De novo Drug Design using Reinforcement Learning with Multiple GPT Agents

Xiuyuan Hu[1,2]*, **Guoqing Liu**[2]†, **Yang Zhao**[1], **Hao Zhang**[1]†
[1]Department of Electronic Engineering, Tsinghua University
[2]Microsoft Research AI4Science
huxy22@mails.tsinghua.edu.cn, guoqingliu@microsoft.com,
zhao-yang@tsinghua.edu.cn, haozhang@tsinghua.edu.cn

Abstract

De novo drug design is a pivotal issue in pharmacology and a new area of focus in AI for science research. A central challenge in this field is to generate molecules with specific properties while also producing a wide range of diverse candidates. Although advanced technologies such as transformer models and reinforcement learning have been applied in drug design, their potential has not been fully realized. Therefore, we propose MolRL-MGPT, a reinforcement learning algorithm with multiple GPT agents for drug molecular generation. To promote molecular diversity, we encourage the agents to collaborate in searching for desirable molecules in diverse directions. Our algorithm has shown promising results on the GuacaMol benchmark and exhibits efficacy in designing inhibitors against SARS-CoV-2 protein targets. The codes are available at: `https://github.com/HXYfighter/MolRL-MGPT`.

1 Introduction

In recent years, significant strides have been made in computer-aided drug discovery (CADD) thanks to the development of various fields, including proteomics, genomics and deep learning [13, 66]. The conventional drug discovery process is typically time-consuming and financially demanding, with a low success rate. However, advanced machine learning techniques have the potential to reverse this predicament, greatly benefiting the economy and society's development [50, 58]. Currently, a major issue in the field of pharmacology is goal-directed *de novo* drug design, which involves generating new drug molecules with specific biochemical properties, such as designing compounds with high binding affinity to a designated protein target [15, 38]. Despite the numerous proposed machine learning algorithms for molecular generation, the chemical space is vast, and the relationship between molecular properties and structures is intricate, making it challenging to obtain satisfactory results in practical applications [42].

Molecular diversity is a critical concern in drug design because a diverse set of candidates can provide more choices for downstream screening and avoid drug resistance and unknown side effects [12, 41]. However, existing algorithms encounter challenges in designing diverse drug molecules. Many algorithms tend to generate sets of highly similar compounds, which is of little value for the subsequent drug development process.

Over the past several years, generative language models have made remarkable strides in natural language processing, vision, and audio. Among these models, the generative pre-trained transformer (GPT) is particularly notable, which has demonstrated impressive capabilities of language under-

*Work was done while Xiuyuan Hu was a research intern at Microsoft Research.
†Corresponding author: Hao Zhang (1st), Guoqing Liu.

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

https://doi.org/10.52202/075280-0325

standing and generation [45, 46, 11]. In the field of chemistry, transformer-based language models pre-trained on the SMILES (simplified molecular input line entry system) representation of molecules have emerged. Through transfer learning, these models can be adapted to a range of tasks, including molecular property prediction, reaction prediction, and molecular optimization [26, 27, 4].

Reinforcement learning (RL) has emerged as a promising approach for *de novo* drug design [39, 68, 29]. The basic idea is to consider molecular property predictors (scoring functions) as rewards and train an RL agent to iteratively generate candidate compounds with increasingly high scores. These RL algorithms can explore the vast chemical space remarkably faster than human chemists, but they may have limitations in molecular diversity. Although multi-agent reinforcement learning (MARL) is a commonly used technique for promoting diversity in searching problems [34, 14], it has not yet been effectively utilized in the field of drug design.

To address the limitations of current approaches, we propose MolRL-MGPT (**Mol**ecular design using **R**einforcement **L**earning with **M**ultiple **GPT** agents), a novel MARL framework for *de novo* drug molecular design that utilizes GPT models as agents. Our approach treats molecular design as a cooperative Markov game, where multiple lightweight GPT agents collaborate to generate high-scoring molecules during the RL process. These agents share identical pre-trained parameters on a molecular SMILES dataset for initialization and have a common optimization objective for specific properties. To enhance the diversity of generated candidates, we incorporate an auxiliary loss function that encourages agents to explore in diverse directions. Our algorithm has demonstrated superior performance compared to various baselines on the GuacaMol benchmark. We also apply it to resolve the real-world problem of designing candidates against two SARS-CoV-2 protein targets, resulting in potentially desirable candidates with good binding affinity, drug-likeness, and synthetic accessibility. Moreover, we further validate the effectiveness of our design by comparative and ablation experiments on GNK3β, JNK3 and QED maximization tasks.

2 Related works

Machine learning has become a formidable instrument in molecular generation with applications to *de novo* drug design, aiding scientists in identifying novel molecules that possess the desired properties for drug discovery. The molecular representation is the basis of molecular generation and optimization algorithms, which can be roughly divided into three categories: 1D string, 2D image, and 3D geometry [15].

SMILES is the most commonly used 1D representation of molecules, which employs strings of characters to encode a molecule. Although the SMILES of each molecule is not unique, there is only one canonical SMILES, and each SMILES corresponds to a maximum of one molecule. It is noteworthy that most SMILES strings are invalid; that is, their corresponding structures cannot exist in the real world. Some algorithms and techniques applied in natural language processing (NLP) have been adopted to generate molecules using SMILES representations, such as variational autoencoder (VAE) [22, 16], recurrent neural network (RNN) [49], generative adversarial network (GAN) [23] and Bayesian optimization (BO) [37].

On the other hand, 2D and 3D molecular representations are more intuitive and have become popular in molecular design in recent years. For 2D molecular graphs with vertices corresponding to atoms and edges corresponding to chemical bonds between atoms, techniques such as graph neural network (GNN), genetic algorithm (GA) and flow network for graph data have already been applied to drug development [1, 28, 30, 62, 35, 21]. For 3D geometries of compounds, which theoretically contain the most structural information of the molecules, although models including diffusion have been introduced to the chemistry field [20, 33], they cannot currently design candidates with desired biochemical properties as 1D/2D *de novo* drug design approaches.

RL-based drug design algorithms Reinforcement learning is the most popular technique for molecule generation. In molecular design, deep neural networks are usually employed as agents in studies that use RL for 1D string generation, which are then fine-tuned via customized reward functions [39, 9, 41, 59]. Likewise, when RL is applied to generating 2D molecular graphs, the states correspond to incomplete representations, and the actions involve adding substructures (such as atoms, bonds, and rings) to specific positions [65, 68, 29, 1, 63, 19]. What is more, recent studies

https://doi.org/10.52202/075280-0325

have incorporated 3D geometries into the RL process to consider the spatial properties of molecules during their generation [51, 52, 18].

Transformers for chemical language Transformer is a type of deep neural network architecture entirely based on attention mechanism and has been widely applied in the field of NLP [57]. It has demonstrated a better ability to process long text sequences and parallel computing capabilities than traditional architectures such as RNN. Some works have already applied transformers in the field of chemistry. For instance, ChemFormer [26], MolGPT [4] and [24] have focused on SMILES-based pre-trained transformer models. Meanwhile, PROTAC-RL [67], TamGent [61] and [44] have concentrated on transformer-based drug design against protein targets, while MCMG [59] has used transformer as a component to enhance the learning capability of the algorithm.

Diversity in drug design Previous research has proposed using reinforcement learning to generate diverse molecules for drug development [8, 59, 63, 41]. However, these studies have not adequately addressed this concern, and multi-agent reinforcement learning is yet to be effectively applied in *de novo* drug design.

3 Methodology

3.1 Problem definition

The fundamental aspect of the molecular generation problem formulation is a scoring function $s(x)$ of molecular properties, also known as an oracle, with the input of x being a molecule and the output being a real number. Molecular properties can include physical properties such as molecular weight and the number of aromatic rings, as well as chemical properties such as logP and drug-likeness. Additionally, the molecular properties that we are most concerned with in real-world drug design are often related to biological activity, with binding affinity to protein targets being the most common, which can be estimated using docking software.

Standardly, the scoring function $s(\cdot)$ is typically constrained within the interval $[0, 1]$ when applied to a valid molecular input, where a higher score corresponds to a better molecular property. Invalid molecules consistently receive a score of -1. Researchers may implement a transformation function $t(\cdot)$ for each molecule property predictor $p(x)$ to achieve uniformity with the standard form, and create a multi-objective scoring function through a weighted combination of various oracles.

$$\begin{gathered} s(x) \in [0,1] \cup \{-1\}, \\ s(x) = t(p(x)) \ \text{ or } \ s(x) = \sum_i w_i \cdot t_i(p_i(x)), \ \sum_i w_i = 1 \end{gathered} \tag{1}$$

The evaluation of a *de novo* molecule generation algorithm usually requires the assessment of a set of generated high-scoring molecules, and a common approach is to report the average score and diversity of the top-k scoring generated molecules, where $k \in \mathbb{N}^+$ is given. A widely used metric of molecular diversity is internal diversity (IntDiv) [6]:

$$\text{IntDiv}(A) := \frac{1}{|A|(|A|-1)} \sum_{(x,y)\in A\times A, x\neq y} d_T(\mathcal{F}(x), \mathcal{F}(y)) \tag{2}$$

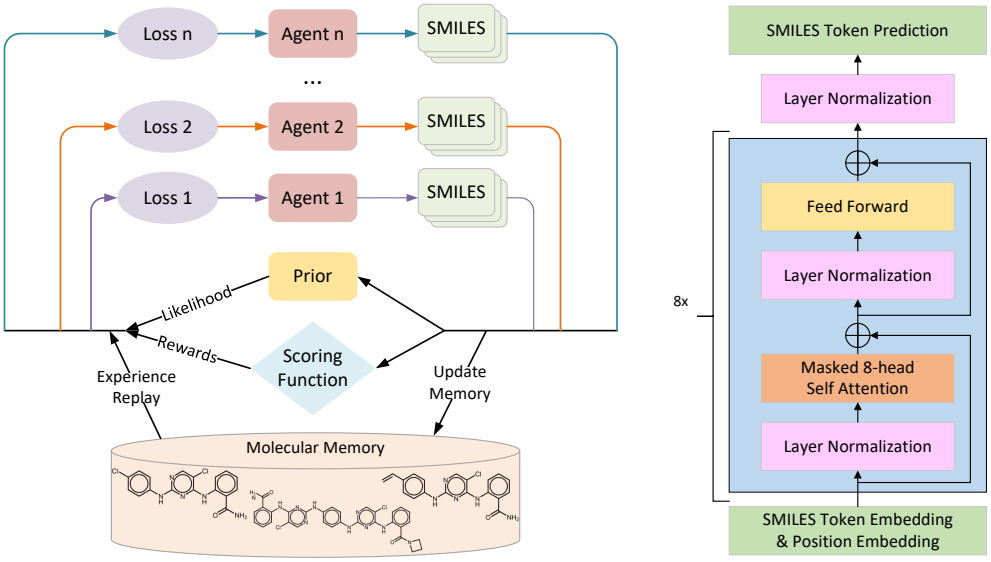
where A is a set of compounds, d_T represents the Tanimoto distance [55], and $\mathcal{F}(x)$ refers to the extended-connectivity fingerprint (ECFP) [47] of a molecule x.

In drug molecular design tasks, we primarily focus on evaluating the final set of generated molecular candidates. Generally, we pay little attention to the generation process, including factors like time and computing resource consumption, as these costs are relatively insignificant compared to the conventional drug discovery process.

Hence, we can represent the problem of designing novel drug candidates as a cooperative Markov game consisting of multiple generative model agents. At every iteration i ($i = 1, 2, \ldots, s$), each agent k ($k = 1, 2, \ldots, n$) generates m molecules (actions), and the scoring function acts as an environment providing rewards in the form of scores, which in turn are used to update the agents. The game's objective is primarily to maximize the average of the highest k scores of generated molecules and secondarily to improve the diversity of the candidates.

https://doi.org/10.52202/075280-0325

3.2 MolRL-MGPT algorithm



(a) (b)

Figure 1: (a) Overview of our MolRL-MGPT algorithm. (b) The model architecture of the GPT prior model and agents.

As illustrated in Fig 1 (a), the MolRL-MGPT algorithm consists of iterations in which multiple agents are updated in a reinforcement learning way. Specifically, in each iteration, each agent first samples a set of SMILES strings with high likelihood and retrieves several high-scoring molecules in the past iterations from the molecular memory. Then, the scoring function is used to obtain the reward of each generated SMILES string. The GPT agents conduct this process in order. The loss functions are designed as follows.

Loss function In MolRL-MGPT, the reward function is defined as the scores predicted by the scoring function. Our primary objective is to increase the average scores of SMILES strings generated by each agent. Furthermore, to prevent the agents from disregarding the knowledge acquired during the pre-training phase, we impose a penalty on the deviation between the new policies and the prior model, which is also employed to initialize all the agents. The loss function of the 1st GPT agent is designed as follows:

$$L_1(x;\Theta_1) = [\log P(x)_{\text{Prior}} - \log P(x)_{\text{Agent}_1} + \sigma_1 \cdot s(x)]^2 \tag{3}$$

where Θ_1 is the parameters of the 1st agent, x is a generated molecule, σ_1 is a coefficient for controlling the term of scores, and $P(x)_{\text{model}}$ refers to the likelihood of generating x by model. It should be noted that typically $P(x)_{\text{Prior}} < P(x)_{\text{Agent}}$.

Additionally, we encourage the agents to explore different directions in the chemical space instead of conducting repetitive searches by introducing a term indicating the deviation between the current agent and previous ones. The loss function of the kth GPT agent is designed as follows:

$$\begin{aligned} L_k(x;\Theta_k) =& L_1(x;\Theta_k) - \sigma_2 \sum_{j=1}^{k-1} s(x) \cdot |\log P(x)_{\text{Agent}_k} - \log P(x)_{\text{Agent}_j}| \\ =& [\log P(x)_{\text{Prior}} - \log P(x)_{\text{Agent}_k} + \sigma_1 \cdot s(x)]^2 \\ & - \sigma_2 \sum_{j=1}^{k-1} s(x) \cdot |\log P(x)_{\text{Agent}_k} - \log P(x)_{\text{Agent}_j}| \end{aligned} \tag{4}$$

where $k = 1, 2, \ldots n$, Θ_k is the parameters of the kth agent, and σ_2 is a coefficient for the encouragement term.

The selection of coefficients σ_1, σ_2 can be adapted according to the specific task at hand. Moreover, we propose to implement a decreasing schedule for σ_1 such that σ_1 decreases as $s(x)$ increases during the RL process. This is because we expect the increase in scores to correspond to a decrease in loss.

Molecular Memory MolRL-MGPT utilizes a memory consisting of high-scoring molecules with a maximum size of 1000. The memory is updated with every SMILES string sampled, and compounds stored in the memory are sorted based on their scores.

3.3 Implementations

Pre-training on chemical language Instead of the commonly used recurrent neural network (RNN) architecture for generating chemical language in SMILES strings, we adopt a transformer-based architecture. As shown in Fig 1 (b), our model is a mini version of the GPT model with 8 layers of transformer blocks, each consisting of 8 attention heads. The embedding size is 256, and the maximum length of SMILES strings is set to 128. The pre-training approach of our model involved learning molecular structure patterns. Our prior model has only around 6.4M parameters, much less than the GPT-2 model [46].

Two versions of the prior model are trained using unsupervised learning on the ChEMBL [36] and ZINC-100M [53] datasets, respectively. The data are preprocessed by removing molecules with ionized structures and SMILES strings longer than 100 characters, which are not typical for small molecule drugs. Additionally, we use SMILES randomization for data augmentation, which has been proven helpful in enhancing generating capacity [3]. Ten training epochs are conducted with a batch size of 2048 and a maximum learning rate of 0.001, using a learning rate schedule featuring warm-up and cosine decay. The training on ChEMBL (roughly 2 million SMILES) takes around 5 hours using a single NVIDIA A100 GPU.

The pre-trained SMILES models should be objectively evaluated by the valid ratio of the generated molecules, representing the proportion of valid molecules present in the generated SMILES. Generally, this value can exceed 90% [43]. Our pre-trained transformer model has achieved a valid ratio of 98% in generating molecules, indicating its success in capturing the inherent grammar and rules of molecular SMILES strings.

Experience replay Experience replay is a widely employed technique in deep reinforcement learning. The agent or learner records its experience in a memory buffer and randomly samples past experiences, enabling it to learn from previous experiences with reduced correlation between consecutive training samples. This technique helps agents avoid overfitting, generalize to unseen situations, and achieve a more stable and efficient learning process [32]. In MolRL-MGPT, we replay the "successful" experiences by randomly sampling 5 molecules from the 25 highest-scoring molecules in the molecular memory and computing the loss on these molecules together with SMILES strings generated in the current iteration.

***Similarity penalization** Some previous works for *de novo* drug design add a penalizing term for high similarity of identical skeletons between new molecules and previously found high-scoring molecules in order to encourage the agent to search in unexplored space rather than repeatedly finding the same or similar compounds [9, 60]. However, our experiments indicate that this trick does not work with our algorithm.

4 Experiments

To demonstrate the performance of MolRL-MGPT, we conduct three groups of experiments. Firstly, we run our algorithm on the public *de novo* molecular design benchmark, GuacaMol, and compare it with existing advanced methods. Secondly, we apply MolRL-MGPT to the design of inhibitors against SARS-CoV-2 protein targets, which is a current real-world challenge for human beings. Thirdly, we conduct comparative and ablation experiments on GNK3β, JNK3 and QED maximization tasks to validate the effectiveness of modules in our design.

7409 https://doi.org/10.52202/075280-0325

4.1 GuacaMol benchmark

GuacaMol [10] is a widely recognized open-source evaluation framework for *de novo* molecular design algorithms. It contains 20 goal-directed tasks mimicking the drug discovery objectives, corresponding to 20 standard scoring functions described in Sec 3.1. These tasks cover commonly used objectives in drug design, such as structural, physicochemical, and biochemical properties.

For comparison, we select the following baselines: (1) SMILES GA [64], a genetic algorithm based on SMILES; (2) SMILES LSTM [49], an LSTM network generating SMILES strings autoregressively, combined with a hill-climb algorithm for optimization; (3) Graph GA [28], a genetic algorithm with crossovers and mutations performed on molecular graphs; (4) Reinvent [39], a deep reinforcement learning framework for training an RNN model generating SMILES; (5) GEGL [1], a genetic expert-guided learning framework for training an RNN for molecular generation. Results of some other baselines are shown in the Appendix.

The prior model for MolRL-MGPT was pre-trained on the official GuacaMol dataset, which is a subset of ChEMBL. The hyper-parameters are set such that we run each tasks for 5000 tasks (break if the score has achieved 1.000) with 4 GPT agents, and the batch size (number of sampled SMILES strings) of each agent is 256. The values of coefficients are set to $\sigma_1 = 1000$ with a linear decreasing schedule, and $\sigma_2 = 0.1$. The entire set of 20 tasks tasks less than 400 hours to complete when run on a single NVIDIA A100 GPU.

Table 1: Scores of MolRL-MGPT and other baselines on the GuacaMol benchmark.

Tasks	SMILES GA	SMILES LSTM	Graph GA	Reinvent	GEGL	**MolRL-MGPT**
1. Celecoxib rediscovery	0.732	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**
2. Troglitazone rediscovery	0.515	**1.000**	**1.000**	**1.000**	0.552	**1.000**
3. Thiothixene rediscovery	0.598	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**
4. Aripiprazole similarity	0.834	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**
5. Albuterol similarity	0.907	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**
6. Mestranol similarity	0.790	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**
7. $C_{11}H_{24}$	0.829	0.993	0.971	0.999	**1.000**	**1.000**
8. $C_9H_{10}N_2O_2PF_2Cl$	0.889	0.879	0.982	0.877	**1.000**	0.939
9. Median molecules 1	0.334	0.438	0.406	0.434	**0.455**	0.449
10. Median molecules 2	0.380	0.422	0.432	0.395	**0.437**	0.422
11. Osimertinib MPO	0.886	0.907	0.953	0.889	**1.000**	0.977
12. Fexofenadine MPO	0.931	0.959	0.998	**1.000**	**1.000**	**1.000**
13. Ranolazine MPO	0.881	0.855	0.920	0.895	0.933	**0.939**
14. Perindopril MPO	0.661	0.808	0.792	0.764	**0.833**	0.810
15. Amlodipine MPO	0.722	0.894	0.894	0.888	0.905	**0.906**
16. Sitagliptin MPO	0.689	0.545	**0.891**	0.539	0.749	0.823
17. Zaleplon MPO	0.413	0.669	0.754	0.590	0.763	**0.790**
18. Valsartan SMARTS	0.552	0.978	0.990	0.095	**1.000**	0.997
19. deco hop	0.970	0.996	**1.000**	0.994	**1.000**	**1.000**
20. scaffold hop	0.885	0.998	**1.000**	0.990	**1.000**	**1.000**
Total	14.396	17.340	17.983	16.350	17.627	**18.052**

As shown in Table 1, the MolRL-MGPT algorithm outperforms baselines in 13 molecular design tasks in the GuacaMol benchmark, and its total score of 20 tasks also ranks first. These results indicate that MolRL-MGPT excels in general scenarios of *de novo* drug design.

4.2 Designing inhibitors against SARS-CoV-2 targets

Molecular docking is a computational method used to predict the binding modes of small molecules to a protein target. It involves predicting the spatial orientation and binding affinity of the small molecule in the active site of the protein. This information is useful in drug discovery since it enables identifying potential drug candidates and understanding how they interact with their targets. Autodock Vina [56] is currently the most widely used molecular docking software. However, we opt to use Quick Vina 2 [2], a novel and more efficient alternative.

https://doi.org/10.52202/075280-0325

The quantitative binding affinity is named docking score, which is calculated based on the energies of the interaction between the ligand and the receptor, and a lower docking score indicates a more stable and, therefore, more likely binding pose. Typically, docking scores are negative, and desirable docking scores range from -10 to -14 kcal/mol. Therefore, we use a reverse sigmoid function as the transformation function of the docking score:

$$t_{\text{docking}}(p) = \frac{1}{1 + 10^{0.625 \cdot (p+10)}} \tag{5}$$

SARS-CoV-2 (Severe Acute Respiratory Syndrome Coronavirus 2), commonly referred to as the novel coronavirus, is a respiratory virus which, in late 2019, emerged as a global pandemic leading to the COVID-19 disease that mainly targets the respiratory system. This disease is a severe public health concern that continues to pose a crisis worldwide and requires a comprehensive response to mitigate its spread and negative effects. Therefore, we apply our algorithm to the design of inhibitors against protein targets of SARS-CoV-2, which is a significant real-world issue. Following [48], we select two targets: `PLPro_7JIR`[3] and `RdRp_6YYT`[4].

Besides docking scores, we also consider two additional oracles often employed in practical drug design: (1) **QED** (Quantitative Estimate of Drug-likeness), which quantifies the drug-likeness of a molecule based on the concept of desirability of eight molecular properties [7], ranging in $[0, 1]$; (2) **SA** (Synthetic Accessibility), which incorporates fragment contributions and a complexity penalty [17], ranging in $[1, 10]$. Their transformation functions are linear:

$$t_{\text{QED}}(p) = p, \quad t_{\text{SA}}(p) = \frac{10 - p}{9} \tag{6}$$

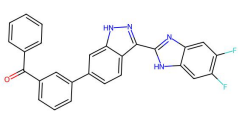
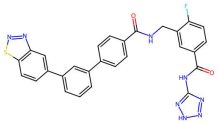
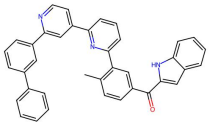
Moreover, the scoring function for designing inhibitors against SARS-CoV-2 protein targets is a linear combination of docking scores, QED scores and SA scores:

$$s_{\text{total}}(x) = 0.8 \cdot s_{\text{docking}}(x) + 0.1 \cdot s_{\text{QED}}(x) + 0.1 \cdot s_{\text{SA}}(x) \tag{7}$$

We run the RL process for 1000 iterations on each target with 4 GPT agents, and the batch size of each agent is 128. The whole process for one target takes approximately 100 hours on a single NVIDIA A100 GPU and 64 CPU cores.[5] Details of the generated candidates are as follows.

PLPro_7JIR target PLPro (papain-like protease) is an attractive target for SARS-CoV-2 since it plays a fundamental role in cleavage and maturation of viral polyproteins, assembly of the replicase-transcriptase complex, and disruption of host responses. 7JIR is a C111S mutant form of the structure of PLPro [40]. Three candidate inhibitors against the PLPro_7JIR target generated by MolRL-MGPT are shown in Table 2.

Table 2: Candidate inhibitors against the PLPro_7JIR target generated by MolRL-MGPT. The SMILES of the three candidates are:
(1) O=C(c1cccc(-c2ccc3c(-c4nc5cc(F)c(F)cc5[nH]4)n[nH]c3c2)c1)c1ccccc1;
(2) c1(-c2cc(-c3cc4c(cc3)snn4)ccc2)ccc(C(NCc2cc(C(=O)Nc3nn[nH]n3)ccc2F)=O)cc1;
(3) c1c(-c2c(C)ccc(C(=O)c3cc4ccccc4[nH]3)c2)nc(-c2ccnc(-c3cccc(-c4ccccc4)c3)c2)cc1.

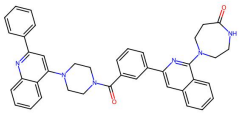
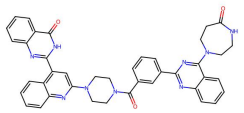
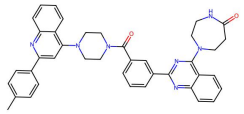
Molecule			
docking score (↓)	-11.3	-11.1	-11.2
QED score (↑)	0.310	0.258	0.214
SA score (↓)	2.530	2.729	2.549

[3] `https://www.rcsb.org/structure/7JIR`
[4] `https://www.rcsb.org/structure/6YYT`
[5] Docking consumes more than 95% of the time, although parallel computing of Quick Vina software is implemented.

https://doi.org/10.52202/075280-0325

RdRp_6YYT target RdRp (RNA-dependent RNA polymerase) works for the replication of genome and the transcription of genes of SARS-CoV-2, and 6YYT is the PDB identification code of its structure [25]. Three candidate inhibitors against the RdRp_6YYT target generated by MolRL-MGPT are shown in Table 3.

Table 3: Candidate inhibitors against the RdRp_6YYT target generated by MolRL-MGPT. The SMILES of the three candidates are:
(1) O=C1CCN(c2nc(-c3cccc(C(=O)N4CCN(c5cc(-c6ccccc6)nc6ccccc56)CC4)c3)cc3ccccc23)CCN1;
(2) O=C1CCN(c2nc(-c3cccc(C(=O)N4CCN(c5cc(-c6nc7ccccc7c(=O)[nH]6)c6ccccc6n5)CC4)c3)nc3ccccc23)CCN1;
(3) Cc1ccc(-c2cc(N3CCN(C(=O)c4cccc(-c5nc(N6CCNC(=O)CC6)c6ccccc6n5)c4)CC3)c3ccccc3n2)cc1.

Molecule			
docking score (↓)	-12.3	-13.1	-13.2
QED score (↑)	0.237	0.253	0.241
SA score (↓)	2.772	3.104	2.806

The generated candidates against the PLPro_7JIR and RdRp_6YYT targets exhibit desirable binding affinities (docking scores) and synthetic accessibility. Although the drug-likeness scores of these candidates may not be high, it is reasonable since QED is estimated based on the distribution of existing drug molecules. All these drugs are ineffective in inhibiting the two targets of SARS-CoV-2; thus, designing new dissimilar compounds becomes necessary.

4.3 GSK3β, JNK3 and QED maximization

To demonstrate the effectiveness of our design, we perform ablation experiments on commonly used oracles for simulating real-world drug design with low consumption: GSK3β and JNK3. Their scores are estimated by random forests trained on ExCAPE-DB dataset [54], which measure the bioactivities of molecules against the Glycogen synthase kinase 3 beta target (GSK3β) and the c-Jun N-terminal kinase 3 target (JNK3). Previous research has shown that inhibiting these targets can benefit the treatment of Alzheimer's Disease [31].

We carry out ablation experiments to validate the following settings of MolRL-MGPT:

1. The optimal number of agents for a fixed total batch size,
2. Whether the loss term that encourages agents to search in different directions (ED) really works,
3. Whether the experience replay (ER) really works,
4. Whether the decreasing schedule of σ_1 (DS) really works,
5. Whether the possible technique of similarity penalization (SP) works [6].

The base algorithm is denoted as MolRL-MGPT, which consists of 4 agents and incorporates the modules ED, ER, DS, without SP. For both the GSK3β and JNK3 tasks, we present the average and standard deviation of the mean scores and internal diversities of the top-100 high-scoring molecules generated by each strategy.

Besides ablation experiments, we also compare the performance of MolRL-MGPT with several baselines on GSK3β, JNK3 and QED maximization tasks. The baselines are Graph GA [28], Reinvent [39], JT-VAE [30] and GFlowNet [5]. More details are provided in the Appendix.

Table 4 indicates that the algorithm demonstrates the best performance with four agents, and additional agents do not improve the results. Encouraging agents to explore different paths in the chemical

[6]Penalize on the score of a molecule if its similarity to one of previously generated candidates is larger than 0.8.

https://doi.org/10.52202/075280-0325

Table 4: Results of experiments on GSK3β and JNK3 maximization.

	GSK3β top-100		JNK3 top-100	
	mean score	IntDiv	mean score	IntDiv
1 agent	1.000 ± 0.000	0.318 ± 0.020	0.954 ± 0.012	0.343 ± 0.017
2 agents	1.000 ± 0.000	0.335 ± 0.017	0.960 ± 0.012	0.357 ± 0.028
MolRL-MGPT	1.000 ± 0.000	0.362 ± 0.015	0.961 ± 0.010	0.372 ± 0.025
8 agents	1.000 ± 0.000	0.360 ± 0.020	0.958 ± 0.015	0.369 ± 0.018
w/o ED	1.000 ± 0.000	0.285 ± 0.023	0.961 ± 0.008	0.345 ± 0.025
w/o ER	0.964 ± 0.005	0.332 ± 0.019	0.918 ± 0.008	0.356 ± 0.023
w/o DS	0.997 ± 0.001	0.358 ± 0.016	0.940 ± 0.014	0.370 ± 0.027
w/ SP	1.000 ± 0.000	0.360 ± 0.021	0.956 ± 0.009	0.365 ± 0.015
GFlowNet	0.649 ± 0.072	0.715 ± 0.104	0.437 ± 0.219	0.716 ± 0.145
GraphGA	0.919 ± 0.016	0.365 ± 0.024	0.875 ± 0.025	0.380 ± 0.015
JT-VAE	0.235 ± 0.083	0.770 ± 0.067	0.159 ± 0.040	0.781 ± 0.127
Reinvent	0.965 ± 0.011	0.308 ± 0.035	0.942 ± 0.019	0.368 ± 0.021

Table 5: Results of experiments on QED maximization.

	QED top-100	
	mean score	IntDiv
MolRL-MGPT	0.948 ± 0.000	0.862 ± 0.004
GFlowNet	0.938 ± 0.001	0.809 ± 0.017
GraphGA	0.928 ± 0.001	0.845 ± 0.005
JT-VAE	0.921 ± 0.003	0.856 ± 0.012
Reinvent	0.948 ± 0.000	0.658 ± 0.035

space does promote diversity of the generated molecules, and using experience replay and decreasing schedule of σ_1 benefits mean scores of candidates. Additionally, the similarity penalization trick does not appear to be effective in MolRL-MGPT. As a consequence, we have verified the effectiveness of our design.

As shown in Table 4 and 5, compared with baselines with competitive mean scores, MolRL-MGPT performs better in internal diversity.

5 Conclusion and discussion

In this paper, we present MolRL-MGPT, a multi-agent reinforcement learning framework for *de novo* drug molecular design, which adopts transformer models as agents and the fundamental idea is to encourage agents to collaborate to search with different directions in the chemical space. MolRL-MGPT demonstrates superior performance on the GuacaMol benchmark and does well in designing inhibitors against SARS-CoV-2 protein targets.

Admittedly, there exist some possible approaches for further improvements to our algorithm:

- **Better data source**: As we all know, data sources play a decisive role in the performance of ML algorithms in chemistry and biology. Higher quality pre-training data or more annotated data tailored to specific tasks may further improve the performance of MolRL-MGPT.
- **Scoring function**: The design of the scoring function may also improve the algorithm's performance. For example, in multi-property joint tasks, adjusting the coefficients of terms in the scoring function may be beneficial, and more accurate and fast software for predicting molecular properties is also helpful.
- **Insights for specific objectives**: In practical drug development, with more specialized and in-depth research on each objective, we should utilize the knowledge specific to certain tasks more fully to design candidate drug molecules better.

In summary, MolRL-MGPT is a promising and feasible approach for *de novo* drug design. It provides pharmaceutical researchers with a fast and effective method to generate a diverse set of molecular structures that meet specified conditions, as long as the scoring functions of these conditions are provided. We believe that MolRL-MGPT will be of great assistance in drug discovery.

7413 https://doi.org/10.52202/075280-0325

References

[1] Sungsoo Ahn, Junsu Kim, Hankook Lee, and Jinwoo Shin. Guiding deep molecular optimization with genetic exploration. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 12008–12021. Curran Associates, Inc., 2020.

[2] 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.

[3] Josep Arús-Pous, Simon Viet Johansson, Oleksii Prykhodko, Esben Jannik Bjerrum, Christian Tyrchan, Jean-Louis Reymond, Hongming Chen, and Ola Engkvist. Randomized smiles strings improve the quality of molecular generative models. *Journal of cheminformatics*, 11(1):1–13, 2019.

[4] Viraj Bagal, Rishal Aggarwal, P. K. Vinod, and U. Deva Priyakumar. Molgpt: Molecular generation using a transformer-decoder model. *Journal of Chemical Information and Modeling*, 62(9):2064–2076, 2022.

[5] Emmanuel Bengio, Moksh Jain, Maksym Korablyov, Doina Precup, and Yoshua Bengio. Flow network based generative models for non-iterative diverse candidate generation. *Advances in Neural Information Processing Systems*, 34:27381–27394, 2021.

[6] Mostapha Benhenda. Can ai reproduce observed chemical diversity? *bioRxiv*, page 292177, 2018.

[7] 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.

[8] Thomas Blaschke, Josep Arús-Pous, Hongming Chen, Christian Margreitter, Christian Tyrchan, Ola Engkvist, Kostas Papadopoulos, and Atanas Patronov. Reinvent 2.0: An ai tool for de novo drug design. *Journal of chemical information and modeling*, 2020.

[9] Thomas Blaschke, Ola Engkvist, Jürgen Bajorath, and Hongming Chen. Memory-assisted reinforcement learning for diverse molecular de novo design. *Journal of chemical information and modeling*, 2020.

[10] Nathan Brown, Marco Fiscato, Marwin HS Segler, and Alain C Vaucher. Guacamol: benchmarking models for de novo molecular design. *Journal of chemical information and modeling*, 59(3):1096–1108, 2019.

[11] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[12] Philip Michael Dean and Richard A Lewis. *Molecular diversity in drug design*. Springer, 1999.

[13] Jianyuan Deng, Zhibo Yang, Iwao Ojima, Dimitris Samaras, and Fusheng Wang. Artificial intelligence in drug discovery: applications and techniques. *Briefings in Bioinformatics*, 23(1):bbab430, 2022.

[14] Wei Du and Shifei Ding. A survey on multi-agent deep reinforcement learning: from the perspective of challenges and applications. *Artificial Intelligence Review*, 54:3215–3238, 2021.

[15] Yuanqi Du, Tianfan Fu, Jimeng Sun, and Shengchao Liu. Molgensurvey: A systematic survey in machine learning models for molecule design. *arXiv preprint arXiv:2203.14500*, 2022.

[16] Peter Eckmann, Kunyang Sun, Bo Zhao, Mudong Feng, Michael K Gilson, and Rose Yu. Limo: Latent inceptionism for targeted molecule generation. In *International Conference on Machine Learning*. PMLR, 2022.

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

[18] Tianfan Fu, Wenhao Gao, Connor W Coley, and Jimeng Sun. Reinforced genetic algorithm for structure-based drug design. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems NeurIPS*, 2022.

[19] Tianfan Fu, Cao Xiao, Lucas M Glass, and Jimeng Sun. Moler: incorporate molecule-level reward to enhance deep generative model for molecule optimization. *IEEE transactions on knowledge and data engineering*, 34(11):5459–5471, 2021.

[20] Niklas Gebauer, Michael Gastegger, and Kristof Schütt. Symmetry-adapted generation of 3d point sets for the targeted discovery of molecules. *Advances in neural information processing systems*, 32, 2019.

https://doi.org/10.52202/075280-0325

[21] Zijie Geng, Shufang Xie, Yingce Xia, Lijun Wu, Tao Qin, Jie Wang, Yongdong Zhang, Feng Wu, and Tie-Yan Liu. De novo molecular generation via connection-aware motif mining. In *International Conference on Learning Representations*, 2023.

[22] 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.

[23] 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 preprint arXiv:1705.10843*, 2017.

[24] Jiazhen He, Eva Nittinger, Christian Tyrchan, Werngard Czechtizky, Atanas Patronov, Esben Jannik Bjerrum, and Ola Engkvist. Transformer-based molecular optimization beyond matched molecular pairs. *Journal of cheminformatics*, 14(1):18, 2022.

[25] Hauke S Hillen, Goran Kokic, Lucas Farnung, Christian Dienemann, Dimitry Tegunov, and Patrick Cramer. Structure of replicating sars-cov-2 polymerase. *Nature*, 584(7819):154–156, 2020.

[26] Ross Irwin, Spyridon Dimitriadis, Jiazhen He, and Esben Jannik Bjerrum. Chemformer: a pre-trained transformer for computational chemistry. *Machine Learning: Science and Technology*, 3(1):015022, 2022.

[27] Kevin Maik Jablonka, Philippe Schwaller, and Berend Smit. Is gpt-3 all you need for machine learning for chemistry? In *AI for Accelerated Materials Design NeurIPS 2022 Workshop*, 2022.

[28] Jan H Jensen. A graph-based genetic algorithm and generative model/monte carlo tree search for the exploration of chemical space. *Chemical science*, 10(12):3567–3572, 2019.

[29] Wengong Jin, Regina Barzilay, and T. Jaakkola. Multi-objective molecule generation using interpretable substructures. In *International Conference on Machine Learning*, page 4849–4859. PMLR, 2020.

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

[31] Yibo Li, Liangren Zhang, and Zhenming Liu. Multi-objective de novo drug design with conditional graph generative model. *Journal of cheminformatics*, 10(1):1–24, 2018.

[32] Long-Ji Lin. Self-improving reactive agents based on reinforcement learning, planning and teaching. *Machine learning*, 8(3):293–321, 1992.

[33] Youzhi Luo and Shuiwang Ji. An autoregressive flow model for 3d molecular geometry generation from scratch. In *International Conference on Learning Representations (ICLR)*, 2022.

[34] Anuj Mahajan, Tabish Rashid, Mikayel Samvelyan, and Shimon Whiteson. Maven: Multi-agent variational exploration. *Advances in Neural Information Processing Systems*, 32, 2019.

[35] Krzysztof Maziarz, Henry Jackson-Flux, Pashmina Cameron, Finton Sirockin, Nadine Schneider, Nikolaus Stiefl, Marwin Segler, and Marc Brockschmidt. Learning to extend molecular scaffolds with structural motifs. In *International Conference on Learning Representations (ICLR)*, 2022.

[36] David Mendez, Anna Gaulton, A Patrícia Bento, Jon Chambers, Marleen De Veij, Eloy Félix, María Paula Magariños, Juan F Mosquera, Prudence Mutowo, Michał Nowotka, et al. Chembl: towards direct deposition of bioassay data. *Nucleic acids research*, 47(D1):D930–D940, 2019.

[37] Henry Moss, David Leslie, Daniel Beck, Javier Gonzalez, and Paul Rayson. Boss: Bayesian optimization over string spaces. *Advances in neural information processing systems*, 33:15476–15486, 2020.

[38] Varnavas D Mouchlis, Antreas Afantitis, Angela Serra, Michele Fratello, Anastasios G Papadiamantis, Vassilis Aidinis, Iseult Lynch, Dario Greco, and Georgia Melagraki. Advances in de novo drug design: From conventional to machine learning methods. *International journal of molecular sciences*, 22(4):1676, 2021.

[39] Marcus Olivecrona, Thomas Blaschke, Ola Engkvist, and Hongming Chen. Molecular de-novo design through deep reinforcement learning. *Journal of Cheminformatics*, 9, 2017.

[40] Jerzy Osipiuk, Saara-Anne Azizi, Steve Dvorkin, Michael Endres, Robert Jedrzejczak, Krysten A Jones, Soowon Kang, Rahul S Kathayat, Youngchang Kim, Vladislav G Lisnyak, et al. Structure of papain-like protease from sars-cov-2 and its complexes with non-covalent inhibitors. *Nature communications*, 12(1):743, 2021.

https://doi.org/10.52202/075280-0325

[41] Tiago Pereira, Maryam Abbasi, Bernardete Ribeiro, and Joel P Arrais. Diversity oriented deep reinforcement learning for targeted molecule generation. *Journal of cheminformatics*, 13(1):1–17, 2021.

[42] Pavel G Polishchuk, Timur I Madzhidov, and Alexandre Varnek. Estimation of the size of drug-like chemical space based on gdb-17 data. *Journal of computer-aided molecular design*, 27(8):675–679, 2013.

[43] Daniil Polykovskiy, Alexander Zhebrak, Benjamin Sanchez-Lengeling, Sergey Golovanov, Oktai Tatanov, Stanislav Belyaev, Rauf Kurbanov, Aleksey Artamonov, Vladimir Aladinskiy, Mark Veselov, et al. Molecular sets (moses): a benchmarking platform for molecular generation models. *Frontiers in pharmacology*, 11:565644, 2020.

[44] Hao Qian, Cheng Lin, Dengwei Zhao, Shikui Tu, and Lei Xu. Alphadrug: protein target specific de novo molecular generation. *PNAS Nexus*, 1(4):pgac227, 2022.

[45] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. Improving language understanding by generative pre-training. 2018.

[46] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

[47] David Rogers and Mathew Hahn. Extended-connectivity fingerprints. *Journal of Chemical Information and Modeling*, 50(5):742–754, 2010.

[48] David M Rogers, Rupesh Agarwal, Josh V Vermaas, Micholas Dean Smith, Rajitha T Rajeshwar, Connor Cooper, Ada Sedova, Swen Boehm, Matthew Baker, Jens Glaser, et al. Sars-cov2 billion-compound docking. *Scientific Data*, 10(1):173, 2023.

[49] Marwin HS Segler, Thierry Kogej, Christian Tyrchan, and Mark P Waller. Generating focused molecule libraries for drug discovery with recurrent neural networks. *ACS central science*, 4(1):120–131, 2018.

[50] Marwin HS Segler, Mike Preuss, and Mark P Waller. Planning chemical syntheses with deep neural networks and symbolic ai. *Nature*, 555(7698):604–610, 2018.

[51] Gregor Simm, Robert Pinsler, and José Miguel Hernández-Lobato. Reinforcement learning for molecular design guided by quantum mechanics. In *International Conference on Machine Learning*, pages 8959–8969. PMLR, 2020.

[52] Gregor N. C. Simm, Robert Pinsler, Gábor Csányi, and José Miguel Hernández-Lobato. Symmetry-aware actor-critic for 3d molecular design. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021.

[53] Teague Sterling and John J. Irwin. Zinc 15 – ligand discovery for everyone. *Journal of chemical information and modeling*, 15(11):2324–2337, 2015.

[54] Jiangming Sun, Nina Jeliazkova, Vladimir Chupakhin, Jose-Felipe Golib-Dzib, Ola Engkvist, Lars Carlsson, Jörg Wegner, Hugo Ceulemans, Ivan Georgiev, Vedrin Jeliazkov, et al. Excape-db: an integrated large scale dataset facilitating big data analysis in chemogenomics. *Journal of cheminformatics*, 9(1):1–9, 2017.

[55] T. T. Tanimoto. *An elementary mathematical theory of classification and prediction*. IBM Internal Report, 1958.

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

[57] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[58] Hanchen Wang, Tianfan Fu, Yuanqi Du, Wenhao Gao, Kexin Huang, Ziming Liu, Payal Chandak, Shengchao Liu, Peter Van Katwyk, Andreea Deac, et al. Scientific discovery in the age of artificial intelligence. *Nature*, 620(7972):47–60, 2023.

[59] Jike Wang, Chang-Yu Hsieh, Mingyang Wang, Xiaorui Wang, Zhenxing Wu, Dejun Jiang, Benben Liao, Xujun Zhang, Bo Yang, Qiaojun He, et al. Multi-constraint molecular generation based on conditional transformer, knowledge distillation and reinforcement learning. *Nature Machine Intelligence*, 3(10):914–922, 2021.

https://doi.org/10.52202/075280-0325

[60] Mingyang Wang, Chang-Yu Hsieh, Jike Wang, Dong Wang, Gaoqi Weng, Chao Shen, Xiaojun Yao, Zhitong Bing, Honglin Li, Dongsheng Cao, et al. Relation: A deep generative model for structure-based de novo drug design. *Journal of Medicinal Chemistry*, 65(13):9478–9492, 2022.

[61] Kehan Wu, Yingce Xia, Yang Fan, Pan Deng, Haiguang Liu, Lijun Wu, Shufang Xie, Tong Wang, Tao Qin, and Tie-Yan Liu. Tailoring molecules for protein pockets: a transformer-based generative solution for structured-based drug design. *arXiv preprint arXiv:2209.06158*, 2022.

[62] 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*, 2021.

[63] Soojung Yang, Doyeong Hwang, Seul Lee, Seongok Ryu, and Sung Ju Hwang. Hit and lead discovery with explorative rl and fragment-based molecule generation. *Advances in Neural Information Processing Systems*, 34:7924–7936, 2021.

[64] Naruki Yoshikawa, Kei Terayama, Masato Sumita, Teruki Homma, Kenta Oono, and Koji Tsuda. Population-based de novo molecule generation, using grammatical evolution. *Chemistry Letters*, 47(11):1431–1434, 2018.

[65] Jiaxuan You, Bowen Liu, Zhitao Ying, Vijay Pande, and Jure Leskovec. Graph convolutional policy network for goal-directed molecular graph generation. *Advances in neural information processing systems*, 31, 2018.

[66] Linlin Zhao, Heather L. Ciallella, Lauren M. Aleksunes, and Hao Zhu. Advancing computer-aided drug discovery (cadd) by big data and data-driven machine learning modeling. *Drug Discovery Today*, 25(9):1624–1638, 2020.

[67] Shuangjia Zheng, Youhai Tan, Zhenyu Wang, Chengtao Li, Zhiqing Zhang, Xu Sang, Hongming Chen, and Yuedong Yang. Accelerated rational protac design via deep learning and molecular simulations. *Nature Machine Intelligence*, 4(9):739–748, 2022.

[68] Zhenpeng Zhou, Steven M. Kearnes, Li Li, Richard N. Zare, and Patrick F. Riley. Optimization of molecules via deep reinforcement learning. *Scientific Reports*, 9, 2019.

https://doi.org/10.52202/075280-0325

A GuacaMol benchmark

Table 6: More results of the experiments on the GuacaMol benchmark.

Tasks	dataset	Graph MCTS	GFlowNet	**MolRL-MGPT**
1. Celecoxib rediscovery	0.505	0.355	0.409	**1.000** $\pm$ 0.000
2. Troglitazone rediscovery	0.419	0.311	0.211	**1.000** $\pm$ 0.000
3. Thiothixene rediscovery	0.456	0.311	0.342	**1.000** $\pm$ 0.000
4. Aripiprazole similarity	0.595	0.380	0.586	**1.000** $\pm$ 0.000
5. Albuterol similarity	0.719	0.749	0.458	**1.000** $\pm$ 0.000
6. Mestranol similarity	0.629	0.402	0.396	**1.000** $\pm$ 0.000
7. $C_{11}H_{24}$	0.684	0.410	0.535	**1.000** $\pm$ 0.000
8. $C_9H_{10}N_2O_2PF_2Cl$	0.747	0.631	0.224	0.939 $\pm$ 0.003
9. Median molecules 1	0.334	0.225	0.218	0.447 $\pm$ 0.006
10. Median molecules 2	0.351	0.170	0.195	0.423 $\pm$ 0.004
11. Osimertinib MPO	0.839	0.784	0.792	0.977 $\pm$ 0.001
12. Fexofenadine MPO	0.817	0.695	0.715	**1.000** $\pm$ 0.000
13. Ranolazine MPO	0.792	0.616	0.680	**0.939** $\pm$ 0.000
14. Perindopril MPO	0.575	0.385	0.459	0.809 $\pm$ 0.005
15. Amlodipine MPO	0.696	0.533	0.430	**0.906** $\pm$ 0.001
16. Sitagliptin MPO	0.509	0.458	0.042	0.822 $\pm$ 0.003
17. Zaleplon MPO	0.547	0.488	0.072	**0.790** $\pm$ 0.008
18. Valsartan SMARTS	0.259	0.040	0.000	0.997 $\pm$ 0.000
19. deco hop	0.933	0.590	0.587	**1.000** $\pm$ 0.000
20. scaffold hop	0.738	0.478	0.475	**1.000** $\pm$ 0.000
Total	12.144	9.009	7.826	**18.049** $\pm$ 0.003

B Designing inhibitors against SARS-CoV-2 targets

Table 7: The mean scores and internal diversity of the top-100 drug candidates against the PLPro_7JIR target generated by MolRL-MGPT and other baselines.

Methods	**Docking score (↓)**	**QED score (↑)**	**SA score (↓)**	**IntDiv**
JT-VAE	-8.76 ± 0.35	0.795 ± 0.038	2.994 ± 0.140	0.836 ± 0.032
GFlowNet	-9.11 ± 0.21	0.726 ± 0.015	2.823 ± 0.076	0.825 ± 0.010
GraphGA	-10.83 ± 0.08	0.380 ± 0.013	3.638 ± 0.162	0.740 ± 0.017
Reinvent	-10.75 ± 0.05	0.392 ± 0.008	2.649 ± 0.035	0.619 ± 0.023
MolRL-MGPT	**-11.02 ± 0.06**	**0.386 ± 0.006**	**2.550 ± 0.047**	**0.745 ± 0.008**

Table 8: The mean scores and internal diversity of the top-100 drug candidates against the RdRp_6YYT target generated by MolRL-MGPT and other baselines.

Methods	**Docking score (↓)**	**QED score (↑)**	**SA score (↓)**	**IntDiv**
JT-VAE	-8.33 ± 0.25	0.719 ± 0.019	2.959 ± 0.094	0.828 ± 0.018
GFlowNet	-8.89 ± 0.16	0.656 ± 0.033	2.854 ± 0.061	0.770 ± 0.015
GraphGA	-11.26 ± 0.12	0.262 ± 0.010	3.520 ± 0.049	0.658 ± 0.009
Reinvent	-11.30 ± 0.04	0.275 ± 0.006	2.917 ± 0.035	0.616 ± 0.021
MolRL-MGPT	**-11.84 ± 0.07**	**0.278 ± 0.005**	**2.894 ± 0.072**	**0.670 ± 0.013**

https://doi.org/10.52202/075280-0325