# **On the Characterization of Hard-to-Detect Bridging Faults**

Irith Pomeranz<sup>1</sup>, School of ECE, Purdue University, W. Lafayette, IN 47907 Sudhakar M. Reddy<sup>2</sup>, ECE Dept., University of Iowa, Iowa City, IA 52242 Sandip Kundu, Intel Corp., Austin, TX 78746

# Abstract

We investigate a characterization of hard-to-detect bridging faults. For circuits with large numbers of lines (or nodes), this characterization can be used to select target faults for test generation when it is impractical to target all the bridging faults (or all the realistic bridging faults). We demonstrate that the faults selected based on the proposed characterization are indeed hard-to-detect by showing that the fault coverage of a given test set with respect to this subset is lower and more sensitive to the test set than the fault coverage obtained with respect to a random subset of the same size, with respect to the complete set of faults, and when possible, with respect to a subset of realistic bridging faults of the same size. We also demonstrate that a test set for the selected subset of faults detects other faults more effectively than when a test set is derived for a randomly selected subset of faults of the same size.

## 1. Introduction

The number of bridging faults in a circuit with L lines (or nodes) is on the order of  $L^2$ . Consequently, for large circuits, only a subset of the bridging faults can be considered in practice during test generation. In order to limit the number of bridging faults targeted for test generation, a set of bridging faults called *realistic bridging faults* is selected in [1]-[3] based on layout information. This selection procedure identifies faults that are most likely to occur taking into account that physical defects are more likely to bridge two lines if the lines are closer to each other. In large industrial designs, such methods typically generate bridging fault lists that are an order of magnitude larger than the number of nodes in the circuit. For this reason, typically only a portion of these faults that have the highest probability of occurrence are actually targeted for test generation. However, it has been observed that very high percentages of these highest-probability faults are also easy-to-detect. Consequently, the effectiveness of the tests generated for these targeted faults is low as they leave many untargeted faults undetected.

A fundamentally different approach to the selection of a subset of bridging faults is to focus on hard-to-detect faults. As with other fault models, hardto-detect faults are faults with relatively few tests, and therefore, they are not likely to be detected by tests for other faults. Moreover, tests for hard-to-detect faults are likely to detect a large number of easier-to-detect faults. In this work, we consider a way to characterize hard-todetect bridging faults. The proposed characterization provides a ranking of the faults according to their difficulty of detection. Based on this ranking, it is possible to select a subset of faults of the desired size as targets for test generation. When layout information is available, the proposed ranking can be used for selecting a subset of the realistic bridging faults that are the hardest-to-detect. When layout information is not available and it is impractical to consider all the bridging faults in the circuit, the proposed ranking provides a way to select a set of target faults that maximizes the detection of other faults, which are not included in the selected subset.

The proposed characterization is done based on information available from structural analysis of the circuit and from logic simulation of a (small) subset of vectors. Thus, its computational complexity is low. We describe the characterization of bridging faults according to their difficulty of detection in Section 2. We then report the results of three experiments to demonstrate the effectiveness of this characterization.

In Section 3, we perform bridging fault simulation of deterministic test sets for single stuck-at faults and of random test sets in benchmark circuits. We compare the fault coverages with respect to the following subsets of faults. (1) The complete set of bridging faults. (2) A subset of bridging faults consisting of approximately 1% of the faults that are the hardest-to-detect according to the proposed characterization. (3) A randomly selected subset of faults of the same size as that selected using the proposed characterization. (4) A subset of realistic bridging faults of the same size when they are available. The comparison shows that the fault coverage with respect to the "hard-to-detect" faults is significantly lower than the other fault coverages and more sensitive to the test set. Thus, the selected faults are indeed hard-to-detect.

<sup>1.</sup> Research supported in part by NSF Grant No. CCR-0098091 and in part by SRC Grant No. 2001-TJ-950.

<sup>2.</sup> Research supported in part by NSF Grant No. CCR-0097905 and in part by SRC Grant No. 2001-TJ-949.

In Section 4, we consider circuits with small numbers of inputs, for which tests can be selected out of the complete set of input vectors without the bias of specific test generation or test compaction heuristics. We use a (greedy) covering procedure to derive compact test sets for subsets of bridging faults of increasing sizes. We use subsets of hard-to-detect faults as measured by the proposed characterization, as well as random subsets of the same size for comparison. We compare test sets derived for fault subsets of the same size by considering the fault coverage obtained with respect to the complete set of bridging faults. The results of this experiment show that complete coverage of all the detectable bridging faults is obtained for smaller subset sizes when subsets of hard-to-detect faults are used. This experiment shows that tests for hard-to-detect faults based on the proposed characterization are more effective at detecting other, untargeted faults. We demonstrate the same point for larger circuits in Section 5. Here, we use a specific test compaction procedure that has fault selection embedded in it. We replace the fault selection process by the selection of hard-to-detect faults, and compare the resulting fault coverages considering the complete set of bridging faults.

We consider AND-type and OR-type bridging faults [4] as well as four-way bridging faults [5], [6]. The four-way bridging fault model addresses at the gate-level the fact that the effects of a bridging fault depend on the relative strengths of the driving nodes and on the threshold voltages of the driven nodes [7]. We experiment with AND-type and OR-type bridging faults in Sections 3 and 4, and with four-way bridging faults in Section 5.

We use multi-level implementations of Berkeley PLAs for experiments with circuits that have small numbers of inputs, and the combinational logic of ISCAS-89 benchmark circuits for experiments with circuits that have larger numbers of inputs.

## 2. Hard-to-detect bridging faults

In this section, we first describe the proposed characterization for AND-type and OR-type bridging faults. We then apply it to four-way bridging faults.

## 2.1. AND-type and OR-type bridging faults

An AND-type bridging fault between lines  $g_1$  and  $g_2$  is represented as  $(g_1,g_2,AND)$ . An OR-type bridging fault between lines  $g_1$  and  $g_2$  is represented as  $(g_1,g_2,OR)$ .

The proposed characterization is based on two sets of values associated with every (single) line in the circuit. The first set of values is obtained by logic simulation of a small set of input vectors, denoted by  $V_{char}$ . The size of  $V_{char}$  is denoted by  $N_{char}$ . During the logic simulation process of  $V_{char}$ , we collect information about the values assigned to every line in the circuit. We denote by v[g] the value of line g under vector  $v \in V_{char}$ . These values are used as follows. For a test t to detect a bridging fault associated with a pair of lines  $g_1,g_2$ , it is necessary for t to assign  $g_1 = 0$  and  $g_2 = 1$ , or  $g_1 = 1$  and  $g_2 = 0$ . Using the values v[g] collected during logic simulation of  $V_{char}$ , we can associate with the pair  $g_1,g_2$  the number of vectors  $v \in V_{char}$  such that  $v[g_1] = 0$  and  $v[g_2] = 1$ , or  $v[g_1] = 1$ and  $v[g_2] = 0$ . The lower this number, the more difficult it is to set  $g_1$  and  $g_2$  to different values, and the faults associated with  $g_1,g_2$  are likely to be harder-to-detect.

To make the characterization more accurate, we compute for every line g a set of *necessary assignments* A(g). A necessary assignment of g is a value that must be assigned to a line in the circuit in order to propagate a change in the value of g toward the primary outputs. The set A(g) is similar to the set of necessary assignments for a stuck-at fault on line g, except that we do not associate a faulty value with line g. To define the necessary assignments for g, we use structural analysis of the circuit. We consider the path from g to the closest primary output or fanout stem (whichever comes first). Every off-path input along this path must have the non-controlling value for a change in the value of g to propagate.

For illustration, we show in Figure 1 the combinational logic of ISCAS-89 benchmark circuit *s* 27. To propagate a change in the value of line 4, line 16 must be set to 0, line 18 must be set to 1, and line 5 must be set to 0. We obtain  $A(4) = \{(5,0), (16,0), (18,1)\}$ , where in every pair  $(h,\alpha) \in A(g), h$  is a line and  $\alpha$  is the value that must be assigned to *h* in order to propagate a change in the value of *g*.



### Figure 1: The combinational logic of s 27

Using the method described above, we obtain a relatively small number of necessary assignments for every line by using a simple structural analysis of the circuit. More sophisticated implication-based techniques can be used to find necessary assignments. Dominator gates can also be used for this purpose. However, these methods would be more time-consuming than the method we use here, and would result in larger numbers of necessary assignments that would have larger memory requirements. It is interesting to note that for fanout stems, our structural analysis does not yield necessary assignments. We collect a set of necessary assignments A(g) for every line g before we simulate the set of vectors  $V_{char}$ . During the logic simulation process of  $V_{char}$ , we update a variable sat(v,g) according to whether or not  $v \in V_{char}$ satisfies the necessary assignments A(g). We set sat(v,g) = 1 if the necessary assignments of g are satisfied by v, and we set sat(v,g) = 0 otherwise. The variables sat(v,g) are used as follows. For a test t to detect a bridging fault associated with  $g_{1,g_{2}}$ , it is necessary for t to assign  $g_{1}=0$  and  $g_{2}=1$ , or  $g_{1}=1$  and  $g_{2}=0$ . At the same time, t must satisfy the necessary assignments for propagating a change in the value either of  $g_{1}$  or of  $g_{2}$ .

Using  $\{v(g)\}$  and  $\{sat(v,g)\}$ , we define a number  $\delta(g_1,g_2,AND)$  for an AND-type bridging fault and a number  $\delta(g_1,g_2,OR)$  for an OR-type bridging fault to capture the difficulty of detecting the fault, as follows.

The number  $\delta(g_1,g_2,AND)$  is equal to the number of vectors  $v \in V_{char}$  such that (1)  $v[g_1] = 0$ ,  $v[g_2] = 1$ and  $sat(v,g_2) = 1$  (with these values, an AND-type bridging fault will change the value of  $g_2$  from 1 to 0, and a necessary condition for the change in the value of  $g_2$  to propagate is  $sat(v,g_2) = 1$ ); or (2)  $v[g_1] = 1$ ,  $v[g_2] = 0$ and  $sat(v,g_1) = 1$  (with these values, an AND-type bridging fault will change the value of  $g_1$  from 1 to 0, and a necessary condition for the change in the value of  $g_1$  to propagate is  $sat(v,g_1) = 1$ ).

The number  $\delta(g_1, g_2, OR)$  is equal to the number of vectors  $v \in V_{char}$  such that (1)  $v[g_1] = 0$ ,  $v[g_2] = 1$  and  $sat(v, g_1) = 1$ ; or (2)  $v[g_1] = 1$ ,  $v[g_2] = 0$  and  $sat(v, g_2) = 1$ .

We use  $\delta(g_1,g_2,AND)$  and  $\delta(g_1,g_2,OR)$  to associate a difficulty of detection with every fault. The lower the value, the harder-to-detect the fault is expected to be. Thus, we can rank the faults according to their difficulty of detection using  $\delta(g_1,g_2,AND)$  and  $\delta(g_1,g_2,OR)$ . To select *N* hard-to-detect faults, we select the *N* faults with the lowest values of  $\delta$ .

## 2.2. Four-way bridging faults

Under the four-way bridging fault model [5], [6], a pair of lines  $g_{1}$ , $g_{2}$  is associated with four faults corresponding to two possible combinations of values (0,1 and 1,0) on  $g_{1}$ , $g_{2}$ , and two possibilities for selecting a line whose value may change in the presence of a fault. Thus, the first fault is activated when  $g_{1}=0$ ,  $g_{2}=1$ , and it causes the value of  $g_{1}$  to change from 0 to 1. The second fault is activated when  $g_{1}=0$ ,  $g_{2}=1$ , and it causes the value of  $g_{1}=0$ ,  $g_{2}=1$ , and it causes the value of  $g_{1}$  to change from 1 to 0. The third fault is activated when  $g_{1}=1$ ,  $g_{2}=0$ , and it causes the value of  $g_{2}$  to change from 1 to 0. The fourth fault is activated when  $g_{1}=1$ ,  $g_{2}=0$ , and it causes the value of  $g_{2}$  to change from 1 to 0. The fourth fault is activated when  $g_{1}=1$ ,  $g_{2}=0$ , and it causes the value of  $g_{2}$  to change from 1 to 0. The fourth fault is activated when  $g_{1}=1$ ,  $g_{2}=0$ , and it causes the value of  $g_{2}$  to change from 1 to 0.

To use the characterization of hard-to-detect faults based on  $\delta$  for four-way bridging faults, we extend the definition of  $\delta$  as follows. We denote by  $(g_1, \alpha_1, g_2, \alpha_2)$  a bridging fault between lines  $g_1$  and  $g_2$ , which is activated on  $g_1$  when  $g_1 = \alpha_1$  and  $g_2 = \alpha_2$ . Detection of this fault requires setting  $g_1 = \alpha_1$  and  $g_2 = \alpha_2$ , and propagating the effects of a change in the value of  $g_1$  to an output. The number  $\delta(g_1, \alpha_1, g_2, \alpha_2)$  is thus equal to the number of vectors  $v \in V_{char}$  such that  $v[g_1] = \alpha_1$ ,  $v[g_2] = \alpha_2$  and  $sat(v, g_1) = 1$ . Again,  $\delta$  allows a ranking of the faults according to their difficulty of detection. A lower value of  $\delta(g_1, \alpha_1, g_2, \alpha_2)$  indicates that the fault  $(g_1, \alpha_1, g_2, \alpha_2)$  is harder-to-detect.

### **3. Results of fault simulation**

In this section, we report the results of an experiment where we fault simulate various test sets using different subsets of AND-type and OR-type bridging faults, one of them including hardest-to-detect faults according to the characterization defined in the previous section. The results indicate that these faults are indeed hard-to-detect.

We simulate the following subsets of faults.

(1) The set  $F_{all}$  of all the non-feedback AND-type and OR-type bridging faults between every pair of lines that are not fanout branches or inputs of the same gate.

(2) A subset  $F_{hard}$  consisting of approximately 1% of the faults in  $F_{all}$  that have the lowest values of  $\delta(g_1, g_2, AND)$  and  $\delta(g_1, g_2, OR)$ . The subset  $F_{hard}$  is defined as follows.

Initially,  $F_{hard} = \phi$ . For  $\delta = 0, 1, \dots, \delta_{max}$ , we add to  $F_{hard}$  every fault  $(g_1, g_2, AND) \in F_{all}$  with  $\delta(g_1, g_2, AND) = \delta$  and every fault  $(g_1, g_2, OR) \in F_{all}$  with  $\delta(g_1, g_2, OR) = \delta$ . The value of  $\delta_{max}$  is determined such that it is the first value of  $\delta$  for which the size of  $F_{hard}$  reaches or exceeds

1% of the number of faults in  $F_{all}$ . (3) A subset  $F_{rand}$  of the same size as  $F_{hard}$ , where the faults are selected randomly out of  $F_{all}$ .

(4) A subset  $F_{real}$  of the same size as  $F_{hard}$  that includes the faults identified as most likely to occur by a realistic bridging fault extraction tool [1]. We consider only nonfeedback bridging faults provided by the extraction tool in order to be consistent with the selection of  $F_{hard}$  and  $F_{rand}$ . We consider only several of the circuits, for which sets  $F_{real}$  are available to us.

We simulate two types of test sets, deterministic compact test sets for single stuck-at faults from [8], and random test sets of the same size as the deterministic test sets. We use  $N_{char} = 100$  random vectors for the set  $V_{char}$  to compute the detection difficulty measures  $\delta(g_{1},g_{2},AND)$  and  $\delta(g_{1},g_{2},OR)$ . The results are shown in Tables 1 and 2, as follows.

In Table 1, after the circuit name, we show the total number of faults in  $F_{all}$  and the number of faults in the selected subsets  $F_{hard}$ ,  $F_{rand}$  and  $F_{real}$ . We then show the

Table 1: Parameters of simulated faults and tests

	fau	lts	ave. δ				
circuit	total	sel	all	hard	rand	real	tsts
s208	10792	554	32.13	0.00	31.58		27
s298	16612	228	33.02	0.97	34.91	44.93	24
s344	32210	332	32.23	0.33	30.41	41.76	15
s382	29622	411	35.56	1.72	37.35	44.47	25
s400	30932	352	35.90	1.13	37.02	44.30	24
s420	45756	6716	32.19	0.00	32.25		43
s444	37526	514	35.02	1.05	37.41	43.11	24
s510	50190	574	30.89	0.00	31.01		54
s526	43330	667	30.94	0.00	33.45	41.50	50
s641	144752	1761	35.46	0.34	35.19		22
s820	90696	14566	15.26	0.00	15.29		94
s953	175568	12995	40.40	0.00	40.59		76
s1196	283014	20183	25.52	0.00	25.53		118
s1423	463760	7508	31.28	0.43	31.45		26
s1488	419444	39469	14.85	0.00	14.79		101

### **Table 2: Fault coverages**

	determ. f.c.					rando	m f.c.	
circuit	all	hard	rand	real	all	hard	rand	real
s208	98.09	82.85	96.57		83.71	10.11	83.57	
s298	98.85	62.28	99.56	98.68	97.51	34.65	98.68	94.30
s344	95.11	29.52	95.18	97.59	90.16	12.95	89.16	90.66
s382	99.35	82.73	99.76	99.51	97.88	44.77	98.30	96.11
s400	99.21	73.30	99.43	99.43	93.22	40.06	91.19	92.33
s420	97.61	92.45	97.23		71.90	3.80	71.75	
s444	99.11	75.88	99.42	99.81	97.99	58.37	98.05	97.47
s510	97.12	61.50	96.17		94.94	33.97	93.73	
s526	99.11	73.61	99.40	98.22	95.52	27.29	96.85	92.75
s641	99.21	81.89	99.15		95.97	17.55	96.02	
s820	96.05	83.78	96.08		81.01	44.86	80.65	
s953	99.21	96.85	99.18		89.13	47.13	89.21	
s1196	97.49	85.45	97.66		87.55	45.98	87.82	
s1423	98.80	68.66	98.67		92.78	11.40	92.83	
s1488	97.99	82.97	97.97		89.40	54.64	89.04	
average	98.15	75.58	98.10		90.58	32.50	90.46	······

average value of  $\delta$  computed over all the faults in  $F_{all}$ , over the faults in  $F_{hard}$ , over the faults in  $F_{rand}$ , and over the faults in  $F_{real}$ . The average value of  $\delta$  for a subset of faults  $F_s$  is computed as

```
\sum_{i=1}^{3} \{\delta(g_{1},g_{2},\alpha):(g_{1},g_{2},\alpha) \in F_{s}, \alpha \in \{AND, OR\}\}
```

```
|F_{\rm c}|
```

In the last column of Table 1 we show the number of tests simulated.

In Table 2, after the circuit name, we show the fault coverages obtained by simulating the deterministic single stuck-at test sets from [8] under the sets of faults  $F_{all}$ ,  $F_{hard}$ ,  $F_{rand}$  and  $F_{real}$ . We then show the fault coverages obtained by simulating random test sets of the same size under the sets of faults  $F_{all}$ ,  $F_{hard}$ ,  $F_{rand}$  and  $F_{real}$ .

From Table 1 it can be seen that the average value of  $\delta$  for faults in  $F_{hard}$  is significantly lower than the average value of  $\delta$  for faults in  $F_{all}$ ,  $F_{rand}$  or  $F_{real}$ . From Table 2 it can be seen that the fault coverages computed with respect to the subset of hard-to-detect faults  $F_{hard}$  are significantly lower than the fault coverages computed with respect to  $F_{all}$ ,  $F_{rand}$  or  $F_{real}$ . In addition, the fault coverage computed over the faults in  $F_{hard}$  is more sensitive to the test set being simulated. With random test sets, the fault coverage over  $F_{hard}$  is significantly lower than the fault coverage obtained by deterministic test sets. The difference in fault coverage between the two test sets is significantly lower when considering the randomly selected set  $F_{rand}$ , the realistic set  $F_{real}$  or the set of all the faults  $F_{all}$ . It is important to note that if fault sampling is done for the purpose of fault simulation,  $F_{rand}$  provides a better approximation of the actual fault coverage; however, when the goal is to select a subset of target faults for test generation, the faults in  $F_{hard}$  are more difficult to detect and thus should be targeted explicitly.

Information not provided by this experiment is the extent to which the subset of hard-to-detect faults includes undetectable faults. We address this issue in the following section.

## 4. Test selection using hard-to-detect faults

In this section, we describe the results of an experiment where we derive compact test sets for subsets of hard-to-detect faults of increasing sizes. For comparison, we also obtain compact test sets for randomly selected subsets of faults. The test sets are generated for fault subsets of the same sizes. We compare the fault coverage achieved by the two types of test sets with respect to the complete set of faults  $F_{all}$ . This experiment provides an indication of the relative ability of tests for hard-to-detect faults to detect other faults.

To remove as much as possible the effects of test generation heuristics on the comparison, we consider in this section circuits with small numbers of inputs. For a given subset of faults  $F_s$ , we derive a compact test set by using a greedy covering procedure that selects tests out of the set of all possible input vectors of the circuit. The greedy heuristic first selects a yet-undetected fault in  $F_s$  that has the smallest number of tests. Out of the tests for the selected fault, it selects the test that detects the largest number of yet-undetected faults in  $F_s$ .

The subsets of hard-to-detect faults are determined as follows. For  $\delta = 0, 1, \dots$ , if there is at least one fault with  $\delta(g_1, g_2, AND) = \delta$  or  $\delta(g_1, g_2, OR) = \delta$ , we define a subset of faults  $F_{\delta}$  that consists of every fault with  $\delta(g_1, g_2, AND) \le \delta$  or  $\delta(g_1, g_2, OR) \le \delta$ . We denote the resulting subsets of faults  $F_{\delta_1}, F_{\delta_2}, \dots, F_{\delta_n}$ .

We also define subsets of the same sizes consisting of randomly selected faults. We denote the random subsets by  $F_{r_1}, F_{r_2}, \dots, F_{r_m}$ , respectively. We have  $|F_r| = |F_{\delta}|$  for  $1 \le i \le m$ .

We are interested in the first subset  $F_{\delta_i}$  (or  $F_{r_i}$ ) for which the compact test set detects all the detectable bridging faults in the circuit (i.e., the test set achieves 100% fault efficiency with respect to the set  $F_{all}$  of all the bridging faults). However, we observed that, especially with random selection of target faults, the fault coverage may decrease when the size of the fault subset is increased. To remove the effects of fault coverage fluctuations, we captured the first subset size after which 100% fault efficiency was obtained for all the larger subsets considered. We denote this subset by  $F_{100\%}^{hard}$  for the proposed selection criterion, and by  $F_{100\%}^{rand}$  for randomly selected faults. We use  $N_{char} = 500$  random vectors for the characterization of hard-to-detect faults.

The results of this experiment are shown in Table 3. In Table 3, after the circuit name, we show the total number of faults and the number of undetectable faults. We then show the proportion of faults included in  $F_{100\%}^{hard}$ , and in  $F_{100\%}^{rand}$ . Under column *tests* we show the number of tests included in the test set selected for  $F_{100\%}^{hard}$ , and in the test set selected for  $F_{100\%}^{hard}$ .

Table 3: Achieving 100% fault efficiency

	faults		sele	cted	tests	
circuit	total	undet	hard	rand	hard	rand
Z9sym	30574	805	0.838	0.993	161	161
add6	8912	80	0.374	0.975	37	36
adr4	3882	68	0.722	0.998	16	17
alu1	1530	1	0.387	0.980	24	27
alu2	4992	11	0.551	0.986	32	31
alu3	7740	40	0.478	0.997	36	38
co14	3988	335	0.794	0.998	42	42
dk17	3632	6	0.825	0.990	22	22
dk27	710	2	0.803	0.934	9	9
dk48	3568	3	0.686	0.925	19	20
radd	2600	38	0.550	0.997	13	13
rd53	2692	28	0.178	0.990	24	24
z4	2218	29	0.425	0.992	14	15
average			0.585	0.981		

From Table 3 it can be seen that using the proposed characterization, all the selected as well as unselected detectable faults are detected, on the average, after only 0.585 of the faults are targeted. For random selection of faults, this happens after 0.981 of the faults are targeted. The test set sizes are approximately the same when 100% fault efficiency is reached for both fault subsets.

We also observed that for very small subset sizes, the proposed characterization leads to the selection of a relatively large proportion of undetectable faults, for which no tests are selected. As a result, the initial fault coverage for small subsets of faults may be small. This can be alleviated by replacing undetectable faults with faults having the lowest values of  $\delta$  that are not included in the selected subset. In this context, methods such as the one in [9] for identifying redundant single stuck-at faults without explicit test generation could be modified to identify undetectable bridging faults and exclude them from the subset of selected faults. For larger subset sizes, this effect becomes insignificant.

For the circuits considered in Table 3, information about undetectable faults and the values of  $\delta$  associated with them is given in Table 4. After the circuit name, we show the total number of faults and the number of undetectable faults. We then consider the first three values of  $\delta$ ,  $\delta = 0,1,2$ . For every value of  $\delta$ , we show the total number of faults with  $\delta(g_1,g_2,AND) = \delta$  or  $\delta(g_1,g_2,OR) = \delta$ , and the number of undetectable faults with this value of  $\delta$ . Table 4 shows that many undetectable faults have  $\delta = 0$ . However, not all the faults with  $\delta = 0$  are undetectable. Thus, it is important to include these faults in the set of faults targeted for test generation.

**Table 4: Undetectable faults** 

	all		$\delta = 0$		$\delta = 1$		$\delta = 2$	
circuit	flts	und	flts	und	flts	und	flts	und
Z9sym	30574	805	5780	747	3896	0	3423	2
add6	8912	80	60	60	0	0	1	0
adr4	3882	68	53	49	8	0	0	0
alu1	1530	1	0	0	1	0	2	0
alu2	4992	11	31	6	84	0	45	0
alu3	7740	40	67	24	88	0	123	1
co14	3988	335	1493	326	186	0	36	0
dk17	3632	6	236	3	352	0	182	0
dk27	710	2	24	2	45	0	23	0
dk48	3568	3	710	0	88	0	53	0
radd	2600	38	30	30	0	0	0	0
rd53	2692	28	24	23	10	0	12	0
z4	2218	29	20	20	0	0	1	0

# **5. Test compaction using hard-to-detect faults** In the experiment reported next, we use a test compaction procedure for bridging faults to further demonstrate the effectiveness of targeting hard-to-detect faults.

The test compaction procedure we use considers the four-way bridging fault model from [5], [6]. A subset of bridging faults was originally selected as part of this compaction procedure such that for every pair of lines  $g_i,g_j$  used for defining four faults, the difference between the indices |i-j| does not exceed a preselected constant. We denote the subset of bridging faults selected in this way by  $F_{ind}$  (since it is based on line indices). We set the maximum value of |i-j| such that approximately 10% of the circuit faults are included in  $F_{ind}$ .

Using the proposed characterization of hard-todetect faults, we then replace the subset  $F_{ind}$  with a subset of faults  $F_{hard}$  of the same size. We compare the results of the compaction procedure when applied to  $F_{ind}$  and when applied to  $F_{hard}$  by considering two parameters. (1) The average value of  $\delta$  for the selected faults. (2) The percentage of faults detected out of the set  $F_{all}$  of all the four-way bridging faults when the test set is generated for  $F_{ind}$  and for  $F_{hard}$ .

The test compaction procedure starts from a given deterministic test set for single stuck-at faults, and it adds a minimal number of tests to this test set in order to detect bridging faults. We use the stuck-at test set as the set  $V_{char}$  for identifying hard-to-detect faults.

We considered non-feedback four-way bridging faults between every pair of lines that are outputs of multi-input gates and are not inputs of the same gate. For the larger circuits (*s* 5378 and larger), considering the set of all the bridging faults  $F_{all}$  was not practical because of its size. We therefore restricted the set  $F_{all}$  to include bridging faults involving lines  $g_i, g_j$  such that  $|i-j| \le D$  for *D* that resulted in close to one million bridging faults. Both  $F_{ind}$  and  $F_{hard}$  are contained in  $F_{all}$ . The results are shown in Tables 5 and 6, as follows.

**Table 5: Results of test compaction** 

	faults		ir	nd	hard	
circuit	total	select	ave.δ	f.c.	ave.δ	f.c.
s298	10188	1116	3.84	84.95	0.00	53.76
s344	17596	1876	2.39	92.91	0.00	65.14
s382	17816	1812	3.92	92.33	0.00	64.24
s400	20480	2180	3.75	90.18	0.00	63.72
s510	59460	6336	8.83	80.18	0.00	18.34
s526	37744	4056	7.01	86.27	0.00	52.19
s641	16900	1724	3.06	94.32	0.00	88.11
s820	126440	12716	6.72	75.50	0.00	20.01
s1196	274104	27764	15.09	86.43	0.00	39.20
s1423	385292	38784	4.48	95.16	0.00	90.21
s1488	589220	59628	6.38	84.19	0.00	15.73
s5378*1	980964	98788	12.45	96.69	0.00	74.59
s9234*2	950968	95940	10.74	89.58	0.00	72.07
s13207*3	896856	93380	20.68	89.21	0.00	75.89

\*1 D=900 \*2 D=400 \*3 D=300

**Table 6: Fault coverages for test compaction** 

	f.c. w	rt all		f.c. wrt all		
circuit	ind	hard	circuit	ind	hard	
s298	86.32	88.71	s820	73.75	74.25	
s344	91.91	92.59	s1196	85.11	85.84	
s382	90.70	93.43	s1423	96.57	98.17	
s400	89.36	92.05	s1488	83.87	84.02	
s510	78.77	79.28	s5378*1	94.94	96.84	
s526	88.65	90.47	s9234*2	89.75	90.57	
s641	95.46	97.29	s13207*3	87.82	89.23	

In Table 5, after the circuit name, we show the total number of four-way bridging faults and the number of faults targeted by the compaction procedure (the number of faults in  $F_{ind}$  and  $F_{hard}$ ). For the set  $F_{ind}$ , we show the average value of  $\delta$  for the faults in  $F_{ind}$ , and the fault coverage with respect to  $F_{ind}$ . We also show the same information when test generation is performed for the set of faults  $F_{hard}$ . The values of D that result in fewer than one million bridging faults in the larger circuits are given below Table 5.

In Table 6, we show the fault coverage, with respect to  $F_{all}$ , of the test sets obtained when  $F_{ind}$  and when  $F_{hard}$  are targeted for test generation.

From Table 5 it can be seen that the compaction procedure detects significantly more faults out of  $F_{ind}$  than out of  $F_{hard}$ . This is partly due to the fact that the set of hard-to-detect faults includes undetectable faults. Nevertheless, Table 6 shows that when the set  $F_{all}$  of all the four-way bridging faults is simulated, the tests generated for  $F_{hard}$  detect larger percentages of the faults than tests generated for  $F_{ind}$ . Since most of the faults in  $F_{all}$  are easy-to-detect, the differences in fault coverage are

not large, but the fault coverages are consistently higher when using the proposed subset of hard-to-detect faults.

## 6. Concluding remarks

We proposed a characterization of hard-to-detect bridging faults based on the results of logic simulation of a small set of vectors, and on necessary assignments for propagation of changes of single line values in the circuit. For a fault characterized as hard-to-detect, only a small number of vectors assign different values to the fault sites and satisfy the necessary assignments for propagating a change from the fault site whose value will change in the presence of the fault. For circuits with large numbers of lines, this characterization can be used to select target faults for test generation when it is impractical to target all the faults (or all the realistic faults).

We demonstrated that the faults selected by the proposed characterization are indeed hard-to-detect by showing that the fault coverage with respect to this subset is lower than the fault coverage obtained with respect to an arbitrary subset of the same size, and with respect to the complete set of faults. We considered realistic bridging faults when possible. We simulated deterministic single stuck-at test sets as well as random test sets in order to demonstrate this point.

We also demonstrated that a test set for the selected faults detects other, unselected faults more effectively than when a test set is derived for a randomly selected subset of faults of the same size.

#### References

- F. J. Ferguson and J. P. Shen, "A CMOS Fault Extractor for Inductive Fault Analysis", IEEE Trans. on Computer-Aided Design, Nov. 1988, pp. 1181-1194.
- [2] S. T. Zachariah and S. Chakravarty, "A Scalable and Efficient Methodology to Extract Two Node Bridges from Large Industrial Circuits", in Proc. 2000 Intl. Test Conf., Oct. 2000, pp. 750-759.
- [3] Z. Stanojevic and D. M. H. Walker, "FedEx A Fast Bridging Fault Extractor", in Proc. 2001 Intl. Test Conf., Oct. 2001, pp. 696-703.
- [4] M. Abramovici, M. A. Breuer and A. D. Friedman, *Digital Systems Testing and Testable Design*, Computer Science Press, 1990.
- [5] S. Sengupta et. al., "Defect-Based Tests: A Key Enabler for Successful Migration to Structural Test", Intel Technology Journal, Q.1, 1999.
- [6] V. Krishnaswamy, A. B. Ma, P. Vishakantaiah, "A Study of Bridging Defect Probabilities on a Pentium (TM) 4 CPU", in Proc. Intl. Test Conf., 2001, pp. 688-695.
- [7] P. C. Maxwell and R. C. Aitken, "Biased Voting: A Method for Simulating CMOS Bridging Faults in the Presence of Variable Gate Logic Thresholds", in Proc. 1993 Intl. Test Conf., Oct. 1993, pp. 63-72.
- [8] S. Kajihara, I. Pomeranz, K. Kinoshita and S. M. Reddy, "Cost-Effective Generation of Minimal Test Sets for Stuck-at Faults in Combinational Logic Circuits", IEEE Trans. on Computer-Aided Design, Dec. 1995, pp. 1496-1504.
- [9] M. Iyer and M. Abramovici, "FIRE: A Fault-Independent Combinational Redundancy Identification Algorithm", IEEE Trans. on VLSI Systems, June 1996, pp. 295 -301.