Fault Dictionary Compaction by Output Sequence Removal

Vamsi Boppana and W. Kent Fuchs
Coordinated Science Laboratory
University of Illinois at Urbana-Champaign
Urbana, IL 61801

Abstract

Fault dictionary compaction has been accomplished in the past by removing responses on individual output pins for specific test vectors. In contrast to the previous work, we present techniques for eliminating entire sequences of outputs and for efficiently storing the remaining output sequences. Experimental results on the ISCAS 85 and ISCAS 89 benchmark circuits show that the sizes of dictionaries proposed are substantially smaller than the full fault dictionary, while the dictionaries retain most or all of the diagnostic capability of the full fault dictionary.

1 Introduction

The most significant problem with the fault dictionary approach to failure location is that the the size of the full fault dictionary may be large and impractical for large circuits [1], [2],[3]. Several methods of compaction including the Pass/Fail dictionary [4], Compact [4] dictionary and the Sequential [5] dictionary have been developed. However, these schemes consider only specific output pins for compaction. Other alternative approaches to dictionary compression include Drop-on-K, First Failing Patterns and Detection Frequencies [5]. These provide considerable savings, but suffer from resolution loss. In contrast to the above techniques, this paper describes methods to eliminate *sequences* of output responses after a traversal of the diagnostic experiment tree for the identification of such output sequences.

2 Output Sequence Removal

There are three types of output sequences that we can potentially remove from the fault dictionary.

Type 1: If a node i at level p of the diagnostic tree has only one fault in the list of faults, then we need not continue the diagnostic experiment for combinational circuits as we have already found the required fault. Hence, it is unnecessary to store the output sequence produced by the fault in the list at node i for all vectors $p \ldots (l-1)$. It has been noted in our experiments that such sequences can be dropped for sequential circuits also, even if the node i is not fully distinguished, without any significant loss in the resolution. In our implementation for sequential circuits, we handle partially specified output responses by treating X as a separate alphabet and having pointers whenever necessary from other nodes in the tree.

Type 2: For a node i at level p the full fault dictionary stores the output responses produced by each of the faults in the list of faults at i for all vectors $0 \ldots p-1$. Each of these output responses is the same, otherwise the faults would not be members of the same list at node i. Hence, the same output sequence is replicated in the full fault dictionary. We show in Section 3, that this information can be stored without replication.

Type 3: If the application of the test sequence $p_{n_1} \ldots p_{n_k}$ to a node i at level p, results in a single node j, then all the faults in the lists at both nodes i and j are the same for combinational circuits. Hence, this test sequence is not useful for distinguishing any pair of faults in the list at node i. This implies that during diagnosis, if we are at node i, then the observation of output responses produced by each of the tests in this sequence is unnecessary. This can be utilized to reduce the size of the dictionary as well as to reduce the number of test responses needed to be observed and will result in a reduction in the number of tests to be applied for diagnosis if the circuit is combinational. It has also been observed during out experiments that such sequences can be dropped for sequential circuits also, without any significant loss of resolution.

2.1 Example

The following example illustrates the three types of removable output sequences identified above. Figure

This research was supported in part by the Semiconductor Research Corporation(SRC) under grant 93-DP-109, and in part by the Joint Services Electronics Program (JSEP) under grant N0014-90-J-1270.

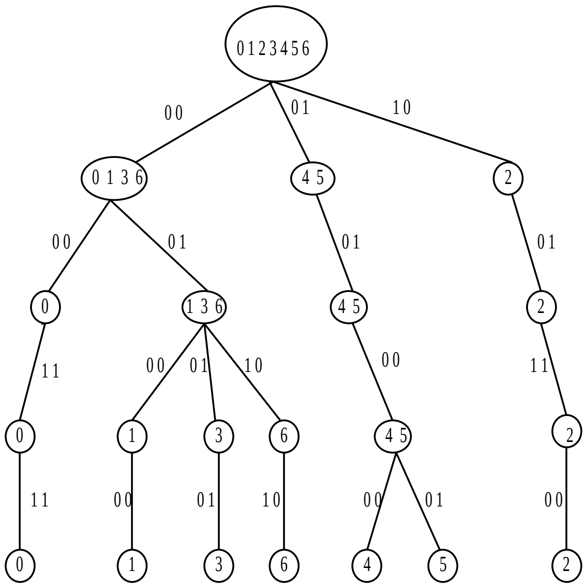

Figure 1: Diagnostic experiment tree.

FVector

	CSS
0	4
1	6
2	3
3	7
4	9
5	10
6	8

CSSTable

	ICSS	# ops	op responses
1	0	1	00
2	0	3	01 01 00
3	0	1	10
4	1	1	00
5	1	1	01
6	5	1	00
7	5	1	01
8	5	1	10
9	2	1	00
10	2	1	01

Figure 2: $DC1$ for the tree in Figure 1.

1 shows the diagnostic experiment tree for a fault dictionary with 7 faults, 4 vectors and 2 output bits.

Type 1: In Figure 1, fault 2 has been distinguished from all other faults after the application of test 0. Hence , the output responses for fault 2 for test vectors 1,2 and 3 can be eliminated.

Type 2: In Figure 1, the output sequence 00 01 can be stored only once for all the faults 1, 3 and 6.

Type 3: In Figure 1, tests 1 and 2 do not have any effect on the node (4,5) and hence do not provide useful output information for distinguishing between faults 4 and 5.

3 Small Dictionaries

Common Subsequence CSS(i): *The common subsequence of a node i at level p in the diagnostic experiment tree is defined as the sequence of output responses due to vectors $0 \ldots (p-1)$, say S,* iff *there is no subsequence S_1 of S that is of type 1.* In Figure 1, we observe that the CSS of the node with the list of faults (1 3 6) is 00 01. Every Common Subsequence inherits another common subsequence called the Inherited Common Subsequence defined below:

Inherited CSS ICSS(i): *Let the node whose CSS number is i be n. Then, the inherited common subsequence of the common subsequence i is defined as the sequence of output responses due to vectors $0 \ldots j$, where $j+1$ is the least level where a node m can be found with a fault list exactly identical to the list at node n.* In Figure 1, the CSS 01 01 00 of the node with the fault list (4 5) inherits the ICSS 01, because j as defined above equals 0. We now introduce two dictionaries $DC1$ and $DC2$, with $DC1$ eliminating sequences of types 1 and 2 and $DC2$ eliminating sequences of types 1 , 2 and 3. $DC2$ is smaller than $DC1$ in most but not all cases.

3.1 Dictionary $DC1$

This dictionary is comprised of a vector and a table. The vector is called the $FVector$ and the table is the $CSSTable$. Each of these can be represented in the compact bit packed representation below. **FVector:** This vector has F entries, where F is the total number of collapsed single stuck-at faults in the circuit. The entry $FVector[i]$ in this vector gives the common subsequence associated with the fault i when it is at a leaf of the diagnostic experiment tree. For example in $DC1$ constructed for the diagnostic experiment of Figure 1 shown in Figure 2, we have CSS number 7 stored for fault 3. **CSSTable:** Each entry in this table is indexed by the common subsequence number. Each entry in this table has three fields. The first field in the entry $CSSTable[i]$ has the inherited subsequence number ICCS(i). The second field has the number of vectors whose output response sequence when concatenated with the output sequence obtained from the the inherited common subsequence gives the common subsequence i. The third field is a bit stream comprised of the actual output responses that when concatenated with the output sequence obtained from the inherited

FVector

	CSS
0	4
1	6
2	3
3	7
4	9
5	10
6	8

CSSTable

	ICSS	opseq
1	0	00
2	0	01
3	0	10
4	1	00
5	1	01
6	5	00
7	5	01
8	5	10
9	2	00
10	2	01

DTable

	0	1	2	3
0	1	1	0	1
1	1	1	1	0
2	1	0	0	0
3	1	1	1	0
4	1	0	0	1
5	1	0	0	1
6	1	1	1	0

Figure 3: $DC2$ for the tree in Figure 1.

common subsequence gives the common subsequence i. This is a variable length field in the dictionary, but the storage can still be in a bit packed manner, because we know exactly how many bits are present in this field from the second field in the same entry.

As an example, in Figure 2, the entry for CSS 4 has ICSS=1, the number of vectors whose output responses are stored in this entry is 1 and the output response is 00. Hence, to construct the CSS 4, we go to the entry for CSS 1, given by the inherited CSS field, and find that the inherited CSS is 0 (which means that there is no inherited CSS). Hence the output sequence of CSS 1 is 00, which implies that the output sequence of CSS 4 is 00 00.

3.2 Dictionary $DC2$

This dictionary is comprised of a vector and two tables. The vector is called the $FVector$ and the two tables are called $DTable$ and the $CSSTable$. Again, each of these can be represented in a compact bit packed representation. **FVector:** This vector is *identical* to the $FVector$ constructed in $DC1$. **DTable:** The rows of the $DTable$ are indexed by fault numbers and the columns are indexed by test numbers. One bit is maintained for each fault, test vector pair. The entry $DTable[i][j]$ is 0, iff the fault lists that include the fault i before and after the application of the test j are the same. It is through the use of this table that we can decide at diagnosis time, whether the application of a test vector is essential. In Figure 3, the entries $DTable[4][1] = 0$ and $DTable[5][1] = 0$, imply that the fault list containing 4,5 (i.e., (4 5)) does not change after the application of test vector 1. **CSSTable:** There are only two fields for each entry $CSSTable[i]$. The first and third fields of $DC1$ are present, but the second field is absent. This is because a single output response is stored in the third field instead of the multiple output responses, since information on useful tests is already available from $DTable$. This is how the dictionary $DC2$ eliminates sequences of Type 3.

As an example, the entry for CSS 2 in $DC2$ has just 01 in contrast to the 01 01 00 stored by $DC1$. But, we see that the only two CSS's that inherit CSS 2 are 9 and 10 and from $FVector$ we see that they correspond to faults 4 and 5. Now, from $DTable[4][1] = 0, DTable[5][1] = 0, DTable[4][2] = 0$ *and* $DTable[5][2] = 0$, we can conclude that the sequence 01 00 is not needed for the diagnostic experiment as are the tests 1 and 2 for distinguishing between faults in the list (4 5).

3.3 Diagnostic Capabilities

In both of the dictionaries introduced above, the information eliminated is redundant if the circuit is combinational. For sequential circuits, $DC2$ has been constructed by dropping sequences, even if the node under consideration in the diagnostic tree has not been fully distinguished. It has been noted in our experiments that this does not cause any significant change in the resolution of the dictionary. Hence, $DC1$ and $DC2$ have the same diagnostic capability of the full fault dictionary for combinational circuits, whereas $DC1$ alone retains this capability for sequential circuits. It should be noted that the output responses eliminated may be important for the identification of some non-modeled faults.

3.4 Generation of $DC1$ and $DC2$

$DC1$ and $DC2$ have been generated through a traversal of the diagnostic experiment tree. A single full fault simulation without fault dropping is performed to dynamically construct the diagnostic experiment tree. The process of constructing the tree and the dictionaries can be done dynamically, one level at a time. This removes the need for excessive storage when generating the dictionaries.

4 Experimental Results

Experiments were performed on the ISCAS 85 and ISCAS 89 benchmark circuits to study the sizes and performance of the dictionaries $DC1$ and $DC2$. We

Table 1: Experimental results on ISCAS 85 circuits.

Cir.	FF (Kb)	PF/ FF%	DC1/ FF%	DC2/ FF%	UT RAT. %
c432	201	14.3	19.4	23.6	26.2
c499	1334	3.1	9.5	7.7	40.3
c880	1836	3.9	11.9	7.3	20.5
c1355	4432	3.1	27.9	5.1	3.6
c1908	13132	4.0	14.3	4.9	3.8
c2670	85126	0.7	5.3	1.1	10.9
c3540	21739	4.5	9.2	5.5	5.6
c5315	173725	0.8	9.2	1.5	6.2
c6288	11399	3.1	18.9	8.9	20.9
c7552	366930	0.9	11.9	1.3	2.8

Table 2: Experimental results on ISCAS 89 circuits.

Cir.	FF (Mb)	PF/ FF%	DC1/ FF%	DC2/ FF%	Res FF%	Res DC%
s298	0.9	16.7	55.0	9.2	94.522	94.520
s344	0.8	9.1	66.1	6.5	96.771	96.764
s641	4.7	4.2	62.4	2.9	97.764	97.640
s713	4.7	4.4	51.2	2.9	96.275	96.081
s820	31.2	5.2	65.1	2.8	99.571	99.571
s832	31.9	5.2	64.9	2.8	99.388	99.388
s1238	18.1	7.1	59.5	4.1	99.710	99.710
s1423	1.3	20.0	11.4	11.5	59.440	59.140
s5378	405.9	2.0	46.3	1.2	89.993	89.709
s35932	9532.6	0.3	41.4	0.4	98.025	98.025

also give a measure for estimating the amount of reduction in the number of test responses that need to be observed during diagnosis for combinational circuits. This measure also gives an estimate on the average number of tests that need to be applied for diagnosis. The *useful test ratio* is defined as : $useful\ test\ ratio = 1-(no\ of$ type-3 $seq. - no\ of$ type-1 $seq)/(no\ of\ seq - no\ of$ type-1 $seq)$. For the example of Figure 1, we have the *useful test ratio* = 10/12.

Table 1 presents the results for ISCAS 85 circuits whereas the results for ISCAS 89 circuits are given in Table 2. In Tables 1 and 2, the first column (FF) gives the size of the full fault dictionary. The test set used in the creation of the dictionaries was from the HITEC test generator. Experiments were performed only on circuits whose test sets have reasonable diagnostic resolution. The column PF/FF gives the percentage ratio of the size of the Pass/Fail dictionary to the full fault dictionary. The next two columns give the percentage ratios of the sizes of $DC1$ and $DC2$ with respect to the the full fault dictionary. The next column in Table 1 gives the percentage *useful test ratio*, whereas the next two columns in the Table 2, give the diagnostic resolutions of the full dictionary and $DC2$. Although not shown, the diagnostic expectations [5] of $DC2$ and the full fault dictionary are very similar except for cases with low diagnostic resolution.

From Tables 1 and 2, we can make several observations. The sizes of the dictionaries $DC1$ and $DC2$ are significantly smaller than the full fault dictionary. The size of $DC2$ is almost always smaller than that of $DC1$. The sizes of the dictionaries presented are less than even the Pass/Fail dictionary size for many cases. This is significant because the sizes of both the Compact [4] and the Sequential [5] dictionaries are lower bounded by the size of the Pass/Fail dictionary. However, both Compact and Sequential achieve the resolution of the full fault dictionary as against the small loss suffered by $DC2$.

5 Conclusions

We have identified three kinds of sequences that can be eliminated from fault dictionaries. Two dictionary schemes that eliminate these redundancies were presented and experimental results on the ISCAS bench mark circuits show that these dictionaries give a substantial reduction in the size of the dictionary with little or no loss of resolution. It has also been shown that the number of tests whose outputs need to be observed is far less than the full test set size for a large number of diagnosis experiments as indicated by the low *useful test ratios*. These ratios also indicate that the number of tests that need to be applied for diagnosing combinational circuits may be very small.

References

[1] R. E. Tulloss, "Size Optimization of Fault Dictionaries," *Proc. SemiConductor Test Conf.*, 1978, pp.264-265.

[2] P. G. Ryan, S. Rawat and W. K. Fuchs, "Two-Stage Fault Location," *Proc. Intl. Test Conf.*, Oct. 1991, pp. 963-968.

[3] R. E. Tulloss, "Fault Dictionary Compression: Recognizing when a Fault may be Unambiguously Represented by a Single Failure Detection," *Proc. Intl. Test Conf.*, Nov. 1980, pp. 368-370.

[4] I. Pomeranz and S. M. Reddy, "On the Generation of Small Dictionaries for Fault Location," *Proc. Intl. Conf. on Computer-Aided Design*, Nov. 1992, pp. 272-279.

[5] P. G. Ryan, W. K. Fuchs and I. Pomeranz, "Fault Dictionary Compression and Equivalence Class Computation for Sequential Circuits," *Proc. Intl. Conf. on Computer-Aided Design*, Nov. 1993, pp. 508-511.