A Matrix Representation for Resolution in Propositional Logic

命題論理における導出のための行列表現

川 口 雄 一
Yuuichi KAWAGUCHI

'Resolution' is an algorithm for logical formulae. In this paper, logical formulae are denoted by matrices. Resolution steps are processed by elementary transformations of matrices. By this representation, the complexity of resolution is estimated to be finite. Improvement of the algorithm is a subject of future work.

「レゾリューション」（導出）とは論理式に対する計算方法である。本稿では、論理式を行列により表現する。基本行列変形によりレゾリューションを進める。この表現と計算の方法によるレゾリューションの複雑さ（計算量）は有限である。計算の効率を改善することはこれからの課題である。

Key words: resolution （レゾリューション）
propositional logic （命題論理）
matrix representation （行列表現）

Liberal Arts and Sciences, School of Nursing and Nutrition, Tenshi College
(Received 21 January 2008／Accepted 2 May 2008)

I. Introduction

'Resolution' [Gal86, Hed04] is an algorithm for logical formulae. By resolution, a given logical formula is proven unsatisfiable or not.

In this paper, a new representation for logical formulae is proposed. Logical formulae are denoted by matrices. Resolution steps are processed by elementary transformatuons of matrices. Though the word 'matrix' is also used in papers [BE97, Bib81, Bib79], it is used for representing a different idea from that in this paper.

A paper [Kaw02] shows that there are a common structure between logical problems such as SAT, and algebraic problems such as simultaneous equations. Simultaneous equations are solvable by an efficient method, however, SAT is not yet. A problem of simultaneous equations is expressed by a matrix and is solved by transforming the matrix expression. In this paper, matrices are used for a representation for logical formulae (CNF). The goal of this study is to find some elemetary transformations for those matrices.

At the end of this paper, the complexity of resolution by the matrix representation is discussed. The SAT problem is proven to be NP-complete and any method of which complexity is in the class '*P*' has not been found yet. The method proposed in this paper also not in the class P.

II. Resolution

Definition 1 (CNF) A *literal* is a variable or a negation ($\neg$) of a variable. A *clause* is a disjunction of literals. A conjunctive normal form (*CNF*) is a conjunction of clauses. (end)

Resolution is a way for showing whether or not a given CNF is *unsatisfiable*.

Definition 2 (Resolution Step) It holds that $C_1 \wedge \cdots \wedge C_n \wedge (P \vee x) \wedge (Q \vee \neg x)$ is unsatisfiable $\Longleftrightarrow C_1 \wedge \cdots \wedge C_n \wedge (P \vee x) \wedge (Q \vee \neg x) \wedge (P \vee Q)$ is unsatisfiable, where P, Q, $C_1, \ldots, C_n$ are clauses and x is a variable.

The clause $(P \vee Q)$ is called a *resolvent* of the clauses $(P \vee x)$ and $(Q \vee \neg x)$ is often called a resolvent for the variable 'x'

The process of adding a resolvent to the original CNF is called a *resolution step.* (end)

If an empty clause is obtained after some resolution steps for a given CNF, then the CNF is unsatisfiable.

III. Matrix Representation

By an example, a matrix representation of CNF and resolution steps are shown.

A CNF '$(x \vee y) \wedge (\neg x \vee y) \wedge (\neg y \vee z) \wedge (\neg y \vee \neg z)$' is given. By using a matrix, the CNF is denoted as

$$\left(\begin{array}{c|rrr} & x & y & z \\ \hline \text{a.} & 1 & 1 & 0 \\ \text{b.} & -1 & 1 & 0 \\ \text{c.} & 0 & -1 & 1 \\ \text{d.} & 0 & -1 & -1 \end{array}\right).$$

The names 'a.,' 'b.,' 'c.,' 'd.' for rows and names 'x,' 'y,' 'z' for variables (or for columns) are placed for convenience and may be omitted.

The symbol '1' stands for the presence of the variable. The symbol '-1' stands for the presence of the negated ($\neg$) variable. The symbol '0' stands for the absence of the variable. For example, the row 'a.' corresponds to the clause '$x \vee y$.'

For the variable 'x,' there can be one resolvent:

- from rows 'a.' and 'b.,' the resolvent 'y' is obtained.

There is no other resolvent for the variable 'x.'

The result of resolution steps for the variable 'x' is shown as

$$\left(\begin{array}{rr|rrr} & & x & y & z \\ \hline & \text{a.} & 1 & 1 & 0 \\ & \text{b.} & -1 & 1 & 0 \\ & \text{c.} & 0 & -1 & 1 \\ & \text{d.} & 0 & -1 & -1 \\ \text{a, b} \rightarrow & \text{e.} & 0 & 1 & 0 \end{array}\right).$$

For the variable 'y,' from rows where the symbol '0' is placed in the 'x' column, there can be two resolvents:

- from rows 'c.' and 'e.,' the resolvent 'z' is obtained.
- from rows 'd.' and 'e.,' the resolvent '$\neg z$' is obtained.

The result of the resolution steps for the variable 'y' is shown as

$$\left(\begin{array}{rr|rrr} & & x & y & z \\ \hline & \text{a.} & 1 & 1 & 0 \\ & \text{b.} & -1 & 1 & 0 \\ & \text{c.} & 0 & -1 & 1 \\ & \text{d.} & 0 & -1 & -1 \\ & \text{e.} & 0 & 1 & 0 \\ \text{c, e} \rightarrow & \text{f.} & 0 & 0 & 1 \\ \text{d, e} \rightarrow & \text{g.} & 0 & 0 & -1 \end{array}\right).$$

For the variable 'z,' from rows where the symbol '0' is placed in the 'x' and 'y' columns, there can be one resolvent:

- from rows 'f.' and 'g.,' the resolvent '$\square$,' the empty clause, is obtained.

The result of resolution steps for the variable 'z' is shown as

$$\left(\begin{array}{rr|rrr} & & x & y & z \\ \hline & \text{a.} & 1 & 1 & 0 \\ & \text{b.} & -1 & 1 & 0 \\ & \text{c.} & 0 & -1 & 1 \\ & \text{d.} & 0 & -1 & -1 \\ & \text{e.} & 0 & 1 & 0 \\ & \text{f.} & 0 & 0 & 1 \\ & \text{g.} & 0 & 0 & -1 \\ \text{f, g} \rightarrow & \text{h.} & 0 & 0 & 0 \end{array}\right).$$

The resolution steps halt. Consequently we obtained the empty clause. The given CNF is proven to be unsatisfiable.

IV. Discussion

1. Not unsatisfiable

For example, the resolvent of the two clauses '$x \vee y \vee z$' and '$\neg x \vee \neg y \vee z$' for the variable '$x$' is the clause '$y \vee \neg y \vee z$.' The clause includes a disjunction of 'y' and '$\neg y$,' and it is therefore not unsatisfiable. To show this fact, the symbol " is used as follows:

$$\left(\begin{array}{rr|rrr} & & x & y & z \\ \hline & \text{a.} & 1 & 1 & 1 \\ & \text{b.} & -1 & -1 & 1 \\ \text{a, b} \rightarrow & \text{c.} & 0 & * & 1 \end{array}\right).$$

In the obtained clause at the row 'c.,' the symbol '*' indicates the disjunction '$y \vee \neg y$.' The clause is not unsatisfiable (*i.e.*, always satisfiable). Such clauses are not used in the following resolution steps.

2. Complexity

Supposing that there are m variables ($x_1, \ldots, x_m$) and l clauses $C_1, \ldots, C_l$, then initially there are $m \cdot l$ elements (*i.e.*, the size of the matrix) in the matrix representation, where each element is one of 1,–1,0 and each clause is a disjunction of some x_i's and some $\neg x_j$'s ($1 \leq i,\ j \leq m$).

Considering that we deal with only k–SAT problems ($k \leqq m$), *i.e.*, all the numbers of literals in clauses are equal to k, then the number of clauses is less than or equal to 2^k. It holds that the size of the initial matrix $m \cdot l = m \cdot 2^k \leqq m \cdot 2^m$

For the variable 'x_1,' there can be $(l/2)^2 = l^2/4$ resolvents in the worst case. In those resolvents all x_1's have disappeared, *i.e.*, represented by '0.' In those new resolvents, for the variable 'x_2,' there can be $(l/2 \cdot l^2/4)^2 = l^4/4^3$ resolvents. For the variable 'x_3,' there can be $(l/2 \cdot l^4/4^3)^2 = l^8/4^7$ resolvents, and so on.

The sum of numbers of resolvents is

$l^2/4+l^4/4^3+\cdots+l^{(m-1)^2}/4^{(m-1)^2-1}$. This is estimated to be $\mathcal{O}(l^{(m-1)^2}/4^{(m-1)^2-1})$, and

$$\begin{aligned} l^{(m-1)^2}/4^{(m-1)^2-1} &\leqq 2^{m(m-1)^2}/4^{(m-1)^2-1} \\ &=4\cdot 2^{m(m-1)^2}/4^{(m-1)^2} \\ &=4\cdot(2^m/4)^{(m-1)^2} \\ &=4\cdot(2^{(m-2)})^{(m-1)^2} \\ &=4\cdot 2^{(m-2)(m-1)^2} \end{aligned}$$

Consequently it is estimated to be $\mathcal{O}(2^{m^3})$. This is the number of clauses at the end of the calculation by matrix representation, and shows the complexity of the resolution steps.

The size of the truth table for a k-CNF $C_1\wedge\cdots\wedge C_l$ is $2^m\cdot k\cdot l$. This is estimated to be $\mathcal{O}(2^m\cdot m\cdot 2^m)=\mathcal{O}(m\cdot 2^{2m})$. The method shown in this paper is more inefficient than the method by the truth table.

V. Conclusion

This paper shows a matrix representation of CNF. Its resolution steps are processed by elementary transformations of the matrix. The complexity of the steps is shown.

In a paper [Fuj07], the number of variables m is about one million. The efficiency of the method shown in this papaer is low, and it is more inefficient than the truth table. Improvement of the method is a subject of future work.

Acknowledgements

The author thanks friends for their suggestions, and thanks anonymous reviewer for constructive suggestions, which helped to mprove the clarity of this paper.

References

[BE97] Wolfgang Bibel and Elmar Eder. Decomposition of tautologies into regular formulas and strong completeness of connection-graph resolution. *Journal of the ACM*, Vol. 44, No. 2, pp. 320。ゥ344, March 1997.

[Bib79] W. Bibel. Tautology testing with a generalized matrix reduction method. *Theoretical Computer Science*, Vol. 8, pp. 31–44, 1979.

[Bib81] Wolfgang Bibel. On matrices with connections. *Journal of the ACM*, Vol. 28, No. 4, pp. 633。ゥ645, 1981.

[Fuj07] 藤田昌宏. SAT アルゴリズムの最新動向. 電子情報通信学会誌, Vol. 90, No. 12, pp. 1067–1972, 12月2007.

[Gal86] Jean H. Gallier. *Logic for Computer Science - Foundations of Automatic Theorem Proving*. Harper & Row, 1986.

[Hed04] Shawn Hedman. *A First Course in Logic*, Vol. 1 of *Oxford Texts in Logic*. Oxford University Press, 2004.

[Kaw02] Yuuichi Kawaguchi. A common structure of logical and algebraic algorimths. In R. Downey and et.al, editors, *Proceedings of the 7th & 8th Asian Logic Conferences*, pp. pp.222–233, Chongqing, China, August 2002.