



ZDD を用いた Minimal Hitting Set の列挙とその応用

井上 武 (JST ERATO)

D. Knuth が提案した Minimal Hitting Set 列挙アルゴリズムを実装し、計算時間が入力集合族の大きさではなく、それを表す ZDD の大きさに依存することを実験的に確認する。また、アルゴリズムの特性を活かした応用例として、システムの脆弱性発見手法を提案する。

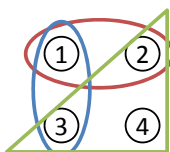
D. Knuth による Minimal Hitting Set 列挙アルゴリズム

Minimal Hitting Set の列挙とは

- 集合族 $f \subseteq 2^E$ に対し、 H が Hitting Set であるとは、任意の $S \in f$ に対して $H \cap S \neq \emptyset$ であること
- f はハイパーグラフとも呼ばれる
- 問題: 極小な (他を含まない) Hitting Set を全て列挙せよ

$$f^\# = \{H \mid S \in f \rightarrow H \cap S \neq \emptyset\}^{\text{minimal}}$$

- 例: $f = \{1,2\}, \{1,3\}, \{2,3,4\}$
- $f^\# = \{1,2\}, \{1,3\}, \{1,4\}, \{2,3\}$



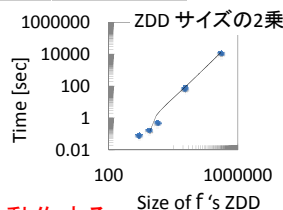
ハイパーグラフ表現

- Hitting Set をひとつ見つけるのは簡単 \rightarrow 全体集合 E
- Minimal Hitting Set も簡単 $\rightarrow E$ から極小になるまで取り除く
- 効率的に列挙するのは難しい
- 既存手法の計算時間は、入出力サイズ $O(|f|)$, $O(|f^\#|)$ に依存 たぶん

実験: 計算時間は $|f|$ ではなく ZDD サイズに依存

- 計算環境: Xeon 3 GHz (1コアのみ利用), CentOS Linux, g++4.4
- 実験データ f : 配電網の制約充足解
- 実験結果

$ E $	$ f $	Size of f 's ZDD	$ f^\# $	Time [sec]
235	4.05e30	834	998	0.09
352	1.11e50	1,759	1,451	0.19
468	1.51e70	3,171	2,164	0.54
235	2.81e24	22,426	162,993	82.36
352	2.39e41	297,684	14,471,969	12,891.21



今回のデータでは

- 計算時間は ZDD サイズに依存
- $|f|$ には非依存 ($|f^\#|$ にも直接は依存しない?)
- $\rightarrow f$ は膨大でも ZDD が小さくなれば高速に動作する

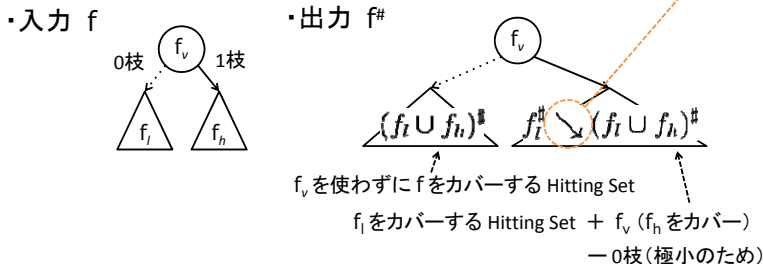
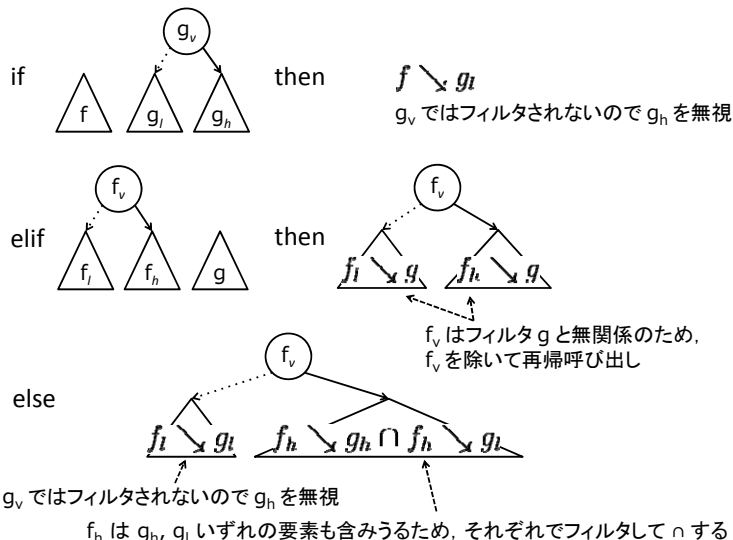
アルゴリズム MINHIT: 再帰処理によって f の ZDD を $f^\#$ の ZDD に変換

D. Knuth, Exercise 237, in TACP, vol.4, fascicle 1, p.145, 2009.

- 入出力は集合族 f と Minimal Hitting Set 集合 $f^\#$ を表す ZDD
- 最上位変数 f_v を除去して得られる下位 ZDD f_i, f_h に対し再帰的に Hitting Set を計算
- 極小な Hitting Set のみを列挙するために 0枝側で計算した Hitting Set を1枝側では除外 (Non-Super Set)
- 高速化のために、計算結果をキャッシュして再利用

- Non-Super Set の列挙 (f を g でフィルタするイメージ)

$$f \searrow g = \{\alpha \in f \mid \beta \in g \Rightarrow \alpha \not\supseteq \beta\}$$



このアルゴリズムの完全性は未証明 (実験的には簡単に確認) である
また、キャッシュを用いるため計算量の解析は難しいと予想している

Minimal Hitting Set 列挙の応用: システムの脆弱性発見

Sorry, unshown on the web