A Statistical Fault Coverage Metric for Realistic Path Delay Faults

Wangqi Qiu*, Xiang Lu+, Jing Wang*, Zhuo Li+, D. M. H. Walker*, Weiping Shi+

*Dept. of Computer Science
Texas A&M University
College Station TX 77843-3112
Tel: (979) 862-4387
Fax: (979) 847-8578
Email: walker@cs.tamu.edu

+Dept. of Electrical Engineering
Texas A&M University
College Station TX 77843-3124
Tel: (979) 458-0093
Fax: (979) 845-2630
Email: wshi@ee.tamu.edu

Abstract

The path delay fault model is the most realistic model for delay faults. Testing all the paths in a circuit achieves 100% delay fault coverage according to traditional path delay fault coverage metrics. These metrics result in unrealistically low fault coverage if only a subset of paths is tested, and the real test quality is not reflected. For example, the traditional path delay fault coverage of any practical test for circuit c6288 is close to 0 because this circuit has an exponential number of paths. In this paper, a statistical and realistic path delay fault coverage metric is presented. Then the quality of several existing test sets (path selection methods) is evaluated in terms of local and global delay faults using this metric, in comparison with the transition fault and traditional path delay fault coverage metrics.

1. Introduction

Delay testing detects timing defects and ensures that the design meets the desired performance specifications. The three commonly used delay fault models are the transition fault model [1], the gate delay fault model [2], and the path delay fault model [3]. The transition fault model assumes that the delay fault affects only one gate in the circuit, and the extra delay caused by the fault is large enough to prevent the transition from reaching any primary output within the specification time. In other words, the transition fault can be detected on any sensitized path through the fault site. The transition fault coverage is measured as the percentage of faults which are detected by a test set [4]. The gate delay fault model is more general than the transition fault model because it considers the amount of extra delay due to a defect. Therefore, a gate delay fault may only be detected through a long path. The quality of a test set is defined as how close the minimum actually detected delay fault sizes are to the minimum possibly detectable fault sizes [5]. The main advantage of these two models is that the number of faults in the circuit is linear in the number of gates. Also, the stuck-at fault test generation procedure can be easily modified for transition fault test generation [6].

Under the path delay fault model a circuit is considered faulty if the delay of any of its paths exceeds the specification time. The path delay fault model is more realistic in modeling physical delay defects because the model can also detect small distributed delay defects caused by process variation, or the combination of local and distributed delay. However, a major limitation of this fault model is that the number of paths in the circuit (and therefore the number of path delay faults) can be exponential in the number of gates. For example, ISCAS85 benchmark circuit c6288, a 16-bit multiplier, has close to 10^{20} paths.

Many techniques have been used to reduce the number of paths that must be tested in the path delay fault model. The simplest idea is to test the paths with maximum delays in the circuit. These paths are also called the longest paths or critical paths. However, circuit optimization tends to compress the distribution of path delays in a circuit, so many paths are close to the maximum delay [7]. Because of manufacturing process variation, any of these paths can be the actual longest path. Therefore a group of longest paths must be selected for testing. In practice, the path selection criteria can be based on if the nominal path delay is more than a certain threshold, e.g. 80% of the maximum specified delay of the circuit. It is assumed that the delays of the selected paths bound the maximum circuit delay with high confidence. The path selection is much more realistic if the structural and spatial correlations between path delays are used [8], and then the number of paths that must be tested can be significantly reduced. However, delay faults are caused by both local and global process disturbances during IC manufacturing [9], or their combination. An example of a local disturbance is a particle that causes a resistive bridge between two nets, or a resistive contact. An example of a global disturbance is variation in transistor gate length across a chip. The delay faults caused by local disturbances are termed local delay faults [10] and those caused by global disturbances are termed global delay faults [8] or distributed path delay faults [11]. The path delay fault model assumes that delay faults are only caused by global delay faults. Thus if no path through a local delay fault site is selected for testing, the delay fault cannot be detected. To solve this problem, some path selection methods ensure that for every line in the circuit, the longest path through it must be selected [12][13][14][15].

Unfortunately, there is little research focusing on realistic delay fault coverage estimation. Therefore it is hard to evaluate the quality of different path selection methods. The traditional metric of path delay fault coverage is the percentage of the paths which are tested under robust [3][16], non-robust [16], or functional [17] sensitization criteria, that is, coverage = number of tested paths / total number of structural paths. A structural path is a sequence of gates and nets without considering sensitization. Based on this metric, testing p long paths has the same fault coverage as testing p short paths, which does not reflect the real test quality. In addition, since the total number of structural paths is exponential in the number of gates, clearly this fault coverage metric results in very low fault coverage for any test set, which is far from the reality. Some research eliminated untestable paths [18][19], and then the coverage = number of tested paths / number of total testable paths. But these methods are very expensive because the sensitization of all the paths must be checked and the coverage is still unrealistically low (around 20%). For example, as circuit c6288 may have an exponential number of testable paths [20], the traditional path delay fault coverage of any practical test is close to 0.

In this work, the traditional path delay fault model is extended to a more realistic delay fault model, which assumes that delay faults are caused by global delay faults only or the combination of local and global delay faults. Our model can be seen as a cross between the path and gate delay fault models. In addition, a new statistical delay fault coverage metric is proposed. Capacitive coupling is not considered here because it makes path delays pattern-sensitive, which is not assumed in the traditional path or gate delay fault models.

The remainder of the paper is organized as follows: Section 2 analyzes the path delay correlation. Section 3 describes the realistic delay fault coverage metric. Section 4 includes experimental results and compares the delay fault coverage for different test sets. Section 5 presents our conclusions with directions for future research.

2. Path Delay Correlation

Any two paths in the circuit have correlation in delays. Two paths have *structural correlation* when they share a common path segment. For example, in Figure 1 path *a-d-e* and *b-d-e* are structurally correlated because they share segment *d-e*. Two paths can also have *spatial correlation* because the path delays are functions of the manufacturing process parameters, such as transistor gate length, which are spatially correlated. For two paths which are physically close to each other, the delay correlation is high because the paths have very similar process parameters.

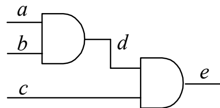

Figure 1. Example of structural correlated paths.

Figure 2 shows the delay space [11] for two paths, assuming the path delay is a one-dimensional function of process parameters. The delay space is the bounded region in which the probable delay value combinations are represented. It is assumed that each path has min-max delays. If the two paths have *no correlation*, the delay value combination can be anywhere within the rectangle. If they are *perfectly correlated*, the delay space shrinks to a line, which means if path 1 has the max (min) delay under a combination of certain process parameters, path 2 also reaches its max (min) delay under the same combination of process parameters. In reality, the correlation is somewhere in between, and the realistic delay space is the shaded area. Using correlation information, the delays on the untested paths can be predicted by the delays on the tested paths [21].

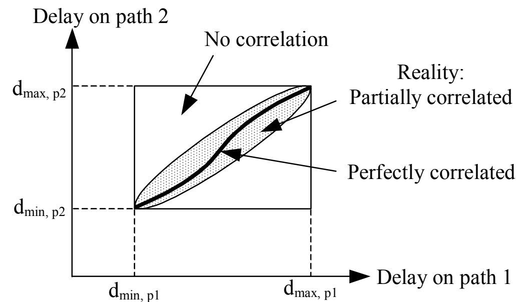

Figure 2. Delay spaces for different path correlations.

An inter-die process variation model [22] is used in this work. In this model, the delay of a path is expressed as a linear function of process variables. If the delay of path p_1 is less than that of path p_2 under any process parameter combination, it is said that p_1 is covered by p_2. Thus, if p_2 is tested, p_1 does not have to be tested. One limitation of this model is that it does not consider intra-die process variation. However, by assuming 100% intra-die process correlation, the upper bound of delay fault coverage can be computed, as shown in the next section.

3. Delay Fault Coverage Metric

A realistic delay fault coverage can be computed as the percentage of *faulty chips* that can be *detected as faulty* by a test set [11]. If presented in a probability formula, the coverage for test set *t* is:

P(*t* detects delay fault | chip has a delay fault) (1)

Because this is a general metric and model-independent, to make it usable, it is necessary to map this abstract metric to a realistic delay fault model, which considers both local and global delay faults.

In this work it is assumed that there is at most one local delay fault (or no local delay fault) that can occur on any single line in the circuit. We term the *fault site* the position of the local delay fault. The whole circuit is also subject to process variation, which may cause a timing failure by itself (global delay fault) or in combination with a local delay fault.

Under this model fault detection is probabilistic instead of deterministic. For example, suppose there are two paths, P_1 and P_2, through a resistive open fault site, and the local extra delay is not large enough for either path to be definitely slow. Figure 3 shows the delay space for this fault. t_{max} is the maximum specified delay of the circuit. The circuit has some probability that path 1 or 2 is slow (Delay < t_{min} is not considered in this work). Suppose test set t_1 tests path 1 only and test set t_2 tests path 2 only. Neither t_1 nor t_2 can guarantee the detection of the fault, e.g. t_1 cannot detect the delay fault in area *A*. Instead each test set only has some probability of detection. Both test sets are required to guarantee detection. In this research the notion of detection probability (DP) [23] for a single fault site is used.

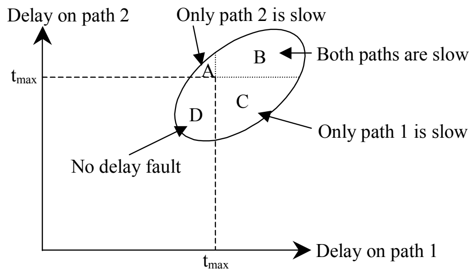

Figure 3. Delay space of a fault.

Using this probability model, the general metric expressed in formula (1) can be translated into formula (2) to compute the DP for fault site *i* with local extra delay Δ (the size of the local delay fault):

$DP_{i,\Delta}(t)$ =P(at least one tested path through *i* is slow | at least one path through *i* is slow) (2)

In the example whose delay space is shown in Figure 3, according to formula (2), if test set *t* tests path 1 only, the DP is area(B∪C)/area(A∪B∪C); if *t* tests path 2 only, the DP is area(B∪A)/area(A∪B∪C); and if *t* tests both paths, the DP is 100%.

The above analysis is for a given local extra delay Δ. For fault site *i* with an arbitrary Δ, the DP for site *i* is computed as:

$$\mathrm{DP}_i(\boldsymbol{t}) = \int_{\Delta > \Delta_{0,i}} \mathrm{DP}_{i,\Delta}(\boldsymbol{t}) \cdot \mathrm{p}_i(\Delta)\mathrm{d}\Delta \quad (3)$$

where $\Delta_{0,i}$ is the value of local extra delay below which there is no delay fault. $p_i(\Delta)$ is the PDF of Δ at fault site *i*, and is computed using the PDF of delay caused by physical defects, such as resistive opens [24] or shorts [25].

The overall fault coverage for test set *t* is:

$$\mathrm{FC}(\boldsymbol{t}) = \sum_i \mathrm{DP}_i(\boldsymbol{t}) \cdot \boldsymbol{w}_i \times 100\% \quad (4)$$

where w_i is the weight for fault site *i* ($\sum_i w_i = 1$). w_i depends on the location of the fault [24]. For example, the fault sites with many long paths through them are more likely to cause delay faults than the fault sites which have only short paths through them. Therefore, testing more paths through a high weighted fault site is an efficient way to increase the fault coverage. w_i is also sensitive to the ratio of local/global delay faults. If the ratio is high, the weights are almost equal for all fault sites. If it is low, the fault sites with only short paths passing them through can have weights close to 0. In this work equal weights are used for simplicity.

If no local delay fault is considered, only formula (2) is used in the computation, with Δ=0, and *i* is removed because the whole circuit, instead of a particular site, is considered.

According to formula (2), if the path delays are not independent variables (and in reality, they are not), the DP computation is dependent on the delay space. For example, in Figure 3, the areas of *A*, *B*, *C* change if the delay space changes, and then the DP changes accordingly. Therefore, if accurate correlation information is not known, the DP computation is not easy. To solve this problem, two extremities are assumed. If no correlation is assumed, path delays are independent variables. This assumption results in the lower bound of fault coverage. If 100% intra-die process correlation is assumed (only inter-die process variation is considered) [22], the upper bound of fault coverage is computed.

Applying this coverage metric (formulae 2-4) is inexpensive because only a small subset of paths must be considered. For example, Figure 4 shows the delays of four paths, each having a distribution due to process variation, through a certain fault site. Suppose path P_1 is tested by *t*, and the longest testable path P_0 is not tested. When $\Delta_0<\Delta<\Delta_1$, $DP_{i,\Delta}(t)$ is 0; when $\Delta>\Delta_2$, $DP_{i,\Delta}(t)$ is 100%, because the tested path P_1 is definitely slow; when $\Delta_1<\Delta<\Delta_2$, $DP_{i,\Delta}(t)$ increases from 0 to 100% as Δ increases. Thus, the fault coverage computation is required only in this interval. The main cost to compute the fault efficiency, which is the number of tested faults over the number of testable faults, is on the sensitization check for all the paths whose length is within this interval. However, if all the structural paths are assumed testable, a lower bound coverage can be computed and experiments show the error on ISCAS85 circuits is <4%. The cost of enumerating structural paths is low because the total number of structural paths through a gate can be computed from the number of paths of its immediate fanin and fanout gates.

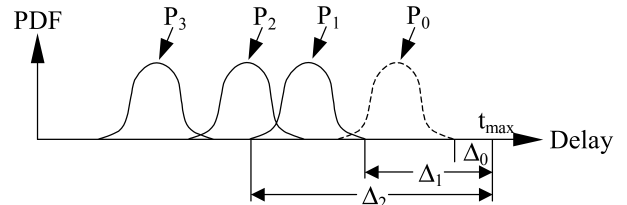

Figure 4. Fault coverage computation.

The fault coverage metric suggests a test strategy:

1. Apply transition fault tests to detect large local delay faults, and from industrial experience most local delay faults are large.

2. Apply at-speed test to one of the longest paths, e.g. the path with maximum nominal delay, through each gate or line, to eliminate or reduce the 0-DP area between Δ_0 and Δ_1 in Figure 4, because this is the second largest coverage loss factor.
3. Test more possible longest paths (such as P_2 in Figure 4, if P_0 does not exist) to increase the DP between Δ_1 and Δ_2.

Figure 5 shows the conceptual relation between fault coverage and the percentage of tested paths, using different fault coverage metrics. The paths are sorted by their nominal delays in descending order. If there is no local delay fault, the fault coverage increases quickly after the first several potentially critical paths are tested, and reaches 100% after all potentially critical paths are tested. If the percentage of local delay faults is high (in reality it is), the curves have some "jumps" because at these points the first path through some fault sites gets tested. It is clear that the new fault coverage metric is closer to industrial experience and more realistic. The traditional fault coverage is assumed to be computed as the number of tested paths over the total number of testable paths.

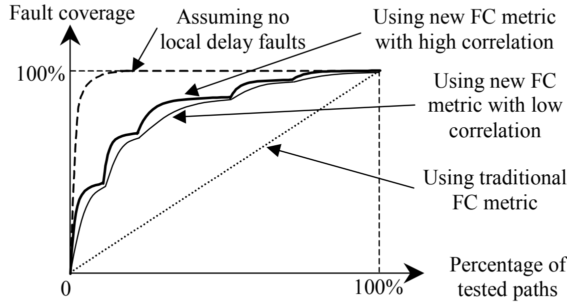

Figure 5. Fault coverage vs. percentage of tested paths.

4. Experimental Results

ISCAS85 benchmark circuits are used in the experiments. The TSMC 180 nm technology with five metal layers is used to generate the layouts. Delays for each gate and interconnect are extracted and assumed to have a Normal distribution with 3σ=10% of the nominal value. The transistor gate length, the width and thickness of the five metal layers and the thickness of the five interlayer dielectrics are considered parameters in the process variation model [22]. The maximum specified delay t_{max} is set to be 8% longer than the nominal delay of the longest testable path.

In the first experiment resistive opens are assumed on gate outputs. It is assumed that 80% of the opens have infinite resistance so that they can be detected by a transition fault test, and the remaining 20% are resistive, with log(R) uniformly distributed, where R is the open resistance [24]. This case is similar to the traditional gate delay fault model because the local delay fault size has a distribution. The delay fault coverage developed in this work is more accurate than the gate delay fault coverage because process variation is also considered. The KLPG (K longest testable paths per gate) test generator [26] is used to generate K=500 testable paths through each gate (this process takes <10 minutes), for the fault coverage computation purpose. For >99% of the gates, K=500 covers all the possible longest testable paths through the gate, assuming a ±10% path delay variation. The fault efficiency is computed because the fault sites with no transition fault test are not included in the computation and the false paths are eliminated by the KLPG test generator.

Table 1 shows the fault efficiency for the ISCAS85 circuits, using three test sets: 1. Transition fault test set; 2. KLPG-2 test set, which tests two longest paths through each gate, with one path having a rising transition and the other having a falling transition at the gate output; 3. Critical path test set, which tests the C_{size} longest testable paths throughout the circuit, where C_{size} is the circuit size (number of gates in the circuit), so that the number of vectors in this test set is about the same as the KLPG-2 test set. The transition fault test set is generated by a commercial ATPG tool. The other two test sets are generated by the KLPG test generator, and compressed by a simple greedy algorithm. All three test sets are assumed to be applied at speed.

Table 1. Fault efficiency comparison using the statistical path delay fault coverage metric.

Circuit	# of Vectors			Resistive Opens & Process Variation						Process Variation Only			
				TF	KLPG-2		Critical		TF+C	KLPG-2		Critical	
	TF	KLPG-2	Critical	UB(%)	LB(%)	UB(%)	LB(%)	UB(%)	UB(%)	LB(%)	UB(%)	LB(%)	UB(%)
c432	91	97	111	98.85	99.64	99.98	57.60	57.78	99.51	100	100	100	100
c499	92	373	200	98.68	99.59	99.96	25.88	25.95	99.27	99.81	99.94	100	100
c880	91	148	224	98.82	99.61	99.95	24.28	24.38	99.25	99.83	100	100	100
c1355	225	433	545	97.23	99.53	99.89	18.72	18.80	98.01	99.72	100	100	100
c1908	250	343	878	98.51	99.44	99.92	31.88	31.99	99.11	99.77	100	100	100
c2670	178	435	802	98.69	99.49	99.78	15.28	15.35	98.94	99.25	99.75	100	100
c3540	304	837	1 426	97.97	99.51	99.90	25.56	25.66	98.60	100	100	100	100
c5315	202	468	610	99.03	99.50	99.83	10.62	10.68	99.18	99.17	100	100	100
c6288	95	981	1 521	97.51	99.12	99.65	12.76	12.83	98.14	96.44	100	98.89	100
c7552	348	936	1 764	98.96	99.61	99.93	28.43	28.50	99.35	100	100	100	100

Column 5 shows the fault efficiency for the transition fault test. For most fault sites, the transition fault test does not test through the longest paths, but the fault efficiency is still reasonably high. The reason is that it is assumed that 80% of the resistive opens cause large extra delay. These numbers reflect the reality that the transition fault test detects most of delay faults. It should be noted that these numbers are the upper bound because it is too expensive for the KLPG test generator to generate all the paths whose length is close to the paths that the transition fault test sensitizes (since most of these paths are short, and the KLPG test generator generates long paths first). Therefore a 100% path delay correlation is used to compute the upper bound.

Columns 6 and 7 show the lower and upper bound of the fault efficiency for the KLPG-2 test set. The lower bound is computed by assuming no path delay correlation (even no structural correlation). The upper bound is computed by using an inter-die process variation model [22] and assuming 100% intra-die process correlation. The bounds are close because the majority of delay faults can be modeled as transition faults. It can be seen that the upper bound fault efficiency is almost 100% for most circuits. The reason is that for most fault sites, only 2-3 paths can be the longest paths with process correlation. Thus most fault sites have 100% DP. The fault efficiency upper bound for circuits c2670, c5315 and c6288 is lower than the other circuits because the number of longest paths per fault site for these circuits is relatively large even with process correlation. Columns 8 and 9 show the lower and upper bound of the fault efficiency for the critical path test set. The coverage loss mainly comes from the fact that many local delay fault sites have no test, because they are not on a long path. Column 10 shows the upper bound of the fault efficiency for applying both transition fault and critical path test sets, which are used in industry. The higher fault efficiency of this test reflects industrial experience [27].

Figure 6 shows the fault efficiency for c7552, assuming the K value in the KLPG test increases from 2 to 10. As can be seen, only a small number of paths are needed through each fault site to achieve high fault efficiency, and the benefit of testing one of the longest paths (the fault efficiency increase from the transition fault test to the KLPG-2 test) is more significant than that of testing more long paths (the increase from the KLPG-2 test to the KLPG-10 test).

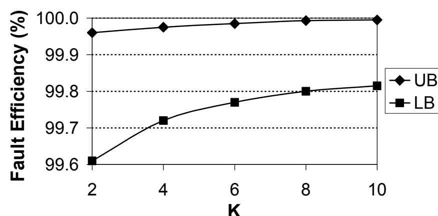

Figure 6. Fault efficiency for c7552, assuming the K longest paths through each gate are tested (K=2,…,10).

In the second experiment, no local delay faults are assumed. The circuits are only subject to inter-die process variation. This case is equivalent to the traditional path delay fault model, and the new metric computes a more accurate and reasonable fault coverage. The KLPG test generator is used to generate all the potentially critical paths to compute the fault efficiency (the CPU time is <5 minutes per circuit).

Columns under "Process Variation Only" in Table 1 show the fault efficiency for this case, using the KLPG-2 and critical path test set. The fault efficiency for the transition fault test is 0 for all these circuits, since no potentially critical paths are tested by luck (transition fault test usually sensitizes short paths to keep the test generation cost low). It can be seen that the KLPG-2 test results in high fault efficiency. This is because many paths in this test set are also potentially critical paths. After the process correlation is applied, most paths are trimmed and for most circuits there are <10 "must be tested" paths remaining (circuit c2670 has 14 paths remaining, which is the maximum number). The experiments show that the KLPG-2 test set covers all the "must be tested" paths for most circuits (KLPG-2 for c499 misses one path and KLPG-2 for c2670 misses two). It is not surprising that the critical path test set achieves higher fault efficiency, and the number of potentially critical paths in all the circuits is less than C_{size}, except for c6288.

For comparison, Table 2 shows the fault efficiency for the three test sets using the traditional transition and path delay fault coverage metrics. It can be seen that the transition fault efficiency does not reflect the real test quality since both transition fault and KLPG-2 tests have 100% fault efficiency. The traditional path delay fault efficiency for the transition fault test is not included because the cost is too high to identify all the sensitized paths. The total number of testable paths, which is used in the traditional path delay fault efficiency computation, is from the RESIST test generator [20]. As some aborted paths may also be testable, the numbers in columns 5 and 6 are the upper bound. For example, in circuit c6288 12 592 testable paths are found by RESIST and about 10^{18} paths are aborted. The traditional path delay fault efficiency is much lower than the real test quality.

Table 2. Fault efficiency comparison using traditional delay fault coverage metrics.

Circuit	Transition FE (%)			Path Delay FE (%)	
	TF	KLPG-2	Critical	KLPG-2	Critical
c432	100	100	57.81	3.01	4.30
c499	100	100	25.99	0.28	0.15
c880	100	100	24.41	2.19	2.38
c1355	100	100	18.86	1.94	2.41
c1908	100	100	32.04	0.37	0.90
c2670	100	100	15.48	4.44	8.34
c3540	100	100	25.73	1.01	1.89
c5315	100	100	10.79	1.71	2.83
c6288	100	100	13.08	11.69	19.19
c7552	100	100	28.55	1.82	4.07

5. Conclusions and Future Work

We have proposed a realistic delay fault coverage metric which considers the combined effects of spot defects and process variation. The traditional path delay fault coverage based on the percentage of tested paths is too low and does not reflect the real quality of a test set. The new coverage metric considers inter-die process correlation and suggests that a transition fault test should be applied first to detect most large delay faults, then an at-speed test should cover every line in the circuit by testing one of the longest paths through the line, and finally test more long paths through each line to increase the delay fault detection probability under process variation. As can be seen in Figure 6, when the number of tested paths per gate K becomes larger, the coverage levels off. To further increase coverage, more paths for the lines with relatively low detection probability must be tested. Assigning different K values for different lines is part of our future work. For simplicity, only inter-die process variation and single path sensitization is considered in this work. Intra-die process variation and multi-path sensitization will be considered because some local delay faults are only detectable through a set of paths.

A local delay fault can be caused by resistive shorts as well as opens, which are assumed in this work. The fault coverage metric for resistive shorts is more complicated but can be extended from this work. The single local delay fault site assumption is still valid because it has been shown that in general the extra delay caused by a resistive short can appear on only one of the bridged nets [25].

Acknowledgements

This research was funded by the Semiconductor Research Corporation under contract 2000-TJ-844 and the National Science Foundation under contract CCR-1109413.

References

[1] Z. Barzilai and B. K. Rosen, “Comparison of AC Self-Testing Procedures,” *IEEE Int’l Test Conf.*, Philadelphia, PA, Oct. 1983, pp. 89-94.

[2] J. L. Carter, V. S. Iyengar and B. K. Rosen, “Efficient Test Coverage Determination for Delay Faults,” *IEEE Int’l Test Conf.*, Washington, DC, Sept. 1987, pp. 418-427.

[3] G. L. Smith, “Model for Delay Faults Based Upon Paths,” *IEEE Int’l Test Conf.*, Philadelphia, PA, Oct. 1985, pp. 342-349.

[4] K. Heragu, V. D. Agrawal and M. L. Bushnell, “Statistical Methods for Delay Fault Coverage Analysis,” *VLSI Design Conf.*, New Delhi, India, Jan. 1995, pp. 166-170.

[5] V. Iyengar, B. K. Rosen and I. Spillinger, “Delay Test Generation 1 – Concepts and Coverage Metrics,” *IEEE Int’l Test Conf.*, Washington, DC, Sept. 1988, pp. 857-866.

[6] J. Waicukauski, E. Lindbloom, B. K. Rosen and V. S. Iyengar, “Transition Fault Simulation,” *IEEE Design & Test of Computers*, vol. 4, no. 5, Apr. 1987, pp. 32-38.

[7] T. W. Williams, B. Underwood and M. R. Mercer, “The Interdependence Between Delay-Optimization of Synthesized Networks and Testing,” *ACM/IEEE Design Automation Conf.*, San Francisco, CA, June 1991, pp. 87-92.

[8] G. M. Luong and D. M. H. Walker, “Test Generation for Global Delay Faults,” *IEEE Int’l Test Conf.*, Washington, DC, Oct. 1996, pp. 433-442.

[9] W. Maly, “Computer-Aided Design for VLSI Circuit Manufacturability,” *Proc. of the IEEE*, vol. 78, no. 2, Feb. 1990, pp. 356-392.

[10] D. M. H. Walker, “Tolerance of Delay Faults,” *IEEE Int’l Workshop on Defect and Fault Tolerance in VLSI Systems*, Dallas, TX, Nov. 1992, pp. 207-216.

[11] M. Sivaraman and A. J. Strojwas, “Delay Fault Coverage: A Realistic Metric and an Estimation Technique for Distributed Path Delay Faults,” *IEEE/ACM Int’l Conf. on Computer Aided Design*, San Jose, CA, Nov. 1996, pp. 494-501.

[12] J. J. Liou, L. C. Wang and K. T. Cheng, “On Theoretical and Practical Considerations of Path Selection for Delay Fault Testing,” *IEEE/ACM Int’l Conf. on Computer Aided Design*, San Jose, CA, Nov. 2002, pp. 94-100.

[13] A. K. Majhi, V. D. Agrawal, J. Jacob and L. M. Patnaik, “Line Coverage of Path Delay Faults,” *IEEE Trans. on VLSI Systems*, vol. 8, no. 5, Oct. 2000, pp. 610-613.

[14] A. Murakami, S. Kajihara, T. Sasao, I. Pomeranz and S. M. Reddy, “Selection of Potentially Testable Path Delay Faults for Test Generation,” *IEEE Int’l Test Conf.*, Atlantic City, NJ, Oct. 2000, pp. 376-384.

[15] M. Sharma and J. H. Patel, “Finding a Small Set of Longest Testable Paths that Cover Every Gate,” *IEEE Int’l Test Conf.*, Baltimore, MD, Oct. 2002, pp. 974-982.

[16] C. J. Lin and S. M. Reddy, “On Delay Fault Testing in Logic Circuits,” *IEEE Trans. on Computer-Aided Design*, vol. 6, no. 5, Sept. 1987, pp. 694-703.

[17] K. T. Cheng and H. C. Chen, “Classification and Identification of Non-Robust Untestable Path Delay Faults,” *IEEE Trans. on Computer-Aided Design*, vol. 15, no. 8, Aug. 1996, pp. 845-853.

[18] K. T. Cheng and H. C. Chen, “Delay Testing for Non-Robust Untestable Circuits,” *IEEE Int’l Test Conf.*, Baltimore, MD, Oct. 1993, pp. 954-961.

[19] W. K. C. Lam, A. Saldanha, R. K. Brayton and A. L. Sangiovanni-Vincentelli, “Delay Fault Coverage and Performance Trade-Offs,” *ACM/IEEE Design Automation Conf.*, Dallas, TX, June 1993, pp. 446-452.

[20] K. Fuchs, M. Pabst and T. Rossel, “RESIST: A Recursive Test Pattern Generation Algorithm for Path Delay Faults Considering Various Test Classes,” *IEEE Trans. on Computer-Aided Design*, vol. 13, no. 12, Dec. 1994, pp. 1550-1562.

[21] J. B. Brockman and S. W. Director, “Predictive Subset Testing: Optimizing IC Parametric Performance Testing for Quality, Cost, and Yield,” *IEEE Trans. on Semiconductor Manufacturing*, vol. 2, no.3, Aug. 1989, pp. 104-113.

[22] X. Lu, Z. Li, W. Qiu, W. Shi and D. M. H. Walker, “Longest Path Selection for Delay Test Under Process Variation,” *Asian and South Pacific Design Automation Conf.*, Yokohama, Japan, Jan. 2004, pp. 98-103.

[23] V. R. Sar-Dessai and D. M. H. Walker, “Resistive Bridge Fault Modeling, Simulation and Test Generation,” *IEEE Int’l Test Conf.*, Atlantic City, NJ, Sept. 1999, pp. 596-605.

[24] R. R. Montanes and J. P. Gyvez, “Resistance Characterization for Weak Open Defects,” *IEEE Trans. on Design & Test of Computers*, vol. 19, no. 5, Sept.-Oct. 2002, pp. 18-25.

[25] Z. Li, X. Lu, W. Qiu, W. Shi and D. M. H. Walker, “A Circuit Level Fault Model for Resistive Opens and Bridges,” *IEEE VLSI Test Symp.*, Napa Valley, CA, Apr. 2003, pp. 379-384.

[26] W. Qiu and D. M. H. Walker, “An Efficient Algorithm for Finding the K Longest Testable Paths Through Each Gate in a Combinational Circuit,” *IEEE Int’l Test Conf.*, Charlotte, NC, Sept. 2003, pp. 592-601.

[27] D. Belete, A. Razdan, W. Schwarz, R. Raina, C. Hawkins and J. Morehead, “Use of DFT Techniques in Speed Grading a 1GHz+ Microprocessor,” *IEEE Int’l Test Conf.*, Baltimore, MD, Oct. 2002, pp. 1111-1119.