A Transistor-Level Stochastic Approach for Evaluating the Reliability of Digital Nanometric CMOS Circuits

Hao Chen
ECE Department
University of Alberta
Edmonton, Canada
hc5@ualberta.ca

Jie Han
ECE Department
University of Alberta
Edmonton, Canada
jhan8@ualberta.ca

Fabrizio Lombardi
ECE Department
Northeastern University
Boston, MA, U.S.A
lombardi@ece.neu.edu

***Abstract*—Over the last few decades, most quantitative measures of VLSI performance have improved by many orders of magnitude; this has been achieved by the unabated scaling of the sizes of MOSFETs. However, scaling also exacerbates noise and reliability issues, thus posing new challenges in circuit design. Reliability becomes a major concern due to many and often correlated factors, such as parameter variations and soft errors. Existing reliability evaluation tools focus on algorithmic development at the logic level that usually uses a constant error rate for gate failure and thus leads to approximations in the assessment of a VLSI circuit. This paper proposes a more accurate and scalable approach that utilizes a transistor-level stochastic analysis for digital fault modeling. It accounts for very detailed measures, including the probability of failure of individual transistors, the topology of logic gates, timing sequences and the applied input vectors. Simulation results are provided to demonstrate both the efficiency and the accuracy of the proposed approach.**

Keywords- Reliability evaluation, Stochastic computation, Soft errors, Logic circuits, Nanometric CMOS

I. INTRODUCTION

The reliable operation of integrated circuits has become a major concern as at nanometric feature sizes, manufacturing defects, transient faults, parameter variations and aging effects are very compelling challenges [1]. As CMOS technology scales down, the evaluation of the reliability of logic circuits has been pursued using various computational methodologies [2 – 5]. While an analytical approach can handle the assessment of small circuits at no loss of accuracy, the exact and efficient computation of the reliability remains difficult to achieve for large circuits. Therefore, a compromise is usually made for the accuracy of the evaluation versus the computational complexity and memory requirements. Such compromise must consider signal correlation (as introduced by reconvergent fanouts in combinational circuits or feedbacks in sequential circuits). Simulation has also been used as an alternative when an analytical approach becomes intractable. In a simulation-based approach, experimental data is gathered to characterize the behavior of a circuit by randomly sampling its activity. As an example, Monte Carlo (MC) simulation has been widely used when an analytical approach is not available or easy to implement. A disadvantage of simulation is that numerous pseudo-random numbers must be generated; therefore, a large number of simulation runs must be executed to reach a stable output, so making the evaluation of large circuits a very time-consuming process. Recently, some new approaches that take advantages from both analytical as well as simulation-based methods have been developed [6] [7]. These approaches employ random binary sequences to gather the probabilistic information in a circuit and provide highly accurate and efficient analyses of circuit reliability.

As existing reliability evaluation approaches focus on algorithmic development aiming for both high accuracy and low computational complexity, they rely on a logic-level analysis so that a fast evaluation can be provided at the early stage of a logic design process. However, they exhibit a major shortcoming: a constant probability of gate failure is usually assumed in a gate-level analysis, which actually has no physical basis for its applicability (as faults and defects usually affect individual devices such as transistors [8] [9]). For example, process variations, due to random dopant fluctuation or manufacturing imperfections in the CMOS fabrication process, have prominently emerged to impact performance and degrade the reliability of electronic circuits. The physical characteristics of devices have subsequently resulted in probabilistic circuit behaviors that manifest as a switching error of a transistor [8]. Manufacturing defects can also result in stuck-at faults in transistors [9]. The error probability also depends on the topology of a logic gate as well as its input vector. Moreover, the signal sequence needs to be considered. For example, if both the pull-up and pull-down networks are OFF in a CMOS gate, then the gate output is dependent on the previous output state (assuming no leakage).

As comprehensive circuit analysis (dealing with electrical and timing information such as in Monte Carlo SPICE simulation) is thought to increase the computational complexity, thus further complicating the reliability assessment problem, a transistor-level analysis could circumvent these disadvantages and therefore provide the basis for a more accurate analysis. A design automation tool that considers reliability at the transistor level has recently been proposed for estimating the reliability of CMOS logic gates, as well as that of some small circuits such as full adders [10]. However, it has not been

applied to the analysis of large circuits since it still incurs an excessive complexity in computation. In this paper, a transistor-level stochastic analysis is proposed for the accurate and efficient reliability evaluation of digital circuits. Stochastic models are initially developed for transistors by extending the probabilistic analysis of gate-level SCMs. Logic gates are then modeled by considering sequential as well as combinational effects, such as timing sequences, gate topology and inputs to transistor operations. Since the probability is encoded into stochastic binary streams and signal correlation is carried on the bit-wise dependencies of the streams, the proposed approach is scalable for use in the analysis of large circuits.

This paper is organized as follows. Section II reviews stochastic computation and its application on gate-level analysis. Section III presents the stochastic modeling for a transistor-level analysis and Section IV presents the logic gate models. Section V outlines the circuit analysis approach and Section VI reports simulation results. Section VII concludes the paper.

II. Stochastic Computation for Gate-Level Reliability Analysis

In stochastic computation, real numbers are represented by *random binary bit streams* and information is carried on the statistics of the binary streams [11]. Stochastic computation offers advantages such as computational simplicity, fault tolerance and high speed [12-16]. Its effectiveness has been shown in several applications including stochastic decoding [17], neural computation [12] and fault-tolerant computing [13, 14]. In stochastic logic, signal probabilities are encoded into binary bit streams, i.e., serially in the time domain. A specific probability is usually represented by the proportion of the mean number of 1's in a uniformly distributed random bit streams. Figure 1 shows an inverter and an XOR gate performing encoded stochastic computation. As Boolean operations can be mapped to arithmetic operations, the inverter probabilistically implements the complement operation of $P(\mathrm{Y}=1)=1-P(\mathrm{X}=1)$ and the XOR implements the function shown in Figure 1(b). Note that in Figure 1, a sequence length of 10 bits is used for illustration purposes, a larger sequence length is usually needed in practice.

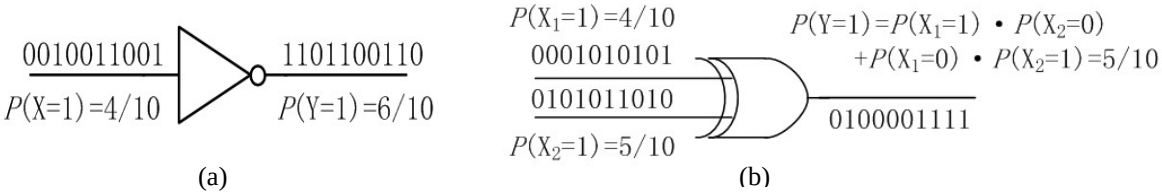


Figure 1. Stochastic computation by (a) an inverter and (b) an XOR gate.

Recently, a stochastic approach has been proposed for the reliability analysis of logic circuits [7]. A stochastic computational model (SCM) is used to evaluate an unreliable logic gate with a constant gate error rate. For example, a gate affected by a bit-flipping error can be modeled using the stochastic XOR as

$$P(output=1)=\mathrm{XOR}_{\mathrm{sto}}(\mathrm{p},\varepsilon)=\mathrm{p}(1-\varepsilon)+(1-\mathrm{p})\varepsilon, \tag{1}$$

where $\varepsilon=\mathrm{P}(\text{gate faulty})$ and $\mathrm{p}=\mathrm{P}(\text{output}=1|\text{gate not faulty})$. Therefore, an unreliable AND gate affected by a flipping error has an output probability given by

$$P(\mathrm{Z}=1)=\big(1-P(\mathrm{X}_1=1)\cdot P(\mathrm{X}_2=1)\big)\cdot\varepsilon+P(\mathrm{X}_1=1)\cdot P(\mathrm{X}_2=1)\cdot(1-\varepsilon). \tag{2}$$

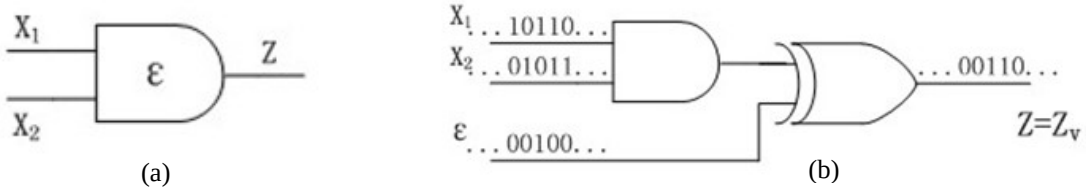


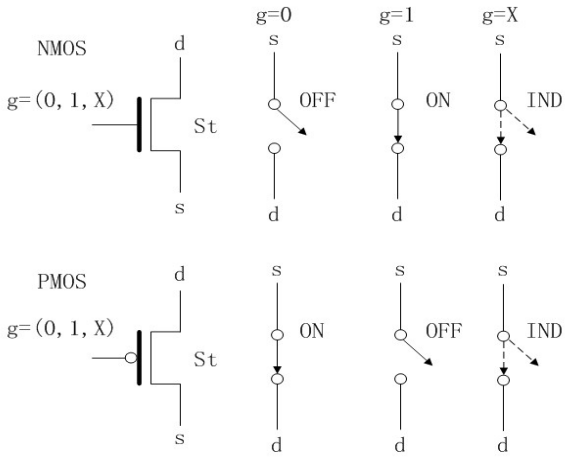
Figure 2. (a) An unreliable AND gate and (b) its SCM implementation for flipping errors [7].

As shown in Figure 2, an unreliable AND gate (Figure 2(a)) is implemented by an SCM using an XOR gate (Figure 2(c)) [7]. When applied to replace unreliable gates in a circuit, the SCMs provide an accurate and efficient approach for evaluating the reliability of logic circuits. Since signal dependencies are inherently maintained in the distribution patterns of the random binary bit streams, the SCM approach efficiently handles signal correlation introduced by reconvergent fanouts; therefore, the computational complexity is significantly reduced. Moreover, although precision is limited by the inherent randomness of the binary bit streams used in stochastic computation, the evaluation results are highly accurate and the approach is scalable for use in the analysis of large circuits [7].

III. STOCHASTIC MODELING OF TRANSISTORS

The CMOS transistor is a voltage-controlled current source. In digital design, the transistor is usually considered to operate as a switch (Figure 3). As a switch, the transistor gets its source and drain conducted, if the gate voltage is "high" (for NMOS) or "low" (for PMOS). Thus, the ON/OFF state of a transistor is determined by the applied gate voltage. When transistors are used in a gate and the gate voltage falls off the noise margins, the transistor operates in an indefinite manner, so its state is referred to as "indefinite" or "IND." Hence, there are three operational states in the transistor model used in this paper: ON, OFF and IND (Figure 3(a)). These states are determined by three different gate inputs, i.e., voltage as high, low and outside of the noise margins (corresponding to g as logic "1," logic "0," and "X," respectively, in Figure 3). Mapping between the gate input and the operation of the NMOS and PMOS transistors is summarized in Figure 3(b) and it is given as follows:

- For the NMOS transistor, the input 0 results in OFF; the input 1 results in ON; and the input X results in IND.
- For the PMOS transistor, the input 0 results in ON; the input 1 results in OFF; and the input X results in IND.



(a)

Transistor type / Gate input	NMOS State (St)	PMOS State (St)
$g = 0$	OFF	ON
$g = 1$	ON	OFF
$g = X$	IND	IND

(b)

Figure 3. (a) Transistor model as a probabilistic switch. g : gate terminal; d: drain terminal; s: source terminal; St: state (ON/OFF/IND) of the transistor. (b) Mapping between the gate input and the transistor operations.

Since a transistor may be affected by a transient error, its state could be erroneous. A transistor can therefore be modeled as a probabilistic switch such that the probability distribution of the state (ON, OFF and IND) of the transistor is determined by the input signal probability. Here, this probabilistic switching of the transistor is modeled using the stochastic computational models (SCMs) presented in the previous section. As shown in Figure 4, the gate input is represented by a random bit stream, and so is the switching error probability of the transistor. If the transistor is affected by a flipping error with an error rate $\varepsilon = \mathrm{P}(\text{transistor faulty})$, then the gate input can be considered to be changed by a stochastic XOR as

$$g' = \mathrm{XOR}_{\mathrm{sto}}(\mathrm{g}, \varepsilon) = \mathrm{g} \cdot (1 - \varepsilon) + (1 - \mathrm{g}) \cdot \varepsilon. \tag{3}$$

The newly-generated gate input is then used to determine the state of the transistor (considered now as reliable). This results in a stochastic model for an unreliable NMOS or PMOS transistor as shown in Figure 4. A similar stochastic model can be used to estimate the transistor's behavior when affected by a different type of error. For example, the stuck-ON/OFF fault can be modeled using the following equations:

$$g'_{Stuck-ON} = \mathrm{OR}_{\mathrm{sto}}(\mathrm{g}, \varepsilon) = \mathrm{g} + \varepsilon - \mathrm{g} \cdot \varepsilon = \varepsilon + \mathrm{g} \cdot (1 - \varepsilon) \tag{4}$$

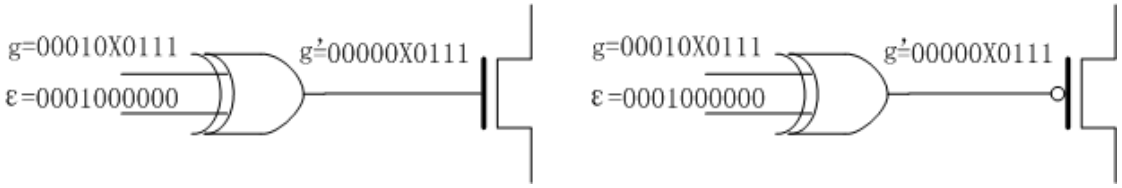
$$g'_{Stuck-OFF} = \mathrm{AND}_{\mathrm{sto}}(\mathrm{g}, \bar{\varepsilon}) = \mathrm{g} \cdot (1 - \varepsilon) \tag{5}$$

So, a stochastic transistor model can be constructed as follows:

- If the transistor is affected by a flipping error, then the stochastic XOR is used;
- If the transistor is affected by a stuck-ON error, the stochastic OR is used;
- If the transistor is affected by a stuck-OFF error, the stochastic inverter and AND is used.

In (3), (4) and (5), an input X is considered to always produce the same output (i.e., X), regardless of the stochastic logic being performed. Due to space limitation, only the model of the flipping error is presented in this paper.

Differently from the logic-level SCM approach in [7], in which the random bits in the binary streams are considered equivalent and with no order, the stochastic sequences used in this paper match the operation of the transistor in multiple clock cycles. This allows to account for errors that occur within a single and multiple clock cycles. Additionally, the temporal sequences in the binary bits ensure the correct modeling of the floating state that could result from the pull-up and pull-down operations of the transistors, as explained in more detail in the next section.



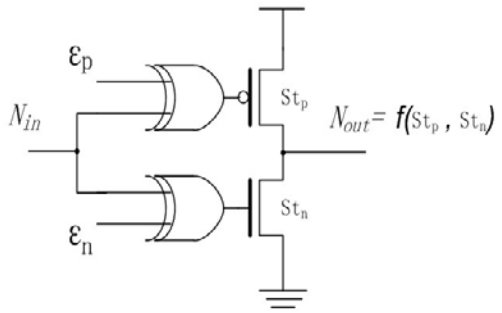
(a) (b)

Figure 4. Stochastic transistor models for the flipping error: (a) NMOS and (b) PMOS.

IV. TRANSISTOR-LEVEL STOCHASTIC MODELING OF LOGIC GATES

Faults and defects are likely to affect the correct operation of individual transistors in the logic gates of a combinational circuit; so for an accurate and realistic reliability analysis, the gate error rate should be derived in terms of the transistor error probability, while considering also the gate topology as well as the input vectors.

Similar to the logic-level SCM approach in [7], the transistor-level stochastic approach uses stochastic random sequences to represent both signal and error probabilities. However, the traditional SCM approach is static in the sense that circuit reliability is evaluated without considering signal sequences and timing information, thus it may not always be directly applicable to the temporal operation of the transistors and of the sequential elements such as the flip-flops. Therefore the stochastic streams in the new model are defined differently to account for signal sequencing. Initially, consider the CMOS inverter as an example (Figure 5(a)). Given an input sequence N_{in} and the switching error rate sequences ε_p and ε_n, the ON/OFF states of the PMOSFET P and the NMOSFET N are obtained via functional bit-parallel simulation of the input sequences. Then the output node sequence N_{out} is found as a function of the state of each transistor, i.e., $N_{out} = f(St_P, St_N)$; this can generally be estimated according to the functionality of the gate. For the inverter it is given as follows: (1) when P (pull-up network) is ON and N (pull-down network) is OFF, the output is logic 1; (2) when P (pull-up network) is OFF and N (pull-down network) is ON, the output is logic 0; (3) when P and N are simultaneously OFF, the output is floating, or Z (i.e., it depends on its previous value); and (4) when P and N are simultaneously ON or any of P and N is indefinite or IND, the output is defined as unknown, or X. This is also shown in the table of Figure 5(b).



(a)

Pull-up/down network operation / Gate output N_{out}	Pull-up Network P	Pull-down Network N
0	OFF	ON
1	ON	OFF
Z	OFF	OFF
X	ON	ON
	IND	Don't care
	Don't care	IND

(b)

Figure 5.（a）Proposed stochastic model for the inverter, and （b）The gate output as determined by the pull-up and pull-down networks.

Since random binary bit streams are used for each circuit node in a functional/fault simulation (i.e., serially in the time domain), the sequencing property of the bits are defined in such a way that each bit in the sequences represents a logic value at a certain node in one clock cycle. Therefore, it is possible to define and calculate the floating output Z as the previous bit

value in a sequence. This is based on the assumptions that the node leakage is negligible and that the node charge will remain at the same level until a refresh operation occurs. Since the unknown (leakage) output X usually falls into the undefined voltage region, it is assumed that as the worst case, it is a faulty output (it usually cannot be immediately restored by the gates). The proposed method is amenable to a parallel-bit simulation for combinational circuits, while for sequential elements (such as flip-flops), sequential simulation may still be required.

To better understand the proposed approach, more general cases can be illustrated using NAND2 and NOR2, where the pull-down and pull-up networks consist of more than one transistor. Let the input sequences N_{in1} and N_{in2} have a length of L and each bit represents the signal value during one clock cycle; therefore, a sequence represents the sequential states in L clock cycles. The error rate of each transistor is then encoded into the stochastic sequences.

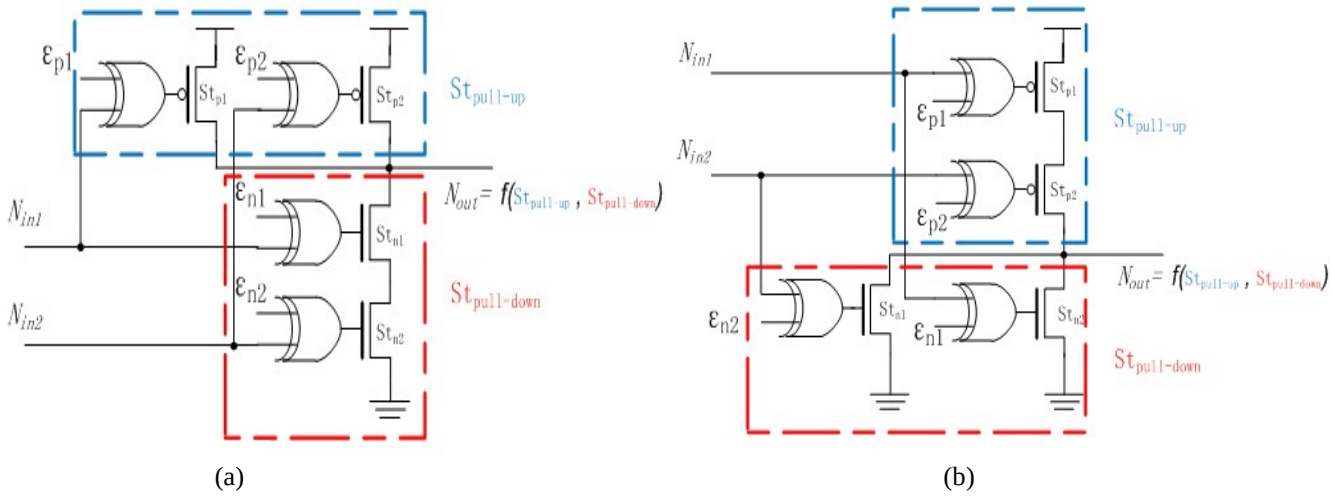


Figure 6. Transistor-level stochastic models for logic gates: (a) NAND2 and (b) NOR2.

As shown in Figure 6, the operation of each transistor is characterized by an error rate ε and a gate input as two sequences of length L. Assuming that the transistors are independent, the newly-generated input sequences by the XOR gates determine the ON/OFF state of each transistor. The output of the pull-up/down network can be computed based on the states of the individual transistors. For a pull-up/down network with multiple transistors, its operational status is determined by the topology of the pull-up/down network as follows:

- For the transistors connected in series in a pull-up/down network, the network is "ON" when all transistors are ON; this is equivalent to applying a stochastic AND gate to the states of the transistors. As shown in Figure 6(a) for example, $St_{pull-down} = AND(St_{n1}, St_{n2})$. The detailed mapping relationships for two transistors connected in series are shown in Table 1(a), and they can readily be extended to any number of transistors connected in series.
- For the transistors connected in parallel in a pull-up/down network, the network is "ON" when any of the transistors is ON; this is equivalent to applying a stochastic OR gate to the states of the transistors. As shown in Figure 6(a) for example, $St_{pull-up} = OR(St_{p1}, St_{p2})$. The detailed mapping relationships for two transistors connected in parallel are shown in Table 1(b), and they can similarly be extended to any number of transistors connected in parallel.

Table 1. Mappings between transistors and networks: (a) series network; and (b) parallel network.

(a)

Transistor state / Network state	Transistor #1	Transistor #2
ON	ON	ON
OFF	OFF	Don't care
	Don't care	OFF
IND	IND	ON
	ON	IND
	IND	IND

(b)

Transistor state / Network state	Transistor #1	Transistor #2
ON	ON	Don't care
	Don't care	ON
OFF	OFF	OFF
IND	IND	OFF
	OFF	IND
	IND	IND

The output of the gate N_{out} is established by considering the states of the pull-up and pull-down networks as follows (also shown in Figure 5(b)):

- If the pull-up network is ON and the pull-down network is OFF, then the output is 1.
- If the pull-up network is OFF and the pull-down network is ON, then the output is 0.
- If the pull-up network is OFF and the pull-down network is OFF, then the output is Z, which depends on the previous value.
- If the pull-up network is ON and the pull-down network is ON, or any of the networks is IND, then the output is X.

The gate error rate/reliability can then be calculated by comparing the faulty and fault-free output sequences. Hence, the proposed approach to modeling a logic gate consists of three types of mapping: 1) mappings from the gate inputs to the operations of the transistors, as illustrated in Figure 3; 2) mappings from the transistors to a pull up/down network, as shown in Table 1; and 3) mappings from the pull up and pull down networks to the gate outputs, as shown in Figure 5(b).

V. Circuit-Level Evaluation Approach

For a circuit made of unreliable transistors, its reliability can be estimated by evaluating the stochastic bit streams following propagation from the primary inputs to the primary outputs. Practically, this can be done by comparing the obtained output sequences for the unreliable and reliable circuit case; such a procedure can be implemented for the benchmark circuit C17 as follows. Initially, the stochastic (unreliable) circuit is obtained by adding an XOR gate to each of the transistors in C17. Then, the input signals as well as the transistor error probability (that is now an input to the XOR gate) are initialized by generating random bit streams. The streams are propagated through the stochastic circuit and the original fault-free circuit, as shown in Figure 7. Subsequently, XOR gates are used to detect the mismatch of the stochastic sequences from the unreliable and the reliable circuits. Since the C17 has more than one primary output, the joint circuit error probability can be obtained by using a stochastic OR gate to detect any error present in the multiple stochastic output sequences. The final structure is shown in Figure 7.

The evaluation procedure using the transistor-level stochastic approach as proposed in this paper is given as follows:

1. Construct the stochastic circuit by adding an XOR gate to each of the transistors in the circuit (for flipping errors);
2. Generate the initial random bit streams for the signal probabilities of the primary inputs and the error probabilities for the transistors;
3. Propagate the stochastic streams from the primary inputs to the primary outputs in both the reliable and unreliable circuits;
4. Use XOR and OR gates to decode the joint error probability of the circuit from the obtained stochastic bit streams.

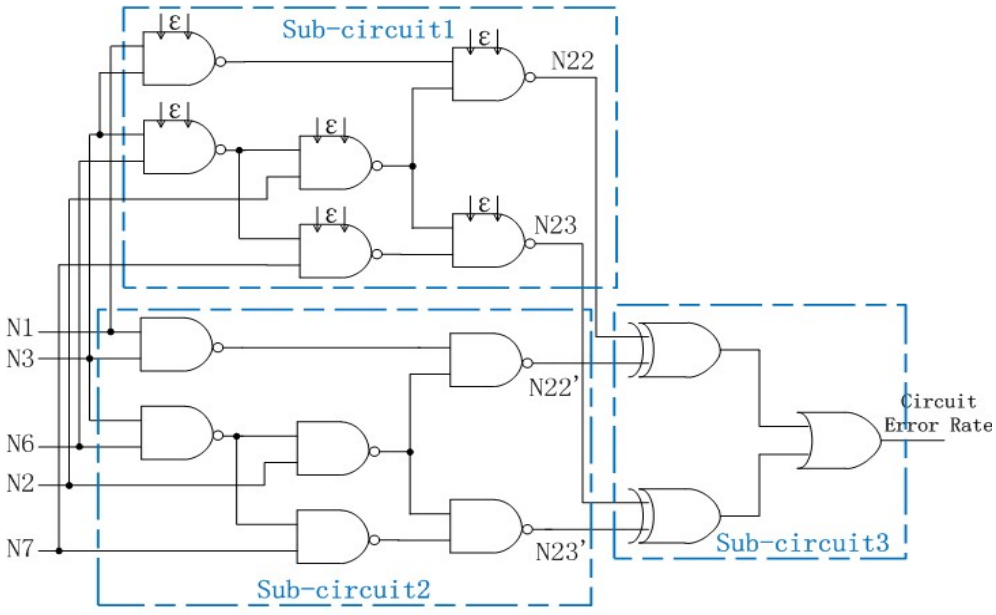


Figure 7. Computational structure for the reliability evaluation of C17. Sub-circuit1: the stochastic circuit, implemented using the NAND gate of Figure 6(a); Sub-circuit2: the original fault-free circuit, implemented using regular NAND gates; Sub-circuit3: XOR and OR gates for obtaining the joint error probability from the output stochastic sequences.

VI. SIMULATION RESULTS

For validating the applicability and the accuracy of the transistor-level stochastic models, the proposed approach is compared with the transistor-level Monte Carlo (MC) simulation and the gate-level SCM approach for the ISCAS-85 benchmarks [18]. Simulations were performed on a 2.00-GHz AMD microprocessor with 2 GB memory. In the MC simulation, random input vectors are applied and faults are randomly injected into the circuits. The circuit reliability is then obtained by the statistical outcomes using a large number of simulation runs. Compared to the MC simulation, the proposed stochastic approach is more efficient as it requires a significantly smaller number of pseudo-random generations in the stochastic computing process. This is confirmed by the simulation results shown in Table 2. In the MC simulation, a total number of one million simulations were run for each circuit to ensure a relatively stable output reliability, while in the stochastic approach, a sequence length of 10,000 bits were used and produced a relatively stable output reliability. It can be seen that while both approaches provide an accurate evaluation of circuit reliability, the proposed approach requires a significantly smaller runtime compared to the MC simulation.

In the gate-level SCM approach, it is assumed that the correct functioning of a gate requires the correct functioning of all its transistors. Thus, a simple equation is used to relate the reliability of the transistors to that of a gate, i.e., $\varepsilon_{gate} = 1 - (1 - \varepsilon_{transistor})^n$, where n is the number of transistors in the gate. The proposed transistor-level approach considers the gate topology and the applied input vectors, so it produces different error rates for different types of gates. Therefore, in Table 1 the gate-level approach results in a difference as large as 400% compared to the transistor-level approach. Due to the use of a conservative gate error rate in the gate-level approach, a lower circuit reliability is generated. The bit-parallel nature of the proposed approach can be further explored to reduce its computational complexity through the potential parallelization of the stochastic simulation.

Table 2. Simulation results for ISCAS-85 benchmarks by Monte Carlo simulation, the gate-level SCM approach and the proposed transistor-level approach.

Circuit	*Characteristics*			*Monte Carlo Simulation* $\varepsilon = 10^{-3}$ $sample = 1,000,000$		*SCM* $\varepsilon_{gate} = 1 - (1 - 10^{-3})^n$ $L = 1,000$, $input = 1,000$		*Proposed stochastic approach* $\varepsilon = 10^{-3}$ $L = 10,000$	
	gates	*inputs*	*outputs*	*Circuit error rate*	*Time*	*Circuit error rate*	*Time*	*Circuit error rate*	*Time*
C17	6	5	2	9.1×10^{-3}	3.002m	1.94×10^{-2}	0.046s	8.9×10^{-3}	0.534s
C432	250	36	7	5.5×10^{-2}	109.55m	2.31×10^{-1}	26.115s	5.6×10^{-2}	14.645s
C499	202	41	32	6.05×10^{-2}	195.29m	2.98×10^{-1}	24.723s	6.01×10^{-2}	26.293s
C880	383	60	26	7.01×10^{-2}	213.35m	3.67×10^{-1}	44.796s	7.27×10^{-2}	34.446s
C1355	546	41	32	8.47×10^{-2}	309.60m	4.73×10^{-1}	61.306s	8.62×10^{-2}	51.549s
C1908	880	33	25	1.44×10^{-1}	487.55m	6.59×10^{-1}	80.798s	1.43×10^{-1}	78.061s
C2670	1193	157	64	1.73×10^{-1}	796.57m	7.69×10^{-1}	134.16s	1.71×10^{-1}	103.82s
C3540	1669	50	22	2.21×10^{-1}	1225.43m	8.22×10^{-1}	172.01s	2.19×10^{-1}	156.43s
C5315	2307	178	123	3.05×10^{-1}	1764.86m	8.89×10^{-1}	267.22s	2.97×10^{-1}	206.13s

VII. CONCLUSION

Accurate and efficient reliability evaluation techniques are very important as CMOS technology continues to move in the nanometric regimes. A transistor-level analysis accounts for features such as temporal signal sequences, logic gate topology and different input vectors to determine the reliable operation of circuits; hence, it is more accurate than existing gate-level evaluation methodologies. This paper presents such an approach using stochastic transistor models for the evaluation of nanometric CMOS circuits. Simulation results have shown the accuracy and efficiency of the proposed approach. It is scalable to the evaluation of large circuits and can be further improved by considering more accurate physical models of the transistor as basic element of digital circuit design.

REFERENCES

[1] International Technology Roadmap for Semiconductors (ITRS) 2009, SIA, http://www.itrs.net/reports.html

[2] S. Krishnaswamy, G. F. Viamontes, I. L. Markov, and J. P. Hayes, "Probabilistic transfer matrices in symbolic reliability analysis of logic circuits," *ACM Trans. Des. Autom. Electron. Syst*., vol. 13, no. 1, pp. 1–35, 2008.

[3] J. Han, H. Chen, E. Boykin, J. Fortes, "Reliability evaluation of logic circuits using probabilistic gate models," *Microelectronics Reliability*, vol. 51, no. 2, 2011, pp. 468-476.

[4] T. Rejimon and S. Bhanja, "Scalable probabilistic computing models using Bayesian networks," in *Proc. Int. Midwest Symp. Circuits Syst.*, 2005, pp. 712–715.

[5] M. R. Choudhury and K. Mohanram, "Reliability analysis of logic circuits," *IEEE TCAD*, vol. 28, no. 3, pp. 392–405, March 2009.

[6] S. Krishnaswamy, S. M. Plaza, I. L. Markov, and J. P. Hayes, "Signature-based SER analysis and design of logic circuits," *IEEE Trans. Comput*.-Aided Design Integr. Circuits, vol. 28, no. 1, pp. 74–86, Jan. 2009.

[7] H. Chen, J. Han, "Stochastic Computational Models for Accurate Reliability Evaluation of Logic Circuits," *Proc. Great Lakes Symp. VLSI (GLVLSI)*, Providence, RI, USA, pp. 61-66 (2010)

[8] W. Ibrahim, V. Beiu, "Reliability of NAND-2 CMOS gates from threshold voltage variations," *IIT '09, pp. 135-139*.

[9] P. Zarkesh-ha, and A. A. M. Shahi, "Logic Gate Failure Characterization for Nanoelectronic EDA Tools" *Proc. IEEE Int. Symp. DFT, Albuquerque, NM, USA, Oct. 2010, pp.16–23*.

[10] W. Ibrahim, V. Beiu, "Using Bayesian networks to accurately calculate the reliability of complementary metal oxide semiconductor gates," *IEEE Transactions on Reliability, to appear, Dec. 2010*.

[11] B. R. Gaines, "Stochastic Computing," *Spring Joint Computer Conf.*, 1967, Vol. 30, pp. 149-156.

[12] B. Brown and H. Card, "Stochastic neural computation I: Computational elements," *IEEE Tran. Computers*, vol. 50, pp. 891–905, Sept. 2001.

[13] X. Li, W.K. Qian, M. Riedel, K. Bazargan, D. Lilja, "A Reconfigurable Stochastic Architecture for Highly Reliable Computing," *Proc. Great Lakes Symp. VLSI (GLVLSI)*, Boston, MA, USA, pp. 315-320, 2009

[14] W. Qian, X. Li, M. D. Riedel, K. Bazargan, and D. J. Lilja, "An architecture for fault-tolerant computation with stochastic logic," *IEEE Tran. Computers*, vol. 60, pp. 93–105, Jan. 2011.

[15] J. Han, J. Gao, Y. Qi, P. Jonker, J.A.B. Fortes. "Toward Hardware- Redundant, Fault-Tolerant Logic for Nanoelectronics," *IEEE Design and Test of Computers*, July/August 2005, vol. 22, no. 4, 328-339.

[16] S. S. Tehrani, S. Mannor, and W. J. Gross, "Survey of stochastic computation on factor graphs," in *Proc. 37th IEEE Int. Symp. Multiple-Valued Logic*, Oslo, Norway, May 2007, pp. 54–59.

[17] C. Winstead, V. C. Gaudet, A. Rapley, and C. B. Schlegel, "Stochastic iterative decoders," in *Proc. Intl Symp. Info. Theory*, pp. 1116-1120, 2005.

[18] M. Hansen, H. Yalcin, and J. P. Hayes, "Unveiling the ISCAS-85 Benchmarks: A Case Study in Reverse Engineering," IEEE Design and Test, vol. 16, no. 3, pp. 72-80, July-Sept. 1999.