6K-03

ZDDを用いた集合分割の族の表現法

高橋 翔哉 †　　湊 真一 ‡

† 北海道大学 工学部　　‡ 北海道大学 大学院情報科学研究科

1 はじめに

集合分割とは，ある集合を重なりのない部分に分けたもののことを指す．厳密な定義は次節で行う．集合分割の仕方は集合の要素数に対して爆発的に増加することが知られている．

一方で，大規模な集合族を扱う際に有効な Zero-suppressed Binary Decision Diagram (ZDD)[1] というデータ構造が知られている．順列や文字列といった大規模な集合族を ZDD を用いて表現する手法が過去に提案されてきた．本研究では，ZDD を用いて集合分割の族を表現する手法を提案する．提案手法は，集合分割族を表現する ZDD を再帰的に構築し，高速かつコンパクトに任意の集合分割族を表現することを目指す．本稿では，様々な集合分割族に対する提案手法の計算機実験結果を示し，その考察を行う．

2 集合分割

ある単純集合 X の部分集合族 $\{P_1, ..., P_m\}$ が，

- $\emptyset \notin P_i$ $(i = 1, ..., m)$
- $P_1 \cup P_2 \cup ... \cup P_m = X$
- $P_i \cap P_j = \emptyset$ $(i \neq j)$

を満たすとき，$\{P_1, ..., P_m\}$ を X の分割と呼ぶ．以降，このような P_i $(i = 1, ..., m)$ をセルと呼ぶこととする．例えば $X = \{a, b, c\}$ に対する分割は，全部で $\{a, b, c\}, \{a\}\{b, c\}, \{a, c\}\{b\}, \{a, b\}\{c\}, \{a\}\{b\}\{c\}$ の 5 通りとなる．

n 要素の集合に対する集合分割の総数はベル数と呼ばれ，n 番目のベル数 B_n に対し，$B_0 = B_1 = 1$ として，$B_{n+1} = \sum_{k=0}^{n} \binom{n}{k} B_k$ で表される．ベル数の導出式からも分かるように集合分割の仕方は集合の大きさに対して爆発的に増加するため扱いにくく，集合分割族の表現に関する研究はあまりなされていない．

3 提案手法

提案手法では，集合分割族を表現する ZDD を再帰的に構築する．ZDD とは集合族を表現する場合分け二分木を圧縮することで得られる非巡回有向グラフである．ZDD は根節点と呼ばれる親を持たない節点が 1 つ，終端節点と呼ばれる子を持たない節点が 2 つ，それ以外が分岐節点と呼ばれる 2 つの出枝を持つ節点で構成される．分岐節点にはアイテム名がラベル付けされ，1-枝，0-枝と呼ばれる 2 つの出枝でそのアイテムを選択するか否かを表現する．2 つの終端節点にはそれぞれ 1,0 がラベル付けされ，その終端に辿り着く集合が，表現する集合族に含まれるか否かを表す．ZDD は，和集合や共通集合をとる集合演算や特定のアイテムを含む集合族を取り出すといった演算を，圧縮を維持したまま実行できることが知られている．

任意の集合分割はセルの集合で表すことができるので，ZDD の 1 つのアイテムに 1 つのセルを対応付ける．このとき，n 要素の集合分割に対するセルは n 次元のベクトルで表現できることを利用する [2]．あるセルが j 個目の要素を含んでいるか否か (1,0) をベクトルの $(n - j + 1)$ 番目の要素で表し，n 次元のベクトルを 2 進数のバイナリ列とみなした際の 10 進数 k を ZDD のアイテム x_k に対応させる (つまり ZDD では $2^n - 1$ 種類のアイテムを用意することになる)．例えば $X = \{a, b, c\}$ の分割のセル $P_1 = \{a\}, P_2 = \{b, c\}$ は，それぞれ $[0\ 0\ 1], [1\ 1\ 0]$ というベクトルで表現でき，$(001)_2, (110)_2$ とみなせばアイテム x_1, x_6 と対応付けられるので，分割 $\{a\}\{b, c\}$ は x_1x_6 という 2 つのアイテムの組合せで表すことができる．以上により，ZDD で様々なセルの組合せの族すなわち集合分割族を表現することが可能となる．

上記の対応付けを踏まえ，集合分割族を表現する ZDD を再帰的に構築するため，まずは n 要素の集合分割から $n + 1$ 要素の集合分割を再帰的に生成することを考える．このとき，n 要素の分割に対して，

操作 1　$n + 1$ 個目の要素のみを含むセルを加える．
操作 2　ある 1 つのセルに $n + 1$ 個目の要素を加える．

のどちらかを行うことで $n + 1$ 要素の分割が生成される．これらの操作は ZDD における，

操作 1'　n 要素の分割の族を表現する ZDD にアイテム x_{2^n} の節点の 1-枝を直結させる．
操作 2'　n 要素の分割の族を表現する ZDD からアイテム x_i $(1 \leq i \leq 2^n - 1)$ を含む分割をすべて取り出し，そのそれぞれに対して x_i から x_{i+2^n} への変換 (セル x_i に x_{2^n} を追加) を行う．

とそれぞれ対応している．以上から，操作 1' を行い，操作 2' で構築される ZDD と和集合演算を行っていくことで，$n + 1$ 要素の分割を表現する ZDD を再帰的に構築することが可能となる．

A Method for Representing Families of Set Partitions Using ZDDs
Shoya Takahashi†　Shin-ichi Minato‡
†Dept. of Engineering, Hokkaido University
‡Graduate School of Information Science and Technology, Hokkaido University

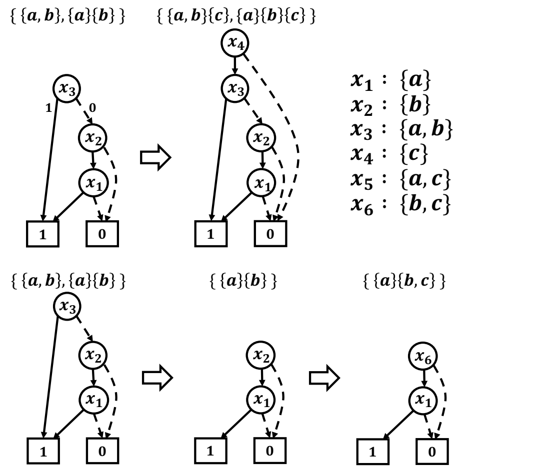

図 1: (上図) $n = 3$ における操作 1'　(下図) $n = 3, i = 2$ における操作 2' を行う際の ZDD 構築の様子

提案手法によって，特定の制約を満たした集合分割族を表現する ZDD を効率的に構築することができる．例として各セルの最大の要素数を k ($k \geq 1$) 以下とした集合分割族を表現する ZDD を構築することを考える．上記のような制約を満たさないセルが生成されるのは，要素数が k であるセルに対して操作 2 を行うことによって新しく要素が加えられた場合のみである．以上から，操作 2' において，x_i と対応付けられたセルの要素数が k 未満，すなわち添字 i を 2 進数表記した際の "1" の数が k 未満であれば操作を行い，そうでなければ操作を行わないようにすることで，制約を満たす集合分割族を表現する ZDD の構築が可能となる．この他にも "特定の要素の組が必ず同じセルに含まれる" などのセルの要素に関する制約を考慮した集合分割族を表現する ZDD は提案手法によって効率的に構築することができる．

4　計算機実験

n 要素の集合分割すべてを表現する ZDD および各セルの最大要素数を 3 以下とした n 要素の集合分割を表現する ZDD の実験結果を表 1，表 2 に示す．表の "ZDD size" は ZDD の節点数，"time" は ZDD の構築時間，"of part" は ZDD に含まれる分割の数，"comp" は圧縮率 (ZDD に含まれる分割の数/節点数) を表す．実験環境には，64-bit Ubuntu 16.04 LTS，Intel Core i7-3930K 3.2GHz CPU，64GB RAM を用いた．

どちらの場合についても ZDD の節点数および構築時間は n の大きさに対して指数的に増加しているが，n 要素の分割すべての ZDD よりも上記のような制約を考慮した集合分割族に対する ZDD の方が節点数および構築時間の増加が比較的緩やかであることが確認できる．これは，制約によって ZDD で用いられるアイテム数が減ったことに加え，制約を満たしたセルを用いる分割の数が多いために ZDD の節点共有も多く行われるためと考えられる．また，どちらの場合についても n の大きさに従って高い圧縮率が得られているのが確認できる．n 要素の集合分割族の総数は，ベル数 B_n とすれば 2^{B_n} 通り存在し，その表現のために B_n ビットの記憶量がどのようにしても必要となることを考えると，提案手法による集合分割族の表現は有効であるといえる．

表 1: n 要素の集合分割すべてを表現する ZDD

n	ZDD size	time(sec)	of part	comp
1	1	0.0000	1	1.0000
2	3	0.0000	2	0.6667
3	8	0.0000	5	0.6250
4	21	0.0000	15	0.7143
5	56	0.0000	52	0.9286
6	153	0.0000	203	1.3268
7	428	0.0000	877	2.0491
8	1,221	0.0020	4,140	3.3907
9	3,536	0.0100	21,147	5.9805
10	10,353	0.0420	115,975	11.2021
11	30,548	0.1490	678,570	22.2132
12	90,621	0.5410	4,213,597	46.4969
13	269,816	3.1440	27,644,437	102.4570

表 2: 各セルの最大要素数を 3 以下とした n 要素の集合分割を表現する ZDD

n	ZDD size	time(sec)	of part	comp
1	1	0.0000	1	1.0000
2	3	0.0000	2	0.6667
3	8	0.0000	5	0.6250
4	20	0.0000	14	0.7000
5	49	0.0000	46	0.9388
6	118	0.0000	166	1.4068
7	274	0.0000	652	2.3796
8	614	0.0000	2,780	4.5277
9	1,342	0.0010	12,644	9.4218
10	2,879	0.0040	61,136	21.2352
11	6,076	0.0110	312,676	51.4608
12	12,637	0.0250	1,680,592	132.9900
13	25,953	0.0490	9,467,680	364.8010

5　まとめ

本稿では，ZDD を用いて集合分割の族を表現する手法を提案した．提案手法によって，いくつかの制約を考慮した集合分割族を表現する ZDD を効率的に構築可能であることを確認した．

今後は，より適当な集合分割族の表現法や制約を付けた集合分割族を模索するとともに，実応用先の検討および実験を行いたい．

謝辞

本研究の一部は科研費基盤 (S)15H05711 の助成による．

参考文献

[1] Shin-ichi Minato, "Zero-Suppressed BDDs for Set Manipulation in Combinatorial Problems", Proc. of 30th ACM/IEEE Design Automation Conf.(DAC 1993), pp.272-277, 1993.

[2] Donald E. Knuth, "The Art of Computer Programming, Volume 4A : Combinatorial Algorithms, Part 1", Addison-Wesley, 2014.