Resistive Bridge Fault Model Evolution
From Conventional to Ultra Deep Submicron Technologies

Ilia Polian\textsuperscript{1} Sandip Kundu\textsuperscript{2} Jean-Marc Galli\textsuperscript{ère}\textsuperscript{3} Piet Engelke\textsuperscript{1}
Michel Renovell\textsuperscript{3} Bernd Becker\textsuperscript{1}
\textsuperscript{1}Albert-Ludwigs-University Georges-Köhler-Allee 51
79110 Freiburg i. Br., Germany
polian@informatik.uni-freiburg.de
engelke@informatik.uni-freiburg.de
\textsuperscript{2}Intel Corp.
Austin, TX 78746, USA
sandip.kundu@intel.com
\textsuperscript{3}LIRMM – UMII
161 Rue Ada
34392 Montpellier, France
galliere@lirmm.fr
renovell@lirmm.fr

Abstract

We present three resistive bridging fault models valid for different CMOS technologies. The models are partitioned into a general framework (which is shared by all three models) and a technology-specific part. The first model is based on Shockley equations and is valid for conventional but not deep submicron CMOS. The second model is obtained by fitting SPICE data. The third resistive bridging fault model uses Berkeley Predictive Technology Model and BSIM4; it is valid for CMOS technologies with feature sizes of 90nm and below, accurately describing non-trivial electrical behavior in that technologies. Experimental results for ISCAS circuits show that the test patterns obtained for the Shockley model are still valid for the Fitted model, but lead to coverage loss under the Predictive model.

Keywords: Resistive bridging faults, Deep submicron technology modeling

1 Introduction

Resistive bridging faults (RBF) are accurate representations of resistive short defects, which are increasingly important for deep submicron and nanoscale CMOS technologies. The first systematic studies on these defects were published in the beginning of the 1990s [1, 2]. Since the bridge resistance is a continuous parameter that is not known in advance, handling of such faults is not a trivial task. Some authors approached the problem by picking a fixed bridge resistance value [3] or by applying a locally exhaustive test set to the bridge site [4].

However, an approach based on interval algebra [5, 6] allowed treating the whole continuum of bridge resistance values \( R_{sh} \) from zero Ohm to infinity by handling a finite number of discrete intervals. The key observation which enables this efficient method is the following: a resistive short defect changes the voltages on the bridged lines from zero Volt (logic-0) or \( V_{DD} \) (logic-1) to some intermediate values; and these intermediate voltages will be different for different \( R_{sh} \) values. In order to describe the behavior of a defective circuit, however, it is not necessary to consider the exact voltage on the bridged lines (continuous information). It is rather relevant whether these voltages are interpreted as logic-0 or logic-1 by subsequent logic gates (binary, i.e. discrete information). For instance, if a node which is driven to a logic-1 is bridged with a node driven to a logic-0, the voltage on the first node will be close to \( V_{DD} \) for very high \( R_{sh} \) values and lowest for \( R_{sh} = 0 \Omega \). A gate fed by that line may interpret a faulty value (logic-0) for \( R_{sh} = 0 \Omega \) and the correct value (logic-1) for some higher \( R_{sh} \) values, depending on the logic threshold of the gate. In this case, there is some critical resistance value \( R_{crit} \), such that the faulty value is interpreted for \( R_{sh} < R_{crit} \) and the correct value is interpreted for \( R_{sh} > R_{crit} \). The output of the gate is determined by whether \( R_{sh} \) belongs to the interval \([0, R_{crit}]\) or not, rather than on the actual value of \( R_{sh} \).

By propagating the intervals to the outputs, the bridge resistance intervals for which the fault is detected are determined [5, 6]. It has been demonstrated that during this process the intervals may become discontinuous unions of intervals [7]. Non-trivial behavior has been shown for sequential circuits [8] and for feedback faults [9]. Nevertheless, the propagation process is still discrete. The interval arriving on a circuit output is called analogue detectability interval (ADI). By calculating the ADI for each output and each applied test vector, the bridge resistance range for which the fault is detected by a test set is determined. This range is called C-ADI (covered ADI). It is possible to calculate the global ADI (G-ADI), which includes all the bridge resistances for which the fault could have been detected. A short defect with \( R_{sh} \not\in G-ADI \) is redundant. A test set can be graded (fault simulated) by relating C-ADI to G-ADI. Several RBF simulators have been proposed [8, 10, 11, 12, 13], some of which employ approximate methods for calculating G-ADI. Also several RBF ATPG tools have been reported [4, 12, 13, 14, 15].

The critical resistance calculation is the only part of the interval-based approach that is technology-dependent. Earlier works employed critical resistance calculation procedure based on Shockley’s transistor equations [5, 16, 6, 8] or stor-
ing of the SPICE simulation results in look-up-tables [10]. While Shockley equations (developed in the 1940s) [17] are not necessarily valid for modern deep submicron manufacturing technologies, the look-up-table approach tends to be less flexible.

In this paper we describe the general electrical model (framework) of local circuit behavior at the bridge defect site. It relates the voltage potentials on the bridge nodes to the current flowing through the network. The critical resistance can be calculated as a function of the logic threshold of a succeeding gate. The general framework is based on $I-V$ output characteristics of the PMOSFET and NMOSFET CMOS generations. The third model is based on Berkeley Predictive Technology Model (BPTM, which is provided by the Device Group at UC Berkeley) [18] in connection with Berkeley Short-channel IGFET Model 4 (BSIM4),\(^1\) which is valid for 90nm technologies and is used for predicting the transistor behavior in future 65nm and 45nm technologies. BPTM/BSIM4 accounts for the many non-trivial electrical phenomena in these nanoscale technologies, including Non-Uniform Lateral Doping (NULD), Narrow Width Effect, Short-Channel Effect, Drain-Induced Barrier Lowering (DIBL), Drain-Induced Threshold Shift (DITS) and Bulk Charge Effect. Hence, the three models describe past, present and future technologies; nevertheless they share the same basic framework.

We demonstrate three instantiations of the general framework. The first model uses Shockley equations and is valid for conventional (not deep submicron) technology (it corresponds to the model used in [5, 6]). The second model is obtained by fitting SPICE data and works well for current CMOS generations. The third model is based on Berkeley Predictive Technology Model (BPTM, which is provided by the Device Group at UC Berkeley) [18] in connection with Berkeley Short-channel IGFET Model 4 (BSIM4),\(^1\) which is valid for 90nm technologies and is used for predicting the transistor behavior in future 65nm and 45nm technologies. BPTM/BSIM4 accounts for the many non-trivial electrical phenomena in these nanoscale technologies, including Non-Uniform Lateral Doping (NULD), Narrow Width Effect, Short-Channel Effect, Drain-Induced Barrier Lowering (DIBL), Drain-Induced Threshold Shift (DITS) and Bulk Charge Effect. Hence, the three models describe past, present and future technologies; nevertheless they share the same basic framework.

We integrated all three models into the fault simulator [8] and the ATPG [15] and applied these tools to ISCAS benchmarks. Of particular interest was the validity of test vectors generated for the old Shockley model for newer models. This is related to the question whether new test generation is required after design shrinking.

The remainder of the paper is organized as follows: the general framework is described in the next section. Technology-specific $I-V$ models are discussed in Section 3. Experimental results are reported in Section 4. Section 5 concludes the paper.

2 General Framework

Figure 1 (left) shows two nodes $n_1$ and $n_0$ bridged by a resistive short defect with resistance $R_{sh}$. We assume that gate $D_1$ drives the logic-1 value on $n_1$ and gate $D_0$ drives logic-0 on $n_0$, respectively. $n_1$ is connected to the inputs of gates $A_1, A_2$ and $A_3$, with logic gate thresholds $Th_{A1}, Th_{A2}$ and $Th_{A3}$, respectively; $n_0$ drives gates $B_1$ and $B_2$ with the respective thresholds $Th_{B1}$ and $Th_{B2}$. Figure 1 (right) shows possible voltage characteristics of the voltage $V_{n1}$ on $n_1$ and $V_{n0}$ on $n_0$ as a function of the bridge resistance $R_{sh}$. For $R_{sh} = 0$ (non-resistive bridge), $V_{n1}$ and $V_{n0}$ assume the same intermediate value (note that this value corresponds to the voltage considered by Voting [19] and Biased Voting [20] models). With increasing bridge resistance, the voltages on $n_1$ and $n_0$ diverge; for an infinite $R_{sh}$, $V_{n1}$ would equal $V_{DD}$ and $V_{n0}$ would equal 0V, which are the fault-free values.

Logic gate thresholds $Th_{A1}$ through $Th_{B2}$ are independent from bridge resistance. One possible distribution of gate thresholds is depicted in Figure 1 (right) as horizontal lines. Critical resistances are determined from the intersections of these lines with the $R_{sh}$-$V$ characteristics. For instance, $Th_{A1}$ is the critical resistance for gate $A_1$, i.e., if the bridge resistance is below $R_{A1}$, then gate $A_1$ will interpret the (faulty) logic value of 0, while for $R_{sh} > R_{A1}$ it will interpret the fault-free value. Note that there is no critical resistance for $Th_{A2}$, as the horizontal line has no intersection with the voltage characteristic $V_{n1}$. This means that gate $A_2$ will interpret the (fault-free) logic-1 value irrespective of the bridge resistance and no faulty effect will be propagated through gate $A_2$ for this test vector. Similarly, there is a critical resistance for gate $B_1$ but none for gate $B_2$. We are interested in determining the critical resistance from the gate threshold analytically.

Figure 2 shows some electrical details of the example circuit. (For simplicity, only one gate is driven by $n_1$ and $n_0$, respectively.) Gates $D_1$ and $D_0$ internally consist of a $p$ transistor network and an $n$ transistor network. Since $D_1$ drives a logic-1 on $n_1$, its $p$ network establishes a connection to $V_{DD}$ while its $n$ network is disabled. Similarly, the $n$ network of gate $D_0$ establishes a connection to ground. We can ignore the $n$ network of gate $D_1$ and the $p$ network of gate $D_0$ for further analysis, as these networks are switched off. Due to the resistive short defect, there is a conducting path

\[^1\] We used BSIM4.4.0 released in March 2004. It is available from the URL http://www-device.eecs.berkeley.edu/~bsim3/bsim4.html.
This section describes calculation of \( I-V_{ds} \) characteristics for \( p \) and \( n \) transistor networks. We present three different models: Shockley model, Fitted model and Predictive model. The Shockley model is based on conventional transistor equations that ignore some of the electrical phenomena in deep submicron technologies. The Fitted model uses equations with free variables that are fitted in order to match actual SPICE data. Finally, the Predictive model is again fully analytical and employs BSIM4 equations. Note that if a different \( I-V_{ds} \) model is in use (e.g. a company-internal solution), it can be integrated into the general framework in a straightforward way.

In the following, we describe the three models. We concentrate on the \( n \) transistor network, as the handling of the \( p \) network is largely identical. We start with the simple case in which the outputs of two inverters are bridged. In this case, both the \( p \) and \( n \) network consist of just one MOSFET, respectively. This analysis is also valid for a network of parallel transistors (e.g. the \( p \) network of a NAND gate and the \( n \) network of a NOR gate). In this case, the equations for the inverter case must be updated by simply replacing \( W_{NOT,n} \) by \( k \cdot W_{NOR,n} \), where \( W_{NOT,n} \) is the width of the \( n \) transistor in the inverter, \( W_{NOR,n} \) is the width of each of the \( n \) transistors in the NOR gate, and \( k \) is the number of active \( n \) transistors under the given test vector. For serial connections of transistors (e.g. the \( p \) network of a NOR gate and the \( n \) network of a NAND gate), no such simple mapping exists. Figure 3 shows the two activated transistors in the two-inverter case (left) and the serial \( n \) network in a NAND2 gate bridged with an inverter (right).

### 3.1 Shockley Model

Shockley’s conventional \( I-V_{ds} \) equation is given as

\[
I_{ds,n}(V_{ds,n}) = \mu_n C_{ox} \frac{W}{L} (V_{gs,n} - V_{tn0}) V_{ds,n} \left( V_{ds,n} + \frac{V_{ds,n}^2}{2} \right)
\]

where \( \mu_n \) is the mobility, \( C_{ox} \) the oxide capacity per area unit, \( W \) the channel width, \( L \) the channel length and \( V_{tn0} \) the zero bias threshold voltage for an NFET. Since the transistor is on, the gate voltage must be \( V_{DD} \) (logic-1). As the source terminal of the \( n \) transistor is connected to ground, the gate-source voltage \( V_{gs,n} \) equals \( V_{DD} \) (see the left part of Figure 3).

The inverse equation \( I^{-1} \) (which yields voltage \( V_{ds,n} \) for a given current \( I_0 \)) is obtained by solving the quadratic equation \( I_0 = I_{ds,n}(V_{ds,n}) \). As described in the previous section, the critical resistance can be calculated using the \( I-V_{ds} \) equation and its inverse. Closed-form formulae for critical resistance in the Shockley model can be found in [5, 6, 8]. The equation for the critical resistance of a gate that succeeds the bridged line driven by the \( n \) network is given as

\[
R_{crit,n} = \frac{|V_{tpo}|-Th+\sqrt{(V_{DD}-|V_{tpo}|)^2-\frac{2I_{ds,n}(Th)}{C_{ox}\mu_p w_p L_p}}}{I_{ds,n}(Th)}
\]

where \( Th \) is the logic threshold of the gate, \( V_{tpo}, \mu_p, w_p \) and \( L_p \) are the zero bias threshold voltage, the mobility, the channel width and the channel length for a PFET, respectively, and \( I_{ds,n}(Th) \) is calculated using Eq. (2). This equation is valid for a single transistor and a parallel transistor network (using an appropriate value for the width). For a serial network, equivalent transistor calculation is performed, which accounts for the body-bias effect. See [5] for details.
3.2 Fitted model

The Fitted model uses the following \( I-V_{ds} \) equation for the \( n \) transistor network:

\[
I_{ds,n}(V_{ds,n}) = A_n W_n \left( (V_{DD} - B_n)V_{ds,n} - \frac{V_{ds,n}^2}{2} \right) \tag{4}
\]

where \( A_n \) and \( B_n \) are obtained by fitting SPICE results. There are several sets of \((A_n, B_n)\) parameters for different transistor configurations, which eliminates the need for the equivalent transistor calculation required in the Shockley model. Note that for \( A_n = \mu_n C_{ox}/L_n \) and \( B_n = V_{th0} \), the Fitted model is identical to the Shockley model. We also performed experiments with the Alpha-Power Law model [21], which also involves fitting, but we obtained better results using Eq. (4). The \( R_{crit} \) equation for a gate with logic threshold \( T_h \) driven by the \( n \) transistor network is

\[
R_{crit,n} = \frac{|B_p| - T_h + \sqrt{(V_{DD} - |B_p|)^2 - 2I_{ds,n}(T_h)A_p W_p}}{I_{ds,n}(T_h)} \tag{5}
\]

where \( I_{ds,n}(T_h) \) is calculated using Eq. (4).

Table 1 shows critical resistance values calculated by HSPICE, the Shockley model and the Fitted model for different channel widths \( W_p \) and \( W_n \) (more results are available but omitted due to limited space). We consider the bridge between the outputs of two inverters and the bridge between two NAND gates (in the latter case we distinguish between one and two active \( p \) transistors): Note that the critical resistance values are for the gate \( A \) driven by the \( p \) transistor network for the two-inverter-bridge and for the gate \( B \) driven by the \( n \) transistor network for the two-NAND-bridge. It is clearly seen that the values computed by the Fitted model track the HSPICE values much better than the values obtained by the Shockley model. Even for the simple two-inverter case, the deviation of the Shockley model from the HSPICE reference is up to 5%; for the two-NAND case it sometimes exceeds 35%. In contrast, the Fitted model never has a deviation larger than 0.4%.

3.3 Predictive model

3.3.1 Single transistor or parallel network

According to BPTM/BSIM4, the \( I-V_{ds} \) characteristic in the relevant \( V_{ds} \) region is described by the equation

\[
I_{ds,n}(V_{ds,n}) = \frac{W_{ch0,n}Q_{ch0,n} V_{ds,n}}{L_{ch0,n}} \left( 1 - \frac{V_{ds,n}}{V_{th0,n}} \right) \tag{6}
\]

where \( \mu_{eff,n} \) is the effective mobility, \( Q_{ch0,n} \) is the channel charge density and \( E_{sat,n} \) is the critical electrical field at which the carrier velocity becomes saturated. \( V_{th0} \) is defined as \((V_{gateff,n} + 2\nu l)/A_{bulk,n} \), where \( V_{gateff,n} \) is the effective \( V_{gs,n} - V_{th,n} \), \( \nu_l \) is the thermal voltage \( (k_B T/q) \) and \( A_{bulk} \) models the bulk charge effect. All these parameters can be calculated from over 100 process parameter values. (Example values for 65nm and 45nm are given on the BPTM homepage [18]). In two-inverter case, \( V_{bs} \) is set to 0 and \( V_{gs} \) is set to \( V_{DD} \) for NMOS and \( -V_{DD} \) for PMOS (see the left part of Figure 3). For parallel networks, the width \( W_n \) has to be set appropriately, as mentioned above.

Table 1: Results of Shockley and Fitted models compared to HSPICE; \( L_p = L_n = 0.35\mu m, V_{DD} = 3.3V \), logic gate threshold = \( V_{DD}/2 \)

<table>
<thead>
<tr>
<th></th>
<th>2-inverter bridge</th>
<th>2-NAND bridge, 1 active PFET</th>
<th>2-NAND bridge, 2 active PFETs</th>
</tr>
</thead>
<tbody>
<tr>
<td>( W_p, \mu m )</td>
<td>2 4 8</td>
<td>2 4 8</td>
<td>2 4 8</td>
</tr>
<tr>
<td>( W_n, \mu m )</td>
<td>1 2 4</td>
<td>1 2 4</td>
<td>1 2 4</td>
</tr>
<tr>
<td>( R_{crit,n,\Omega} )</td>
<td>2422 1159 567</td>
<td>1686 838 418</td>
<td>4250 2064 1018</td>
</tr>
<tr>
<td>Shockley</td>
<td>2304 1152 576</td>
<td>2305 1152 576</td>
<td>4660 2330 1165</td>
</tr>
<tr>
<td>Fitted</td>
<td>2415 1157 566</td>
<td>1685 838 419</td>
<td>4266 2072 1019</td>
</tr>
</tbody>
</table>

Note that, as in the Shockley model and the Fitted model, the inverse function is obtained by solving a quadratic equation. In this way, BPTM/BSIM4 is relatively easy to integrate into the general critical resistance calculation framework. However, some of the parameters in Eq. (6) exhibit a second-order dependency on \( V_{ds,n} \). This is because the transistor threshold depends on \( V_{ds,n} \) through its DBL (Drain-Induced Barrier Lowering) and DITS (Drain-Induced Threshold Shift) terms. The transistor threshold is required for the calculation of \( V_{gateff,n} \) and consequently into \( Q_{ch0,n}, \mu_{eff,n} \) and \( V_{th0} \). When calculating \( V_{ds,n} = I^{-1}(I_0) \), \( V_{ds,n} \) is naturally unknown in advance. In order to account for this dependency, we applied the following iterative method: we calculated the parameters of Eq. (6) for \( V_{ds,n} = 0 \) and computed \( V_{ds} = I^{-1}(I_0) \). Then, we re-calculated the parameters using that \( V_{ds} \) value and re-computed \( V_{ds,n} = I^{-1}(I_0) \) using new parameters. We found that the maximal deviation from the old \( V_{ds,n} \) value was maximally 2mV, and that changes in current were under \( 10^{-14} \mu A \). Hence, this iteration can be omitted when optimizing for speed.

3.3.2 Serial network

Figure 3 shows the serial \( n \) transistor network of a NAND2 gate. We will first describe the calculation of \( I^{-1} \) (obtaining the voltage \( V_{n0} \) from the current \( I_0 \)), and then the calculation of the current from the voltage.

We call the voltage between the two NFETs \( V^* \). We can determine it as \( I^{-1}(I_0) \) using essentially the same function that we used in the two-inverter case. To compute \( V_{n0} \) from \( V^* \), we first set \( V_{bs} \) to \( -V^* \) and \( V_{gs} \) to \( V_{DD} - V^* \) and re-extract the parameters. Then, we apply \( I^{-1} \) once again and obtain some \( V' \). Note that \( V' < V^* \). Finally, \( V_{n0} \) is given as \( V^* + V' \).

In order to calculate \( I_0 \) from \( V_{n0} \), we use an iterative algorithm. We denote the current flowing through the lower NFET (connected to ground) as \( I_1 \) and the current flowing through the upper NFET as \( I_2 \). Obviously, \( I_1 \) and \( I_2 \) must be equal. Our algorithm modifies \( I_1 \) and \( I_2 \) until they become nearly the same. The algorithm uses the voltage \( V^* \) as the iteration variable.
\( V^* := V_{n0}/2; \)

(2) Calculate \( I_1 \) (using \( I_{ds} \) formula for one transistor with \( V_{gs,n} = V^*, V_{bs,n} = 0, V_{gs,n} = V_{DD} \)).

(3) Calculate \( I_2 \) (using \( V_{ds,n} = V_{n0} - V^*, V_{bs,n} = -V^* \), \( V_{gs,n} = V_{DD} - V^* \)).

(4) \( I^* := (I_1 + I_2)/2; \)

(5) if \( |I_1 - I_2| < \varepsilon \) return \( I^* \);

(6) Calculate \( V^* \) from \( I^* \) (using \( V_{bs,n} = 0, V_{gs,n} = V_{DD} \)).

(7) goto (2);

Note that the algorithm converges because \( V_{n0} - V^* \) always holds after the first iteration. For \( \varepsilon = 1 \mu A \), the algorithm always terminated after 6 iterations.

## 4 Experimental Results

We integrated the support of the Fitted and the Predictive models into the fault simulator [8] and the automatic test pattern generator [15], which already supported the Shockley model. We used HSPICE with a BSIM3v3 0.35 \( \mu m \) model card to obtain the values \( A_p, A_n, B_p \) and \( B_n \) for the Fitted model. For the Predictive model we used the 65nm BSIM4 model card available at the BPTM URL [18]. We applied ATPG and fault simulation to ISCAS 85 and combinational parts of ISCAS 89 (indicated as ‘cs’) circuits. The fault set consisted of 10,000 randomly selected non-feedback resistive bridging faults (where available). All measurements were performed on a 2GHz Linux machine with 1 GB RAM.

ATPG results are reported in Table 2. The table contains the name of the circuit, the number of considered faults, and the number of test vectors generated for the three technology-specific models. It can be seen that the number of generated test vectors varies across the models with no clear trend; it can not be claimed that the Fitted or the Predictive model are “more complex” from the ATPG’s point of view, despite the non-trivial circuit behavior that they account for.

Note that, although the number of RBF test vectors is typically higher than the number of stuck-at test vectors generated by state-of-the-art ATPG systems [22, 23], the number of considered faults is also larger. It is demonstrated in [15] that the fault efficiency (i.e. the average number of faults detected by one test pattern) is higher for our tool than for other RBF ATPGs [4, 3] as well as stuck-at ATPGs.

We used our fault simulator to study the implications of the support of the Fitted and the Predictive models into the fault simulator [8] and the automatic test pattern generator [15], which already supported the Shockley model. We used HSPICE with a BSIM3v3 0.35 \( \mu m \) model card to obtain the values \( A_p, A_n, B_p \) and \( B_n \) for the Fitted model. For the Predictive model we used the 65nm BSIM4 model card available at the BPTM URL [18]. We applied ATPG and fault simulation to ISCAS 85 and combinational parts of ISCAS 89 (indicated as ‘cs’) circuits. The fault set consisted of 10,000 randomly selected non-feedback resistive bridging faults (where available). All measurements were performed on a 2GHz Linux machine with 1 GB RAM.

### Table 2: ATPG results for the three models

<table>
<thead>
<tr>
<th>Circuit</th>
<th>#faults</th>
<th>Shockley</th>
<th>Fitted</th>
<th>Predictive</th>
</tr>
</thead>
<tbody>
<tr>
<td>c0095</td>
<td>77</td>
<td>18</td>
<td>20</td>
<td>16</td>
</tr>
<tr>
<td>c0432</td>
<td>5253</td>
<td>732</td>
<td>659</td>
<td>489</td>
</tr>
<tr>
<td>c0880</td>
<td>10000</td>
<td>745</td>
<td>816</td>
<td>341</td>
</tr>
<tr>
<td>c1355</td>
<td>10000</td>
<td>178</td>
<td>240</td>
<td>243</td>
</tr>
<tr>
<td>c1908</td>
<td>10000</td>
<td>267</td>
<td>329</td>
<td>391</td>
</tr>
<tr>
<td>c2670</td>
<td>10000</td>
<td>366</td>
<td>371</td>
<td>459</td>
</tr>
<tr>
<td>c3540</td>
<td>10000</td>
<td>489</td>
<td>464</td>
<td>576</td>
</tr>
<tr>
<td>c5315</td>
<td>10000</td>
<td>384</td>
<td>382</td>
<td>419</td>
</tr>
<tr>
<td>c7552</td>
<td>10000</td>
<td>357</td>
<td>377</td>
<td>388</td>
</tr>
<tr>
<td>cs00208</td>
<td>3986</td>
<td>124</td>
<td>120</td>
<td>145</td>
</tr>
<tr>
<td>cs00298</td>
<td>4468</td>
<td>125</td>
<td>158</td>
<td>141</td>
</tr>
<tr>
<td>cs00349</td>
<td>7881</td>
<td>197</td>
<td>172</td>
<td>149</td>
</tr>
<tr>
<td>cs00382</td>
<td>7809</td>
<td>255</td>
<td>253</td>
<td>198</td>
</tr>
<tr>
<td>cs00386</td>
<td>9384</td>
<td>77</td>
<td>72</td>
<td>164</td>
</tr>
<tr>
<td>cs00400</td>
<td>8290</td>
<td>245</td>
<td>276</td>
<td>205</td>
</tr>
<tr>
<td>cs00420</td>
<td>10000</td>
<td>317</td>
<td>305</td>
<td>405</td>
</tr>
<tr>
<td>cs00444</td>
<td>10000</td>
<td>312</td>
<td>320</td>
<td>238</td>
</tr>
<tr>
<td>cs00510</td>
<td>10000</td>
<td>320</td>
<td>334</td>
<td>263</td>
</tr>
<tr>
<td>cs00526</td>
<td>10000</td>
<td>372</td>
<td>432</td>
<td>264</td>
</tr>
<tr>
<td>cs00641</td>
<td>10000</td>
<td>304</td>
<td>333</td>
<td>352</td>
</tr>
<tr>
<td>cs00713</td>
<td>10000</td>
<td>332</td>
<td>364</td>
<td>324</td>
</tr>
<tr>
<td>cs00820</td>
<td>10000</td>
<td>454</td>
<td>481</td>
<td>341</td>
</tr>
<tr>
<td>cs00832</td>
<td>10000</td>
<td>445</td>
<td>489</td>
<td>350</td>
</tr>
<tr>
<td>cs00838</td>
<td>10000</td>
<td>473</td>
<td>511</td>
<td>653</td>
</tr>
<tr>
<td>cs00953</td>
<td>10000</td>
<td>422</td>
<td>429</td>
<td>344</td>
</tr>
<tr>
<td>cs01238</td>
<td>10000</td>
<td>508</td>
<td>524</td>
<td>583</td>
</tr>
<tr>
<td>cs01423</td>
<td>10000</td>
<td>514</td>
<td>536</td>
<td>418</td>
</tr>
<tr>
<td>cs01488</td>
<td>10000</td>
<td>234</td>
<td>230</td>
<td>254</td>
</tr>
<tr>
<td>cs01494</td>
<td>10000</td>
<td>225</td>
<td>221</td>
<td>247</td>
</tr>
<tr>
<td>cs05378</td>
<td>10000</td>
<td>824</td>
<td>846</td>
<td>650</td>
</tr>
<tr>
<td>cs09234</td>
<td>10000</td>
<td>904</td>
<td>904</td>
<td>736</td>
</tr>
<tr>
<td>cs1196</td>
<td>10000</td>
<td>427</td>
<td>429</td>
<td>447</td>
</tr>
<tr>
<td>cs13207</td>
<td>10000</td>
<td>1228</td>
<td>1192</td>
<td>879</td>
</tr>
<tr>
<td>cs15850</td>
<td>10000</td>
<td>1060</td>
<td>1027</td>
<td>757</td>
</tr>
<tr>
<td>cs344</td>
<td>10000</td>
<td>143</td>
<td>138</td>
<td>134</td>
</tr>
<tr>
<td>cs35932</td>
<td>10000</td>
<td>516</td>
<td>726</td>
<td>607</td>
</tr>
<tr>
<td>cs38417</td>
<td>10000</td>
<td>1178</td>
<td>1114</td>
<td>732</td>
</tr>
<tr>
<td>cs38584</td>
<td>10000</td>
<td>1822</td>
<td>1883</td>
<td>1197</td>
</tr>
</tbody>
</table>

Note that the algorithm converges because \( V_{n0} - V^* \) always holds after the first iteration. For \( \varepsilon = 1 \mu A \), the algorithm always terminated after 6 iterations.


circuit | fitted | pred. | circuit | fitted | pred.
-------|-------|-------|-------|-------|-------
c0095   | 99.98 | 99.76 | c00526 | 99.99 | 99.84 |
c0432   | 100.00| 99.93 | c00641 | 100.00| 98.86 |
c0880   | 100.00| 99.92 | c00713 | 100.00| 97.87 |
c1355   | 99.99 | 97.21 | c00820 | 99.96 | 99.49 |
c1908   | 99.98 | 99.24 | c00832 | 99.93 | 99.32 |
c2670   | 99.99 | 99.25 | c00838 | 100.00| 97.43 |
c3540   | 99.99 | 98.62 | c00953 | 100.00| 99.87 |
c5315   | 100.00| 99.77 | c01238 | 99.98 | 97.69 |
c7552   | 99.99 | 99.85 | c01423 | 100.00| 99.88 |
c00208  | 100.00| 97.48 | c01488 | 100.00| 98.29 |
c00298  | 99.98 | 99.69 | c01494 | 100.00| 98.04 |
c00349  | 100.00| 97.96 | c05378 | 100.00| 99.82 |
c00382  | 100.00| 99.91 | c09234 | 99.97 | 98.73 |
c00386  | 100.00| 91.88 | c13207 | 99.98 | 99.33 |
c00400  | 100.00| 99.91 | c15850 | 99.99 | 99.44 |
c00420  | 100.00| 97.36 | c35932 | 99.98 | 99.02 |
c00444  | 99.99 | 99.88 | c38417 | 99.99 | 99.77 |
c00510  | 100.00| 99.83 | c38584 | 99.89 | 98.56 |

Table 3: Fault coverage of Shockley-ATPG vectors for Fitted and Predictive models

5 Conclusions

We presented three electrical-level models for calculating the critical resistance, which is required for accurate handling of resistive bridging faults. The models have a common general framework and technology-specific parts based on $I-V_{ds}$ characteristics. We proposed a model for conventional technology described by Shockley equations, one valid for current submicron designs based on fitting SPICE data, and one intended for future nanoscale CMOS employing predictive transistor modeling with BSIM4. We integrated the support for all three models into our simulator and ATPG tools. It turned out that the test complexity, expressed in the number of vectors required for 100% fault coverage, does not show a trend towards one of the models. Moreover, we investigated the efficiency of test vectors generated for the Shockley model in detecting defects in submicron technologies using the two other models, in order to estimate the need for a new ATPG run after a design shrinking. Very high coverages are still obtained for all circuits under the Fitted model, but only for some circuits using the Predictive model, suggesting that a re-evaluation of fault coverage is required at least.

The models accurately reflect non-trivial electrical behavior in nanoscale technologies. However they concentrate on the static phenomena. Our further research will concentrate on modeling dynamic effects of the resistive short and open defects.

Acknowledgment

We are thankful to Prof. P. Maurine of LIRMM for his insights on the fitting approach, and to Prof. B. Nikolic of UC Berkeley for his hint on BPTM/BSIM4. Sandip Kundu contributed to this work while visiting the University of Freiburg as a guest professor.

6 References