

DiffDec: Structure-Aware Scaffold Decoration with an End-to-End Diffusion Model

Junjie Xie,^{||} Sheng Chen,^{||} Jinping Lei,^{*} and Yuedong Yang^{*}



Cite This: *J. Chem. Inf. Model.* 2024, 64, 2554–2564



Read Online

ACCESS |



Metrics & More

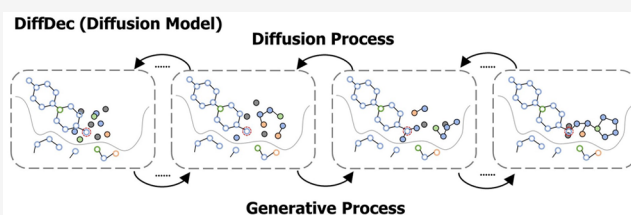


Article Recommendations



Supporting Information

ABSTRACT: In molecular optimization, one popular way is R-group decoration on molecular scaffolds, and many efforts have been made to generate R-groups based on deep generative models. However, these methods mostly use information on known binding ligands, without fully utilizing target structure information. In this study, we proposed a new method, DiffDec, to involve 3D pocket constraints by a modified diffusion technique for optimizing molecules through molecular scaffold decoration. For end-to-end generation of R-groups with different sizes, we designed a novel fake atom mechanism. DiffDec was shown to be able to generate structure-aware R-groups with realistic geometric substructures by the analysis of bond angles and dihedral angles and simultaneously generate multiple R-groups for one scaffold on different growth anchors. The growth anchors could be provided by users or automatically determined by our model. DiffDec achieved R-group recovery rates of 69.67% and 45.34% in the single and multiple R-group decoration tasks, respectively, and these values were significantly higher than competing methods (37.33% and 26.85%). According to the molecular docking study, our decorated molecules obtained a better average binding affinity than baseline methods. The docking pose analysis revealed that DiffDec could decorate scaffolds with R-groups that exhibited improved binding affinities and more favorable interactions with the pocket. These results demonstrated the potential and applicability of DiffDec in real-world scaffold decoration for molecular optimization.



INTRODUCTION

Computational methods have sped up the process of drug development in a cost-effective manner and minimized the risk of failure in the later stages.¹ Existing methods can utilize known compounds to design novel compounds, also known as ligand-based drug design (LBDD).² Traditional LBDD methods like the pharmacophore model³ or quantitative structure activity relationship model⁴ learn the structure–activity relationship of ligands from bioactive data to design new drug candidates.⁵ To efficiently explore the vast chemical space, deep generative models have gradually become the promising new engine in the majority of drug design tools.^{6,7} Based on different molecular representations, the existing molecular generative models can be roughly divided into two categories: SMILES-based and graph-based. SMILES-based methods^{8–12} generally generate desired compounds through language models like long short-term memory¹³ and transformer.¹⁴ Differently, graph-based methods represent a molecule as a graph where its node-edge structure naturally matches the atom-bond information on molecules.^{15–17} Such methods have rapidly developed especially with the recently developed E(3)-equivariant graph neural network architectures¹⁸ that could embed the 3D molecular information.

While most of these deep generative models focus on *de novo* molecular generation, an important stage of drug design is lead optimization, which aims to optimize the favorable properties of compounds by changing part of their atoms.¹⁹ In particular, one

important type of molecular optimization is scaffold decoration^{20,21} that retains the core scaffold and modifies molecules in specific motifs such as R-groups. Recently, LibINVENT²² leveraged chemical reaction templates to slice the molecules into scaffolds and decorations and then trained a deep generative model to generate R-groups that decorated the extracted molecular scaffolds. This reaction-based R-group decoration method guarantees that the generated molecules are highly synthetic and accessible. However, this ligand-based generative model was based on recurrent neural networks and SMILES molecular representation,²³ which were not capable of taking the 3D structure information on protein pockets into consideration.

In parallel to LBDD, structure-based drug design (SBDD)²⁴ directly learns protein structure information, aiming to design drugs that selectively and specifically target disease-causing receptors.²⁵ Nowadays, there is a rapid increase in the accessible structural data because of the breakthrough in structural biology (e.g., X-ray crystallography²⁶ and Cryo-EM²⁷) and protein structure prediction (e.g., AlphaFold²⁸ and ESMFold²⁹), which

Special Issue: Machine Learning in Bio-cheminformatics

Received: September 12, 2023

Revised: December 18, 2023

Accepted: December 27, 2023

Published: January 24, 2024



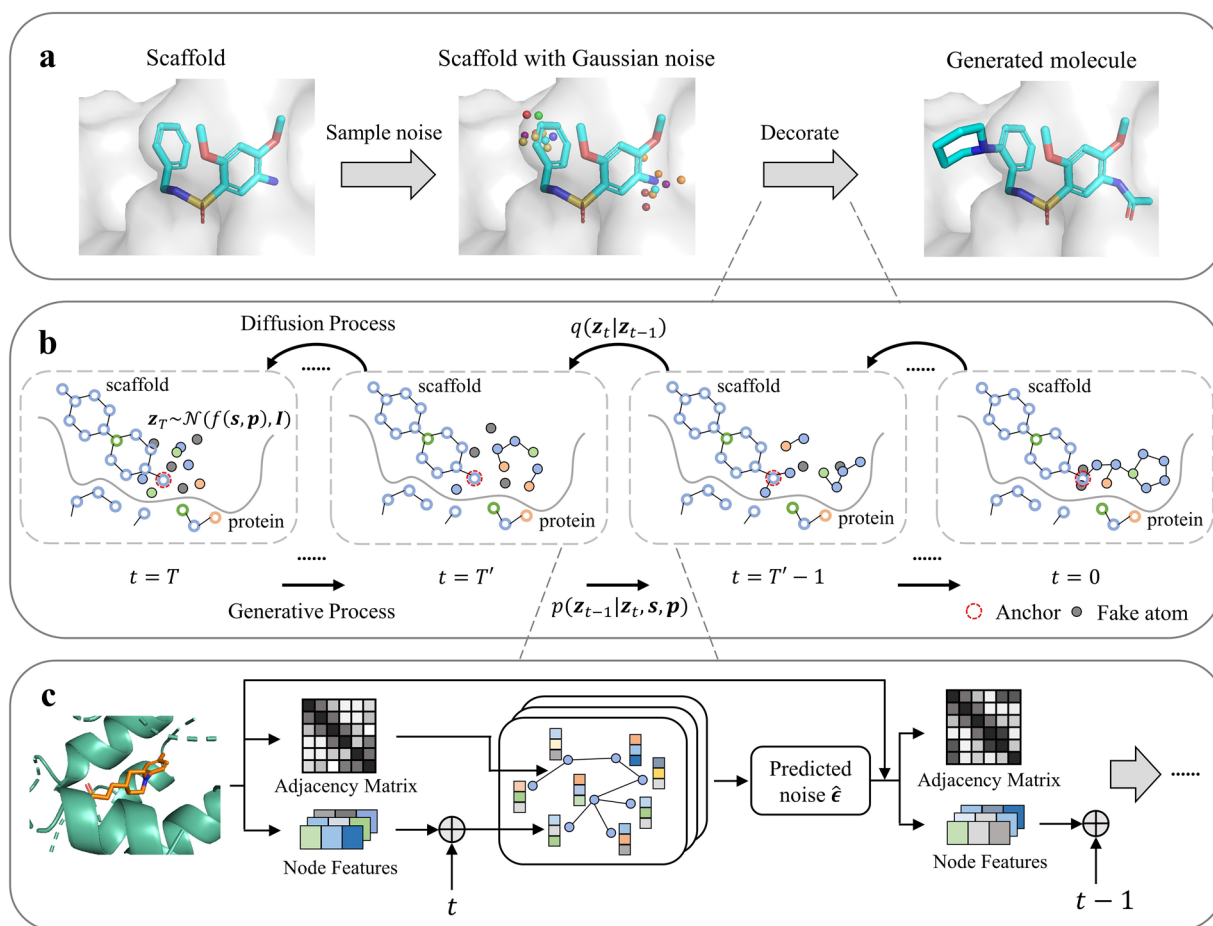


Figure 1. Overview of DiffDec. (a) Flowchart of structure-aware scaffold decoration using DiffDec. (b) Implementation details of DiffDec. The diffusion process iteratively adds Gaussian noise to the data, while the generative process gradually denoises the noise distribution under the condition of scaffold and protein pocket to recover the ground truth R-groups. (c) A detailed denoising procedure from time step t to $t - 1$ using EGNN.

empowers SBDD to be applied on most diseases with known protein targets. Traditional SBDD methods, such as molecular dynamics simulation³⁰ and molecular docking,³¹ are commonly employed for virtual screening of known compounds rather than designing novel molecules. Recently, increasing efforts have attempted to incorporate 3D protein structures in molecule design tasks. For example, Pocket2Mol³² utilized equivariant graph neural networks to learn the chemical and geometric constraints imposed by protein pockets and sampled molecules in an autoregressive manner. FLAG³³ generated 3D molecules using realistic substructures fragment-by-fragment around the binding pocket. Unfortunately, these autoregressive generation methods are prone to suffering from inaccurate accumulation for unrealistic 3D molecular structures.³⁴ As a full-atom generation manner rather than an autoregressive manner, the diffusion model³⁵ is recently applied to the field of small molecule generation. DiffLinker³⁶ employed a conditional diffusion model and a linker size predictor to design molecular linkers for a set of fragments and proteins. DiffSBDD³⁷ proposed an E(3)-equivariant 3D-conditional diffusion model to generate ligands conditioned on the binding pockets. SILVR³⁸ conditioned a diffusion-based equivariant generative model to generate novel compounds based on fragments compatible with a protein's binding site. However, such methods have not been used to decorate molecular scaffolds for optimization.

In this study, we have developed a new method, DiffDec, to optimize molecules through molecular scaffold decoration

conditioned on the 3D protein pocket by an E(3)-equivariant graph neural network and diffusion model. Specifically, we constructed two 3D scaffold decoration data sets through reaction-based slicing of 3D protein–ligand complex in CrossDocked data set.³⁹ According to the scaffold and the context pocket, we trained a 3D conditional diffusion model to generate structure-aware R-groups. For an end-to-end generation of different-sized R-groups, we newly introduced a fake atom mechanism. Through comprehensive evaluations, DiffDec exhibited superior performance compared to baseline methods in terms of validity, R-group recovery, similarity, binding affinity, and generation of realistic geometric substructures on the two benchmark data sets. Notably, DiffDec could identify the growth anchors and generate R-groups well for the scaffolds without the provided anchors. More importantly, case studies revealed that the compounds decorated using DiffDec exhibited realistic 3D structures and a higher number of protein–ligand interactions compared with the reference molecules.

MATERIALS AND METHODS

Data Sets. In the scaffold decoration data set constructed by Fialková et al.,²² the sliced scaffolds and R-groups are represented by SMILES, and the 3D structural information is unavailable. Therefore, we curated two benchmark data sets for single and multi R-group scaffold decoration based on three-dimensional protein–ligand complex structures by using the CrossDocked data set³⁹ and reaction-based slicing method.

Specifically, the CrossDocked data set included 22.5 million protein–ligand complexes at various levels of quality. We filtered out data points whose binding pose RMSD is greater than 1 Å and obtained a refined subset with 184,057 data points. The original protein structures in the refined subset were clipped to substructures that only included the atoms within the 10 Å region around the binding molecule. Then, the subset data was clustered using mmseqs2⁴⁰ at 30% protein sequence identity. Following the splitting strategies proposed by Luo et al.,⁴¹ we used 100,000 protein–ligand pairs for subsequent processing as the original training set and 100 complexes from the remaining clusters as the original test set. Following the same slicing method in LibINVENT,²² the ligands in the original training set and test sets were sliced into scaffolds and R-groups, respectively, utilizing 37 customized reaction-based rules and ultimately the final training set and test sets were obtained. This reaction-based division method improved the validity and synthetic accessibility of scaffold decoration with simple chemical reactions. For the single R-group scaffold decoration data set, we set the maximum cut in the configuration file of slicing molecules to 1. It contained 76,108 tuples with molecular scaffolds, pockets, and R-groups in the training set and 43 tuples in the test set. For a more challenging multi R-group data set, we changed the maximum cut to 4. It contained 154,792 tuples in the training set and 102 tuples in the test set. The detailed information on these two data sets is shown in Table S1 and Figure S1.

Overview of DiffDec. DiffDec is capable of performing scaffold decoration in the context of pockets in 3D space. We used a set of atoms to represent molecules and proteins. Following the framework of denoising diffusion probabilistic model (DDPM),⁴² atoms in the target pocket p and given scaffold s are represented as context $c = \{s, p\} = \{(r_i, h_i)\}_{i=1}^N$, where r_i is the 3D geometric coordinate of the i th heavy atom, h_i is the corresponding element types, and N is the number of atoms of the scaffold s and protein p . As shown in Figure 1, our objective is to generate the decorations, i.e. R-groups $x = \{(r_i, h_i)\}_{i=1}^M$ to form a complete molecule conditioned on the scaffold and protein pocket. The method consists of a diffusion process and a generative process, where the diffusion process gradually adds noise at each time step, while the generative process learns how to recover data distribution from the noise distribution using a neural network.

Diffusion Process. Following other recently developed diffusion-based models for molecule generation,^{37,43} we iteratively add noise to the data points of R-groups x using a Gaussian distribution $\mathcal{N}$ at each time step but leave the corresponding context c unchanged throughout the diffusion process. We depict the latent noised representation using z_t for $t = 0, \dots, T$. Given the previous state z_{t-1} , the conditional distribution of the data state z_t at time step t is formulated by the multivariate normal distribution

$$q(z_t|z_{t-1}) = \mathcal{N}(z_t; \bar{\alpha}_t z_{t-1}, \bar{\sigma}_t^2 \mathbf{I}) \quad (1)$$

where $\bar{\alpha}_t \in \mathbb{R}^+$ decides how much of the original signal to keep, and $\bar{\sigma}_t \in \mathbb{R}^+$ controls how much noise to add. Since the full diffusion process is Markovian, the entire noising process can be written as

$$q(z_0, z_1, \dots, z_T|x) = q(z_0|x) \prod_{t=1}^T q(z_t|z_{t-1}) \quad (2)$$

For the distribution, q is normal and independent at each time step, and we can obtain the derivation for the distribution of z_t given x as follows

$$q(z_t|x) = \mathcal{N}(z_t|\alpha_t x, \sigma_t^2 \mathbf{I}) \quad (3)$$

where $\alpha_t = \bar{\alpha}_t \alpha_{t-1}$ and $\sigma_t^2 = \bar{\sigma}_t^2 + \bar{\alpha}_t^2 \bar{\sigma}_{t-1}^2$. Following Kingma et al.,⁴⁴ we can obtain the signal-to-noise ratio $\text{SNR}(t) = \alpha_t^2 / \sigma_t^2$. In general, α_t usually follows either a learned or predefined schedule from $\alpha_0 \approx 1$ toward $\alpha_T \approx 0$.⁴⁴ And we can also follow a variance-preserving diffusion process, where $\alpha_t = \sqrt{1 - \bar{\sigma}_t^2}$.⁴⁵

This closed-form formula indicates that we can achieve a noisy data distribution at any time step without iterative calculations.

With Bayes theorem, the posterior of the denoising transitions from time step t to $t - 1$ can be computed in closed-form conditioned on x

$$q(z_{t-1}|x, z_t) = \mathcal{N}(z_{t-1}; \mu_t(x, z_t), \varsigma_t^2 \mathbf{I}) \quad (4)$$

where $\mu_t(x, z_t) = \frac{\bar{\alpha}_t \alpha_{t-1}^2}{\sigma_t^2} z_t + \frac{\alpha_{t-1} \bar{\sigma}_t^2}{\sigma_t^2} x$ and $\varsigma_t^2 = \frac{\bar{\sigma}_t \alpha_{t-1}}{\sigma_t}$.

For the continuous atom coordinates, a nonzero distribution invariant to the translation cannot be obtained.¹⁸ However, we can use the data distributions on the linear subspace where the center of mass (COM) of the system is zero, and this strategy has been used consistently in diffusion.^{46,47} In actuality, this is accomplished by translating the center of mass of the system $f(c)$ before the diffusion process or generative process. Then, the noise can be added from $\mathcal{N}(0, \mathbf{I})$ instead of $\mathcal{N}(f(c), \mathbf{I})$. The choice of f is highly related to the problem being solved and prior knowledge. In our problem, we define the atoms connected to the R-groups in the scaffold as anchors. Therefore, if the information on anchors is available, $f(c)$ can be defined as the coordinates of anchors, and the diffusion processes of multiple anchors are separate and independent of each other; if not, $f(c)$ can be the center of mass of the scaffold.

For the discrete atomic features, there are also quantities of categorical diffusion models, where we prefer the strategy which places the discrete features on a continuous space.^{36,47} We encode the variables with a one-hot representation and directly add Gaussian noise to it.

To incorporate the R-group size prediction into the diffusion model in an end-to-end fashion, we introduce the fake atom mechanism. In general, the size of the R-group (measured using the number of heavy atoms) does not exceed a maximum of 10. In cases where the R-group contains fewer than 10 heavy atoms, we artificially introduce fake atoms to reach a size of 10 before the diffusion process. The coordinates of these fake atoms are set to $f(c)$, and an additional dimension of atomic type is used to indicate them. The subsequent noising steps remain the same as described previously. By using fixed-size noisy R-groups, denoising can be performed, and the sizes of the R-groups can be determined during the generative process, eliminating the need for an additional prediction module.

Generative Process. The generative process is the reverse of the diffusion process. It will gradually denoise the Gaussian noise obtained from sampling under the fixed context of scaffold and pocket to recover the ground truth R-groups. The generative transition distribution is defined as

$$p(z_{t-1}|z_t, c) = q(z_{t-1}|\hat{x}, z_t) \quad (5)$$

where $\hat{x}$ is an approximation of the data point. We use a neural network φ to predict the Gaussian noise $\hat{\epsilon}_t = \varphi(z_t, c, t)$ added

from time step $t - 1$ to t to estimate $\hat{\mathbf{x}}$.⁴² Therefore, we can obtain

$$\hat{\mathbf{x}} = (1/\alpha_t)\mathbf{z}_t - (\sigma_t/\alpha_t)\hat{\epsilon}_t \quad (6)$$

The neural network φ mentioned above is implemented as an adjusted E(n)-equivariant graph neural network (EGNN).⁴⁸ It uses the noise-added R-groups $\mathbf{z}_p$, the context $\mathbf{c}$, and the time step t as input and outputs the Gaussian noise that added from time step $t - 1$ to t , including coordinates and features $\hat{\epsilon}_t = [\hat{\epsilon}_t^r, \hat{\epsilon}_t^h]$. Since the output noises must be on a subspace with a zero center of gravity, the coordinate noises $\hat{\epsilon}_t^r$ should be projected down by subtracting the initial coordinates:⁴⁷

$$\hat{\epsilon}_t^r, \hat{\epsilon}_t^h = \varphi(\mathbf{z}_t, \mathbf{c}, t) = \text{EGNN}(\mathbf{z}_t, \mathbf{c}, t) - [\mathbf{z}_t^r, \mathbf{0}] \quad (7)$$

EGNN is made up of L layers of Equivariant Graph Convolutional Layers (EGCL), $\mathbf{r}^{l+1}$, and $\mathbf{h}^{l+1} = \text{EGCL}[\mathbf{r}^l, \mathbf{h}^l]$. Specifically, at the l th layer, the atom coordinates $\mathbf{r}$ and features $\mathbf{h}$ are updated alternately as follows

$$\mathbf{m}_{ij} = \phi_m(\mathbf{h}_i^l, \mathbf{h}_j^l, d_{ij}^2) \quad (8)$$

$$\mathbf{h}_i^{l+1} = \phi_h\left(\mathbf{h}_i^l, \sum_{j \neq i} \mathbf{m}_{ij}\right) \quad (9)$$

$$\mathbf{r}_i^{l+1} = \mathbf{r}_i^l + \sum_{j \neq i} \frac{\mathbf{r}_i^l - \mathbf{r}_j^l}{d_{ij} + 1} \phi_r(\mathbf{h}_i^l, \mathbf{h}_j^l, d_{ij}^2) \cdot 1_{\text{R-group}} \quad (10)$$

where $d_{ij} = \|\mathbf{r}_i^l - \mathbf{r}_j^l\|$ is the Euclidean distance between nodes i and j , ϕ_m , ϕ_h , and ϕ_r are learnable components parametrized by fully connected neural networks, and $1_{\text{R-group}}$ is the R-group mask (The R-groups are 1, while the contexts are 0.) since we want to keep the context coordinates fixed throughout the EGCL layers. The equivariance of EGCL can be inferred from its inputs. For the update of atom features, ϕ_m and ϕ_h are only related to node features $\mathbf{h}$ and distance d_{ij} between nodes i and j , which are E(3)-invariant. For the update of atom coordinates, it is related to the difference between $\mathbf{r}_i$ and $\mathbf{r}_j$, which preserves E(3)-equivariance. Therefore, EGNN, which is composed of EGCL, is also E(3)-equivariant.

For graph construction, we assign edges to atom pairs that are closer than 4 Å in the 3D space. This ensures that the resulting graph is not too dense, reducing computational costs while allowing for the passage of messages to be passed effectively.

After passing messages through the EGCL layers, we can get new coordinates $\hat{\mathbf{r}} = [\hat{\mathbf{c}}^r, \hat{\mathbf{z}}_t^r]$ that aggregate information about neighbors as well as features $\hat{\mathbf{h}} = [\hat{\mathbf{c}}^h, \hat{\mathbf{z}}_t^h]$. We focus solely on the generated R-groups and disregard the contextual information in the graph. Therefore, the output of the EGNN specifically captures the information pertaining to the R-groups $[\hat{\mathbf{z}}_t^r, \hat{\mathbf{z}}_t^h]$.

During the generative process, the initial noise can be sampled from $\mathcal{N}(\mathbf{0}, \mathbf{I})$ instead of $\mathcal{N}(f(\mathbf{c}), \mathbf{I})$ due to the introduction of $f(\mathbf{c})$ in the diffusion process. Moreover, during the multi R-group generative process, instead of treating each anchor's generation task sequentially, we incorporate the R-groups of the remaining anchors obtained from the previous time step into the context during the message-passing process. This allows us to effectively denoise one anchor's R-group by leveraging the collective information from all anchors.

Model Training. The model can be trained by optimizing the variational lower bound (VLB)^{44,47}

$$-\log(p(\mathbf{x})) \leq D_{\text{KL}}(q(\mathbf{z}_T|\mathbf{x})\|p(\mathbf{z}_T)) - \mathbb{E}_{q(\mathbf{z}_0|\mathbf{x})}[\log p(\mathbf{x}|\mathbf{z}_0)] + \sum_{t=1}^T L_t \quad (11)$$

where the first term is the prior loss, the second term is the reconstruction loss, and the third term is diffusion loss

$$L_t = D_{\text{KL}}(q(\mathbf{z}_{t-1}|\mathbf{x}, \mathbf{z}_t)\|p(\mathbf{z}_{t-1}|\hat{\mathbf{x}}, \mathbf{z}_t)) \quad (12)$$

It should be noticed that the prior loss without learnable parameters is always close to zero, while the reconstruction loss must be estimated.⁴⁷ Thus, we instead minimize the simplified training objective, i.e. the mean squared error between true noise and predicted noise:^{36,42,44}

$$L = \|\epsilon - \hat{\epsilon}_t\|^2 \quad (13)$$

DiffDec was implemented in Python (version 3.10). The models were trained on one GPU (Quadro RTX 5000) for 1,000 steps and saved checkpoints per 100 steps when training the model, batch size, learning rate, EGCL layers, hidden states, and aggregation method were set to 16, 0.0002, 6, 128, and summation, respectively. All generated candidate molecules were canonicalized using RDKit⁴⁹ and compared to the ground truth molecules. Ablation studies in the Results section are based on three models trained with different pocket information and six models trained with different R-group size information. All of these models were trained separately.

Baselines. We compared our approaches with the following baselines:

- **LibINVENT.**²² LibINVENT utilized SMILES strings to encode molecules. It consisted of an encoder-decoder architecture where both the encoder and decoder were recurrent neural networks. It is a tool capable of proposing chemical libraries of compounds with the same scaffolds while maximizing desirable properties using reinforcement learning.
- **Pocket2Mol.**³² Pocket2Mol employed an autoregressive sampling scheme. It is a GNN-based method that generates 3D molecules atom-wise under the context of the pocket. This method has been successfully employed for the structure-based *de novo* drug design.
- **FLAG.**³³ FLAG is a GNN-based model that generates 3D molecules with valid and realistic substructures fragment-by-fragment based on the structure information on proteins. This approach is also applicable to structure-based *de novo* drug design.

Notably, these algorithms were not designed for 3D structure-aware scaffold decoration. Therefore, we adjusted them to our tasks for a fair comparison. The experiment setup is shown in [Supporting Information Text S1](#).

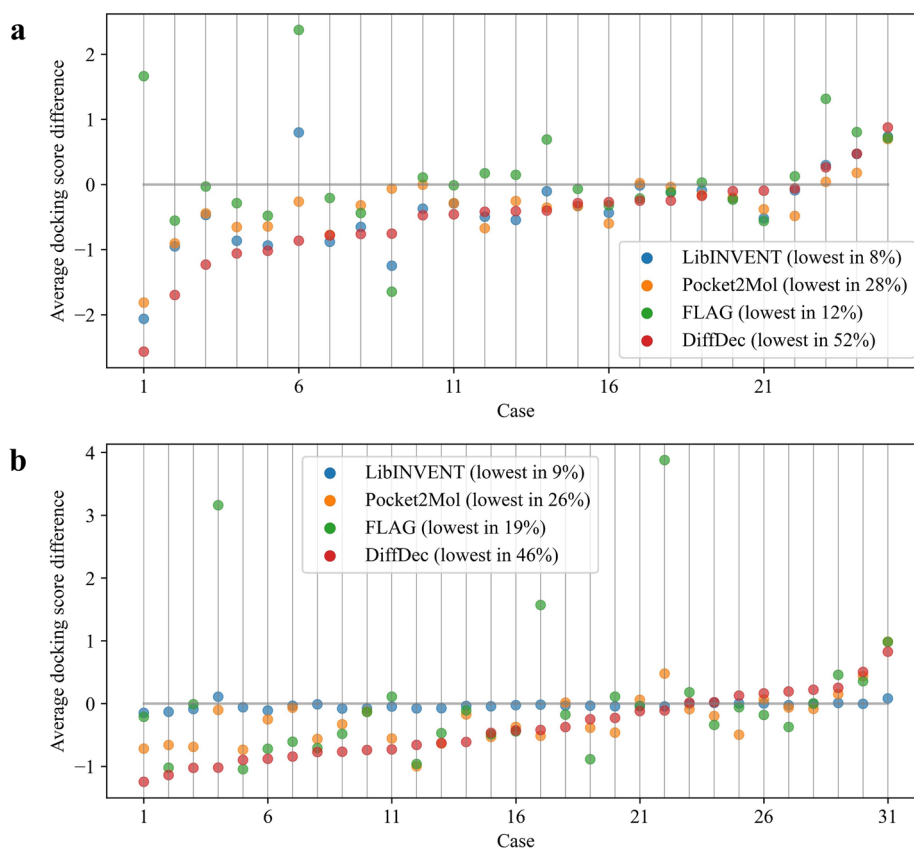
Evaluation Metrics. We assessed the generated molecules with a range of metrics:

- **Validity** refers to the percentage of generated molecules which preserve the original scaffolds and can be parsed by RDKit.
- **Uniqueness** is the proportion of unique compounds in generated ones.
- **Recovery** counts the percentage of generated molecules that recover the reference molecules in the test set.
- **Similarity** means the average Tanimoto similarity⁵⁰ between the generated molecules and reference molecules in the test set.

Table 1. Performance Comparison of Four Methods on the Single R-Group Decoration Task

Method	Validity (% , $\uparrow$)	Uniqueness (% , $\uparrow$)	Recovery (% , $\uparrow$)	Similarity ($\uparrow$)	Vina Score (kcal/mol , $\downarrow$)	High Affinity (% , $\uparrow$)
Scaffolds	-	-	-	-	-7.73	-
Reference molecules	-	-	-	-	-8.47	-
LibINVENT	87.35	41.05	37.33	0.80	-8.10	43.76
Pocket2Mol ^a	51.14	44.27	4.00	0.69	-8.11	<u>44.18</u>
FLAG ^a	87.95	65.30	0.00	0.70	-7.62	34.53
DiffDec (w/o anchors)	<u>93.05</u>	<u>55.33</u>	<u>60.67</u>	<u>0.83</u>	<u>-8.16</u>	42.11
DiffDec	98.00	48.54	69.67	0.86	-8.25	55.06

^aPocket2Mol and FLAG were adjusted to the structure-aware scaffold decoration task.

**Figure 2.** Average Vina score differences between generated molecules and scaffolds across (a) single R-group test cases and (b) multi R-group test cases.

- **Vina score** estimates the binding affinity between the generated molecules and the target protein. Since LibINVENT only generated molecules in the form of SMILES, for a fair comparison, we used the conformations regenerated by RDKit for the molecular docking. Then, we utilized QVina⁵¹ to perform docking and calculate the Vina score (kcal/mol).
- **High affinity** is the percentage of test cases where the Vina scores of the generated molecules are higher than or equal to that of the reference compound.

RESULTS

DiffDec Achieved State-of-the-Art Performance. We compared our method with three deep generative models on two newly constructed reaction-based scaffold decoration data sets, i.e. the single R-group and multi R-group data sets. For each pair of scaffold and pocket in the test sets (43 pairs on the single R-group data set and 102 pairs on the multi R-group data set),

we generated 100 molecules for evaluation to avoid random fluctuation.

On the single R-group decoration task, our method achieved the best performances on metrics validity, recovery, and similarity (Table 1). Specifically, our model attained a validity of 98.00%, while LibINVENT, Pocket2Mol, and FLAG maintained 87.35%, 51.14%, and 87.95%, respectively, indicating that our model could preserve the scaffolds better or generate fewer R-groups in conflict with the scaffolds. The significantly lower performance of Pocket2Mol could be attributed to its atom-wise method in an autoregressive manner to generate atoms and bonds. In terms of the recovery rate, our model exhibited the highest recovery of 69.67%, which was significantly higher than that of LibINVENT (37.33%), Pocket2Mol (4%), and FLAG (0%). These suggested that our method could effectively recover the reference R-groups, which were regarded as realistic decorations for the scaffolds. The worse performance of Pocket2Mol and FLAG may be due to their autoregressive generation scheme, which would lead to

Table 2. Performance Comparison of Four Methods on the Multi R-Group Decoration Task

Method	Validity (% , ↑)	Uniqueness (% , ↑)	Recovery (% , ↑)	Similarity (↑)	Vina Score (kcal/mol, ↓)	High Affinity (% , ↑)
Scaffolds	-	-	-	-	-7.77	-
Reference molecules	-	-	-	-	-8.46	-
LibINVENT	8.62	6.93	26.85	0.70	-7.83	4.57
Pocket2Mol ^a	60.90	50.84	1.34	0.68	-8.04	31.32
FLAG ^a	88.18	<u>64.73</u>	0.00	0.62	-7.73	<u>38.62</u>
DiffDec (w/o anchors)	<u>89.84</u>	71.75	<u>29.54</u>	<u>0.73</u>	<u>-8.15</u>	34.67
DiffDec	97.13	60.12	45.34	0.80	-8.19	40.50

^aPocket2Mol and FLAG were adjusted to the structure-aware scaffold decoration task.

error accumulation and difficulties in recovery of the reference R-groups at each step. In addition, DiffDec reached a uniqueness of 48.54%, indicating that DiffDec could generate a diverse set of R-groups, not just restricted to the references. For the similarity of generated molecules to the references, DiffDec achieved an average value of 0.86, higher than 0.80, 0.69, and 0.70 by LibINVENT, Pocket2Mol, and FLAG, respectively.

To evaluate the binding performance of generated molecules in the target proteins, we docked the compounds to the target protein pockets and computed the corresponding binding affinities, i.e., Vina scores. As expected, our method showed the best performance, with an average docking score of -8.25 kcal/mol, and about 55.06% of generated molecules exhibited higher affinity than the reference ligand, compared with those of LibINVENT (-8.10 and 43.76%), Pocket2Mol (-8.11 and 44.18%), and FLAG (-7.62 and 34.53%). Then, we compared the binding performance of these methods in different cases in detail (Figure 2a). In most cases, the average docking score differences of DiffDec were significantly larger than those of the other three baselines. The molecules generated from DiffDec exhibited the best binding affinity in about 52% of the cases, while the ones from LibINVENT, Pocket2Mol, and FLAG were best for 8%, 28%, and 12% of all cases. We also compared the distributions of ligand efficiencies among these methods for the generated molecules (Figure S2a). As shown in Figure S2a, the distribution of DiffDec exhibited a higher ligand efficiency value at the density peak (approximately 0.34) than the other methods, and the density peak of DiffDec was also the highest. These results illustrated that a significant portion of our generated molecules achieved ligand efficiencies at a favorable level.

On the multi-R-group decoration task, the trend was roughly the same as that of the single R-group task: our DiffDec model still achieved the best performance (Table 2, Figure S2b) with validity of 97.13%, recovery of 45.34%, similarity of 0.80, Vina score of -8.19 kcal/mol, high affinity of 40.50%, and favorable distribution of ligand efficiency. Among these methods, LibINVENT exhibited a low validity and uniqueness of 8.62% and 6.93%. This is because LibINVENT was prone to generating invalid sequences that did not conform to the SMILES syntax rules. For the binding performance comparison of these models, Figure 2b showed that the molecules generated by DiffDec exhibited the best binding affinity in large amounts of the targets (46% vs 9%, 26%, 19% for LibINVENT, Pocket2Mol, FLAG, respectively). These results implied that DiffDec still exhibited strong scaffold decoration capability in the more challenging multi-R-group decoration task. Also, we performed a 5-fold cross-validation on the existing training set. As shown in Tables S2 and S3, the results were basically consistent with those of the independent test set, which demonstrated the results in Table 1 and Table 2 were robust.

To evaluate the geometrical validity of the sampled molecules, we conducted an analysis on the distributions of the bond angles and dihedral angles (which can be calculated by tools like RDKit and PoseBusters⁵²) associated with R-groups, comparing them with the test set by utilizing the Kullback–Leibler (KL) divergence. We identified common bond pairs and bond triples and used RDKit to calculate the bond angles and dihedral angles. As shown in Table 3 and Table 4, DiffDec exhibited substantially

Table 3. KL Divergences of the Bond Angles and Dihedral Angles Associated with R-Groups between the Test Set and the Generated Molecules on the Single R-Group Decoration Task^a

Angles	Pocket2Mol	FLAG	DiffDec
CCC	1.58	0.82	0.81
CCO	2.03	1.54	1.44
CNC	1.55	0.79	0.77
NCC	1.57	1.22	0.59
CC=O	2.36	1.06	0.81
CCCC	2.28	1.98	1.82
cccc	1.68	0.38	0.88
CCCO	2.94	2.65	1.14
Cccc	1.84	0.81	1.29
CC=CC	4.31	3.54	3.88

^aThe lower letters represent the atoms in the aromatic rings.

lower KL divergences than Pocket2Mol and FLAG (fragment-wise generation scheme) over most of the angle types on both single and multi R-group decoration tasks. These showed that our approach could generate molecules with realistic sub-

Table 4. KL Divergences of the Bond Angles and Dihedral Angles Associated with R-Groups between the Test Set and the Generated Molecules on the Multi R-Group Decoration Task^a

Angles	Pocket2Mol	FLAG	DiffDec
CCC	1.78	0.92	0.95
CCO	2.38	1.25	1.42
CNC	2.33	1.33	0.90
NCC	1.63	1.11	0.85
CC=O	1.93	1.11	0.81
CCCC	1.69	1.47	1.39
cccc	1.58	0.52	0.74
CCCO	2.29	2.20	0.69
Cccc	1.38	1.37	1.36
CC=CC	3.78	2.79	2.21

^aThe lower letters represent the atoms in the aromatic rings.

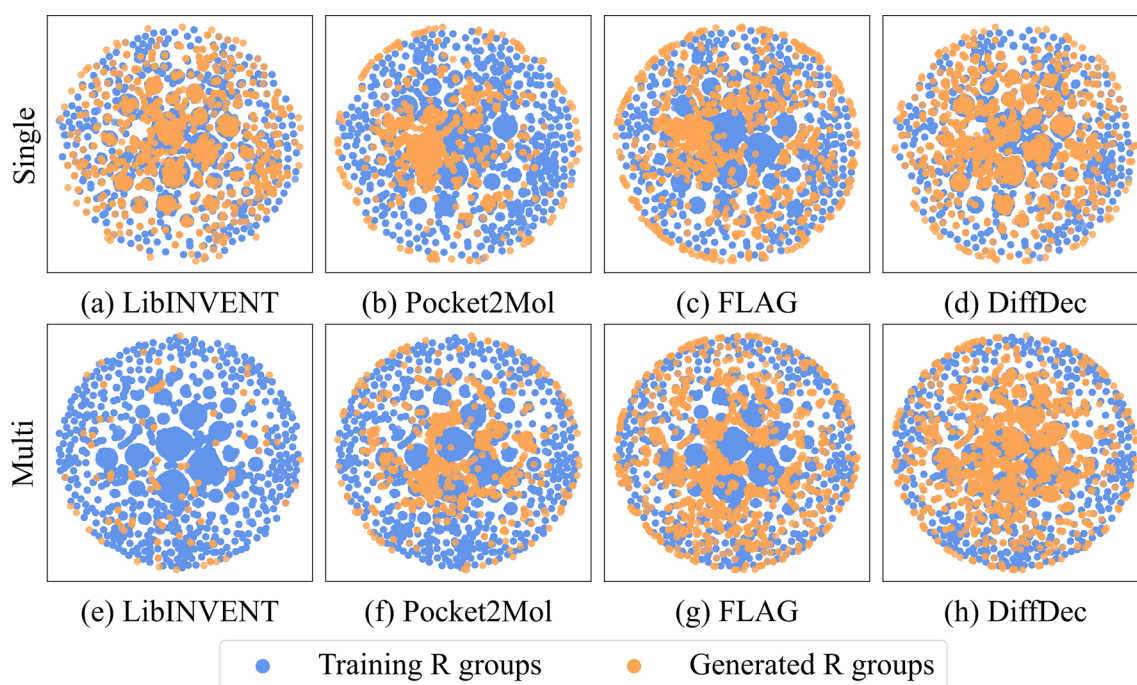


Figure 3. T-SNE plots of training R-groups and generated R-groups from LibINVENT, Pocket2Mol, FLAG, and DiffDec on single and multi-R-group decoration tasks.

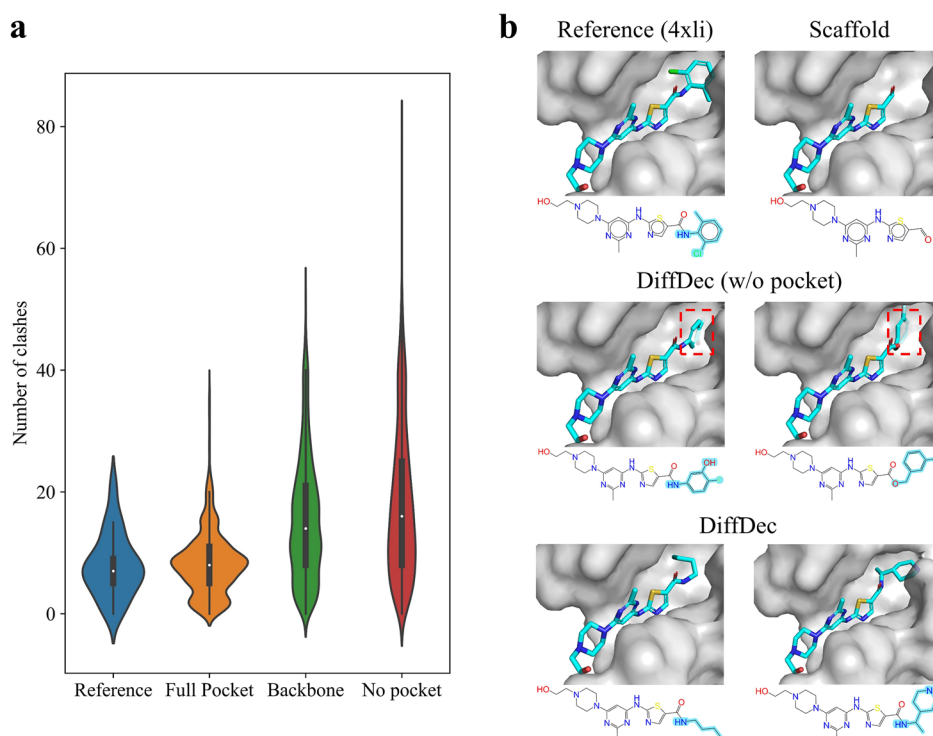


Figure 4. Results of the ablation study of pocket information. (a) Distribution of numbers of clashes in the reference test set and molecules sampled from DiffDec models with different pocket representations. (b) R-groups generated by DiffDec without pockets and with pockets as context (PDB id is 4xli).

structures while maintaining the geometric properties of the data well.

We further visualized the distributions of R-groups on the single and multi R-group decoration tasks. As shown in Figure 3a-d, most of the R-groups generated by LibINVENT and FLAG appeared in the training set, while Pocket2Mol generated

a large proportion of compounds outside of the training sets due to its sequential atom-by-atom generation strategy that would lead to unrealistic unseen substructures. On the contrary, DiffDec achieved a balanced result to keep the distribution of training data and to generate novel R-groups outside the training space. On the multi-R-group decoration task, all methods

Table 5. Ablation Study of the Fake Atom Module on the Single R-Group Decoration Task

Method	Validity (% , $\uparrow$)	Uniqueness (% , $\uparrow$)	Recovery (% , $\uparrow$)	Similarity ($\uparrow$)	Vina Score (kcal/mol, $\downarrow$)	High Affinity (% , $\uparrow$)
DiffDec (w/o anchors, size GNN)	95.51	55.23	51.67	0.82	-8.09	38.85
DiffDec (w/o anchors)	93.05	55.33	60.67	0.83	-8.16	42.11
DiffDec (w/o anchors, given size)	95.16	38.33	62.00	0.85	-8.18	46.57
DiffDec (size GNN)	97.87	48.27	54.67	0.85	-8.15	49.89
DiffDec	98.00	48.54	69.67	0.86	-8.25	55.06
DiffDec (given size)	98.07	41.26	79.33	0.88	-8.29	60.97

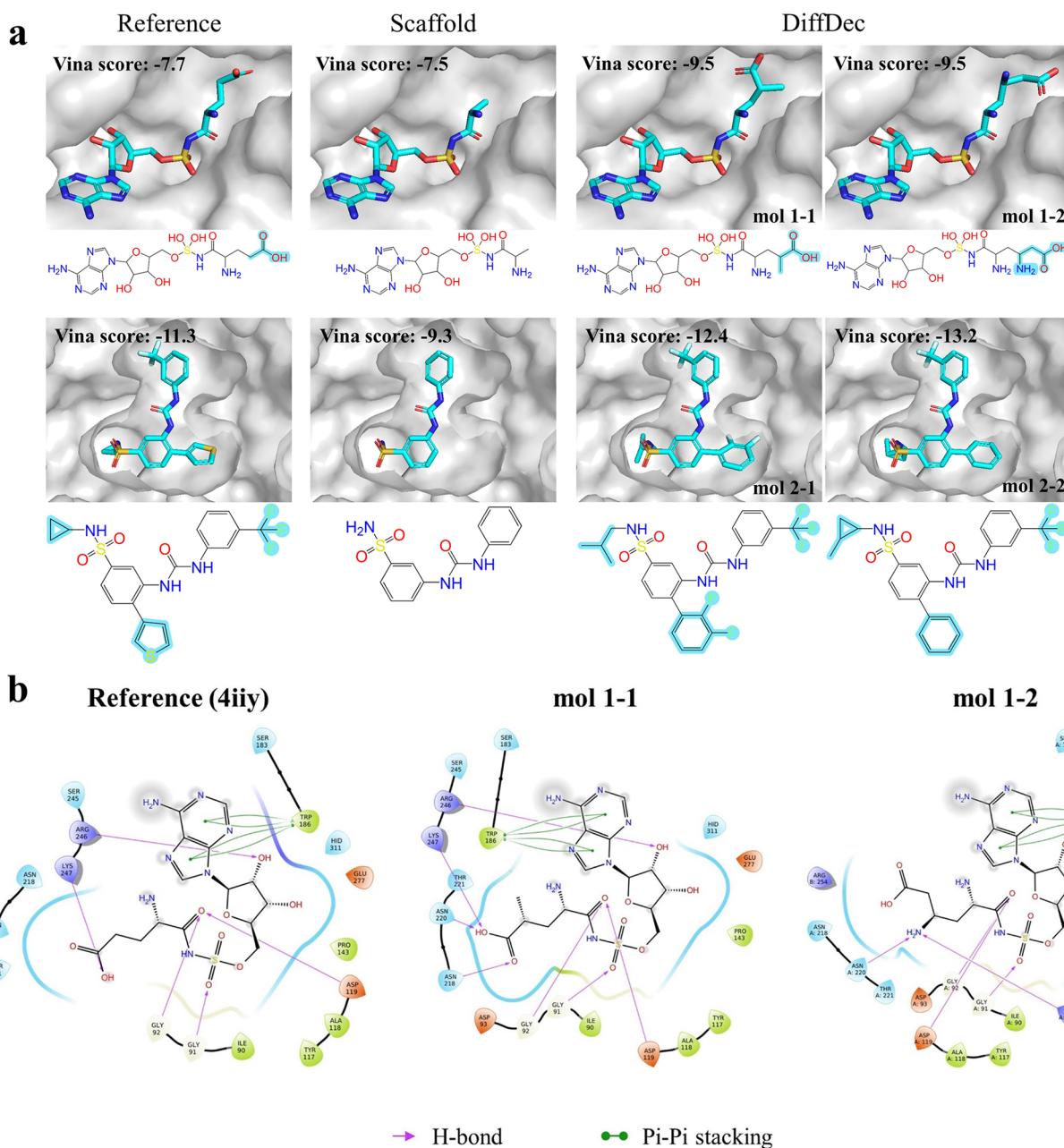


Figure 5. (a) R-groups generated by DiffDec on single and multi-R-group decoration tasks. The two rows represent two different scaffolds and pockets (PDB ids are 4iiy and 4ja8, respectively). (b) Protein–ligand interaction of case 4iiy (PDB id is 4iiy).

showed similar distributions except LibINVENT which generated quite a small number of R-groups (Figure 3e) due to the low validity (8.62%). These results demonstrated that DiffDec could primarily recover the R-groups in the training set and also generate novel molecules.

DiffDec Could Generate R-Groups Well without Provided Anchors. In the actual scenarios of lead optimization, the specific location of anchors is usually not known, so we investigated the R-group generation performance of DiffDec without provided anchors (w/o anchors). When not providing anchor information, on the single R-group decoration

task, DiffDec (w/o anchors) achieved validity, recovery, and similarity of 93.05%, 60.67%, and 0.83, which were only 4.95%, 9.00%, and 0.03 lower than DiffDec (98.00%, 69.67%, and 0.86) (Table 1). For binding affinity of generated molecules to the target pocket, DiffDec (w/o anchors) exhibited an average Vina score and high affinity percentage of -8.16 kcal/mol and 42.11%, which were comparable to those of DiffDec (-8.25 and 55.06%). The reason for these declines is that DiffDec without anchors provides information needed to consider both the generation of the R-group and the appropriate anchor to attach it to, which resulted in more challenging generation. On the multi R-group decoration task, DiffDec (w/o anchors) achieved validity of 89.84%, recovery of 29.54%, similarity of 0.73, Vina score of -8.15 kcal/mol, and high affinity of 34.67%, which decreased by 7.29%, 15.80%, 0.07, -0.04 , and 5.83% compared to DiffDec (97.13%, 45.34%, 0.80, -8.19 , and 40.50%) (Table 2). Even so, DiffDec (without anchor) outperformed the other three baseline models with anchor information on most metrics. Therefore, even if we did not provide anchors, our method could automatically identify anchors and generate R-groups with acceptable quality and comparable performance.

The Importance of Pocket Information. To illustrate DiffDec's awareness of pocket information, we retrained DiffDec and evaluated the performances under two conditions: with pocket backbone information and without any pocket information. We calculated the number of clashes between the generated R-groups and the surrounding pockets for comparison. The clash is defined as when the distance between two atoms is less than the sum of their van der Waals radius. As shown in Figure 4a, the number of clashes generated by DiffDec with full-atomic pocket representation (with an average of 8 clashes per molecule) was almost identical to that of the reference complex in the test set (with an average of 7 clashes per molecule). When the protein pocket was represented as the backbone or removed, the average number of clashes was increased to 15 or 18, respectively. Then, we performed R-group generation for ABL2/ARG kinase (4xli)⁵³ as an example by DiffDec without pockets and with pockets as the context for comparison (Figure 4b). When using DiffDec without pocket information, the generated R-groups collided with the surrounding amino acid residues in the pocket (red dashed box). While in DiffDec with full-atomic pocket information, the R-groups were generated according to the condition of the surrounding pockets and orientated to an appropriate position to cause fewer clashes with the pocket.

The Importance of the Fake Atom Module. To illustrate the importance of the fake atom module in DiffDec, we removed the module from DiffDec and utilized a trained graph neural network to predict the size of the R-groups (size GNN). By comparison, we showed the results of DiffDec using the R-group sizes in ground truth (given size). As shown in Table 5, DiffDec with size GNN achieved a recovery of 54.67%, Vina score of -8.15 kcal/mol, and high affinity ratio of 49.89%. These values were significantly lower than those of DiffDec (69.67%, -8.25 kcal/mol, and 55.06%) and DiffDec (given size) (79.33%, -8.29 kcal/mol, and 60.97%). In addition, the DiffDec (w/o anchors) model also exhibited a better recovery ratio, Vina score, and high affinity ratio than DiffDec (w/o anchors, size GNN). These results suggested that the fake atom module could predict a better R-group size than GNN to accommodate the surrounding pocket and benefit subsequent generation.

Case Study. We provided several case studies for DiffDec on single and multi R-group generation and showed examples of

generated molecules with higher binding affinities (lower Vina score) than the reference compounds (Figures 5, S5, and S6). For the examples shown, we used the as-is geometries generated by DiffDec to perform docking and calculate the Vina score. As shown in Figures 5, S5, and S6, DiffDec could generate compounds that bound geometrically complementary to the target pocket, and the generated molecules exhibited significantly improved binding affinities compared to those of the scaffold and reference ligands. Then we analyzed the binding modes and interactions of molecule 1-1 and 1-2 with surrounding amino acid residues using Maestro of Schrödinger⁵⁴ to interpret the binding affinity improvement of these generated molecules. Compared to the reference molecule, the generated R-groups of molecules 1-1 and 1-2 formed an additional hydrogen bond with Asn220 of the protein, and molecule 1-1 formed another additional H-bond interaction with Asn218 (Figure 5). Thus, the binding affinities of molecules 1-1 and 1-2 were higher than those of the reference ligand. When decorating multi-R-groups (Figure 5 mol2-1 and mol2-2), DiffDec could also generate molecules with significantly improved affinities and recover the complex trifluoromethyl group in the reference compound. Therefore, DiffDec is capable of generating molecules with improved binding affinities and more favorable binding interactions for lead optimization. More examples and their Vina scores were summarized in Supporting Information Figures S5–S6.

Furthermore, we selected two isoforms JAK1 (PDB id is 6bbu) and JAK2 (PDB id is 6bbv)⁵⁵ as examples and used an identical scaffold (Figure S3) for scaffold decoration inside these two isomeric proteins. The generated molecules were cross-docked into these two proteins, respectively, to calculate the Vina score, and the distributions are shown in Figure S4. The results revealed that DiffDec can generate R-groups with specificity and selectivity for specific protein pockets.

DISCUSSION AND CONCLUSIONS

In this study, we developed an E(3)-equivariant 3D-conditional diffusion model, DiffDec, for molecular scaffold decoration conditioned on 3D protein pockets. The model could generate single and multi R-groups for scaffold decoration inside protein pockets based on two newly constructed benchmark data sets.

Comprehensive evaluations demonstrated that DiffDec achieved state-of-the-art performance on R-group generation and outperformed the baseline methods on conventional metrics, such as validity, recovery, similarity, and binding affinity. The analysis of the bond angles and dihedral angles proved the geometric validity of our approach. The t-SNE projection indicated that our model could not only recover the R-groups in the training set well but also generate a number of novel R-groups out of learning space. In addition, our method could identify the growth anchors and generate R-groups well for scaffolds without provided anchors. Besides, our model conditioned on full-atomic pocket representation generated R-groups with a minimum number of clashes. Our newly introduced fake atom module could predict the suitable R-group sizes according to the surrounding pocket. Case studies indicated that DiffDec generated decorated molecules which were geometrically complementary to the target pocket and exhibited higher binding affinities and more favorable interactions with the surrounding pocket than the reference compound.

However, there are still some limitations of DiffDec in the current framework. First, the output of DiffDec is a 3D atomic

point cloud of the generated R-groups, and the covalent bonds between pairs of atoms are additionally computed by a bond inference algorithm (e.g., OpenBabel⁵⁶). Therefore, the accuracy of OpenBabel will impact the validity and recovery ratio of the generated molecules. Optimization of this bond generation problem could be the direction of our further work. Second, the R-group generation process in DiffDec requires step-by-step denoising from Gaussian noise, which is computationally intensive and limits the sampling efficiency. This may be solved by combining distributed CPU and GPU computations in a mixed fashion.

In summary, DiffDec provides a flexible and effective method for 3D structure-aware scaffold decoration. We believe that our method will provide a practical solution in real scenarios and inspire future scaffold decoration work.

■ ASSOCIATED CONTENT

Data Availability Statement

The data sets and source code are available at <https://github.com/biomed-AI/DiffDec>.

SI Supporting Information

The Supporting Information is available free of charge at <https://pubs.acs.org/doi/10.1021/acs.jcim.3c01466>.

Text S1, experiment setup of the other methods; Table S1, data statistics for the CrossDocked, single R-group and multi R-group data sets; Figure S1, distributions of R-groups' size; Tables S2, S3, performance of the 5-fold cross-validation on single and multi R-group decoration tasks; Figure S2, distributions of ligand efficiency of molecules on single and multi R-group decoration tasks; Figure S3, the scaffold used for decoration with JAK1 and JAK2; Figure S4, distributions of Vina scores with JAK1 and JAK2; and Figures S5, S6, more example generations of each sample by DiffDec (PDF) SMILES file (ZIP)

■ AUTHOR INFORMATION

Corresponding Authors

Jinping Lei – School of Pharmaceutical Sciences, Sun Yat-sen University, Guangzhou 510006, China; orcid.org/0000-0002-6888-2973; Email: leiip@mail.sysu.edu.cn

Yuedong Yang – School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou 510006, China; orcid.org/0000-0002-6782-2813; Email: yangyd25@mail.sysu.edu.cn

Authors

Junjie Xie – School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou 510006, China; AixplorerBio Inc., Jiaxing 314031, China; orcid.org/0009-0008-9927-0452

Sheng Chen – School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou 510006, China; AixplorerBio Inc., Jiaxing 314031, China; orcid.org/0000-0003-1428-6778

Complete contact information is available at: <https://pubs.acs.org/doi/10.1021/acs.jcim.3c01466>

Author Contributions

[†]JX and SC contributed equally to this work.

Notes

The authors declare no competing financial interest.

■ ACKNOWLEDGMENTS

The authors thank Dr. David Daqiang Xu for providing useful suggestions for paper revisions. This work has been supported by the National Key R&D Program of China (2023YFF1204900), the National Natural Science Foundation of China (T2394502, 82003651), the Guangdong-Hong Kong Technology Cooperation Funding Scheme (2023A0505010015), and the Guangzhou Basic and Applied Basic Research Project (202201011795).

■ REFERENCES

- (1) Macalino, S. J. Y.; Gosu, V.; Hong, S.; Choi, S. Role of computer-aided drug design in modern drug discovery. *Arch. Pharmacol. Res.* **2015**, *38*, 1686–1701.
- (2) Yu, W.; MacKerell, A. D. Computer-aided drug design methods. *Methods Mol. Biol.* **2017**, *1520*, 85–106.
- (3) Kim, K.-H.; Kim, N. D.; Seong, B.-L. Pharmacophore-based virtual screening: a review of recent applications. *Expert Opin. Drug Discovery* **2010**, *5*, 205–222.
- (4) Achary, P. G. Applications of quantitative structure-Activity relationships (QSAR) based virtual screening in drug design: a review. *Mini-Rev. Med. Chem.* **2020**, *20*, 1375–1388.
- (5) Acharya, C.; Coop, A.; E Polli, J.; D MacKerell, A. Recent advances in ligand-based drug design: relevance and utility of the conformationally sampled pharmacophore approach. *Curr. Comput.-Aided Drug Des.* **2011**, *7*, 10–22.
- (6) Blaschke, T.; Olivecrona, M.; Engkvist, O.; Bajorath, J.; Chen, H. Application of generative autoencoder in de novo molecular design. *Mol. Inf.* **2018**, *37*, 1700123.
- (7) Xue, D.; Gong, Y.; Yang, Z.; Chuai, G.; Qu, S.; Shen, A.; Yu, J.; Liu, Q. Advances and challenges in deep generative models for de novo molecule generation. *Wiley Interdiscip. Rev.: Comput. Mol. Sci.* **2019**, *9*, No. e1395.
- (8) Segler, M. H.; Kogej, T.; Tyrchan, C.; Waller, M. P. Generating focused molecule libraries for drug discovery with recurrent neural networks. *ACS Cent. Sci.* **2018**, *4*, 120–131.
- (9) Langevin, M.; Minoux, H.; Levesque, M.; Bianciotto, M. Scaffold-constrained molecular generation. *J. Chem. Inf. Model.* **2020**, *60*, 5637–5646.
- (10) Wang, J.; Zheng, S.; Chen, J.; Yang, Y. Meta learning for low-resource molecular optimization. *J. Chem. Inf. Model.* **2021**, *61*, 1627–1636.
- (11) Zheng, S.; Tan, Y.; Wang, Z.; Li, C.; Zhang, Z.; Sang, X.; Chen, H.; Yang, Y. Accelerated rational PROTAC design via deep learning and molecular simulations. *Nat. Mach. Intell.* **2022**, *4*, 739–748.
- (12) Tan, Y.; Dai, L.; Huang, W.; Guo, Y.; Zheng, S.; Lei, J.; Chen, H.; Yang, Y. DRlinker: Deep Reinforcement Learning for Optimization in Fragment Linking Design. *J. Chem. Inf. Model.* **2022**, *62*, S907–S917.
- (13) Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780.
- (14) Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L. u.; Polosukhin, I. Attention is All you Need. *Advances in Neural Information Processing Systems*. 2017.
- (15) Li, Y.; Hu, J.; Wang, Y.; Zhou, J.; Zhang, L.; Liu, Z. DeepScaffold: a comprehensive tool for scaffold-based de novo drug discovery using deep learning. *J. Chem. Inf. Model.* **2020**, *60*, 77–91.
- (16) Lim, J.; Hwang, S.-Y.; Moon, S.; Kim, S.; Kim, W. Y. Scaffold-based molecular design with a graph generative model. *Chem. Sci.* **2020**, *11*, 1153–1164.
- (17) Chen, Z.; Min, M. R.; Parthasarathy, S.; Ning, X. A deep generative model for molecule optimization via one fragment modification. *Nat. Mach. Intell.* **2021**, *3*, 1040–1049.
- (18) Garcia Satorras, V.; Hoogeboom, E.; Fuchs, F.; Posner, I.; Welling, M. E(n) Equivariant Normalizing Flows. *Advances in Neural Information Processing Systems*. 2021; pp 4181–4192.
- (19) Hughes, J. P.; Rees, S.; Kalindjian, S. B.; Philpott, K. L. Principles of early drug discovery. *Br. J. Pharmacol.* **2011**, *162*, 1239–1249.

- (20) Maziarz, K.; Jackson-Flux, H.; Cameron, P.; Sirockin, F.; Schneider, N.; Stiefl, N.; Segler, M.; Brockschmidt, M. Learning to Extend Molecular Scaffolds with Structural Motifs. *arXiv preprint*. 2022, arXiv:2103.03864. <https://arxiv.org/abs/2103.03864> (accessed 2024-01-22).
- (21) Yang, Y.; Zheng, S.; Su, S.; Zhao, C.; Xu, J.; Chen, H. SyntaLinker: automatic fragment linking with deep conditional transformer neural networks. *Chem. Sci.* **2020**, *11*, 8312–8322.
- (22) Fialková, V.; Zhao, J.; Papadopoulos, K.; Engkvist, O.; Bjerrum, E. J.; Kogej, T.; Patronov, A. LibINVENT: reaction-based generative scaffold decoration for in silico library design. *J. Chem. Inf. Model.* **2022**, *62*, 2046–2063.
- (23) Weininger, D. SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules. *J. Chem. Inf. Comput. Sci.* **1988**, *28*, 31–36.
- (24) Ferreira, L. G.; Dos Santos, R. N.; Oliva, G.; Andricopulo, A. D. Molecular docking and structure-based drug design strategies. *Molecules* **2015**, *20*, 13384–13421.
- (25) Rudrapal, M.; Khairnar, S. J.; Jadhav, A. G. In *Drug Repurposing*; Badria, F. A., Ed.; IntechOpen: Rijeka, 2020; Chapter 1.
- (26) Drenth, J. *Principles of protein X-ray crystallography*; Springer Science & Business Media, 2007.
- (27) Bai, X.-C.; McMullan, G.; Scheres, S. H. How cryo-EM is revolutionizing structural biology. *Trends Biochem. Sci.* **2015**, *40*, 49–57.
- (28) Jumper, J.; Evans, R.; Pritzel, A.; Green, T.; Figurnov, M.; Ronneberger, O.; Tunyasuvunakool, K.; Bates, R.; Židek, A.; Potapenko, A.; Bridgland, A.; Meyer, C.; Kohl, S. A. A.; Ballard, A. J.; Cowie, A.; Romera-Paredes, B.; Nikolov, S.; Jain, R.; Adler, J.; Back, T.; Petersen, S.; Reiman, D.; Clancy, E.; Zielinski, M.; Steinegger, M.; Pacholska, M.; Berghammer, T.; Bodenstein, S.; Silver, D.; Vinyals, O.; Senior, A. W.; Kavukcuoglu, K.; Kohli, P.; Hassabis, D. Highly accurate protein structure prediction with AlphaFold. *Nature* **2021**, *596*, 583–589.
- (29) Lin, Z.; Akin, H.; Rao, R.; Hie, B.; Zhu, Z.; Lu, W.; Smetanin, N.; Verkuil, R.; Kabeli, O.; Shmueli, Y.; dos Santos Costa, A.; Fazel-Zarandi, M.; Sercu, T.; Candido, S.; Rives, A. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science* **2023**, *379*, 1123–1130.
- (30) Hollingsworth, S. A.; Dror, R. O. Molecular dynamics simulation for all. *Neuron* **2018**, *99*, 1129–1143.
- (31) Morris, G. M.; Lim-Wilby, M. S. L. Molecular docking. *Methods Mol. Biol.* **2008**, *443*, 365–82.
- (32) Peng, X.; Luo, S.; Guan, J.; Xie, Q.; Peng, J.; Ma, J. Pocket2mol: Efficient molecular sampling based on 3d protein pockets. *International Conference on Machine Learning*. 2022; pp 17644–17655.
- (33) Zhang, Z.; Min, Y.; Zheng, S.; Liu, Q. Molecule generation for target protein binding with structural motifs *The Eleventh International Conference on Learning Representations*. 2023.
- (34) Aksan, E.; Kaufmann, M.; Cao, P.; Hilliges, O. A spatio-temporal transformer for 3d human motion prediction. *2021 International Conference on 3D Vision (3DV)*. 2021; pp 565–574.
- (35) Sohl-Dickstein, J.; Weiss, E.; Maheswaranathan, N.; Ganguli, S. Deep unsupervised learning using nonequilibrium thermodynamics. *International Conference on Machine Learning*. 2015; pp 2256–2265.
- (36) Igashov, I.; Stärk, H.; Vignac, C.; Satorras, V. G.; Frossard, P.; Welling, M.; Bronstein, M.; Correia, B. Equivariant 3D-Conditional Diffusion Models for Molecular Linker Design. *arXiv preprint*. 2022, arXiv:2210.05274. <https://arxiv.org/abs/2210.05274> (accessed 2024-01-22).
- (37) Schneuing, A.; Du, Y.; Harris, C.; Jamasb, A.; Igashov, I.; Du, W.; Blundell, T.; Lió, P.; Gomes, C.; Welling, M.; Bronstein, M.; Correia, B. Structure-based Drug Design with Equivariant Diffusion Models. *arXiv preprint*. 2023, arXiv:2210.13695. <https://arxiv.org/abs/2210.13695> (accessed 2024-01-22).
- (38) Runcie, N. T.; Mey, A. S. SILVR: Guided Diffusion for Molecule Generation. *J. Chem. Inf. Model.* **2023**, *63*, 5996–6005.
- (39) Francoeur, P. G.; Masuda, T.; Sunseri, J.; Jia, A.; Iovanisci, R. B.; Snyder, I.; Koes, D. R. Three-dimensional convolutional neural networks and a cross-docked data set for structure-based drug design. *J. Chem. Inf. Model.* **2020**, *60*, 4200–4215.
- (40) Steinegger, M.; Söding, J. MMseqs2 enables sensitive protein sequence searching for the analysis of massive data sets. *Nat. Biotechnol.* **2017**, *35*, 1026–1028.
- (41) Luo, S.; Guan, J.; Ma, J.; Peng, J. A 3D Generative Model for Structure-Based Drug Design *Advances in Neural Information Processing Systems*. 2021; pp 6229–6239.
- (42) Ho, J.; Jain, A.; Abbeel, P. Denoising Diffusion Probabilistic Models. *Advances in Neural Information Processing Systems*. 2020; pp 6840–6851.
- (43) Guan, J.; Qian, W. W.; Peng, X.; Su, Y.; Peng, J.; Ma, J. 3D Equivariant Diffusion for Target-Aware Molecule Generation and Affinity Prediction. *arXiv preprint*. 2023, arXiv:2303.03543. <https://arxiv.org/abs/2303.03543> (accessed 2024-01-22).
- (44) Kingma, D.; Salimans, T.; Poole, B.; Ho, J. Variational Diffusion Models. *Advances in Neural Information Processing Systems*. 2021; pp 21696–21707.
- (45) Song, Y.; Sohl-Dickstein, J.; Kingma, D. P.; Kumar, A.; Ermon, S.; Poole, B. Score-Based Generative Modeling through Stochastic Differential Equations. *arXiv preprint*. 2021, arXiv:2011.13456. <https://arxiv.org/abs/2011.13456> (accessed 2024-01-22).
- (46) Xu, M.; Yu, L.; Song, Y.; Shi, C.; Ermon, S.; Tang, J. GeoDiff: a Geometric Diffusion Model for Molecular Conformation Generation. *arXiv preprint*. 2022, arXiv:2203.02923. <https://arxiv.org/abs/2203.02923> (accessed 2024-01-22).
- (47) Hoogeboom, E.; Satorras, V. G.; Vignac, C.; Welling, M. Equivariant diffusion for molecule generation in 3d. *International Conference on Machine Learning*. 2022; p 8867–8887.
- (48) Satorras, V. G.; Hoogeboom, E.; Welling, M. E. (n) equivariant graph neural networks. *International conference on machine learning*. 2021; pp 9323–9332.
- (49) Landrum, G. Rdkit: Open-source cheminformatics software. 2016. https://github.com/rdkit/rdkit/releases/tag/Release_2016_09_4 (accessed 2024-01-22).
- (50) Bajusz, D.; Rácz, A.; Héberger, K. Why is Tanimoto index an appropriate choice for fingerprint-based similarity calculations? *J. Cheminf.* **2015**, *7*, 20.
- (51) Alhossary, A.; Handoko, S. D.; Mu, Y.; Kwok, C.-K. Fast, accurate, and reliable molecular docking with QuickVina 2. *Bioinformatics* **2015**, *31*, 2214–2216.
- (52) Buttenschon, M.; Morris, G. M.; Deane, C. M. PoseBusters: AI-based docking methods fail to generate physically valid poses or generalise to novel sequences. *Chem. Sci.* **2024**, DOI: 10.1039/D3SC04185A.
- (53) Ha, B. H.; Simpson, M. A.; Koleske, A. J.; Boggon, T. J. Structure of the ABL2/ARG kinase in complex with dasatinib. *Acta Crystallogr., Sect. F: Struct. Biol. Commun.* **2015**, *71*, 443–448.
- (54) Maestro, S. *Maestro*. 2023.
- (55) Vazquez, M. L.; Kaila, N.; Strohschach, J. W.; Trzuppek, J. D.; Brown, M. F.; Flanagan, M. E.; Mitton-Fry, M. J.; Johnson, T. A.; TenBrink, R. E.; Arnold, E. P.; Basak, A.; Heasley, S. E.; Kwon, S.; Langille, J.; Parikh, M. D.; Griffin, S. H.; Casavant, J. M.; Duclos, B. A.; Fenwick, A. E.; Harris, T. M.; Han, S.; Caspers, N.; Dowty, M. E.; Yang, X.; Banker, M. E.; Hegen, M.; Symanowicz, P. T.; Li, L.; Wang, L.; Lin, T. H.; Jussif, J.; Clark, J. D.; Telliez, J.-B.; Robinson, R. P.; Unwalla, R. Identification of N-{cis-3-[Methyl(7H-pyrrolo[2,3-d]pyrimidin-4-yl)-amino]cyclobutyl}prop ane-1-sulfonamide (PF-04965842): A Selective JAK1 Clinical Candidate for the Treatment of Autoimmune Diseases. *J. Med. Chem.* **2018**, *61*, 1130–1152.
- (56) O’Boyle, N. M.; Banck, M.; James, C. A.; Morley, C.; Vandermeersch, T.; Hutchison, G. R. Open Babel: An open chemical toolbox. *J. Cheminf.* **2011**, *3*, 33.