

必須割当て情報に基づくホップフィールド型

ニューラルネットワークを用いたテスト生成の高速化

日大生産工(学部) ◦瀬口 知洋 日大生産工 細川 利典

日大生産工 山内 ゆかり 日大生産工 新井 雅之

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 (Very Large Scale Integrated circuits : VLSI) が大規模化・複雑化している. そのため, 検出困難な故障に対するテスト生成時間の増加が問題となっている. その問題を解決するための手法として, Chakradhar らによってホップフィールドネットワークを用いた組合せ回路のテスト生成法が提案されている[1].

現在, ニューラルネットワークは幅広い分野での適用がなされ, 特にホップフィールドネットワークは, NP-完全として知られる巡回セールスマン問題を高速に解くことが実証されている[2]. Chakradhar らによる手法[1]では, ニューロンの状態が 0, 1 の離散値をとる 2 値ニューラルネットワークによって解かれ, 従来のテスト生成法とは本質的に違うものとなる. 従来のテスト生成アルゴリズムで検出困難な故障に対するアプローチとして期待することができ. この手法が提案された段階では, コンピュータの演算性能などが十分ではなかったため, 大規模なニューラルネットワークでの実装は膨大な実行時間を要する. これにより, 従来のテスト生成アルゴリズムと比較して有効性が示されなかった. しかしながら, 従来手法が提案された段階よりも現在のコンピュータの演算性能はさらに向上しているため, 本論文で提案する手法を適用することで大規模回路の実装に対して期待がもてる.

本論文では大規模ニューラルネットワークでの実行時間削減のために論理回路の各故障の必須割当て

[3]を求める. 必須割当て情報に基づきニューラルネットワーク上でのニューロンの値を固定することによって, ニューロンの状態更新の際にランダムに選択される更新対象のニューロン数を削減し, テスト生成の高速化を図る.

2. ホップフィールドネットワークを用いたテスト生成法

2.1. 2 値ホップフィールドモデル

ホップフィールドネットワークは相互結合型のネットワーク構造であり, ニューロンの閾値とニューロン間の相互作用の重みによってその振る舞いが決定される. ここで, ニューロン i の状態を V_i と表し, $V_i \in \{0, 1\}, i = 1, 2, \dots, N$ とする. N はニューロンの数であり, $V_i(t)$ は, ある時刻 t におけるニューロンの状態を表す. 本手法では, 非同期式のニューラルネットワークを想定し, ランダムに選ばれた 1 個のニューロンの状態を式(1)に従って状態を更新する.

$$V_i(t+1) = \begin{cases} 1 & \text{if } \sum_{i \neq j} T_{ij} V_j(t) + I_i > 0 \\ 0 & \text{if } \sum_{i \neq j} T_{ij} V_j(t) + I_i < 0 \\ V_i(t) & \text{if } \sum_{i \neq j} T_{ij} V_j(t) + I_i = 0 \end{cases} \quad (1)$$

式(1)において, T_{ij} はニューロン i からニューロン j のニューロン間の重み, I_i はニューロン i の閾値を表す.

式(2)にホップフィールドネットワークで定義されているエネルギー関数を示す.

$$E = -\frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N T_{ij} V_i V_j - \sum_{i=1}^N I_i V_i + K \quad (2)$$

式(2)において, K は定数である.

Acceleration of Test Generation Using a Hopfield Neural Network Based on Necessary Assignment Information

Chihiro SEGUCHI, Toshinori HOSOKAWA, Yukari YAMAUCHI and Masayuki ARAI

ホップフィールドネットワークでは、すべてのニューロン*i*とニューロン*j*について、 $T_{ij} = T_{ji}$ でかつ、すべてのニューロン*i*について $T_{ii} = 0$ であれば、ネットワークはエネルギーが減少する方向に状態変化し、状態変化がそれ以上起こらなくなる安定した状態に常に到達することが示されている。本手法では、ニューラルネットワーク全体のエネルギー値がニューロンの状態更新によって 0 になった際に、論理ゲートの外部入力に対応するニューロンの状態をテストパターンとする。しかしながら、必ずしも求めたい最適解に収束する保証はないため、局所解から脱出するために、ボルツマンマシンが提案されている[4]。文献[4]では、*k*番目のニューロンについて、状態が 0 のときと 1 のときのエネルギー差を ΔE_k とすると、式(3)で表される確率 P_k でニューロン*k*の状態値を 1 に設定するように決められている。

$$P_k = \frac{1}{1 + e^{-\Delta E_k/T}} \quad (3)$$

ここで、パラメータ*T*はネットワークの温度を表し、温度はアニーリングスケジュール[5]を用いて低下させる。

2.2. 論理ゲートのネットワーク化

任意の組合せ回路はニューラルネットワークによって表現可能である[1]。回路内の各信号線はニューロンによって表され、信号線の値は 2 値のニューロン状態と対応している。ニューラルネットワークは 2 入力論理ゲートに対応し、3 入力以上の論理ゲートは 2 入力論理ゲートを組み合わせて構成される。組合せ回路はニューラルネットワークのニューロン間の重み*T*と閾値*I*によって表現される。回路内の全エネルギーについて、関数機能に無矛盾なときは、エネルギーが最小になり、関数機能に矛盾が生じたときは、より高いエネルギーになるようにする。論理ゲートから変換されたネットワークの閾値や結合間の重みは、各論理ゲートの機能に矛盾が生じないようにエネルギー関数によって設定される[1]。

2.3. 組合せ回路におけるテスト生成

組合せ回路は、ニューロンの基本集合より構成することができる。図 1 に組合せ回路におけるホップフィールドネットワークを用いたテスト生成モデルについて示す。このニューラルネットワークは、故障のない正常回路と故障を挿入した故障回路を接続して構成されている。これらの外部出力は出力インターフェイス回路へ接続されている。出力インターフェイス回路は、故障検出するための必要十分条件を実現したものであり、図 1 にその構造を示す。ある故障に対するテストベクトルが存在する場合、これらのニューラルネットワークは無矛盾となる状態が存在する。これらの制約ニューラルネットワークにおいての最適解は、与えられた故障に対するテストベクトルとなる。また、本手法では、図 2 のアルゴリズムに従ってテスト生成を行う。

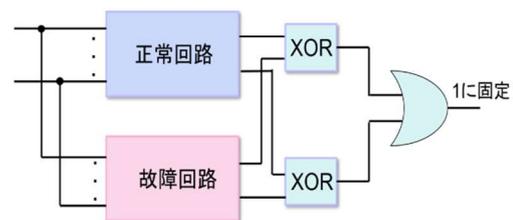


図1. 本手法におけるテスト生成モデル

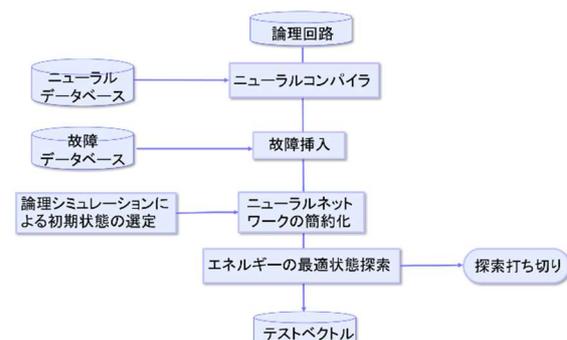


図2. テスト生成アルゴリズム

3. 必須割当て

必須割当て[3]とは、故障を検出するために必要な信号線の論理値の割当てである。本論文の必須割当てアルゴリズムでは、故障挿入、含意操作、一意活性化[5]を用いる。図 3 に必須割当てアルゴリズムを示す。また、以下にそれぞれの操作を説明する。

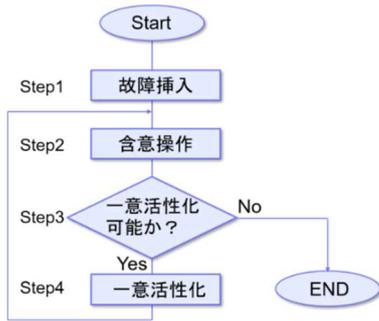


図3. 必須割当てアルゴリズム

3.1 故障挿入

故障挿入とは、ある信号線に対して故障を仮定した場合に、故障の影響を外部出力まで伝搬させるために故障を仮定した信号線に故障値の逆の論理値を割当てて操作である。ある信号線の故障を 0 縮退故障と仮定した場合、正常値として 1 を割当て、同様に 1 縮退故障と仮定した場合、ある信号線には 0 を割当てて。

3.2 含意操作

含意操作とは、ある信号線の論理値を割当てると、他の信号線の論理値が一意に決定される場合に信号線に論理値を割当てて操作である。含意操作には、ゲートの入出力関係から一意に論理値が決定される直接含意とゲートの入出力関係だけでは論理値を決定することができない間接含意が含まれる。また、直接含意には前方含意と後方含意が含まれる。図 4 に直接含意の例を示す。図 4(a)の前方含意では AND ゲートの入力信号線に割当てられた値によって出力信号線の値が一意に決定される。また、図 4(b)の後方含意では、AND ゲートの出力信号線に割当てられた値によって入力信号線の値が一意に決定される。

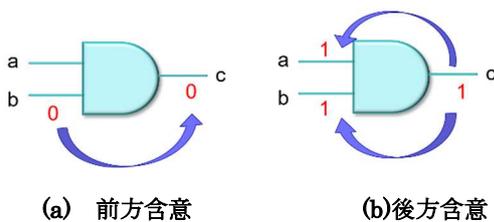


図4. 直接含意操作の例

3.3 一意活性化

一意活性化[5]とは、D フロントアが唯一のとき、D フロントアを外部出力に伝搬する際に、必ず通る部分回路を活性化するために部分回路上の各ゲートの故障が到達不可能な信号線であるサイドインプットに非制御値を割当てて操作である。文献[6]では回路構造と現在の信号線に割当てられた値から動的に一意活性化可能な信号線を求める方法が提案されている。

4. 必須割当てに基づくホップフィールドネットワークテスト生成法

必須割当てに基づくホップフィールドネットワークテスト生成法とは、2章で説明した手法により論理回路をニューラルネットワークに変換する。さらに、3章で説明したアルゴリズムによって必須割当てを求める。図 5 に信号線 3 の 0 縮退故障の例を示す。また、図 6 に図 5 の縮退故障におけるテスト生成モデルをネットワーク化したものを示す。図 6 の(a)では、従来手法を適用した際のニューラルネットワークを示し、図 6 の(b)では、提案手法である必須割当てを求めたニューラルネットワークを示す。図 6 の(b)では、図 6 の(a)と比較した際に状態更新の際にランダムに選択されるニューロンの候補数が全体から 2 つ削減される。これにより、従来手法と比較した際に、ニューラルネットワーク全体のエネルギーが減少する方向に状態変化しやすくなる。

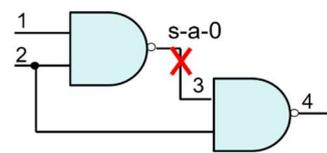
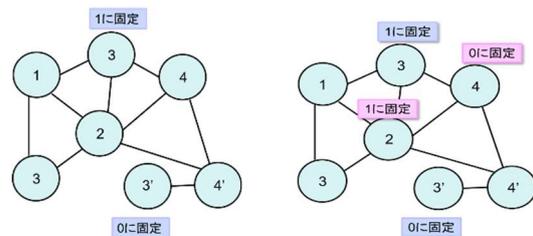


図5. 縮退故障の例



(a) 必須割当てなし (b) 必須割当てあり

図6. ホップフィールドネットワーク

5. 実験結果

本論文では、2章で述べた手法を用いてテスト生成を行い、その結果を検証する。以下に実験環境を述べる。CPU: Intel Core i7-4710MQ, 2.50GHz, メモリ: 8GB, OS: windows10, 使用言語: C言語。実験対象回路は ISCAS89 ベンチマーク回路を使用した。また、エネルギーの最適状態探索では、ニューロンの更新回数上限を 10 回に設定し、実験を行った。

表 1 に実験結果を示す。提案手法では、従来手法よりも故障検出率が平均 30.43%, 最大で 61.41% の向上した。また、実行時間は cs5378 の回路のみ大幅に削減できた。これにより、提案手法は、大規模な回路でのテスト生成の高速化に有効であることが示された。しかしながら、ニューロンの更新上限 10 回では、大規模回路での故障検出率向上や特定の信号線の故障検出は難しくなるため、ホップフィールドネットワークのエネルギー関数が局所解から効率的に脱出するためのニューロンの更新上限回数やアニーリングスケジュールを考える必要がある。

表 1 ISCAS89 の実験結果

回路名	対象故障数	ATPG手法	故障検出数	テストパターン数	故障検出率[%]	CPU時間(sec)
cs208	215	従来手法	115	6	53.48	0.18
		提案手法	212	33	98.60	0.37
cs400	424	従来手法	313	12	73.82	0.60
		提案手法	403	39	95.05	4.78
cs510	564	従来手法	360	12	63.83	1.29
		提案手法	538	53	95.39	6.83
cs820	850	従来手法	243	13	28.59	15.22
		提案手法	765	106	90.00	126.83
cs1423	1515	従来手法	1094	14	72.21	13.10
		提案手法	1260	28	83.16	16.51
cs5378	4551	従来手法	3349	28	73.59	741.30
		提案手法	3911	80	85.94	419.56

6. まとめ

本論文では、ホップフィールドネットワークを用いたテスト生成法において、必須割当て情報を用いることによる高速化を提案した。ISCAS89 ベ

ンチマーク回路において、平均 30.43% の故障検出率の向上及び平均 28.96% のテスト生成時間の高速化ができた。しかしながら、大規模回路での故障検出率向上や特定の信号線の故障検出には課題が残る。

今後の課題としては、故障検出率向上のためのニューロンの更新上限回数設定やエネルギー関数における効率的な局所解からの脱出などが挙げられる。また、実行時間の削減として、並列計算可能な個所を抽出し、FPGA で実装するハード・ソフト協調設計を実装することにより、更なる効果を発揮すると考えられる。

参考文献

- [1] Chakradhar, S.T., Bushnell, M.L. and Agrawal, V.D. "Automatic Test Generation Using Neural Networks", Proc. ICCAD '88, (1988), pp.416-419
- [2] Hopfield, J.J. and Tank, D.W. "Neural Computation of Decisions in Optimization Problems", Biol. Cybern., Vol. 52, (1985), pp.141-152.
- [3] S. B. Akers, C. Joseph, and B. Krishnamurthy, "On the Role of Independent Fault Sets in the Generation of Minimal Test Sets", in Proc. Intl. Test Conf., (1987), pp. 1100-1107
- [4] G.E. Hintom, T.J. Sejnowski, and D.H. Ackley. Boltzmann Machines, "Constraint Satisfaction Networks that Learn", Tech Report CMU-CS-84-119, Carnegie-Mellon U., May, (1984).
- [5] 藤原秀雄, "デジタルシステム的设计とテスト", 工学図書, (2004), pp135-175.
- [6] M. Abramovici, M.A. Breuer and A.D. Friedman, "Digital systems testing and testable design", IEEE. (1995).