



の組合せは、欠落箇所候補が  $n$  入力ゲートであると仮定すると、 $2^n - n - 2$  通り存在する。入力数の多いゲートが組合せ箇所に複数含まれる場合は、組合せが爆発的に増加することになる。そのために処理時間の大幅な増加が予想され、従来法では対応していなかった。

本稿ではこれら二つの項目を特定するための手法として、誤り追跡入力による不一致ゲート入力パターンを用いた手法を提案する。ここで不一致ゲート入力パターンとは、誤り箇所であるゲートにおいて、実際の出力値と出力が期待される値（期待出力値）が一致しない、ゲートの入力パターンを指す。以下、不一致ゲート入力パターンのことを単に不一致ゲート入力と呼ぶ。手法の処理手順を次に示す。

- (1) 誤り追跡入力の伝搬
- (2) 期待出力値の検出
- (3) 誤り箇所の入力パターンに含まれる  $(X, \bar{X})$  に 1 または 0 を代入し、不一致ゲート入力を検出
- (4) 不一致ゲート入力より欠落ゲートの信号線を特定
- (5) 冗長なゲートをゲート欠落箇所に挿入し、拡張 X-伝搬法により修正

図 1 のゲート欠落を含む回路例に対する、不一致ゲート入力を用いた信号線特定手法の例を以下に示す。

図 1 の部分に全入力パターンを与えることが可能な場合、図 2 に示す 4 種類の不一致ゲート入力が発生する。これら 4 種類から、欠落したゲートの入力信号線である (a, b) の信号値のみが変化していることがわかる。この性質を用いると、欠落ゲートの信号線と機能の両方の特定ができる。しかし、そのためには誤り箇所の全入力パターンに対して不一致ゲート入力であるかの判断が必要となる。全外部入力パターンを与えて調べれば確実であるが、処理時間が外部入力数に対して指数的に増加することになる。

そこで本手法では処理時間を短縮するために、限られたパターン数で誤りの限定に高い効果を上げている誤り追跡入力を用いることにより、欠落ゲートの信号線特定までを行い、ゲートの機能の特定を従来の拡張 X-伝搬法により行う。

$$\begin{aligned}
 (a, b, c, d) &= (0, 0, 1, 1), \\
 & (0, 1, 1, 1), \\
 & (1, 0, 1, 1), \\
 & (1, 1, 1, 1)
 \end{aligned}$$

図 2 不一致ゲート入力の例

表 2 想定誤り数別の検出率

想定した誤りの数	1	2	3	平均
検出率 (%)	100	89	83	88

表 3 組合せ候補箇所数別の検出率

組合せ候補箇所数	1	2~9	10~50	51~100	101 以上
検出率 (%)	94	92	81	71	82

## 5. 実験と考察

ISCAS ベンチマーク回路 C432, C499, C880, C1355, C1980, C2670, C3540, C5315 にゲート欠落を想定した回路 725 例を対象として実験を行った。拡張 X-伝搬法の処理過程ですでに生成されている 100 個の誤り追跡入力を適用した。また、検出効果を表す指標として、想定したゲート欠落箇所を検出が可能となった回路例の、全回路例に対する割合を用いる。

表 2 に示す実験結果から、本手法により従来法では修正が不可能であった回路例のうち、ゲート欠落誤りのみが回路中に単独で存在する場合には、100% 修正可能となった。また、全体としても約 88% の例で修正可能となった。

表 3 に、拡張 X-伝搬法による誤り箇所と仮定された箇所の組合せの候補数別に整理した結果を示す。この表から、候補が絞り込まれると検出率が向上することがわかる。このことから、検出率の向上には組合せ候補の箇所数の絞り込みが有効であると考えられる。

## 6. まとめ

本稿の提案手法により、ゲート過剰誤りに関してはわずかな変更のみで 100% 検出可能となった。また、単独ゲート欠落誤りに対して 100% 検出可能となり、全回路例に対しては約 88% という高い検出率が得られた。

今後の課題としては、多重誤りにゲート欠落が含まれる場合についても、検出率 100% を目指すことが挙げられる。そのためには、組合せ候補数の絞り込みと複数の不一致ゲート入力の確保の両方を同時に実現する、誤り追跡入力の生成が必要と考えられる。

### 参考文献

- [1] M. Tomita, T. Yamamoto, F. Sumikawa, K. Hirano, "Rectification of multiple logic design errors in multiple output circuits," 31st DAC, pp. 212-217, June 1994.