

2 分探索木による論理関数の簡単化の一手法

後 藤 公 雄

A Simplification Method of Logic Functions
by Using Two Branches Searching Tree

Kimio GOTO

Abstract

In this paper, one method for simplification of logical functions using two-branches-searching-tree, is proposed. In this algorithm, the logical polynomial function expressed by the sum-product form is expanded into two polynomial ones with the same forms by consensus expansion procedure. These two polynomials follow the same procedures. These operations are continued to be executed in turn. Actually they are implemented by using a two-branches-searching-tree and the decimal numbers which are described as i and $2n+1-i$ respectively instead of affirmative variable and negative one. This makes the deletion of unnecessary literal string easier. The program for this algorithm was realized and compared with that for the adjacent-minterms-group-generation algorithm. As a result, this method was found to be superior to the latter.

1. は じ め に

論理関数の簡単化について筆者らは種々の手法を試みてきた^{1),2),3),4)}。ここでは主項のみを求める手法としてベトリック関数を用いる手法を提案する。この手法では、すでに知られている節展開法^{5),6)}とは異なり2進木にしたがって解くことができる。本報告ではこのベトリック関数を解くにあたってコンセンサス法⁷⁾(共有展開法⁸⁾)に類する方法を適用する場合を述べる。本報告は一面ではDasら⁹⁾の節列法に類似しているが、前者は論理変数のリテラルそのものを和積項の和項の要素としているのに対し、本報告¹⁰⁾はこれらのリテラルを10進数に置換している点が異なる。

2. 必要な諸定理とアルゴリズム

2.1 定 理

和積形式の多項式 F で、リテラル x_i を1とおいた多項式を f_1 、 x_i を0とおいた多項式を f_2 とすると、共有展開法により、

$$F = x_i \cdot f_1 + f_2 \quad (1)$$

が成立する⁸⁾。

2.2 アルゴリズム

ここでは和積形式の関数を与えられているときこれを簡単化して全主項を求める方法について述べる。

[ステップ1] 与えられた積和形式の関数 g を最小項の和に変換する。

[ステップ2] 全体の最小項の中でステップ1で求めた最小項を除去し、残りの最小項の和で関数 \bar{g} を表す。

[ステップ3] この \bar{g} を再度否定し、元の g を求めると、この g は和積形成に変換される。

[ステップ4] 各リテラルを10進数に変換する。この際上位から i 番目のリテラルで肯定形のは i 、否定形のは $2n+1-i$ と置く。

[ステップ5] このようにして求めたベトリック関数 P の各和項を、それを構成するリテラルの文字列として表し、これらの文字列の集合を P_s とする。

[ステップ6] P_s の構成要素に含まれるリテラルの中で発生頻度が最大になるリテラルを X_K とする。

[ステップ7] X_K を含む P_s の要素を除去し、残り

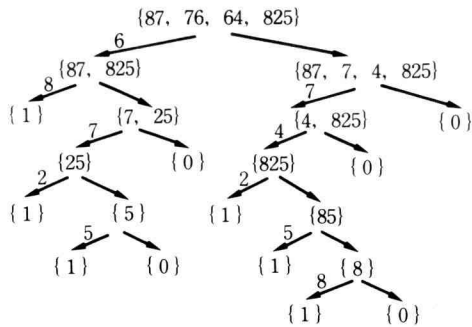


図1. 2分探索木による解法

Fig. 1. Solution method by using 2 branch searching tree

の要素のみで構成される集合 P_S^L を作り, P_S と P_S^L を結ぶ分枝を発生させ, 分枝上に X_K を付ける。

[ステップ8] X_K を含む P_S の要素から X_K のみを除去して残ったリテラルを新要素に書き直して新たな集合 P_S^R を作る。

[ステップ9] P_S^L と P_S^R についてステップ5, 6および7を繰り返す。

[ステップ10] ステップ7の手法によって集合の構成要素が全部なくなれば成功終端として {1} を記入する。またステップ8の手法によって集合の構成要素の少なくとも1個が消失すれば, 失敗終端として {0} を記入する。

[ステップ11] 成功終端 {1} を発生した分枝上を遡行し, 分枝上に付いているリテラルよりなる文字列を作り, それらを要素とする集合 P_R を作る。

[ステップ12] 集合 P_R の要素の中で他に要素のカバーされるものがあれば, カバーする要素を除去する。

[ステップ13] 最後に集合の各要素のリテラルを論理変数に変換すると各要素は主項となる。

2.3 事例

ステップ3の和積形式の関数が(ここでは便宜上最小項の事例は避ける),

$$F = (\overline{A} + \overline{B})(\overline{B} + \overline{C})(\overline{C} + D)(\overline{A} + B + \overline{D}) \quad (2)$$

で表されるとき, 2進木を用いて解いてみる。これを図1に示す。ここ図より, 解の P_R は

$$P_R = \{68, 675, 478\} \quad (3)$$

$$\therefore F = \overline{A}\overline{C} + \overline{B}\overline{C}D + \overline{A}BD \quad (4)$$

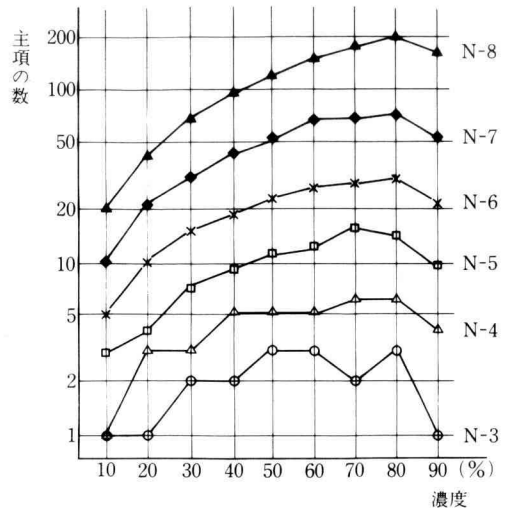


図2. 濃度と主項の数

Fig. 2. Relation between truth table density and number of prime implicant

が求まる。

3. 実際のプログラムの実行結果

上述したアルゴリズムを用いて作成したプログラムによって最小項の論理和形式が与えられた論理関数から主項を求める場合の実行結果を図2, 3に示す。図2は本アルゴリズムによって求めた最小項の濃度と主項の関係を表し, 図3は本法と隣接最小項グループ成長法³⁾による場合について濃度と主項生成の演算時間の関係を比較している。図2の結果は他の方法による場合と同じ結果を示し, 図3では2つの方法について濃度と演算時間の関係が全く対照的になっている。図2, 3はともに論理変数の個数をパラメータとしている。これらの図にはFORTRAN77を用いて作成したプログラムを大型計算機 M170F (富士通) で実行した結果が示してある。

4. 結果の検討

濃度の増大とともに隣接最小項グループ成長法では急激に演算時間が増大するのは主項生成が急激に増大することに起因する。しかし本方法では濃度の増大によって逆に演算時間が減少し, しかも濃度70%まではその変化はなだらかで70%を越すと比較的急激に演

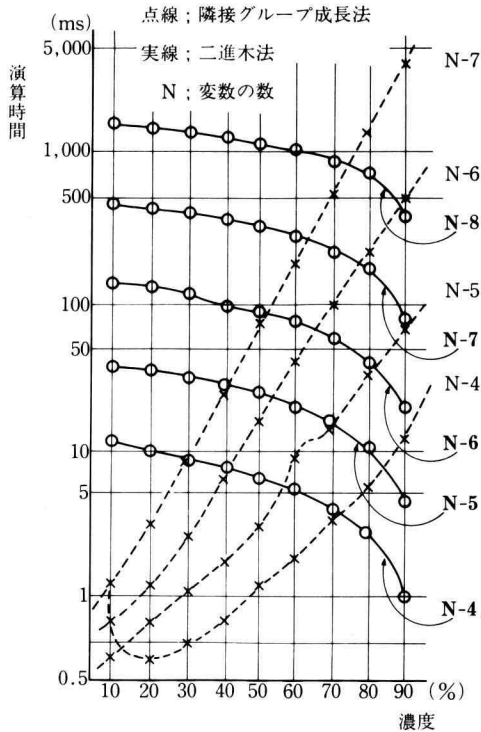


図3. 実行時間の比較

Fig. 3. Comparison of program operation time

算時間が減少する。これは2進木の繰返し頻度が濃度によって余り変わらないことを意味している。

5. む す び

最小項の濃度変化による演算時間の不変性に注目し、全体として演算時間を減少させる検討を行う予定である。具体的にはペトリック関数の解き方や、プロ

グラムの作成にあたり探索木を縦型探索法から横型探索法に変更するなどの諸点について検討を進めたい。

なお、本研究の推進に努力された昭和62年度卒業研究生杉山卓哉君に謝意を表する。

6. 参 考 文 献

- 1) 後藤：“最小項添字による隣接グループの予測による論理関数素項の計算手法について”，昭61電気学会全大，1381.
- 2) 後藤：“最小項の隣接グループ作成による論理関数の簡単化手法について”，昭61電気関係学会関西支部連大，G6-10.
- 3) 後藤：“隣接最小項グループとその分割法による論理関数簡単化手法の比較”，昭62電気学会全大，1443.
- 4) 後藤他：“隣接最小項グループの逆方向作成法による主項の生成”，昭62信学会全大，572.
- 5) J.R. Slagle 他：“A New Algorithm for Generating Prime Implicants”，IEEE Trans., vol. C-19, no. 4, p. 338 (April 1970).
- 6) 上林，岡田，矢島：“節展開法を用いた論理関数の主項の生成”，信学会論文誌，79/2, vol. J62-D, no. 2, p. 89.
- 7) P. Tison：“Generalization of Consensus Theory and Application to the Minimization of Boolean Functions”，IEEE Trans., vol. EC-16, no. 4, p. 446 (August 1967).
- 8) 矢坂：“論理関数の共有展開”，信学技報，EC85-2.
- 9) S.R. Das et al.：“Clause-Column Table Approach for Generating All The Prime Implicants of Switching Functions”，IEEE Trans., short notes, p. 1239 (Nov. 1972).
- 10) 後藤他：“2進木による論理関数の簡単化手法”，昭63情報処理学会第63回全大，7X-3.