

拡張許容関数集合を用いたトランスダクション法の改良

1 R-1

幸田武範† 田中克典‡ 上林彌彦†

京都大学工学研究科† 京都大学工学部‡

1 はじめに

許容関数は回路最適化手法の1つであるトランスダクション法において、回路の冗長性を表すために用いられる概念である。トランスダクション法は、許容関数によって表現される回路の冗長性をもとに結線の追加や削除などの回路変換を行う手法である。一般に許容関数計算時に行う冗長結線の選択は容易ではなく、トランスダクション法においても回路内の局所的な情報をもとに複数の選択肢の中から1つを選んでいるにすぎない。そのため、各部分で最良と思われる選択を行っても、全体としては最良のものとならないことがある。

そこで我々は、これらの選択の決定を行わずに、未決定性を残したまま許容関数を計算し、特定の条件を満たしたとき初めて未決定要素の決定を行う方式を考案したので報告する。

2 従来型トランスダクション法とその問題点

本章ではトランスダクション法についての基本的事項と従来法の問題点について述べる。

ある結線、入力端子、ゲートの実現する関数 f を別の関数 f' に置き換えても出力端子の実現する関数が変わらないとき、この f' を許容関数 (Permissible Function) と呼ぶ。許容関数は複数考えられる。許容関数の集合のうち同時に置き換え可能なものを CSPFs (Compatible Sets of Permissible Functions) と呼ぶ。許容関数集合は 0、1、* (don't care) の3値を要素としてもつ。

CSPFs の計算の際、不完全指定要素 * の決定方針によって、回路変形の自由度が大きく変化する。例えば、NOR 素子の場合、出力が 0 のとき入力結線のうちいずれかが 1 になればよいため、残りを * とすることができる。そのため、図1のように入力結線で論理関数が 1 となるものが複数存在するとき、1 を割当てる結線の選択が問題になる。この選択を各ゲートの CSPFs 計算時に行うため、回路入力に近いゲートまで計算を行うと、あるゲートでの選択が結果的に最良のものではないことがある。そのため、従来型トランスダクション法では最適な結果が得られないことがある。

図1に CSPFs 計算の例を示す。 v_1 及び v_2 の入力結線の CSPFs は各々2通りの選択肢がある。各々ゲートの CSPFs 計算の段階でどちらか一方に決め、 v_3 、 v_4 に伝搬する。そして、 v_3 、 v_4 の入力結線の CSPF を計算し、 v_5 に伝搬し、 v_5 の入力結線の CSPFs を計算する。

その結果、 v_5 の入力結線の CSPF は4通りのうちいずれかになる。4通りのうち1通りの CSPFs がその入力結線が切断可能であることを示している。この結線を切断するためには $v_1 \sim v_4$ で適切な選択を行う必要がある。しかし、従来法ではあとの複数段にわたる影響を考慮せずに、CSPFs の決定を行うため、このような判断は難しい。本

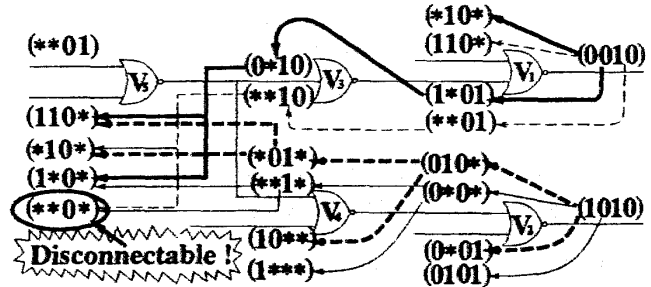


図1: CSPFs の計算 (矢印は CSPFs 計算の流れ)

手法ではこの問題に対処すべく、拡張許容関数集合を用いる。

3 拡張許容関数集合

3.1 拡張許容関数集合

ここでは NOR 素子のみを扱うが、他のゲートについても一般化は容易である。前章で示した CSPFs 計算時の問題を防ぐため、提案手法では CSPFs に未決定要素 0* (0 または * の候補)、 1* (1 または * の候補) を加えた拡張許容関数集合 SSPF (Super Set of Permissible Functions) を用いている。SSPF を用いることにより前章の問題は解

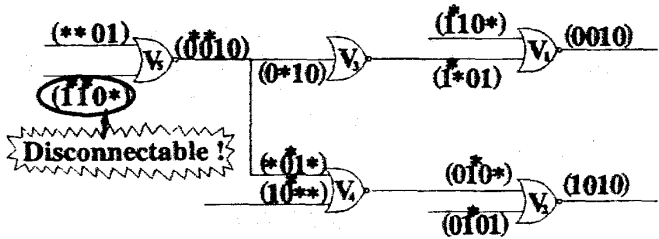


図2: SSPF の計算

決できる。図2のように、 v_1 、 v_2 での入力結線の CSPFs の選択は必要なくなり、 v_5 で切断可能な入力結線を見出すことができる。

3.2 拡張許容関数集合の計算

ゲート出力の SSPF が計算されているとき、入力結線の SSPF はゲート入出力関数により計算される (表1)。

表1: ゲート入力の SSPF の計算

出力	入力結線の論理関数	
	0	1
0	*	{ 1 の入力結線が単数なら 1 1 の入力結線が複数なら 1* }
1	0	-
*	*	*
0*	*	1*
1*	0*	-

0*、 1* といった未決定要素の決定は回路変形の際などに行われる。

Transduction Method based on Super Set of Permissible Functions.

Takenori KOUDA† Katsunori TANAKA‡
Yahiko KAMBAYASHI†

Department of Information Science, Kyoto University†
Faculty of Engineering, Kyoto University‡

3.3 拡張許容関数の伝搬

あるゲートの出力には、その出力につながる各ゲートより SSPF が伝搬されてくる。表 2 に示す優先度に従い、それらの値の中で優先度の最も高い値を当該ゲート出力の SSPF の値として採用する。

表 2: SSPF の優先度

高 ← 低	
1	1*
0	0*

4 拡張許容関数を用いた回路変形

4.1 接続可能性と切断可能性の判定

接続可能性の判定は、被接続ゲート v_1 の SSPF $G(v_1)$ と接続ゲート v_2 の論理関数 $f(v_2)$ を用いて行う。

$$Conn = G(v_1) \#_1 f(v_2)$$

$Conn$ が恒真関数になったとき、接続可能である。但し、 v_1 の SSPF が 1* かつ v_2 の論理関数が 1 のとき、 v_1 の SSPF が 1 である可能性がなくなるので、 v_1 の SSPF を * に決定する。

表 3: 2 項演算子 #1

#1	$G(v_1)$				
$f(v_2)$	0	1	*	0*	1*
0	1	1	1	1	1
1	1	0	1	1	1

切断可能性の判定は、入力結線 c の SSPF $G(c)$ を用いて行う。

$$DisConn = \#_2 G(c)$$

$DisConn$ が恒真関数になったとき、切断可能である。結線切断後、出力の SSPF が 0* で、全ての入力結線の SSPF の中に 1* が無くなった場合、出力の SSPF が 0 に決定される可能性はなくなるので、出力の SSPF は * に決定される。

表 4: 単項演算子 #2

#2	$G(c)$				
	0	1	*	0*	1*
	1	0	1	1	1

4.2 未決定要素の決定による後戻り処理

図 2 の v_1 の SSPF の第 1 要素のように、出力の SSPF が 0、入力の SSPF が (1*, 1*) の組合せであったとき、入力の SSPF は (1, 1)、(1, *), (*, 1) という許容関数集合の組合せだけでなく、(*, *) という不都合な許容関数集合の組合せまで含んでしまう。そのため、回路変形の際に未決定値の決定が起こるとき、その決定を出力側へ伝えなければ、回路出力に不都合が生じる可能性がある。

図 3 に未決定要素の決定の例を示す。 v_4 の太線で示された入力結線を切断する (①) と v_4 の論理関数が増える。そのため、 v_4 の SSPF の値は 0* から * に決定される (②)。また同様に、 v_2 の入力結線の SSPF が変化する (③) ことによって、その出力の SSPF も変化する (④)。そのため v_2 の入力結線の SSPF を再計算する必要がある (⑤)。同様に v_4 、 v_2 、 v_1 、 v_3 、 v_5 の順番に SSPF の再計算をする (⑥~⑦)。その再計算の結果、 v_5 の太線で示された結線は SSPF が 1 に決定されるため切断できない。また、出力の SSPF の値が 1* であるゲートに、実現する論理関数の対応する値が 1 であるゲートを接続する

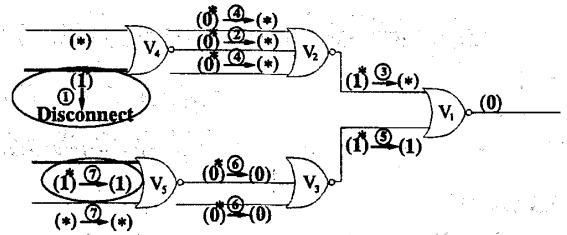


図 3: 結線切断時の決定の例 (括弧内の値は SSPF)

とき、前述のように、出力の SSPF が 1* から * に変化する。同様の計算が必要である。よって、従来法と異なり、回路出力方向へ後戻りし再計算することが必要となる。

4.3 決定の衝突

また、未決定値の決定から生じる伝搬が衝突し、不都合が発生させる場合がある。

最適化対象の回路中の部分回路に図 3 のような回路があった場合、 v_4 の太線の入力結線を切断した (①) とき、そのゲートの出力の SSPF の値を 0* から * に決定する (②)。すると、前述の決定伝搬の再計算が起こり、 v_2 、 v_3 の出力の SSPF の値が 1* から * に決定される (③、④)。すると v_1 の出力の SSPF の値は 0 であるため、この決定の伝搬は矛盾する。そのためこのような場合には、 v_1 の入力のどれか 1 つの SSPF を 1 に決定する必要がある。

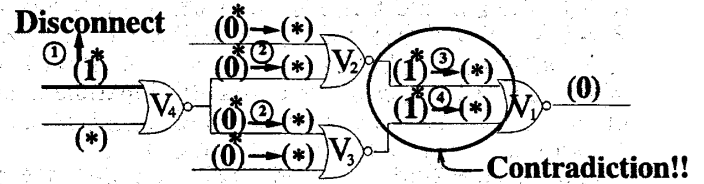


図 4: 矛盾する決定の例 (括弧内の値は SSPF)

5 終わりに

本手法で用いている SSPF は回路変形の自由度を上げることができ、工夫次第ではより最適な解を出す可能性があると考えられる。4.2 章で述べたように、ある結線を切断することによって他の結線が切断できなくなることがある。そのため、切断する結線の選択法を工夫することが重要な問題として残されている。

謝辞

有益な御助言を頂いた上林研究室の皆様様に深謝する。

参考文献

- [1] S.Muroga, Y.Kambayashi, H.C.Lai, J.Niel, Culliney: "The Transduction Method - Design of Logic Networks Based on Permissible Functions", IEEE Transactions on Computers, Vol.38, No.10, pp.1404-1424, 1989
- [2] S.Minato, N.Ishiura, S.Yajima, "Shared Binary Decision Diagram with Attributed Edges for Efficient Boolean Function Manipulation", Proceedings of 27th Design Automation Conference(1990), 52-57.