CA-006

SAT ソルバを用いた C1P 分割問題の解法
An algorithm for C1P Partition Problem using SAT Solver

原田崇司 † **竹内聖悟** †
Takashi Harada Shogo Takeuchi

1. はじめに

0-1 行列の列を並び替えることにより各行の 1 を連続させられるとき，その 0-1 行列は Consecutive Ones Property (C1P) を満たすという．0-1 行列が C1P を満たすかは多項式時間で判定可能である [1,2]．けれども，C1P を満たさない $r \times c$ の 0-1 行列 M を C1P を満たす $r_1 \times c$ の部分行列 M_1 と $r_2 \times c$ の部分行列 M_2 へと分割できるかを判定する問題は **NP** 完全である [3]．ただし，$r_1 + r_2 = r$ である．

本稿では，次の最適化問題について考える．
入力: 0-1 行列 M
出力: M の分割サイズ最小の C1P 分割 $\mathcal{M}$
分割 $\mathcal{M} = [R_1, R_2, \ldots, R_k]$ が M の C1P 分割であるとは，部分行列 R_1, R_2, $\ldots$, R_k が C1P を満たすことをいう．ただし，R_i は行列 M の行集合の部分集合である．

0-1 行列 M に対して十分小さな分割サイズの C1P 分割が存在すると，Harada らが提案するデータ構造を用いて高速なパケット分類を実現できる [4]．けれども，0-1 行列を C1P を満たす部分行列へと分割する問題は研究されていない．

本稿では，0-1 行列に対する分割サイズ最小の C1P 分割を求めるアルゴリズムを提案する．提案手法は初めに，Top-Down ZDD [5] を用いて C1P を満たさない部分行列を列挙する．次にこれらの部分行列からグラフ彩色問題と同じように C1P 分割問題を SAT に符号化して [7]，このインスタンスを minisat を用いて解く [8]．更に，提案手法の有効性をランダムに生成した 0-1 行列を使用した計算機実験によって確かめる．

2.C1P 分割問題

本章では，C1P 分割問題を定義する．そのために，初めに C1P について説明する．

定義 2.1. *0-1* 行列 M の各行の *1* を連続させるように列を並び替えられるとき，M は *C1P* を満たすという．

例えば，表 1 の 0-1 行列は，c_1, c_3, c_2, c_4 と列を並べると表 2 のように各行の 1 が連続するので C1P を満たす．これに対して，表 3 の 0-1 行列は，列 c_1, c_2, c_3, c_4 をどのように並べても各行の 1 が連続しないので C1P を満たさない．表 3 の行列のように C1P を満たさない行列を Non-C1P 行列という．

$r \times c$ の 0-1 行列 M が C1P を満たすか否かを判定する問題は，partition refinement という技法を用いることにより $O(r+c+m)$ で計算できる [2]．ただし，m は行列中の 1 成分の数である．

0-1 行列の行による分割を定義する．行列 M の行を r で列を c で表す．行列 M の行全体の集合を R，列全体の集合を C でそれぞれ表す．$r \times c$ の行列 M の分割 $\mathcal{M}$ を $\mathcal{M} = \{R_i | i \in I\}$ と表す．ただし，$R_i \subseteq R$ で $I = \{1, 2, \ldots, k(\leq r)\}$ は R の部分集合を識別するための添字の集合である．以上の定義を踏まえ以下に C1P 分割問題を定義する．

定義 2.2. *C1P* 分割問題
入力: *0-1* 行列 M
出力: $|I|$ が最小となる M の分割 $\mathcal{M}$
ただし，$\mathcal{M}$ のすべての要素 R_i は *C1P* を満たす．

例えば，表 3 の 0-1 行列に対する，表 4, 5 に示す分割 $\{R_1 = \{r_1, r_2, r_4, r_5\}, R_2 = \{r_3, r_6\}\}$ は R_1, R_2 が C1P を満たすので，表 3 の 0-1 行列の C1P 分割である．

3.Minimal Conflicting Set の列挙

0-1 行列 M の行集合 R の部分集合 R_i が C1P を満たさないとき，R_i のことを Conflicting Set とよぶ．更に，Conflicting Set R_i の任意の真部分集合 $R_i' \subsetneq R_i$ が C1P を満たすとき R_i を Minimal Conflicting Set とよぶ [9]．提案する手法は，C1P 分割問題を SAT ソルバを用いて解くために，初めに R の Minimal Conflicting Set を列挙する．そして，Minimal Conflicting Sets から SAT へと帰着して，SAT ソルバで解く．

表 6 の 0-1 行列に対する Conflicting Sets の ZDD を TdZdd を用いて素朴に構築すると図 1 の ZDD を得る．更に，この ZDD を探索することにより下記に示す Conflicting Sets が得られる．

$$
\begin{gathered}
\{r_3,r_4,r_5\},\{r_1,r_4,r_6\},\{r_1,r_2,r_4,r_6\},\{r_3,r_4,r_6\},\\
\{r_3,r_5,r_6\},\{r_4,r_5,r_6\},\{r_1,r_4,r_7\},\{r_1,r_2,r_4,r_7\},\\
\{r_3,r_4,r_7\},\{r_3,r_5,r_7\},\{r_4,r_5,r_7\},\{r_1,r_6,r_7\},\\
\{r_1,r_2,r_6,r_7\},\{r_3,r_6,r_7\},\{r_4,r_6,r_7\},\{r_5,r_6,r_7\},\\
\{r_1,r_4,r_8\},\{r_1,r_2,r_4,r_8\},\{r_3,r_4,r_8\},\{r_3,r_5,r_8\},\\
\{r_4,r_5,r_8\},\{r_1,r_4,r_6,r_8\},\{r_1,r_2,r_4,r_6,r_8\},\\
\{r_3,r_4,r_6,r_8\},\{r_3,r_5,r_6,r_8\},\{r_4,r_5,r_6,r_8\},\\
\{r_1,r_4,r_7,r_8\},\{r_1,r_2,r_4,r_7,r_8\},\{r_3,r_4,r_7,r_8\},\\
\{r_3,r_5,r_7,r_8\},\{r_4,r_5,r_7,r_8\},\{r_6,r_7,r_8\}
\end{gathered}
$$

しかし，SAT 符号化に必要となるのは Conflicting Sets ではなく *Minimal* Conflicting Sets なので，上記の Conflicting Sets から極小ではない部分集合を取り除

† 高知工科大学 情報学群
School of Information, Kochi University of Technology

Copyright © 2019 by Information Processing Society of Japan and
The Institute of Electronics, Information and Communication Engineers
All rights reserved.

表 1　C1P 行列の例

	c_1	c_2	c_3	c_4
r_1	1	0	1	0
r_2	1	1	1	1
r_3	0	1	1	0
r_4	0	1	0	1
r_5	1	0	1	0
r_6	0	0	1	0

表 2　表 1 を替び替え

	c_1	c_3	c_2	c_4
r_1	1	1	0	0
r_2	1	1	1	1
r_3	0	1	1	0
r_4	0	0	1	1
r_5	1	1	0	0
r_6	0	1	0	0

表 3　Non-C1P 行列の例

	c_1	c_2	c_3	c_4
r_1	1	0	1	0
r_2	0	0	0	1
r_3	0	1	1	1
r_4	1	1	1	0
r_5	0	1	1	0
r_6	1	1	0	0

表 4　表 3 の分割 M_1

	c_1	c_2	c_3	c_4
r_1	1	0	1	0
r_2	0	0	0	1
r_4	1	1	1	0
r_5	0	1	1	0

表 5　表 3 の分割 M_2

	c_1	c_2	c_3	c_4
r_3	0	1	1	1
r_6	1	1	0	0

表 6　C1P 分割問題のインスタンスの例

	c_1	c_2	c_3	c_4	c_4	c_4	c_4	c_8
r_1	1	1	0	1	1	0	1	1
r_2	0	0	0	0	0	0	0	1
r_3	1	1	0	1	0	0	0	1
r_4	1	1	0	0	0	1	1	1
r_5	1	1	0	0	1	0	1	1
r_6	1	1	1	1	0	0	1	0
r_7	0	0	1	0	1	1	0	1
r_8	1	0	0	0	1	1	0	1

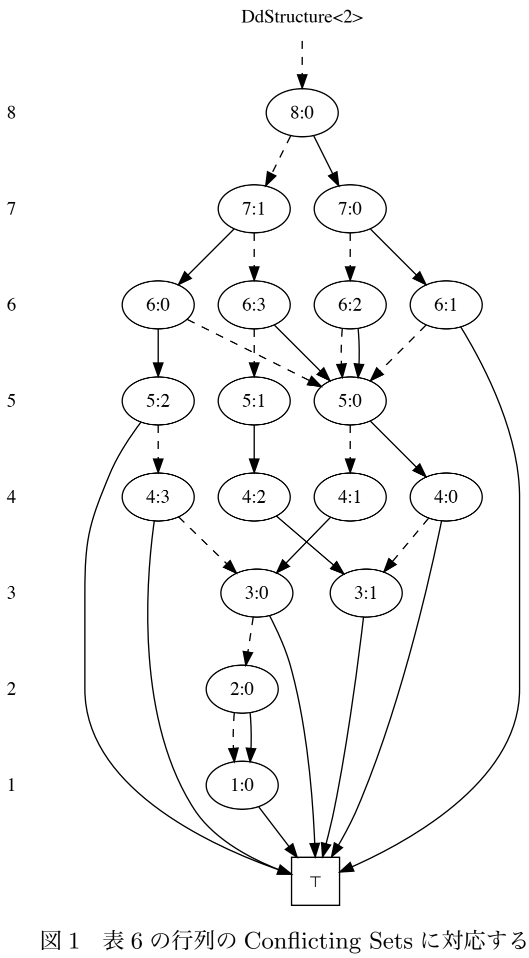

図 1　表 6 の行列の Conflicting Sets に対応する ZDD

いて式 (1) の Minimal Conflicting Sets を求める必要がある．提案手法では，Conflicting Set をグラフへと変換し，グラフを効率よく扱うためのライブラリである Graphillion [10] を用いて Minimal Conflicting Sets を計算している．

$$
\begin{aligned}
&\{r_3,r_4,r_5\},\{r_1,r_4,r_6\},\{r_3,r_4,r_6\},\{r_3,r_5,r_6\},\\
&\{r_4,r_5,r_6\},\{r_1,r_4,r_7\},\{r_3,r_4,r_7\},\{r_3,r_5,r_7\},\\
&\{r_4,r_5,r_7\},\{r_1,r_6,r_7\},\{r_3,r_6,r_7\},\{r_4,r_6,r_7\},\\
&\{r_5,r_6,r_7\},\{r_1,r_4,r_8\},\{r_3,r_4,r_8\},\{r_3,r_5,r_8\},\\
&\qquad\qquad\{r_4,r_5,r_8\},\{r_6,r_7,r_8\}
\end{aligned}
\tag{1}
$$

4.Non-C1P 部分行列から SAT への符号化

本章では，極小 Non-C1P 部分行列の集合から C1P 問題の SAT 符号化について説明する．

本章では，行列 M を K 個の部分行列 M_1, M_2, ..., M_K へと分割するものとする．つまり，0-1 行列 M に対して分割サイズ K の C1P 分割があるかを判定する問題，K-C1P 分割問題について考える．r_i を 0-1 行列 M の i 行目とするとき，変数 r_{ij} を

$$
r_{ij} = \begin{cases} 1, & r_i \text{は部分行列 } M_j \text{に含まれる} \\ 0, & r_i \text{は部分行列 } M_j \text{に含まれない} \end{cases} \tag{2}
$$

と定義する．K-C1P 分割問題の変数の数は rK となる．

K-C1P 分割問題の制約条件は

1. 任意の列 r_i は少なくとも一つの部分行列に属す
2. 任意の列 r_i は高々一つの部分行列に属す
3. $\{r_i,\ r_j,\ \ldots,\ r_k\}$ が Minimal Conflicting Set のとき，r_i, r_j, r_k の少なくとも一つが異なる部分行列

Copyright © 2019 by Information Processing Society of Japan and
The Institute of Electronics, Information and Communication Engineers
All rights reserved.

表 7　表 3 の部分行列 M_1

	c_1	c_2	c_3	c_4	c_4	c_4	c_4	c_8
r_2	0	0	0	0	0	0	0	1
r_4	1	1	0	0	0	1	1	1
r_6	1	1	1	1	0	0	1	0

表 8　表 3 の部分行列 M_2

	c_1	c_2	c_3	c_4	c_4	c_4	c_4	c_8
r_3	1	1	0	1	0	0	0	1
r_7	0	0	1	0	1	1	0	1
r_8	1	0	0	0	1	1	0	1

表 9　表 3 の部分行列 M_3

	c_1	c_2	c_3	c_4	c_4	c_4	c_4	c_8
r_1	1	1	0	1	1	0	1	1
r_5	1	1	0	0	1	0	1	1

に属す

となり，それぞれを節で表現すると以下となる．

1. $(r_{i1} \vee r_{i2} \vee \ldots, \vee r_{iK})$
2. $(\overline{r_{i1}} \vee \overline{r_{i2}})$, $(\overline{r_{i1}} \vee \overline{r_{i3}})$, $\ldots$, $(\overline{r_{i1}} \vee \overline{r_{iK}})$, $(\overline{r_{i2}} \vee \overline{r_{i3}})$, $(\overline{r_{i2}} \vee \overline{r_{i4}})$, $\ldots$, $(\overline{r_{i2}} \vee \overline{r_{iK}})$, $\ldots$, $(\overline{r_{i(K-1)}} \vee \overline{r_{iK}})$
3. $(\overline{r_{i1}} \vee \overline{r_{j1}} \vee \overline{r_{k1}})$, $(\overline{r_{i2}} \vee \overline{r_{j2}} \vee \overline{r_{k2}})$, $\ldots$, $(\overline{r_{iK}} \vee \overline{r_{jK}} \vee \overline{r_{kK}})$

5. 計算機実験

提案手法の有効性を確かめるために C++ 言語を用いて計算機実験を行った．実験に用いた PC は，主要メモリ 4GB，CPU が Intel(R) Core(TM) i3-8100 CPU@3.60GHz で OS は Arch Linux 5.0.4 である．提案手法を実装するにあたり，Conflicting Sets を列挙するために TdZdd [5, 6] を，Conflicting Sets から Minimal Conflicting Sets を取り出すのに Graphillion [10] を，形式化した SAT を解くために SAT ソルバ minisat [8] を使用した．計算機実験のために，行数 15∼30，列数 15∼30 の 15×15 の組合せの 0-1 行列をランダムに生成した．これらの 0-1 行列に対して，分割サイズ最小となる C1P 分割とその計算時間とを計測した．C1P 分割のサイズ，節の数を表 10,11 に示す．表 10,11 の 1 行目，1 列目はそれぞれ，0-1 行列の列と行の数を表す．また，表中の † は 10 分以内に SAT ソルバが停止しなかったことを意味する．

提案手法は，符号化した SAT を SAT ソルバで解くところで時間が一番かかった．また，Conflicting Sets を列挙することは 30 × 30 の行列に対しても 1 秒未満で終わるが，Conflicting Sets を Minimal Conflicting Sets への変換には数秒要した．

表 10 に示すように，行数の増加よりも列数の増加が問題を難しくしている．

列の数が小さいときには，C1P サイズが最悪の分割サイズ $|r|/2$ に比べて最大で 42% ほど小さくなった．ただし，$|r|$ は行列の行の数を表す．この結果は，文献 [4] のアルゴリズムを含む様々なアルゴリズムの道具として C1P 分割が使える可能性を示す．

6. まとめ

本稿では，0-1 行列に関する性質 C1P に基づく最適化問題を定義し，この問題に対する TdZDD と SAT ソルバを用いた解法を提案した．提案手法はランダムに生成された 0-1 行列に対して，30×29 の行列に対して自明でない解を得ることができた．

今後の課題は主に三つある．提案手法は 0-1 行列の Minimal Conflicting Sets を得るために，TdZdd を用いて一旦 Conflicting Sets を列挙した後に，そこから Minimal Conflicting Sets を求めている．これは冗長であるので，0-1 行列に対する Minimal Conflicting Sets に対応する ZDD を直接構築する手法を提案することが今後の課題の一つである，

提案手法は Minimal Conflicting Sets から素朴に SAT への符号化を行っているが，SAT への符号化の良し悪しが SAT ソルバで問題を現実的な時間で解けるか否かに大きく関わる [11]．よって，二つ目の課題は，C1P 分割問題を SAT ソルバで解きやすい形へと符号化する手法を提案することである．

最後に，小さな分割サイズの C1P 分割が存在するための 0-1 行列の性質を研究することが今後の重要な課題である．

参考文献

[1] K.S. Booth, and G.S. Lueker, “Testing for the consecutive ones property, interval graphs, and graph planarity using pq-tree algorithms,” Journal of Computer and System Sciences, vol.13, no.3, pp.335 - 379, 1976.

[2] M. Habib, R. McConnell, C. Paul, and L. Viennot, “Lex-bfs and partition refinement, with applications to transitive orientation, interval graph recognition and consecutive ones testing,” Theoretical Computer Science, vol.234, no.1, pp.59 - 84, 2000.

[3] M.R. Garey, and D.S. Johnson, Computers and Intractability: A Guide to the Theory of NP-Completeness, W. H. Freeman & Co., New York, NY, USA, 1979.

[4] T. Harada, Y. Ishikawa, K. Tanaka, and K. Mikawa, “A packet classification method via cascaded circular-run-based trie,” IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences, Sep. 2019 (in press).

Copyright © 2019 by Information Processing Society of Japan and
The Institute of Electronics, Information and Communication Engineers
All rights reserved.

表 10　C1P 分割のサイズ

	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
15	5	6	6	7	7	6	†	7	7	†	†	7	7	†	7	†
16	6	6	7	7	7	7	†	7	†	†	†	†	†	†	†	†
17	6	6	7	7	8	8	8	8	8	8	8	8	8	†	†	†
18	6	7	8	7	7	8	8	8	8	†	†	†	8	†	†	†
19	7	7	7	7	7	7	8	8	8	9	9	9	8	9	9	†
20	7	7	8	8	7	8	9	9	†	9	†	†	†	†	†	†
21	7	7	7	8	9	8	9	9	†	9	9	10	10	†	10	10
22	7	7	8	8	9	9	8	9	10	10	10	10	10	†	10	†
23	7	8	8	8	9	10	9	10	11	10	10	11	11	10	11	11
24	8	8	8	10	9	10	9	10	11	10	11	11	11	11	11	11
25	8	9	8	10	10	11	10	11	12	11	11	12	11	†	12	12
26	8	9	9	10	10	10	10	12	11	12	11	12	†	†	12	12
27	8	9	8	10	9	10	11	12	12	11	13	12	13	†	13	13
28	8	9	9	11	11	10	11	12	12	13	13	13	†	†	13	†
29	8	10	10	10	10	12	12	12	11	13	14	13	14	13	14	14
30	10	10	11	11	10	12	11	12	12	13	13	13	14	14	14	†

表 11　節の数（$K = 2$ のとき）

	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
15	830	862	798	906	934	908	940	910	934	940	940	934	938	940	938	940
16	1094	1022	1112	1122	1120	1110	1146	1122	1150	1146	1152	1152	1124	1152	1152	1152
17	1278	1258	1342	1340	1328	1382	1348	1384	1386	1364	1390	1390	1392	1394	1394	1394
18	1438	1642	1640	1618	1596	1624	1604	1658	1662	1666	1668	1666	1662	1668	1668	1636
19	1850	1764	1820	1928	1824	1954	1956	1870	1904	1932	1936	1970	1906	1970	1972	1976
20	2094	2042	2216	2280	2252	2254	2278	2276	2320	2278	2316	2318	2248	2316	2320	2320
21	2332	2504	2442	2614	2570	2588	2676	2648	2694	2654	2614	2700	2698	2702	2700	2698
22	2826	2858	2828	2940	3028	3046	3048	3026	3114	3038	3040	3074	3114	3082	3118	3122
23	2972	3254	3334	3152	3142	3526	3478	3556	3540	3528	3576	3584	3586	3572	3586	3540
24	3692	3802	3856	4010	3952	4062	4010	3988	4048	4040	4088	4088	4090	3966	4096	4096
25	4266	4192	4368	4372	4564	4384	4528	4410	4638	4552	4540	4642	4634	4650	4648	4602
26	4708	4762	5002	5106	5114	5160	5210	5236	5240	5196	5234	5200	5250	5204	5246	5202
27	5126	5592	5134	5682	5682	5772	5670	5796	5790	5890	5894	5894	5900	5904	5900	5854
28	5830	5816	5758	6378	6412	6260	6348	6390	6576	6600	6598	6598	6602	6604	6600	6608
29	6136	6836	6676	6848	6974	7306	7268	7224	7132	7340	7364	7352	7356	7306	7364	7360
30	7544	7574	7938	8000	7780	8030	7920	8066	7986	8098	8094	7892	8170	8120	8116	8180

[5] H. Iwashita, and S. Minato, “Efficient top-down zdd construction techniques using recursive specifications,” , 2013.

[6] 戸田貴久，斎藤寿樹，岩下洋哲，川原純，湊真一，“Zdd と列挙問題 - 最新の技法とプログラミングツール，” コンピュータソフトウェア，vol.34, no.3, pp.97–120，aug 2017.

[7] A.V. Gelder, “Another look at graph coloring via propositional satisfiability,” Discrete Applied Mathematics, vol.156, no.2, pp.230 - 243, 2008, Computational Methods for Graph Coloring and it’s Generalizations.

[8] N. Eén, and N. Sörensson, “An extensible sat-solver,” Theory and Applications of Satisfiability Testing, eds. E. Giunchiglia, and A. Tacchella, pp.502–518, Berlin, Heidelberg, 2004, Springer Berlin Heidelberg.

[9] C. Chauve, U.U. Haus, T. Stephen, and V.P. You, “Minimal conflicting sets for the consecutive ones property in ancestral genome reconstruction,” Comparative Genomics, eds. F.D. Ciccarelli, and I. Miklós, pp.48–58, Berlin, Heidelberg, 2009, Springer Berlin Heidelberg.

[10] T. Inoue, H. Iwashita, J. Kawahara, and S.i. Minato, “Graphillion: software library for very large sets of labeled graphs,” International Journal on Software Tools for Technology Transfer, vol.18, no.1, pp.57–66, Feb 2016.

[11] D.E. Knuth, The Art of Computer Programming, Volume 4, Fascicle 6: Satisfiability, Addison-Wesley Professional1st edition, 2015.

Copyright © 2019 by Information Processing Society of Japan and The Institute of Electronics, Information and Communication Engineers All rights reserved.