二分決定グラフによる列挙の幾何への応用
～正多面体の列挙～

堀山 貴史 (埼玉大学)

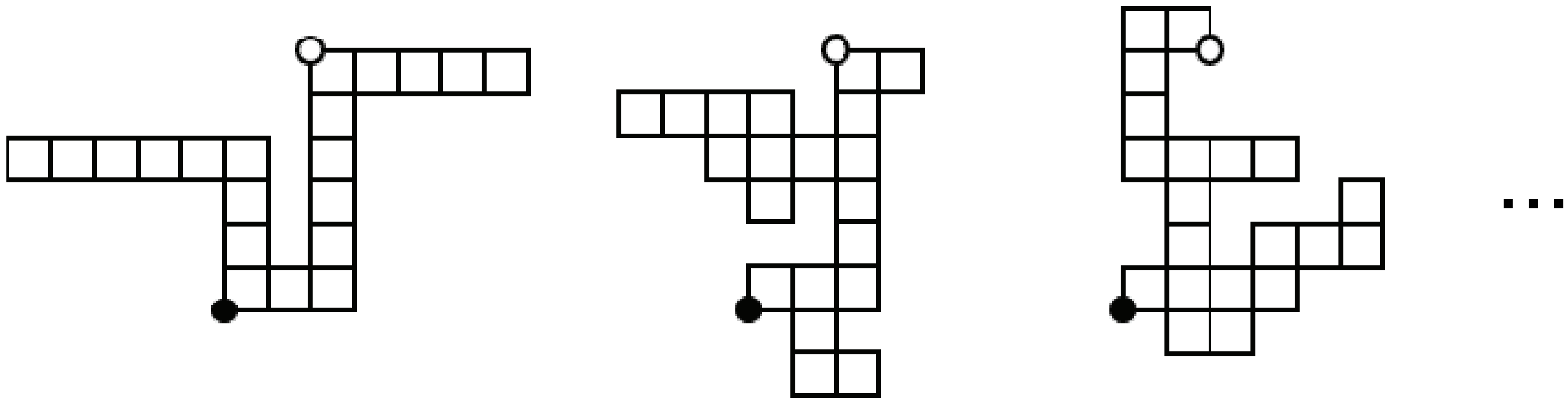
第1回 離散構造処理系シンポジウム (2010年 5月 28日)

研究テーマ

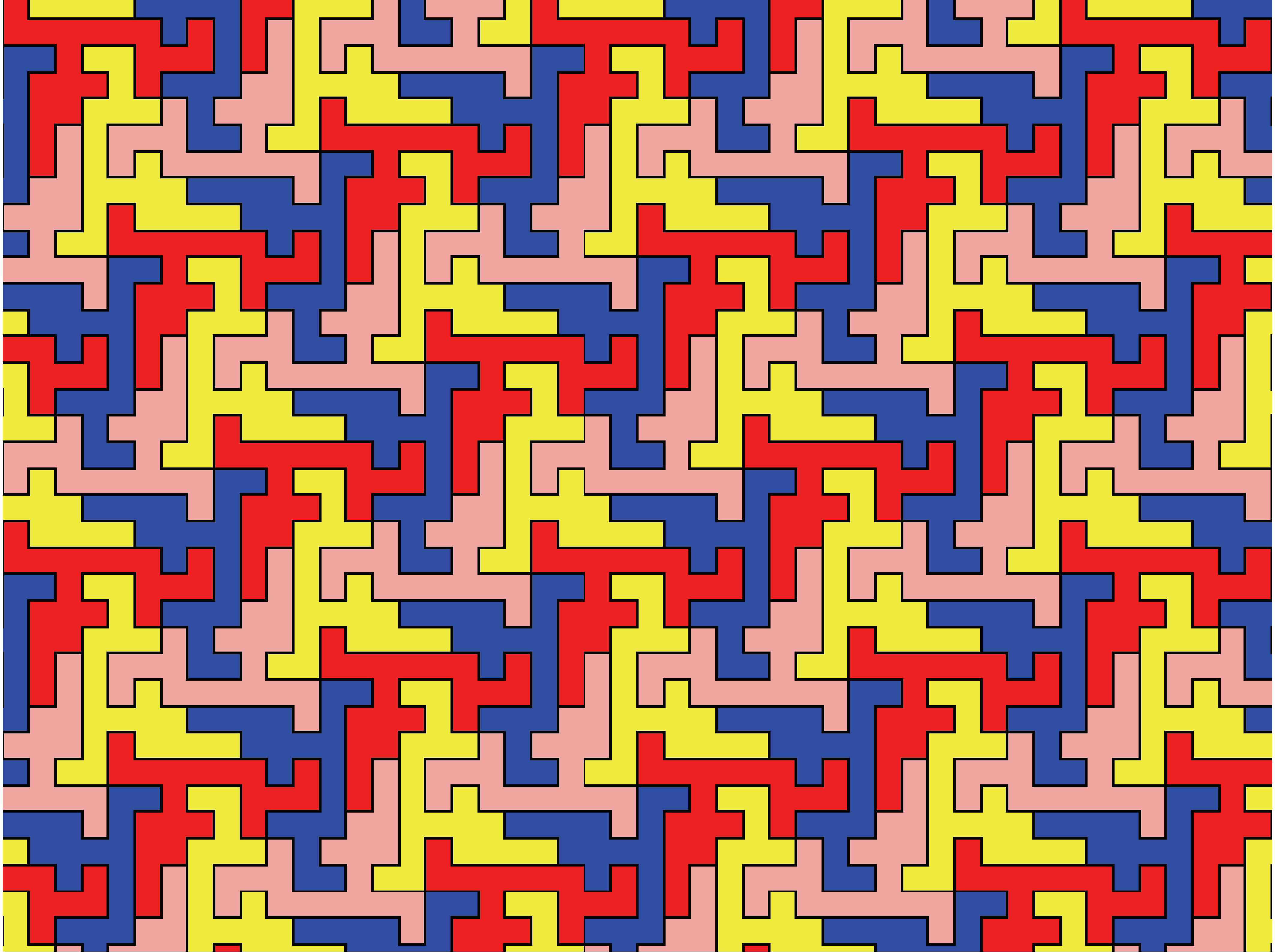
- BDD 関連
  - 乗算の各ビットを表す BDD のサイズは、どんな変数順序でも指数　[Bryant 1991]
  - 除算の各ビットも指数
  - データを BDD で表して、推論
    - 与えられた BDD は単調関数か？、Horn 関数か？
    - 性質 f が成立するか？
    - 事象 x_q が成立する条件？ (x_a x_b … → x_q となる x_a x_b … を探す)
- VLSI 設計 関連
- オンライン・アルゴリズム 関連
- オークション 関連
- 列挙アルゴリズム 関連
  - タイリングの列挙

タイリングの列挙

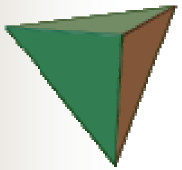
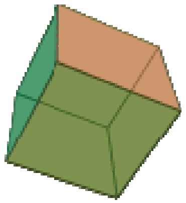
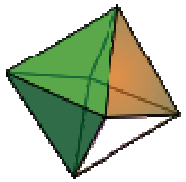
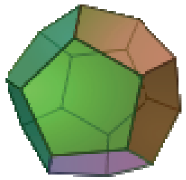
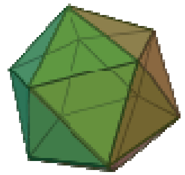
タイリング図形を次々と見せてくれるウェブページを用意しました
http://www.al.ics.saitama-u.ac.jp/horiyama/research/

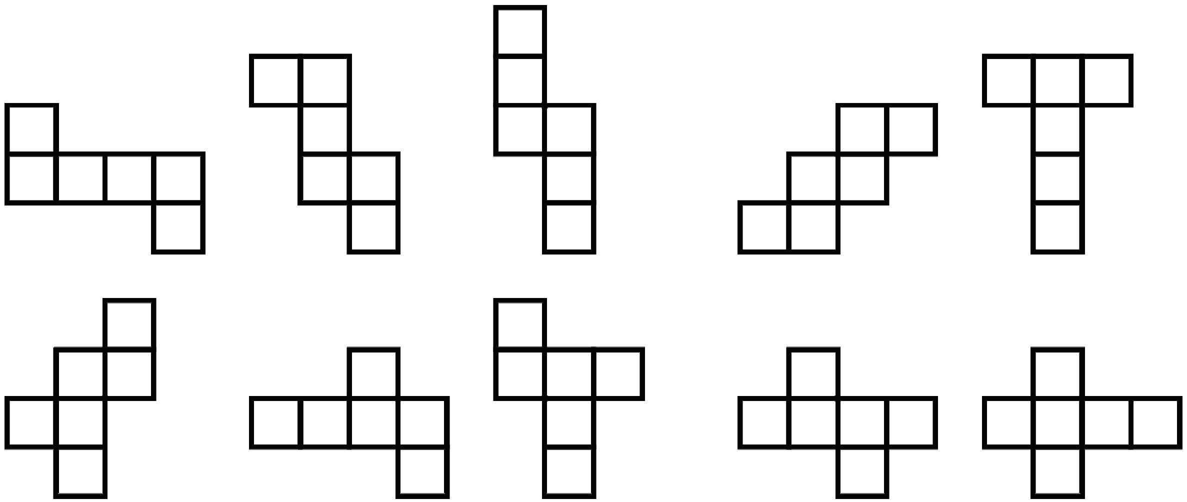


タイリング：
絵画 (エッシャーなど)、織物、壁面装飾、工業製品のデザイン、結晶物理学など
身の回りに色々な形で見られます
エッシャーの絵は http://www.mceschers.com/ に沢山あります



正多面体の展開図の列挙

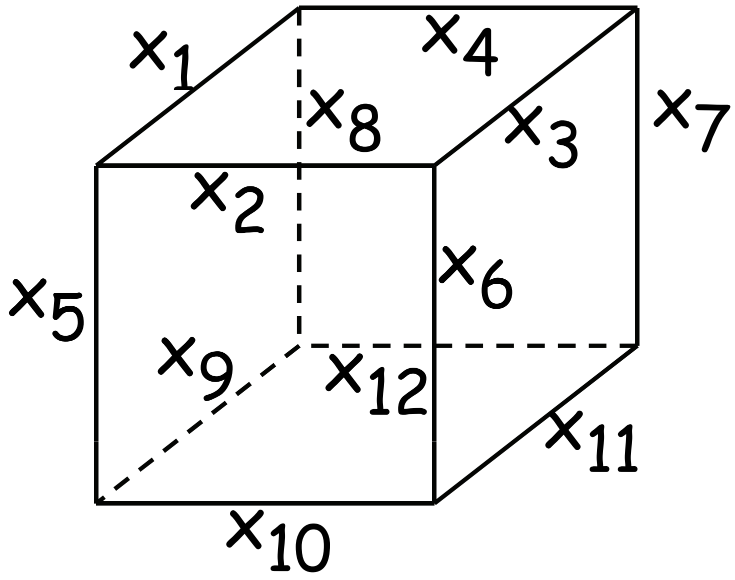
正四面体	正六面体	正八面体	正十二面体	正二十面体
				
2 種類	11 種類	11 種類	43,380 種類	43,380 種類



多面体の図は wikipedia より
http://en.wikipedia.org/wiki/Platonic_solid

正多面体の展開図の列挙

- 各辺の「切る」/「切らない」を、論理変数で表す



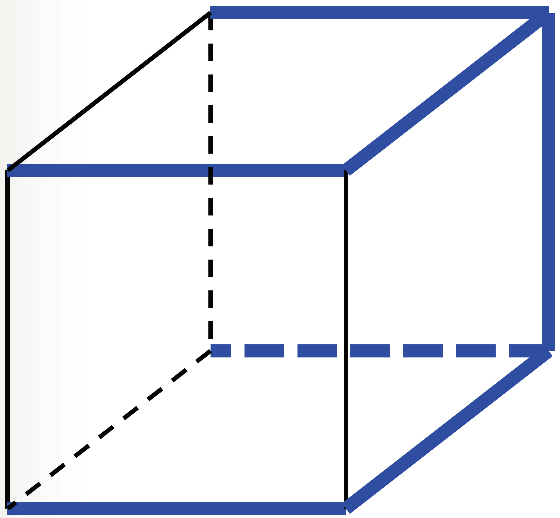
- **展開図**になるための**制約条件**を、BDD で表す
  → 1-path 1 つが、展開の仕方 1 つに対応
- **対称性を排除**する
- 「切る」/「切らない」をもとに、**展開図を作成**

展開図になるための制約条件

- 展開図

⇔ 「切る」辺が、全域木になる [文献 (?)]

⇔ 1. 「切る」辺は、頂点数 - 1 本
   2. 各頂点には、「切る」辺が少なくとも 1 本
   3. 「切る」辺がサイクルを持たない

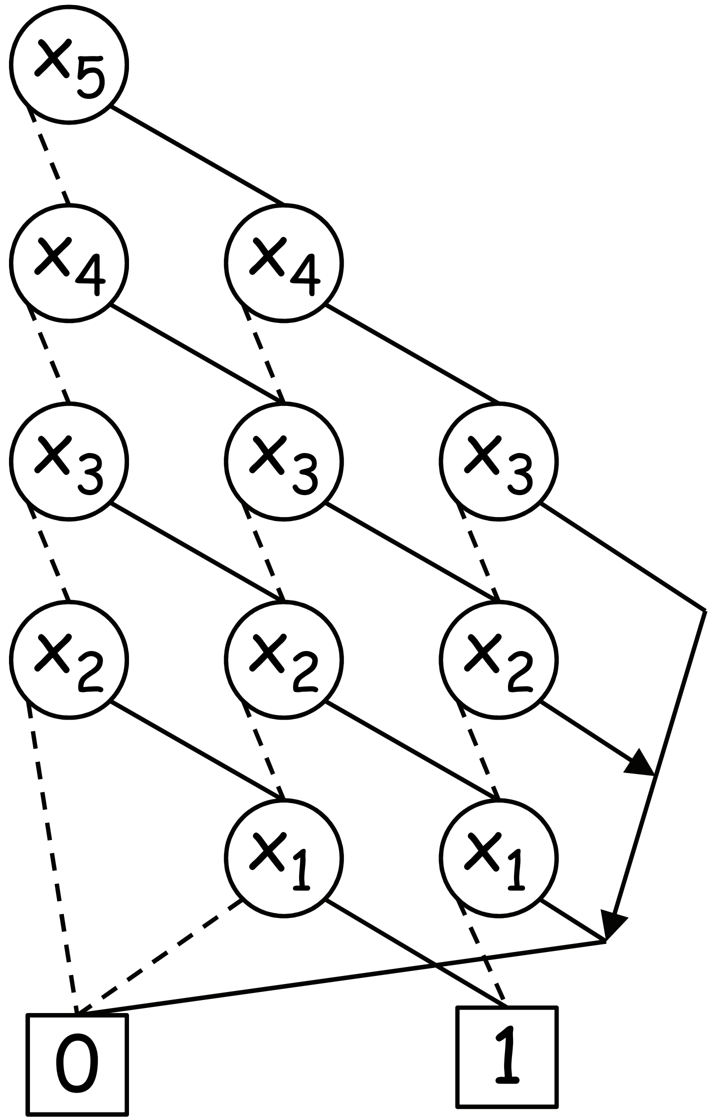


展開図になるための制約条件 (1)

- 「切る」辺は、頂点数 - 1 本

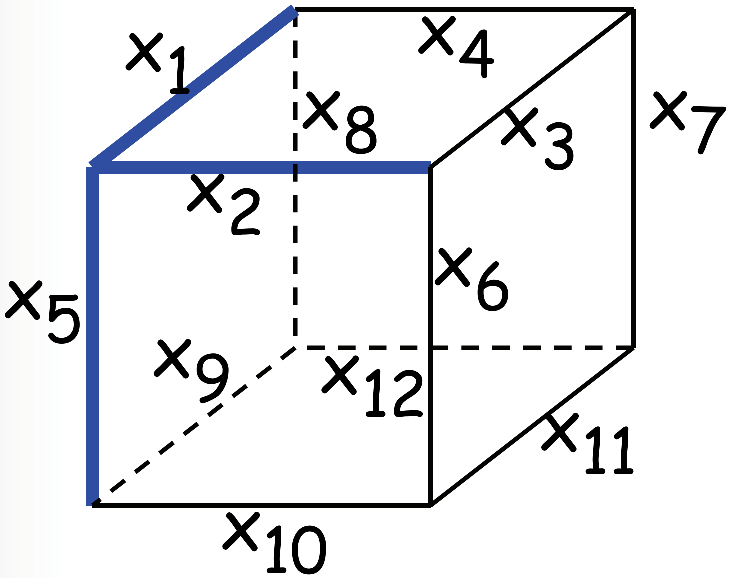
Ex) 2 out of
5 possible positions

1 の個数を数える



展開図になるための制約条件 (2)

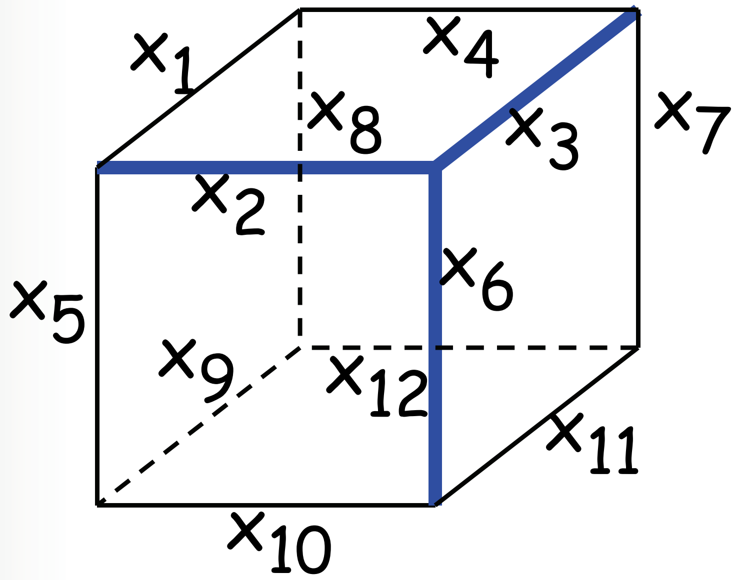
- 各頂点には、「切る」辺が少なくとも 1 本



$(x_1 \vee x_2 \vee x_5)$

展開図になるための制約条件 (2)

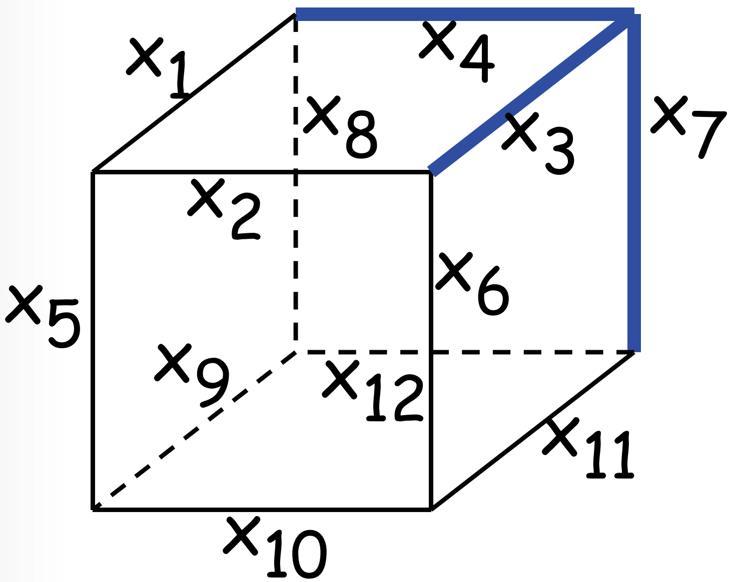
- 各頂点には、「切る」辺が少なくとも 1 本



$$(x_1 \vee x_2 \vee x_5)$$
$$\wedge (x_2 \vee x_3 \vee x_6)$$

展開図になるための制約条件 (2)

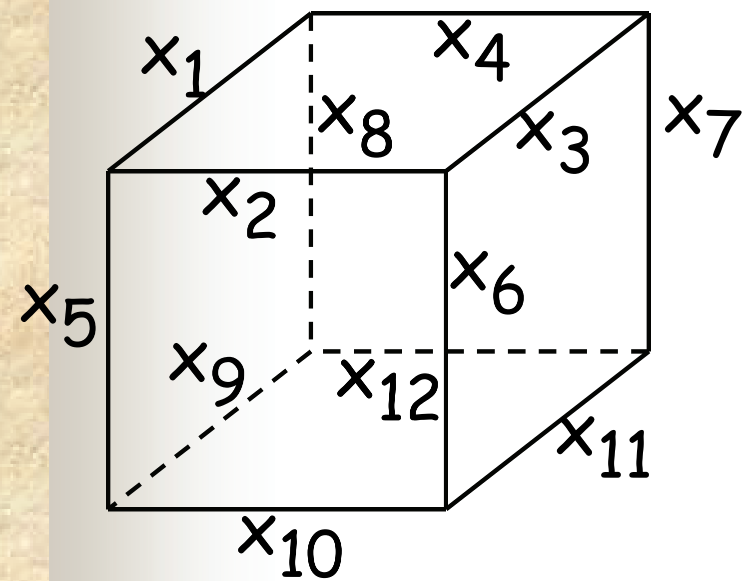
- 各頂点には、「切る」辺が少なくとも 1 本



$$
\begin{aligned}
&(x_1 \lor x_2 \lor x_5) \\
\land\ &(x_2 \lor x_3 \lor x_6) \\
\land\ &(x_3 \lor x_4 \lor x_7) \\
\land\ &\cdots
\end{aligned}
$$

展開図になるための制約条件（3）

- 「切る」辺がサイクルを持たない
  = ￢（「切る」辺がサイクルを持つ）



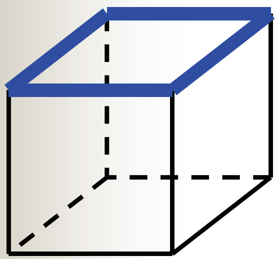
サイクル
… 各面の周囲を貼り合わせたもの

→ 面を 1 つずつ加えて、
サイクルを列挙する

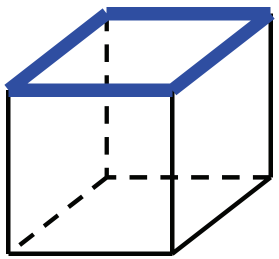
展開図になるための制約条件 (3)

- 「切る」辺がサイクルを持たない
  = ¬ (「切る」辺がサイクルを持つ)

追加する面

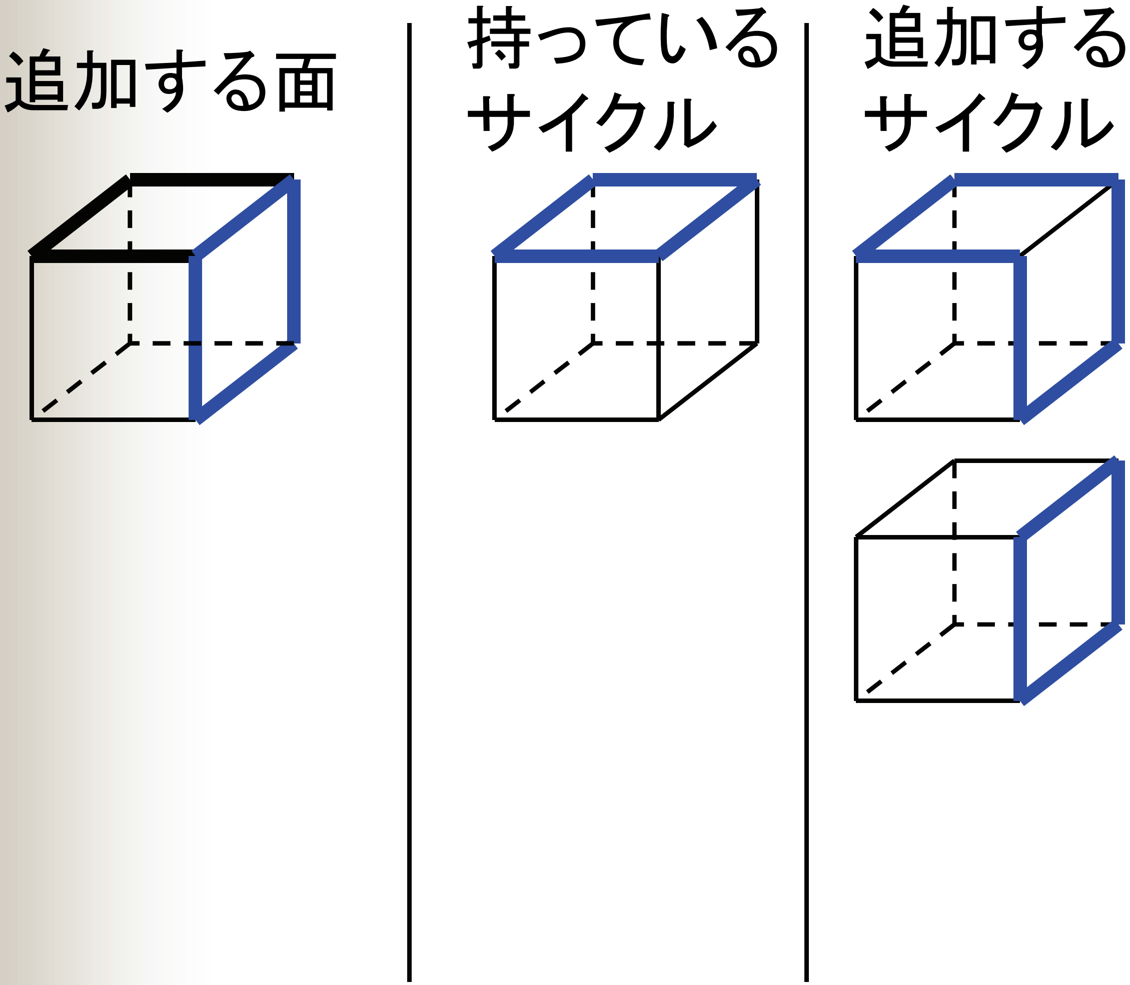


追加する
サイクル



展開図になるための制約条件（3）

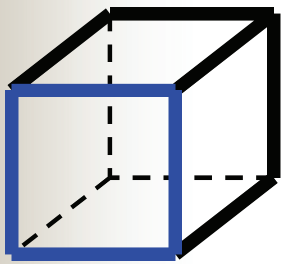
- 「切る」辺がサイクルを持たない
  = ￢(「切る」辺がサイクルを持つ)



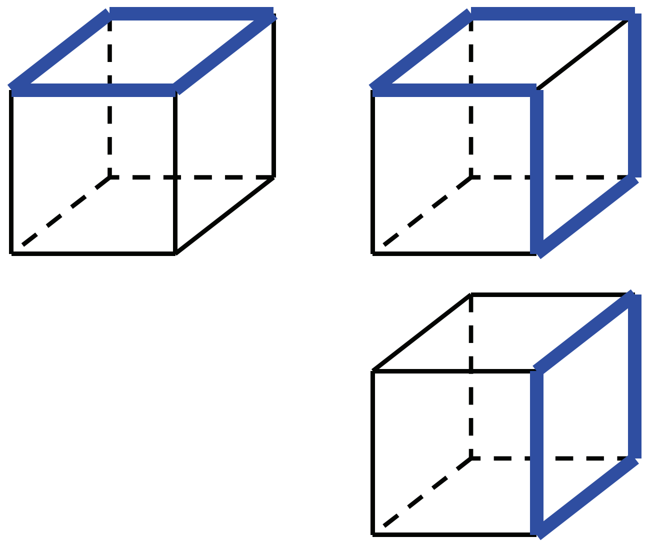
展開図になるための制約条件（3）

- 「切る」辺がサイクルを持たない
  = ￢(「切る」辺がサイクルを持つ)

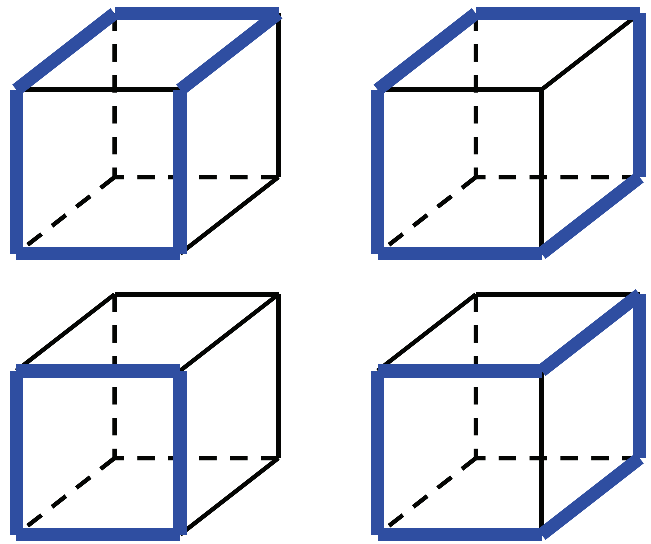
追加する面



持っている
サイクル



追加する
サイクル



追加するサイクル
・面の周囲
・持っているサイクルに、面の周囲で「切る」/「切らない」を反転

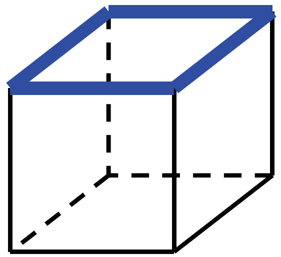
展開図になるための制約条件 (3)

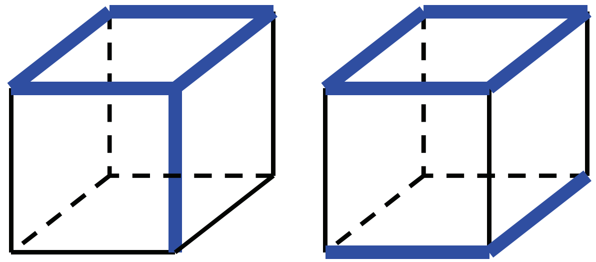
- 「切る」辺がサイクルを持たない
  = ¬(「切る」辺がサイクルを持つ)
- サイクルの列挙
  - $f_{サイクルの集合} := 0$
  - foreach 面 $(x_{i1}, x_{i2}, x_{i3}, x_{i4})$ {
    - $f_1 := x_{i1} \wedge x_{i2} \wedge x_{i3} \wedge x_{i4} \wedge$ 他変数の負リテラル
    - $f_2 := f_{サイクルの集合}$ のBDDで、
      $x_{i1}, x_{i2}, x_{i3}, x_{i4}$ の 0-枝 と 1-枝を入れ替える
    - $f_{サイクルの集合} := f_{サイクルの集合} \vee f_1 \vee f_2$

    }
  - $f_{サイクルの集合} := f_{サイクルの集合} \wedge \underbrace{(x_1 \vee x_2 \vee \cdots \vee x_n)}_{「全辺\ 切らない」を除外}$

展開図になるための制約条件 (3)

- 「切る」辺がサイクルを持たない
  = ¬ (「切る」辺がサイクルを持つ)
- サイクルの列挙





$f_{\text{サイクルの集合}}(111100000000) = 1$

$f_{\text{サイクルの集合}}(111101000000) = 0$

$f_{\text{サイクルの集合}}(111100000110) = 0$

- $f_{\text{サイクルの集合}}$ の monotone 拡大をとる

$f_{\text{monotone 拡大}}(1111********) = 1$

正多面体の展開図の列挙

✔ 展開図になるための制約条件を、BDD で表す
→ 1-path 1 つが、展開の仕方 1 つに対応

- 対称性を排除する
  - 辺の置換を定義し、対称の同値類で代表元を残す

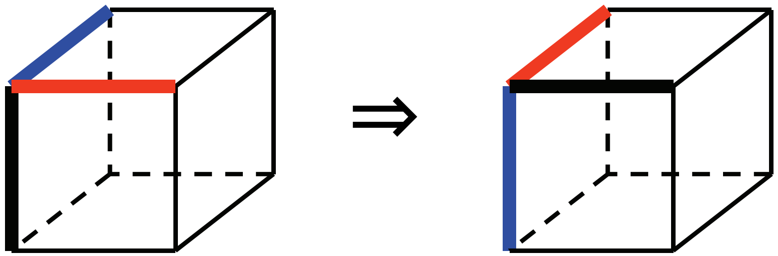
    例) 正二十面体の置換を定義する辺は、30 本
    置換は、120 個

- 「切る」/「切らない」をもとに、展開図を作成

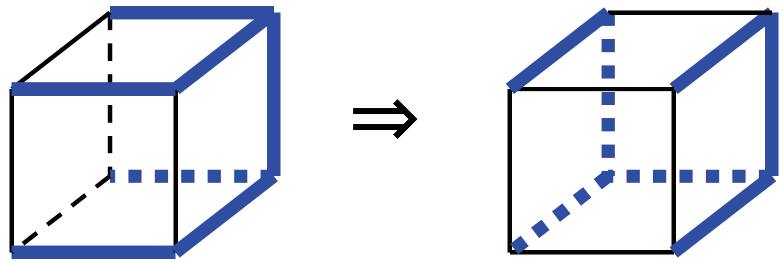
対称性の排除

- 置換による「切る」辺の移動

置換



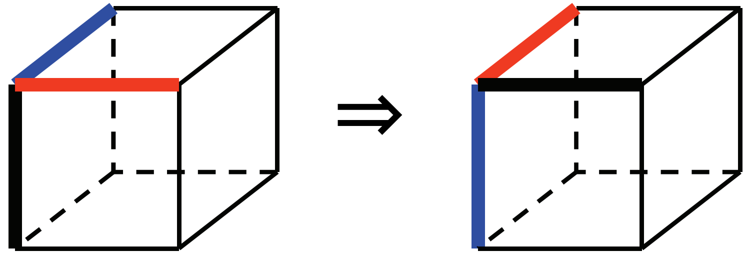
「切る」辺



$f_{\text{展開図}}(011100100111) = 1$　　$f_{\text{展開図}}(101000111011) = 1$

辞書順で大きい方を残す

対称性の排除

置換 

$f_{展開図}(011100100111) = 1$

$f_{展開図}(101000111011) = 1$

$\vdots$

$f := f_{展開図} \wedge (x_1 \equiv y_5)(x_2 \equiv y_1)(x_3 \equiv y_8) \cdots$

(置換による対応を作る)

$f(011100100111\ 101000111011) = 1$

$f(101000111011\ 000011110111) = 1$

$\vdots$

$f' := f \wedge ((x_1 \overline{y_1}) \vee (x_1 \equiv y_1)(x_2 \overline{y_2}) \vee (x_1 \equiv y_1)(x_2 \equiv y_2)(x_3 \overline{y_3}) \cdots)$

(x が y より大なら残す)

$\cancel{f'(011100100111\ 101000111011) = 1}$

$f'(101000111011\ 000011110111) = 1$

$\vdots$

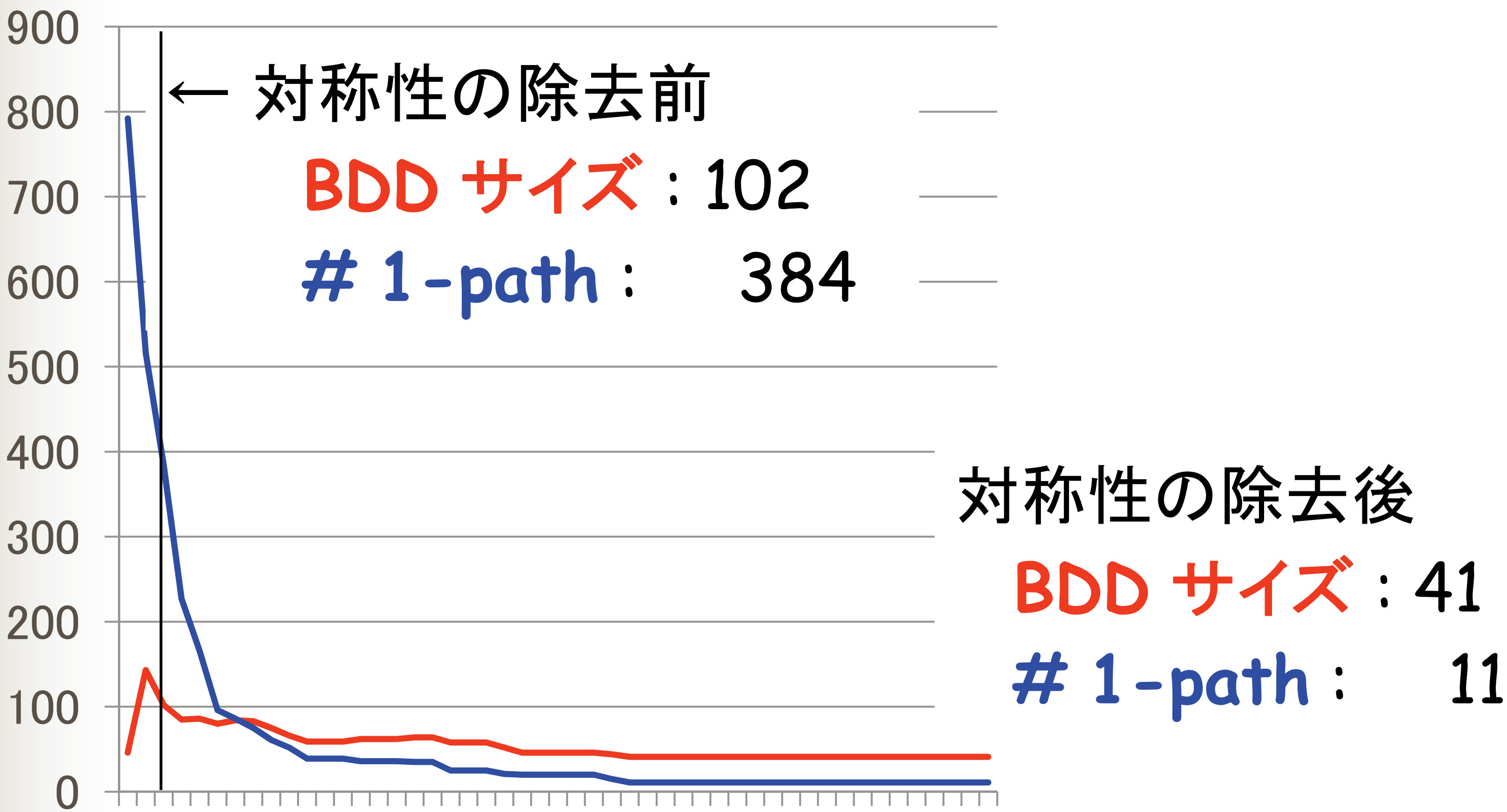
$f'' := \exists y_1 \exists y_2 \cdots f'$ (y の部分を消す)

$f''(101000111011) = 1$

$\vdots$

実験結果

- BDD のサイズ、1-path の個数の推移 (正六面体)

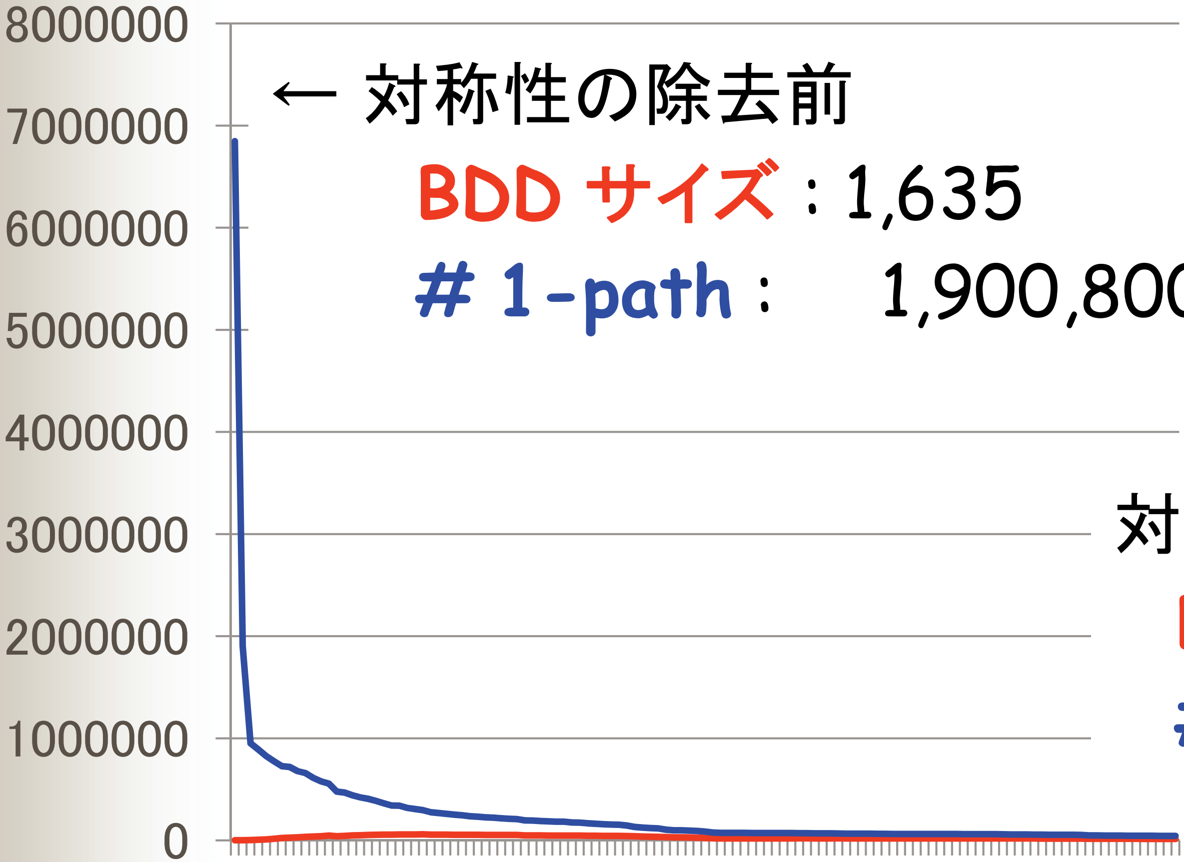


実行時間 0.04 秒

Intel Xeon 3.0 GHz / Debian GNU/Linux

実験結果

- BDD のサイズ、1-path の個数の推移 (正二十面体)



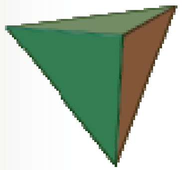
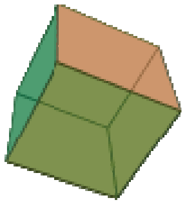
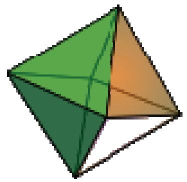
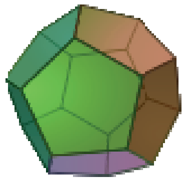
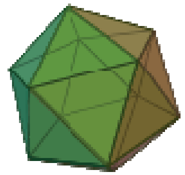
← 対称性の除去前
BDD サイズ : 1,635
1-path :　1,900,800

対称性の除去後
BDD サイズ : 13,723
1-path :　43,380

実行時間 565 秒

Intel Xeon 3.0 GHz / Debian GNU/Linux

実験結果

	正四面体	正六面体	正八面体	正十二面体	正二十面体
					
展開図	2 種類	11 種類	11 種類	43,380 種類	43,380 種類
BDD	8	46	35	9,595	13,737
ZDD	5	33	20	8,875	9,600
ZDD'	5	25	25	6,220	11,576

「切る」/「切らない」 の 1/0 を入れ替えた ZDD のサイズ

まとめ

- 正多面体の展開図を列挙
  - 展開図になるための制約条件を、BDD で表す

    → 1-path 1 つが、展開の仕方 1 つに対応
  - 対称性を排除する
  - 「切る」/「切らない」をもとに、展開図を作成
- 幾何図形の列挙が他にもできる？