# Diagnostic and Detection Fault Collapsing for Multiple Output Circuits

Raja K. K. R. Sandireddy and Vishwani D. Agrawal
Auburn University
Department of Electrical and Computer Engineering, Auburn, AL 36849, USA
*sandira@auburn.edu, vagrawal@eng.auburn.edu*

## Abstract

*We discuss fault equivalence and dominance relations for multiple output combinational circuits. The conventional definition for equivalence says that "Two faults are equivalent if and only if the corresponding faulty circuits have identical output functions". This definition, which is based on indistinguishability of the faults, is extended for multiple output circuits as "Two faults of a Boolean circuit are equivalent if and only if the pair of the output functions is identical at each output of the circuit". This is termed as diagnostic equivalence in this paper. "If all tests that detect a fault also detect another fault, not necessarily on the same output, then the two faults are called detection equivalent". Two detection equivalent faults need not be indistinguishable. The definitions for fault dominance follow on similar lines. A novel algorithm based on redundancy identification has been proposed to find the equivalence and dominance collapsed sets based on diagnostic and detection collapsing. Applying the algorithm to a 4-bit ALU would collapse the total fault set of 502 faults to 253 and 155, respectively, according to diagnostic equivalence and dominance. The collapsed sets have 234 and 92 faults, respectively, for detection equivalence and dominance. In comparison, the traditional structural equivalence and dominance collapsing results in 301 and 248 faults, respectively. Finally, we use library-based functional collapsing in a hierarchical system and find that smaller fault sets are obtained with an order of magnitude reduction in CPU time for very large circuits.*

## 1. Introduction

The classical definition of equivalence says, "two faults are equivalent if and only if the corresponding faulty circuits have identical output functions". Equivalent faults are indistinguishable because they cannot be isolated from each other at the primary outputs by any input vector. Fault equivalence can be classified as structural equivalence and functional equivalence. Structural equivalence is identifiable from the circuit graph or structure. For example, all single stuck-at-1 (s-a-1) faults on the inputs and output of an OR gate are structurally equivalent. Functional equivalences involve circuits consisting of multiple gates.

Another form of collapsing that can further reduce the fault set size is dominance fault collapsing. A fault, all of whose tests detect some other fault, is said to be dominated by the other fault. For an OR gate, the output stuck-at-0 (s-a-0) fault dominates a single s-a-0 fault on any input. In the equivalence collapsed set, when a fault is not detected, the status of the entire set of faults that is equivalent to it is known. Such is not the case in the dominance collapsed set [10]. Still there are advantages of using the latter for ATPG. Structural fault collapsing alone can reduce the fault set size to about 40 to 60% of all faults. Most ATPG programs use only structural equivalence fault collapsing. The Fastest program, developed at the University of Wisconsin, can do both equivalence and dominance fault collapsing, but it does only structural collapsing [19].

In Section 2, we discuss the background and previous contributions to fault collapsing. In Section 3, the conventional definitions of equivalence and dominance are analyzed with special attention to multiple output circuits. In Section 4, a redundancy based technique has been proposed to find the dominance relations between faults. The results are discussed in Section 5. An application of the new functional collapsing algorithm is demonstrated for large circuits using hierarchical collapsing [15, 29].

## 2. Background

We use a graph model described in the literature [5, 29]. The fault equivalence and dominance relations are represented by a directed graph. In this graph each fault is represented by a node. If fault $f_1$ dominates fault $f_2$ then this is represented by a directed edge from node $f_2$ to $f_1$. This edge indicates that any test for $f_2$ must detect $f_1$. Clearly, the presence of edges $f_1 \rightarrow f_2$ and $f_2 \rightarrow f_1$ indicates that the two faults $f_1$ and $f_2$ are equivalent. Fault dominance graph, or simply a dominance graph, represents the dominance relations among the faults of a circuit.

Figure 1 shows the dominance graph for all faults of an OR gate. The subscript fault notation has been used, that is, $a_0$ means that the fault is on line named '$a$' and is s-a-0. The dominance graph is conveniently represented by its dominance matrix shown in Table 1. A 1 entry at the intersection of a row and a column means that the fault corresponding to the column dominates the fault corresponding to the row. For example, the 1 in the second row and the last

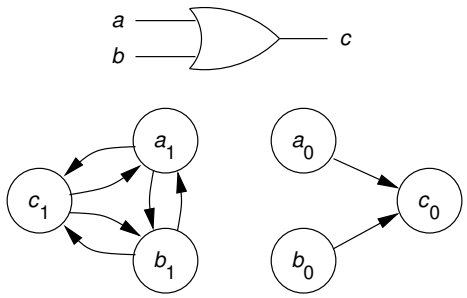


**Figure 1. Dominance graph of an OR gate.**

**Table 1. Dominance matrix of OR gate.**

| | $a_0$ | $a_1$ | $b_0$ | $b_1$ | $c_0$ | $c_1$ |
|---|---|---|---|---|---|---|
| $a_0$ | 1 | 0 | 0 | 0 | 1 | 0 |
| $a_1$ | 0 | 1 | 0 | 1 | 0 | 1 |
| $b_0$ | 0 | 0 | 1 | 0 | 1 | 0 |
| $b_1$ | 0 | 1 | 0 | 1 | 0 | 1 |
| $c_0$ | 0 | 0 | 0 | 0 | 1 | 0 |
| $c_1$ | 0 | 1 | 0 | 1 | 0 | 1 |

column indicates that $c_1$ dominates $a_1$. Equivalence of two faults is expressed by two 1's placed at both intersections of the rows and columns of those faults. Since there is also a 1 in the last row and second column indicating that $a_1$ dominates $c_1$, it can be said that $a_1$ and $c_1$ are equivalent. This dominance matrix is used in the algorithm of Section 4.1 to represent all the dominance relations between the faults.

## 2.1. Functional Collapsing

Though the above dominance relations are generally called structural, the dominance matrix can also include the relations between the functionally collapsible faults. It should be noted that the functional fault collapsing inherently includes structural collapsing.

### 2.1.1 Functional Equivalence

For an input vector, $V$, to be a test for a fault, we have [3]

$$F_0(V) \oplus F_1(V) = 1 \tag{1}$$

where $F_0$ is the fault-free function and $F_1$ is the faulty function, respectively. Consider a second fault that produces a faulty function $F_2$. According to the definition of fault equivalence, two equivalent faults have exactly the same tests. Therefore, for two faults to be equivalent, we have

$$[F_0(V) \oplus F_1(V)] \oplus [F_0(V) \oplus F_2(V)] = 0 \tag{2}$$

$$\Rightarrow F_1(V) \oplus F_2(V) = 0 \tag{3}$$

which means that the two faulty functions are identical. These equations are functionally depicted in Figure 2.

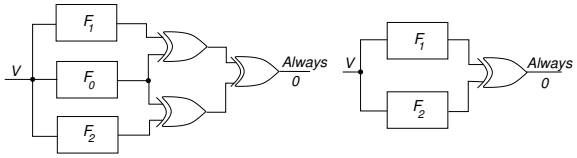


**Figure 2. Two ways to view fault equivalence.**

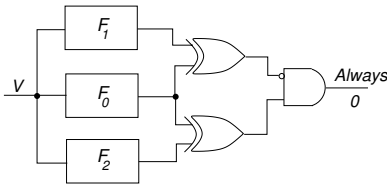


**Figure 3. Viewing fault dominance.**

### 2.1.2 Functional Dominance

If a fault $f_1$, with faulty function $F_1$, dominates another fault $f_2$, with faulty function $F_2$, then the two faults are functionally equivalent for the input vector set that tests the fault $f_2$, i.e., all tests of $f_2$ satisfy Equation 3. Let vector V detect $f_2$, so it must satisfy the following equation

$$F_0(V) \oplus F_2(V) = 1 \tag{4}$$

Since $f_1$ dominates $f_2$, any vector that satisfies Equation 4 must satisfy Equation 1. Also, by contra-positive law, any vector that does not satisfy Equation 1 must not satisfy Equation 4. These conditions are combined in Equation 5 that must be satisfied by all input vectors.

$$[F_0(V) \oplus F_2(V)] \oplus [\overline{F_0(V) \oplus F_1(V)}] = 0 \tag{5}$$

This relation, as shown in Figure 3, was explained in the paper by Agrawal *et al.* [3].

## 2.2. Previous Work

There has been considerable work in the area of fault collapsing. Several authors [12, 14, 16, 20, 23, 30, 32] concentrate on finding the fault equivalences, while others [3, 21, 29] deal with fault dominance relations. Recent papers also give methods to find fault equivalences using ATPG [14, 32] and simulation [7]. Fault equivalence identification can be based on redundancy information [8], and hence test generation can prove equivalence [16]. There are techniques to find the relations between the faults on a fan-out stem, its branches and the reconvergent points [2, 22, 25, 31].

Consider the full adder circuit as shown in Figure 4. This circuit has a total of 60 faults, the structural equivalent collapsed set has 38 faults and the structural dominance set has 30 faults. As reported in [3], the functional equivalence collapsing leads to 26 faults, while functional dominance collapsing resulted in a fault set of 14 (further reducible to 12).

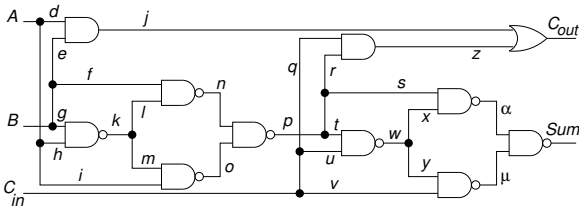


**Figure 4. A full adder circuit.**

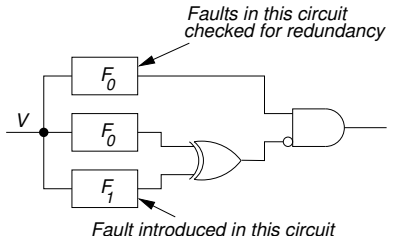


**Figure 5. Identifying functional dominance.**

## 3. Definitions

We first quote the conventional definitions of equivalence and dominance [1, 10] and then extend them to multiple output circuits.

**Definition 1:** *Fault Equivalence - Two faults of a Boolean circuit are equivalent if and only if faulty functions are identical at each output of the circuit.*

**Definition 2:** *Fault Dominance - A fault $f_i$ is said to dominate fault $f_j$ if (a) the set of all vectors that detects fault $f_j$ is a subset of all vectors that detects fault $f_i$ and (b) each vector that detects $f_j$ implies identical values at the corresponding outputs of faulty versions of the circuit [18, 27].*

This definition has been later modified as, "A fault $f_i$ is said to dominate fault $f_j$ if the faults are equivalent with respect to the test set of fault $f_j$ [1, 10]."

These definitions are extended for possible interpretations for multiple output circuits.

**Definition 3:** *Diagnostic equivalence - Two faults of a Boolean circuit are called equivalent if and only if the pair of the output functions is identical at each output of the circuit.*

This definition of equivalence, which is identical to Definition 1, implies indistinguishability of two faults.

**Definition 4:** *Detection equivalence - Two faults are called detection equivalent if and only if all tests that detect one fault also detect the other fault, not necessarily at the same output.*

Two detectable faults that are diagnostic equivalent are also detection equivalent. Faults that are detection equivalent need not be indistinguishable. Typically, a test vector generation process involves fault simulation and the faults that get detected by a test vector are dropped. Inherently, the fault simulation uses detection relations, while it is often incorrectly believed to be functional (diagnostic) relation. This observation motivated the work presented in this paper.

**Definition 5:** *Diagnostic dominance - If all tests of a fault $f_1$ detect another fault $f_2$ on the exact same outputs where $f_1$ was detected, then $f_2$ is said to dominate $f_1$.*

**Definition 6:** *Detection dominance - If all tests of a fault $f_1$ detect another fault $f_2$, irrespective of the output where $f_1$ was detected, then $f_2$ is said to detection dominate $f_1$.*

This detection dominance is same as the *test covering* relation proposed by Abramovici *et al.* [2] or *test implication* proposed by To [31]. The detection equivalence is referred to as *test equivalence* by Lioy [22] and To [31]. Like equivalence, diagnostic dominance between two detectable faults also implies detection dominance. We note that Definitions 3 and 5 (diagnostic type) are identical to Definitions 1 and 2 that are often discussed in the literature.

## 4. A New Result on Functional Dominance

Finding functional dominances using the implementation as shown in Figure 3 is a computationally expensive procedure. This is because we need to implement it for all the permutations of faults taken two at a time. A modified and less expensive scheme is shown in Figure 5 where, initially all the three blocks are the fault free copies of the circuit with function $F_0$. Consider a non-redundant fault, say $x_1$, and introduce it in the bottom block whose function is now designated as $F_1$. Consider another fault, say $y_0$, which is dominated by $x_1$ in the given circuit. Whenever $y_0$ in the top block is activated and propagated to the AND gate, it is blocked by the output of the XOR gate (a logic 1), because fault $x_1$ is also detectable when $y_0$ is detected. So, all faults that are dominated by $x_1$ in the given circuit are redundant in the top block. In a single iteration of the ATPG, we will find all faults that are dominated by the fault introduced in block $F_1$. A redundant fault in the given circuit (stand-alone $F_0$) will appear to be dominated, in Figure 5, by any fault introduced in bottom block. While that is a correct conclusion, without elaborating on the reasons, we consider only non-redundant faults in the given circuit.

In the implementation of Figure 3, the ATPG is run $n(n-1)$ times, $n$ being the number of non-redundant faults in the circuit, and in each run of ATPG, we test the redundancy of one fault. Using the scheme in Figure 5, the ATPG is run $n$ times and in each run, we carry out the redundancy test for $n-1$ faults. The algorithm used for functional collapsing is as follows:

### 4.1. Algorithm

1. Select a non-redundant fault from the given circuit and build the circuit as shown in Figure 5 with the fault introduced in the bottom block whose function is $F_1$.
2. Check for redundant faults in the top block $F_0$.
3. For each redundant fault found in step 2, a 1 is placed in the dominance matrix at the intersection of the row corresponding to the redundant fault and the column corresponding to the fault in the bottom block. Thus, we obtain all values of a column of the dominance matrix in a single iteration.
4. Go to step 1 until there is no non-redundant fault left.

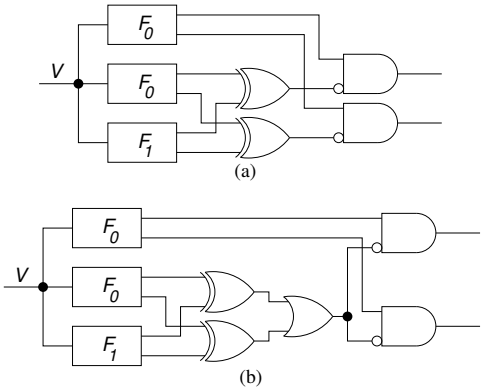


**Figure 6. Schemes for collapsing faults with a) diagnostic and b) detection criteria.**

5. At the end of the algorithm, we get the dominance matrix with all functional dominance relations included.
6. Transitive closure of the dominance matrix is computed, which is then reduced using the *algorithm equivalence* of Prasad *et al.* [29]. This reduced matrix consists of the dominance relations within an equivalence collapsed set of faults.
7. If dominance collapsing is required, then the reduced matrix of the previous step is further reduced according to *algorithm dominance* of Prasad *et al.* [29].

The scheme of Figure 5 is for a single-output circuit. For a circuit with multiple outputs, the scheme of Figure 6 is used. It illustrates a case of two outputs and the generalization for more than two outputs is straightforward. The scheme of Figure 6(a) conforms to the diagnostic collapsing as given by Definitions 3 and 5 of Section 3, while that of Figure 6(b) does detection collapsing. It should be noted that the scheme of Figure 6(b) ensures that a test vector which detects the dominating fault on an output line blocks all AND gates because of the additional OR gate, unlike in Figure 6(a). This causes more redundancies and thereby more dominance relations between faults, that is, more 1's in the dominance matrix. Hence, the fault set obtained using detection dominance will be much smaller than that obtained for diagnostic dominance for multiple output circuits.

Consider the full adder circuit shown in Figure 4. Diagnostic dominance collapsing results in a set of 12 faults while detection dominance yields a set of 6 faults, which is the smallest set among all the reported results to date.

## 5. Results

We first present four example circuits for illustration. The smallest is a simple XOR function implemented with four NAND gates and the largest is the 4-bit ALU circuit (74181). The results according to the definitions of diagnostic and detection collapsing of Section 3 are tabulated in Table 2. These results are compared with the structural equivalence collapsed set obtained using Hitec [26] and Fastest [19], and the structural dominance collapsed set obtained by Fastest [19]. The column "Functional [3]" has the values obtained using a hierarchical fault collapsing technique [3]. The XOR gates of the ALU circuit are replaced with four NAND gate implementation. The collapse ratio of each collapsed set is shown by the accompanying fraction in parenthesis. The collapse ratio is defined as the relative size of collapsed set with respect to set of all faults [10].

For the 8-bit adder, the previous best result is reported by Agrawal *et al.* [3]. Their hierarchical collapsing technique resulted in the collapsed set of 112 faults (collapse ratio 0.24) while the implementation described in this paper leads to a set of 48 faults (collapse ratio 0.10), a reduction of over 50%. This result is for detection dominance. The diagnostic dominance for multiple outputs results in a collapse ratio of 0.21, which still is smaller than that previously reported [3]. The dominances that are missed by the hierarchical collapsing technique of [3] are the functional dominances between the faults of different logic cells that cannot be found using the transitive closure of the dominance graph. Based on our experience, the collapsing using the detection dominance leads to collapse ratios in the range of 0.10-0.20. The paper by Agrawal *et al.* [4] achieves a collapse ratio of about 25%.

From Table 2, we see that the fault set of the full adder is collapsed to a set of 26 faults according to the diagnostic equivalence collapsing. There are three more equivalences which are found using detection equivalence collapsing. These faults, in the subscripted fault notation, are $j_0$ collapsed with $k_1$, $q_1$ collapsed with $u_1$, and $z_0$ collapsed with $w_1$. The lines in the full adder are named as shown in Figure 4. The fault $q_1$ which is detected at the output line $C_{out}$ by vectors 100 and 010, applied to inputs $A$, $B$, $C_{in}$, is considered as detection equivalent with $u_1$ which is detected by the same vectors at the output $Sum$.

The ATPG used for collapsing algorithms is Hitec/Proofs [26]. There were some aborted faults while checking for redundancies in step 4 of the Algorithm in Section 4.1. Had these aborted faults been treated as redundant we would have inserted additional 1's in the dominance matrix resulting in a smaller, though possibly erroneous, collapsed set. So, the results in Table 2 may not represent the smallest possible collapsed set. This is the reason why the diagnostic and detection equivalence collapsed sets of 8-bit adder are the same while the corresponding full adder collapsed set sizes are different. When no fault is aborted, using a better ATPG, the algorithms would result in the minimal collapsed fault set. But this should not be considered a problem since the use of this algorithm is recommended only for smaller circuits, where we would not have any aborted faults.

The results obtained with the algorithms have been verified as described below. A test generator is run to derive test vectors for the fault sets obtained from our algorithms. Then a fault simulator is used to determine whether the test vectors detect all faults of the circuit except the redundant ones. The Gentest ATPG [11] is used for this purpose and the number of test vectors with default options on Gentest is tabulated in Table 3. The test vectors are compared with the test vectors required for the structural equivalence col-

**Table 2. Comparison of fault collapsing results.**

| Circuit name | All faults | Number of collapsed faults (Collapse Ratio) | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | Structural [19, 26] | | Functional [3] | | Functional collapsing - New results | | | |
| | | | | | | Diagnostic criterion | | Detection criterion | |
| | | Equiv. | Dom. | Equiv. | Dom. | Equiv. | Dom. | Equiv. | Dom. |
| XOR | 24 | 16 (0.67) | 13 (0.54) | 10 (0.42) | 4 (0.17) | 10 (0.42) | 4 (0.17) | 10 (0.42) | 4 (0.17) |
| Full Adder | 60 | 38 (0.63) | 30 (0.50) | 26 (0.43) | 14 (0.23) | 26 (0.43) | 12 (0.20) | 23 (0.38) | 6 (0.10) |
| 8-bit Adder | 466 | 290 (0.62) | 226 (0.49) | 194 (0.42) | 112 (0.24) | 194 (0.42) | 96 (0.21) | 194 (0.42) | 48 (0.10) |
| ALU (74181) | 502 | 301 (0.6) | 248 (0.49) | – | – | 253 (0.5) | 155 (0.31) | 234 (0.47) | 92 (0.18) |

**Table 3. Comparison of the test vectors.**

| Circuit name | No. of test vectors (no. of target faults) | | | |
|---|---|---|---|---|
| | Structural | | Functional | |
| | Equiv. | Dom. | Diagnostic Dom. | Detection Dom. |
| Full Adder | 6 (38) | 6 (30) | 7 (12) | 6 (6) |
| 8-bit Adder | 33 (290) | 28 (226) | 32 (96) | 28 (48) |
| ALU | 44 (293) | 44 (240) | 39 (147) | 38 (84) |

**Table 4. Hierarchical fault Collapsing.**

| Circuit name | All faults | No. of collapsed faults | | | |
|---|---|---|---|---|---|
| | | Flattened (Structural) | | Hierarchical (Functional) | |
| | | Equiv. | CPU s | Equiv. | CPU s |
| Full Adder | 60 | 38 | 0.01 | 26 | 0.01 |
| 8-bit | 466 | 290 | 0.02 | 194 | 0.03 |
| 16-bit | 930 | 578 | 0.03 | 386 | 0.04 |
| 32-bit | 1858 | 1154 | 0.08 | 770 | 0.08 |
| 64-bit | 3714 | 2306 | 0.2 | 1538 | 0.1 |
| 128-bit | 7426 | 4610 | 0.7 | 3074 | 0.2 |
| 256-bit | 14850 | 9218 | 2.6 | 6146 | 0.5 |
| 512-bit | 29698 | 18434 | 10.4 | 12290 | 1.3 |
| 1024-bit | 59394 | 36866 | 44.8 | 24578 | 4.2 |
| 2048-bit | 118786 | 73730 | 188.1 | 49154 | 13.6 |

lapsed [26] and dominance collapsed [19] sets. Accompanying each entry of the test vectors, the value in parenthesis is the number of the target faults provided to the test generator. The target faults are different from that in Table 2 in case of ALU, because there were 8 redundant faults which are not considered here. Though there is a reduction in the number of test vectors for ALU, we still have a long way to go because the minimum number of test vectors to detect all the faults is only 12 [6]. It has been observed through experiments that the number of test vectors is dependent on the fault order and the selection of a test vector from among many vectors that detect a target fault.

### 5.1. Hierarchical Fault Collapsing

The redundancy-based algorithms of this paper should be used only with smaller circuits, because the time taken to compute the collapsed set increases exponentially as the circuit size grows. The collapsing results with ALU have been included just for demonstration. The collapsing techniques described in this paper can be used with hierarchical fault collapsing [15, 29]. The collapsed sets of smaller circuits like multiplexer, half-adder, xor, full-adder, etc., can be saved as library information. This information can then be used to collapse faults in larger circuits consisting of the library elements. The collapsed set of the larger circuit can again be saved for use at the next higher level of hierarchy. It should be noted that we can use detection collapsing only for the sub-circuits, whose outputs are all primary outputs. This technique has been demonstrated by collapsing different sized ripple carry adders and the results are tabulated in Table 4. All adders in the Hierarchical column are the circuits described hierarchically using a full adder. The column Functional is diagnostic equivalence collapsing set size and is compared with structural equivalence collapsing set obtained from the flattened circuit. Our collapsing program uses the *update* for computing transitive closure as described by Dave *et al.* [13]. The CPU times reported in both cases are those of the same program so that we can make a better comparison for the two collapsing methods. The time reported in seconds is clocked on a 360MHz Sun UltraSparc 5_10 machine with 128MB memory. The same program provides a dominance collapsing option and for a 64-bit adder, the collapsed set has 768 faults, a collapse ratio of 20.67%. This collapsing technique can be used for any circuit that can be described hierarchically to obtain better collapse ratios in lesser time than that required for a flattened circuit. The CPU times for the flattened circuits are also for our program and these are either similar to or better than those for other available programs [19, 26].

## 6. Conclusion

The structural collapsing techniques generally yield a collapse ratio of about 50%. When the detection dominance collapsing is used, the collapse ratio drops to the range of 10 to 20%. The fault set sizes obtained using the presented algorithm are considerably smaller than the previously reported results. The advantage of such a small collapse ratio may be in the reduction of the fault simulation effort and in the number of test vectors, though it is not clearly reflected

in Table 3. There are methods [9, 28] that aim at obtaining, though do not guarantee, the smallest vector set to detect a given fault set. The number of test vectors when compared to the lower bound given by a method due to Akers and Krishnamurthy [6] indicates that further reduction is possible. Care is needed in using dominance collapsed set since there can be instances where the dominated fault is redundant and the dominating fault (not included in the collapsed set) is testable. For fault diagnosis, we can only use the diagnostic equivalence collapsed set.

We have used an ATPG to find redundancies. There are recent methods [13, 17, 24] of redundancy identification, not based on ATPG but using non-exhaustive search, that are less time consuming. Their use may provide trade off between the CPU time efficiency of the algorithms and the smallest possible collapsed set found.

## References


[1] M. Abramovici, M. A. Breuer, and A. D. Friedman, *Digital Systems Testing and Testable Design*. Piscataway, New Jersey: IEEE Press, 1990.

[2] M. Abramovici, D. T. Miller, and R. K. Roy, "Dynamic Redundancy Identification in Automatic Test Generation," *IEEE Trans. On CAD*, vol. 11, no. 3, pp. 404–407, Mar. 1992.

[3] V. D. Agrawal, A. V. S. S. Prasad, and M. V. Atre, "Fault Collapsing via Functional Dominance," in *Proc. International Test Conf.*, 2003, pp. 274–280.

[4] V. D. Agrawal, A. V. S. S. Prasad, and M. V. Atre, "It is Sufficient to Test 25-Percent of Faults," in *Proc. IEEE* $7^{th}$ *VLSI Design & Test Workshops*, Aug 2003.

[5] S. B. Akers, C. Joseph, and B. Krishnamurthy, "On the Role of Independent Fault Sets in the Generation of Minimal Test Sets," in *Proc. International Test Conf.*, 1987, pp. 1100–1107.

[6] S. B. Akers and B. Krishnamurthy, "Test Counting: A Tool for VLSI Testing," *IEEE Design & Test of Computers*, vol. 6, no. 5, pp. 58–77, Oct. 1989.

[7] H. Al-Asaad and R. Lee, "Simulation-Based Approximate Global Fault Collapsing," in *Proc. International Conf. on VLSI*, 2002, pp. 72–77.

[8] M. E. Amyeen, W. K. Fuchs, I. Pomeranz, and V. Boppana, "Fault Equivalence Identification using Redundancy Information and Static and Dynamic Extraction," in *Proc.* $19^{th}$ *IEEE VLSI Test Symp.*, 2001, pp. 124–130.

[9] C. Benmehrez and J. F. McDonald, "Measured Performance of a Programmed Implementation of the Subscripted D-Algorithm," in *Proc.* $20^{th}$ *Design Automation Conf.*, June 1983, pp. 308–315.

[10] M. L. Bushnell and V. D. Agrawal, *Essentials of Electronic Testing for Digital, Memory and Mixed-Signal VLSI Circuits*. Boston, MA: Kluwer Academic Publishers, 2000.

[11] W. T. Cheng and T. J. Chakraborty, "Gentest: An Automatic Test Generation System for Sequential Circuits," *Computer*, vol. 22, no. 4, pp. 43–49, April 1989.

[12] F. W. Clegg, "Use of SPOOF's in the analysis of Faulty Logic Networks," *IEEE Trans. Computers*, vol. 22, no. 3, pp. 229–234, Mar. 1973.

[13] K. K. Dave, V. D. Agrawal, and M. L. Bushnell, "Using Contrapositive Law in an Implication Graph to Identify Logic Redundancies," in *Proc.* $18^{th}$ *International Conf. VLSI Design*, Jan. 2005, pp. 723–729.

[14] T. Grüning, U. Mahlsdedt, and H. Koopmeiners, "DIATEST: A Fast Diagnostic Test Pattern Generator for Combinational Circuits," in *Proc. International Conf. CAD*, Nov. 1991, pp. 194–197.

[15] R. Hahn, R. Krieger, and B. Becker, "A Hierarchical Approach to Fault Collapsing," in *Proc. European Design & Test Conf.*, 1994, pp. 171–176.

[16] I. Hartanto, V. Boppana, and W. K. Fuchs, "Diagnostic Fault Equivalence Identification Using Redundancy Information & Structural Analysis," in *Proc. International Test Conf.*, Oct. 1996, pp. 294–302.

[17] M. A. Iyer and M. Abramovici, "FIRE: A Fault-Independent Combinational Redundancy Identification Algorithm," *IEEE Trans. on VLSI Systems*, vol. 4, no. 2, pp. 295–301, June 1996.

[18] N. Jha and S. Gupta, *Testing of Digital Systems*. Cambridge University Press, 2003.

[19] T. P. Kelsey, K. K. Saluja, and S. Y. Lee, "An efficient Algorithm for Sequential Circuit Test Generation," *IEEE Trans. on Computers*, vol. 42, no. 11, pp. 1361–1371, Nov. 1993.

[20] A. Lioy, "Looking for Functional Fault Equivalence," in *Proc. International Test Conf.*, Oct. 1991, pp. 858–863.

[21] A. Lioy, "Advanced Fault Collapsing," *IEEE Design & Test of Computers*, vol. 9, no. 1, pp. 64–71, Mar. 1992.

[22] A. Lioy, "On the Equivalence of Fanout-Point Faults," *IEEE Trans. on Computers*, vol. 42, no. 3, pp. 268–271, Mar. 1993.

[23] E. J. McCluskey and F. W. Clegg, "Fault Equivalence in Combinational Logic Networks," *IEEE Trans. on Computers*, vol. C-20, no. 11, pp. 1286–1293, Nov. 1971.

[24] V. J. Mehta, K. K. Dave, V. D. Agrawal, and M. L. Bushnell, "A Fault-Independent Transitive Closure Algorithm for Redundancy Identification," in *Proc.* $16^{th}$ *International Conf. VLSI Design*, Jan. 2003, pp. 149–154.

[25] M. Nadjarbashi, Z. Navabi, and M. R. Movahedin, "Line Oriented Structural Equivalence Fault Collapsing," in *IEEE Workshop on Model and Test*, 2000.

[26] T. M. Niermann and J. H. Patel, "HITEC: A Test Generation Package for Sequential Circuits," in *Proc. European Design Automation Conference*, Feb. 1991, pp. 214–218.

[27] J. F. Poage and E. J. McCluskey, "Derivation of Optimum Test Sequences for Sequential Machines," in *Proc.* $5^{th}$ *Annual Symposium on Switching Theory and Logic Design*, Oct. 1964, pp. 121–132.

[28] I. Pomeranz, L. N. Reddy, and S. M. Reddy, "COMPACTEST: A Method to Generate Compact Test Sets for Combinational Circuits," *IEEE Trans. CAD*, vol. 12, no. 7, pp. 1040–1049, July 1993.

[29] A. V. S. S. Prasad, V. D. Agrawal, and M. V. Atre, "A New Algorithm for Global Fault Collapsing into Equivalence and Dominance Sets," in *Proc. International Test Conf.*, Oct. 2002, pp. 391–397.

[30] D. R. Schertz and G. Metze, "A New Representation for faults in Combinational Digital Circuits," *IEEE Trans. On Computers*, vol. C-21, no. 8, pp. 858–866, Aug. 1972.

[31] K. To, "Fault Folding for Irredundant and Redundant Combinational Circuits," *IEEE Trans. on Computers*, vol. C-22, no. 11, pp. 1008–1015, Nov. 1973.

[32] A. Veneris, R. Chang, M. S. Abadir, and M. Amiri, "Fault Equivalence and Diagnostic Test Generation Using ATPG," in *Proc. IEEE International Symposium on Circuits and Systems*, May 2004, pp. 221–224.