

MOG: MOLECULAR OUT-OF-DISTRIBUTION GENERATION WITH ENERGY-BASED MODELS

Anonymous authors

Paper under double-blind review

ABSTRACT

Recent advances of deep generative models opened up a new horizon for *de novo* drug discovery. However, a well-known problem of existing works on molecule generation is that the generated molecules highly resemble those in the training set. Models that do not require training molecules such as RL-based models circumvent this problem, but they lack information about existing molecules. In this paper, we propose *Molecular Out-of-distribution Generation* (MOG), a novel framework that explicitly generates OOD molecules with respect to given molecules by combining two aspects of energy-based models (EBMs): generation and out-of-distribution (OOD) detection. This can be done by introducing *multiple energy pivots* to Langevin dynamics in generation and increase energy instead of minimizing it. We also utilize a property predictor to provide the property gradient of molecules to the modified Langevin dynamics. To validate the ability to explore the chemical space beyond the known molecular distribution, we experiment with MOG to generate molecules of high absolute values of docking score, which is the affinity score based on a physical binding simulation between a target protein and a given molecule. Docking score is a strong proxy to drug activity unlike penalized logP or QED and requires stronger exploration as it is nonlinear to local molecular structures and has many local optima. MOG is able to generate molecules with high docking scores compared to existing methods. Moreover, we further show the energy-increasing strategy based on EBMs can be universally applied to existing models and enhance their resulting novelty.

1 INTRODUCTION

To find molecules with desired chemical properties comprises the core of drug discovery. The target properties encompass from relatively simple chemical statistics to complex scores that depend on three-dimensional interaction. The former includes octanol-water partition coefficients (logP), synthetic accessibility (SA), quantitative estimate of drug-likeness (QED) (Bickerton et al., 2012), and molecular weight (MW). The latter includes quantum-mechanical properties, molecular dynamics, and docking scores (Ferreira et al., 2015).

Deep molecular generative models rise as promising substitutes for conventional experimental drug discovery approaches. The most common metrics to evaluate the generated molecules among the above properties are penalized logP and QED score (Jin et al., 2018; You et al., 2018; Shi et al., 2019; Zang & Wang, 2020; Luo et al., 2021; Liu et al., 2021). However, as Coley (2020), Cieplinski et al. (2020), and Xie et al. (2020) have criticized, optimization of these properties is far apart from the real-world drug discovery problems. For example, the best optimized molecule in terms of penalized logP in the state-of-the-art model is a trivial long chain of maximum number of carbons (Luo et al., 2021), since penalized logP prefers large molecules. On the other hand, docking score is known as a powerful proxy to drug activity (Cieplinski et al., 2020). The score is based on the three-dimensional binding simulation of a target protein and a drug candidate. In this respect, we target to construct a generative model that optimizes docking scores in this paper. Unlike penalized logP and QED score, docking score is nonlinear to local molecular structures and has many local optima (Coley, 2020). Therefore, more powerful exploration in the chemical space is essential to generate molecules that dock well, and further, be applicable to the real-world.

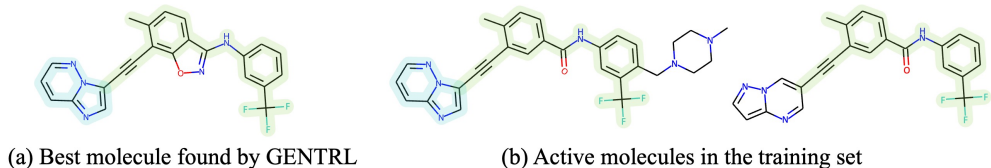


Figure 1: **The molecules found by Zhavoronkov et al. (2019) and its training molecules.** Zhavoronkov et al. (2019) trained a deep RL-based model, GENTRL, on a large set of known discoidin domain receptor 1 (DDR1) inhibitors, and found the best potent DDR1 inhibitor (a) (referred to as ‘compound 1’ in the original paper). However, as Walters & Murcko (2020) have pointed out, the molecule exhibits great similarity compared to some of the DDR1 inhibitors included in the training set (b), and the application to the real-world is limited. The duplicated substructures are marked as green and blue areas.

Unfortunately, the common pitfall of existing molecule generation models based on distributional learning is that the exploration is confined to the training distribution, and the generated molecules highly resemble those in their training set. Walters & Murcko (2020) pointed out the best molecule found by the model of Zhavoronkov et al. (2019) exhibits “the striking similarity” with known active molecules included in the training set (see Figure 1). Models that do not require training molecules are free from this problem, but they introduce other problems such as long training time, the sensitivity of balance between exploration and exploitation, large variance, and importantly, a lack of information about the known distribution.

In this paper, we introduce a new perspective that applies an out-of-distribution (OOD) detection scheme to generate OOD molecules to tackle the limited exploration problem of deep generative models. OOD generation touches the fundamental aim of *de novo* drug discovery, which is to discover *unknown* molecules that lie beyond our current knowledge. Specifically, we propose *Molecular Out-of-distribution Generation* (MOG), a novel framework to generate docking score-optimized molecules with exploration beyond the known molecular distribution. The distribution can be comprised of any molecules, such as training molecules or known active molecules against a certain protein. The proposed MOG explicitly aims to generate OOD molecules against these in-distribution molecules using an energy-based model (EBM). The modified Langevin dynamics of MOG in the generation phase pushes molecules into regions of high energy values, towards those of *energy pivots*. This corresponds to push the molecules into regions of low likelihoods, thereby largely encourages the exploration in the chemical space. We additionally incorporate a property predictor that predicts docking scores into the modified Langevin dynamics, thus generates molecules that are novel and optimized at the same time.

We experimentally validate the exploration ability of MOG by random generation and docking score optimization with three target proteins. MOG is able to generate molecules of significantly improved novelty and docking scores compared to baselines in all of the experiments. We further augment MARS with our strategy which increases energy using an EBM and show our strategy can be widely applied to existing molecule generation methods to enhance their novelty.

Our main contributions are summarized as follows:

- We newly propose to adopt OOD generation that exploits an OOD detection technique in the domain of deep drug discovery to tackle the well-known problem of poor novelty with respect to training molecules.
- We propose a simple yet effective MOG framework, in which the modified Langevin dynamics generates OOD molecules of high properties by exploring the overlaps of low likelihood regions predicted by an EBM and high property regions predicted by an auxiliary property network.
- We experimentally validate the effectiveness of proposed MOG through docking score optimization tasks, where exploration beyond the training set is crucial in performance. We further demonstrate our OOD generation strategy based on EBMs is universal, by augmenting the existing generation method with the EBM-based energy scoring and showing considerably improved novelty.

2 BACKGROUND AND RELATED WORK

2.1 MOLECULE GENERATION

Molecule generation models While early works adopt SMILES strings (Weininger, 1988) to represent molecules, more recent works utilize molecular graphs as inputs. Existing methods on generating molecular graphs of desired properties include models based on variational autoencoders (VAEs) (Gómez-Bombarelli et al., 2018; Jin et al., 2018; Liu et al., 2018), models based on generative adversarial networks (GANs) (Lima Guimaraes et al., 2017; De Cao & Kipf, 2018), flow-based models (Shi et al., 2019; Zang & Wang, 2020; Luo et al., 2021), and Markov chain Monte Carlo (MCMC) sampling in the chemical space (Xie et al., 2020). For VAE-based models, Bayesian optimization (BO) in the latent space is used to optimize chemical properties. For GAN- or flow-based models, reinforcement learning (RL) is commonly used for optimization, while direct application of RL (You et al., 2018) also has been explored. Recently, the employment of energy-based models (EBMs) has been proposed (Liu et al., 2021) and comprises a new branch in molecule generation.

In molecule generation models that require training data, the exploration in the chemical space beyond the known data distribution is limited as they focus on interpolating the learned distribution and generate in-distribution molecules. Models that do not use training data such as some RL- and MCMC sampling-based models circumvent this issue and thereby generally show better performance, but they suffer from other problems explained in Section 1. Different from these works, MOG exploits training molecules but generate molecules that do not follow the training distribution.

Docking score optimization Docking score optimization is underexplored in molecule generation models despite its importance in the real-world drug discovery, and only few RL-based models have dealt with docking scores. MORLD (Jeon & Kim, 2020), a generative model that utilizes MolDQN (Zhou et al., 2019), and REINVENT (Olivecrona et al., 2017), a SMILES-based model that utilizes REINFORCE algorithm, have been employed to conduct the docking score optimization task (Cieplinski et al., 2020; Thomas et al., 2021). FREED (Yang et al., 2021) utilizes prioritized experience replay (PER) for exploration to optimize docking scores.

2.2 ENERGY-BASED MODELS AND OOD GENERATION

EBMs as generative models The goal of EBMs is to learn a function $E_\theta(x) : \mathcal{X} \subset \mathbb{R}^d \rightarrow \mathbb{R}$ that has learnable parameters θ and maps an input $x \in \mathcal{X}$ to a deterministic scalar called the *energy*. The energy function defines a probability density through the Boltzmann distribution:

$$p_\theta(x) = \frac{e^{-E_\theta(x)}}{\int e^{-E_\theta(x)} dx} = \frac{e^{-E_\theta(x)}}{Z(\theta)}, \quad (1)$$

where the denominator $Z(\theta)$ is referred to as the partition function. To sample from this distribution that has the intractable partition function, Du & Mordatch (2019) propose to use Langevin dynamics (Welling & Teh, 2011). They also propose to weakly L2-regularize energy magnitudes to stabilize training. Liu et al. (2021) extend this framework to the graph domain and generate attributed graphs by applying Langevin dynamics to both node feature matrices and adjacency tensors.

EBMs in OOD detection EBMs are known to exhibit superior out-of-distribution (OOD) detection performance compared to other generative models (Du & Mordatch, 2019). Since the log-likelihood of a sample is proportional to its negative energy according to equation 1, one can discriminate OOD samples of low likelihood based on their energy values. Liu et al. (2020) utilize this property and propose to classify OOD samples by setting a threshold energy value.

OOD Generation Generation of OOD samples has been proposed by several works. Lee et al. (2018) propose to generate OOD samples by a GAN that is jointly trained with a confidence classifier. Vernekar et al. (2019) generate OOD samples using a conditional variational autoencoder (CVAE) to train an OOD classifier. Zheng et al. (2020) and Marek et al. (2021) utilize GAN-based models to generate OOD samples to detect OOD utterances for a robust dialog system. The goal of OOD generation in all of the mentioned works is to augment an OOD classifier for OOD detection, so their resulting OOD samples mimic in-distribution samples or are confined to the boundary of in-distribution samples.

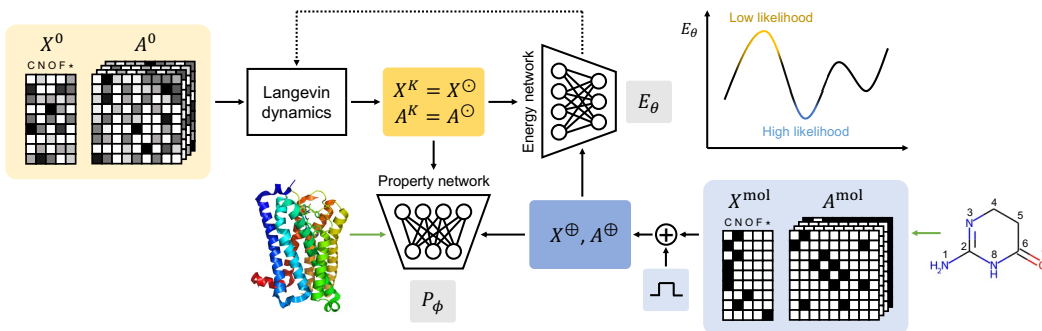


Figure 2: **The training process of MOG.** The energy network and the property network receive both positive samples and hallucinated samples. Molecular graphs of a dataset $(X^{\text{mol}}, A^{\text{mol}})$ are processed to positive samples (X^{\oplus}, A^{\oplus}) by the dequantization, while K steps of Langevin dynamics turn random noises (X^0, A^0) into hallucinated samples $(X^K, A^K) = (X^{\circ}, A^{\circ})$. In this figure, one molecule of QM9 dataset, in which $n = 9$, $a = 4$ (C, N, O, F), and $b = 3$ (single, double, and triple bonds), is used as an example. $*$ is the virtual atom type. The black arrows indicate the conveyance of information during the training process, while the green arrow indicates the pre-definition or pre-conversion before the process. The dashed arrow indicates the transfer of gradient.

3 PROPOSED METHOD

Our key insight is to combine the generation and the OOD detection of EBMs into a single framework to generate OOD molecules. We precisely target to generate OOD samples itself, not to augment an OOD classifier, unlike the aforementioned works on OOD generation. We begin with describing the graph representations of molecules and their notations.

Molecular graph representation A molecule can be represented as a molecular graph as $\mathcal{G} = (X, A)$, where X is the node feature matrix and A is the adjacency tensor. Let a and b denote the number of atom types and bond types, respectively. n denotes the maximum number of heavy atoms (atoms besides hydrogen) of a molecule in a given dataset. Following Madhawa et al. (2019), Zang & Wang (2020) and Liu et al. (2021), if a molecular graph has less than n nodes, we add virtual nodes to keep the dimensions of X and A the same for all molecules. Likewise, we add a virtual edge between any two nodes that are not connected in a molecular graph. Therefore, $X \in \{0, 1\}^{n \times (a+1)}$ and $A \in \{0, 1\}^{n \times n \times (b+1)}$ for all molecules, as we consider the virtual node and edge as an additional atom type and bond type, respectively.

3.1 THE ARCHITECTURE OF MOG

As shown in Figure 2, MOG is composed of two networks, namely, *energy network* and *property network*. The role of both networks is to provide gradients to the subsequent Langevin dynamics in the generation phase, as will be described in Section 3.3. The energy network provides information to control the expected likelihood of generated molecules with respect to given molecular distribution, and the property network enables the generation of molecules that have high target properties, such as docking scores.

Energy network We model the energy function for molecular graphs $E_{\theta}(X, A)$ by a graph neural network whose parameters are θ . Specifically, we adopt a GraphEBM (Liu et al., 2021) for the energy network. A relational graph convolutional network (R-GCN) (Schlichtkrull et al., 2018) is used to learn the node representations, and the l -th message passing is defined as follows:

$$H^{l+1} = \sum_{v=1}^{b+1} \sigma \left(D^{-\frac{1}{2}} A_{:,:,v} D^{-\frac{1}{2}} H^l W_v^l \right), \quad (2)$$

where $H^0 = X$, $A_{:,:,v}$ is the v -th channel of A , and D is a diagonal matrix that satisfies $D_{i,i} = \sum_{u=1}^n A_{i,u,v}$. $\{W_v^l\}_{v=1}^{b+1}$ is the trainable weight matrices of layer l , and $\sigma(\cdot)$ denotes an

activation function. After L layers of R-GCN, the graph-level representation h_G and the energy of the molecular graph are computed as follows:

$$h_G = \sum_{i=1}^n H_{i,:}^L \in \mathbb{R}^d, \quad (3)$$

$$E = h_G^\top W \in \mathbb{R}, \quad (4)$$

where $H^L \in \mathbb{R}^{n \times d}$ is the final node representation matrix, and $W \in \mathbb{R}^{d \times 1}$ is the trainable weight matrix.

Property network To generate optimized molecules through Langevin dynamics that makes use of gradients, we introduce a learnable and differentiable property function $P_\phi(X, A)$ that predicts the docking score of a molecule against a predefined protein target. The function is represented by a graph neural network whose parameters are ϕ . The architecture of the property network is chosen to be the same as the energy network, except for one extra MLP layer at the end.

3.2 LEARNING IN-DISTRIBUTION WITH AN ENERGY-BASED MODEL

Let p_D be the distribution of the real data. We can train the energy network in a maximum likelihood (ML) manner as follows:

$$\mathcal{L}_{ML} = \mathbb{E}_{(X,A) \sim p_D} [-\log p_\theta(X, A)], \quad (5)$$

where $-\log p_\theta(X, A) = E_\theta(X, A) + \log Z(\theta)$ according to equation 1. The objective is known to have the following gradient (Hinton, 2002; Turner, 2005):

$$\nabla_\theta \mathcal{L}_{ML} = \mathbb{E}_{(X^\oplus, A^\oplus) \sim p_D} [\nabla_\theta E_\theta(X^\oplus, A^\oplus)] - \mathbb{E}_{(X^\ominus, A^\ominus) \sim p_\theta} [\nabla_\theta E_\theta(X^\ominus, A^\ominus)], \quad (6)$$

where p_θ is the distribution defined by the energy function as in equation 1. (X^\ominus, A^\ominus) is referred to as a hallucinated sample (Du et al., 2020). This gradient decreases the energies of positive samples (X^\oplus, A^\oplus) while increases those of hallucinated samples (X^\ominus, A^\ominus) .

To obtain (X^\oplus, A^\oplus) from p_D , we conduct dequantization (Dinh et al., 2017; Kingma & Dhariwal, 2018) by adding random noises to make positive samples continuous:

$$\begin{aligned} X^\oplus &= X^{\text{mol}} + cu_X, & u_X &\sim \mathcal{U}[0, 1]^{n \times (a+1)}, \\ A^\oplus &= A^{\text{mol}} + cu_A, & u_A &\sim \mathcal{U}[0, 1]^{n \times n \times (b+1)}, \end{aligned} \quad (7)$$

where \mathcal{U} indicates an uniform distribution, $c \in [0, 1)$ is a scaling hyperparameter, and $(X^{\text{mol}}, A^{\text{mol}})$ is the initial discrete molecular graph.

To sample (X^\ominus, A^\ominus) from p_θ , we follow Liu et al. (2021) to use Langevin dynamics on both X and A . Specifically, we apply following operation iteratively:

$$\begin{aligned} X^k &= X^{k-1} - \lambda \nabla_X \mathcal{L}_{LD} + \omega^k, & \omega^k &\sim \mathcal{N}(0, \sigma^2)^{n \times (a+1)}, \\ A^k &= A^{k-1} - \lambda \nabla_A \mathcal{L}_{LD} + \eta^k, & \eta^k &\sim \mathcal{N}(0, \sigma^2)^{n \times n \times (b+1)}, \end{aligned} \quad (8)$$

where the loss of Langevin dynamics is defined as follows:

$$\mathcal{L}_{LD} = E_\theta(X^{k-1}, A^{k-1}). \quad (9)$$

k denotes the iteration step, and λ is the step size of Langevin dynamics. ω^k and η^k are noise matrices whose elements follow a Gaussian distribution \mathcal{N} of variance σ^2 . As $k \rightarrow \infty$ and $\lambda \rightarrow 0$, the resulting (X^k, A^k) become samples from p_θ and can be treated as (X^\ominus, A^\ominus) . In practice, we apply K steps of Langevin dynamics to randomly initialized (X^0, A^0) to obtain $(X^K, A^K) = (X^\ominus, A^\ominus)$, where $X^0 \sim \mathcal{U}[0, 1+c]^{n \times (a+1)}$ and $A^0 \sim \mathcal{U}[0, 1]^{n \times n \times (b+1)}$. X^k and A^k are clipped to the range $[0, 1+c]$ and $[0, 1]$, respectively, after each step.

The training procedure is illustrated in Figure 2. To make the gradient as depicted in equation 6, the loss function for the energy network is defined as follows:

$$\mathcal{L}_{\text{energy}} = E_\theta(X^\oplus, A^\oplus) - E_\theta(X^\ominus, A^\ominus) + \alpha \cdot \mathcal{L}_{\text{reg}}, \quad (10)$$

where $\mathcal{L}_{\text{reg}} = E_\theta(X^\oplus, A^\oplus)^2 + E_\theta(X^\ominus, A^\ominus)^2$ is for regularizing the magnitudes of energies (Du & Mordatch, 2019), and $\alpha \in \mathbb{R}$ is the weight of the regularization loss. On the other hand, the loss for the property network $\mathcal{L}_{\text{property}}$ is a mean squared error loss against the real properties values, which are provided by an external scoring system, QuickVina 2 (Alhossary et al., 2015). The property network is trained to predict docking scores of both positive samples and hallucinated samples.

3.3 OUT-OF-DISTRIBUTION GENERATION WITH MULTIPLE ENERGY PIVOTS

To generate OOD molecules, we propose to generate samples of high energies instead of low ones. We make use of validity correction introduced in Zang & Wang (2020) at the end of the generation, so every (X, A) is guaranteed to correspond to a valid molecule, eliminating the need to follow in-distribution to generate valid molecules. We propose to constrain energies to be high by designating a high target energy value, which we call the *energy pivot*, and use the mean squared error (MSE) against the energy pivot as a loss function in Langevin dynamics¹. This is equivalent to constraining likelihood to be close to a certain low value. Specifically, we utilize Langevin dynamics in equation 8 in generation phase, but different from equation 9 which generates (X, A) of low energies, we employ the following loss function:

$$\mathcal{L}_{LD} = \underbrace{\{E^* - E_{\theta}(X^{k-1}, A^{k-1})\}^2}_{\text{Constraint of energy pivot}} - \underbrace{P_{\phi}(X^{k-1}, A^{k-1})}_{\text{Property maximization}}, \quad (11)$$

where the first term confines the energy values to be close to the energy pivot E^* while the second term encourages maximizing the chemical property predicted by the property network.

To validate the use of the energy pivot actually generates OOD samples, we experiment with a 2D dataset that consists of eight Gaussian blobs. We only use the first term in equation 11. The details of implementation are included in Section B.1. As shown in Figure 3, our proposed Langevin dynamics can generate OOD samples distinguished from the training samples indeed. In addition, we can observe as the value of the energy pivot increases, the degree of deviation from the in-distribution increases in a continuous fashion. Therefore, in our MOG, we set multiple energy pivots and optimize each randomly initialized molecule with a different pivot to cover various degrees of OOD. The use of multiple energy pivots also obviates the need for intensive tuning.

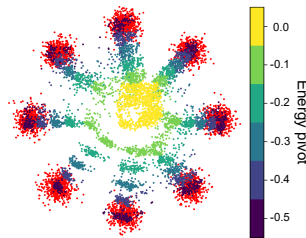


Figure 3: **OOD generation on a 2D example.** The training distribution consists of eight isotropic Gaussian blobs, and the samples are indicated by red. The generated OOD samples with a certain energy pivot are indicated by the corresponding color.

4 EXPERIMENTS

Baselines We compare MOG against the following molecule generation baselines: **REINVENT** (Olivecrona et al., 2017) is a SMILES strings-based RL model; **JT-VAE** (Jin et al., 2018) is a VAE-based model that optimizes molecules over the latent space using the gradient from a property predictor; **GCPN** (You et al., 2018) is a RL model which generates molecules atom-by-atom; **GraphNVP** (Madhawa et al., 2019) is a flow model that generates molecules in an one-shot manner; **GraphAF** (Shi et al., 2019) is a RL-finetuned flow model that generates molecules autoregressively; **MoFlow** (Zang & Wang, 2020) is an one-shot flow model with a validity guarantee; **MORLD** (Jeon & Kim, 2020) is a RL model that is based on MoldQN; **GraphDF** (Luo et al., 2021) is an autoregressive RL-finetuned flow model that utilizes discrete latent variables; **GraphEBM** (Liu et al., 2021) is an EBM that generates in-distribution molecules by minimizing energy values; **FREED** (Yang et al., 2021) is a fragment-based RL model that utilizes PER for exploration.

4.1 RANDOM GENERATION

Evaluation metrics We evaluate the generated molecules with following metrics: **Validity** is the percentage of molecules that do not violate chemical valency; **Uniqueness** is the percentage of valid molecules that are unique; **Novelty** is the percentage of valid molecules that are not included in the in-distribution molecules; **Novelty (sim. < x)** is the stronger novelty metric, which indicates the percentage of valid molecules with similarity less than x compared to the nearest neighbor \mathcal{G}_{SNN} in the in-distribution molecules: $\frac{1}{n} \sum_{\mathcal{G} \in \mathcal{V}} \mathbf{1}[\text{sim}(\mathcal{G}, \mathcal{G}_{SNN}) < x]$, where \mathcal{V} is the set of n valid molecules, and $\text{sim}(\mathcal{G}, \mathcal{G}')$ is the pairwise Tanimoto similarity over Morgan fingerprints. Previous works (Olivecrona et al., 2017; Jin et al., 2020; Xie et al., 2020) have set the threshold as $x = 0.4$.

¹We also explored maximizing energies without introducing the energy pivot. The resulting Langevin dynamics with $\mathcal{L}_{LD} = -E_{\theta}(X^{k-1}, A^{k-1})$ fails to generate diverse molecules and converges to a single molecule.

However, since the molecule in Figure 1(a) has Tanimoto similarities of 0.3194 and 0.2932 with the left and right molecules in Figure 1(b), respectively, and yet the similarities are “striking” by chemists’ eyes, 0.4 is clearly too generous to determine whether generated molecules are novel enough. We set $x = 0.3$ and $x = 0.2$ for the QM9 and ZINC250k datasets, respectively.

Experimental setup We first evaluate MOG on the random generation setting, following previous works (Madhawa et al., 2019; Popova et al., 2019; Shi et al., 2019; Zang & Wang, 2020; Luo et al., 2021; Liu et al., 2021). The property term in equation 11 is ignored for random generation. We train MOG in two molecular datasets, QM9 (Ramakrishnan et al., 2014) and ZINC250k (Irwin et al., 2012), and generate 3000 molecules. Implementation details are included in Section B.2.

Table 1: **Random generation results on the QM9 dataset.** Results of baselines are obtained by running open-source codes. We report the mean and the standard deviation of 3 different runs.

Method	Validity (%)	Uniqueness (%)	Novelty (%)	Novelty (sim. < 0.3) (%)
GraphNVP (Madhawa et al., 2019)	82.83±0.43	99.23±0.26	69.80±0.44	9.85±0.36
MoFlow (Zang & Wang, 2020)	100.00±0.00	99.72±0.04	98.68±0.13	58.17±0.26
GraphDF (Luo et al., 2021)	100.00±0.00	99.20±0.18	97.96±0.12	44.83±0.62
GraphEBM (Liu et al., 2021)	100.00±0.00	99.43±0.38	98.11±0.40	54.38±0.52
MOG (Ours)	100.00±0.00	99.41±0.12	99.34±0.23	62.51±1.09

Table 2: **Random generation results on the ZINC250k dataset.** Result with † is taken from Popova et al. (2019). Other results of baselines are obtained by running open-source codes. We report the mean and the standard deviation of 3 different runs.

Method	Validity (%)	Uniqueness (%)	Novelty (%)	Novelty (sim. < 0.2) (%)
JT-VAE (Jin et al., 2018)	100.00±0.00	100.00±0.00 †	99.79±0.09	0.94±0.06
GCPN (You et al., 2018)	100.00±0.00	99.93±0.03	100.00±0.00	81.66±0.59
GraphAF (Shi et al., 2019)	100.00±0.00	99.51±0.11	99.98±0.03	30.57±0.68
MoFlow (Zang & Wang, 2020)	100.00±0.00	99.99±0.01	100.00±0.00	55.39±0.14
GraphDF (Luo et al., 2021)	100.00±0.00	99.82±0.07	100.00±0.00	46.23±0.22
GraphEBM (Liu et al., 2021)	100.00±0.00	99.09±0.04	100.00±0.00	62.61±0.20
FREED (Yang et al., 2021)	100.00±0.00	100.00±0.00	100.00±0.00	1.31±0.17
MOG (Ours)	100.00±0.00	99.99±0.01	100.00±0.00	94.49±0.21

Results The random generation results are shown in Table 1 and Table 2. MOG outperforms other baselines in terms of novelty, especially when the similarity threshold is set to 0.2, while it also achieves high validity and uniqueness. Since GraphEBM generates molecules with the same architecture but different Langevin dynamics, the improvement in novelty compared to GraphEBM can be concluded as the effect of multiple energy pivots. The overall results support that increasing energy instead of minimizing it is advantageous to generate different molecules from training molecules. It is also notable that even though FREED does not use training molecules, it shows poor novelty on the ZINC250k dataset because its fragment vocabulary is extracted from the dataset. On the other hand, the second-best model on ZINC250k, GCPN, does not use training molecules nor a fragment vocabulary, hence explains its fairly high novelty. The superiority of MOG over GCPN demonstrates the effectiveness of OOD generation using the knowledge of known molecular distribution.

4.2 DOCKING SCORE OPTIMIZATION

Evaluation metrics In addition to validity, uniqueness, novelty, and novelty (sim. < 0.2) explained in Section 4.1, we report the top-3 scores of generated molecules found in 3 runs, following previous works (Jin et al., 2018; You et al., 2018; Popova et al., 2019; Shi et al., 2019; Zang & Wang, 2020; Luo et al., 2021; Liu et al., 2021).

Experimental setup We validate the ability of MOG to generate novel molecules that possess high absolute values of docking score. The target proteins about which the docking scores are calculated are parp1, fa7, and 5ht1b, following Yang et al. (2021). Lower docking scores indicate higher affinity against a given target protein, which implies higher drug activity. We set $P_\phi(X, A)$ to predict negative docking scores for each protein. We train MOG in the ZINC250k (Irwin et al., 2012) dataset and generate 3000 molecules. Implementation details are included in Section B.2.

Table 3: Docking score optimization results with parp1 target protein. Results of baselines are obtained by running open-source codes. We report the mean and the standard deviation of 3 different runs.

Method	parp1 docking score			Validity (%)	Uniqueness (%)	Novelty (%)	Novelty (sim. < 0.2) (%)
	1st	2nd	3rd				
REINVENT (Olivecrona et al., 2017)	-13.0	-12.4	-12.4	91.74±1.09	99.71±0.27	99.82±0.06	0.00±0.00
MORLD (Jeon & Kim, 2020)	-13.3	-12.8	-12.7	100.00±0.00	99.08±0.43	100.00±0.00	89.46±1.04
GraphDF (Luo et al., 2021)	-19.6	-19.2	-19.1	100.00±0.00	100.00±0.00	99.96±0.01	62.54±0.42
GraphEBM (Liu et al., 2021)	-18.9	-18.4	-17.5	100.00±0.00	99.89±0.04	100.00±0.00	98.59±0.27
FREED (Yang et al., 2021)	-15.4	-15.4	-14.8	100.00±0.00	98.07±1.18	100.00±0.00	0.83±0.28
MOG (Ours)	-24.6	-23.8	-23.4	100.00±0.00	100.00±0.00	100.00±0.00	99.90±0.01

Table 4: Docking score optimization results with fa7 target protein. Results of baselines are obtained by running open-source codes. We report the mean and the standard deviation of 3 different runs.

Method	fa7 docking score			Validity (%)	Uniqueness (%)	Novelty (%)	Novelty (sim. < 0.2) (%)
	1st	2nd	3rd				
REINVENT (Olivecrona et al., 2017)	-11.0	-10.6	-10.1	92.11±0.85	99.85±0.03	99.94±0.03	0.05±0.05
MORLD (Jeon & Kim, 2020)	-10.7	-10.6	-10.6	100.00±0.00	99.22±1.03	100.00±0.00	88.51±5.75
GraphDF (Luo et al., 2021)	-20.0	-19.4	-19.2	100.00±0.00	100.00±0.00	100.00±0.00	69.21±0.19
GraphEBM (Liu et al., 2021)	-20.1	-20.0	-19.8	100.00±0.00	99.83±0.07	100.00±0.00	98.74±0.19
FREED (Yang et al., 2021)	-12.5	-11.9	-11.8	100.00±0.00	97.47±1.74	100.00±0.00	0.78±0.21
MOG (Ours)	-21.1	-20.4	-19.9	100.00±0.00	99.89±0.04	100.00±0.00	99.64±0.10

Table 5: Docking score optimization results with 5ht1b target protein. Results of baselines are obtained by running open-source codes. We report the mean and the standard deviation of 3 different runs.

Method	5ht1b docking score			Validity (%)	Uniqueness (%)	Novelty (%)	Novelty (sim. < 0.2) (%)
	1st	2nd	3rd				
REINVENT (Olivecrona et al., 2017)	-13.2	-13.1	-13.0	92.97±0.75	99.62±0.16	99.81±0.04	0.01±0.02
MORLD (Jeon & Kim, 2020)	-12.8	-12.3	-12.2	100.00±0.00	99.84±0.08	100.00±0.00	86.58±2.70
GraphDF (Luo et al., 2021)	-21.4	-20.7	-20.5	100.00±0.00	100.00±0.00	99.97±0.00	60.61±0.35
GraphEBM (Liu et al., 2021)	-22.3	-20.2	-19.3	100.00±0.00	99.82±0.07	100.00±0.00	98.65±0.35
FREED (Yang et al., 2021)	-15.1	-15.1	-15.1	100.00±0.00	97.00±3.20	100.00±0.00	0.75±0.20
MOG (Ours)	-28.0	-27.3	-26.8	100.00±0.00	99.91±0.06	100.00±0.00	99.66±0.24

Results The docking score optimization results are reported in Table 3, Table 4, and Table 5. We observe that MOG can generate molecules that have significantly higher absolute docking scores compared to the baselines, while achieving near-perfect validity, uniqueness, and novelty. GraphEBM shows secondly high novelty, but it fails to escape from local optima and generates molecules of sub-optimal docking scores compared to MOG. Along with Table 7 in Section A.1, these results strongly demonstrate the Langevin dynamics modified by energy pivots in conjunction with the property network greatly promotes to search for the better optimum, resulting in the high absolute docking scores.

4.3 IMPROVEMENT IN MARS

Original setup of MARS To show the proposed OOD generation strategy using an EBM can be universally adopted to existing molecule generation models to encourage the exploration beyond known molecules, we augment MARS (Xie et al., 2020) with energy scoring. MARS is a model that samples molecules in a Markov chain defined on the chemical space. A molecule editing graph neural network proposes a molecule \mathcal{G}' at each step, and the proposed molecule is either rejected or accepted according to the acceptance rate:

$$\mathcal{A}(\mathcal{G}, \mathcal{G}') = \min \left\{ 1, \frac{\pi^{1/T}(\mathcal{G}')}{\pi^{1/T}(\mathcal{G})} \right\}, \quad (12)$$

where \mathcal{G} is the current molecule, $\pi(\mathcal{G})$ is the score of the molecule, and T is the temperature controlled by a cooling schedule. The scoring function in the original MARS is defined as follows:

$$\pi(\mathcal{G}) = \text{GSK3}\beta(\mathcal{G}) + \text{JNK3}(\mathcal{G}) + \text{QED}(\mathcal{G}) + \text{SA}(\mathcal{G}), \quad (13)$$

which is the summation of the inhibition scores against glycogen synthase kinase-3 β (GSK3 β) and c-Jun N-terminal kinase-3 (JNK3), the drug-likeness (QED), and the normalized synthetic accessibility (SA) score. Implementation details are included in Section B.3.

Evaluation metrics The goal of MARS in GSK3 β +JNK3+QED+SA setting is to generate molecules that jointly inhibit GSK3 β and JNK3 while being drug-like and synthetically accessible. MARS generates 5000 molecules and evaluate following metrics: **Success rate** is the fraction of generated molecules that are positive on all objectives (QED(\mathcal{G}) \geq 0.6, SA(\mathcal{G}) \geq 0.67, GSK3 β (\mathcal{G}) \geq 0.5, JNK3(\mathcal{G}) \geq 0.5); **Novelty (sim. < 0.4)** is the percentage of generated molecules with similarity less than 0.4 compared to the nearest neighbor in the training set as explained in Section 4.1; **Diversity** is the diversity of generated molecules: $\frac{2}{n(n-1)} \sum_{\mathcal{G} \neq \mathcal{G}'} 1 - \text{sim}(\mathcal{G}, \mathcal{G}')$, where $\text{sim}(\mathcal{G}, \mathcal{G}')$ is the pairwise Tanimoto similarity over Morgan fingerprints; **Product of metrics** is the product of the above three metrics, which is the comprehensive and mainly emphasized metric.

Employment of energy objective in MARS We propose to further include energy scores of molecules from an EBM trained on the same training set used in MARS. The training set consists of known active molecules against GSK3 β and JNK3, and maximizing the energy score encourages to generate OOD molecules against these molecules. Specifically, we modify equation 13 as follows: $\pi_{\text{energy}}(\mathcal{G}) = \pi(\mathcal{G}) + \nu \cdot E_{\theta}(\mathcal{G})$, where $\pi(\mathcal{G})$ is the original scoring function, $E_{\theta}(\mathcal{G})$ is the energy score calculated by the EBM, and ν is the weight of the energy score. We then utilize $\pi_{\text{energy}}(\mathcal{G})$ in equation 12. Implementation details are included in Section B.3.

Table 6: **GSK3 β +JNK3+QED+SA optimization results.** MARS+OOD indicates MARS augmented with the EBM energy score. Results of MARS are obtained by running its open-source code. Results of other baselines are taken from Xie et al. (2020). We report the mean and the standard deviation of 5 different runs of 5000 generated molecules. Because of the high variance of MARS that comes from MCMC sampling, we mark both the best results and their comparable results ($p > 0.05$ from the t-test) as bold.

Method	Success rate (%)	Novelty (sim. < 0.4) (%)	Diversity	Product of metrics
GCPN (You et al., 2018)	0.0	0.0	0.000	0.00
JT-VAE (Jin et al., 2018)	5.4	100.0	0.277	0.02
RationaleRL (Jin et al., 2020)	75.0	55.5	0.706	0.29
GA+D (Nigam et al., 2019)	85.7	100.0	0.363	0.31
MARS (Xie et al., 2020)	91.4 \pm 3.1	82.2 \pm 3.8	0.734 \pm 0.014	0.55 \pm 0.03
MARS+OOD (Ours)	90.7 \pm 1.1	90.6 \pm 2.1	0.734 \pm 0.013	0.60 \pm 0.02

Results As shown in Table 6, the MARS model augmented with the energy score achieves significantly improved novelty, while showing comparably high success rate and diversity. This is because while the original MARS lacks any constraints to make the generated molecules differ from known active molecules, the modified MARS takes the energy score into account to generate OOD molecules. Our modified MARS generates molecules that do not share any big substructures with the known active molecules indeed, unlike the original MARS (see Figure 4 in Section A.2). We emphasize that this strategy to treat energy values calculated by an EBM as scores or rewards can be easily incorporated into most existing models.

5 CONCLUSION

In this work, we propose to utilize OOD generation in deep drug discovery to resolve the problem of poor exploration in existing molecule generation methods. Specifically, we propose MOG, a novel framework to generate OOD molecules of high absolute docking scores with the modified Langevin dynamics that makes use of both an EBM and a property predictor. The dynamics increases the energy of molecules with the multiple energy pivots and thereby explores low likelihood regions. We experimentally show that MOG can generate novel molecules and find better optima of docking scores. We also demonstrate our OOD generation strategy based on EBMs can greatly aid existing molecule generation methods to improve novelty. We believe that our work opened a door to more interesting follow-up works on *de novo* drug discovery of deep molecule generation models, which focus on generating *de novo* molecules in a true sense.

REFERENCES

- Amr Alhossary, Stephanus Daniel Handoko, Yuguang Mu, and Chee-Keong Kwoh. Fast, accurate, and reliable molecular docking with quickvina 2. *Bioinformatics*, 31(13):2214–2216, 2015.
- G Richard Bickerton, Gaia V Paolini, Jérémy Besnard, Sorel Muresan, and Andrew L Hopkins. Quantifying the chemical beauty of drugs. *Nature chemistry*, 4(2):90–98, 2012.
- Tobiasz Cieplinski, Tomasz Danel, Sabina Podlowska, and Stanislaw Jastrzebski. We should at least be able to design molecules that dock well. *arXiv preprint arXiv:2006.16955*, 2020.
- Connor W Coley. Defining and exploring chemical spaces. *Trends in Chemistry*, 2020.
- Nicola De Cao and Thomas Kipf. Molgan: An implicit generative model for small molecular graphs. *ICML 2018 workshop on Theoretical Foundations and Applications of Deep Generative Models*, 2018.
- Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real nvp. *International Conference on Learning Representations*, 2017.
- Yilun Du and Igor Mordatch. Implicit generation and modeling with energy based models. *Advances in Neural Information Processing Systems*, 32:3608–3618, 2019.
- Yilun Du, Shuang Li, Joshua Tenenbaum, and Igor Mordatch. Improved contrastive divergence training of energy based models. *arXiv preprint arXiv:2012.01316*, 2020.
- Leonardo G Ferreira, Ricardo N Dos Santos, Glaucius Oliva, and Adriano D Andricopulo. Molecular docking and structure-based drug design strategies. *Molecules*, 20(7):13384–13421, 2015.
- Anna Gaulton, Anne Hersey, Michał Nowotka, A Patricia Bento, Jon Chambers, David Mendez, Prudence Mutowo, Francis Atkinson, Louisa J Bellis, Elena Cibrián-Uhalte, et al. The chembl database in 2017. *Nucleic acids research*, 45(D1):D945–D954, 2017.
- Rafael Gómez-Bombarelli, Jennifer N Wei, David Duvenaud, José Miguel Hernández-Lobato, Benjamín Sánchez-Lengeling, Dennis Sheberla, Jorge Aguilera-Iparraguirre, Timothy D Hirzel, Ryan P Adams, and Alán Aspuru-Guzik. Automatic chemical design using a data-driven continuous representation of molecules. *ACS central science*, 4(2):268–276, 2018.
- Geoffrey E Hinton. Training products of experts by minimizing contrastive divergence. *Neural computation*, 14(8):1771–1800, 2002.
- John J Irwin, Teague Sterling, Michael M Mysinger, Erin S Bolstad, and Ryan G Coleman. Zinc: a free tool to discover chemistry for biology. *Journal of chemical information and modeling*, 52(7):1757–1768, 2012.
- Woosung Jeon and Dongsup Kim. Autonomous molecule generation using reinforcement learning and docking to develop potential novel inhibitors. *Scientific reports*, 10(1):1–11, 2020.
- Wengong Jin, Regina Barzilay, and Tommi Jaakkola. Junction tree variational autoencoder for molecular graph generation. In *International conference on machine learning*, pp. 2323–2332. PMLR, 2018.
- Wengong Jin, Regina Barzilay, and Tommi Jaakkola. Multi-objective molecule generation using interpretable substructures. In *International Conference on Machine Learning*, pp. 4849–4859. PMLR, 2020.
- Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- Diederik P Kingma and Prafulla Dhariwal. Glow: generative flow with invertible 1×1 convolutions. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pp. 10236–10245, 2018.
- Matt J Kusner, Brooks Paige, and José Miguel Hernández-Lobato. Grammar variational autoencoder. In *International Conference on Machine Learning*, pp. 1945–1954. PMLR, 2017.

- Greg Landrum et al. Rdkit: Open-source cheminformatics software, 2016. URL <http://www.rdkit.org/>, <https://github.com/rdkit/rdkit>, 2016.
- Kimin Lee, Honglak Lee, Kibok Lee, and Jinwoo Shin. Training confidence-calibrated classifiers for detecting out-of-distribution samples. In *International Conference on Learning Representations*, 2018.
- Gabriel Lima Guimaraes, Benjamin Sanchez-Lengeling, Carlos Outeiral, Pedro Luis Cunha Farias, and Alán Aspuru-Guzik. Objective-reinforced generative adversarial networks (organ) for sequence generation models. *arXiv*, pp. arXiv-1705, 2017.
- Meng Liu, Keqiang Yan, Bora Oztekin, and Shuiwang Ji. Graphebm: Molecular graph generation with energy-based models. In *Energy Based Models Workshop-ICLR 2021*, 2021.
- Qi Liu, Miltiadis Allamanis, Marc Brockschmidt, and Alexander L Gaunt. Constrained graph variational autoencoders for molecule design. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pp. 7806–7815, 2018.
- Weitang Liu, Xiaoyun Wang, John Owens, and Yixuan Li. Energy-based out-of-distribution detection. *Advances in Neural Information Processing Systems*, 33, 2020.
- Youzhi Luo, Keqiang Yan, and Shuiwang Ji. Graphdf: A discrete flow model for molecular graph generation. *International conference on machine learning*, 2021.
- Kaushalya Madhawa, Katushiko Ishiguro, Kosuke Nakago, and Motoki Abe. Graphnvp: An invertible flow model for generating molecular graphs. *arXiv preprint arXiv:1905.11600*, 2019.
- Petr Marek, Vishal Ishwar Naik, Anuj Goyal, and Vincent Auvray. Oodgan: Generative adversarial network for out-of-domain data generation. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Industry Papers*, pp. 238–245, 2021.
- Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. In *International Conference on Learning Representations*, 2018.
- AkshatKumar Nigam, Pascal Friederich, Mario Krenn, and Alan Aspuru-Guzik. Augmenting genetic algorithms with deep neural networks for exploring the chemical space. In *International Conference on Learning Representations*, 2019.
- Marcus Olivecrona, Thomas Blaschke, Ola Engkvist, and Hongming Chen. Molecular de-novo design through deep reinforcement learning. *Journal of cheminformatics*, 9(1):1–14, 2017.
- Mariya Popova, Mykhailo Shvets, Junier Oliva, and Olexandr Isayev. Molecularrrnn: Generating realistic molecular graphs with optimized properties. *arXiv preprint arXiv:1905.13372*, 2019.
- Ragunathan Ramakrishnan, Pavlo O Dral, Matthias Rupp, and O Anatole Von Lilienfeld. Quantum chemistry structures and properties of 134 kilo molecules. *Scientific data*, 1(1):1–7, 2014.
- Michael Schlichtkrull, Thomas N Kipf, Peter Bloem, Rianne Van Den Berg, Ivan Titov, and Max Welling. Modeling relational data with graph convolutional networks. In *European semantic web conference*, pp. 593–607. Springer, 2018.
- Chence Shi, Minkai Xu, Zhaocheng Zhu, Weinan Zhang, Ming Zhang, and Jian Tang. Graphaf: a flow-based autoregressive model for molecular graph generation. In *International Conference on Learning Representations*, 2019.
- Morgan Thomas, Robert T Smith, Noel M O’Boyle, Chris de Graaf, and Andreas Bender. Comparison of structure-and ligand-based scoring functions for deep generative models: a gpcr case study. *Journal of Cheminformatics*, 13(1):1–20, 2021.
- Richard Turner. Cd notes. 2005.

- Sachin Vernekar, Ashish Gaurav, Vahdat Abdelzad, Taylor Denouden, Rick Salay, and Krzysztof Czarnecki. Out-of-distribution detection in classifiers via generation. *arXiv preprint arXiv:1910.04241*, 2019.
- W Patrick Walters and Mark Murcko. Assessing the impact of generative ai on medicinal chemistry. *Nature Biotechnology*, 38(2):143–145, 2020.
- David Weininger. Smiles, a chemical language and information system. 1. introduction to methodology and encoding rules. *Journal of chemical information and computer sciences*, 28(1):31–36, 1988.
- Max Welling and Yee W Teh. Bayesian learning via stochastic gradient langevin dynamics. In *Proceedings of the 28th international conference on machine learning (ICML-11)*, pp. 681–688. Citeseer, 2011.
- Yutong Xie, Chence Shi, Hao Zhou, Yuwei Yang, Weinan Zhang, Yong Yu, and Lei Li. Mars: Markov molecular sampling for multi-objective drug discovery. In *International Conference on Learning Representations*, 2020.
- Soojung Yang, Doyeong Hwang, Seul Lee, Seongok Ryu, and Sung Ju Hwang. Hit and lead discovery with explorative rl and fragment-based molecule generation. *arXiv preprint arXiv:2110.01219*, 2021.
- Jiaxuan You, Bowen Liu, Rex Ying, Vijay Pande, and Jure Leskovec. Graph convolutional policy network for goal-directed molecular graph generation. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pp. 6412–6422, 2018.
- Chengxi Zang and Fei Wang. Moflow: an invertible flow model for generating molecular graphs. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 617–626, 2020.
- Alex Zhavoronkov, Yan A Ivanenkov, Alex Aliper, Mark S Veselov, Vladimir A Aladinskiy, Anastasiya V Aladinskaya, Victor A Terentiev, Daniil A Polykovskiy, Maksim D Kuznetsov, Arip Asadulaev, et al. Deep learning enables rapid identification of potent ddr1 kinase inhibitors. *Nature biotechnology*, 37(9):1038–1040, 2019.
- Yinhe Zheng, Guanyi Chen, and Minlie Huang. Out-of-domain detection for natural language understanding in dialog systems. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 28:1198–1209, 2020.
- Zhenpeng Zhou, Steven Kearnes, Li Li, Richard N Zare, and Patrick Riley. Optimization of molecules via deep reinforcement learning. *Scientific reports*, 9(1):1–10, 2019.

A ADDITIONAL EXPERIMENTAL RESULTS

A.1 EFFECTS OF ENERGY PIVOTS AND PROPERTY NETWORK

To examine the effect of the energy pivots and the property network, we conduct the docking score optimization task with respect to parp1 without them. We use following loss functions instead of equation 11 in Langevin dynamics of generation phase:

$$\mathcal{L}_{LD,XX} = E_{\theta}(X^{k-1}, A^{k-1}), \quad (14)$$

$$\mathcal{L}_{LD,XO} = E_{\theta}(X^{k-1}, A^{k-1}) - P_{\phi}(X^{k-1}, A^{k-1}), \quad (15)$$

$$\mathcal{L}_{LD,OX} = \{E^* - E_{\theta}(X^{k-1}, A^{k-1})\}^2, \quad (16)$$

where the first and second subscripts of \mathcal{L}_{LD} indicate the usage of the multiple energy pivots and the property network, respectively. $\mathcal{L}_{LD,OO}$ is equal to equation 11.

Table 7: **parp1 docking score optimization results without energy pivots and the property network.** We report the mean and the standard deviation of 3 different runs of 3000 generated molecules.

Energy pivot	Property network	parp1 docking score			Validity (%)	Uniqueness (%)	Novelty (%)	Novelty (sim. < 0.2) (%)
		1st	2nd	3rd				
X	X	-11.6	-10.4	-10.1	100.00 ±0.00	99.09±0.04	100.00 ±0.00	62.61±0.20
X	O	-3.6	-	-	100.00 ±0.00	0.03±0.00	100.00 ±0.00	100.00 ±0.00
O	X	-15.5	-15.4	-14.9	100.00 ±0.00	97.87±0.26	100.00 ±0.00	97.11±0.09
O	O	-24.6	-23.8	-23.4	100.00 ±0.00	100.00 ±0.00	100.00 ±0.00	99.90 ±0.01

As shown in Table 7, using both the energy pivots and the property network is essential to generate novel molecules and find better optima of docking scores. Comparing the first and the third rows, the use of multiple energy pivots greatly improves novelty. The energy pivots also improve the top-3 docking scores even without the property network, implying more extensive exploration. Note that the Langevin dynamics using only the property network without the energy pivots converges to a single molecule, so the novelty value is meaningless.

A.2 VISUALIZATION OF MOLECULES GENERATED BY MARS AND MARS+OOD

Among the generated molecules of original MARS and MARS with the additional EBM energy score (MARS+OOD), respectively, we display the molecules of maximum Tanimoto similarity with certain training molecules in Figure 4.

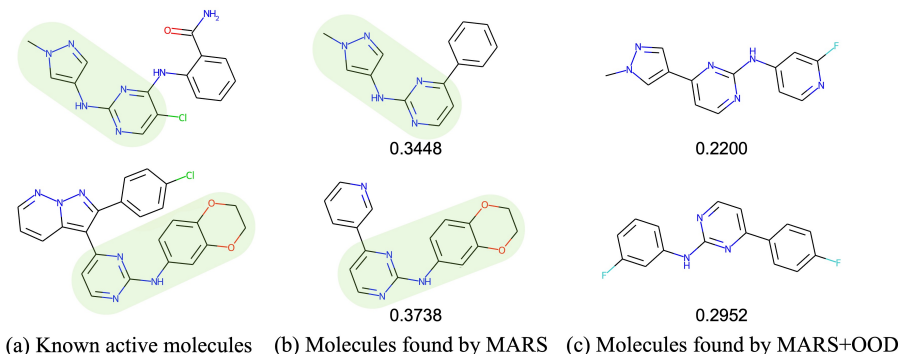


Figure 4: **Molecules in training set and molecules of maximum Tanimoto similarity generated by MARS and MARS+OOD.** MARS+OOD indicates MARS augmented with the EBM energy score. For each row, we display (a) a known active molecule against GSK3 β and JNK3 included in the training set, (b) a molecule that has maximum Tanimoto similarity with (a) among the generated molecules by MARS, and (c) a molecule that has maximum Tanimoto similarity with (a) among the generated molecules by MARS+OOD. The number below each molecule in (b) and (c) is the corresponding Tanimoto similarity compared to (a). The duplicated structures in (a) and (b) are indicated by green areas.

As shown in Figure 4, the maximally similar molecules of MARS possess a duplicated substructure with the corresponding training molecules, as depicted by the green areas. On the contrary, MARS+OOD generates molecules that do not share any big substructures with the training molecules, even in the maximally similar molecule. The maximum Tanimoto similarity values are lower in MARS+OOD accordingly.

B IMPLEMENTATION DETAILS

B.1 DETAILS OF THE 2D OOD GENERATION

For the OOD generation experiment on a 2D dataset, we set the training distribution as eight isotropic Gaussian blobs that have the standard deviation of 0.3. The EBM used consists of 3 MLP layers with hidden dimension of $d = 128$, and the activation function is LeakyReLU with a slope of 0.2. Spectral normalization (Miyato et al., 2018) is applied after each layer, and gradient clipping of $[-0.05, 0.05]$ is used in each step of Langevin dynamics. The standard deviation of the noise in Langevin dynamics and the weight of the regularization loss are set to $\sigma = 0.005$ and $\alpha = 1$, respectively. The step size in Langevin dynamics of $\lambda = 5$ and the number of steps in Langevin dynamics of $K = 100$ are used. The model is trained 10000 iterations with a batch size of 2048 and an Adam optimizer (Kingma & Ba, 2014).

B.2 DETAILS OF RANDOM GENERATION AND DOCKING SCORE OPTIMIZATION

Datasets and Metric The information of molecular datasets is summarized in Table 8. $b = 3$ (see Section 3 for the notation of molecular graphs) for single, double, and triple bonds in both datasets. All molecules are preprocessed to their kekulized form and all hydrogens are removed by RDKit (Landrum et al., 2016). For ZINC250k, we use the same train/validation split used by Kusner et al. (2017). The novelty is calculated with respect to a whole dataset (QM9 or ZINC250k), which includes both train and validation sets, thereby counts novel molecules not only against direct training molecules but also against the molecules that lie in the in-distribution.

Table 8: **Information of the QM9 and ZINC250k datasets.** The maximum number of nodes and the number of node types correspond to n and a of the notation in Section 3, respectively.

Dataset	Number of molecules	Maximum number of nodes	Number of node types
QM9	133885	9	4
ZINC250k	249455	38	9

Implementation details of MOG To generate N molecules with equation 11, N multiple energy pivots are set to linear internal division points in $[-0.5, 0]$ and $[5, 10]$ for the random generation and the docking score optimization experiments, respectively. Throughout the experiments, we set the exhaustiveness of QuickVina 2 as 1 to calculate docking scores. The activation functions are Swish and ReLU for the energy and property networks, respectively. We use $L = 3$ layers of R-GCN with hidden dimension of $d = 64$. Spectral normalization (Miyato et al., 2018) is used after each layer, and gradient clipping of $[-0.01, 0.01]$ is applied to each step of Langevin dynamics in both training and generation phases. We set the standard deviation of the noise in Langevin dynamics as $\sigma = 0.005$ and the weight of the regularization loss as $\alpha = 1$. For the QM9 dataset, the dequantization scale of $c = 0.2$, the step size in Langevin dynamics of $\lambda = 10$, and the number of steps in Langevin dynamics of $K = 30$ are used. For the ZINC250k dataset, the dequantization scale of $c = 0$, the step size in Langevin dynamics of $\lambda = 30$, and the number of steps in Langevin dynamics of $K = 150$ are used. For the energy network, a learning rate of 1×10^{-4} and the number of training epochs of 20 are used. For the property network, an initial learning rate of 1×10^{-3} with weight decay of 1×10^{-4} and the number of training iterations of 200 are used. For both networks, a batch size of 128 and an Adam optimizer (Kingma & Ba, 2014) are used. Most of the hyperparameters regarding the energy network are set following Liu et al. (2021).

Implementation details of baselines We follow the corresponding original papers for most of the settings of the baselines. We describe the specifics and the differences from the original papers here. For REINVENT, we utilize the ZINC250k dataset to construct the vocabulary and train the prior. For GraphNVP, we use the pretrained GraphNVP provided by the authors. For MORLD, we set negative values of docking scores from QuickVina 2 as the final reward function and benzene as the initial molecule. For GraphDF, we use the same hyperparameters the authors used to optimize against penalized logP or QED score. Same as in the optimization of penalized logP, the intermediate reward is set to -1 if the step violates chemical bond valency, otherwise, we assign 0. We use the same final reward function the authors used for penalized logP optimization, for docking score optimization. For GraphEBM, the same goal-directed generation strategy the authors used is utilized to optimize docking scores. For FREED, we utilize the predictive error-PER.

B.3 DETAILS OF IMPROVEMENT IN MARS

Implementation details of MARS We strictly follow the setting in the original paper of MARS. The fragment vocabulary consists of the top 1000 frequently appearing fragments that contain no more than 10 heavy atoms from the ChEMBL dataset (Gaulton et al., 2017) by breaking single bonds. The temperature is set as $T = 0.95^{\lfloor t/5 \rfloor}$. The sampling paths all start with ethane. An Adam optimizer Kingma & Ba (2014) is used with an initial learning rate of 3×10^{-4} .

Implementation details of EBM The architecture of the EBM used in the MARS augmentation experiment is the same as the energy network of MOG. We train the EBM with the known inhibitors against both GSK3 β and JNK3, which are 315 molecules in total. When training the EBM, the step size of $\lambda = 30$ and the number of steps of $K = 150$ are used in Langevin dynamics. The number of training epochs of 20, a batch size of 32, a learning rate of 1×10^{-4} , and an Adam optimizer are used. The initial weight of the energy score is set as $\nu = 0.1$ with an exponential decay of 0.99 per step, and a warmup step of 600 is used.