Retrosynthetic Planning with Experience-Guided Monte Carlo Tree Search

Siqi Hong
Sun Yat-sen University
Hankz Hankui Zhuo (zhuohank@mail.sysu.edu.cn)
Sun Yat-sen University
Kebing Jin
Sun Yat-sen University
Guang Shao
Sun Yat-sen University
Zhanwen Zhou
Sun Yat-sen University

Article

Keywords:

Posted Date: March 7th, 2022

DOI: https://doi.org/10.21203/rs.3.rs-1400871/v1

Retrosynthetic Planning with Experience-Guided Monte Carlo Tree Search

Abstract

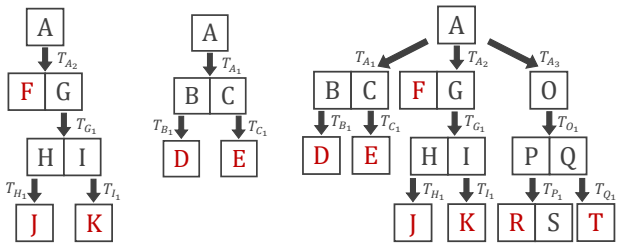
Retrosynthetic planning problem is to analyze a complex molecule and give a synthetic route using simple building blocks. The huge number of chemical reactions leads to a combinatorial explosion of possibilities, and even the experienced chemists often have difficulty to select the most promising transformations. The current approaches rely on human-defined or machine-trained score functions which have limited chemical knowledge or use expensive estimation methods such as rollout to guide the search. In this paper, we propose EG-MCTS, a novel MCTS-based retrosynthetic planning approach, to deal with retrosynthetic planning problem. Instead of exploiting rollout, we build an Experience Guidance Network to learn knowledge from synthetic experiences during the search. Experiments on benchmark USPTO datasets show that, our EG-MCTS gains significant improvement over state-of-the-art approaches both in efficiency and effectiveness. Routes designed by EG-MCTS for real drugs or compounds exhibit the effectiveness of our approach on assisting chemists performing retrosynthetic analysis. Our EG-MCTS system solves for almost a quarter more and twice times faster than the traditional computer-aided MCTS search method. In a comparative experiment with the literature, our computer-generated routes were generally viewed to be equivalent to reported literature routes by chemists.

1 Introduction

Chemical synthetic analysis, i.e., retrosynthesis, aims at designing a pathway to synthesize the target molecule using a set of available building blocks (Corey 1991). Computer-assisted approaches have been an active research topic since Corey and Wipke (1969) created the first computer program for retrosynthetic planning, after which great progress (Segler and Waller 2017; Segler, Preuss, and Waller 2018; Schreck, Coley, and Bishop 2019; Kishimoto et al. 2019; Chen et al. 2020; Gottipati et al. 2020; Wang et al. 2020; Kim et al. 2021) has been made with the development of large reaction databases (Lowe 2017). The retrosynthetic task is challenging since the search space of available reactions in each step is prohibitively large.

There have been approaches on single-step retrosynthesis, such as template-based (Coley et al. 2017b; Dai et al. 2019; Coley, Green, and Jensen 2019) and template-free (Liu et al. 2017; Zheng et al. 2020; Somnath et al. 2020; Yan et al. 2020; Lin et al. 2020; Tetko et al. 2020), which aim to predict the most promising reactions for target molecules. Different from single-step retrosynthesis, in this paper we focus on the multi-step retrosynthesis, which is more challenging since we need to consider various combinations of substantial reactions of multiple steps. There have been approaches proposed to tackle this challenge by building score functions, which are either human-defined or machine-trained, to guide the search of reactions. For example, Segler, Preuss, and Waller (2018) combined Monte Carlo Tree Search (MCTS) (Kocsis and Szepesvári 2006) with two policy networks and a filter network, called 3N-MCTS, to perform chemical synthesis planning. Jiang et al. (2019) viewed this problem as a Markov decision process and used deep reinforcement learning techniques to deal with it. DINGOS (Button et al. 2019) combined the empirical rule-based strategy with a machine learning model to produce design molecules with high similarity to the given targets. Kishimoto et al. (2019) proposed a human-defined score function to select reactions that have lowest cost based on a depth-first proof-number search. Coley et al. (2019) proposed an approach toward fully autonomous chemical synthesis that combines techniques in artificial intelligence for planning and robotics for execution. Molga, Dittwald, and Grzybowski (2019) proposed Chematica program, a commercial software platform to design synthetic pathways. Klucznik et al. (2018) executed the routes planned autonomously by Chematica in the laboratory and provided the validation of the computer approach in synthetic design. Chen et al. (2020) proposed an approach, called Retro*, to do A* search of reactions with the guidance of previously trained neural networks. Recently, Kim et al. (2021) proposed a self-improving procedure to enhance the existing approaches, such as Retro*. We call this enhanced approach Retro*+ for simplicity. Toniato et al. (2021) firstly proposed a machine learning-based, unassisted approach to clean the reaction datasets and improved the prediction quality of single-step retrosynthesis. Reinforcement learning based approaches (Schreck, Coley, and Bishop 2019; Wang et al. 2020) were also proposed to build score functions with the similarity of the retrosynthetic problems to strategy games (Szymkuć et al. 2016). The score functions were trained through self-play to evaluate the synthesis cost of molecules.

Despite the success of previous approaches, the learn-



(a) A sub-optimal route. (b) An optimal route. (c) Our approach to collect experience, performing MCTS on molecule A.

Figure 1: The sub-optimal route, optimal route and search tree of molecule A. Each box contains a molecule and every arrow is a reaction template. Those molecules marked in red are building blocks.

ing components they relied on are often based on existing single-step reaction databases (such as USPTO (Lowe 2017)), such as the three networks in 3N-MCTS (Segler, Preuss, and Waller 2018), the policy and value networks in Retro* (Chen et al. 2020). The knowledge they can acquire largely depends on the quality and quantity of the databases. More importantly, the existing databases contain only single-step reactions. Machines cannot derive multi-step information and knowledge from them, that it is difficult to build a path-level and forward-looking score function.

For example, a retrosynthetic route (with four steps) constructed by previous approach Retro (Chen et al. 2020) based on the reaction database is shown in Figure 1(a), where molecule A is decomposed into molecules F and G with template T_{A_2}, G is decomposed into H and I, H is decomposed into J, and I is decomposed into K. However, instead of T_{A_2}, there may exist another reaction template, namely T_{A_1}, which can be used to decompose molecule A in the template library and generate an optimal route (with three steps), as shown in Figure 1(b), where molecule A is decomposed into B and C with template T_{A_1}, B is decomposed into D, and C is decomposed into E. Since there is no reaction[1] decomposing A into B and C in the reaction database, such optimal route will not be constructed by Retro* (Chen et al. 2020) based on the reaction database. This results in the estimated probability for T_{A_1} being lower than T_{A_2}. Likewise, the self-improved approach Retro*+ (Kim et al. 2021) is not able to find the optimal route, since Retro*+, as a Greedy Best-First Search algorithm, prefers those templates with higher probability. Once Retro*+ finds template T_{A_2} lead to a successful route, it will improve the probability of T_{A_2}, which will further reduce the possibility of finding the optimal route with T_{A_1}.* **Based on this observation, we conjecture that leveraging all potential templates from the template library to help construct synthetic routes could be helpful for guiding the retrosynthetic planning.**

[1]Note that "reaction" is different from "reaction template". A reaction can be seen as an instance of its corresponding reaction template.

Besides, we also observe that there are many experiences that fail to construct successful route to synthesize target molecules with the building blocks during self-play. *For example, the synthetic route through molecule O and P is not a successful one in Figure 1(c), since S does not belong to the building blocks.* **Those failed experiences can be used to learn score functions for guiding retrosynthetic planning without similar failures.** Note that previous approaches, such as Retro* (Chen et al. 2020), Retro*+ (Kim et al. 2021), and those RL-based approaches (Schreck, Coley, and Bishop 2019; Wang et al. 2020), only consider successful constructed experiences when learning score functions.

Based on the above-mentioned two observations, we propose a novel MCTS-based search approach, namely `EG-MCTS`[2], standing for **E**xperience-**G**uided **M**onte **C**arlo **T**ree **S**earch, to generate routes for synthesizing target molecules.[3] We first learn an Experience Guidance Network (EGN) to estimate the score function of reaction templates by collecting retrosynthetic experiences. We then generate retrosynthetic routes for target molecules with the learnt EGN. To explore the low-probability but potentially successful reaction templates in the template library when collecting synthetic experiences, `EG-MCTS` uses MCTS to explore reaction templates and records the scores of these templates for training the score function. For example, in Figure 1(c), our `EG-MCTS` approach performs MCTS on molecule A and finds that template T_{A_1} leads to a fewer-step route during the MCTS exploration. `EG-MCTS` records the experiences about T_{A_1}. To leverage the failed experiences, we estimate the scores of reaction templates with the failed experiences along with the successful experiences. *For example, in Figure 1(c), the route through molecule O and P fails (or has not been verified) to reach a successful synthetic route. We estimate that the score of reaction template T_{P_1} is $1/2$, considering it breaks molecule P into R and S, where R belongs to the building blocks while S does not belong to the building blocks.*

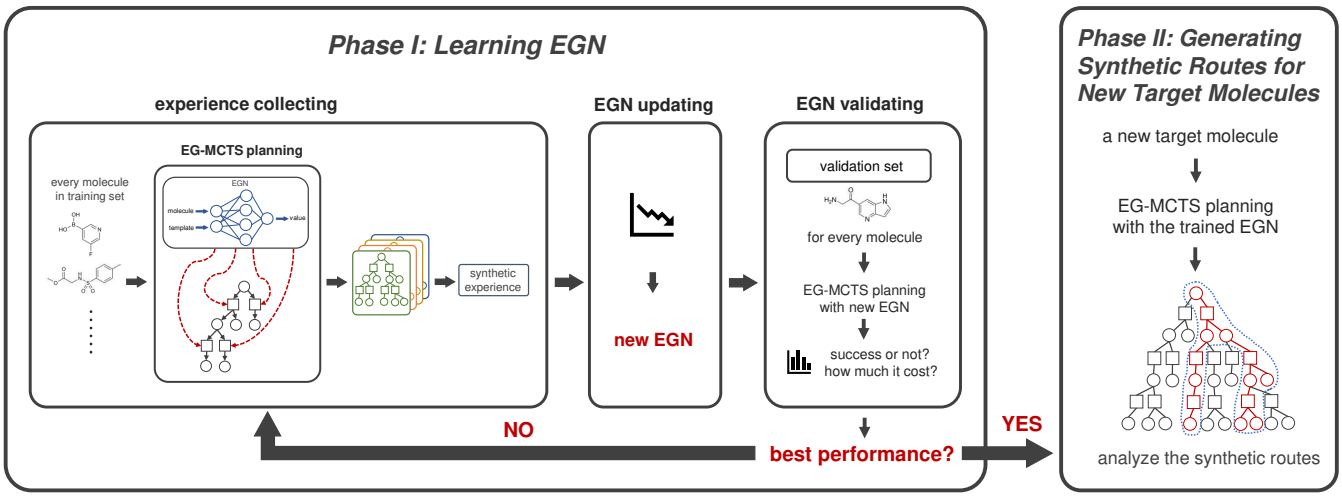
2 Results and Discussion

2.1 Formulation of Retrosynthetic Planning

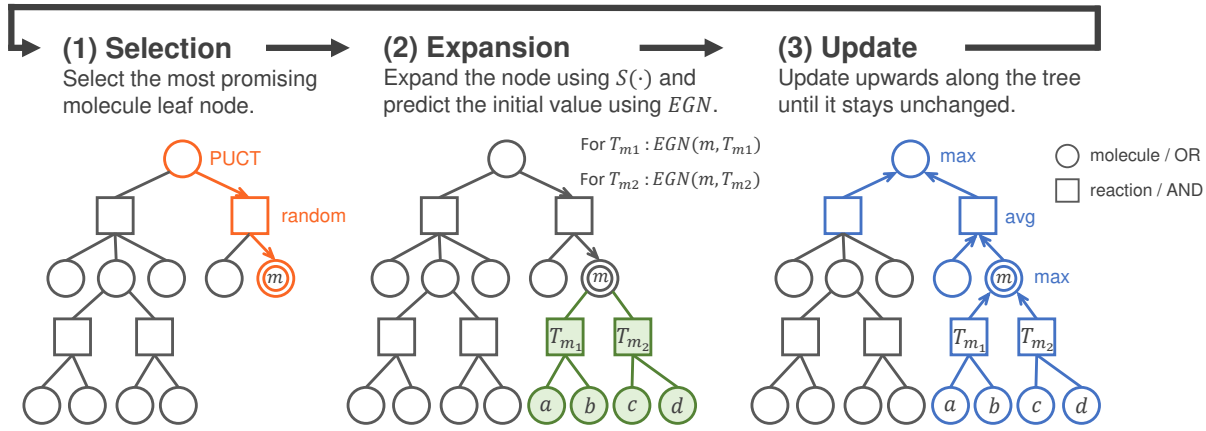
In general, the input of retrosynthetic planning, or RS planning, is composed of a target molecule m_0, a building blocks set $\mathcal{B}$, and a single-step retrosynthetic model $S(\cdot)$. $\mathcal{B}$ is composed of a set of simple, commercially available molecules. A single-step retrosynthetic model $S(\cdot)$ takes a molecule m as input, predicts k reaction templates T with the highest probability, and outputs their probabilities P as well. It can be formulated as $S(m) : \{T_j, P(m, T_j)\}_{j=1}^{k}$, where $P(m, T_j)$ indicates the probability j^{th} template T_j given molecule m. There have been off-the-shelf approaches (Coley et al. 2017b; Segler and Waller 2017; Segler, Preuss, and Waller 2018; Chen et al. 2020) developed to build this model

[2]Codes for reproducing this paper are released in supplementary material.

[3]We follow the common practice to ignore the reagents and other chemical reaction conditions.



(a) EG-MCTS approach consists of two phases, learning EGN and generating routes for new targets with the help of the well-trained EGN. In phase I, we first collect experience based on the search trees built by EG-MCST planning for every molecule in training set. Then we use these experience as the training data of EGN and update EGN. In the third part, we validate the new EGN performance by applying it on the validation set. If it reaches the best performance, the first phase stops. Otherwise, go back to the first part, experience collecting. In phase II, we use the well-trained EGN to guide the EG-MCST planning for a new target molecule and analyze the synthetic routes from the search tree.



(b) Three modules of the EG-MCTS planning procedure. Section, expansion and update are executed in a loop until the search cost is exhausted.

Figure 2: Overview of EG-MCTS approach and the procedure of the key part, EG-MCTS planning.

effectively. In this work, we borrow the single-step model developed by Kim et al. (2021). The output of RS planning is a synthetic route from $\mathcal{B}$ to m_0, i.e., a series of chemical reactions whose reactants are directly from $\mathcal{B}$ or synthesized from $\mathcal{B}$.

2.2 EG-MCTS Overview

Our EG-MCTS approach is composed of two phases, i.e., (I) learning an Experience Guidance Network (EGN) for guiding the search, and (II) generating synthetic routes for molecules with the learnt EGN (shown in Fig 2(a)).

In order to deal with the difficulty in defining a score function and the lack of path-level synthetic routes for learning, in Phase I we aim to use a network-guided MCTS planning to collect synthetic experience, and then use the experience to update the network. Monte Carlo Tree Search (MCTS) (Kocsis and Szepesvári 2006) as a general search approach, has been demonstrated successful in games, such as Go (Silver et al. 2016, 2017, 2018). The core component of MCTS is to use an "upper confidence bound" (UCB) to balance the trade-off between exploration and exploitation, such that MCTS can solve problems with large branching factors. A variant of MCTS, PUCT (Rosin 2011), has been successfully applied for RS planning (Segler, Preuss, and Waller 2018). We use a neural network instead of the traditional Rollout strategy to calculate heuristic values of searching nodes. This network, namely Experience Guidance Network (EGN), estimates a score for each template acting on each molecule as the initial evaluation value.

In Phase I shown in Fig 2(a), we first initialize the EGN

with random weights. For each target molecule in training set, we build a search tree using EG-MCTS planning with EGN and collect the synthetic experience based on the search tree as the training data of EGN. Then we update the EGN. After getting the new EGN, we verify its performance on the validation set. If it reaches the optimal performance, Phase I stops and returns the well-trained EGN. Otherwise, the Phase I will loop in the order of experience collecting, EGN updating and EGN validating.

So far we have obtained the well-trained EGN from Phase I and in Phase II, we use it to guide EG-MCTS planning. After generating the search tree for a new target molecule, we analyze the synthetic routes from the tree.

The key part, EG-MCTS planning appears in both Phase I and II, helping to collect synthetic experience and generate the synthetic routes. The search tree built by EG-MCTS planning is represented as an AND-OR tree. The OR node (molecule node) contains a molecule and the AND node (reaction node) contains a reaction template. The planning procedure can be found from Figure 2(b), which is composed of three modules, i.e., *Selection*, *Expansion* and *Update*. The *Selection* module selects the most promising molecule node m, and the *Expansion* module expands the selected node using the single-step retrosynthetic model and predicts the initial value using EGN. After that, the *Update* module updates upwards along the tree. These three molecule modules loop continuously until the search cost is exhausted. Note that "circles" and "squares" indicate molecule nodes and reaction nodes, respectively. "Double circles" indicate the molecule nodes are selected by the *Selection* module and the path marked orange shows the *Selection* process. Those nodes marked green are expanded by the *Expansion* module, and the blue path shows the *Update* process.

2.3 Experimental Details

Datasets In order to train the single-step retrosynthetic model $S(\cdot)$, we use the publicly available reaction dataset extracted from United States Patent Office (USPTO) up to September 2016 provided by Lowe (2017). The building blocks set $\mathcal{B}$ comes from *eMolecules*[4], a collection of $231M$ commercially available molecules. The single-step retrosynthetic model $S(\cdot)$ is a template-based model that treats the template prediction problem as a multi-class classification problem following previous literature (Coley et al. 2017b,a). $S(\cdot)$ is trained on the reaction dataset from USPTO with the assistance of RDChiral[5] (Coley, Green, and Jensen 2019), and the training details refer to literature (Chen et al. 2020; Kim et al. 2021).

The input of $S(\cdot)$ is a molecule, and the input of the EGN is the combination of a molecule and a reaction template. We need to represent them by real vectors. For a molecule, we use the Morgan fingerprint of radius 2 with 2048 bits. For a reaction template, its fingerprint could be computed by rdkit[6], using the function *CreateStructuralFingerprintForReaction* and the fingerprint is then folded into 2048 dimensions.

We hope the EGN to have strong generalization ability through learning the synthetic experience of molecules in training set. In order to obtain those molecules with rich and valuable experience, we build a Network of Organic Chemistry (NOC) (Fialkowski et al. 2005; Bishop, Klajn, and Grzybowski 2006; Grzybowski et al. 2009) based on USPTO and *eMolecules*. The construction details and filter process refer to Appendix. After processing, we get $1,193$ training molecules rich in experience as training set, 165 validation molecules and 180 test molecules. We also use the test set of Retro* (Chen et al. 2020) and Retro*+ (Kim et al. 2021), called Retro*-190, which consists of 190 molecules. In order to ensure the fairness and effectiveness of the experiment, we do some similarity statistical experiments: for a test molecule $m \in \mathcal{M}_{test}$, we calculate the highest similarity and the average similarity between it and the molecules in the training set, denoted as $S_{max}(m)$ and $S_{avg}(m)$. For all molecules in our test set, the average of S_{max} is 0.62 and the average of S_{avg} is 0.36. And the average of S_{max} in Retro*-190 is 0.61 and the average of S_{avg} is 0.35.

Baselines To verify the effectiveness of `EG-MCTS`, we compare our approach against other representative baselines in RS planning problem: (1) **Retro*+** and **Retro*-0+** (Kim et al. 2021) are neural-based A*-like algorithms based on Retro* (Chen et al. 2020) with a self-improved single-step retrosynthetic model. Retro*+ uses a neural value network trained in the USPTO and Retro-0*+ is its non-learning version. Its code and test set is available.[7] (2) **DFPN-E** (Kishimoto et al. 2019) combines the Depth-First Proof-Number (DFPN) Search with Heuristic Edge Initialization. Following the implementation details and parameter settings in the literature, we have implemented DFPN-E. (3) **MCTS-rollout** uses a basic tree structure whose nodes are molecule sets and edges are reaction templates and uses rollout to evaluate the values of templates. The tree structure and search algorithm can be referred to Segler, Preuss, and Waller (2018). The *max rollout depth* is 5 and the exploration constant c is 0.5. (4) **Greedy DFS** always gives priority to the reaction with the highest probability. And we set its max depth to be 10. And the node of DFS search tree is defined as a set of molecules like MCTS-rollout. To understand more about the importance of the EGN , we also perform an ablation study by testing the non-learning version (5) `EG-MCTS-0` set the initial Q value to be 0.5 for all actions.

All experiments use the same building blocks set $\mathcal{B}$. As for single-step retrosynthetic model $S(\cdot)$, all algorithms use the model of Retro*+, except Retro*-0+ (because it has its own model).

2.4 Experimental Results

We test several baseline algorithms together with our `EG-MCTS` in the test set of 180 molecules and Retro*-190. Our evaluation metrics include the efficiency of the planning and the quality of the solution routes.

[4]http://downloads.emolecules.com/free/2019-11-01/

[5]https://github.com/connorcoley/rdchiral

[6]https://www.rdkit.org/

[7]https://github.com/binghong-ml/retro_star

Algorithm	Success rate of iter limit(%)					Avg iter	Avg T nodes	Avg M nodes
	100	200	300	400	500			
EG-MCTS	**85.00**	**90.00**	**92.78**	**93.33**	**94.44**	**60.75**	**837.56**	**1133.90**
EG-MCTS-0	77.78	78.89	80.56	80.56	81.11	128.96	1411.80	1904.21
Retro*+	81.11	85.56	86.67	87.22	90.56	85.97	927.46	1396.27
Retro*-0+	80.56	82.78	86.67	86.675	89.44	87.87	1056.01	1612.05
MCTS-rollout	73.33	77.78	74.21	74.21	78.89	133.69	-	-
DFPN-E	56.11	62.22	68.89	72.22	76.67	170.34	2271.56	3012.49
Greedy DFS	45.00	48.89	50.00	51.11	54.44	268.59	-	-

Table 1: Planning efficiency performance on our test set of 180 molecules.

Algorithm	Success rate of iter limit(%)					Avg iter	Avg T nodes	Avg M nodes
	100	200	300	400	500			
EG-MCTS	**85.79**	**92.63**	**94.21**	**95.79**	**96.84**	**55.84**	**869.59**	**1193.79**
EG-MCTS-0	57.37	63.68	68.42	71.05	73.68	186.15	2525.20	3339.52
Retro*+	71.05	85.26	88.95	90.00	91.05	100.15	1209.79	1767.81
Retro*-0+	67.37	82.10	93.16	95.26	96.32	96.14	1421.90	2108.50
MCTS-rollout	43.68	47.37	54.74	58.95	62.63	254.32	-	-
DFPN-E	50.53	58.42	64.21	68.42	75.26	208.12	3123.33	4635.08
Greedy DFS	38.42	40.53	44.21	45.26	46.84	300.56	-	-

Table 2: Planning efficiency performance on the test set Retro*-190 of 190 molecules.

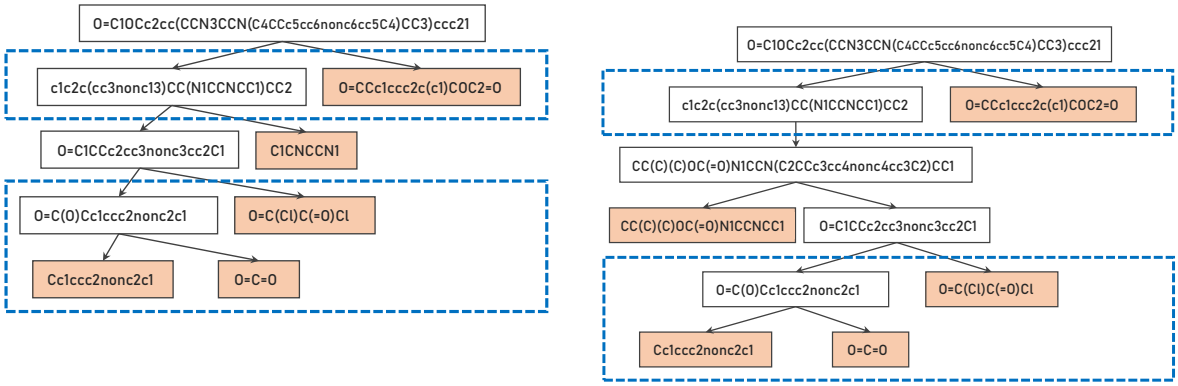
Planning Efficiency For the efficiency of planning, since the call of $S(\cdot)$ occupies most of running time, and there is always a model call in every iteration, we use the average number of iterations (*avg iter*) as a measure of time and we compare the *success rate* of all approaches under the same iteration limit, referred to others (Chen et al. 2020; Kim et al. 2021; Kishimoto et al. 2019) . We also compare the average number of molecule nodes (*avg M nodes*) and reaction nodes (*avg T nodes*) expanded by the various approaches during the searching processes.

Table 1 and Table 2 show the planning efficiency performance of all approaches on our test set and Retro*-190, respectively. The metrics *avg iter*, *avg T nodes* and *avg M nodes* are under the iteration limit of 500. With the assistance of our EGN, the performance of EG-MCTS is much better than the non-learning version in all metrics, demonstrating the performance improvement brought by our EGN. EG-MCTS is 3.88% more successful than the suboptimal approach, Retro*+ and uses 25.22 fewer iterations than Retro*+ in our test set. In Retro*-190, our EG-MCTS also has a great advantage in the metric *avg iter*. The *success rate of iter limit* of the Table 1 and Table 2 show the effect of iteration limit on the success rate of these algorithms. We can see that our EG-MCTS performs super well at the beginning on both two test sets. These phenomena indicate that our collected experience through self-play is of better quality and more instructive. The EGN can help the search to focus on more promising actions and to avoid entering a hopeless path so that accelerate the searching process.

Algorithm	Our test set			Retro*-190		
	LR	SR	Avg	LR	SR	Avg
EG-MCTS	**7**	**117**	**5.85**	**13**	**51**	**5.07**
EG-MCTS-0	90	20	8.15	20	23	5.87
Retro*+	96	12	8.37	26	24	6.03
Retro*-0+	104	10	8.48	40	24	6.25
MCTS-rollout	98	13	8.23	30	26	6.06
DFPN-E	100	15	8.31	23	17	6.00

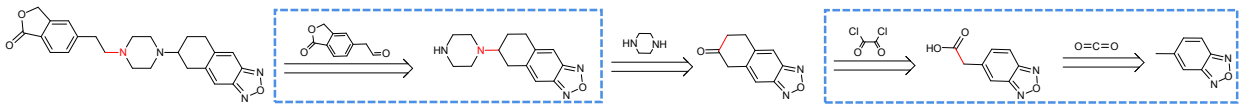
Table 3: Route quality performance on 132 molecules successfully solved on our test set and 103 molecules successfully solved on Retro*-190.

Route Quality Except Greedy DFS, there are 132 molecules successfully solved by all approaches on our test set and 103 molecules successfully solved on Retro*-190. To measure the quality of the solution routes, we compare the route length, that is the number of reactions in the route. The results are shown in Table 3. The metric *LR (longest routes)* of an approach indicates the number of longest routes generated by the approach over all of the successfully solved molecules. Specifically, for each molecule successfully solved by all approaches, if an approach generates the longest route over all approaches, *LR* of this approach is increased by one. Similarly, the metric *SR (shortest routes)* of an approach indicates the number of shortest routes generated by the approach over all of the successfully solved molecules. The metric *Avg* indicates an average of

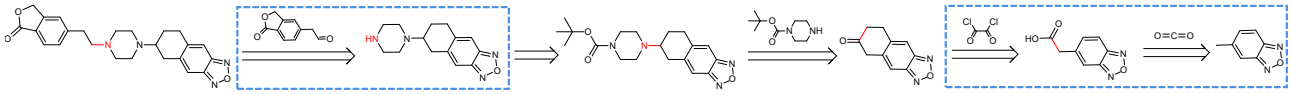


(a) Solution route given by EG−MCTS. (b) Solution route given by Retro*+.

Figure 3: Solutions given by EG−MCTS and Retro*+ for the same target (CAS NO.:1374357-00-2). Orange nodes are from $\mathcal{B}$.



(a) Chemical solution route given by our EG−MCTS.



(b) Chemical solution route given by Retro*+.

Figure 4: Chemical information of solution routes shown in Figure 3. The molecules over the arrow are from $\mathcal{B}$. The atoms and bonds marked red are reaction center, which change in the reaction.

length of all routes generated by each approach.

Our approach EG−MCTS has superior comprehensive performance among all approaches, showing the guiding role of our EGN in finding high-quality routes. Although Retro*+ and Retro*-0+ perform well in planning efficiency, but the quality of the routes they give is not so good on both two test sets. We consider the reason may be that when performing self-improvement, they simply increase the probability of those paths which have been proven successful. In our EG−MCTS, we learn a a comprehensive score for the path, so we can fully consider all potential paths.

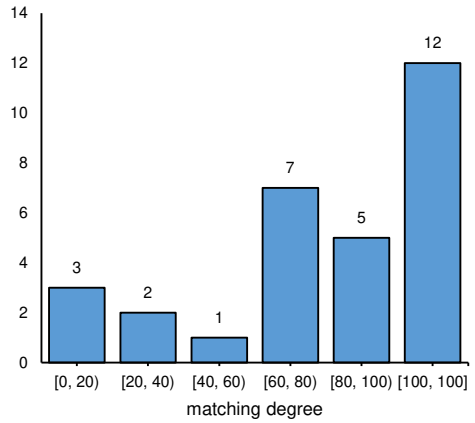
We illustrate two solution routes for the same target molecule (CAS NO.:1374357-00-2) given by our EG−MCTS and Retro*+ in Figure 3 and their chemical information in Figure 4. The dotted box parts show that two routes share the same first reaction and bottom decomposition. Our approach leaves out an extra step in the middle, selecting the better decomposition template in the second step with the help of EGN.

2.5 Case study

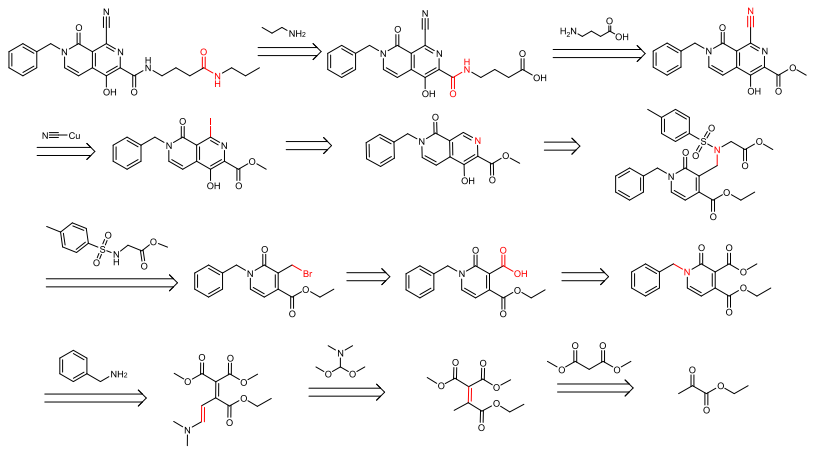
EG−MCTS Versus Literature In order to verify the validity of the routes our EG−MCTS generated, we compare the routes generated by EG−MCTS with the published routes for 30 testing molecules. The information of 30 testing molecules refer to Appendix. Similar to previous work (Chen et al. 2020; Kim et al. 2021), we set the maximal number of iterations to be 500 for each target molecule. The difference is that we will not stop the search until 500 iterations have been run out, so for each target molecule, multiple routes can be found. We only choose the route that best matches the published route. Then we calculate the matching degree between the best route and the published route for each test molecule. The calculation method of the matching degree is that if the step of the route appears in the published route, it is considered that the step is matched. Note that we only match the decomposition reactants and the main products, and don't care about the by-products. We use the number of matching steps divided by the number of steps of generated route as the matching degree. Figure 5(a) shows the statistics of the matching degree over 30 test molecules.

There are 40% of the generated routes that *almost exactly match* the published routes. Note that "almost exactly match" indicates the each step of generated routes appears in the published routes but the final molecules (buliding blocks) in the generated routes continue to be decomposed in the published routes. Figure 5(b) shows an exemplary 11-step route generated by our EG−MCTS for the molecule (CAS NO.:1392842-01-1) of inhibiting HIF hydroxylase enzyme activity reported in 2012, which fully matches the published route in the patent (Ng et al. 2012).

Another 40% of the generated routes mostly match the

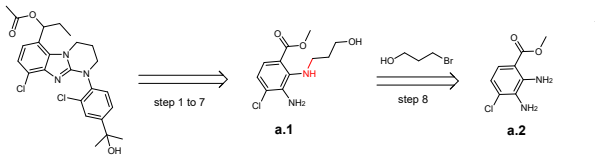


(a) The statistics of the matching degree over 30 testing molecules.

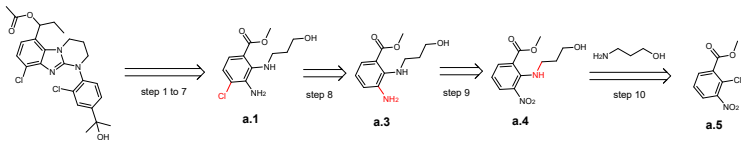


(b) An exemplary 11-step route generated by EG-MCTS for the molecule (CAS NO.:1392842-01-1) of inhibiting HIF hydroxylase enzyme activity reported in 2012, which matches the published route. The atoms and bonds marked red are reaction center, which change in the reaction.

Figure 5: (a). The statistics of the matching degree. (b). An exemplary route which matches the published route.

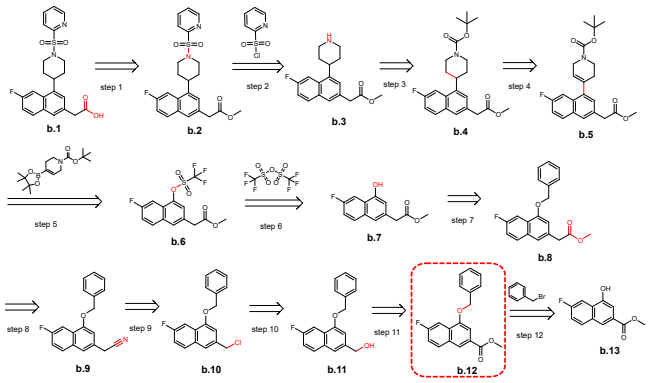


(a) Chemical solution route given by our EG-MCTS.

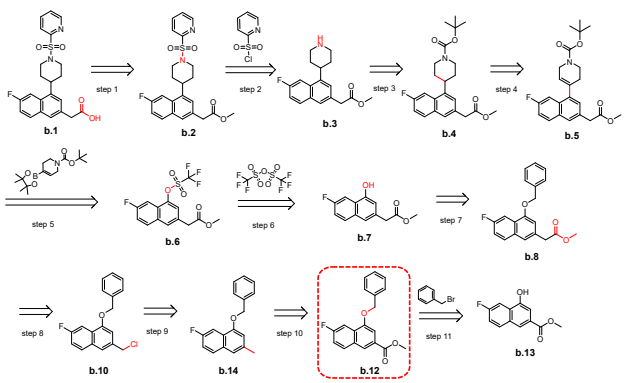


(b) Chemical published route.

Figure 6: A highly-matching example showing the differences between our EG-MCTS generated route and the published one in the patent (Aso et al. 2009). Steps 1 to 7 are the same and not shown in the figure. The intermediate molecule produced in step 7 are then decomposed in two different ways. The CAS Number of the target molecule is 1173980-10-3. The atoms and bonds marked red are reaction center, which change in the reaction.



(a) Chemical solution route given by our EG-MCTS.



(b) Chemical published route.

Figure 7: Another highly-matching example showing the differences between our EG-MCTS generated route and the published one in the patent (Firooznia et al. 2013). Steps 1 to 7 are the same. The molecules in the red dotted frame are the same intermediate molecules, but are obtained through different decomposition ways. The CAS Number of the target moleuclue is 1443043-01-3. The atoms and bonds marked red are reaction center, which change in the reaction.

published routes, with an average matching ratio of 77.23%. We observe that the difference mainly occurs in the later part of the retrosynthetic routes, while the routes are completely, especially in the first 5 to 7 steps. We also observed that there

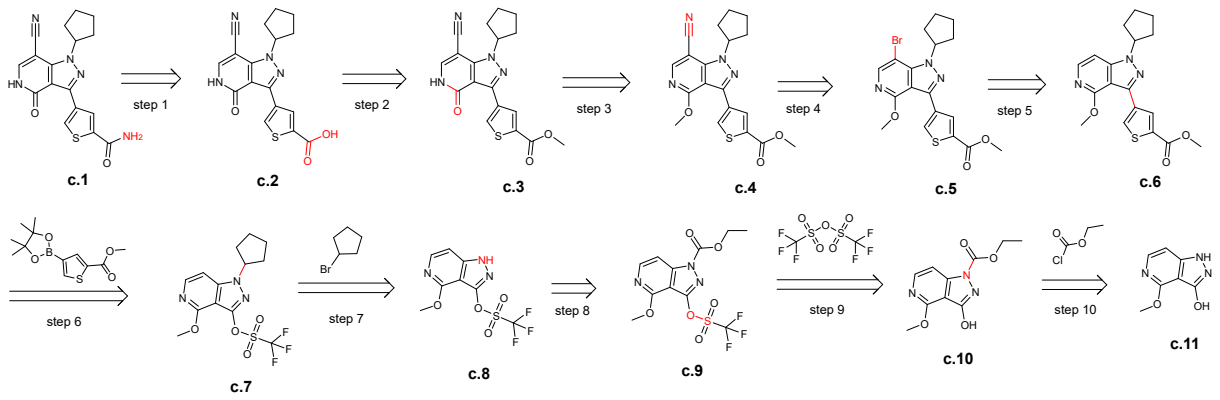


Figure 8: A lowly-matching example route given by our EG−MCTS. The atoms and bonds marked red are reaction center, which change in the reaction.

are two main differences. One is that because our EG−MCTS is goal-oriented, i.e., to break target molecules into building blocks, EG−MCTS gives priority to the successful decomposition ways which are different from the published routes, as the step 8 in Figure 6(a) compared to steps 8 to 10 in Figure 6(b). Note that identical steps 1 to 7 are not shown in the Figure 6. The route shown in Figure 6(b) is reported in the patent (Aso et al. 2009). In step 8 of EG−MCTS in Figure 6(a), compound **a.2**, methyl 4-chloro-2, 3-diaminobenzoate is reacted with 3-Bromopropyl alcohol. But it may not work since EG−MCTS chooses the less reactive amino group between the two amino groups in compound **a.2**. Another is that although the intermediate decomposition steps are different, the final decomposition results are identical. As shown in Figure 7, the generated route and the published route reported in the patent (Firooznia et al. 2013) have different intermediate steps, i.e., steps 8 to 11 in Figure 7(a) and steps 8 to 10 in Figure 7(b), but have the same intermediate decomposition compound **b.12**, which is in the red dotted frame. In the generated route, the carboxylic ester **(b.12)** is firstly reduced to the alcohol **(b.11)** in step 11, and in step 10 the alkyl halide **(b.10)** is obtained from the alcohol **(b.11)** by chlorination. These two reactions have been included in the patent (Chen et al. 2010). Step 9 is the substitution reaction of the alkyl halide **(b.10)** with cyanide reagent and produces the nitrile-containing compound **b.9**. Step 8 is the alcoholysis of nitriles to esters under the catalysis of acids. The number of steps of the generated route is one more than the published route, but each step also seems to be acceptable.

There are 6 of 30 generated routes whose matching degree is lower than 60%. Figure 8 shows a route different from the published route reported in the patent (Nara et al. 2013a). Step 10 is the acylation of acid chloride and the amine **(c.11)** to the amide **(c.10)** and step 9 is the substitution reaction of alcohol hydroxyl of compound **c.10** with trifluoromethanesulfonic anhydride to provide the trifluoromethanesulfonyl of compound **c.9**. In step 8, the amide group of compound **c.9** undergoes the amidohydrolysis. Step 7 is the substitution reaction that turns the secondary amine **(c.8)** to the tertiary amine **(c.7)**. Step 6 is the coupling of aryl compounds with arylboronic acid derivatives (Suzuki Coupling) and step 5 is the halogenation of aromatic compounds, both of which have been included in the patent (Nara et al. 2013b). The substitution reaction on alkyl halide **(c.5)** with cyanide reagent gives the nitrile-containing compound **c.4** in step 4. Compound **c.4** is then deprotected to the lactam by demethylation in step 3. The ester group of compound **c.3** is then hydrolyzed to the acid in step 2. In the last step, compound **c.2** is aminated to give the amide **(c.1)**.

Although each step of these routes follows some chemical reaction principles, some intermediate molecules of these routes may not exist in reality or have not yet been synthesized, due to the failure to consider the chemical environment. For example, the groups of the molecule itself cannot coexist and the positions and groups at which reactions can occur are various and do not definitely proceed as they do in the planning routes. After searching, we could not find the CAS number of compounds **c.2**, **c.3**, **c.4**, **c.8**, **c.9**, **c.10** appearing in the route shown in Figure 8, which means that they may not exist in reality or have not yet been synthesized. These disturbing problems are common in existing retrosynthetic planning approaches.

Drug Design We apply our EG−MCTS approach to the synthesis of some commercialized star drug molecules with complex structures to find out whether the planning synthetic routes have practical guiding significance. Here are five used molecules in drug design experiments: mannopeptimycin aglycone, Paxlovid, Sofosbuvir, Taxol and Molnupiravir.

The first drug molecule used in drug design experiments is mannopeptimycin aglycone, which is the cyclic hexapeptide aglycone of the mannopeptimycins, a group of glycopeptides known for potent activity against drug-resistant bacteria. The CAS number of mannopeptimycin aglycone is 1622135-35-6. We ignore its stereochemical structure to get the target molecule for EG−MCTS, as compound **d.1** shown in Figure 9. The generated retrosynthetic route for mannopeptimycin aglycone is shown in Figure 9. Note that our EG−MCTS sometimes ignores some side intermediates. We add the necessary intermediates to the route and mark

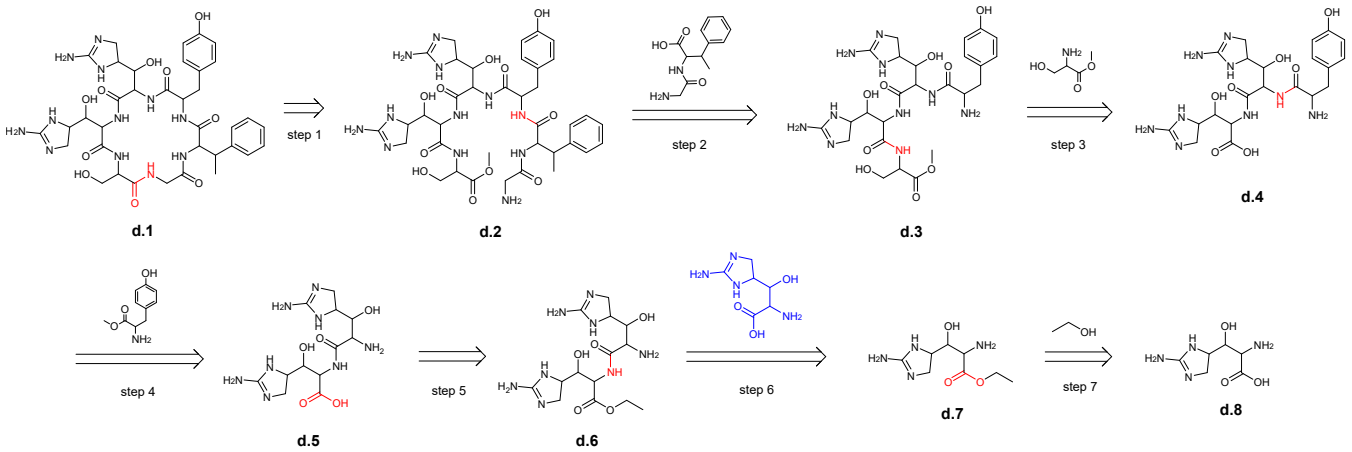
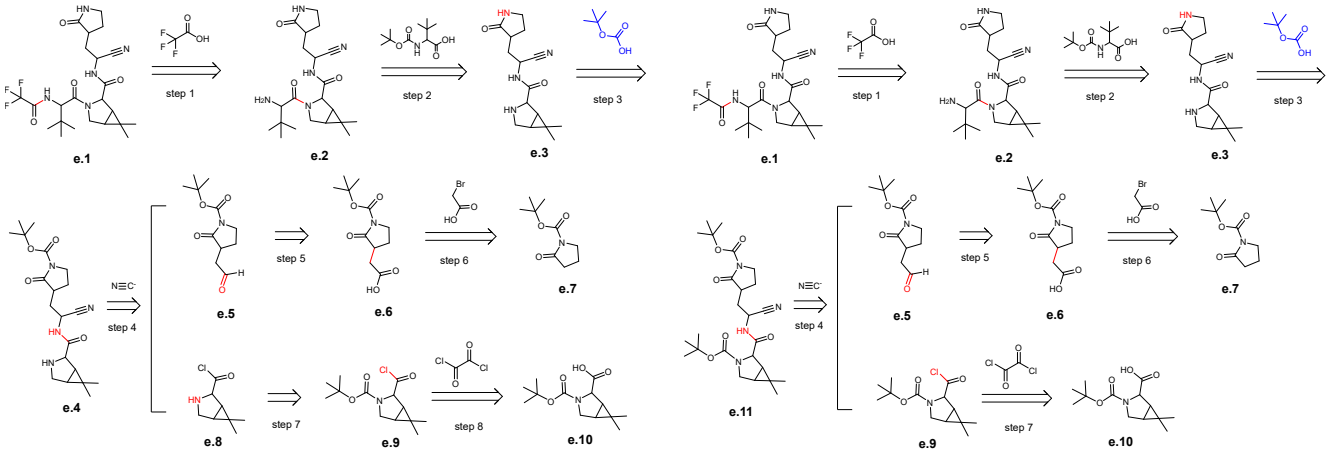


Figure 9: The generated route given by our EG-MCTS for mannopeptimycin aglycone. The CAS number of mannopeptimycin aglycone is 1622135-35-6 and we ignore its stereochemical structure. The atoms and bonds marked red are reaction center, which change in the reaction. We add the necessary intermediates to the route and mark them in blue.



(a) The generated route given by our EG-MCTS for Paxlovid.

(b) The adjusted route.

Figure 10: The generated route given by our EG-MCTS for Paxlovid. The CAS number of Paxlovid is 2628280-40-8 and we ignore its stereochemical structure. The atoms and bonds marked red are reaction center, which change in the reaction. We add the necessary intermediates to the route and mark them in blue.

them in blue.

The designed route starts from the esterification of compound **d.8** with ethanol (step 7). In step 6, the amine group of compound **d.7** undergoes the condensation reaction with the carboxyl group of aid (the blue compound) to form the amid **(d.6)**. Step 5 is the hydrolysis of the ester into the carboxyl. Compound **d.5** then undergoes the acylation reaction with Methyl 2-amino-3-(4-hydroxyphenyl)propanoate in step 4, followed by two successive condensation reactions of carboxyl and amino groups in steps 3 and 2. The last step is the intramolecular acylation reaction, in which the amine group and the ester group of compound **d.2** are involved to form a hexapeptide ring.

The second molecule used in drug design experiments is Paxlovid, which is the first oral antiviral drug authorized by the FDA for the treatment of COVID-19. The CAS number of Paxlovid is 2628280-40-8. We also ignore its stereochemical structure and use our EG-MCTS to get its retrosynthetic route as shown in Figure. Note that we also add the necessary intermediates to the route and mark them in blue.

The generated route starts with two building blocks, compound **e.7** and **e.10**. The hydroxyl group of compound **e.10** undergoes the esterification reaction with oxalyl chloride in step 8 and the amide **(e.9)** is then hydrolyzed in step 7. On the other side, compound **e.7** first reacts with bromoacetic acid to produce the acid derivative **(e.6)** in step 6 and then the carboxylic acid **(e.6)** is reduced to the aldehyde **(e.5)** in step 5. In step 4, compounds **e.5**, **e.8** and cyanide participate in a ternary reaction to get the compound **e.4** with the cyano group and the amide group. Step 3 is the amide hydrolysis and generates compound **e.3**. In step 2, a condensation reaction occurs between compound **e.3** and the compound,

which is above the "step 2" arrow, and the amide hydrolysis occurs at the same time, resulting in compound **e.2**. The last step is also the condensation reaction of carboxyl group and amino group, happening between compound **e.2** and trifluoroacetic acid.

The generated routes for the other three drugs are listed in Appendix. From the two routes, even ignoring the stereochemical structure and some reactants, our generated routes are definitely not perfect. There are many details to be perfected, such as whether the presence of intermediate compounds is reasonable and whether the reactions will go as planned. For a specific example, the structural stability of compound **e.8** in the Figure 10(a) is questionable, as acylation may occur between the amine group and the acid chloride inside compound **e.8**. Although the generated routes given by our EG-MCTS are not mature enough, but they are heuristic for synthetic organic chemists while performing retrosynthesis for complex compounds and can guide them in which direction to consider. It would be even more helpful if chemists could adjust the generated routes according to these imperfect and inaccurate details and finally get a relatively feasible path. For example, for the detail of the structural instability of compound **e.8**, we can make minor adjustments to the generated route as shown in Figure 10(b). In the adjusted route, we use compound **e.9** instead of compound **e.8** to participate in the ternary reaction with compound **e.5** and cyanide to generate new compound **e.11** (step 3). The two amide groups of compound **e.11** are then hydrolyzed at the same time in step 3, discarding the two tert-butyl hydrogen carbonate. Small adjustments like this make the resulting routes more reasonable.

2.6 Conclusion

In this paper, we propose EG-MCTS, a novel MCTS-based retrosynthetic planning approach. Different from existing machine-trained approaches which are limited to the existing datasets, we investigate the way of acquiring chemical synthetic knowledge and experience. Our experimental results on real-world benchmark datasets exhibit our EG-MCTS gains significant improvement over existing approaches. The comparison between the generated routes and the published routes also confirms the validity and feasibility of our approach. We use our EG-MCTS to perform retrosynthetic planning for realistic drugs or compounds, and the results exhibit that EG-MCTS is instructive. At the same time, the experiments on real compounds have exposed the inadequacies of our approach, which are also common problems of retrosynthetic planning approaches, that is, the understanding and learning of chemical reaction principles are not thorough and comprehensive. It can be embodied as whether the presence of compounds is reasonable and whether the reactions will go as planned and so on. We believe that if these problems are solved, the quality of the generated routes can be greatly improved.

In planning community, there have been techniques of high-performance with respect to planning and learning Zhuo and Kambhampati (2017); Zhuo and Yang (2014); Zhuo, Muñoz-Avila, and Yang (2014); Zhuo et al. (2010); Shen et al. (2020). It would be interesting to investigate "borrowing" those techniques to deal with the retrosynthetic planning problem in the future.

In general, EG-MCTS can be applied to those planning problems which need a forward score function to guide the search but lack path-level datasets to learn the score function of high-quality. We will consider this as our future work.

3 Method

We first describe the RS planning problem as a Markov Decision Process. Then we introduce the key part, EG-MCTS planning. Finally, we describe the two phases of EG-MCTS approach in detail.

3.1 Retrosynthetic Planning Problem

RS planning can be viewed as a Markov Decision Process (MDP) (Sutton and Barto 2018), defined by a state space $\mathcal{S}$, an action space $\mathcal{A}(s)$, a transition model $\mathcal{T}(s, a, s')$, a policy $\pi(a|s)$ and a reward function $\mathcal{R}(s, a, s')$. In RS planning, a state $s \in \mathcal{S}$ is a set of molecules, and the initial state $s_0 = m_0$ is composed of the target molecule m_0. Actions are reaction templates applied to one of the molecules m in state s. The transition function $\mathcal{T}(s, a, s')$ is deterministic for simplicity. The policy $\pi(a|s)$ is the probability distribution of all allowed functions. The reward function $\mathcal{R}(s, a, s')$ can be simplified as $\mathcal{R}(m, T)$, indicating the reward taken by applying reaction template T on molecule m.

3.2 EG-MCTS Planning

We first introduce the key part, EG-MCTS Planning. We observe that AND-OR tree structure is suitable for RS planning (Heifets and Jurisica 2012; Kishimoto et al. 2019; Chen et al. 2020; Kim et al. 2021), capturing the relations between reactions and corresponding molecules. The result of EG-MCTS planning can be represented as an AND-OR tree.

An AND-OR tree has two different types of nodes, i.e., AND node that succeeds only if all of its child nodes are successful, and OR node that succeeds only if at least one child node is successful. In RS planning, a molecule is viewed as *successful* if there exists at least one reaction that can break it down to $\mathcal{B}$. A reaction is viewed as *successful* if all of its reactants are successful. The retrosynthetic searching process can be represented as an AND-OR tree, whose OR and AND nodes are molecules and reaction templates, respectively. Note that a reaction template can be seen as a reaction relation among substructures of reactants and products. For example, "$\bar{x} \rightarrow \bar{a} + \bar{b}$" is a reaction template, where $\bar{x}$, $\bar{a}$, $\bar{b}$ are substructures of molecules x, a and b in reaction "$x \rightarrow a + b$", respectively. In EG-MCTS planning, the OR node (molecule node) contains a molecule and a value V_m, and the AND node (reaction node) contains a reaction template and a value $\bar{Q}$. We denote a molecule node m as *successful* if its molecule belongs to $\mathcal{B}$ or one of its child reaction nodes is denoted as *successful*. We denote a molecule node as *unsuccessful* if all of its child nodes are denoted as *unsuccessful* or there is no reaction template available to be applied to m. Likewise, we denote a reaction node T as *successful* if all of its child molecule nodes are denoted as

successful, and denote it as *unsuccessful* if one of its child nodes is denoted as *unsuccessful*.

We address the three modules of EG-MCST planning in detail below.

Selection In order to select a promising molecule node, we need to build a selection module to repeatedly select reaction templates for molecule nodes and (sub-)molecules for reaction nodes, until a leaf molecule node is found. Intuitively, for a molecule node, we select the most promising reaction templates based on the PUCT policy as used by (Rosin 2011), as shown in Equation (1):

$$T^* = \underset{T \in child(m)}{\arg\max} \left(\frac{\bar{Q}(m,T)}{N(T)} + cP(m,T)\frac{\sqrt{N(T')}}{1+N(T)} \right) \tag{1}$$

In Equation (1), $\bar{Q}(m,T)$ is an average score over all previous scores, which will be repeatedly updated according Equation (3) given by the *Update* module. $P(m,T)$ is given by the single-step retrosynthetic model $S(\cdot)$, and $N(T)$ records the number of times that node T has been updated. T' is the grandparent reaction node of the reaction node T. The exploration constant c is a hyper-parameter.

For a reaction node, if it has child nodes which have not been expanded, the algorithm will give priority to this kind of child nodes and randomly choose one. Otherwise, randomly select one among the child nodes which have not been proved successful.

Expansion The single-step retrosynthetic model $S(\cdot)$ is applied to the molecule m contained in the selected molecule node, and it predicts the top-k promising reaction templates. If the output set is empty, indicating no available reaction templates, the node is *unsuccessful*. Otherwise, each reaction template T_j is added to the tree as a child reaction node of the selected molecule node with $\bar{Q}(m,T_j) = Q_0(m,T_j)$ given by the EGN. After applying the template T_j on m, we get the corresponding reactant set R_j. Each reactant r in R_j is also added as a child molecule node of the reaction node T_j.

Update The update step starts from the selected molecule node and upwards along the tree. At the molecule node, the algorithm checks whether the node is *successful* or *unsuccessful*. If it is not proved to be *unsuccessful*, the algorithm updates its V_m to the highest $\bar{Q}$ among its child nodes:

$$V_m(m) = \underset{T \in child(m)}{max} \bar{Q}(m,T) \tag{2}$$

At the reaction node, the algorithm firstly updates its update count $N(T) = N(T) + 1$. Then the algorithm records its Q value in the $N(T)^{th}$ update, denoted as $Q_{N(T)}(m,T)$. $Q_{N(T)}(m,T)$ is given by the reward function $\mathcal{R}(m,T)$. The reward function returns $z > 1$ if the reaction node is proved to be *successful*, and $-z$ if it is *unsuccessful*. Otherwise, the reward function calculates the average V_m among its child nodes. After getting the reward in the $N(T)^{th}$ update, the algorithm updates the average score $\bar{Q}$ of the reaction node:

$$\bar{Q}(m,T) = \frac{1}{N(T)+1} \sum_{j=0}^{N(T)} Q_j(m,T) \tag{3}$$

Note that $Q_0(m,T)$ is given by EGN when the reaction node T is added to the tree, which is not counted in its update count, and $Q_j(m,T), j \in [1, N(T)]$ is given by the reward function.

Algorithm 1: Learning EGN

Input: Training molecule set $\mathcal{M}_{train}$, validation molecule set $\mathcal{M}_{validation}$, building blocks set $\mathcal{B}$, one-step retrosynthetic model $S(\cdot)$

Output: well-trained EGN f_θ

1: Initialize EGN f_{θ_0} with random parameters θ_0;
2: **for** i=1, max_round **do**
3: Initialize a training data $\mathcal{D}^i_{train} = \{\}$;
4: **for** $m \in \mathcal{M}_{train}$ **do**
5: Build search tree $\mathcal{T}_m$: $\mathcal{T}_m = EG\text{-}MCTS\text{-}planning(m, \mathcal{B}, S(\cdot), f_{\theta_{i-1}})$;
6: Collect training data $\mathcal{D}$ from $\mathcal{T}_m$: $\mathcal{D} = experience\text{-}collecting(\mathcal{T}_m)$;
7: $\mathcal{D}^i_{train} = \mathcal{D}^i_{train} \cup \mathcal{D}$;
8: **end for**
9: Update EGN with $\mathcal{D}^i_{train}$: $f_{\theta_i} = EGN\text{-}updating(f_{\theta_{i-1}}, \mathcal{D}^i_{train})$;
10: **for** $m \in \mathcal{M}_{validation}$ **do**
11: search m using EG-MCTS-planning: $EG\text{-}MCTS\text{-}planning(m, \mathcal{B}, S(\cdot), f_{\theta_i})$;
12: **end for**
13: Complete the success rate $\mathcal{R}_{s_i}$ and the average number of iterations $\mathcal{R}_{a_i}$ on $\mathcal{M}_{validation}$;
14: Complete the highest success rate $\mathcal{R}_{s_{max}}$ and the lowest average number of iterations $\mathcal{R}_{a_{min}}$ of the last five rounds on $\mathcal{M}_{validation}$;
15: **if** $\mathcal{L}_{loop}(\mathcal{R}_{s_i}, \mathcal{R}_{s_{max}}, \mathcal{R}_{a_i}, \mathcal{R}_{a_{min}})$ is false **then**
16: **return** f_{θ_i} ;
17: **end if**
18: **end for**

3.3 Phase I: Learning EGN

The detailed learning procedure can be found from Algorithm 1. We first initialize the EGN with random weights θ_0, which is denoted by f_{θ_0}. At each training round $i \geq 1$, for each target molecule $m \in \mathcal{M}_{train}$, we build a search tree $\mathcal{T}_m$ using EG-MCTS planning with $f_{\theta_{i-1}}$ (Step 5). We then collect the training data based on $\mathcal{T}_m$ (Step 6). After that we update the EGN with the training data and get the new EGN f_{θ_i}(Step 9). We verify the performance of the new EGN on the validation molecule set, i.e., perform EG-MCTS-planning for each molecule $m \in \mathcal{M}_{validation}$(Step 11). If the success rate and average number of iterations can not satisfy the loop condition $\mathcal{L}_{loop}$, then the learning algorithm stops and return the well-trained EGN. In the following subsections, we will address three procedures *experience-collecting*, *EGN-updating* and *EGN validating* of Algorithm 1, respectively.

Experience Collecting The Experience Guidance Network learns from chemical synthetic experience and uses experience to guide the future search. It takes a reaction template T and a molecule m as inputs, then predicts the score

of template T acting on molecule m. It works based on the following assumptions:

- The score of a reaction template acting on a molecule is independent of others, so independent prediction is reasonable.
- The same decomposition action (m, T) may appear in the search of different target molecules, so EGN, which has learned the value of action (m, T) from past searching, will give a more accurate value while meeting the same action.
- The most potential reaction templates of two similar compounds are likely to be the same. The well-trained network which has learned from the past synthetic experience showing that the reaction template T works well in molecule m will encourage the search to select T when similar molecule m' is encountered.

We hope that the learned network could be universally applied to guide any searches, even for molecules that have never been seen before. Specifically, in the i^{th} round of training of the EGN, for every molecule m in the training set $\mathcal{M}_{train}$, EG-MCTS planning builds a search tree $\mathcal{T}_m$. For every reaction node T in the tree $\mathcal{T}_m$, it and its parent molecule node m composes a decomposition action (m, T). We collect every decomposition action (m, T) and the $\bar{Q}$ stored in the corresponding reaction node T to form the experience set $\mathcal{D}^i_{train} = \{(m, T), \bar{Q}\}$.

EGN Updating The EGN is a single-layer fully connected neural network with input dimension of 4096 and hidden dimension of 256. It outputs a scalar $Q \in (0, 1)$ representing the predicted value. At training round i, the neural network $Q = f_{\theta_{i-1}}(m, T)$ is trained for 20 epochs on dataset $\mathcal{D}^i_{train}$ to minimize $\mathcal{L}_{MSE}$, using Adam optimizer (Kingma and Ba 2015). We apply dropout (Srivastava et al. 2014) as a means of regularization with the dropout rate 0.1.

$$\mathcal{L}_{MSE} = \left(Q - \bar{Q}(m, T)\right)^2 \tag{4}$$

EGN Validating We then verify the new EGN f_{θ_i} on the validation set. Specifically, the algorithm records the highest success rate $\mathcal{R}_{s_{max}}$ and the lowest average number of iterations $\mathcal{R}_{a_{min}}$ of the last five training round on the validation set. At the training round i, the algorithm completes the success rate $\mathcal{R}_{s_i}$ and the average number of iterations $\mathcal{R}_{a_i}$ of EG-MCTS planning with f_{θ_i} on the validation set. The loop condition $\mathcal{L}_{loop}$ can be expressed as: $\mathcal{L}_{loop}$ is true if $\mathcal{R}_{s_i} - \mathcal{R}_{s_{max}} > \varepsilon_1$ or $\mathcal{R}_{a_{min}} - \mathcal{R}_{a_i} > \varepsilon_2$. Otherwise, it is false. ε_1 and ε_2 are hyper-parameters.

3.4 Phase II: Generating Synthetic Routes for New Target Molecules

To generate synthetic routes for the target molecule m_0, we first exploit the *EG-MCTS-planning* procedure, i.e., Step 5 of Algorithm 1, to generate a tree with the learnt EGN f_θ:

$$EG\text{-}MCTS\text{-}planning(m_0, \mathcal{B}, S(\cdot), f_\theta).$$

We then initialize a queue with the root node of the tree and an empty reaction list. The following process is repeated until the queue is empty:

- We get the first node m from the queue.
- If m is not from $\mathcal{B}$ and it has a successful child reaction node T, we put all children $\{r_j\}_{j=1}^n$ of this reaction node T into the queue and add the reaction $m \rightarrow \{r_j\}_{j=1}^n$ to the reaction list. If it is not from $\mathcal{B}$ and it does not have a successful child reaction node, the search fails and the reaction list is set to empty.
- If the queue is empty, the search succeeds and the algorithm returns the reaction list.

With the above process, we have the reaction list as the synthetic route of a target molecule.

Note that in our experiment, we empirically set the exploration constant c to be 0.5, the reward z to be 10 for a successful reaction node and -10 for a failed reaction node, respectively. We set ε_1 of the loop condition Θ to be 0.015, and ε_2 to be 3, respectively.

Data and code availability

The data used in the experiment and the source code of `EG-MCTS` are both available at https://github.com/jjljkjljk/EG-MCTS.

References

Aso, K.; Kobayashi, T., Katsumi; Takai; Kojima, T.; Tokumaru, K.; Mochizuki, M.; and Hoashi, Y. 2009. Preparation of tricyclic compounds as CRF receptor antagonists.

Bishop, K. J.; Klajn, R.; and Grzybowski, B. A. 2006. The Core and Most Useful Molecules in Organic Chemistry. *Angewandte Chemie International Edition*, 45(32): 5348–5354.

Button, A.; Merk, D.; Hiss, J. A.; and Schneider, G. 2019. Automated de novo molecular design by hybrid machine intelligence and rule-driven chemical synthesis. *Nature machine intelligence*, 1(7): 307–315.

Chen, B.; Li, C.; Dai, H.; and Song, L. 2020. Retro*: Learning Retrosynthetic Planning with Neural Guided A* Search. In III, H. D.; and Singh, A., eds., *Proceedings of the 37th International Conference on Machine Learning*, volume 119, 1608–1616. PMLR.

Chen, L.; Firooznia, F.; Gillespie, P.; He, Y.; Lin, T.-A.; Mertz, E.; So, S.-S.; Yun, H.; and Zhang, Z. 2010. Preparation of naphthylacetic acids as antagonists or partial agonists at the CRTH2 receptor.

Coley, C. W.; Barzilay, R.; Jaakkola, T. S.; Green, W. H.; and Jensen, K. F. 2017a. Prediction of Organic Reaction Outcomes Using Machine Learning. *ACS central science*, 3(5): 434–443.

Coley, C. W.; Green, W. H.; and Jensen, K. F. 2019. RDChiral: An RDKit Wrapper for Handling Stereochemistry in Retrosynthetic Template Extraction and Application. *Journal of Chemical Information and Modeling*, 59(6): 2529–2537.

Coley, C. W.; Rogers, L.; Green, W. H.; and Jensen, K. F. 2017b. Computer-Assisted Retrosynthesis Based on Molecular Similarity. *ACS central science*, 3(12): 1237–1245.

Coley, C. W.; Thomas, D. A.; Lummiss, J. A. M.; Jaworski, J. N.; Breen, C. P.; Schultz, V.; Hart, T.; Fishman, J. S.; Rogers, L.; Gao, H.; Hicklin, R. W.; Plehiers, P.; Byington, J.; Piotti, J. S.; Green, W. H.; Hart, A. J.; Jamison, T. F.; and Jensen, K. F. 2019. A robotic platform for flow synthesis of organic compounds informed by AI planning. *SCIENCE*, 365(6453): 11.

Corey, E. J. 1991. The Logic of Chemical Synthesis: Multistep Synthesis of Complex Carbogenic Molecules (Nobel Lecture). *Angewandte Chemie International Edition in English*, 30(5): 455–465.

Corey, E. J.; and Wipke, W. T. 1969. Computer-Assisted Design of Complex Organic Syntheses. *Science*, 166(3902): 178–192.

Dai, H.; Li, C.; Coley, C. W.; Dai, B.; and Song, L. 2019. Retrosynthesis Prediction with Conditional Graph Logic Network. In Wallach, H. M.; Larochelle, H.; Beygelzimer, A.; d'Alché-Buc, F.; Fox, E. B.; and Garnett, R., eds., *Advances in Neural Information Processing Systems 32*, volume 32, 8870–8880. Curran Associates, Inc.

Fialkowski, M.; Bishop, K. J.; Chubukov, V. A.; Campbell, C. J.; and Grzybowski, B. A. 2005. Architecture and Evolution of Organic Chemistry. *Angewandte Chemie International Edition*, 117(44): 7429–7435.

Firooznia, F.; Lin, T.-A.; Mertz, E.; Sidduri, A.; So, S.-S.; and Tilley, J. W. 2013. Preparation of piperidinyl naphthylacetic acids as antagonists or partial agonists at the CRTH2 receptor.

Gottipati, S. K.; Sattarov, B.; Niu, S.; Pathak, Y.; Wei, H.; Liu, S.; Blackburn, S.; Thomas, K. M. J.; Coley, C. W.; Tang, J.; Chandar, S.; and Bengio, Y. 2020. Learning to Navigate The Synthetically Accessible Chemical Space Using Reinforcement Learning. In III, H. D.; and Singh, A., eds., *Proceedings of the 37th International Conference on Machine Learning*, volume 119, 3668–3679. PMLR.

Grzybowski, B. A.; Bishop, K. J.; Kowalczyk, B.; and Wilmer, C. E. 2009. The wired universe of organic chemistry. *Nature Chemistry*, 1(1): 31–36.

Heifets, A.; and Jurisica, I. 2012. I.: Construction of new medicines via game proof search.

Jiang, P.; Doan, H.; Madireddy, S.; Assary, R. S.; and Balaprakash, P. 2019. Value-Added Chemical Discovery Using Reinforcement Learning. *CoRR*, abs/1911.07630.

Kim, J.; Ahn, S.; Lee, H.; and Shin, J. 2021. Self-Improved Retrosynthetic Planning. In Meila, M.; and Zhang, T., eds., *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, 5486–5495. PMLR.

Kingma, D. P.; and Ba, J. 2015. Adam: A Method for Stochastic Optimization. In Bengio, Y.; and LeCun, Y., eds., *3rd International Conference on Learning Representations*.

Kishimoto, A.; Buesser, B.; Chen, B.; and Botea, A. 2019. Depth-First Proof-Number Search with Heuristic Edge Cost and Application to Chemical Synthesis Planning. In Wallach, H.; Larochelle, H.; Beygelzimer, A.; d'Alché-Buc, F.; Fox, E.; and Garnett, R., eds., *Advances in Neural Information Processing Systems 32*, volume 32, 7224–7234. Curran Associates, Inc.

Klucznik, T.; Mikulak-Klucznik, B.; McCormack, M. P.; Lima, H.; Szymkuć, S.; Bhowmick, M.; Molga, K.; Zhou, Y.; Rickershauser, L.; Gajewska, E. P.; Toutchkine, A.; Dittwald, P.; Startek, M. P.; Kirkovits, G. J.; Roszak, R.; Adamski, A.; Sieredzińska, B.; Mrksich, M.; Trice, S. L.; and Grzybowski, B. A. 2018. Efficient Syntheses of Diverse, Medicinally Relevant Targets Planned by Computer and Executed in the Laboratory. *Chem*, 4(3): 522–532.

Kocsis, L.; and Szepesvári, C. 2006. Bandit Based Monte-Carlo Planning. In Fürnkranz, J.; Scheffer, T.; and Spiliopoulou, M., eds., *17th European Conference on Machine Learning*, volume 4212, 282–293. Springer.

Lin, K.; Xu, Y.; Pei, J.; and Lai, L. 2020. Automatic retrosynthetic route planning using template-free models. *Chem. Sci.*, 11: 3355–3364.

Liu, B.; Ramsundar, B.; Kawthekar, P.; Shi, J.; Gomes, J.; Nguyen, Q. L.; Ho, S.; Sloane, J.; Wender, P.; and Pande, V. S. 2017. Retrosynthetic Reaction Prediction Using Neural Sequence-to-Sequence Models. *ACS Central Science*, 3(10): 1103–1113.

Lowe, D. 2017. Chemical reactions from US patents (1976-Sep2016).

Molga, K.; Dittwald, P.; and Grzybowski, B. A. 2019. Navigating around Patented Routes by Preserving Specific Motifs along Computer-Planned Retrosynthetic Pathways. *Chem*, 5(2): 460–473.

Nara, H.; Daini, M.; Kaieda, A.; Kamei, T.; Imaeda, T.; and Kikuchi, F. 2013a. Preparation of fused dihydropyridinone derivatives as janus kinase (JAK) inhibitors.

Nara, H.; Daini, M.; Kaieda, A.; Kamei, T.; Imaeda, T.; and Kikuchi, F. 2013b. Preparation of fused dihydropyridinone derivatives as janus kinase (JAK) inhibitors.

Ng, D.; Arend, M. P.; Flippin; and A, L. 2012. Naphthyridine derivatives as inhibitors of hypoxia inducible factor (HIF) hydroxylase and their preparation and use for the treatment of HIF-mediated diseases.

Rosin, C. D. 2011. Multi-armed Bandits with Episode Context. *Annals of Mathematics and Artificial Intelligence*, 61(3): 203–230.

Schreck, J. S.; Coley, C. W.; and Bishop, K. J. 2019. Learning retrosynthetic planning through simulated experience. *ACS central science*, 5(6): 970–981.

Segler, M. H. S.; Preuss, M.; and Waller, M. P. 2018. Planning chemical syntheses with deep neural networks and symbolic AI. *Nature*, 555(7698): 604–610.

Segler, M. H. S.; and Waller, M. P. 2017. Neural-Symbolic Machine Learning for Retrosynthesis and Reaction Prediction. *Chemistry – A European Journal*, 23(25): 5966–5971.

Shen, J.; Zhuo, H. H.; Xu, J.; Zhong, B.; and Pan, S. J. 2020. Transfer Value Iteration Networks. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium*

on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020, 5676–5683. AAAI Press.

Silver, D.; Huang, A.; Maddison, C. J.; Guez, A.; Sifre, L.; van den Driessche, G.; Schrittwieser, J.; Antonoglou, I.; Panneershelvam, V.; Lanctot, M.; Dieleman, S.; Grewe, D.; Nham, J.; Kalchbrenner, N.; Sutskever, I.; Lillicrap, T. P.; Leach, M.; Kavukcuoglu, K.; Graepel, T.; and Hassabis, D. 2016. Mastering the game of Go with deep neural networks and tree search. *Nature*, 529(7587): 484–489.

Silver, D.; Hubert, T.; Schrittwieser, J.; Antonoglou, I.; Lai, M.; Guez, A.; Lanctot, M.; Sifre, L.; Kumaran, D.; Graepel, T.; Lillicrap, T.; Simonyan, K.; and Hassabis, D. 2018. A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play. *Science*, 362(6419): 1140–1144.

Silver, D.; Schrittwieser, J.; Simonyan, K.; Antonoglou, I.; Huang, A.; Guez, A.; Hubert, T.; Baker, L.; Lai, M.; Bolton, A.; Chen, Y.; Lillicrap, T. P.; Hui, F.; Sifre, L.; van den Driessche, G.; Graepel, T.; and Hassabis, D. 2017. Mastering the game of Go without human knowledge. *Nature*, 550(7676): 354–359.

Somnath, V. R.; Bunne, C.; Coley, C. W.; Krause, A.; and Barzilay, R. 2020. Learning Graph Models for Template-Free Retrosynthesis. *CoRR*, abs/2006.07038.

Srivastava, N.; Hinton, G.; Krizhevsky, A.; Sutskever, I.; and Salakhutdinov, R. 2014. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research*, 15(56): 1929–1958.

Sutton, R. S.; and Barto, A. G. 2018. *Reinforcement Learning: An Introduction*. The MIT Press, second edition.

Szymkuć, S.; Gajewska, E. P.; Klucznik, T.; Molga, K.; Dittwald, P.; Startek, M.; Bajczyk, M.; and Grzybowski, B. A. 2016. Computer-Assisted Synthetic Planning: The End of the Beginning. *Angewandte Chemie International Edition*, 55(20): 5904–5937.

Tetko, I. V.; Karpov, P.; Van Deursen, R.; and Godin, G. 2020. State-of-the-art augmented NLP transformer models for direct and single-step retrosynthesis. *Nature Communications*, 11(1).

Toniato, A.; Schwaller, P.; Cardinale, A.; Geluykens, J.; and Laino, T. 2021. Unassisted noise reduction of chemical reaction datasets. *Nature Machine Intelligence*, 3: 1–10.

Wang, X.; Qian, Y.; Gao, H.; Coley, C.; Mo, Y.; Barzilay, R.; and Jensen, K. F. 2020. Towards efficient discovery of green synthetic pathways with Monte Carlo tree search and reinforcement learning. *Chem. Sci.*, 11: 10959–10972.

Yan, C.; Ding, Q.; Zhao, P.; Zheng, S.; YANG, J.; Yu, Y.; and Huang, J. 2020. RetroXpert: Decompose Retrosynthesis Prediction Like A Chemist. In Larochelle, H.; Ranzato, M.; Hadsell, R.; Balcan, M. F.; and Lin, H., eds., *Advances in Neural Information Processing Systems 33*, volume 33, 11248–11258. Curran Associates, Inc.

Zheng, S.; Rao, J.; Zhang, Z.; Xu, J.; and Yang, Y. 2020. Predicting Retrosynthetic Reactions Using Self-Corrected Transformer Neural Networks. *Journal of Chemical Information and Modeling*, 60(1): 47–55.

Zhuo, H. H.; and Kambhampati, S. 2017. Model-lite planning: Case-based vs. model-based approaches. *Artif. Intell.*, 246: 1–21.

Zhuo, H. H.; Muñoz-Avila, H.; and Yang, Q. 2014. Learning hierarchical task network domains from partially observed plan traces. *Artif. Intell.*, 212: 134–157.

Zhuo, H. H.; and Yang, Q. 2014. Action-model acquisition for planning via transfer learning. *Artif. Intell.*, 212: 80–103.

Zhuo, H. H.; Yang, Q.; Hu, D. H.; and Li, L. 2010. Learning complex action models with quantifiers and logical implications. *Artif. Intell.*, 174(18): 1540–1569.

Appendix

(I) NOC Construction

The Network of Organic Chemistry (NOC) is a directed graph. Every node consists of a molecule and the edge from node A to node B indicates that there is a reaction where A belongs to its reactants and B to its products.

We first initialize the directed graph, adding each molecule in *eMolecules* as a node to the graph. The following process is then repeated until the graph is no longer changing: we traverse each reaction in USPTO, and if all of its reactants are in the graph, then each of its products will be added to the graph as a new node. And for each new node, new edges from every reactant to it will be added to the graph.

In this directed graph, every node has its $outdegree$ and $cost$. The definition of $outdegree$ is the same as that of a normal directed graph. After the graph is constructed, every molecule in the graph has its synthetic route, which forms a synthetic tree. So the $cost$ of a molecule is the longest path length from its root to its leaf nodes in the synthetic tree.

There are 4650 molecules with $outdegree \geq 2$ and $cost \geq 4$, from which 907 molecules which are difficult to solve using Greedy DFS are selected. $Outdegree \geq 2$ means that the molecule is on the synthetic pathways of at least two complex molecules so has richer experience. Because the molecules with higher cost would be break down to those with lower cost, EG-MCTS will collect the experience of these lower-cost molecules during the searching for those with higher cost. Due to this, we put a limit on the cost to avoid experience redundancy. In order to enrich the synthetic experience, we also select some molecules with higher cost. There are 1499 molecules with $cost \geq 9$, and after DFS search we select 631 molecules which are then divided randomly into three parts: 286, 165, and 180 respectively. These 286 molecules will be combined with the 907 molecules mentioned above as the final training set of $1,193$. The remaining 165 and 180 compounds are used as the validation set and the test set.

(II) Testing Molecules Used in `EG-MCTS` Versus Literature

The Table 4 shows the 30 testing molecules used in the experiment comparing the generated routes given by `EG-MCTS` and the published routes. It shows the CAS Number of each molecule. If the corresponding synthetic route is

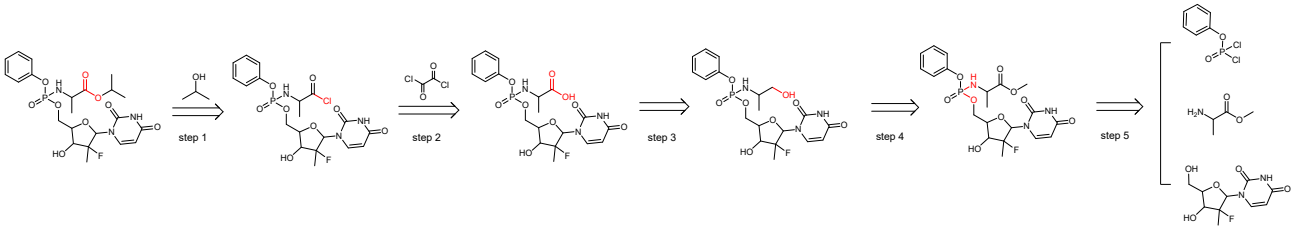
Index	CAS Number	Published Route
1	1448441-60-8	WO 2013107283
2	895520-52-2	WO 2006069153
3	1100216-25-8	WO 2009009411
4	1392842-01-1	WO 2012106472
5	1443043-01-3	US 20130150407
6	1173981-96-8	US 20090186879
7	1173980-10-3	US 20090186879
8	866920-26-5	WO 2005097786
9	1352087-71-8	FR 2960876
10	1448441-53-9	International Journal of Cancer
11	749922-13-2	WO 2006028451
12	1451094-21-5	US 20130225588
13	1100217-13-7	WO 2009009411
14	1100216-27-0	WO 2009009411
15	1617516-73-0	US 20140194476
16	1173979-95-7	US 20090186879
17	1203552-27-5	WO 2010000773 and WO 2013079708
18	1040247-00-4	WO 2008089459
19	1173978-72-7	Bioorganic Medicinal Chemistry
20	1173979-96-8	US 20090186879
21	1392843-72-9	WO 2012106472
22	769169-77-9	US 20040198778
23	1451094-35-1	US 20130225588
24	1498291-86-3	WO 2013180265
25	1801756-11-5	WO 2013107283
26	1392841-71-2	WO 2012106472
27	1873306-29-6	WO 2016016368
28	345963-30-6	WO 2002018361
29	1392841-74-5	WO 2012106472
30	1246199-40-5	US 20090186879

Table 4: 30 testing molecules used in the experiment.

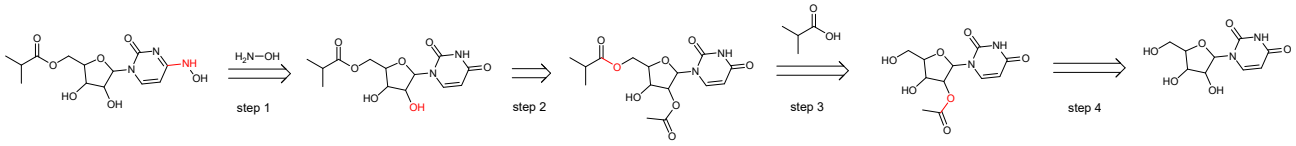
reported in the patent, the table also shows the Patent Number. Otherwise it shows the journal name.

(III) The Generated Routes for Other Three Drug Molecules

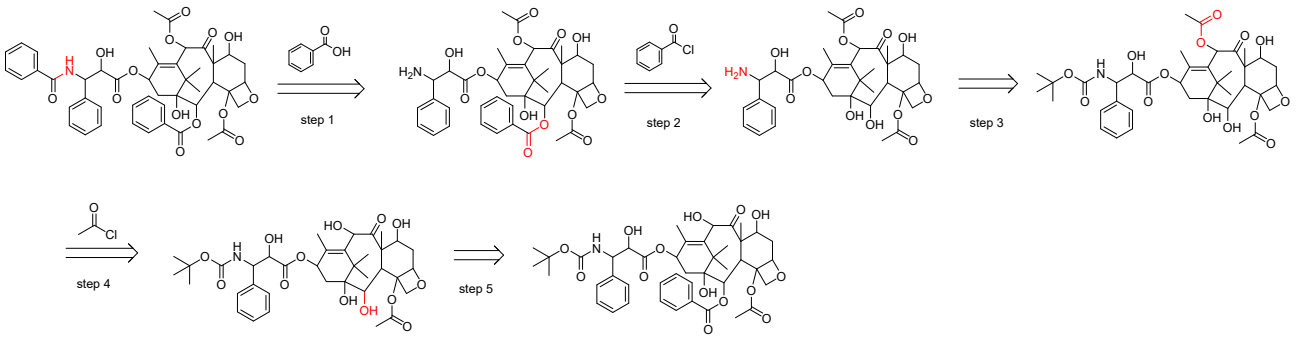
Figure 11 shows the generated routes given by our EG-MCTS for Sofosbuvir, Molnupiravir and Taxol. We also ignore their stereochemical structures to get the target molecules for EG-MCTS.



(a) Route given by our EG-MCTS for Sofosbuvir, whose CAS Number is 1190307-88-0.



(b) Route given by our EG-MCTS for Molnupiravir, whose CAS Number is 2349386-89-4.



(c) Route given by our EG-MCTS for Taxol, whose CAS Number is 33069-62-4.

Figure 11: The generated routes given by our EG-MCTS for other three drug molecules. The atoms and bonds marked red are reaction center, which change in the reaction.