A Planning System for Organic Chemistry Total Synthesis based on A* Search Algorithm

Yusheng JIANG † Hayato YAMANA ‡

† Graduate School of Fundamental Science and Engineering, Waseda University 3-4-1 Okubo, Shinjuku-ku, Tokyo, 169-8555 Japan

‡ Faculty of Science and Engineering, Waseda University 3-4-1 Okubo, Shinjuku-ku, Tokyo, 169-8555 Japan

‡ National Institute of Informatics, 2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo, 101-8439 Japan

E-mail: {amadeus, yamana}@yama.info.waseda.ac.jp

Abstract In this paper, we present our proposal and research progress about an organic total synthesis planning system, through applying A* search algorithm on graph-shaped organic compound database. It takes huge time and effort for organic chemists to come up with a possible synthesis way towards a complicated organic structure, and their synthesis route cannot be guaranteed successful. By introducing planning, evaluation and recommendation system, we can greatly speed up their total synthesis research. Currently, there is very few research in relation with organic total synthesis auto-planning, and the newest research is using brute-force search algorithm. By applying A* search algorithm, we are confident in speeding up the process. Different from brute-force depth-first-search, A* search algorithm would evaluate possible moves through heuristic estimation functions; In this specific case, the estimation function would be given out by judging according to the structure of compound, while complicated structures like middle-scale rings or weakly protected functional groups would be considered high in cost, and would be less considered or even omitted by search algorithm. Through this way, we can prune many hard-to-realize routes and reduce the number of branching options in search.

Keyword A* Search Algorithm, Organic Total Synthesis Planning, Route Search, Evaluation, Recommendation

1. INTRODUCTION

People have started to pursue the automation of chemistry research since the invention of computer science. Coming with the development of instrumental chemistry, chemists have broadened their eyesight into picometre and femtosecond scale of view. With powerful statistic tools, chemists can gather much more information than before. However, in the field of organic synthesis, huge amount of effort was put in, but seldom do chemists see good outcomes. Organic chemists nowadays are doing the same thing as what E. J. Corey is doing in 1960s or what K. C. Nicholau is doing in 1980s, i.e., sketching reaction routes on papers with pens, while manually parsing through chemical reaction indexes. The only difference is that they are now gaining access to information on databases like CAS SciFinder [1] or CrossFire Reaxys [2], instead of reading monumental collection of Beilstein Indexes. It is a truth that they are still using outdated way of research now.

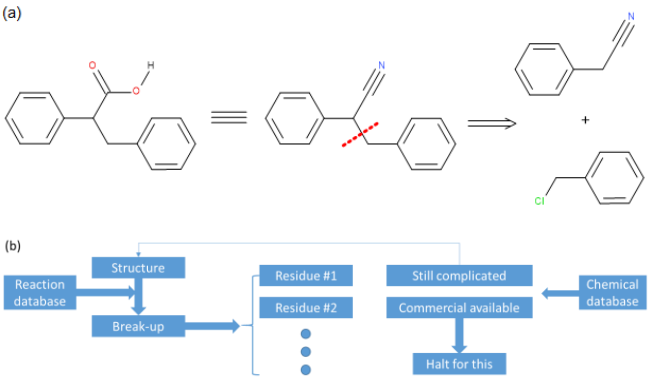
As a matter of fact, people have considered making a planning system for organic total synthesis since 1950s, while 2 USSR scientists Vléduts and Finn mentioned their proposal of an "information machine of chemistry" to "store unlimited amount of chemical information" to fulfill several goals including "search for ways of synthesizing a given compound from a definite number of permissible initial compounds" in 1957[6]. Their goal for digitalized chemical database has already been reached, but only the purpose of synthesis planning is yet left to realize [1,7]. When we access online chemical database services, for synthetic planning, e.g., SciFinder can only give out step-by-step reaction suggestions. However, we must know that step-by-step planning is not always going to give out one overall best solution.

For people outside of organic chemistry, the authors must explain the field of organic synthesis to help them understand the difficulty of auto-planning here. Usually, the target of organic synthesis is a complicated organic structure, usually the structure of a natural product or the structure of a typical new medicine. These structures, usually composed of several ring structures, usually fused rings, and several chirality centers. The main goal is to construct the structure from several basic structures, which are called commercially-available materials. Due to the intrinsic high-level complexity of target molecules, organic synthesis is to some extent considered an art instead of science. So, when the artists perform their skills,

[1] https://scifinder.cas.org/, SciFinder is a product powered by Chemical Abstract Service (CAS).

[2] https://www.reaxys.com/, offered by Elsevier.

they would first break the whole complicated structure into several parts, which in fact, a usage of divide-and-conquer strategy. The divided parts are called residues. When a residue is considered commercially available or the residue has a classic synthetic method, the analysis of the residue will stop; or else, the residue would be considered a target and the procedure would continue. This well-known, widely used method is called retrosynthetic analysis. Here, "retro-" means that it starts from the target, and then goes towards the source so that the procedure is in fact to the opposite direction compared with laboratory synthesis. It was systematically



introduced by E. J. Corey in 1995 [2]. Shown in Figure 1a is an illustration of retrosynthetic analysis.

Figure 1 (a) Illustration of retrosynthetic analysis using the example of 2,3-diphenylpropionic acid. The acid underwent a synthetically-equivalent procedure and was equal to 2,3-diphenylpropionic cyanide. Then the break-up underwent between Carbon 2 and 3 to form the two residues, phenyl ethyl cyanide and phenyl methyl chloride. Both residues are commercial available, and the retrosynthesis reached a result. (b) A flow chart of retrosynthetic analysis.

We can make a comparison for organic synthesis planning and route finding, shown in Table 1. For organic synthesis, different from others is that the number of goal points is more than one, and only reaching every goal can make a clear trial.

Table 1 Retrosynthesis Considered as Route Finding

Problem	Establish a route between a complicated chemical structure with several simpler structures.
Steps, *s*	Average 20~30 steps. Long as >50 steps in some cases.
Terminals (Goals), *g*	>300,000 commercial available materials, according to Merck Millipore's report.[3]
Possible Moves, *p*	>14.2 million chemical reactions are suitable for chemistry synthesis, as reported by CAS.[4]
	>1,000 commonly used reactions [5].
Judgement	Based on various evaluation ways, the judgement of a reaction can differ very much. Basically, a route is judged by an overall cost.

2. RESEARCH BACKGROUND

2.1 Chemical Structures for Computers to Understand

To let computers, know a chemical structure and tell differences between one and another takes human a huge amount of time. Finally, in 1985, under the effort of E.J. Corey and his chemistry and computer science colleagues, Project LHASA (Logic and Heuristics Applied to Synthetic Analysis) succeeded [10], and now computers are clever enough to perform some complicated acts, like translating a chemical structure to its Latin-style IUPAC names [10b].

As we know, chemical structures are shown in the form of a 2D-diagram, which we call them structural formula. Structural formula consists of several ways of expressions, and the most common way is called Kekulé formula. As shown in Figure 1, a typical Kekulé would emit the expression of Carbons and Hydrogen that are connected directly with Carbons, thus in essence, shown the "scaffold part" and the "functionalization part" of a chemical structure [5]. To say it more clearly, a chemistry structure itself is expressed as a combination of a set of molecules and a set of bonds, which is very equivalent to that of a graph data structure [11]. In fact, nowadays almost every one of the most widely used chemistry structure format is stored as graph data structure [12].

In our research, we introduce the molfile format. This type of storage type, comparing to other types, does not contain any other information instead of the structure itself, which, in return, reduces the huge memory demand for storage and processing. Also, molfile is not encrypted and is stored in ANSI code, making it easier for programs to carry out I/O orders [13].

2.2 Network of Organic Chemistry and Usage

Basically, chemical database is not different from any other achieved documents: A chemical name is allocated to one illustration of structure, together with various

[3] http://www.merckmillipore.com/

[4] http://support.cas.org/content/reactions

information on its physical/chemical properties. However, B. A. Grzybowski et. al. introduced a novel method of chemical database architecture: Network of Organic Chemistry (NOC) [1,3,8,9]. For a single chemistry reaction, it is viewed as a connection between reagents and products, which, are both belonging to chemical structures. In this case, NOC stores chemical structures and reaction information together: in a graph-shaped data structure, a vertex can be a chemical structure or a reaction. However, no adjacent vertex is belonging to the same type. Structures would be separated by reactions, while reactions would also be separated by chemical structures. This, in fact, constructs the structure of bipartite graph, which is considered a revolutionary novel way of chemical data structure [3].

Network of Organic Chemistry is, comparing to traditional chemical data structure, much more useful and informative. By adopting Best-first search method, Grzybowski performed a search for Paclitaxel's synthetic route [1]. It took K. C. Nicolaou over 7 years to complete the total synthesis, while the same route was calculated by NOC in less than 7 seconds [1,4].

Although it is an exciting act of revolution, to become useful it still has a long way to go. Also, for our title, we need to reconsider the nature of chemical synthesis. We must make it clear that chemistry synthesis is to create something that is totally new, i.e. something that do not exist in databases. The fact that Grzybowski's best-first search method can recommend the exact synthesis planning route used by Nicolaou is because Nicolaou's synthesis route is fully recorded in NOC. In chemists' stance, a planning system based on known knowledge is not helpful as we have thought, because the problem is about dealing novel compounds that are totally new and have not established its place in NOC.

3. PROBLEM CLARIFICATION

In a given directed graph $G(V,E)$, define a set of vertexes $Mat \subset V$ as a set of goal points (commercial available materials). Insert a new vertex s as starting point (the synthesis target). Our goal is to establish and output a route with the smallest overall cost from s to a subset of Mat. Two kinds of operations are allowed for the route establishment:

(1) Synthetic equivalence: insert a directed edge (x,y) to E, where $x,y \in V$. If $y \notin V$, insert y to V.

(2) Split: Insert a special directed edge with one inlet x and two outlets y_1 and y_2, i.e., $(x,(y_1,y_2))$, to E, where $x,y_1,y_2 \in V$. If $y_1 \notin V$, insert y_1 to V. If $y_2 \notin V$, insert y_2 to V.

The above two operations are shown in Figure 2.

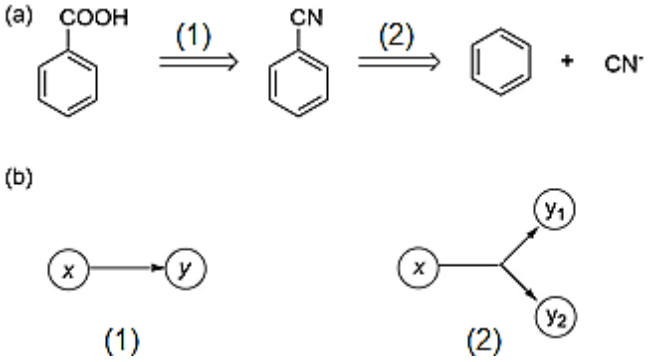


Figure 2. Two operations of retrosynthesis. (a) A simple example containing two types of retrosynthetic processes. Operation 1 makes a connection between one structure and another one; Operation 2 splits one structure to two parts. (b) An illustration indicating the model we established for the two operations. For synthetic equivalent case, the edge we added to E is the ordinary directed edge (x,y). For split cases, the edge we added to E is a special edge with single inlet and double outlet $(x,(y_1,y_2))$. x, y, y_1, y_2 are vertexes (chemical structures).

For the situation that multiple routes exist, output the routes which have the smallest cost based on a cost function, as shown in Figure 3.

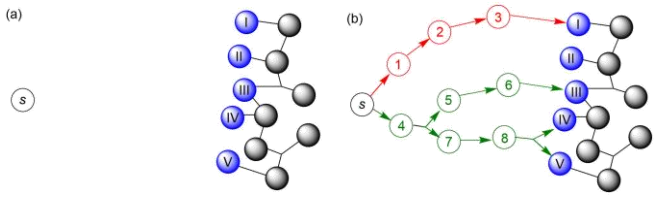


Figure 3. (a) The definition of problem. The right hand side graph represents G. Vertexes colored in blue represent Mat. Vertex s is inserted to G as the starting point. The problem is to find out routes between s and Mat. (b) Two routes (colored in red and green) established according to the operation rules. For the red one, directed edge set $\{s \Rightarrow$

1, $1 \Rightarrow 2$, $2 \Rightarrow 3$, $3 \Rightarrow I$ } together with vertex set {1, 2, 3} are added to the graph G. For the green one, directed edge set { $s \Rightarrow 4$, $4 \Rightarrow (5,7)$, $5 \Rightarrow 6$, $7 \Rightarrow 8$, $6 \Rightarrow III$, $8 \Rightarrow (IV, V)$}, together with vertex set {4, 5, 6, 7, 8} are added to the graph G. We need to calculate the cost of the two route to decide which to recommend.

To avoid the situation that no suitable route is found and the search cannot stop, the number of steps in the route is limited to a certain value.

4. METHODOLOGY

4.1 Algorithm Definition

Our novelty is to adopt A* algorithm to tackle the problem of synthesis planning in the field of chemistry, and then present its efficiency.

Based on our knowledge in this field, we believe that a heuristic search algorithm is the only way to tackle the problem. Other algorithms already applied by other researches such as breadth-first search, result in low efficiency. As shown in Table. 1, the number of possible moves p and the number of steps s is too huge for breadth-first search with the complexity of time $O(p^s)$. Grzybowski's best-first search algorithm, though quick, has limitations when dealing with the situation of planning for new chemicals.

We adopt A* algorithm as a part of our system. Standard A* algorithm [14] for route-finding problems calculates the cost function according to the estimation shown as the following formulas:

$$f(m) = g(m) + h(m)$$

(Formula. 1a)

$$f'(n) = g'(n) + h(n) = g(m) + cost(m,n) + h(n)$$

(Formula. 1b)

Here, $f(m)$ means the shortest route from the start to goal containing node m. m is the node taken out from the priority queue, i.e., the node we are going to expand. $g(m)$ is the shortest path from start node to m, $h(m)$ is the shortest route from m to goal node, generated by a heuristic estimation function. n is a neighboring node of m that have not been expanded, where $cost(m,n)$ is the weight of directed arc (m,n). In A* algorithm, $f(m)$ is the criteria of the priority for pushing the nodes into the priority queue, nodes near the front of the queue have smaller $f(m)$ values. Every time, node m at the front of the queue, with the least $f(m)$ value, is taken out for expansion. For each successor node n expanded from m, we update the value of $f(n)$ according to Formula 1b to gain the smallest value for $f(n)$. This is the situation for route-finding problem and original A* algorithm.

Here we compare our problem with ordinary route-finding problem and map our algorithm with A* algorithm. Our problem is different from ordinary route-finding problems in that: (1) A* algorithm assumes only one goal node for route-finding, for our problem there exist several goal points. (2) A* algorithm assumes a known graph G with clear information of every vertex and edge. For our case, the synthesis target (starting point) is totally new and it should be added as an islet vertex to NOC. It is our target to establish connection between the start node and known graph. So, we do not know the whole graph information. (3) For route-finding problems, the total cost is only shown as the summation of the weight of edges within a selected route, however, in our case, the total cost should be add up to the price of materials. In this case, we can see it is not simple to calculate the cost by applying A* algorithms. The comparison is shown in Table 2.

Table 2 Differences between ordinary route-finding problem and our problem

Items	Route-finding problem	Our problem
Number of goals	Only one	Multiple
Graph	Known	Unknown
Total cost	$\sum_{x} dist(x, x.next), \substack{x \in route \\ x \neq goal}$	$\sum_{x} price(x), x \in goals$ $+ \sum_{y} cost(y), y \in reactions$

Here, we propose our algorithm as follows. We define,

$$f(x) = g(x) + h(x)$$

(Formula 2)

Here x is the node (chemical structure) to expand. $f(x)$ represents the smallest overall cost from the start node to the goals through a route containing node x. $g(x)$ is the overall reaction cost from start node to x. $h(x)$ is a heuristic that estimate the cost to build up chemical structure x from basic starting materials.

We prepare a priority queue for A* algorithm. The queue is sorted by the $f(x)$ value of nodes. Each time, a node with the least $f(x)$ value is taken out of the priority queue for expansion. We parse through the database of reactions. For each reaction $x \Rightarrow \{x.residue\}$ that is suitable for chemical structure x, we perform an update for $f(x.residue)$ and $g(x.residue)$ values. This update process is different for two operations, as mentioned in Figure 2b.

As mentioned in Section 3, there are two kinds of operations we can perform in retrosynthesis: (1) Synthetic equivalence, $x \Rightarrow y$. (2) Split, $x \Rightarrow y_1 + y_2$; as shown by Figure 2. Here x still represents the node to expand, y, y_1, y_2 represent the neighboring nodes. Operation (1) (Synthetic equivalence) will derive only one neighboring node, therefore the problem will become equivalent with the original A* route-finding problem. Then, the update function will become:

$$f'(y) = g'(y) + h(y) = g(x) + cost(x \Rightarrow y) + h(y)$$

(Formula 3)

For the second operation (split), the specific directed edge is pointed to two neighboring nodes y_1 and y_2. In Figure 2b, we can understand we have to contain the cost for synthesizing both residues when we calculate $f(y_1)$ and $f(y_2)$, which represents the minimum overall cost for a route containing y_1 and y_2. This kind of change should be happened to $g(y_1)$ and $g(y_2)$, because the $g(x)$ function indicates the summary cost that have already occurred. In this case, we indicate the following update function as:

$$g'(y_1) = g(x) + cost(x \Rightarrow y_1 + y_2) + h(y_2)$$

(Formula 4a)

$$g'(y_2) = g(x) + cost(x \Rightarrow y_1 + y_2) + h(y_1)$$

(Formula 4b)

$$f'(y_1) = g'(y_1) + h(y_1) = g'(y_2) + h(y_2) = f'(y_2)$$

(Formula 4c)

Shown on the part marked in red color is the main difference we have added to the original A* algorithm formula. For the calculation for one residue, take y_1 as an example, we cannot ignore the cost to synthesize the other residue, y_2. Thus, we add the heuristic estimation of the cost to synthesize y_2 to formula of $g'(y_1)$, shown as Formula 4a. Things also goes for $g'(y_2)$, shown by Formula 4b. Then, by calculating $f'(y_1)$ and $f'(y_2)$, we found that the two residues bear the same value. Therefore, if the update process succeeds, they will show up in the priority queue in a neighbor status.

We can establish a mapping between our algorithm with original A* algorithm now. (1) As shown in Formula 1a and Formula 2, our algorithm adopts the same assumption that the overall route length (expressed as $f(m)$, $f(x)$) should be the sum of the known part of a route (expressed as $g(m)$, $g(x)$) and an estimation of the rest part of the route (expressed as $h(m)$, $h(x)$). (2) For the first operation, we reduced the problem to the same as route-finding problem, as Formula 1b is equivalent to Formula 3. For the second operation, by adding elements to the expression of $g'(y_1)$ and $g'(y_2)$, eventually the expression for $f'(y_1)$ and $f'(y_2)$, shown in Formula 4c, are also equivalent to Formula 1b.

4.2 Heuristic Function Definition

We define the heuristic function as,

$$h(x) = \alpha h_1(x) + \beta h_2(x) + \gamma h_3(x)$$

where,

$$h_1(x) = molecule\ weight\ of\ x$$

$$h_2(x) = ring\ structure\ \ evaluation\ \ of\ \ x$$

$$h_3(x) = unstable\ \ functional\ \ group\ \ evaluation\ \ of\ \ x$$

(Formula 5)

Here is an explanation to Formula 5: (1) For a molecule, the bigger (heavier) the molecule is, the harder it is to make it up through simple structures, Thus, we add $h_1(x)$ which is the molecular weight of x to the formula. (2) Ring structure is often the most important main obstacle to construct in organic chemical structures. In this case, we need to give ring structures a penalty when estimating synthesis costs. Here, we provide a judge function: $h_2(x)$. This function will return a constant value based on the ring structure contained in the structure of x. (3) For special form of structure, we need to judge whether there are unstable functional group, including the environmental-unstable, communicative-unstable and intrinsic-unstable ones. Unstable structures would also devote to increase in the difficulty of synthesis, giving a penalty value. So, we offer a function: $h_3(x)$. This function will return a value

based on the existence of unstable functional groups. (4) α, β, γ are constant parameters.

4.3 Employing the classical synthetic strategy

In Organic Synthesis, there are empirical rules gathered by researchers, which can be of employed in our A* algorithm to make the heuristic function more accurate. Figure 4 shows an example of an empirical rule for synthesis planning.

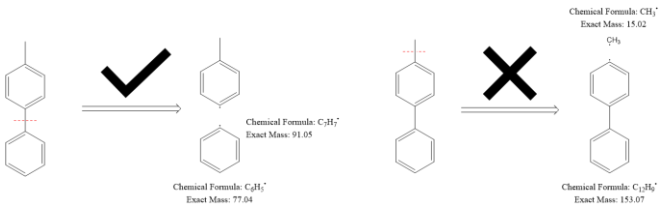


Figure 4. The rule indicates it is better to split the structure from the "middle" part, which would derive two residues of similar weight.

We decide that it is an optional choice to be added to our formula. If we are going to introduce these empirical rules, there exist disadvantages: (1) Some empirical rules are strictly restricted in range of use. (2) Some empirical rules coincide with each other. (3) Employing empirical rules in judgement will surely making our formula longer and longer, and will introduce more parameters, which is a vital damage to our parameter controls.

However, it is still possible to employ such rules by adjusting the $g'(y)$ and $f'(y)$. For example, if we are going to employ the rule showing in Figure 3, we can employ this change to the formula:

$$g'(y_1) = g(x) + cost(x \Rightarrow y_1, y_2) + h(y_2) + \delta|h(y_2) - h(y_1)|^2 \quad \text{(Formula 6)}$$

where, δ is also a parameter.

4.4 Preparation of Database

To implement our system, we need at least two databases, as shown in Figure 1b, i.e., database of reactions, and database of chemicals (with commercial price information). Commercial databases of reactions are not suitable for our system because the arrangement of information is not in the style we want. To improve the efficiency for space usage, we need to rearrange the structure of database.

Commercial databases usually store all reactions they collect. They collect data for reactions and classify and sort them according to reactants, instead of reaction types. In this case, same type of reactions would be saved separately. This kind of storage type of reaction is not in our interest, for it takes up enormous space.

Thus, we must change the stored items from individual reactions to reaction rules, which means that same type of reactions would be combined, which can (1) save much space (2) reduce the risk of error. Chemical reactions are referred to as connection between functional groups[5].

$$Func(A) + Func(B) \rightarrow Func(C)$$

For retrosynthesis, it is presented as:

$$Func(C) \Rightarrow Func(A) + Func(B)$$

A, B, C represent chemical structures, while $Func(X)$ means the functional group extracted from structure X.

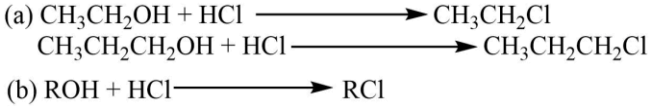


Figure 5. Comparison between storing reactions and reaction rules. (a) Traditional databases tend to store all individual reactions, while not distinguishing reactions of the same type. (b) R is a wildcard for alkyl chain. We can find that the two reactions in Figure 4a is of the same type, both are suitable for the wildcard replacement of the form shown in Figure 4b. The -OH and -Cl are functional groups, which is the place where changes happen when reactions occur.

4.5 Cost Function and Evaluation Method

As shown in Table 2, we arbitrarily define the overall cost function to be the economic cost happened during synthesis procedures. In fact, though important, money does not necessary become the only way to judge a

[5] Functional group is an organic chemistry noun that means specific groups (moieties) of atoms or bonds within molecules that are responsible for the characteristic chemical reactions of those molecules [5].

synthesis scheme. According to situations, we may find that it is better to use some other method which seems costlier.

```
1  Function A*(Start, Graph G(V,E), Goal{})
2    ClosedSet ← ∅
3    OpenSet ← {Start}
5    For ∀x ∈ V
6      f(x) ← ∞, g(x) ← ∞
7    g(Start) ← 0, f(Start) ← h(Start)
8    While (OpenSet ≠ ∅) do
9      current ← {v|v ∈ OpenSet, f(v) is min}
10     //The vertex with smallest f(v)
11     OpenSet.pop(current)
12     ClosedSet.push(current)
13     If current ∈ Goal
14       f(current) ← g(current) + Price(current)
15     Else For ∀e ∈ E|e is applicable to current
16       If e is x → y type
17         y ← GetResidue(current, e)
18         If y ∉ ClosedSet
19           OpenSet.push(y)
20           g'(y) = g(current) + e.cost
21           If g'(y) < g(y)
22             g(y) ← g'(y)
23             f(y) ← g'(y) + h(y)
24           Endif
25         Endif
26       Elseif e is x → y1 + y2 type
27         y1, y2 ← GetResidue(current, e)
28         If y1 ∉ ClosedSet
29           OpenSet.push(y1)
30           g'(y1) = g(current) + e.cost + h(y2)
31           If g'(y1) < g(y1)
32             g(y1) ← g'(y1)
33             f(y1) ← g'(y1) + h(y1)
34           Endif
35         Endif
36         If y2 ∉ ClosedSet
37           OpenSet.push(y2)
38           g'(y2) = g(current) + e.cost + h(y1)
39           If g'(y2) < g(y2)
40             g(y2) ← g'(y2)
41             f(y2) ← g'(y2) + h(y2)
42           Endif
43         Endif
44       Endif
45     Endif
46   Endwhile
```

Figure 6. A pseudocode of our A* algorithm.

For evaluation, i.e. how we are going to judge the overall reaction scheme, we need a typical way of calculating the overall cost based on monitoring real-life chemistry research. Thus, we need datasets. Unfortunately, commercial databases can offer little help on database construction, here, we need to manually input important information, e.g. information on name reactions and information on starting materials. Also, we need to manually input some famous or classic reaction route to train our system.

5. IMPLEMENTATION STATUS

The project was carried out September, 2017. Currently we are implementing it to evaluate our proposed system. The experiment is initiated from small-scaled data; the first experiment adopts data coming from organic chemistry textbook *Fundamentals of the Organic Chemistry* [5]. During the implementation, we have faced with several problems to tackle which are explained below.

5.1 The problem of how to identify functional groups

A functional group is a substructure of a chemical in which reactions take place. As shown in line 15~45 of Figure 6, we need to screen the reaction database to know the reactions that is applicable to any chemical structure. Note that this screening procedure takes place frequently, as this procedure takes place once for every vertex (chemical structure). The identification procedure must be very fast in order to speed up. This problem can be described as how to identify the topological inclusion reaction of two graphs considering how we store our chemical structure.

The most precise way is that we take out every substructure and check the identity between the substructure and functional groups. Due to the efficiency reason, this way is totally unacceptable, as the number of substructures is too huge. We noticed that the chemical structures are different from ordinary graphs whose vertex degree is limited, such as 4 adjacent vertexes for each vertex. Thus, we decide that the screening procedure of functional groups will take place by roughly checking the degree of each vertex and its adjacent vertexes' element type and bond type. For example, in a structure, one Carbon atom connected with one Carbon atom with a single bond, one Oxygen atom with a double bond and one Hydrogen

with a single bond can be represented as C1O2H1, while this adjacency information is identical with the information of the aldehyde group, proving that there is an aldehyde group starting form this Carbon atom.

5.2 The problem of acquiring proper $h(x)$

As shown in Section 4.2 and Formula 5, the most important part is to define a proper $h(x)$ function. We have defined that the $h(x)$ is consisted with 5 parameters, $\alpha, \beta, \gamma, h_2(x), h_3(x)$. ($h_1(x)$ is a fixed number for any x.) For $h_2(x)$ we define it as the ring structure evaluation for x. We decided that $h_2(x)$ should be a polynomial of x's total ring strain energy. The ring strain energy data is easily accessible through physical chemistry databases. [15] For $h_3(x)$ we define it as the evaluation for unstable functional groups. Here we decide that $h_3(x)$ should be a polynomial of x's total functional groups' bond energy comparing to an imagined full carbon skeleton's bond energy. For example, a structure contains a carbon-iodine bond and two carbon-bromine bond will be calculated as in Formula 7. The bond energy function BE is also acquirable in physical chemical databases [15].

$$|BE(C-C) - BE(C-I)| + 2 \times |BE(C-C) - BE(C-Cl)| \quad \text{(Formula 7)}$$

Then we can calculate the expression of $h(x)$ by analyzing the market price of existing chemicals and its $h_1(x)$, $h_2(x)$ and $h_3(x)$. We are still doing works on collecting data of these chemicals. After we have collected enough data, we can do regression and fitting to identify the value for α, β, γ.

5.3 The problem of the origin of data

One thing that is to be pointed out is that we defined our goal as to help chemists in laboratories plan their research routes, while we have to use price data gathered from chemistry industries due to the fact that laboratories will neglect the cost provided the funding is adequate. The data of laboratories experimental cost is not explicit as industry production cost [16]. This in fact will have a defect to the performance of our calculated results, as the results are generated with industrial data.

6. CONCLUSION AND FUTURE WORK

We proposed a method to adopt A* algorithm into organic synthesis planning system, through carefully tuning for heuristic estimation function. We reported our implementation process and pointed out several problems. As for the future work, the authors will continue implement our proposed system followed by adding some other cost functions to improve our system. We would like to gain collaboration chance with other laboratories and/or chemistry corporations.

REFERENCE

[1] S. Szymkuć, E. P. Gajewska, T. Klucznik, K. Molga, P. Dittwald, M. Startek, M. Bajczyk, B. A. Grzybowski, *Angew.Chem. Int. Ed.* **2016**, 55,5904–5937

[2] E. J. Corey, X-M. Cheng. *The Logic of Chemical Synthesis*. **1995**, New York: Wiley. ISBN 0-471-11594-0.

[3] M. Fialkowski, K.J.M.Bishop, V.A.Chubukov, C.J.Campbell, B. A. Grzybowski, *Angew.Chem. Int. Ed.* **2005**, 44,7263– 7269

[4] K. C. Nicolaou, D.Vourloumis, N.Winssinger, P. S. Baran, *Angew.Chem. Int. Ed.* **2000**, 39,44–122

[5] J. E. McMurry, *Fundamentals of Organic Chemistry*, **2010**, Boston: Cengage Learning. ISBN 0-534-39573-2

[6] a) G. E. Vléduts, V. K. Finn, *Inf. Storage Retr*. **1963**, 1, 101–116. b) G. E. Vléduts, *Inf. Storage Retr*. **1963**, 1, 117–146.

[7] M. Peplow, *Nature* **2014**, 512, 20–22.

[8] K. J. M. Bishop, R. Klajn, B. A. Grzybowski, *Angew. Chem. Int. Ed.* **2006**, 45, 5348–5354

[9] B. A. Grzybowski, K. J. M. Bishop, B. Kowalczyk, C. E. Wilmer, *Nat. Chem.* **2009**, 1, 31–36.

[10] a) E. J. Corey, W. J. Howe, R. D. Cramer, *J. Am. Chem. Soc.* **1972**, 94, 421 –430. b) D. A. Evans, *Angew. Chem. Int. Ed.* **2014**, 53, 11140–11146.

[11] P. Judson, *Knowledge-based Expert Systems in Chemistry: Not Counting on Computers*, **2009**, Cambridge: RSC. ISBN 0-854-04160-5

[12] K. R. Cousins, *J. Am. Chem. Soc.* **2011**, 133, 8388–8388.

[13] P. Murray-Rust, H.S. Rzepa, *J Cheminform.* **2011**, 3: 44.

[14] N. J. Nilsson, *IFIP Congress*, **1968**, 2, 1556-1562.

[15] P. Atkins, *Atkins' Physical Chemistry*, **2014**, Oxford: Oxford University Press. ISBN 0-199-69740-X

[16] C. A. Heaton, *The Chemical Industry,* **1993**, Netherlands: Springer. ISBN 0-751-40018-1