

静的学習を用いたテスト生成におけるバックトラック数の削減評価

日大生産工（院） ○小松 正樹 日大生産工 細川 利典
 九大 吉村 正義 明大 山崎 浩二

1. はじめに

近年、半導体微細化技術の進歩に伴い、大規模集積回路（Large Scale Integrated Circuit: LSI）が大規模化し、テスト生成[1][2]時間の増加が問題となっている。

テスト生成の高速化のため、一意に値を決定できる信号線をできるだけ多く求める必要がある。そのための手法として間接含意[3]が提案されている。間接含意は SOCRATES[3]で提案された静的学習[3]で求める。テスト生成および静的学習の処理で、回路内の信号値を一意に決定していく含意操作[2]を行っている。含意操作は直接含意と間接含意に分けられ、間接含意は直接含意の結果から求められる。静的学習によりテスト生成の前処理の段階で間接含意を求め、テスト生成中の含意操作で利用することで、テスト生成時間を高速化する。

静的学習の利点として、テスト生成中の解の探索空間が削減されることが挙げられるが、静的学習自身の実行時間、静的学習の結果による無駄な範囲の含意操作処理が問題となっている。そのため、有効な静的学習を解析する。本論文では静的学習を行わない場合、学習規則 A[3]を用いる場合の各故障のテスト生成中のバックトラック数[1]の削減を示すことによって、静的学習の有効性を示す。

2. 含意操作

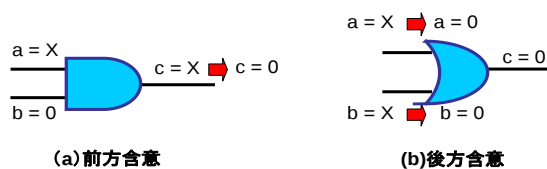


図 1 直接含意例

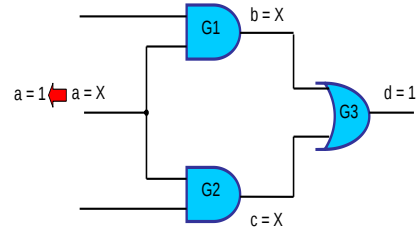


図 2 間接含意例

本章では含意操作について説明する。含意操作はゲートの入出力の接続関係から値を求めることのできる直接含意[4][5]と、ゲートの入出力の接続関係だけでは値を求めることのできない間接含意[3]がある。また直接含意は前方含意と後方含意に分けられる。

図 1 に直接含意例を示す。図 1(a)は前方含意の例である。信号線 b の信号値が 0 である場合、AND ゲートの接続関係から信号線 c の信号値は 0 と決定できる。このようにゲートの入力から出力側に値を決定する処理が前方含意である。図 1 (b)は後方含意の例である。信号線 c の信号値が 0 である場合、OR ゲートの接続関係から、信号線 a および b の信号値は 0 と決定できる。このように出力側から入力側に値を決定する処理が後方含意である。

直接含意は論理ゲートの入出力関係と推移律を利用して容易に得ることができる。

図 2 に間接含意例を示す。信号線 d の信号値 1 とした場合、直接含意では他の信号線は決定できないが、 $b=1$ または $c=1$ のいずれかが成り立たなければならないため、いずれの場合も $a=1$ が成り立つ。図 2 のように “ $d=1$ ならば $a=1$ ” という処理が間接含意である。

Evaluation of Backtrack Reduction on Test Generation Using Static Learning

Masaki KOMATSU, Toshinori HOSOKAWA, Masayosi YOSHIMURA
 and Kouji YAMAZAKI

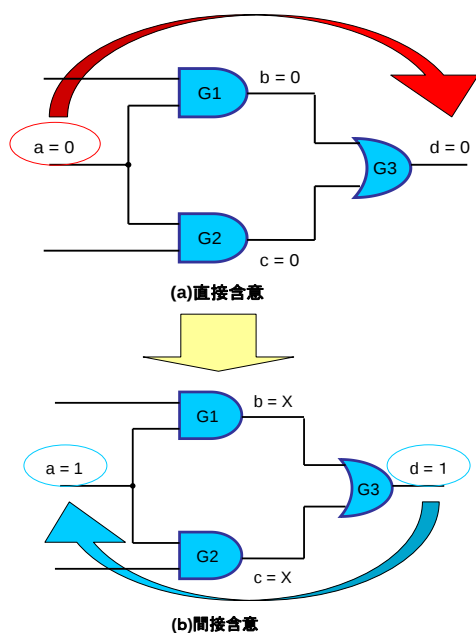


図 3 静的学習の例

3. 静的学習

静的学習では直接含意の結果から間接含意を求める。そのため、命題が真ならばその対偶も真であるという対偶規則を用いることができる。本論文の静的学習では次の学習規則 A[3]を満たす直接含意が存在するとき間接含意を保存する。

[学習規則 A]

信号線 s の値を $v(v \in \{0,1\})$ とする前方含意操作により、ある 2 入力以上のゲートの出力信号線 t が値 $w(w \in \{0,1\})$ をとり、かつ、 w がそのゲートの非制御値であるとき、間接含意として “ $t = \neg w$ ならば $s = \neg v$ ” を保存する。ここで非制御値は AND ゲートに対する論理値 1 のようにゲートの入力値がすべて決まったときに限り出力が決定される値である。

図 3 に静的学習の例を示す。図 3(a) は、まず信号線 a の信号値を 0 とし、前方含意操作を行なう直接含意の例を示す。その結果、“ $a=0$ ならば $d=0$ ” という命題が成り立つ。この対偶 “ $d=1$ ならば $a=1$ ” という命題も真となる。信号線 d は 2 入力以上のゲートの出力信号線であり、値 0 はゲートの非制御値のため、図 3 (b) のように “ $d=1$ ならば $a=1$ ” という間接含意を学習する。

学習を行なう信号線の順序は文献[6]で学習順序として効果が高いとされる回路の論理段数を基にした前方への幅優先探索による学習順序を用いる。

4. バックトラック

含意操作で矛盾が生じるか、どの外部出力にも故障が伝搬できない場合、他の信号値の組合せを調べるためにバックトラック [1] を行う。バックトラックは、決定木の最後の決定ノードの信号線に割当てた信号値を 0 から 1、または 1 から 0 に反転する操作である。

図 4 に間接含意を用いないテスト生成の例を示す。図 4 の回路の信号線 f の 0 縮退故障を検出するテスト生成を行う。図 5 は図 4 の間接含意を用いないテスト生成で生成される決定木である。まず信号線 f の 0 縮退故障を励起するため、信号線 f の信号値 1 にする。他の信号線の信号値が決定しないため、決定木に決定ノードとして信号線 d を信号値 0 で含意することを追加する。次に信号線 d を信号値 0 で含意するために、決定木に決定ノードとして信号線 a を信号値 0 で含意することを追加する。しかし図 4(a) のように信号線 a が信号値 0 であることから、信号線 e が信号値 0 となり、信号線 e と信号線 d から信号線 f が信号値 0 となることで、割当ててある信号値 1 と衝突する。よって信号線 a でバックトラックが起こり、信号線 a に信号値 1 を割当てて。その結果、図 4(b) のように信号線 b を信号値 0、信号線 c を信号値 1 と矛盾なく割当てられる。すべての外部入力の値が決定し、故障が外部出力に伝搬されているため、テスト生成を終了する。

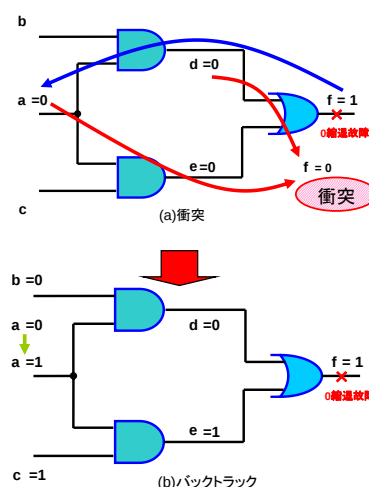


図 4 間接含意を用いないテスト生成

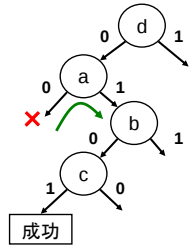


図 5 間接含意を用いない場合の決定木

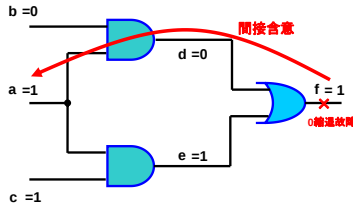


図 6 間接含意を用いるテスト生成

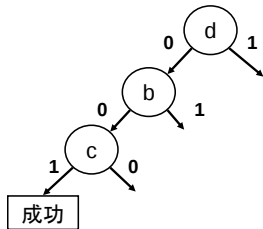


図 7 間接含意を用いた場合の決定木

図 6 に間接含意を用いるテスト生成の例を示す。図 7 は図 6 の静的学習を用いた場合に生成される決定木である。間接含意を用いるテスト生成は、テスト生成の前段階で静的学習を用いて間接含意「信号線 $f=1$ ならば信号線 $a=1$ 」を学習していたとする。間接含意を用いるテスト生成でも同様に、まず信号線 f に信号値 1 を割当て、次に間接含意によって信号線 a の値が 1 に決定する。よって図 7 の決定木に示すように信号線 a の信号値を決定木を用いて判定する必要がなくなる。最後に図 4 と同様に信号線 b に信号値 0 を、信号線 c に信号値 1 を割当てテスト生成を終了する。

図 5 と図 7 を比較すると、間接含意を用いることで信号線 a の値を、決定木を用いて判定する必要がなくなり、信号線 a の値割当てに対するバックトラックも無くなることがわかる。このように間接含意によってテスト生成中の解の探索空間は削減され、バックトラック数は減少する。

5. 実験結果

組合せ回路の各故障に対して、静的学習を実行しない場合と学習規則 A を実行した場合のテスト生成時のバックトラック数を比較した。実験対象として ISCAS89 ベンチマークの組合せ回路の c3540 と c2670 を用いる。表 1 に学習前と学習後におけるバックトラック数の比較を示す。表 1 より c3540 と c2670 において最大 101 回のバックトラック数を削減できる故障が存在した。これは間接含意によって一意に決定する値が増加し、決定木の解の探索空間が削減されたためと考えられる。静的学習がテスト生成に有効な例である。一方で学習を行ったためにバックトラック数が増加し、バックトラックの削減数が負の値となる故障が存在する。これは静的学習によって回路の広範囲に無駄な値割当てが広がり、バックトラック数が増加したと推測される。

6. おわりに

本論文では、静的学習を導入したテスト生成におけるバックトラック数の減少から静的学習の有効性を示した。今後は得られた間接含意の内、テスト生成に有効なものを解析し、有効な静的学習の評価、解析を行う予定である。

参考文献

- [1] H. Fujiwara, “Logic Testing and Design for Testability,” The MIT Press, (1985).
- [2] M. Abramovici, M. A. Breuer and A. D. Friedman, “Digital systems testing and testable design,” IEEE Press, (1995).
- [3] MICHAEL H. SCHULZ, ERWIN TRI SCHLER “SOCRATES : A Highly Efficient Automatic Test Generation System” IEE TRANSACTIONS ON COMPUTER- AIDED DESIGN, VOL 7 NO 1, JANUARY(1988) p130

- [4] 南山哲郎 高松雄三 “組合せ回路における冗長故障の判定法に関する考察” 電子情報通信学会論文誌(1998)
- [5] 梶原誠司 猿渡慶太郎 “静的学習における効率的な間接含意の発見と保存について” 電子情報通信学会 TECHNICAL REPORT OF IEICE (2002)
- [6] Hideyuki ICHIHARA, Seiji KAJIHARA, Kozo KINOSITA : On Processing Order for Obtaining Implication Relations in Static Learning , IEICE Trans. (2000).

表 1 学習前と学習後におけるバックトラック数の比較

回路	信号線	縮退故障	学習前	学習後	削減バックトラック数
c3540	N403	1	101	0	101
	N404	1	101	0	101
	N3259	1	101	1	100
	N2891	0	101	1	100
	N2505	0	101	3	98
	N3253	0	101	3	98
	N3342	1	74	0	74
	N3346	0	74	0	74
	N1730	0	31	1	30
	N1733	0	31	1	30
	N356	1	14	0	14
	N357	1	14	0	14
	N2903	0	13	0	13
	N3524	0	13	0	13
	N3119	0	10	0	10
	N954	1	10	0	10
	N3524	1	8	0	8
	N3523	1	8	0	8
	N2260	1	8	1	7
	N2538	0	7	0	7
	N3523	0	7	1	6
	N2854	1	7	1	6
	N1000	0	5	1	4
	N2004	0	5	1	4
	N3434	1	3	0	3
	N2659	0	3	0	3
	N995	1	3	1	2
	N3547	1	2	0	2
	N1808	1	0	1	-1
	N2831	1	3	4	-1
	N3277	0	0	2	-2
	N2257	0	0	3	-3
	N3475	0	53	101	-48
c2670	N2832	1	53	101	-48
	N2673	1	101	0	101
	N2266	0	101	0	101
	N1870	0	101	1	100
	N1885	0	101	1	100
	N1872	0	101	2	99
	N2323	0	101	2	99
	N2275	0	101	3	98
	N2315	0	101	3	98
	N748	1	101	4	97
	N2307	0	101	4	97
	N1368	0	51	0	51
	N1371	0	51	0	51
	N2103	0	36	9	27
	N2103	1	36	9	27
	N1232	0	36	9	27
	N1361	0	6	0	6
	N1204	0	6	0	6
	N1206	0	6	0	6
	N929	0	6	1	5
	N928	0	6	1	5
	N964	1	3	0	3
	N977	1	3	0	3
	N508	1	1	0	1
	N1850	0	1	0	1
	N154	1	0	2	-2
	N1245	1	0	2	-2
	N2100	0	36	101	-65
	N2100	1	36	101	-65