Simulating Resistive Bridging and Stuck-At Faults

Piet Engelke, Ilia Polian, Michel Renovell, Bernd Becker

Simulating Resistive Bridging and Stuck-At Faults

Piet Engelke[1] Ilia Polian[1] Michel Renovell[2] Bernd Becker[1]

[1]Albert-Ludwigs-University
Georges-Köhler-Allee 51
79110 Freiburg im Breisgau, Germany

[2]LIRMM – UMII
161 Rue Ada
34392 Montpellier, France

Abstract

We present a simulator for resistive bridging and stuck-at faults. In contrast to earlier work, it is based on electrical equations rather than table look-up, thus exposing more flexibility. For the first time, simulation of sequential circuits is dealt with; reciprocal action of fault effects in current time frame and earlier time frames is elaborated on for different bridge resistances. Experimental results are given for resistive bridging and stuck-at faults in combinational and sequential circuits. Different definitions of fault coverage are listed and quantitative results with respect to all these definitions are given for the first time.

Keywords: Resistive bridging faults, Resistive stuck-at faults, probabilistic fault coverage, bridging fault simulation.

1 Introduction

It is commonly put aside that a substantial fraction of short defects have non-zero resistance [1]. The vast majority of the bridging fault models [2, 3, 4, 5, 6, 7, 8, 9], which describe shorts between logical nodes, assume a short resistance of zero Ohm. Also the stuck-at fault model, which can be seen as one describing short defects between a logical node and V_{DD} (stuck-at-1) or ground (stuck-at-0), does not consider resistive connections. There are only few publications and even less available tools dealing with resistive shorts [10, 11, 12, 13].

The main reason for this under-representation is that, unlike for the non-resistive case, there is an unknown value to be taken into account, the resistance. This is because it can not be known in advance which particle will cause the short defect corresponding to the bridge (parameters like its shape, size, conductivity, exact location on the die, evaporation behavior and electromigration can influence the resistance of the short defect). A short defect may be detected by a test vector for one resistance value, and the short between the same nodes may not be detected by the same vector for another resistance. This fundamentally changes the meaning of standard testing concepts, like redundancy, coverage, and so forth.

In order to handle this ambiguity, Renovell et al. [14, 15, 16] introduced the concept of *Analogue Detectability Interval* (ADI). An ADI $[R_1, R_2]$ is defined for a given fault and a given test set. The short having the resistance R_{sh} is detected by the test set if and only if R_{sh} is within this interval: $R_1 \leq R_{sh} \leq R_2$. This concept is applicable both to resistive bridging faults between two logical nodes and to resistive stuck-at faults [16]. The concept of *redundancy* was adapted to the resistive case and two possible probabilistic definitions of *fault coverage* were introduced.

Sar-Dessai and Walker [17, 18] proposed a prototype simulator and ATPG for this fault model (they used the term 'Detectable Resistance Interval' instead of ADI). Lee and Walker [19] presented a simulator with some speed-up techniques. They concentrated on resistive bridging faults in combinational circuits and employed a different definition of fault coverage than Renovell et al.; Maeda and Kinoshita [20] advocate pseudo-exhaustive test application at the bridge site.

In this paper, a simulator for resistive bridging and stuck-at faults is proposed. Its *critical resistance computation procedure* (used for calculating the bounds of the ADI) is based on electrical equations from [15, 21]. In contrast, the corresponding procedure of the only other comparable simulators we are aware of [18, 19] is based on Look-Up-Tables (LUT) generated using SPICE. Thus, if technology parameters (e. g. the supply voltage) change, then the proposed method requires only the new parameter set and no LUTs have to be re-generated. To the best of our knowledge, we are the first to present a simulator dealing with both resistive bridging and stuck-at faults in both combinational and sequential circuits.

Using our simulator, we generated the results for different known and one newly-introduced definition of fault coverage. We are not aware of such a comparison published

before. We point out the trade-off between the accuracy of a definition and the computational effort needed for obtaining the coverage figures.

The remainder of the paper is organized as follows. In Section 2 we introduce the fault model. Discussion on various fault coverage definitions follows in Section 3. The specifics of simulating sequential circuits are pointed out in Section 4. General simulation issues are covered in Section 5. In Section 6, experimental results are reported. Section 7 concludes the paper.

2 Fault Model for Resistive Faults

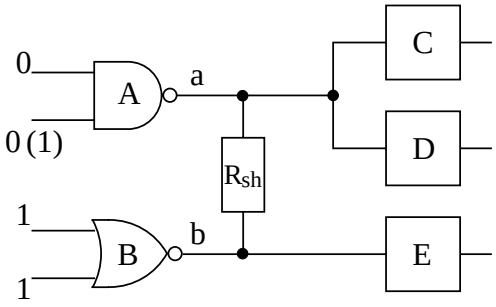

Figure 1: Example circuit

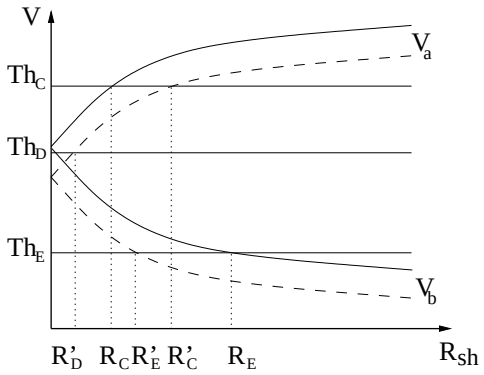

Figure 2: R_{sh}-V-diagram

An example circuit can be seen in Figure 1. The lines *a* and *b* are bridged, with *a* (*b*) being the output of a NAND2 (NOR2) gate. Let us first assume that the logical value of 0 is applied to both inputs of the NAND gate and the logical value of 1 is applied to both inputs of the NOR gate. In CMOS, two p transistors from the pull-up network of the gate A (connected in parallel) drive the node a, and two n transistors (also in parallel) from the pull-down network of the gate B drive the node b. Thus, in absence of the bridge there will be a 1 on *a* and a 0 on *b*. The voltages on *a* and *b* in presence of the bridge, V_a resp. V_b, depend on the bridge resistance R_{sh}. For $R_{sh} = 0\Omega$, there will be some intermediate voltage identical for both lines. For $R_{sh} = \infty$, V_a will equal V_{DD} and V_b will equal 0V, as if the bridge were not present. A possible voltage distribution for intermediate values of R_{sh} (those between 0Ω and ∞) is depicted by solid curves in Figure 2. The abscissa corresponds to different values of R_{sh}, the ordinate shows which voltages are assumed on the lines *a* and *b* if the bridge has such a resistance. With increasing R_{sh}, V_a and V_b diverge, with V_a approaching V_{DD} and V_b approaching 0.

The question of interest in simulation is how the succeeding gates will interpret these voltages. In our example circuit, gates C and D are successors of *a* and gate E is successor of *b*. Neither the gate types of C, D and E are important in this example, nor whether they have additional inputs. Rather, it is relevant whether they interpret the voltage on their input as a logical value of 1 or a logical value of 0. In accordance to previous works, we assume an exact-defined threshold voltage Th, which however may be different for different gate types. Note that there may be many different gate types with the same functionality, e. g. different NOR2 gates having different thresholds. Also, different inputs of the same gate may have different thresholds. Thus, we rule out that some voltage is not recognized as a logical value; any voltage above Th is interpreted as the logical value of 1, and any below as the logical value of 0.[1] Moreover, we also neglect that for different manufactured ICs, the threshold of the same gate may vary.

In Figure 2, the thresholds for the gates C, D, E are shown as horizontal lines labeled by Th_C, Th_D and Th_E, respectively. Consider the gate C. Given a resistance R_{sh}, this gate will either interpret the value on *a* as 1 or as 0. Being more exact, there is a *critical resistance*, denoted as R_C in the picture, so that for $R_{sh} < R_C$ the value on *a* is interpreted as 0 and for $R_{sh} > R_C$ it is interpreted as 1. The intuition behind this is that for a bridge with low resistance, the value 0 on the line *b* has larger impact on the voltage on *a* than for a highly-resistive bridge. Hence, if we want to detect the bridge by propagating the faulty value through the gate C to some observable point, we may be able to do this only for the bridge

[1] In their study of (non-resistive) bridging faults in an AMD design, Ma et al. [22] reported that disregarding potentially ambiguous intermediate voltages in the vicinity of the threshold had an impact on fault coverage which was below 0.007%.

resistance $R_{sh} < R_C$, because otherwise no fault effect will be visible on C's input. Since for the critical resistance R_C $V_a(R_C) = Th_C$ holds, R_C can be determined in Figure 2 by finding the intersection of the curve V_a with Th_C.

For the gate D, the threshold Th_D is below the curve. This means that for any R_{sh} the gate D will recognize the voltage on *a* as logical value of 1 and there is no critical resistance; no fault effect can be propagated. For the gate E, the solid curve V_b is relevant, and there is a critical resistance R_E. E interprets the voltage on *b* as faulty logical value (1) only for $R_{sh} < R_E$.

Now imagine that there is a logical value of 1 on the second input of the NAND gate. Then, only one p transistor will pull up the voltage on the line *a* to the power supply. This results in 1 being driven with less strength on *a*. With 0 driven on *b* with the same strength as before (two parallel n transistors), the voltage characteristic for V_a and V_b in the R_{sh}-V-diagram will be described by curves situated underneath the original ones (one possibility is shown by the dashed curves). This results in new critical resistances R'_C and R'_E; furthermore, there is now a critical resistance for D, R'_D (there was no intersection between the V_a curve and Th_D before). The shown shift of the curves plays an important role in the *multiple strengths problem* discussed later on.

Unlike the approach in [19], which uses Look-Up-Tables created by SPICE for determining the critical resistance R_{sh_L}, here electrical equations from [15, 21] are used for this purpose. For a resistive bridging fault, there is an equation valid for the gate succeeding the bridged node which is driven by the p transistor network (it is the node with logical good-value of 1 on it, being pulled down towards 'logic 0' by the other line involved in the bridge):

$$R_{sh_L} = K_1 \cdot \left[Th_C - V_{gn} + \sqrt{{V_{gn}}^2 - \frac{2}{K_1 C_{ox} \mu_n W_n / L_n}} \right]$$

In this equation, $V_{gn} = V_{dd} - V_{tn0}$ and

$$K_1 = \frac{1}{\mu_p C_{ox} \frac{W_p}{L_p} \left[V_{gp}(V_{dd} - Th_C) - \frac{(V_{dd} - Th_C)^2}{2} \right]}$$

V_{dd} denotes the power supply voltage, C_{ox} the oxide capacity, W_p/L_p and W_n/L_n the length-width ration of the p and n transistor, μ_p and μ_n the mobility, V_{tp0} and V_{tn0} the zero bias threshold voltage, Γ_p and Γ_n the body effect coefficient, Φ_p and Φ_n the electrostatic potential of substrate and Th_C the threshold voltage of the relevant input of the succeeding gate. If the node is driven by the n transistor network, the following equation is used:

$$R_{sh_L} = K_2 \cdot \left[|V_{tp0}| - Th_E + \sqrt{{V_{gp}}^2 - \frac{2}{K_2 C_{ox} \mu_p W_p / L_p}} \right]$$

In this equation, $V_{gp} = V_{dd} - |V_{tp0}|$ and

$$K_2 = \frac{1}{\mu_n C_{ox} \frac{W_n}{L_n} \left[V_{gn} Th_E - \frac{{Th_D}^2}{2} \right]}$$

For resistive stuck-at faults, the equations are: $R_{sh_L} = K_2 \cdot (V_{dd} - Th)$ for a stuck-at-1 fault; and $R_{sh_L} = K_1 \cdot Th$ for a stuck-at-0 fault, with K_1, K_2 as defined above and Th denoting the threshold of the gate succeeding the faulty line.

As we have stated above, the gate C interprets the voltage on *a* as the logical value of 0 for $R_{sh} < R_C$ and as 1 for $R_{sh} > R_C$. We denote this situation by the *analogue detectability interval* [15] $[R_C, \infty]1/0$, which means that 1 is assumed for the values of R_{sh} within the interval and 0 for all other values. We could also have written $[0, R_C]0/1$ to describe the same situation[2]. Note that this interval is only valid for the values (0, 0, 1, 1) applied to the NAND and NOR gates; for (0, 1, 1, 1), the interval would have been $[R'_C, \infty]1/0$. Such an interval can be propagated through the gate C and the following gates to the primary outputs or other observable points using rules described, e. g. in [19].

3 Fault Coverage Definitions

Under the resistive fault model assumption, the faulty effects are dependent on the bridge resistance, which is unknown *ex ante*. The simulation yields for each fault and each output a resistance range, the ADI, for which the fault is detected. An ADI may be an interval like $[R_1, R_2]$. Typically, R_1 is 0Ω, but this is not necessarily the case. Moreover, for circuits having reconvergencies and sequential circuits the ADI may be the union of multiple intervals like $[R_1, R_2] \cup [R_3, R_4]$ [16]. Let the circuit under test have m outputs and let the test vector t be applied to the circuit's inputs. We denote the ADI propagated to the output j by $ADI_j(t)$. Given a test set $t_1, t_2, \ldots, t_k$, the C-ADI of the fault is defined as $\cup_{i=1}^{k} \cup_{j=1}^{m} ADI_j(t_i)$ (C stands for 'covered by the test set').

[2] The voltage for interval boundaries, e. g. $R_{sh} = R_C$, cannot be resolved unambiguously under the proposed model. As it will become clear, our fault coverage definitions are based on an integral over a range of possible resistances. Since values for finitely many single points do not contribute to the integral's value, we ignore the voltages on the interval boundaries.

It is preferable to have one single number indicating the quality of the test set rather than its C-ADI, in order to ensure comparability. There are several definitions of *fault coverage*. Here, they are given different names in order to be distinguishable; in the original literature they are just called fault coverage. Note that they have mostly been proposed for resistive bridging faults but they can be used for resistive stuck-at faults, as well.

Let $\rho(r)$ be the probability density function of the short resistance r. In [15], the Normal distribution is suggested to describe $\rho(r)$, in [19] authors use an other distribution. The *Pessimistic Fault Coverage* (P-FC) introduced in [15] is defined for one fault f as

$$P\text{-FC}(f) = 100\% \cdot \left(\int_{C\text{-ADI}} \rho(r)dr \right) / \left(\int_0^\infty \rho(r)dr \right)$$

This definition relates the 'fraction' of the ranges in which the fault is detected to the complete range from 0 to ∞, 'weighted' by ρ. ρ is usually chosen in a way that the second integral equals to 1. For N faults $f_1, \ldots, f_N$, the average fault coverage is taken:

$$P\text{-FC} = \frac{1}{N} \cdot \sum_{i=1}^{N} P\text{-FC}(f_i)$$

Note that there may be some resistance ranges which are not in C-ADI for any possible test vector. This means that under no circumstances a faulty value will be observed for a short defect with such a resistance (at least if effects on reliability, signal propagation time and IddQ are not considered). Such defects can be seen as *redundant*; the fact that they are included in the second integral makes P-FC pessimistic.

In [16], a definition has been proposed which bases on G-ADI, where G-ADI is defined as C-ADI of an exhaustive test set (consisting of 2^n test vectors for a combinational circuit with n inputs). The letter G stands for 'global'. Here, this fault coverage definition is referred to as G-FC.

$$G\text{-FC}(f) = 100\% \cdot \left(\int_{C\text{-ADI}} \rho(r)dr \right) / \left(\int_{G\text{-ADI}} \rho(r)dr \right)$$

This definition can be considered to be exact. However, up to now there is no known method how to determine G-ADI without simulating all 2^n test vectors. Thus, G-FC can only be computed for circuits with relatively few inputs. Furthermore, the generalization to the sequential case raises the question how unreachable states should be dealt with.

The fault coverage definition from [19] is based on the local analysis of the fault site. Let $R_{\max}$ be the maximal critical resistance of any gate succeeding the bridge. This means that, if R_{sh} is between 0Ω and $R_{\max}$, a faulty effect can be recognized by at least one of the gates driven by the bridge (for at least one excitation). Conclusively, E-FC is defined as

$$E\text{-FC}(f) = 100\% \cdot \left(\int_{C\text{-ADI}} \rho(r)dr \right) / \left(\int_0^{R_{\max}} \rho(r)dr \right)$$

Obviously, this definition makes sure that a fault can be excited, but it does not ensure that it is also propagated to an observable point (E in the name is derived from 'excitation'). Thus, some redundant defects may still be accounted for in the second integral. An advantage of this definition is that $R_{\max}$ can be computed locally at the fault site without much effort. Thus, this definition is more exact (not as pessimistic) than P-FC and has lower computational complexity than G-FC.

Now, a new definition of fault coverage, the *Optimistic fault coverage* (O-FC) is introduced. O-FC of a single fault is set to 100% if its ADI is not empty, i. e. if there is at least one test vector in the test set and at least one bridge resistance (R_{sh}) value for which the faulty effect is propagated to the outputs. Accordingly, the fault coverage is defined for N faults as the fraction of those faults which can be detected by the test set for at least one R_{sh} among all faults. This definition reminds of a non-probabilistic definition utilized in non-resistive models. It can be used for comparison purposes.

Considering the denominators in the definitions of P-FC, E-FC and G-FC, it is clear that $G\text{-ADI} \subseteq [0, R_{\max}] \subset [0, \infty]$. Since ρ assumes only positive values and the numerator in all three definitions is the same, P-FC $\leq$ E-FC $\leq$ G-FC holds for each individual fault. It is furthermore obvious that G-FC is always less or equal than O-FC. Aggregating for all faults, we obtain the following relationship:

$$P\text{-FC} \leq E\text{-FC} \leq G\text{-FC} \leq O\text{-FC}$$

Since G-FC is the most exact fault coverage definition, it should be used whenever possible. As stated before, unfortunately, no efficient method to compute G-ADI is available at present. In contrast, fault coverage with respect to all other definitions can be calculated efficiently. Hence, the relationship above can be utilized to approximate G-FC by E-FC (lower bound) and O-FC (upper bound).

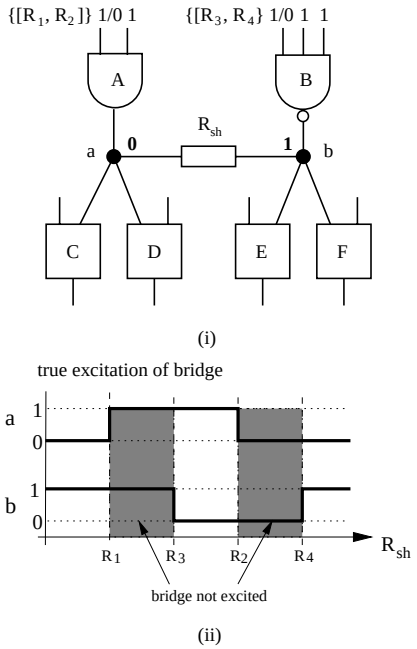

Figure 3: Example bridge and fault excitation

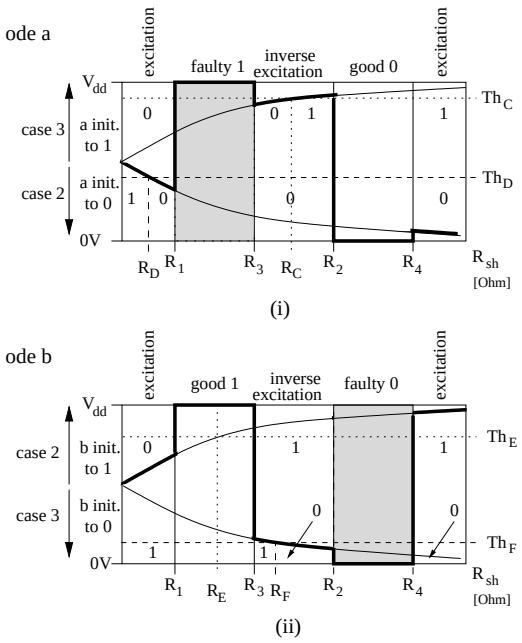

Figure 4: Voltage depending on bridge resistance R_{sh}

4 Sequential Case

When simulating a sequential circuit, a fault effect (an ADI, like $[R_u, R_v]1/0$) can arrive at the inputs of a memory element. This means that for $R_{sh} \in [R_u, R_v]$, logic 1 would be written into the memory element, while for all other R_{sh} values logic 0 would. In the next frame, this interval will be present at the output of the memory element (which is treated like a primary input during simulation). From there, it can be propagated to the lines involved in the bridge and affect its excitation.

Such a situation can be seen in Figure 3 (i). In this case it is no longer the current pattern exciting the bridge (as in combinational case). Instead, the bridge excitation is influenced by the intervals coming from the memory elements. For some R_{sh} values, the bridge is not excited, and even situations are possible in which the faulty-values on both bridged nodes are opposite to the good-values.

4.1 ADIs at Secondary Inputs

As pointed out above, we consider an interval attached to a signal line to define the resistance ranges for which the line's logical value is 1 [21]. Consider the example from the Figure 3. Let $I_a = [R_1, R_2]$ be the interval on the left input of the gate A, which is propagated unmodified to the node *a*. Similarly, let $I_b = [R_3, R_4]$ be the interval on the left input of the gate B. Conclusively, the union of intervals $[0, R_3] \cup [R_4, \infty]$ is propagated to *b*. The logical good-values are 0 at the node *a* and 1 at *b*.

Depending on the bridge resistance R_{sh}, there are the following possibilities:

1. $R_{sh} \in I_a$ and $R_{sh} \in I_b$: Both *a* and *b* are assigned the logical value of 1; the bridge is not excited. All succeeding gates interpret the resulting value as 'logic 1'. 'Logic 1' being the faulty value on *a*, the difference can be propagated through C or D to a primary output.
2. $R_{sh} \notin I_a$ and $R_{sh} \in I_b$: 'Logic 0' at *a*, 'logic 1' at b, this corresponds to the values imposed by good simulation. The bridge is excited and can evoke faulty effects, as in combinational case.
3. $R_{sh} \in I_a$ and $R_{sh} \notin I_b$: 'Logic 1' at *a*, 'logic 0' at *b*. This is the exact reversal of the bridge excitation which would occur in combinational case.
4. $R_{sh} \notin I_a$ and $R_{sh} \notin I_b$: Both *a* and *b* are assigned the logical value of 0; the bridge is not excited. All succeeding gates interpret the resulting value as 'logic 0'. 'Logic 0' being the faulty value on *b*, the difference can be propagated through E or F to a primary output.

In Figure 3 (ii), Cases 1—4 are demonstrated for the *hypothetical* situation that the bridge *has been present* in ear-

lier time frames (thus triggering the creation of the intervals which now emerge from the secondary inputs) but is *absent in the current time frame*. This is not what actually happens in the circuit and shown only to illustrate the above-mentioned cases: Case 1 in $[R_1, R_3]$, Case 2 in $[0, R_1]$ and $[R_4, \infty]$, Case 3 in $[R_3, R_2]$ and Case 4 in $[R_2, R_4]$.

In reality, the bridge present in earlier time frames is obviously also present in the current one. Figure 4, demonstrates what happens at nodes *a* and *b* when the bridge is considered also in current time frame. In intervals $[R_1, R_3]$, and $[R_2, R_4]$, both *a* and *b* have the same logical value which will be interpreted as such by all succeeding gates, regardless of the bridge's resistance (as it is not excited). In interval $[R_3, R_2]$, the gate A pulls the node *a* to V_{DD} while the node *b* is pulled to GND. Thus, the interpretation for gates C and D is given by the upper curve in In Figure 4 (i). If the threshold for C intersects the curve exactly within this interval (R_C), the interpretation will be 0 for $R \in [R_3, R_C]$ and 1 for $R \in [R_C, R_2]$. If the threshold for D is below the curve for the whole interval, D will interpret this voltage as 1. The reverse case applies in intervals $[0, R_1]$ and $[R_4, \infty]$. For the situation at node b and gates E and F, refer to Figure 4 (ii).

4.2 Multiple Strengths Problem

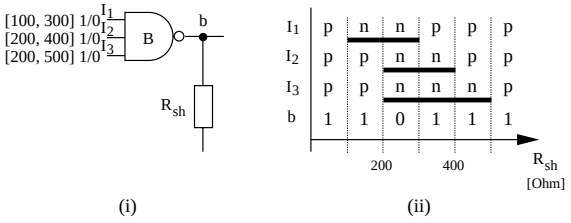

Figure 5: The Multiple Strengths Problem

Consider the NAND3 gate B from Figure 3, as depicted in Figure 5 (i). Let there be an interval on each input of this gate: [100, 300] 1/0 at the input i_1, [200, 400] 1/0 at i_2 and [200, 500] 1/0 at i_3. Let furthermore R_1 be 200 and R_2 be 300.

Depending on what interval R_{sh} is in, a different number of n transistors will be active, as indicated in Figure 5 (ii). For $R_{sh} \in [200, 300]$, there will be the logical value of 0 at the output *b*. In all other cases, the logical value of 1 is obtained at *b*, but it is driven by different number of p transistors. So, the strength of this gate will be different for different R_{sh} values. This will lead to different characteristics and hence different critical resistances depending on

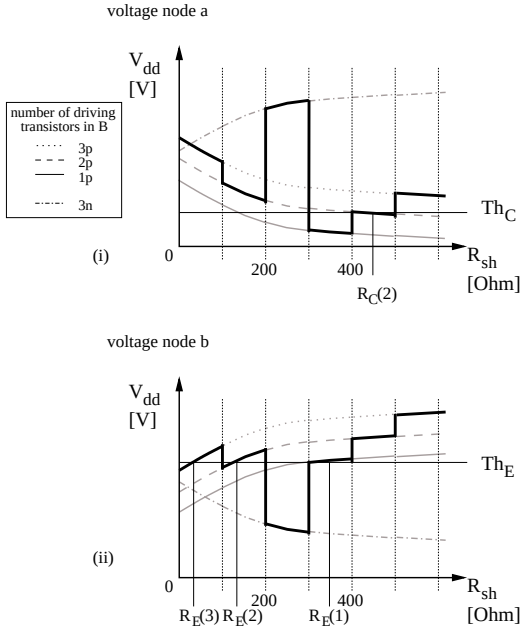

Figure 6: Voltages under Multiple Strengths Problem

which of the ADIs R_{sh} is in, as shown in Figure 6. If the computed critical resistance is out of this ADI, then the voltage will be interpreted as either the logical value of 0 or 1 throughout the interval.

The multiple strengths problem also occurs for resistive feedback bridging faults. We cannot elaborate on this topic here for the reasons of limited space.

5 Simulation Issues

We implemented the simulator in the C++ programming language. It is based on an event-driven engine similar to the ones known for stuck-at fault simulation. The ADI is computed at the fault site using equations from Section 2. For ADI propagation, we employ an efficient technique originally proposed by Huc [21]: instead of distinguishing between the interval $[R_1, R_2]$ 1/0 and $[R_1, R_2]$ 0/1, each interval is implicitly assumed to describe the range in which the signal has the logical value of 1. Thus, the first interval mentioned becomes simply $[R_1, R_2]$ and the second one becomes $[0, R_1] \cup [R_2, \infty]$, and no '1/0' or '0/1' is required. If an interval (or a non-contiguous union of several intervals) arrives at a primary output, the good-value on this output is consulted. If the latter is logic 0, then this interval is the ADI (the interval in which the fault is detected). If the good-value is 1, the ADI is given as $[0, \infty]$ minus the interval. Apart from this technique, we do not employ any speed-ups for

propagation, such as the "PPSFP" method from [19].

If multiple intervals are present at the inputs of the bridged gates, the procedure computing critical resistances checks each sub-range. Thus, the simulator can deal with the Multiple Strengths Problem outlined in Section 4.2. P-FC, E-FC and O-FC are calculated simultaneously, i. e. in the same simulator run. G-FC is computed by first explicitly simulating the exhaustive test set and then the test vectors. More efficient methods to determine G-FC are currently under development.

Resistive feedback faults can also be handled by the simulation approach described in this paper. For detailed description of specific modeling and simulation issues and experimental results for these faults, please refer to [23]. We had to exclude faults between inputs of the same gate and faults involving primary and secondary inputs and outputs due to the limited validity of the electrical model in these cases. Resistive stuck-at faults at a fanout stem and all fanout branches were considered electrically equivalent and thus only one of them was included in the fault list.

6 Experimental Results

1,000 random test vectors were applied to ISCAS 85 and 89 benchmark circuits. The fault set consisted of 10,000 randomly selected non-feedback faults, where available. We employed the density function ρ derived from one used in [19] for all experiments, in order to ensure comparability. All measurements were performed on a 2GHz Linux machine with 1 GB RAM.

Circuit	#faults	found	P-FC	E-FC	G-FC	O-FC
cs00208	3986	3932	83.36	95.34	98.38	98.65
cs00298	4468	4434	83.98	97.57	99.22	99.24
cs00386	9384	9268	80.29	96.50	98.65	98.76
cs01488	10000	9941	82.37	97.84	99.21	99.41
cs01494	10000	9943	82.50	97.78	99.24	99.43
∅			82.50	97.01	98.94	99.10

Table 1: Results including G-FC, bridging faults

Circuit	#faults	found	P-FC	E-FC	G-FC	O-FC
cs00208	190	185	93.46	97.35	97.35	97.37
cs00298	198	198	96.23	100.00	100.00	100.00
cs00386	292	288	94.61	98.63	98.63	98.63
cs01488	1256	1246	95.12	99.20	99.20	99.20
cs01494	1244	1231	94.87	98.95	98.95	98.95
∅			94.86	98.83	98.83	98.83

Table 2: Results including G-FC, stuck-at faults

In Tables 3 and 4 the results for resistive bridging and stuck-at faults, respectively, are reported. Circuit name is followed by the number of considered faults. The third column contains the number of faults detected *for at least one* bridge resistance value. Fault coverage (in percent) according to various definitions is given in the next three columns. Last column contains the run time in CPU seconds.

The figures on G-FC are not quoted as they require an exhaustive simulation which is impractical for the vast majority of the circuits. They are given in Tables 1 and 2 for the combinational parts of sequential benchmark circuits (indicated as cs) having a reasonable number of inputs. Average numbers on fault coverage are quoted in the last line of each table.

It can be seen that for each circuit, P-FC $\leq$ E-FC $\leq$ G-FC $\leq$ O-FC holds. For most circuits as well as on average, the quantitative difference in fault coverage between P-FC and E-FC is larger than among E-FC, G-FC and O-FC. This indicates that P-FC is indeed an overly pessimistic measure and that the computational overhead needed for calculating fault coverage according to other definitions may be a worthy investment.

Since G-FC is considered 'the' exact fault coverage definition, the results from Tables 1 and 2 are of special interest. It can be seen that also here figures for P-FC are an outlier while E-FC and O-FC seem to provide good approximations for G-FC. For the circuits considered, O-FC comes closer to G-FC than E-FC does. However, a reason therefore may be that the considered benchmarks have relatively few reconvergencies.

Comparing the results for resistive bridging faults and resistive stuck-at faults, no clear conclusion can be drawn. While in Tables 1 and 2, the average fault coverage is always higher for stuck-at faults (however this does not hold for each individual entry), average results from Tables 3 and 4 are mixed.

To conclude, G-FC is the most exact fault coverage measure, and methods to compute G-FC without having to apply the exhaustive test set should be developed. As long as such methods do not exist, the following approach can be utilized: *both* E-FC and O-FC should be computed (this can easily be done in one simulation pass). Due to its definition, G-FC is guaranteed to lie between these two numbers. If E-FC and O-FC are close enough to each other, the value of G-FC may be considered approximated accurately enough for all practical purposes. If E-FC and O-FC are too far apart, additional analysis, e. g. considering the propagation path, may be triggered.

Circuit	#faults	found	P-FC	E-FC	O-FC	time [s]
c0017	2	2	78.72	98.59	100.00	0.03
c0095	77	77	84.73	95.90	100.00	1.90
c0432	5253	5215	83.72	98.03	99.28	131.77
c0499	8985	8849	78.30	97.07	98.49	281.72
c0880	10000	9983	84.52	97.01	99.83	251.98
c1355	10000	9979	77.19	97.06	99.79	502.14
c1908	10000	9833	77.71	97.40	98.33	630.39
c2670	10000	9664	77.26	94.95	96.64	413.05
c3540	10000	9904	81.19	97.84	99.04	545.59
c5315	10000	9994	81.73	99.59	99.94	493.58
c6288	10000	9991	87.11	91.88	99.91	1385.30
c7552	10000	9952	80.37	98.60	99.52	592.41
s00027	23	23	89.41	98.43	100.00	0.43
s00208	5207	5099	81.99	94.16	97.93	481.55
s00298	7056	6492	76.21	86.66	92.01	278.33
s00349	10000	9891	80.38	92.72	98.91	498.37
s00382	10000	2815	22.15	25.68	28.15	301.80
s00386	10000	9259	73.77	89.23	92.59	417.54
s00400	10000	2917	22.75	26.30	29.17	308.87
s00420	10000	6510	52.77	60.35	65.10	715.79
s00444	10000	2545	19.67	22.85	25.45	325.32
s00510	10000	9979	85.06	95.24	99.79	5105.30
s00526	10000	2492	20.33	22.87	24.92	443.73
s00641	10000	9810	74.52	96.51	98.10	423.63
s00713	10000	9736	74.30	95.07	97.36	451.88
s00820	10000	6318	53.17	58.81	63.18	622.47
s00832	10000	6334	53.16	58.85	63.34	674.02
s00838	10000	4496	35.95	41.21	44.96	441.92
s00953	10000	9929	83.43	93.57	99.29	1576.96
s01238	10000	9495	78.60	91.35	94.95	192.88
s01423	10000	5669	44.96	52.44	56.69	366.01
s01488	10000	7758	62.30	74.17	77.58	635.00
s01494	10000	7686	61.72	73.45	76.86	654.42
s05378	10000	7925	66.19	76.03	79.25	646.76
s09234	10000	3202	23.69	29.84	32.02	2024.56
s1196	10000	9484	78.05	91.36	94.84	265.30
s1269	10000	9981	84.30	97.23	99.81	1943.51
s13207	10000	4708	34.95	44.88	47.08	1902.89
s1512	10000	6589	53.27	63.02	65.89	550.50
s15850	10000	5348	39.98	50.77	53.48	1591.32
s3271	10000	9991	83.70	96.90	99.91	2703.55
s3330	10000	8280	64.78	80.45	82.80	1124.66
s3384	10000	9863	81.42	97.60	98.63	3483.88
s344	10000	9879	80.22	92.66	98.79	474.89
s35932	10000	8564	65.20	81.77	85.64	831.15
s38417	10000	2615	19.42	23.87	26.15	4723.55
s38584	10000	7426	59.33	70.62	74.26	5704.85
s4863	10000	9916	81.50	98.40	99.16	1143.17
s499	8407	8114	77.95	88.55	96.51	1951.05
s635	10000	0	0.00	0.00	0.00	253.04
s6669	10000	10000	83.43	99.92	100.00	3418.52
s938	10000	4459	35.72	41.01	44.59	484.87
∅			64.08	75.74	78.77	

Table 3: Experimental Results: Resistive bridging faults

Circuit	#faults	found	P-FC	E-FC	O-FC	time [s]
c0017	8	8	96.93	100.00	100.00	0.09
c0095	40	40	97.50	100.00	100.00	0.92
c0432	306	303	95.59	99.02	99.02	6.52
c0499	340	339	95.96	99.71	99.71	9.36
c0880	714	700	94.37	98.04	98.04	15.96
c1355	1028	1020	95.85	99.22	99.22	41.20
c1908	1710	1648	92.91	96.37	96.37	88.83
c2670	2258	1828	77.93	80.95	80.96	78.74
c3540	3294	3152	92.01	95.69	95.69	137.09
c5315	4368	4365	96.09	99.93	99.93	167.56
c6288	4768	4751	95.20	99.64	99.64	618.68
c7552	6810	6357	89.84	93.34	93.35	316.11
s00027	24	24	96.09	100.00	100.00	0.35
s00208	222	206	89.16	92.67	92.79	11.49
s00298	254	210	79.16	82.12	82.68	9.14
s00349	330	319	92.87	96.66	96.67	13.10
s00382	346	73	19.93	20.73	21.10	8.70
s00386	316	272	82.50	86.05	86.08	12.78
s00400	354	74	19.76	20.54	20.90	8.89
s00420	466	243	49.31	51.31	52.15	20.86
s00444	392	70	16.91	17.58	17.86	10.40
s00510	420	420	96.42	100.00	100.00	141.00
s00526	418	70	16.03	16.60	16.75	13.96
s00641	748	669	86.00	89.43	89.44	26.65
s00713	778	681	84.17	87.51	87.53	29.03
s00820	550	266	46.15	47.91	48.36	27.88
s00832	546	263	45.93	47.68	48.17	28.52
s00838	954	379	37.55	39.13	39.73	28.68
s00953	802	771	92.58	96.12	96.13	71.65
s01238	1024	892	83.72	87.01	87.11	13.36
s01423	1452	643	42.26	43.98	44.28	40.06
s01488	1280	931	69.41	72.72	72.73	107.53
s01494	1268	923	69.45	72.77	72.79	109.25
s05378	5818	3990	65.73	68.42	68.58	307.98
s09234	11538	2645	21.69	22.58	22.92	1526.33
s1196	1066	934	84.23	87.52	87.62	13.89
s1269	1192	1192	96.14	100.00	100.00	145.54
s13207	16874	5674	32.19	33.45	33.63	2398.23
s1512	1632	900	52.91	54.98	55.15	63.62
s15850	20312	8534	40.11	41.75	42.01	2487.94
s3271	3348	3291	94.50	98.28	98.30	528.52
s3330	3696	2670	69.29	72.23	72.24	169.53
s3384	3684	3298	85.74	89.33	89.52	601.60
s344	328	318	93.15	96.94	96.95	12.66
s35932	34946	27286	75.15	78.06	78.08	2184.97
s38417	47418	7887	15.69	16.31	16.63	9418.26
s38584	40750	24175	56.79	59.11	59.33	11627.82
s4863	4860	4567	90.20	93.97	93.97	513.16
s499	304	276	87.05	89.71	90.79	48.21
s635	634	0	0.00	0.00	0.00	11.82
s6669	6528	6512	95.71	99.75	99.75	1507.01
s938	954	379	37.55	39.13	39.73	28.77
∅			69.80	72.54	72.70	

Table 4: Experimental Results: Resistive stuck-at faults

7 Conclusions

A simulation technique for resistive short defects, modeled as resistive bridging and stuck-at faults, has been described for combinational and sequential circuits. A simulator for this fault model has been implemented. Its critical resistance computation procedure is based on electrical equations. Experimental results have been reported for resistive bridging and stuck-at faults in combinational and sequential circuits with respect to four different fault coverage definitions, one of them being new. As the only exact definition requires prohibitive computational efforts, we discuss using other definitions as upper and lower bounds for the exact metric.

Main directions for our research in the future will include: efficient ways for the computation of G-FC; incorporating speed-up techniques into the simulator; and automatic test pattern generation for resistive short defects.

8 References

[1] R. Rodríguez-Montañés, E.M.J.G. Bruls, and J. Figueras. Bridging defects resistance measurements in a CMOS process. In *Int'l Test Conf.*, pages 892–899, 1992.

[2] Kenyon C. Y. Mei. Bridging and stuck-at faults. *IEEE Trans. on Comp.*, C-23(7):720–727, July 1974.

[3] F.J. Ferguson and T. Larrabee. Test pattern generation for realistic bridge fault in CMOS ICs. In *Int'l Test Conf.*, pages 492–499, 1991.

[4] F. Joel Ferguson and J. Shen. Extraction and simulation of realistic CMOS faults using inductive fault analysis. In *Int'l Test Conf.*, pages 475–484, 1988.

[5] 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 *Int'l Test Conf.*, pages 63–72, 1993.

[6] J.M. Acken and S.D. Millman. Fault model evolution for diagnosis; accuracy vs precision. In *Custom Integrated Circuits Conference*, pages 13.4.1–13.4.4, 1992.

[7] J. Rearick and J.H. Patel. Fast and Accurate CMOS Bridging Fault Simulation. In *Int'l Test Conf.*, pages 54–62, 1993.

[8] Prithviraj Banerjee and Jacob A. Abraham. A multivalued algebra for modeling physical failures in MOS VLSI circuits. *IEEE Trans. on CAD*, 4(5):312–321, 1985.

[9] I. Polian, P. Engelke, and B. Becker. Efficient bridging fault simulation of sequential circuits based on multi-valued logics. In *Int'l Symp. on Multi-Valued Logic*, pages 216–222, 2002.

[10] H. Hao and E.J. McCluskey. Resistive shorts within CMOS gates. In *Int'l Test Conf.*, pages 292–301, 1991.

[11] M. Favalli, M. Dalpasso, P. Olivo, and B. Ricco. Analysis of dynamic effects of resistive bridging faults in CMOS and BiCMOS digital ICs. In *Int'l Test Conf.*, pages 865–873, 1993.

[12] H. Vierhaus, W. Meyer, and U. Glaser. CMOS bridges and resistive transistor faults: IDDQ versus delay effects. In *Int'l Test Conf.*, pages 83–91, 1993.

[13] M. Favalli and M. Dalpasso. Bridging fault modeling and simulation for deep submicron CMOS ICs. *IEEE Trans. on CAD*, 21(8):941–953, August 2002.

[14] M. Renovell, P. Huc, and Y. Bertrand. CMOS bridge fault modeling. In *VLSI Test Symp.*, pages 392–397, 1994.

[15] M. Renovell, P. Huc, and Y. Bertrand. The concept of resistance interval: A new parametric model for resistive bridging fault. In *VLSI Test Symp.*, pages 184–189, 1995.

[16] M. Renovell, F. Azaïs, and Y. Bertrand. Detection of defects using fault model oriented test sequences. *Jour. of Electronic Testing: Theory and Applications*, 14:13–22, 1999.

[17] V. Sar-Dessai and D.M.H. Walker. Accurate fault modeling and fault simulation of resistive bridges. In *Int. Symp. Defect and Fault Tolerance in VLSI Systems*, pages 102–107, 1998.

[18] V. Sar-Dessai and D.M.H. Walker. Resistive Bridge Fault Modeling, Simulation and Test Generation. In *Int'l Test Conf.*, pages 596–605, 1999.

[19] C. Lee and D. M. H. Walker. PROBE: A PPSFP simulator for resistive bridging faults. In *VLSI Test Symp.*, pages 105–110, 2000.

[20] T. Maeda and K. Kinoshita. Precise test generation for resistive bridging faults of CMOS combinational circuits. In *Int'l Test Conf.*, pages 510–519, 2000.

[21] P. Huc. *Test en tension des courts-circuits en technologie CMOS*. PhD thesis, Université de Montpellier II Sciences et Techniques du Languedoc, Montpellier, France, March 1995.

[22] S. Ma, I. Shaik, and R. Scott-Fetherston. A comparison of bridging fault simulation methods. In *Int'l Test Conf.*, pages 587–595, 1999.

[23] I. Polian, P. Engelke, M. Renovell, and B. Becker. Modelling feedback bridging faults with non-zero resistance. In *European Test Workshop*, 2003.