# 化学反応経路ネットワークにおける ZDD を用いたエネルギー制限付き経路列挙

鈴木 浩史<sup>1,a)</sup> 中野 裕太<sup>3,b)</sup> 住谷 陽輔<sup>2,c)</sup> 湊 真-4,d) 前田 理<sup>2,e)</sup>

概要:化合物にはひとつの組成に対して様々な分子構造が存在し、それぞれで異なる性質を有する.化学 反応における反応経路ネットワークとは、分子構造を頂点とし、遷移可能な分子構造の間に辺を引いたグラフ構造を指す.反応経路ネットワークの解析は、反応設計に携わる化学者の助けとなる重要なタスクである.本稿では、分子構造間の遷移に必要なエネルギーに着目し、エネルギーを制限した反応経路ネットワークの上で、特定の分子構造を始点とする単純経路を列挙する.ただし、経路の総数は組合せ爆発を起こすため、明示的な列挙は避けなければならない.そこで、SIMPATH アルゴリズムにより、暗黙的に全経路を格納したゼロサプレス型二分決定グラフ (ZDD) という圧縮データ構造を構築する.さらに、ZDD が持つ効率的な絞込み機能を応用することで、エネルギーの上限に対する可能な経路の抽出を行う.

Hirofumi Suzuki $^{1,a)}$  Yuta Nakano $^{3,b)}$  Yosuke Sumiya $^{2,c)}$  Shin-ichi Minato $^{4,d)}$  Satoshi Maeda $^{2,e)}$ 

# 1. はじめに

近年、コンピュータの高速化や計算アルゴリズムの効率化により、量子化学計算は化学者にとって有用なツールとなりつつある。量子化学計算では、対象とする分子の構造(系内の原子の座標)が与えられると、電子の運動に関するシュレーディンガー方程式を解くことによって、その構造のポテンシャルエネルギー値を計算する。ポテンシャルエネルギー値は分子の構造によって変化する。このとき、構造を変数とするポテンシャルエネルギー値の関数をポテンシャルエネルギー曲面(PES: Potential Energy Surface)と呼ぶ。量子化学計算においては、化学反応は PES の曲面上のある地点から別の地点に至る「反応経路」としてモデル化される。

- 1 北海道大学 大学院情報科学研究科
- Sapporo, Hokkaido 060-0814, Japan <sup>2</sup> 北海道大学 大学院理学研究院
- Sapporo, Hokkaido 060-0810, Japan
- 3 北海道大学 工学部
- Sapporo, Hokkaido 060-8628, Japan
- 4 京都大学 大学院情報学研究科
- Kyoto, Kyoto 606-8501, Japan
- a) h-suzuki@ist.hokudai.ac.jpb) nakano@art.ist.hokudai.ac.jp
- $^{\rm c)}$  y.sumiya@sci.hokudai.ac.jp
- d) minato@i.kyoto-u.ac.jp
- e) smaeda@eis.hokudai.ac.jp

近年、PES上の反応経路を系統的かつ効率的に自動探索する GRRM [3] と呼ばれる画期的な計算技法が開発されている。 GRRM では、PES の曲面上に存在するポテンシャルエネルギーの極小点(平衡点; EQ)と、EQ間の山を越えて遷移するための峠(遷移状態; TS)という 2 種類の特徴点を、量子化学計算の最新の技法によりすべて導出し、EQ を頂点、TS を辺とするグラフ構造として抽象化することができる。このグラフは「反応経路ネットワーク」と呼ばれ、化学反応はこのグラフ上での EQ から別の EQ に至る経路を探索する問題に帰着される。

図1に、 $C_4H_6$ の反応経路ネットワークを示す.このネットワークは,65個の頂点と 225個の辺から構成される.頂点の中には,ブタジエンやシクロペンテンなど,良く知られている分子も多数含まれる.辺を辿ると,それらの間の異性化反応の経路を議論することができる.また,頂点や辺の色は,対応する EQ および TS のポテンシャルエネルギー値を示しており,異性化に必要なエネルギー(熱反応の場合には温度)を知ることができる.例として,1-ブチンから 1,3-ブタジエンへと至る経路の中で,エネルギー的に最も有利な経路上でのポテンシャルエネルギーを模式的に表したグラフを図 2に示す.熱反応では,ポテンシャルエネルギー値が最小の EQ と最大の TS のポテンシャルエネルギー差が反応の起こりやすさを決定づける.つまり,

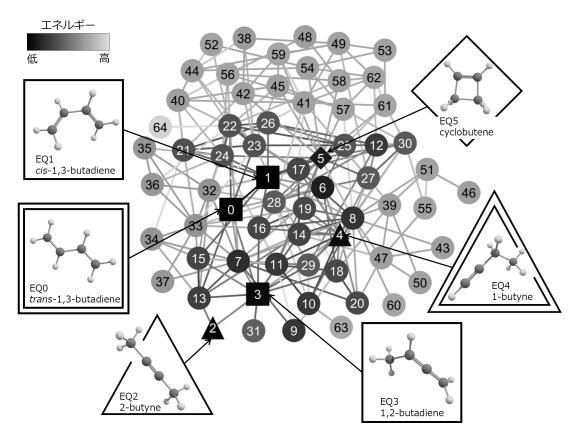


図 1  $C_4H_6$  の反応経路ネットワーク

ポテンシャルエネルギー値の最大一最小差ができるだけ小さく,かつ,有用な化学変換(ありふれた分子から価値の高い分子への変換)に対応する経路が,反応経路ネットワークから抽出すべき経路である.

図1程度の複雑さであれば、熟練者が目で見て最適な経路を選び出すことも不可能ではないかもしれない. しかしながら、EQやTSの数は、原子数に対して指数関数的に増大することが知られている。例えば、 $C_6H_6$  [9] では EQの数は 2004 個にもなってしまうことが知られている。デバイスや医薬などで重要となる分子の原子数はさらに大きく、反応経路ネットワークは非常に複雑である。従って、反応経路ネットワークの効率的な解析手段が必要である。そこで本研究では、反応経路ネットワークに含まれるすべての経路をゼロサプレス型二分決定グラフ (ZDD) [5] で表現し、そこから経路の重要度ランキングを作るための新たな方法として、エネルギー制限付き経路列挙の手法を開発した。

以下の本文では、2章で準備としてエネルギー制限付き 経路と ZDD および経路列挙アルゴリズムの説明を行う.3 章では提案アルゴリズムの説明をし、4章でその実験を行う.5章で本稿をまとめる.

### 2. 準備

本章では、諸定義と本稿の目的を形式的に説明する。その後、提案手法に用いるデータ構造である ZDD、および

ZDD を用いた経路列挙アルゴリズム SIMPATH [2] を紹介する.

## 2.1 エネルギー制限付き経路

頂点集合を V, 辺集合を  $E\subseteq \{\{u,v\}\mid u,v\in V\}$  とする グラフを G=(V,E) と書く. エネルギー関数  $w\colon E\to \mathbb{R}_+$  が定まっているとする. 反応経路ネットワークにおいて, 各頂点  $v\in V$  は EQ, 各辺  $e\in E$  は TS, w(e) は e が表す TS のエネルギーにそれぞれ対応する.

グラフGの異なる二項点 $s,t \in V$ について、s-t 単純経路集合を $\mathcal{P}_G(s,t) \subseteq 2^E$  とする.すなわち,各辺部分集合 $\mathcal{P} \in \mathcal{P}_G(s,t)$  はs-t 単純経路を成す.任意の実数 $\delta \in \mathbb{R}_+$  に対して,単純経路 $P \in 2^E$  は  $\max_{e \in P} w(e) \leq \delta$  であるとき, $\delta$ -制限付き経路であると言う.P が $\delta$ -制限付き経路であるときおよびそのときに限り, $C_G(P;\delta) = 1$  と書く.

ある頂点  $s \in V$  を始点とする任意の単純経路集合を

$$\mathcal{P}_G(s) := \bigcup_{t \in V - s} \mathcal{P}_G(s, t) \tag{1}$$

とし、エネルギー関数 w の下で s を始点とする δ-制限付き 経路の集合を

$$\mathcal{P}_G(s;\delta) := \{ P \in \mathcal{P}_G(s) \mid C_G(P;\delta) \} \tag{2}$$

とする. G 上のエネルギー値の集合  $W_G := \{w(e) \mid e \in E\}$ を定義する.

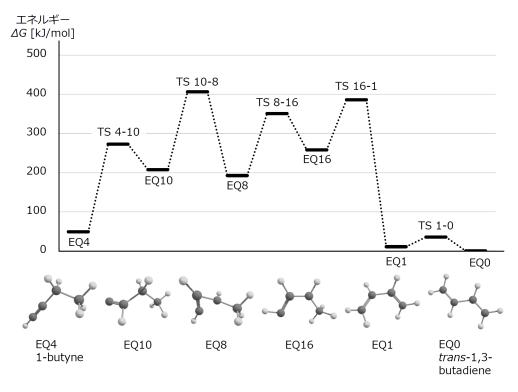


図 2 1-ブチンから 1,3-ブタジエンへと至る経路の中で,エネルギー的に最も有利な経路上でのポテンシャルエネルギー

本稿の目的は,グラフGとエネルギー関数wおよび始点  $s \in V$  が与えられたとき,各 $\delta \in W_G$  について $\mathcal{P}_G(s;\delta)$ を求めることである.しかし,考えうる経路は辺数|E|に対して指数的な数存在しうるため,明示的な計算を避けなければならない.

## 2.2 Zero-suppressed Binary Decision Diagram

 $\mathcal{P}_G(s;\delta)$  を暗黙的に計算し管理するために,ZDD と呼ばれる集合族を圧縮表現するデータ構造を用いる.ZDD は節点集合 N と枝集合 A からなる有向非巡回グラフ Z=(N,A) である.Z はちょうど一つの根  $\rho\in N$  と二つの終端  $\bot$ ,  $\top\in N$  を持つ.ここで,辺集合 E 上に全順序  $e_1<\ldots< e_m$  (m=|E|) が定められていると仮定する.終端を除く各節点  $\alpha\in N\setminus\{\bot,\top\}$  は,ラベル  $\ell(\alpha)\in E$  と 0-/1-枝と呼ばれる二つの出枝を持つ. $\alpha$  の x-枝  $(x\in\{0,1\})$  に接続する節点を  $\alpha$  の x-子と呼び  $\alpha_x$  と書く.このとき, $\alpha_x$  が終端でないならば  $\ell(\alpha)<\ell(\alpha_x)$  が成り立つ.

ZDD は次のように集合族を表現する.  $\rho$  から  $\top$  への一つの有向パスが一つの集合を表現する. パスが  $\alpha$  の 0- (1-枝) を辿るとき,集合は  $\ell(\alpha)$  を含まない (含む).

ZDD には既約な形が存在する. 以下の二つの操作を任意の順序で可能な限り行えばよい.

- $\alpha_1 = \bot$  であるならば  $\alpha$  を削除する.
- 二つの節点  $\alpha, \alpha'$  について, $\alpha_x = \alpha_x'$   $(x \in \{0,1\})$  か つ  $\ell(\alpha) = \ell(\alpha')$  ならば, $\alpha$  と  $\alpha'$  を一つの節点にまと める.

これらの操作は、ZDD から冗長な節点を減らす。図 3 に  $\mathcal{P}_G(s)$  を表現する既約な ZDD の例を示す.

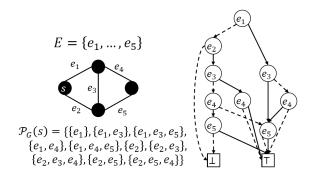


図 3  $\mathcal{P}_G(s)$  を表現する既約な ZDD の例

#### 2.3 SIMPATH

SIMPATH [2] は ZDD を用いた経路列挙アルゴリズムで、Knuth により考案され、その派生アルゴリズムが近年様々な分野に応用されている [4]、[8] . 辺集合 E に全順序が定義されたグラフ G と異なる二頂点  $s,t \in V$  が与えられたとき、SIMPATH は  $P_G(s,t)$  の ZDD を構築する.

古典的な列挙アルゴリズム (例えば [6]) では,可能な s-t 単純経路を一つずつ出力する.そのため,アルゴリズムの計算時間は解の個数  $|P_G(s,t)|$  に依存してしまう.一方で,SIMPATH はトップダウンな動的計画法の要領で,単純経路を明示的に出力することなく ZDD を構築する.これに

IPSJ SIG Technical Report

より、計算時間はできあがる ZDD のサイズ (節点数 |N|) に依存し、解の個数が多くても現実的な時間で列挙できる可能性が広がる。特に、SIMPATH の計算量に関連する議論を [1] に見ることができ、グラフのパス幅というパラメータが小さければ、SIMPATH は効率良く動作する。

## 3. アルゴリズム

ここでは、 $\delta$ -制限付き経路の性質を利用して、すべての  $\delta \in W_G$  に対して  $\mathcal{P}_G(s;\delta)$  の ZDD を効率良く構築するアルゴリズムを提案する.提案アルゴリズムは、SIMPATH を用いて  $\mathcal{P}_G(s)$  の ZDD を構築した後,逐次的に  $\mathcal{P}_G(s;\delta)$  の ZDD に変換する.

#### 3.1 始点のみ指定された経路集合に対する ZDD 構築

SIMPATH は異なる二頂点  $s,t\in V$  に対して  $\mathcal{P}_G(s,t)$  の ZDD を構築するため, $\mathcal{P}_G(s)$  の ZDD を工夫無しに直接構築することはできない.式 (1) に従えば,各頂点  $t\in V-s$  に対して  $\mathcal{P}_G(s,t)$  の ZDD を構築し,それらの共通集合演算を行えば  $\mathcal{P}_G(s)$  の ZDD を得られる.しかし,SIMPATH と共通集合演算をそれぞれ |V|-1 回ずつ適用する必要があるため,効率的とは言えない.そこで,一回の SIMPATHで  $\mathcal{P}_G(s)$  の ZDD を構築する手法を考える.

V に含まれないダミー頂点 t' を用意し、新しい頂点集合  $V':=V\cup\{t'\}$  を作る。各頂点  $v\in V$  と t' を繋ぐ辺を考え、新しい辺集合  $E':=E\cup\{\{v,t'\}\mid v\in V\}$  を作る。V' と E' からなるグラフ G'=(V',E') を考える。

G' の各辺部分集合  $P' \in 2^{E'}$  について,

$$P' \in \mathcal{P}_{G'}(s, t') \Rightarrow (\exists v \in V, P' - \{v, t'\} \in \mathcal{P}_{G}(s, v))$$
 (3)

が成り立つ.一方で,G とある頂点  $v \in V$  について,s-v 単純経路が存在するならば,G' 上で辺  $\{v,t'\}$  を通る s-t' 単純経路が存在する.よって,各辺部分集合  $P \in 2^E$  について,

$$P \in \mathcal{P}_G(s) \Rightarrow (\exists v \in V, P \cup \{\{v, t'\}\}\} \in \mathcal{P}_{G'}(s, t')) \quad (4)$$

が成り立つ。(3) と (4) は, $\mathcal{P}_G(s)$  と  $\mathcal{P}_{G'}(s,t')$  における単純経路の一対一対応が存在することを意味する.よって, $\mathcal{P}_G(s)$  の ZDD を構築する代わりに  $\mathcal{P}_{G'}(s,t')$  の ZDD を構築すれば十分である.これは,G' を構成し SIMPATH を一回適用すればよい(Algorithm 1).

## **3.2 ZDD** の枝消去

各頂点  $v \in V$  について, $w(\{v,t'\}) = -\infty$  であると仮定したとき,G' における  $\delta$ -制限付き s-t' 経路集合を

$$\mathcal{P}_{G'}(s, t'; \delta) := \{ P \in \mathcal{P}_{G'}(s, t') \mid C_{G'}(P; \delta) = 1 \}$$
 (5)

とする.  $\mathcal{P}_G(s)$  と  $\mathcal{P}_{G'}(s,t')$  における単純経路の一対一対

### **Algorithm 1** ConstructZDD(G = (V, E), s)

- 1:  $E' \leftarrow E$
- 2: ダミー頂点 t' を作成
- 3:  $V' \leftarrow V \cup \{t'\}$
- 4: for  $v \in V$  do
- 5:  $E' \leftarrow E' \cup \{\{v, t'\}\}\$
- 6: end for
- 7:  $G' \leftarrow (V', E')$
- 8:  $Z \leftarrow \text{SIMPATH}(G', s, t')$
- 9: return Z

応から, $\mathcal{P}_G(s;\delta)$  と  $\mathcal{P}_{G'}(s,t';\delta)$  の単純経路にも一対一対 応が存在する.よって, $\mathcal{P}_{G'}(s,t';\delta)$  の ZDD を構築すれば 十分である.

愚直には、G'とある  $\delta \in W_G$  に対して、エネルギーが  $\delta$  より大きい辺を消去したグラフ  $G'_{\delta} = (V, E' \setminus \{e \in E \mid w(e) > \delta\})$  を考えることができる。すると、 $\mathcal{P}_{G'_{\delta}}(s,t') = \mathcal{P}_{G'}(s,t';\delta)$  が成り立つ。よって、各  $\delta \in W_G$  に対して  $G'_{\delta}$  を構成し SIMPATH を適用すればよい。しかし、SIMPATH を |E| 回適用する必要があり、効率的とは言えない。そこで、各  $\delta \in W_G$  に対して、 $\mathcal{P}_{G'}(s,t')$  の ZDD から  $\mathcal{P}_{G'}(s,t';\delta)$  の ZDD を構築することを考える。

まず, $P_{G'}(s,t')$  の ZDD を元に,辺  $e \in E$  を通ることを禁止した単純経路集合  $\{P \in \mathcal{P}_{G'}(s,t') \mid e \notin P\}$  の ZDD を構築する手法を与える.これは, $\rho$  から  $\top$  への有向パスであって,ラベル e を持つ節点の 1-枝を辿るものがあれば,それらが  $\bot$  に辿り着くように ZDD を変換すればよい.その実現には,ラベル e を持つ節点から出る 1-枝の先を  $\bot$  に変更すれば十分である(図 4).このとき,ZDD は既約でなくなるので,最後に既約化の処理を行うとよい.この操作を e の枝消去と呼ぶことにし,Algorithm 2 に擬似コードを示す.ここで, $N_e := \{\alpha \in N \mid \ell(\alpha) = e\}$  である.

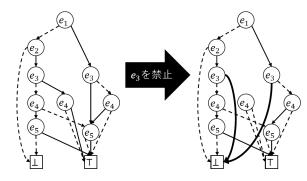


図 4 枝消去の例

次に,E の各辺をエネルギーw の降順に並べたもの $e'_1,\dots,e'_m$   $(i < j \Leftrightarrow w(e'_i) > w(e'_j))$  を考える.これは,ZDD の集合族表現における辺の全順序とは異なる場合があることに注意せよ.すると, $\mathcal{P}_{G'}(s,t';w(e'_i))$  の ZDD に対して $e'_i$  の枝消去を行うことで, $\mathcal{P}_{G'}(s,t';w(e'_{i+1}))$  の ZDD を得ることができる.よって, $e'_i$  の枝消去を $i=1,\dots,m$ 

IPSJ SIG Technical Report

の順に行っていくことで、各  $\delta \in W_G$  に対し  $\mathcal{P}_{G'}(s,t';\delta)$  の ZDD が構築される (Algorithm 3) .

$\overline{\textbf{Algorithm 2} \text{ DeleteArcs}(Z, e)}$	
1: for $\alpha \in N_e$ do	
2: α の 1-子を ⊥ に変更	
3: end for	
4: Z を既約化	
5: return $Z$	

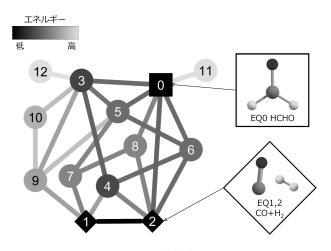
Algorithm	3	EnergyRestrictionPaths	(G = 1)	$(V_{\cdot})$	,E	),	w,	s	)

- 1:  $Z \leftarrow \text{ConstructZDD}(G, s)$
- 2: 各 $\delta \in W_G$  について,  $Z_\delta$  を空の ZDD とする
- 3: 辺集合 E 上のエネルギー w の降順  $e_1',\dots,e_m'$  を求める
- 4: for  $i=1,\ldots,m$  do
- 5:  $Z_{w(e'_i)} \leftarrow Z$
- 6:  $Z \leftarrow \text{DeleteArcs}(Z, e'_i)$
- 7: end for
- 8: **return**  $\{Z_{\delta} \mid \delta \in W_G\}$

# 4. 実験

提案手法の性能を評価するために計算機実験を行った. 実験に用いたプログラムは C++ (g++4.8.4 と -O3 オプション)で記述した. 実験環境は OS が 64-bit Ubuntu 16.04 LTS, CPU が Intel Core i7-3930K 3.2 GHz, RAM が 64 GB である.

まず、図 5 に示す HCHO の反応経路ネットワーク (13 頂点、24 辺のグラフ)で、EQ0 を始点として提案手法を適用した。計算時間は 0.02 秒と高速で、メモリ使用量は 5MB であった。得られた結果を表 1 に示す。提案手法を用いれば、表のように各エネルギー上限に対する経路の総数を知ることができる。また、エネルギー上限に対して新たに到達可能になる頂点 (EQ) は容易にわかる。すなわち、経路そのものや EQ をエネルギーによりランキング付けすることができる。



**図 5** HCHO の反応経路ネットワーク

表 1 HCHO の反応経路ネットワークにおける経路列挙の結果

7 2 11 1 1 1 TH	4447	<b>♦</b> ∇ □ <i>₽</i> <b>♦<i>₽</i><b>.</b>₩₩</b>	ZDD	新たに到達
エネルギー上限	刈心辺	経路総数	サイズ	できる EQ
-114.479690778	1-2	0	0	
-114.360182668	3-4	0	0	
-114.356884423	0-2	2	5	1, 2
-114.349909324	0-3	4	10	3, 4
-114.348927542	5-6	4	10	
-114.348925212	0-5	6	15	5, 6
-114.348898752	0-6	8	18	
-114.348865455	2-6	14	29	
-114.348358396	4-6	24	46	
-114.338657933	3-5	40	53	
-114.318570970	7-8	40	53	
-114.318562497	2-4	68	71	
-114.318531664	1-7	92	114	7, 8
-114.318502503	2-8	128	146	
-114.317956076	0-8	163	166	
-114.317642489	5-7	303	267	
-114.283423523	1-4	499	388	
-114.282725152	1-9	575	423	9
-114.282691930	3-9	925	729	
-114.249846963	3-10	1,026	886	10
-114.245566359	5-9	1,574	1,100	
-114.241175238	9-10	2,318	1,266	
-114.207659484	0-11	2,319	1,268	11
-114.205399288	3-12	2,546	1,419	12

次に、 $C_4H_6$  の反応経路ネットワークでも同様の実験を行った。しかし、図 1 で示した 65 頂点 225 辺のネットワークでは、提案手法はメモリ不足により失敗した。そこで、化学的に意味のある範囲でネットワークを縮約 [7] し、42 頂点 204 辺となったネットワークに対して再度提案手法を適用した。その結果、提案手法は 2291 秒の計算時間と 39GB のメモリ使用量で成功した。エネルギーごとの経路総数を図 6 に、ZDD サイズを図 7 に示す。経路総数は最大で  $7.2 \times 10^{22}$  であった。図 6 からは、エネルギー値 300、400、および 600 付近で経路総数が爆発的に増加していることがわかる。図 7 と照らし合わせると、経路総数と ZDD サイズにはある程度の相関が見られる。しかし、ZDD サイズはおおよそ  $10^8$  程度で収まっている。経路総数と比べると  $10^{14}$  倍程度の圧縮が効いており、効率良いと言える。

#### **5.** おわりに

本稿では、化学反応における反応経路ネットワーク上で、 始点となる EQ が与えられたとき、エネルギー制限付きの 単純経路をすべて求める手法を提案した。提案手法は、集 合族を圧縮表現するデータ構造 ZDD に単純経路集合を格 納する。そのために、ZDD を用いた経路列挙アルゴリズ ム SIMPATH を適用するための工夫と、エネルギー制限を 反映するための枝消去という操作を行う。実際の反応経路 ネットワークを用いて実験を行ったところ、EQ 数 42 の

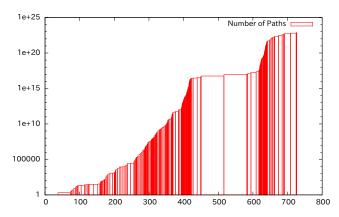


図  $\mathbf{6}$   $C_4H_6$  の縮約した反応経路ネットワークにおけるエネルギーご との経路総数 (横軸がエネルギー (kJ/mol), 縦軸が経路総数 であり、経路総数は対数スケールで表示している)

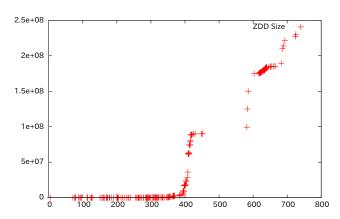


図 7  $C_4H_6$  の縮約した反応経路ネットワークにおけるエネルギー ごとの経路集合を表す ZDD のサイズ (横軸がエネルギー (kJ/mol), 縦軸が ZDD サイズ)

ネットワークにおいて、最大で $7.2 \times 10^{22}$ 個もの単純経路 を求めることに成功した. さらに, エネルギー制限の変化 に対する経路総数の変化を知ることができた. これによ り、経路や EQ に対してエネルギーによるランキング付け が可能となる.

今後は、提案手法により得ることができる情報を、実際 の反応設計に活用する手順について検討したい. ZDD に は様々な集合族処理機能 [5] があるため、ランキング付け の後に重要な経路を絞り込むことにも応用できる. また, より大きな反応経路ネットワークに適用するために、アル ゴリズムの改善を試みたい.

#### 謝辞

反応経路ネットワークの図解や実データの提供, および 様々な助言をしてくださった, 北海道大学 大学院理学研究 院の 原渕 祐 助教と大学院総合化学院の 杉山 佳奈美さん に深く感謝いたします.

## 参考文献

- Inoue, Y. and Minato, S.: Acceleration of ZDD Construction for Subgraph Enumeration via Path-width Optimization, TCS-TR-A-16-80. Hokkaido University (2016).
- Knuth, D. E.: The art of computer programming: Bitwise tricks & techniques; binary decision diagrams, volume 4, fascicle 1 (2009).
- Maeda, S., Harabuchi, Y., Takagi, M., Saita, K., Suzuki, K., Ichino, T., Sumiya, Y., Sugiyama, K. and Ono, Y.: Implementation and performance of the artificial force induced reaction method in the GRRM17 program, Journal of Computational Chemistry, Vol. 39, No. 4, pp. 233–251 (2018).
- [4] Maehara, T., Suzuki, H. and Ishihata, M.: Exact Computation of Influence Spread by Binary Decision Diagrams, Proceedings of the 26th International Conference on World Wide Web, WWW '17, pp. 947-956 (2017).
- Minato, S.: Zero-Suppressed BDDs for Set Manipulation in Combinatorial Problems, DAC, pp. 272–277 (1993).
- Read, R. and Tarjan, R.: Bounds on Backtrack Algorithms for Listing Cycles, Paths, and Spanning Trees, Vol. 5, pp. 237–252 (1975).
- Sumiya, Y., Taketsugu, T. and Maeda, S.: Full rate constant matrix contraction method for obtaining branching ratio of unimolecular decomposition, Journal of computational chemistry, Vol. 38 2, pp. 101–109 (2017).
- Takizawa, A., Miyata, Y. and Katoh, N.: Enumeration of Floor Plans Based on a Zero-Suppressed Binary Decision Diagram, International Journal of Architectural Computing, Vol. 13, No. 1, pp. 25–44 (2015).
- Tokoyama, H., Yamakado, H., Maeda, S. and Ohno, K.: Exploration of Isomers of Benzene by GRRM/SCC-DFTB, Chemistry Letters, Vol. 43, No. 5, pp. 702-704 (2014).