

プログラムマニュアル:  
与えられた化学グラフの構造異性体の列挙

2021 年 6 月 11 日

## 目 次

1	イントロダクション	1
2	用語の説明	2
3	異性体プログラムの実行と計算例	2
3.1	プログラムの入力と出力	2
3.1.1	プログラムの入力	2
3.1.2	プログラムの出力	5
3.2	プログラムの実行と計算例	5
3.2.1	生成プログラムのコンパイルと実行	5
3.2.2	計算例	6
	参考文献	12

# 1 イン트로ダクション

この冊子では、2-branch 構造 [1] に基づく非環状化学グラフの生成アルゴリズムを実装した二つのプログラムについて説明する。

- フォルダ `isomers`

環状化学グラフの構造異性体を生成するプログラムのソースコードが含まれている。このプログラムは C++ で書かれている。

- フォルダ `include`

このプログラムに使われるヘッダファイルのフォルダである

- \* `chemical_graph.hpp`  
化学グラフを操作するためのヘッダファイルである。
- \* `cross_timer.h`  
計算時間を計測するためのヘッダファイルである。
- \* `data_structures.hpp`  
化学グラフのデータ構造を定義するヘッダファイルである。
- \* `fringe_tree.hpp`  
ヘリ木 (fringe tree) [1] を列挙する関数のヘッダファイルである。
- \* `tools.hpp`  
便利なツール関数のヘッダファイルである。
- \* `tree_signature`  
外縁木の正規表現を計算し読み取るための関数のヘッダファイルである。

- フォルダ `instance`

このフォルダはいくつかの入力のサンプルが含まれている。

- \* `sample.sdf`  
節点数 45 (水素原子はのぞく) の環状化学グラフ。
- \* `sample_partition.txt`  
`sample.sdf` で与えられた環状化学グラフを非環状化学グラフに分割するための情報が含まれているファイル。
- \* `sample_fringe_trees.sdf`  
`sample.sdf` で与えられた環状化学グラフに付随する 2-外縁木の集合。

- フォルダ `main`

このフォルダはいくつかのソースファイルが含まれている。

- \* `generate_isomers.cpp`  
2-lean 環状化学グラフの構造異性体を列挙するためのプログラム。
- \* `Pseudocode_Graph_Generation.pdf`  
グラフ探索アルゴリズムの疑似コードを記した pdf ファイルである..

次に、この冊子の構成について説明する。第 2 章では、この冊子およびプログラム内で使用している用語について説明する。第 3 章では、与えられた 2-lean 環状化学グラフの構造異性体を生成するプログラムの入力と出力について説明し、実際の計算例を示す

## 2 用語の説明

この章では、本冊子で使用する専門用語について説明する。

- 化学グラフ

化学グラフは節点集合と枝集合の組で化合物の構造を表すものである。各節点には原子の種類が割り当てられており、各枝には結合の多重度が割り当てられている。この冊子では水素原子が省略された化学グラフを扱っていく。

- 特徴ベクトル

化学グラフにおける各化学元素の数などの情報を与える数値ベクトル。本プロジェクトの特徴ベクトルに使われるディスクリプターの詳細な情報は、[1] を参照。

- 分割情報

入力する化学グラフの基底節点・枝 (base vertex/edge) の指定とその節点・枝成分 (vertex/edge component)[1] を固定するか可変にするかの情報。より詳細な情報は、[1] を参照。

- 外縁木情報

構造異性体を組み立てる際に使用する 2-外縁木のリスト。より詳細な情報は、[1] を参照。

## 3 異性体プログラムの実行と計算例

### 3.1 プログラムの入力と出力

この節では異性体生成プログラムの実行方法と、具体例を入力した場合の計算結果を示す。ここではプログラムのことを、**異性体生成プログラム**と呼ぶ。3.1.1 節ではプログラムの入力の、3.1.2 節では出力の説明を与える。

#### 3.1.1 プログラムの入力

異性体生成プログラムでは六つの情報の入力が必要としている。

一つ目は 2-lean 化学グラフの情報 (SDF フォーマット) である。

二つ目はプログラムの計算時間の上限秒である。(より詳細な情報はプログラムの疑似コードを参照)。

三つ目は蓄える特徴ベクトルの数の上限である。

四つ目は基底接点・枝あたりに蓄えるグラフの数である。

五つ目はパスを列挙する時間の上限である。

六つ目は DAG から列挙するパスの数の上限である。七つ目は出力する化学グラフの個数の上限である。

八つ目は出力される化学グラフを保存する SDF ファイル名である。

九つ目は入力する化学グラフの分割情報である。

十個目は外縁木のリストのファイル名である。

化学グラフはSDF ファイルの形式で入力する. SDF ファイルのフォーマットについて,  
<https://www.chem-station.com/blog/2012/04/sdf.html>  
などの解説が分かりやすい. さらに, 正確な定義書として, 公式資料  
[http://help.accelrys.com/ulm/online/1.0/content/ulm\\_pdfs/direct/reference/  
/ctfileformats2016.pdf](http://help.accelrys.com/ulm/online/1.0/content/ulm_pdfs/direct/reference/ctfileformats2016.pdf)  
を参照するとよい.

分割情報は Stage 4 の出力であり, テキストファイルとして蓄えられている.  
sample.partition.txt は以下のようにになっている:

#### 分割情報ファイル

```
2
14
0 45
1 2 3 4 5 6 7 8
15
0 45
1 2 3 4 5 6 7 8
2
14 1 2 3 5 6 15
0 45
1 2 3 4 5 6 7 8
14 10 15
0 45
1 2 3 4 5 6 7 8
```

以下の表 1 では, 上記の分割ファイルの数値の意味を一行毎に説明する.

表 1: Structure of a Partition Information

ファイル中の値	説明
2	基底節点の数
14	入力した SDF ファイルにおける基底節点の指数
0 45	核高の下限と上限
1 2 3 4 5 6 7 8	外縁木ファイルにおける外縁木の指数
15	
0 45	
1 2 3 4 5 6 7 8	
2	基底枝の数
14 1 2 3 5 6 15	Indices of vertices in the base edge from the input SDF
0 45	核高の下限と上限
1 2 3 4 5 6 7 8	外縁木ファイルにおける外縁木の指数
14 10 15	
0 45	
1 2 3 4 5 6 7 8	

外縁木のリストはテキストファイルに保存されている. `sample_fringe_trees.sdf` は以下のよう  
に与えられている:

外縁木ファイル	
1, C 0 C 1, 1	
2, C 0 0 1, 2	
3, N 0 C 1, 1	
4, C 0 C 1 C 2, 1 1	
5, C 0 C 1 C 2 C 2, 1 1 1	
6, C 0 C 1 0 2 0 2, 1 1 2	
7, N 0 C 1 N 2 N 2, 2 1 1	
8, C 0 C 1 C 2 C 1, 1 1 1	

以下の表 2 では, 上記の外縁木ファイルの数値の意味を一行毎に説明する.

表 2: 外縁木ファイルの構造

ファイル中の値	説明
1, C 0 C 1, 1	外縁木の指数, (元素, 深さ)-順序, 重み より詳細については Pseudocode_Graph_Generation.pdf を参照
2, C 0 0 1, 2	
3, N 0 C 1, 1	
4, C 0 C 1 C 2, 1 1	
5, C 0 C 1 C 2 C 2, 1 1 1	
6, C 0 C 1 0 2 0 2, 1 1 2	
7, N 0 C 1 N 2 N 2, 2 1 1	
8, C 0 C 1 C 2 C 1, 1 1 1	

### 3.1.2 プログラムの出力

異性体生成プログラムを実行した後, 入力グラフの構造異性体は指定の SDF ファイルに書き込まれ, 実行に関する幾つかの情報がターミナル上に出力される. ターミナルに出力される情報は以下である:

- 与えられた化学グラフの構造異性体の数
- 与えられたパラメーター下での生成したグラフの個数
- プログラムの実行時間

## 3.2 プログラムの実行と計算例

この節では異性体生成プログラムの具体的な計算例を与える.

### 3.2.1 生成プログラムのコンパイルと実行

#### ● 環境確認

ISO C++ 2011 標準に対応する C++コンパイラーがあれば問題ないと考えられる. Linux Ubuntu 20.04, コンパイラー g++ ver 9.3 で確認したが, g++がインストールされていない場合は, 次のようにインストールできる.

```
$ sudo apt install g++
```

#### ● コンパイル

このプログラムは以下コマンドでコンパイルできる.

```
$ g++ -o generate_isomers generate_isomers.cpp -O3 -std=c++11
```

#### ● プログラムの実行

このプログラムは以下コマンドで実行できる.

```
$ ./generate_isomers instance.txt a b c d e f output.sdf instance_partition.txt
```

上記において, `generate_isomers` はプログラムの実行ファイルの名前であり, 残りのコマンドライン上のパラメーターは以下である:

`instance.txt` 入力のテキストファイル,  
a 計算時間の上限 (sec),  
b で蓄える特徴ベクトルの数の上限,  
c 基底接点・枝あたりに蓄えるグラフの数,  
d パスを列挙する時間の上限 (sec),  
e DAG から列挙するパスの数の上限,  
f 出力する化学グラフの個数の上限,  
`output.sdf` 出力する化学グラフを蓄える SDF ファイル,  
`instance_partition.txt` 入力する化学グラフの分割情報,  
`instance_fringe_trees.txt` 外縁木のリスト.

### 3.2.2 計算例

具体例として, 異性体生成プログラムを以下のような条件で実行する.

- 入力グラフ: フォルダ `instances` 内の `sample.sdf`
- 計算時間上限: 10 秒
- 特徴ベクトルの個数の上限: 10000
- 基底接点・枝あたりに蓄えるグラフの数: 5
- パスを列挙する時間の上限: 10 秒
- 列挙するパスの数の上限: 10000
- 出力する化学グラフの個数の上限: 2
- 出力する化学グラフを蓄えるファイル名: `output.sdf`
- 入力する化学グラフの分割情報: フォルダ `instances` 内の `sample_partition.txt`
- 入出力グラフの外縁木: フォルダ `instances` 内の `sample_fringe_tree.txt`.

実行のコマンドは以下になる.

```
./generate_isomers ../instance/sample.sdf 10 10000 5 10 10000 2
                    output.sdf ../instance/sample_partition.txt
                    ../instance/sample_fringe_tree.txt
```

プログラムの実行が成功した場合, 以下のようなテキストがターミナル上に現れる.



## ターミナル上の出力

A lower bound on the number of graphs = 10512

Number of generated graphs = 2

Total time : 34.1s.

## output.sdf の内容

1

BH-2L

BH-2L

45 45 0 0 0 0 0 0 0 0999 V2000

0.0000 0.0000 0.0000 O 0 0 0 0 0 0 0 0 0 0 0 0

0.0000 0.0000 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0

0.0000 0.0000 0.0000 N 0 0 0 0 0 0 0 0 0 0 0 0

0.0000 0.0000 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0

0.0000 0.0000 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0

0.0000 0.0000 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0

0.0000 0.0000 0.0000 O 0 0 0 0 0 0 0 0 0 0 0 0

0.0000 0.0000 0.0000 O 0 0 0 0 0 0 0 0 0 0 0 0

0.0000 0.0000 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0

0.0000 0.0000 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0

0.0000 0.0000 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0

0.0000 0.0000 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0

0.0000 0.0000 0.0000 N 0 0 0 0 0 0 0 0 0 0 0 0

0.0000 0.0000 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0

0.0000 0.0000 0.0000 O 0 0 0 0 0 0 0 0 0 0 0 0

0.0000 0.0000 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0

0.0000 0.0000 0.0000 O 0 0 0 0 0 0 0 0 0 0 0 0

0.0000 0.0000 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0

0.0000 0.0000 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0

0.0000 0.0000 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0

0.0000 0.0000 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0

0.0000 0.0000 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0

0.0000 0.0000 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0

0.0000 0.0000 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0

0.0000 0.0000 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0

0.0000 0.0000 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0

0.0000 0.0000 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0

0.0000 0.0000 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0

0.0000 0.0000 0.0000 N 0 0 0 0 0 0 0 0 0 0 0 0  
 0.0000 0.0000 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0  
 0.0000 0.0000 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0  
 0.0000 0.0000 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0  
 0.0000 0.0000 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0  
 0.0000 0.0000 0.0000 N 0 0 0 0 0 0 0 0 0 0 0 0  
 0.0000 0.0000 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0  
 0.0000 0.0000 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0  
 0.0000 0.0000 0.0000 O 0 0 0 0 0 0 0 0 0 0 0 0  
 0.0000 0.0000 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0  
 0.0000 0.0000 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0  
 0.0000 0.0000 0.0000 O 0 0 0 0 0 0 0 0 0 0 0 0  
 0.0000 0.0000 0.0000 O 0 0 0 0 0 0 0 0 0 0 0 0  
 0.0000 0.0000 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0  
 0.0000 0.0000 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0  
 0.0000 0.0000 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0  
 0.0000 0.0000 0.0000 O 0 0 0 0 0 0 0 0 0 0 0 0  
 0.0000 0.0000 0.0000 O 0 0 0 0 0 0 0 0 0 0 0 0  
 1 9 1 0 0 0 0  
 1 27 1 0 0 0 0  
 2 3 1 0 0 0 0  
 2 9 1 0 0 0 0  
 2 13 1 0 0 0 0  
 3 4 1 0 0 0 0  
 4 5 1 0 0 0 0  
 5 6 1 0 0 0 0  
 6 7 1 0 0 0 0  
 6 8 2 0 0 0 0  
 9 10 1 0 0 0 0  
 10 11 1 0 0 0 0  
 10 12 1 0 0 0 0  
 13 14 1 0 0 0 0  
 13 21 1 0 0 0 0  
 14 15 2 0 0 0 0  
 14 16 1 0 0 0 0  
 16 17 2 0 0 0 0  
 16 18 1 0 0 0 0  
 18 19 1 0 0 0 0  
 19 20 1 0 0 0 0  
 21 22 1 0 0 0 0

```

21 23 1 0 0 0 0
23 24 1 0 0 0 0
23 25 1 0 0 0 0
25 26 1 0 0 0 0
25 27 1 0 0 0 0
27 28 1 0 0 0 0
28 29 1 0 0 0 0
29 30 1 0 0 0 0
29 33 1 0 0 0 0
30 31 1 0 0 0 0
30 32 1 0 0 0 0
33 34 1 0 0 0 0
33 35 1 0 0 0 0
35 36 2 0 0 0 0
35 37 1 0 0 0 0
37 38 1 0 0 0 0
38 39 2 0 0 0 0
38 40 1 0 0 0 0
40 41 1 0 0 0 0
41 42 1 0 0 0 0
42 43 1 0 0 0 0
43 44 1 0 0 0 0
43 45 2 0 0 0 0
M END
$$$$
2
BH-2L
BH-2L
46 46 0 0 0 0 0 0 0 0999 V2000
0.0000 0.0000 0.0000 O 0 0 0 0 0 0 0 0 0 0 0 0
0.0000 0.0000 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0
0.0000 0.0000 0.0000 N 0 0 0 0 0 0 0 0 0 0 0 0
0.0000 0.0000 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0
0.0000 0.0000 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0
0.0000 0.0000 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0
0.0000 0.0000 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0
0.0000 0.0000 0.0000 O 0 0 0 0 0 0 0 0 0 0 0 0
0.0000 0.0000 0.0000 O 0 0 0 0 0 0 0 0 0 0 0 0
0.0000 0.0000 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0
0.0000 0.0000 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0
0.0000 0.0000 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0

```

[illegible]

3 4 1 0 0 0 0  
4 5 1 0 0 0 0  
5 6 1 0 0 0 0  
6 7 1 0 0 0 0  
6 8 2 0 0 0 0  
9 10 1 0 0 0 0  
10 11 1 0 0 0 0  
10 12 1 0 0 0 0  
13 14 1 0 0 0 0  
13 21 1 0 0 0 0  
14 15 2 0 0 0 0  
14 16 1 0 0 0 0  
16 17 2 0 0 0 0  
16 18 1 0 0 0 0  
18 19 1 0 0 0 0  
19 20 1 0 0 0 0  
21 22 1 0 0 0 0  
21 23 1 0 0 0 0  
23 24 1 0 0 0 0  
23 25 1 0 0 0 0  
25 26 1 0 0 0 0  
25 28 1 0 0 0 0  
26 27 1 0 0 0 0  
28 29 1 0 0 0 0  
29 30 1 0 0 0 0  
30 31 1 0 0 0 0  
30 32 1 0 0 0 0  
32 33 1 0 0 0 0  
32 34 1 0 0 0 0  
34 35 1 0 0 0 0  
34 38 1 0 0 0 0  
35 36 1 0 0 0 0  
35 37 1 0 0 0 0  
38 39 1 0 0 0 0  
39 40 2 0 0 0 0  
39 41 1 0 0 0 0  
41 42 1 0 0 0 0  
42 43 1 0 0 0 0  
43 44 1 0 0 0 0  
44 45 1 0 0 0 0

```
44 46 2 0 0 0 0
```

```
M END
```

```
$$$$
```

## 参考文献

- [1] Y. Shi, J. Zhu, N. A. Azam, K. Haraguchi, L. Zhao, H. Nagamochi, and T. Akutsu. A two-layered model for inferring chemical compounds with integer programming, J. Mol. Sci. [submitted].