditioning on the protein pocket and previously generated atoms. Furthermore, our comparison extends to recent diffusion-based approaches such as TargetDiff (Guan et al., 2023a) and DecompDiff (Guan et al., 2023b), which have set new benchmarks in the non-autoregressive generation of atom coordinates and types. DecompDiff enhances TargetDiff by integrating bond considerations and introducing decomposed priors for the ligand's arms and scaffolds. For a comprehensive overview of prior works and additional references, please refer to the § B of Appendix.

4.3. Metrics

To evaluate the quality of molecules generated by TAGMOL and the baselines, we adopt a multi-faceted assessment strategy encompassing molecular properties, their conformation, and binding affinity with the target.

4.3.1. MOLECULAR PROPERTIES

For the assessment of molecular characteristics, we calculate the following metrics

QED (Quantitative Estimate of Druglikeness). This metric evaluates a molecule's drug-likeness (Bickerton et al., 2012) by reflecting the typical distribution of molecular properties in successful drug candidates.

SA (Synthetic Accessibility). SA assesses the ease with which a molecule can be synthesized (Ertl & Schuffenhauer, 2009), serving as a vital indicator of its practical manufacturability in a laboratory or industrial setting.

Diversity. Diversity is measured as the average pairwise Tanimoto distances (Bajusz et al., 2015; Tanimoto, 1958) among all ligands generated for a specific protein pocket.

Bond Distance Distribution. We calculate the Jensen-Shannon divergences (JSD) to assess the differences in bond distance distributions between the reference molecules and the generated molecules (Guan et al., 2023a;b).

4.3.2. BINDING AFFINITY

AutoDock Vina(Eberhardt et al., 2021) is employed to calculate the following metrics:

Vina Score. Estimates binding affinity from the 3D pose of generated molecules, where a favorable score suggests strong binding potential.

Vina Min. By conducting a local structure minimization prior to affinity estimation, this metric presents a slightly refined perspective on Vina Score.

Vina Dock. Incorporating a re-docking process, Vina Dock showcases the optimal binding affinity that can be achieved.

High Affinity. This is a comparative metric evaluating the percentage of generated molecules that manifest better binding than a reference molecule for a given protein pocket.

4.3.3. HIT RATE

We report the percentage of molecules meeting *hit criteria* characterized as: QED $\geq$ 0.4, SA $\geq$ 0.5, and Vina Dock $\leq$ -8.18 kcal/mol. These QED, SA thresholds are set in line with the ranges observed in currently marketed drugs (Eckmann et al., 2022). The threshold for the Vina Dock score is chosen to reflect a binding affinity less than 1μm. This threshold is commonly used in medicinal chemistry to ensure moderate biological activity (Yang et al., 2021; Long et al., 2022; Guan et al., 2023b). Optimizing the hyperparameters in TAGMOL for *hit rate* allows us to generate higher number of acceptable molecules as per our criteria.

4.4. Experimental setup

Generative Backbone. For our generative backbone model, we opted for TargetDiff (Guan et al., 2023a) over DecompDiff (Guan et al., 2023b). The decision was based on TargetDiff's self-contained nature, in contrast to DecompDiff, which relies heavily on external computational tools. Specifically, DecompDiff utilizes AlphaSpace2 (Katigbak et al., 2020) for extracting subpockets, which are potential protein binding sites. In fact, our study demonstrates how our guidance mechanism can effectively replace the computational tools employed in DecompDiff.

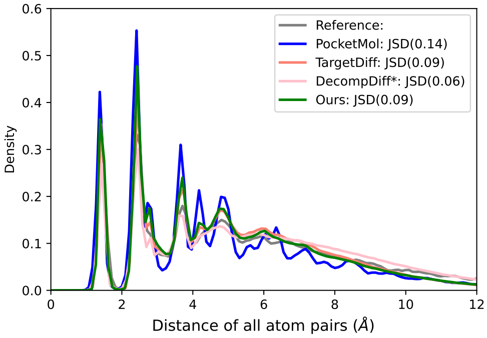

Figure 2. Comparison of distance distributions between all-atom distances of reference molecules in the test set (Reference) and distances in model-generated molecules. The Jensen-Shannon divergence (JSD) between these two distributions is calculated and reported.

Guide Training In our experimental setup, as detailed in § 3.1, we employed a 9-layer Invariant Graph Neural Network (GNN) as the foundational model for constructing *guides* across various properties. Notably, for QED and SA — properties that are not dependent on the target — we adapted the k-Nearest Neighbors (k-NN) graph construction methodology, which omits protein atoms into consideration. The k value was specifically tailored to each property: set at 6 for both QED and SA and increased to 32 for the binding affinity property predictor to optimize the model's performance in different contexts. For training our *binding affinity guide*, we used Autodock Vina scores from the CrossDocked2020 dataset (Eberhardt et al., 2021; Francoeur et al., 2020), and for QED and SA, we calculated scores using the RDKit package (Bento et al., 2020). To effectively guide the denoising phase, we train our guide using the same noise as our backbone model, TargetDiff, which applies Gaussian noise to coordinates and uniform noise to categories. Additional training and evaluation details of *guide* can be found in Appendix § D.

Guide Strengths. For the single objective guidance process, the optimal guide strength $s_{\mathbf{y}}^{opt}$ for each property $\mathbf{y}$ is identified through a grid search. The configuration that delivers the highest value for the intended property is selected. For multi-objective guidance, the optimal guide strength values $s_{\mathbf{y}}^{opt}$ are recalibrated using a set of weights $w_{\mathbf{y}}$, where $\sum_{\mathbf{y}\in\mathbb{Y}} w_{\mathbf{y}} = 1$. These modified guide strengths, $w_{\mathbf{y}} s_{\mathbf{y}}^{\text{opt}}$, are subsequently utilized to steer property optimization as described in Equation 15. In our multi-objective guidance, after an exhaustive evaluation (see Table 7 in Appendix), for optimizing the *hit rate*, we decided to assign equal weights for our final model.

Evaluation Setup. Adhering to the setup outlined in our baseline studies, we generated 100 molecules for each of the

Table 1. Comparison of various properties between reference molecules and those generated by our model and other baselines. (↑) / (↓) indicates whether a larger or smaller number is preferable. The first and second-place results are emphasized with bold and underlined text, respectively. Since we were unable to reproduce the results reported in DecompDiff (Guan et al., 2023b), we reported both our findings and those reported in DecompDiff. They are denoted as "DecompDiff*" and "DecompDiff" respectively.

Methods	Vina Score (↓)		Vina Min (↓)		Vina Dock (↓)		High Affinity (↑)		QED (↑)		SA (↑)		Diversity (↑)		Hit(↑)
	Avg.	Med.	Avg.	Med.	Avg.	Med.	Avg.	Med.	Avg.	Med.	Avg.	Med.	Avg.	Med.	Rate %
Reference	-6.36	-6.46	-6.71	-6.49	-7.45	-7.26	-	-	0.48	0.47	0.73	0.74	-	-	21
liGAN	-	-	-	-	-6.33	-6.20	21.1%	11.1%	0.39	0.39	0.59	0.57	0.66	0.67	13.2
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	12.9
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**	24.3
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**	20.5
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	-
DecompDiff*	-4.85	-6.03	-6.76	-7.09	-8.48	-8.50	64.8%	**78.6%**	0.44	0.41	0.59	0.59	0.63	0.62	24.9
TAGMol	**-7.02**	**-7.77**	**-7.95**	**-8.07**	**-8.59**	**-8.69**	**69.8%**	76.4%	0.55	0.56	0.56	0.56	0.68	0.69	**27.7**

Table 2. Jensen-Shannon Divergence comparing bond distance distributions between reference molecules and generated molecules. Lower values indicate better performance. '-' represents single bonds, '=' represents double bonds, and ':' represents aromatic bonds. The first and second-place results are emphasized with bold and underlined text, respectively

Bond	liGAN	AR	Pocket2 Mol	Target Diff	Decomp Diff*	TAGMol
C-C	0.601	0.609	0.496	**0.369**	0.371	0.384
C=C	0.665	0.620	0.561	0.505	0.539	**0.501**
C-N	0.634	0.474	0.416	0.363	**0.352**	0.365
C=N	0.749	0.635	0.629	**0.550**	0.592	0.559
C-O	0.656	0.492	0.454	0.421	**0.373**	0.422
C=O	0.661	0.558	0.516	0.461	**0.381**	0.430
C:C	0.497	0.451	0.416	0.263	**0.258**	0.269
C:N	0.638	0.551	0.487	**0.235**	0.273	0.252

100 test proteins and reported the metrics as detailed in § 4.3. In our comparative analysis, we endeavored to replicate the results of all baseline models to ensure consistency and fairness. However, replicating the results of DecompDiff as reported by (Guan et al., 2023b) posed challenges. In our study, we report both sets of outcomes, denoting our findings as DecompDiff*, and the original results reported in DecompDiff as simply "DecompDiff". We draw our conclusions based on the best results observed across DecompDiff and DecompDiff*.

4.5. Results

4.5.1. Performance of TAGMol

TAGMol outperforms all baselines, including the reference molecules, in binding-related metrics and *hit rate* (see Table 1). This achievement is especially noteworthy when compared to the state-of-the-art DecompDiff model, which relies on external computation for informed priors in the denoising process. The success of our model, attained without requiring extra data, underscores the importance of effectively learning useful signals from the existing training set and skillfully guiding the diffusion denoising phase.

A key metric where TAGMol excels is the Vina Score, achieving a 22% improvement over the state of the art (AR). This highlights its proficiency not only in generating molecules that bind effectively with proteins but also in producing high-affinity poses. This is further substantiated by the fact that on average an impressive 69.8% of the molecules generated by our model exhibit superior binding affinity compared to reference molecules, surpassing all other baselines.

In terms of molecular properties, TAGMol shows remarkable performance, surpassing diffusion-based models in QED by 14.6% and 22.2% compared to TargetDiff and DecompDiff, respectively, while maintaining similar diversity levels. Although there's a slight decrease in SA, the metric remains within a reasonable range, indicating satisfactory synthesizability. Furthermore, as shown in Table 2, our method closely maintains the bond distribution in alignment with the backbone diffusion model, namely TargetDiff, surpassing the non-diffusion baselines. This outcome emphasizes the efficient synergy between our *guide* model and the *diffusion* model, markedly improving our ability to generate molecules with targeted properties while preserving molecular conformation.

TAGMol, which focuses on optimizing multiple properties, achieves a state-of-the-art *hit rate*, surpassing the backbone model—TargetDiff—by a considerable margin. This is evidenced by the higher fraction of molecules generated by our method that meet the *hit criteria*. The collective results underscore the formidable capabilities of TAGMol in the SBDD domain. These advancements in critical areas of molecular generation and binding affinity demonstrate the model's potential to meet the intricate and diverse challenges of drug design. Figure 3 presents examples of ligand

Table 3. Ablation analysis assessing the properties of generated molecules under different property guidance scenarios. The first and second-place results are emphasized with bold and underlined text, respectively.

Methods	Vina Score (↓)		Vina Min (↓)		Vina Dock (↓)		QED (↑)		SA (↑)		Hit (↑)
	Avg.	Med.	Avg.	Med.	Avg.	Med.	Avg.	Med.	Avg.	Med.	Rate
backbone	-5.47	-6.30	-6.64	-6.83	-7.80	-7.91	0.48	0.48	<u>0.58</u>	<u>0.58</u>	20.5
backbone + BA Opt	**-7.35**	**-8.18**	**-8.38**	**-8.46**	**-9.04**	**-9.04**	0.49	0.50	0.53	0.53	22.6
backbone + QED Opt	-5.48	-6.46	-6.77	-6.93	-7.93	-8.06	**0.56**	**0.57**	<u>0.58</u>	<u>0.58</u>	<u>24.5</u>
backbone + SA Opt	-5.22	-6.03	-6.40	-6.57	-7.53	-7.73	0.47	0.48	**0.61**	**0.60**	19.4
TAGMOL	<u>-7.02</u>	<u>-7.77</u>	<u>-7.95</u>	<u>-8.07</u>	<u>-8.59</u>	<u>-8.69</u>	<u>0.55</u>	<u>0.56</u>	0.56	0.56	**27.7**

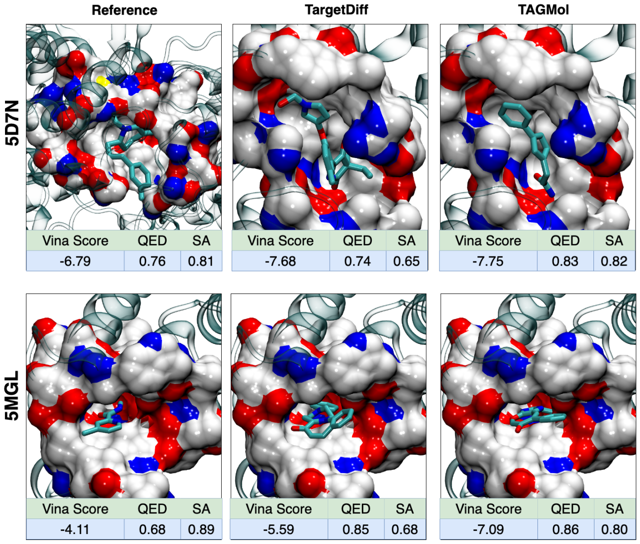

Figure 3. Visualization of reference molecules (left), alongside molecules generated by our backbone, TargetDiff (middle), and TAGMOL (right), for two targets: 5D7N and 5MGL.

molecules generated by our model, featuring valid structures and reasonable binding poses to the target.

4.5.2. SINGLE & MULTI OBJECTIVE GUIDANCE

In Table 3, we conduct an ablation study on our property guidance mechanism for multiple properties separately and all of them in tandem, demonstrating its effectiveness across various aspects. Specifically, we observe superior performance in each property when we provide respective guidance, showcasing the robustness of our guidance mechanism. We observe that when we provide guidance for one property, it does not result in a substantial deterioration in other properties. This highlights the collaborative mechanism between the *denoising* model and the *guide* model, emphasizing their combined power. Notably, a substantial enhancement is observed when we guide for binding affinity. This improvement is likely due to our approach of calculating binding affinity through docking software (Trott & Olson, 2010), which primarily depends on atomic distances. Our guide model's architecture, with its inherent inductive bias in modeling atomic distances, effectively facilitates the guidance of coordinate generation, contributing to this advancement. Consequently, improvements in QED and SA scores are comparatively less pronounced, as these properties exhibit a lower dependence on geometric factors. Lastly, when we provide guidance for all properties simultaneously, we witness substantial improvements across the majority of the properties, and achieves the highest *hit rate*. However, a slight decrease in SA scores is observed, which, nonetheless, lies well within our *hit criteria*. We also present additional visualization plots in Appendix E.4.

5. Conclusions

In this work, we redefined the problem of drug generative modeling by extending our focus beyond merely optimizing binding activity. We emphasize the importance of optimizing additional properties and integrating these considerations into our problem formulation, even when training data is sparse. Secondly, we developed a optimized generative method, TAGMOL, which effectively combines two key components: an *SE(3) equivariant graph diffusion model* to accurately replicate the geometries found in the training set and a multi-objective guidance driven by *SE(3) invariant GNN models*. This innovative combination enables our model to efficiently navigate the diffusion sampling space, focusing on properties of interest through the use of gradients. Lastly, our rigorous empirical evaluation demonstrates that TAGMOL notably enhances metric performance, balancing the optimization of binding affinity with critical pharmacological properties such as QED and SA.

Limitations and Future Work. Although the present work shows promising results, we share some complementary directions worth investigating in the future. First, improved representation of the system using additional inductive bias in terms of symmetry and coarse-graining could furnish more accurate predictions of different properties that are structure dependent. Second, model compression and parameter pruning could speed-up inference without compromising performance. Finally, it would be worth investigating physics-based techniques, such as molecular simulations, to allow generalization to larger family of systems.

6. Broader Impact

Drug discovery is an area that can have broad impact on humanity. Accelerating the discovery of novel drugs can ensure more inclusive healthcare worldwide and can have impact especially on reducing the disparity within and across countries. In the development of our methodology, we have taken care to ensure its ethical application, fully acknowledging that there exists inadvertent potential for misuse. Specifically, there is a risk that our optimization algorithms could be repurposed for creating drugs with detrimental side effects. This underscores the importance of exercising caution and responsibility in the deployment of such algorithms in practical scenarios to prevent any unintended negative consequences from their use.

References

Alonso, H., Bliznyuk, A. A., and Gready, J. E. Combining docking and molecular dynamic simulations in drug design. *Medicinal Research Reviews*, 26, 2006. URL https://api.semanticscholar.org/CorpusID:16337213.

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

Bajusz, D., Rácz, A., and Héberger, K. Why is tanimoto index an appropriate choice for fingerprint-based similarity calculations? *Journal of cheminformatics*, 7(1):1–13, 2015.

Bento, A. P., Hersey, A., Félix, E., Landrum, G., Gaulton, A., Atkinson, F., Bellis, L. J., De Veij, M., and Leach, A. R. An open source chemical structure curation pipeline using rdkit. *Journal of Cheminformatics*, 12:1–16, 2020.

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.

Bjerrum, E. J. and Threlfall, R. Molecular generation with recurrent neural networks (rnns). *ArXiv*, abs/1705.04612, 2017. URL https://api.semanticscholar.org/CorpusID:8280382.

Chenthamarakshan, V., Das, P., Hoffman, S., Strobelt, H., Padhi, I., Lim, K. W., Hoover, B., Manica, M., Born, J., Laino, T., et al. Cogmol: Target-specific and selective drug design for covid-19 using deep generative models. *Advances in Neural Information Processing Systems*, 33: 4320–4332, 2020.

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.

Eckmann, P., Sun, K., Zhao, B., Feng, M., Gilson, M. K., and Yu, R. Limo: Latent inceptionism for targeted molecule generation. *arXiv preprint arXiv:2206.09010*, 2022.

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

Ghorbani, M., Gendelev, L., Beroza, P., and Keiser, M. Autoregressive fragment-based diffusion for pocket-aware ligand design. In *NeurIPS 2023 Generative AI and Biology (GenBio) Workshop*, 2023.

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

Guan, J., Qian, W. W., Peng, X., Su, Y., Peng, J., and Ma, J. 3d equivariant diffusion for target-aware molecule generation and affinity prediction. In *The Eleventh International Conference on Learning Representations*, 2023a. URL https://openreview.net/forum?id=kJqXEPXMsE0.

Guan, J., Zhou, X., Yang, Y., Bao, Y., Peng, J., Ma, J., Liu, Q., Wang, L., and Gu, Q. Decompdiff: Diffusion models with decomposed priors for structure-based drug design. 2023b.

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

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

Jeon, W. and Kim, D. Autonomous molecule generation using reinforcement learning and docking to develop potential novel inhibitors. *Scientific reports*, 10(1):22104, 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 Welling, M. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Köhler, J., Klein, L., and Noé, F. Equivariant flows: exact likelihood generative learning for symmetric densities. In *International Conference on Machine Learning*, pp. 5361–5370. PMLR, 2020.

Lee, S., Jo, J., and Hwang, S. J. Exploring chemical space with score-based out-of-distribution generation. *Proceedings of the 40th International Conference on Machine Learning*, 2023.

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

Liu, Q., Allamanis, M., Brockschmidt, M., and Gaunt, A. L. Constrained graph variational autoencoders for molecule design. In *Neural Information Processing Systems*, 2018. URL https://api.semanticscholar.org/CorpusID:43924638.

Liu, Z., Zhang, W., Xia, Y., Wu, L., Xie, S., Qin, T., Zhang, M., and Liu, T.-Y. MolXPT: Wrapping molecules with text for generative pre-training. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, Toronto, Canada, July 2023. Association for Computational Linguistics.

Long, S., Zhou, Y., Dai, X., and Zhou, H. Zero-shot 3d drug design by sketching and generating. *Advances in Neural Information Processing Systems*, 35:23894–23907, 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.

Olivecrona, M., Blaschke, T., Engkvist, O., and Chen, H. Molecular de-novo design through deep reinforcement learning. *Journal of cheminformatics*, 9(1):1–14, 2017.

Peng, X., Luo, S., Guan, J., Xie, Q., Peng, J., and Ma, J. Pocket2mol: Efficient molecular sampling based on 3d protein pockets. In *International Conference on Machine Learning*, pp. 17644–17655. PMLR, 2022.

Ragoza, M., Masuda, T., and Koes, D. R. Learning a continuous representation of 3d molecular structures with deep generative models. *ArXiv*, abs/2010.08687, 2020. URL https://api.semanticscholar.org/CorpusID:224713572.

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, 2022.

Satorras, V. G., Hoogeboom, E., Fuchs, F. B., Posner, I., and Welling, M. E (n) equivariant normalizing flows. *arXiv preprint arXiv:2105.09016*, 2021.

Skalic, M., Jiménez, J., Sabbadin, D., and De Fabritiis, G. Shape-based generative modeling for de novo drug design. *Journal of chemical information and modeling*, 59(3):1205–1214, 2019.

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.

Stanton, S., Maddox, W., Gruver, N., Maffettone, P., Delaney, E., Greenside, P., and Wilson, A. G. Accelerating bayesian optimization for biological sequence design with denoising autoencoders. In *International Conference on Machine Learning*, pp. 20459–20478. PMLR, 2022.

Tanimoto, T. T. Elementary mathematical theory of classification and prediction. 1958.

Trott, O. and Olson, A. J. Autodock vina: improving the speed and accuracy of docking with a new scoring function, efficient optimization, and multithreading. *Journal of computational chemistry*, 31(2):455–461, 2010.

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. W., Eiden, P., Gao, H., Guzman-Perez, A., Hopper, T., Kelley, B. P., Mathea, M., Palmer, A., Settels, V., Jaakkola, T., Jensen, K. F., and Barzilay, R. Analyzing learned molecular representations for property prediction. *Journal of Chemical Information and Modeling*, 59:3370 – 3388, 2019. URL https://api.semanticscholar.org/CorpusID:198986021.

Yang, Y., Ouyang, S., Dang, M., Zheng, M., Li, L., and Zhou, H. Knowledge guided geometric editing for unsupervised drug design. 2021.

You, J., Liu, B., Ying, Z., Pande, V., and Leskovec, J. Graph convolutional policy network for goal-directed molecular graph generation. *Advances in neural information processing systems*, 31, 2018.

Zhang, Z., Min, Y., Zheng, S., and Liu, Q. Molecule generation for target protein binding with structural motifs.

In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=Rq13idF0F73.

Zhavoronkov, A., Ivanenkov, Y. A., Aliper, A., Veselov, M. S., Aladinskiy, V. A., Aladinskaya, A. V., Terentiev, V. A., Polykovskiy, D. A., Kuznetsov, M. D., Asadulaev, A., et al. Deep learning enables rapid identification of potent ddr1 kinase inhibitors. *Nature biotechnology*, 2019.

Zhou, G., Gao, Z., Ding, Q., Zheng, H., Xu, H., Wei, Z., Zhang, L., and Ke, G. Uni-mol: A universal 3d molecular representation learning framework. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=6K2RM6wVqKu.

Zhou, Z., Kearnes, S. M., Li, L., Zare, R. N., and Riley, P. F. Optimization of molecules via deep reinforcement learning. *Scientific Reports*, 9, 2018. URL https://api.semanticscholar.org/CorpusID:53040150.

Zhou, Z., Kearnes, S., Li, L., Zare, R. N., and Riley, P. Optimization of molecules via deep reinforcement learning. *Scientific reports*, 9(1):10752, 2019.

A. Notation Summary

Table 4. Notations

Notation	Explanation	Domain
$N_{\mathcal{P}}$	Number of protein atoms	$\in \mathbb{R}^1$
N_f	Feature dimension of protein atom	$\in \mathbb{R}^1$
$\mathbf{x}_i^{\mathcal{P}}$	i-th protein atom coordinate	$\in \mathbb{R}^3$
$\mathbf{v}_i^{\mathcal{P}}$	i-th protein atom feature	$\in \mathbb{R}^{N_f}$
$N_{\mathcal{M}}$	Number of ligand atoms	$\in \mathbb{R}^1$
K	Number of ligand atom types	$\in \mathbb{R}^1$
$\mathbf{x}_i^{\mathcal{M}}$	i-th ligand atom coordinate	$\in \mathbb{R}^3$
$\mathbf{x}_{i,t}^{\mathcal{M}}$	i-th ligand atom coordinate at diffusion time step t	$\in \mathbb{R}^3$
$\mathbf{v}_i^{\mathcal{M}}$	i-th ligand atom *one-hot* representation	$\in \mathbb{R}^K$
$\mathbf{v}_{i,t}^{\mathcal{M}}$	i-th ligand atom *one-hot* representation at diffusion time step t	$\in \mathbb{R}^K$
$\mathbf{X}^{\mathcal{M}}$	Matrix representation of all ligand atom coordinates	$\in \mathbb{R}^{N_{\mathcal{M}} \times 3}$
$\mathbf{V}^{\mathcal{M}}$	Matrix representation of all ligand atom types	$\in \mathbb{R}^{N_{\mathcal{M}} \times K}$
$\mathbf{X}^{\mathcal{P}}$	Matrix representation of all protein atom coordinates	$\in \mathbb{R}^{N_{\mathcal{P}} \times 3}$
$\mathbf{V}^{\mathcal{P}}$	Matrix representation of all protein atom features	$\in \mathbb{R}^{N_{\mathcal{P}} \times N_f}$
β_t	Variance Schedule for the diffusion model	$\in \mathbb{R}^1$
q	Diffusion noising transition function	-
θ	Parameters of generative model	-
p_θ	Denoising diffusion transition function	-
$\mathbb{Y}$	Set of properties to be optimized during generation	-
$\mathbf{y}$	A property $\mathbf{y} \in \mathbb{Y}$	-
$\phi_{\mathbf{y}}$	Parameters of guide corresponding to $\mathbf{y}$	-
$\phi_{\mathbb{Y}}$	Set of guide parameters corresponding to $\forall \mathbf{y} \in \mathbb{Y}$	-
$s_{\mathbf{y}}$	Guide strength for property $\mathbf{y}$	$\in \mathbb{R}^1$
$s_{\mathbf{y}}^{opt}$	Optimal Guide strength for property $\mathbf{y}$	$\in \mathbb{R}^1$

B. Related Work

Neural Drug Design Neural models have significantly changed the landscape for drug design that was previously dominated by computational techniques such as MD simulations and Docking (Alonso et al., 2006). It has gone through multiple paradigm shifts in the recent past - ranging from using only 1D representations (SMILES) (Bjerrum & Threlfall, 2017; Gómez-Bombarelli et al., 2018) to 2D representations (molecular graphs) (Liu et al., 2018; Zhou et al., 2018) to 3D representations (Skalic et al., 2019; Ragoza et al., 2020) (molecular coordinates). Recent works have also showcased the importance of using target-aware models for practical drug design. Advancements in target-aware drug design, particularly in the realm of text and graph-based generative methods (Eckmann et al., 2022; Stanton et al., 2022; Chenthamarakshan et al., 2020; Lee et al., 2023), have made significant strides. However, these methods often generate ligands without considering the 3D structure of the target protein. TargetDiff (Guan et al., 2023a) and DecompDiff (Guan et al., 2023b) are two diffusion-based models that consider the 3D structure of the target protein. DecompDiff extends TargetDiff by decomposing the task into generating the arms and scaffold while explicitly considering bonds between the atoms during the diffusion process which is ignored by TargetDiff. Fragment-based generation methods such as (Ghorbani et al., 2023) allow for controlled generation, ensuring that only certain types of molecular fragments are present in the generated ligand. But they still fall behind the performance of diffusion-based approaches in terms of their binding affinity.

Molecular Property Prediction Predicting the properties of molecules through physio-chemical experiments is an expensive and time-consuming solution, that is unsuitable for getting intermediate feedback on AI-designed molecules. Using neural models has allowed high-quality prediction of various molecular properties in an automated fashion. Neural methods use different molecular representations to predict molecules, including SMILES representation with pre-trained transformers (Liu et al., 2023) or molecular structures (Yang et al., 2019; Zhou et al., 2023) for predicting the properties of interest.

Molecular Property Optimization Prior methods for property optimization have focused on using Reinforcement learning to generate molecules with desired properties (You et al., 2018; Zhou et al., 2019; Zhavoronkov et al., 2019; Jeon & Kim, 2020; Olivecrona et al., 2017). However, RL methods are often computationally expensive and challenging to optimize due to the vast search space. LIMO (Eckmann et al., 2022) uses Variational Auto Encoders (VAEs) and gradient gudiance on MLP-based property predictors but does not consider the 3D structure of the target protein or the generated ligand. (Lee et al., 2023) introduced a novel graph-based diffusion model for out-of-distribution (OOD) generation, enhancing explorability through OOD-controlled reverse-time diffusion and property-guided sampling, albeit focusing solely on 2D molecular representations, unlike the target-aware 3D structure generation with multi-objective guidance featured in our work.

C. Property Distribution in Training Data

Figure 4 illustrates the distribution of QED, SA, and Vina Scores within the training split. From the figure, it is evident that a substantial proportion of molecules exhibit low binding and fall below the hit criteria for QED.

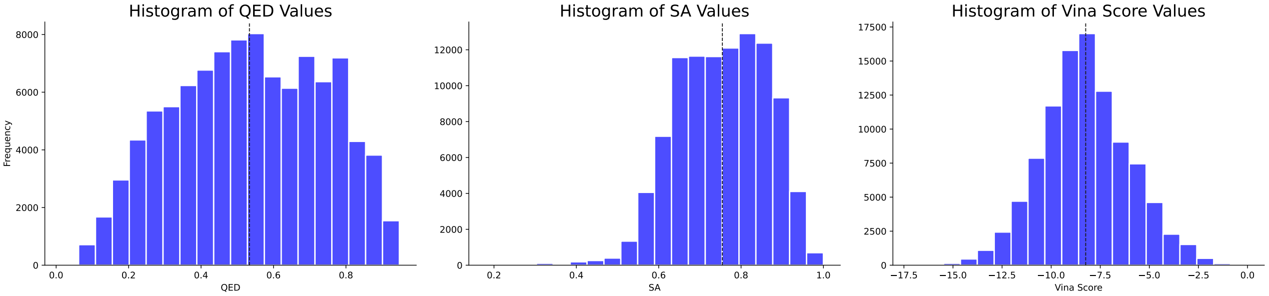

Figure 4. Distribution of QED, SA, and Vina Score properties in the training split, with the mean values represented by dotted lines

D. Training and Evaluation of Guide

D.1. Training

The models are trained with a batch size of 4 to minimize the Root Mean Square Error (RMSE) loss, utilizing a gradient descent algorithm. The initial learning rates were set at 0.001 for the *binding affinity* and *QED guides*, and at 5e-4 for the *SA guide*. Additionally, to maintain the stability of the training process, the gradient norms were clipped at a maximum value of 8. The learning rate underwent exponential decay by a factor of 0.6, with a lower bound set at 1e-5. This decay was triggered in the absence of any improvement in the validation loss across 10 consecutive evaluations. Evaluations were conducted every 2000 training steps, and convergence of all models was achieved within 200,000 steps. All experiments were conducted using the NVIDIA A100 GPU.

D.2. Evaluation

The performance of each guide model was evaluated for their respective property predictions using the same test dataset. As outlined in § 4.1, we employed the same split to evaluate our *binding affinity guide* model. While the train-test split ensures the uniqueness of the protein-ligand complexes between the sets, 34 ligands are common to both, leading to potential train-test leakage. Consequently, for *QED* and *SA* guides that depend solely on ligands, we have excluded these 34 overlapping ligands from the test split (No overlaps). Table 5 presents the performance evaluation of each *guide* model, indicating high R^2 values on the original test dataset. After removing the overlapping ligands from the test set, the performance metrics remained consistent, confirming the model's effective generalization to unseen ligands. While we report the results on fair test split, it's crucial to note that, diverging from the traditional machine learning training and testing process, we train the guide to direct the generated latent space of the diffusion model.

Table 5. Evaluation of the guide model on the original test set with unique protein-ligand complexes, and on a modified test set (No Overlaps) where 32 overlapping ligands have been removed.

Target Property	Test Set (Original)		Test Set (No Overlaps)	
	RMSE	R^2	RMSE	R^2
BA	0.642	0.953	0.652	0.951
QED	0.067	0.892	0.079	0.850
SA	0.054	0.848	0.065	0.821

E. Additonal Experiments

E.1. Search for optimal Guide Strength

To determine the optimal Guide Strength (s), a grid search was conducted over the set of values: $\{0, 1, 2, 5, 10, 20, 50\}$. Here, $s = 0$ represents the scenario without guidance, included for comparative purposes to gauge guide-induced enhancements. The generation process was limited to 50 molecules for a limited set of 4 targets, to accommodate computational constraints. Initially, for each property, we find the optimal guide strength $s_{\mathbf{y}}^{opt}$ over the grid. Once the optimal values for single objective are tuned, for multi-objective guidance, we obtained optimal s values by recalibrating the $s_{\mathbf{y}}^{opt}$ with $w_{\mathbf{y}}$, as detailed in §-4.4.

Table 6 presents the results of our tuning across different guide strengths (s) mentioned in the grid. It is evident that non-zero s values yield improved property metrics, underscoring the efficacy of guidance. For Binding Affinity (BA) guidance, optimal s values emerge as 1 and 2, based on average and median Vina Scores, respectively. Considering both Vina Min and Vina Dock values, $s = 2$ is selected for superior outcomes. In the case of QED guidance, an s value of 20 distinctly outperforms others in terms of both average and median QED scores. With SA guidance, although $s = 20$ and $s = 50$ appear to be optimal, the validity(Guan et al., 2023b) of generated molecules takes a low value of 67% and 46%, respectively. Consequently, $s = 5$ is chosen, providing the second-best SA values, with 84% of molecules being valid and only a slight decrease in average SA value by 0.02. In summary, the selected Guide Strengths are: $s_{ba}^{opt} = 2$, $s_{qed}^{opt} = 20$, and $s_{sa}^{opt} = 5$.

Table 6. Search for optimal Guide Strength (s), conducted individually for each Guide Model (single-objective setting). To minimize computational demands, the generation is limited to 50 samples across 4 targets. Only the metrics of the property being guided are included. Symbols (↑) / (↓) denote preference for higher or lower values, respectively. Top results are highlighted in bold for first place and underlined for second. Italicized figures indicate instances where over 25% of molecules generated under the specified guide strength were not valid.

	BA Guide						QED Guide		SA Guide	
s	Vina Score (↓)		Vina Min (↓)		Vina Dock (↓)		QED (↑)		SA (↑)	
	Avg.	Med.	Avg.	Med.	Avg.	Med.	Avg.	Med.	Avg.	Med.
0	-5.37	-6.17	-6.31	-6.83	-7.65	-7.67	0.51	0.54	0.59	0.59
1	**-7.82**	-7.93	-8.16	-8.24	-8.62	-8.64	0.52	0.53	0.59	0.59
2	-7.69	**-8.08**	**-8.33**	**-8.40**	**-8.86**	**-8.89**	0.52	0.53	0.60	0.59
5	-7.55	-7.74	-8.02	-8.04	-8.67	-8.65	0.55	0.58	0.62	0.62
10	-7.11	-7.09	-7.53	-7.33	-8.42	-8.34	0.54	0.54	0.60	0.59
20	-6.85	-6.69	-7.03	-6.87	-8.16	-8.14	**0.58**	**0.60**	***0.64***	***0.63***
50	-6.11	-5.96	-5.68	-5.91	-7.73	-7.44	0.55	0.55	***0.64***	***0.63***

E.2. Ablation on Single and Multi-Objective Guidance

To assess the impact of guidance on various combinations of properties, we conducted a detailed ablation study. Initially, we present results for single-objective guidance on different properties, then proceed to guidance for two properties at a time, and ultimately, guidance incorporating all properties. Our results, highlighted in Table 7, indicate that single-objective guidance maximizes improvement for the targeted property. When applying guidance to two properties, the enhancements for each are moderate, illustrating a trade-off compared to separate guidance. This trade-off persists with guidance across all properties. However, given the multifaceted nature of drug discovery, where the goal is to produce molecules with high binding affinity and desired properties like QED and SA, our approach significantly enhances *hit rate* and most properties over no guidance. This underscores the effectiveness of guidance in navigating regions that are crucial in discovery.

E.3. Time Complexity

When executed on an NVIDIA A100 GPU, we observed a notable processing time trend for inference. Guidance for properties like QED and SA led to significant improvements (refer to Table 7) with minimal time increases of 1.06x and 1.07x, respectively, compared to the backbone. However, guiding for Binding Affinity (BA) led to a more substantial time increase due to the inclusion of protein atoms, demanding extra computational effort compared to QED or SA guidance, which involves only ligand atoms. Ultimately, TAGMOL recorded its longest processing time of 1.87x when applying

Table 7. Extensive ablation analysis assessing the properties of generated molecules under different property guidance scenarios. The first and second-place results are emphasized with bold and underlined text, respectively.

w_{ba}	w_{qed}	w_{sa}	Vina Score (↓)		Vina Min (↓)		Vina Dock (↓)		QED (↑)		SA (↑)		Hit (↑)
			Avg.	Med.	Avg.	Med.	Avg.	Med.	Avg.	Med.	Avg.	Med.	Rate
0	0	0	-5.47	-6.30	-6.64	-6.83	-7.80	-7.91	0.48	0.48	0.58	0.58	20.5
1	0	0	**-7.35**	**-8.18**	**-8.38**	**-8.46**	**-9.04**	**-9.04**	0.49	0.5	0.53	0.53	22.6
0	1	0	-5.48	-6.46	-6.77	-6.93	-7.93	-8.06	**0.56**	**0.57**	0.58	0.58	24.5
0	0	1	-5.22	-6.03	-6.4	-6.57	-7.53	-7.73	0.47	0.48	**0.61**	**0.6**	19.4
0.5	0.5	0	-7.11	-7.96	-8.13	-8.21	-8.82	-8.85	**0.56**	**0.57**	0.55	0.55	26.1
0.5	0	0.5	-7.2	-7.95	-8.16	-8.26	-8.83	-8.84	0.5	0.51	0.55	0.55	24.7
0	0.5	0.5	-5.43	-6.34	-6.63	-6.86	-7.85	-7.97	0.55	0.56	0.59	0.59	24.1
0.33	0.33	0.34	-7.02	-7.77	-7.95	-8.07	-8.59	-8.69	0.55	0.56	0.56	0.56	**27.7**

guidance for all properties. It is important to note that TAGMOL is still comparable to 1.72x of DecompDiff (Guan et al., 2023a) sampling time.

Table 8. Inference time for different models

Method	Time (s)
backbone	938
backbone + BA opt	1646
backbone + QED opt	995
backbone + SA opt	1007
TAGMOL	1755

E.4. Multi-Objective Rationale

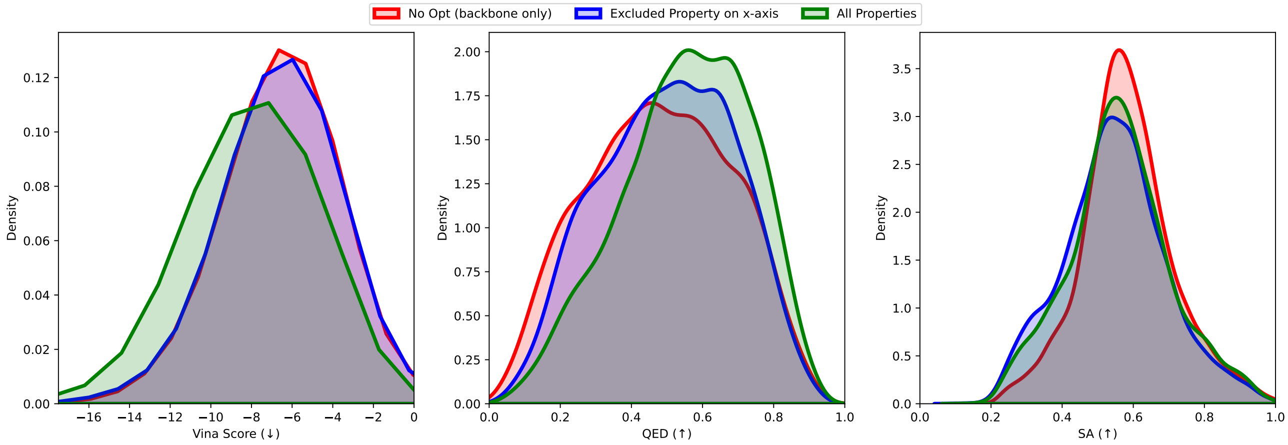

Figure 5. Distribution of molecular properties in molecules generated by the backbone model without any guidance (No Opt), when guided by two properties while excluding the one indicated on the x-axis (Excluded Property on x-axis), and when guided by all three properties (All Properties). (↓) denote properties where lower values are preferred, while (↑) indicate properties where higher values are desirable.

Figure 5 presents the distribution of molecular properties for 10,000 generated molecules (100 per target, across 100 targets). These molecules are generated using three methods: (i) the backbone model without any guidance; (ii) guidance using only two properties, leaving out the third one indicated on the x-axis; and (iii) guidance using all three properties. Optimizing all the properties helps improve the Vina Score and QED compared to the 'No Opt' baseline. Although it doesn't help improve the SA, it is still helpful, as excluding the SA property in optimization worsens the SA scores. Thus, it is useful to provide guidance for all three properties.

F. Pseudo Code for TAGMOL

This section provides a summary of the overall sampling procedures employing multiple guidances.

Algorithm 1 Psuedo Code of TAGMOL

```
Require: The protein binding site $\mathbf{P}$, generative model $\theta$, Property Predictors $\phi_{\mathbb{Y}}$.
Ensure: Generate ligand molecule $\mathbf{M}$ that binds to the protein pocket & optimized for properties $\mathbb{Y}$
  Sample $N_{\mathcal{M}}$ atoms based on a prior distribution relative to the pocket size.
  Move CoM of protein atoms to zero.
  Sample coordinates $\mathbf{X}_T^{\mathcal{M}}$ and atom types $\mathbf{V}_T^{\mathcal{M}}$ based on prior:
  $\mathbf{X}_T^{\mathcal{M}} \sim \mathcal{N}(0, \mathbf{I})$
  $\mathbf{V}_T^{\mathcal{M}} = \texttt{one\_hot}(\arg\max_i g_i)$, where $g \sim$ Gumbel$(0, 1)$
  for $t = T, T-1, \ldots, 1$ do
    Calculate $[\widehat{\mathbf{X}}_{0|t}^{\mathcal{M}}, \widehat{\mathbf{V}}_{0|t}^{\mathcal{M}}]$ using $\theta([\mathbf{X}_t^{\mathcal{M}}, \mathbf{V}_t^{\mathcal{M}}], t, \mathcal{P})$
    Calculate $\tilde{\mu}_t\left(\mathbf{X}_t^{\mathcal{M}}, \hat{\mathbf{X}}_{0|t}^{\mathcal{M}}\right)$ according to posterior in Equation 3
    Calculate $\delta$ according to Equation 16
    Sample $\mathbf{M}_{t-1}$ according to the Equation 15
  end for
```