Variation-Aware Delay Fault Testing for Carbon-Nanotube FET Circuits

Sanmitra Banerjee, *Graduate Student Member, IEEE*, Arjun Chaudhuri, *Graduate Student Member, IEEE*, August Ning, *Graduate Student Member, IEEE*, and Krishnendu Chakrabarty, *Fellow, IEEE*

***Abstract*— Sensitivity to process variations and manufacturing defects are major showstoppers for the high-volume manufacturing of carbon nanotube field-effect transistors (CNFETs). These imperfections affect gate delay and may remain undetected when test patterns obtained using conventional test-generation techniques are used. We propose a new test generation method that takes CNFET-specific process variations into account and identifies multiple testable long paths through each node in a netlist. In contrast to state-of-the-art techniques, our method can also handle variations that have a nonlinear impact on the propagation delay. The generated test patterns ensure the detection of delay faults through the longest path, even under random CNFET process variations. The proposed method shows significant improvement in the statistical delay quality level (SDQL) compared with a state-of-the-art technique and a commercial ATPG tool for multiple benchmarks. We observed a minimum of 17.1% improvement in the SDQL offered by our patterns over a test set of the same size generated by the commercial tool. We also show that our method, when integrated with the conventional transition fault test flow, offers a significant improvement in the quality of test patterns under random variations. Moreover, the proposed method is flexible and can be easily extended to other emerging device technologies.**

***Index Terms*— Carbon nanotubes, delay faults, process variations, small delay defects (SDDs), transition faults.**

I. Introduction

CARBON nanotube field-effect transistors (CNFETs) offer a plethora of excellent electrical properties, including increased energy efficiency, high ON-current/OFF-current ratio, and low subthreshold swing. A 16-bit microprocessor based on the RISC-V instruction set fabricated using industry-standard design flow and processes has been reported in [1]. CNFETs are expected to be increasingly ubiquitous in the future; there are multiple instances of industry engagement in CNFET thermal management, power supply, and nanotube growth [2].

In spite of the superior device characteristics, defect-screening and variation-aware testing hinder high-volume manufacturing of CNFETs. Typical carbon nanotube (CNT) growth techniques often result in uneven CNT diameter, bundling of nanotubes, or misalignment [3]. Catastrophic faults can occur if one or more CNTs are metallic in nature. These CNTs are conducting irrespective of the gate voltage and, thus, can result in stuck-on CNFETs. Moreover, the CNFET fabrication process is immature and can result in process variations involving various parameters. Such variations can affect the ON-current (I_{ON}) and gate delay.

Resistive opens and bridges in vias and interconnects often result in small delay defects (SDDs) in nanometer circuits [4]. SDDs can also originate due to variations in device parameters that typically have a parametric impact on the propagation delay of affected logic gates [5]. Such parameter variations can be caused by imperfections in the fabrication flow, thermo-mechanical stress, crosstalk, power-supply noise, and aging [6]. In stacked through-silicon via (TSV)-based 3-D ICs, delay faults can occur due to misalignment, voids, and pinholes in TSVs [7]. These defects pose yield challenges in 3-D ICs, and Ni *et al.* [7]–[10] have proposed several cost-effective TSV redundancy and repair methods. The ideas of TSV repair using honeycomb topology and reusing TSVs by time-division multiplexing access are of genuine scientific value and are being investigated for applications of 3-D ICs in the post-Moore era. While these methods can mitigate TSV faults in CNFET-based 3-D ICs, the device-level parameter variations in CNFETs will differ from those in Si-MOSFETs.

Traditionally, burn-in tests have been used to identify dies with reliability concerns [5]. However, burn-in is an expensive process and has been shown to damage dies due to extreme stress conditions [11]. This motivates the use of SDD tests as a low-cost alternative. To ensure efficient SDD detection, delay faults must be tested via long paths. Commercial EDA tools attempt to ensure this by using a "constrained" transition delay fault model—only the paths having a nominal timing slack lower than a predefined margin are used for testing.

In this article, we show that CNFET parameter variations have a significant impact on the path delay in CNFET-based logic circuits, and therefore, these variations must be taken into account while shortlisting paths for SDD testing. We also show that variations in process and design parameters affect CNFET devices in complex and nonlinear ways. Based on these findings, we highlight the limitations of test-generation methods based on Si circuits when they are applied to CNFET circuits. We, therefore, propose a delay-fault test generation method that specifically targets CNFET parameter variations. The main contributions of this article are as follows:

Manuscript received August 14, 2020; revised November 7, 2020; accepted November 28, 2020. Date of publication January 8, 2021; date of current version January 28, 2021. This work was supported in part by the DARPA ERI 3DSOC Program under Award HR001118C0096. *(Corresponding author: Sanmitra Banerjee.)*

Sanmitra Banerjee, Arjun Chaudhuri, and Krishnendu Chakrabarty are with the Department of Electrical and Computer Engineering, Duke University, Durham, NC 27708 USA (e-mail: sb535@duke.edu).

August Ning is with the Department of Electrical Engineering, Princeton University, Princeton, NJ 08544 USA.

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TVLSI.2020.3045417.

Digital Object Identifier 10.1109/TVLSI.2020.3045417

1063-8210 © 2021 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.

Authorized licensed use limited to: Princeton University. Downloaded on March 11,2023 at 05:34:33 UTC from IEEE Xplore. Restrictions apply.

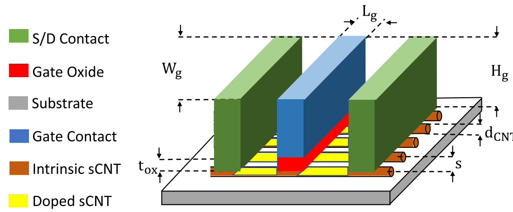


Fig. 1. 3-D schematic of a CNFET.

1) analysis of the impact of process variations on the transistor- and gate-level performances of CNFETs;
2) insights into why CNFET process variations must be taken into account during test generation;
3) a long-path selection algorithm that enables efficient test generation for SDDs in the presence of process variations.

The statistical delay quality level (SDQL) is a commonly used surrogate metric for SDD coverage and test pattern grading [12]; a lower value of SDQL signifies higher effectiveness of the SDD test patterns. We compare the pattern sets obtained using the proposed method with those generated by a related technique from academia, as well as a state-of-the-art commercial ATPG tool. Simulation results for multiple benchmarks show that the test generated using the proposed method consistently offers the lowest SDQL value.

The remainder of this article is organized as follows. In Section II, we review CNFET fundamentals, the CNFET compact model, delay-fault testing, and prior work on variation-induced delay fault detection. We also examine various CNFET fabrication processes. In Section III, we analyze the impact of variation in CNFET process parameters at the transistor and gate levels. Section IV describes the proposed variation-aware delay-fault test-generation method. In Section V, we compare the effectiveness of the proposed method with an SDD testing technique described in the literature and a commercial EDA tool. Section VI concludes this article.

II. Background and Motivation

A. CNFET Fundamentals

Quasi-ballistic transport in CNTs results in high drive current and transconductance [13]. CNFETs are particularly attractive for high-speed applications due to the high Fermi velocity (10^6 m/s) and small signal-switching speed [14]. The subthreshold characteristics of Si-MOSFET beyond the 10-nm technology node are degraded due to carrier tunneling. On the other hand, CNFETs offer excellent subthreshold swing even at the 7-nm node [15]; this motivates the use of CNFETs as a promising alternative for continued scaling. The CNFET device parameters are shown in Fig. 1, and their nominal values at several technology nodes are listed in Table I.

We use the Stanford Virtual Source CNFET (VSCNFET) predictive compact model to simulate CNFETs [16]. The empirical parameters in the model have been extracted by curve fitting with experimental results and numerical simulations based on the nonequilibrium Green's function. In this work, we have considered a top-gate geometry, similar to

TABLE I

Nominal Values of Parameters at Different CNFET Technology Nodes [18], [19]

Parameter	7 nm	11 nm	14 nm	22 nm	45 nm
CNT diameter (d_{CNT}) [nm]	1.2	1.2	1.2	1.2	1.2
Gate length (L_g) [nm]	11	14	19	26.5	40
Oxide thickness (t_{ox}) [nm]	0.7	0.8	0.9	1	1.25
Gate width (W_g) [nm]	63	90	112	160	240
Inter-CNT spacing (s) [nm]	4	4	4	4	4
Gate height (H_g) [nm]	15	20	20	30	40

what was considered in [17]. We showed in [17] that the impact of process variations on CNFET performance does not depend on the inter-CNT capacitive charge screening effects. Thus, we have neglected these screening effects to simplify our analysis. All simulations were performed at a temperature of 298 K.

B. Delay Fault Models

Process variations and manufacturing defects result in parametric faults, increased propagation delay, and slower signal transitions. The gate delay fault model targets defects that affect the propagation delay of gates [20]. The path delay fault model targets the accumulation of delays, including small delays, along a path. Of particular interest are long paths, i.e., paths with small timing slack. The problem of finding all the long paths is known to be NP-hard [21], therefore, often, only a subset of long paths is tested.

The transition fault model is a special case of the gate delay fault model; it assumes that the delay fault at a node exceeds the clock period and is catastrophic enough to cause logic failure [22]. A transition fault can affect the timing slack of all paths through the affected cell. However, a fault can be detected only on paths with a small enough timing slack such that, in the presence of the fault, a signal transition is not captured within the rated clock period. Therefore, to detect small-delay defects, it is critical that the longest (minimum-slack) paths through a fault site are sensitized.

C. Variation-Induced Delay Fault Detection

As transistors are scaled down in advanced technology nodes, the impact of process variations on device performance is increasingly being felt [23]. This is especially true for emerging technologies, such as CNFETs, because the fabrication process is immature and prone to imperfections. While process variations in CNFETs have received only limited attention [13], [24], [25], such imperfections in Si-MOSFETs have been extensively studied. In Si-based circuits, variations can originate either in the interconnect or in the FETs. Changes in the interlayer dielectric (ILD) thickness and metal layer width during the chemical–mechanical polishing affect the interconnect delay and the clock skew [26].

Yield loss is also encountered due to variations in device parameters. Agarwal *et al.* [27] present a spatial correlation-based method to model the impact of interdie and intradie variations in the gate length. A probabilistic method to predict circuit performance under gate-level variations has been presented in [6]. Variations have been considered in the oxide thickness, threshold voltage, and gate length.

Authorized licensed use limited to: Princeton University. Downloaded on March 11,2023 at 05:34:33 UTC from IEEE Xplore. Restrictions apply.

SDDs constitute a special class of transition delay faults that can only be detected by sensitizing long paths, including paths whose slack is reduced due to process variations. Therefore, a method was described in [28] to identify multiple long paths per instance in the presence of variations in ILD thickness, metal width, metal thickness, and gate length.

Note that, while the interconnect delay models obtained from Si technology can be reused to some extent for CNFETs, significant changes are required for the device-level delay models. The methods proposed in [28] and [29] only consider variations that have a linear impact on the propagation delay. In [29], this linear dependence allows the use of a response surface method technique to achieve a 40× speedup in simulation over SPICE-based approaches. Similarly in [28], the path delay is expressed as a linear function of the parameters affected by process variation. The long-path selection problem is then mapped to the polynomial-time feasibility problem in linear programming. However, as we show in Section III-B, the impact of CNFET parameter variations on the gate delay is nonlinear for several parameters. This limits the applicability of previously proposed Si-based methods.

Moreover, in addition to Si-MOSFET-like parameters, variations in the CNT diameter and density are major sources of CNFET yield loss. The CNT diameter determines the band-gap and threshold voltage, while the drive current is directly proportional to the CNT density. These critical parameters are specific to CNFET technology and are, therefore, not considered in any of the state-of-the-art timing analysis models. To simplify the analysis, the method proposed in [28] does not consider intradie variations, and it is assumed that all the FETs in the design have similar parameter variations. However, it is unlikely that this assumption will hold for large designs implemented using early generation CNFET technologies.

In [30], a timing-unaware commercial ATPG tool is used to activate SDDs on long paths by constraining the set of available endpoints (scan flip-flops) for capture. The paths in the design are first classified into three groups—long path, intermediate path, and short path based on their length and the minimum sizes of the delay defect detected through the path. During ATPG, only the scan flip-flops on the long paths are considered as observation points, while the other flops are masked. This forces the ATPG tool to activate the SDDs through the long paths. A multiple-detect (in this case, a 15-detect) technique is used to further increase the probability of the activation of the long paths. Note that the classification of the paths based on their length is critical to the efficiency of this method; however, this is performed based only on the nominal path delays. As we show in Section III-B, CNFET parameter variations have a significant impact on the propagation delay. Due to this, it is possible that a path that is not classified as a “long path” based on its nominal delay can be critical under a process variation scenario. Therefore, the use of this method, especially for CNFET designs, can result in SDD test escapes.

A “three-pass” ATPG-based method to select paths that are critically affected by process variations is proposed in [31]. However, this approach does not ensure that all nodes in the design are covered by the selected set of paths. Therefore, SDDs on nodes that are not covered remain undetected. From the above discussion, it is clear that none of the existing models can predict the impact of process variations on the timing characteristics of CNFET circuits. This motivates the need for a new method to generate test patterns, which takes CNFET parameter variations into consideration. The proposed method considers both interdie and intradie variations in CNFET parameters, and it can efficiently handle the nonlinear dependence between these variations and the gate delay. Our test-generation solution can be easily adapted to other emerging devices. The specific relationships between device parameters and gate delay are inputs for test-generation. In this work, we demonstrate its application to CNFET circuits.

As mentioned earlier, interconnect delay models from Si technology can be also applied to CNFET circuits. Therefore, in this article, we focus on the impact of variation in CNFET device parameters on the effectiveness of SDD testing. Note that, in the proposed method, we are concerned with the impact of parameter variations on the path delays and not the absolute path delays themselves. Interconnect delays are independent of CNFET device parameters, and therefore, we do not take them into account while selecting the list of long paths under random variations.

D. CNFET Fabrication Process

The CNFET fabrication process can lead to parametric faults, such as high leakage current, increased susceptibility to noise, and timing failures. The conductivity of undoped semiconducting CNTs (s-CNTs) can be modulated by band bending using a gate voltage, thus rendering them useful for CMOS logic applications. However, during CNT growth, the presence of undesirable metallic CNTs (m-CNTs) is often observed. Methods for removing m-CNTs have been proposed [32], [33]; however, none of these methods are completely effective, and often, some s-CNTs can be inadvertently etched along with the m-CNTs. This results in a degraded drive current, which affects the gate delay [34].

Single-walled s-CNTs can be grown by chemical vapor deposition (CVD) of methane on supported transition metal–oxide catalysts. This method often results in variation in the CNT diameter due to bundle formation. Alternatively, CNTs can be grown on a source substrate and later transferred to a target sample for CNT growth at a lower thermal budget than CVD. However, in this approach, misaligned CNTs are often observed [35]. The oxygen plasma etching step associated with this process can also result in unwanted CNT removal.

Parameter variations introduced during fabrication can significantly undermine the advantages offered by CNFETs. In addition to affecting the initial yield ramp-up, such variations have a detrimental impact even after the technology matures. Previous studies on silicon technology have shown that within-die variations can affect the maximum clock frequency and leakage power of multicore processors [36], [37]. In a similar study on CNFETs [25], the authors show that, even for a reasonably mature CNT growth process, logic gates are susceptible to parameter

Authorized licensed use limited to: Princeton University. Downloaded on March 11,2023 at 05:34:33 UTC from IEEE Xplore. Restrictions apply.

variations. Therefore, a variation-aware test generation method will be needed even after CNFET technology matures.

III. Process Variations in CNFETs

A. Transistor-Level Impact of Process Variations

In a recent work, we have analyzed the impact of variations associated with CNFET parameters on the ON-state drain current (I_{ON}) [17]. The CNT diameter, d_{CNT}, is controlled by the CNT chirality and determines the bandgap (E_g). As d_{CNT} increases, E_g decreases, and I_{ON} increases. However, with decreasing E_g, the subthreshold leakage current increases, resulting in a degraded $I_{\mathrm{ON}}/I_{\mathrm{OFF}}$, where I_{OFF} is the OFF-state current. Again, as d_{CNT} decreases, I_{ON} decreases, resulting in slower CNFETs. The CNFET series resistance, inversion gate capacitance, and threshold voltage also depend on d_{CNT}.

As the gate length, L_g, increases, the carriers need to travel a longer distance through the channel and are, therefore, prone to collisions. Similarly, as the dielectric thickness t_{ox} increases, the gate capacitance decreases, and the gate control over the channel decreases. Due to this, I_{ON} decreases with an increase in gate length and dielectric thickness. No significant change in I_{ON} is observed with variation in the gate height.

The number of CNTs in the CNFET active layer is given by $N_{\mathrm{CNT}} = \lceil W_g/s \rceil$, where W_g is the gate width and s is the inter-CNT spacing. I_{ON} is quantized with respect to N_{CNT}; thus, I_{ON} changes in steps with W_g and s. Therefore, small variations in W_g and s do not have a noticeable impact unless the nominal values lie at or near a step edge.

Optical response analysis using TEM shows that the diameter of single-walled CNTs follows a Gaussian distribution [38]. Gate length, width, height, and oxide thickness also vary in a Gaussian distribution in bulk and SOI MOSFETs with their nominal value as mean (μ) and a standard deviation (σ) between $0.01 \cdot \mu$ and $0.1 \cdot \mu$ [6], [39]. Experimental results show that, while inter-CNT spacing, s, does not always follow a Gaussian distribution, $N_{\mathrm{CNT}} = \lceil W_g/s \rceil$ can be accurately modeled by a Gaussian PDF [40]. This is possible because variations can nonuniformly affect the spacing between CNTs in a CNFET, and as such, s may not be equal for all pairs of neighboring CNTs. However, the VSCNFET model allows only a single value of s, which is considered as the spacing between all CNT pairs in a CNFET. On the other hand, the use of a Gaussian-distributed s is consistent with the experimentally observed Gaussian PDF for N_{CNT}.

In [17], we quantified the impact of independent parameter variations on I_{ON} using Monte Carlo simulations. We observed that, across all technology nodes, the CNT diameter has the maximum impact on I_{ON}, followed by t_{ox} and L_g. Due to the step dependence, the impact of either W_g or s is the least when their nominal value does not lie at a step edge. I_{ON} is affected by variations in W_g and s only if the variation is large enough to affect the CNT count N_{CNT}, where $N_{\mathrm{CNT}} = \lceil W_g/s \rceil$. For variations in H_g, the value of ($\sigma_{\mathrm{ON}}/\mu_{\mathrm{ON}}$) is low due to the weak dependence of I_{ON} on H_g.

B. Gate-Level Impact of Process Variations

Variations in the CNFET process parameters affect I_{ON}, which, in turn, affects the propagation delay of logic gates. These variations are typically manifested in the form of SDDs. In the proposed delay-fault testing method, we take these process variation scenarios into account while generating the test patterns. As the first step toward this goal, we study the impact of variations in each CNFET parameter on the gate delay. For each gate, we introduce single-parameter variations in all the CNFETs using the Verilog-A file accompanying the VSCNFET compact model. Using HSPICE simulations, the propagation delay of the gate G_k in the presence of variations in parameter p_i, given by d_{ik}, is calculated. Similar to Si-MOSFETs, the gate delay increases linearly with an increase in t_{ox} and L_g. Fig. 2 shows how the normalized delay d_{ik}/d_{0k} varies with variations in the other CNFET parameters, namely d_{CNT}, s, and W_g, for the AND2×1, OR2×1, NAND2 × 1, and NOR2 × 1 standard cells. Note that d_{0k} is the nominal gate delay, and the percentage deviation of a parameter (x_{ik}) is calculated from its nominal value. All simulations were performed at the 7-nm technology node, and the corresponding nominal values listed in Table I were used.

The variation in d_{CNT} has a catastrophic impact on I_{ON}; this is also reflected in the propagation delay of standard cells. As d_{CNT} decreases, I_{ON} decreases, and the propagation delay increases. From Fig. 2, we observe that even a 5% decrease in d_{CNT} results in a 50% increase in the gate delay. Similarly, with an increase in gate length and oxide thickness, I_{ON} decreases, and gate delay increases linearly.

From Fig. 2, we observe that, as W_g changes, the propagation delay varies in a ramp-like fashion. I_{ON} is proportional to $N_{\mathrm{CNT}} = \lceil W_g/s \rceil$; therefore, $I_{\mathrm{ON}} = \lambda_1 \lceil W_g/s \rceil$, where λ_1 is proportionality constant. Similar to Si-MOSFETs, the gate capacitance of the load C_g is λ_2 W_g, where λ_2 is another constant of proportionality. The gate delay is the time required to charge the load capacitance [41]; in the presence of variations in W_g, this is given by

$$d_{W_g,k} = \frac{C_g V_{\mathrm{DD}}}{I_{\mathrm{ON}}} = \frac{\lambda_2 W_g \cdot V_{\mathrm{DD}}}{\lambda_1 \lceil W_g/s \rceil} = K \cdot \frac{W_g}{\lceil W_g/s \rceil} \quad (1)$$

where V_{DD} is the supply voltage and K is a constant independent of W_g. In Fig. 2, when W_g increases from point A to point B, $\lceil W_g/s \rceil$ remains constant. Due to this, $d_{W_g,k}$ increases linearly with W_g. However, when W_g increases from point B to point C, $\lceil W_g/s \rceil$ increases, resulting in a steep drop in D_{gate}. However, note that, while the propagation delay at the ramp peaks remains constant, the width of each ramp and the delay at the ramp trough decrease as W_g increases. Due to this, the delay dependence cannot be modeled accurately by an ideal ramp function. With an increase in the CNT spacing, s, N_{CNT} decreases in steps, and as a result, the gate delay increases in steps, as shown in Fig. 2. The gate height H_g has a negligible impact on the propagation delay.

C. Modeling the Impact of Random Variation Scenarios

In the proposed delay fault test generation method, we select multiple long paths through an instance in a netlist. Our aim is to increase the likelihood that an SDD is detected through a path with minimum slack even in the presence of process variations. For this, we generate random variation

Authorized licensed use limited to: Princeton University. Downloaded on March 11,2023 at 05:34:33 UTC from IEEE Xplore. Restrictions apply.

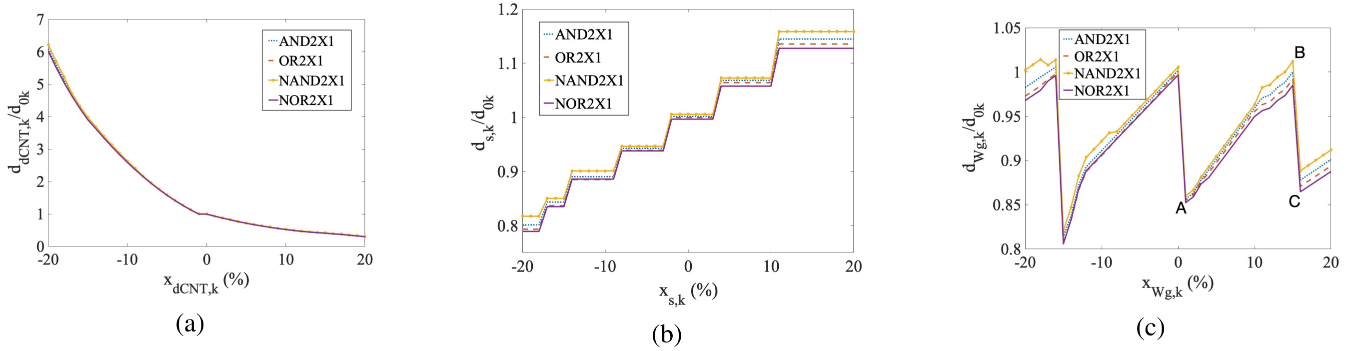


Fig. 2. Propagation delay of various logic gates with variations in (a) d_{CNT}, (b) s, and (c) W_g.

scenarios (RVSs) and calculate the total delay through a path for each scenario.

Definition 1: An RVS is defined as an image of the input netlist where the process parameters for each gate are chosen from a random Gaussian distribution with the corresponding nominal values of the parameter as mean and a predefined standard deviation.

Across different RVS, the process parameters are independently chosen for each gate. This ensures that both interdie and intradie variations are considered for each RVS. However, for a particular RVS, referred to as RVS_m, all the CNFETs in the same gate have the same parameter variations due to close physical proximity.

Suppose that the delay through a path P_j in RVS_m is given by $\delta t_{mj} = \sum_{G_k} D_k$. Here, D_k denotes the propagation delay through gate G_k in the path P_j for RVS_m (under simultaneous variations in all parameters). Note that obtaining the exact value of D_k for any RVS_m requires extensive SPICE simulation for all RVS. This is impractical since the number of possible RVS is exponential in the number of process parameters and the variation range. For example, if we have five parameters, with each parameter varying in the range ±20% from the nominal value with a step size of 1%, the total number of possible RVS per standard cell is 41^5. Obtaining D_k for all these RVS is impractical, and thus, we require a model to predict the gate delay under any variation scenario.

To capture the varying dependencies of the gate delay on different process parameters, we have created a CNFET delay library that has separate lookup tables for each standard cell. Suppose that, in a standard cell G_k, the process parameter p_i has a variation of x_{ik}% from its nominal value μ_i. Using SPICE simulation, we obtain the gate delay d_{ik} under these circumstances and store it in the delay library. The delay contribution of an x_{ik}% variation in the parameter p_i from its nominal value in the gate G_k is given by $c_{ik} = d_{ik} - d_{0k}$, where d_{0k} is the nominal gate delay.

In the mth RVS, RVS_m, there can be simultaneous variations x_{ik} for multiple process parameters. Under such variations, the delay in gate G_k, given by D_k, can then be estimated as $D_k = d_{0k} + \sum_{p_i} c_{ik}$. Note that $c_{ik} = d_{0k}(d_{ik}/d_{0k} - 1)$. From Fig. 2, we observe that d_{ik}/d_{0k} (and therefore c_{ik}) is nonlinear in x_{ik} for d_{CNT}, s, and W_g. Thus, D_k is a nonlinear function of the CNFET device parameters.

In our analysis, we are considering the five CNFET parameters that were shown to have a significant impact on the gate delay [17]. We assume that each parameter can vary in the range ±20% from its nominal value in steps of 1%. Using the delay approximation described above, we need to perform $41 \times 5 = 205$ HSPICE simulations to calculate gate delay over the entire variation space. In Fig. 3, we compare the actual propagation delay with that estimated by our approximation for a few logic gates. For each gate, we have considered 1000 different RVSs where the CNFET parameters vary in an independent Gaussian distribution with their nominal value as the mean μ and a standard deviation $\sigma = 0.05 \cdot \mu$. From the plots, we observe that our approximation based on independent parameter variations can predict the gate delay under random perturbations with reasonable accuracy. To quantify the prediction error for a gate G_k, we calculate the normalized root-mean-square (rms) error given by

$$(\text{rms Error})_k = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\frac{\left|D_{i,k}^{\text{act}} - D_{i,k}^{\text{pre}}\right|}{D_{i,k}^{\text{act}}}} \qquad (2)$$

where $D_{i,k}^{\text{act}}$ ($D_{i,k}^{\text{pre}}$) denotes the actual (predicted) delay of gate G_k in RVS_i. $N = 1000$ is the number of trials (RVS). The rms error for each gate is mentioned in the respective plots in Fig. 3. In addition, the inset of each plot shows the histogram distribution of the relative prediction error, $|D_{i,k}^{\text{act}} - D_{i,k}^{\text{pre}}|/D_{i,k}^{\text{act}}$ over 1000 RVS. Note that, depending on the gate, the prediction error is less than 5% for 60%–80% of all RVS. This shows that the comparatively higher rms error for some gates (e.g., OR2 × 1, AND2 × 1, and XOR2 × 1) is most likely due to a few outlier RVS. The proposed approximate model is particularly appealing as it scales linearly with the number of parameters, as well as the range of variations. In comparison, exhaustive simulation scales exponentially with the number of parameters, and as mentioned earlier, would require 41^5 simulations, which is clearly infeasible in practice. The delay library generation flow is presented in Section IV-A.

IV. Longest Path Selection

Variations in CNFET process parameters affect the propagation delay of logic gates in a netlist, and these faults can be detected using a test set that propagates transitions through the fault sites. In the conventional approach, the minimum-slack testable path through a fault site is used for pattern generation. However, under process variations, the delay through each path changes, and thus, another path through the fault site may have lesser slack than the original selected long path.

Authorized licensed use limited to: Princeton University. Downloaded on March 11,2023 at 05:34:33 UTC from IEEE Xplore. Restrictions apply.

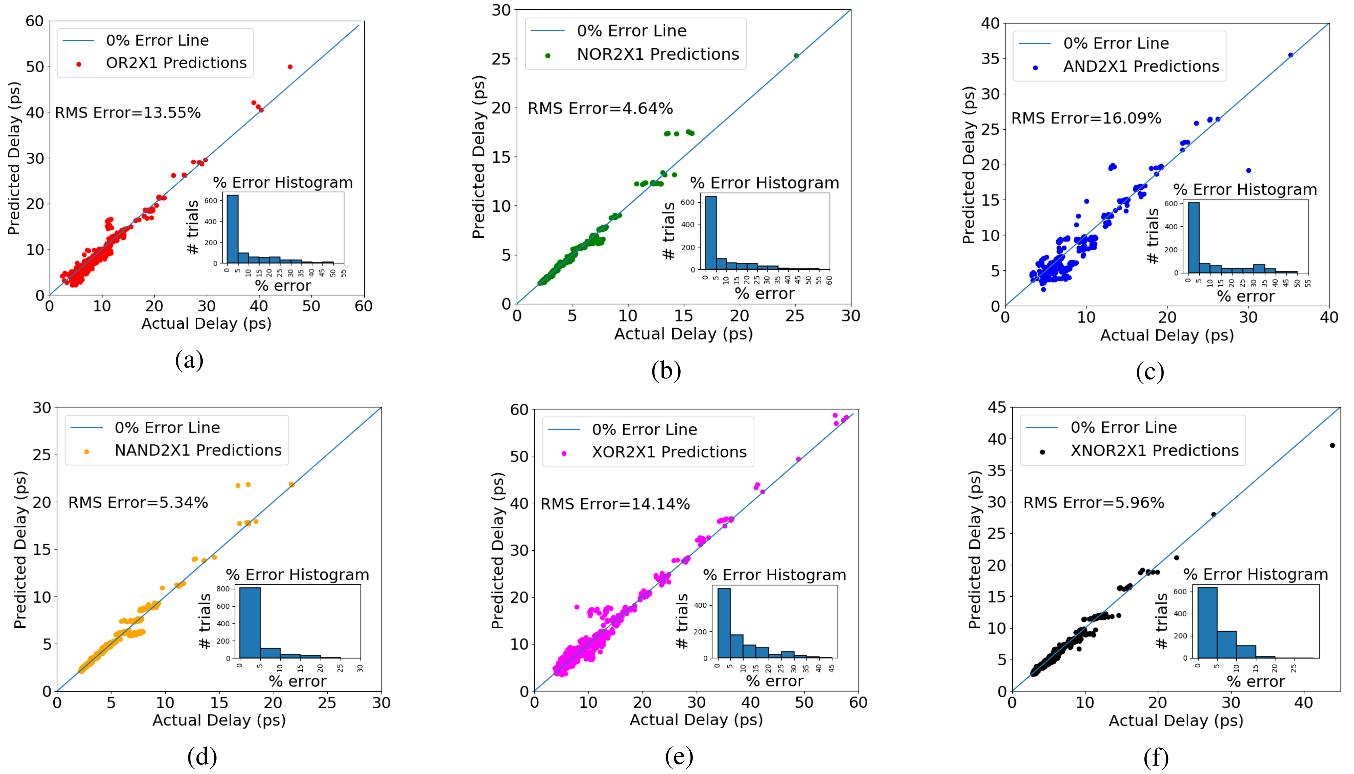


Fig. 3. Comparison between the actual and predicted propagation delay under 1000 RVS for (a) OR2 × 1, (b) NOR2 × 1, (c) AND2 × 1, (d) NAND2 × 1, (e) XOR2 × 1, and (f) XNOR2 × 1. The inset of each plot shows the histogram of the prediction error for the respective gates.

As a result, SDDs might not be propagated through the minimum-slack paths and, therefore, remain undetected. For example, in Fig. 4, suppose P_1 (P_2) is the critical path through the fault site under RVS_1 (RVS_2). If P_1 has a larger nominal delay, the commercial ATPG tool will sensitize P_1 for both these scenarios (RVS_1 and RVS_2), and thus, some SDDs may be undetected in RVS_2. However, in the proposed method, we consider the nonlinear CNFET process variations to shortlist multiple long paths through each gate in the netlist under different random process variation scenarios. As a result, the test patterns obtained using the proposed variation-aware delay fault (VADF) testing method are more likely to propagate an SDD through the longest path, even in the presence of process variations. Therefore, in Fig. 4, VADF test patterns can sensitize both P_1 and P_2. Along with the gate-level netlist, VADF takes the following inputs to achieve this.

1) $\rho_{\max}$: The maximum number of paths to be considered through each gate (node). Note that the tool may not be able to identify $\rho_{\max}$ paths through all nodes.
2) n_s: The total number of RVS that will be simulated by VADF while attempting to find $\rho_{\max}$ longest paths through each node. Increasing n_s results in a higher likelihood of the tool being able to find $\rho_{\max}$ paths for all instances; however, the run time increases as well.
3) PROCESS VARIATION SPECIFICATIONS: The nominal value NV_i, the range of variation $\pm n_i\%$ from NV_i, and the standard deviation σ_i for each parameter p_i.

Fig. 5 shows the flowchart of the proposed longest path selection method. The following steps are sequentially performed

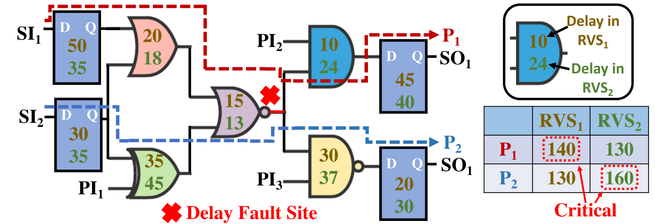


Fig. 4. Critical paths under different RVSs.

to obtain the test patterns to detect SDDs in the presence of parameter variations.

To calculate the propagation delay of a path under different process variation scenarios, we need to analyze how the delay of each gate in the path is affected under these variations. For this purpose, we use HSPICE simulations to create a delay fault library that stores the propagation delay of all standard cells when the different CNFET parameters are independently varied. In the fabrication process, let $p_1, p_2, \ldots, p_N$ be N parameters that are prone to variations, with p_i varying in the range $\pm n_i\%$ from its nominal value, NV_i. Suppose that, for simulation, we consider a step size of $g_i\%$ for each parameter. This means that the value of p_i varies in the following steps: $\{[1-(n_i/100)]\cdot NV_i, [1-(n_i-g_i/100)]\cdot NV_i, [1-(n_i-2g_i/100)]\cdot NV_i, \ldots, NV_i, \ldots, [1+(n_i-g_i/100)]\cdot NV_i, [1+(n_i/100)]\cdot NV_i\}$. For a parameter, the total number of steps, p_i, is, thus, given by $\lfloor(2n_i+1/g_i)\rfloor$. For the kth step, we perform 1000 Monte Carlo iterations where the value of p_i is chosen from a random Gaussian distribution with mean, μ_{ik}, as its value in the kth step. Thus, $\mu_{ik}=[1-(n_i-(k-1)g_i/100)]\cdot NV_i$. The standard deviation

Authorized licensed use limited to: Princeton University. Downloaded on March 11,2023 at 05:34:33 UTC from IEEE Xplore. Restrictions apply.

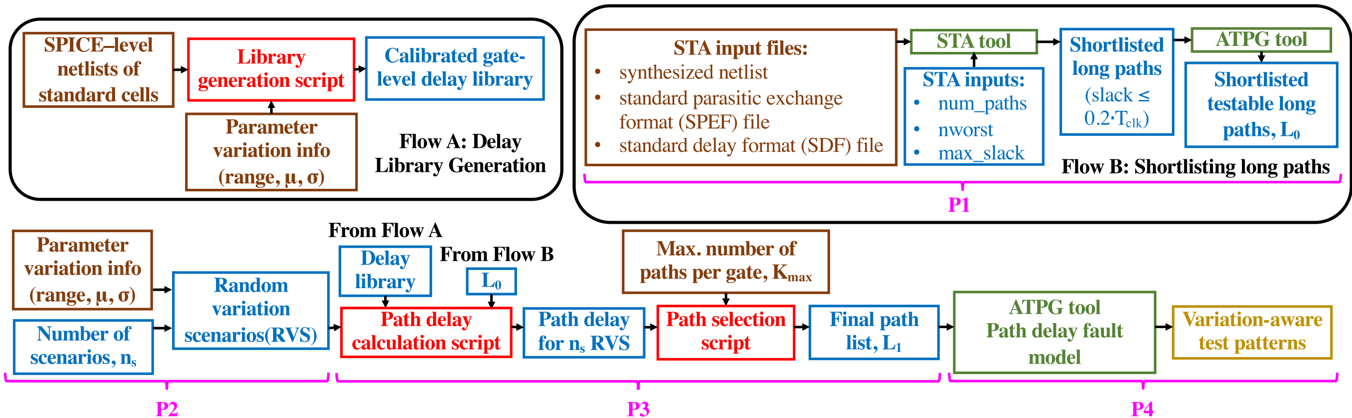


Fig. 5. Flowchart of the longest-path selection using VADF.

$\sigma_{ik} = 0.05 \cdot \mu_{ik}$. During the Monte Carlo iterations, all the other parameters, $p_{j,j\neq i}$, are kept at their nominal values. The mean value of the propagation delay, PD_{ik}, is calculated using HSPICE simulation. This single-parameter variation is considered only for the CNFET delay-library generation to reduce the number of HSPICE simulations required. This assumption is valid because experimental results (see Fig. 3) show that the gate delay under multiple simultaneous variations (RVS) can be predicted with reasonable accuracy by superposing the impact of each individual parameter variation.

A. CNFET Delay Fault Library Generation

The total number of Monte Carlo runs is, therefore, given by $N_{\text{mc}} = \sum_{p_i} \lfloor (2n_i + 1/g_i) \rfloor$. For our simulations, we consider five CNFET parameters—CNT diameter, CNT spacing, gate length, gate width, and oxide thickness. For each parameter, $n_i = 20$ and $g_i = 1$; thus, we perform 205 HSPICE simulations per gate. For each gate G_k and parameter p_i, the normalized change in the propagation delay (given by c_{ik}/d_{0k}) for a x_{ik}% variation from the nominal value is stored in the corresponding delay-library file. Simulation results (see Fig. 3) show that the c_{ik} values can be used to estimate the delay of the cell within the predefined variation range. The delay fault library needs to be generated once for each standard cell and can be used by all designs synthesized using the same cell library.

B. Preprocessing

The total number of paths in a design scales exponentially with the number of gates [21]. Thus, to reduce runtime, we need to create a preliminary shortlist of "long" testable paths denoted by L_0. A path is considered to be "long" if the timing slack on the path is a small (user-defined) fraction of the clock period. The final set of paths selected for test generation, denoted by L_1, is a subset of L_0. To create L_0, we have used a commercial static timing analysis (STA) tool to obtain an appropriate set of paths in a netlist and shortlisted them based on the following criteria.

1) *Criterion 1:* All the paths in L_0 have a nominal slack (in the absence of process variations) that is less than a threshold that is determined based on the path profile of the design. Paths with a high slack margin are not considered because SDDs on those paths are unlikely to affect the circuit functionality. For our simulations, we have considered paths with nominal slack less than 20% of the clock period.
2) *Criterion 2:* All the paths in L_0 are ATPG-testable. Using L_0, VADF selects a subset L_1 of paths that must be tested to cover SDDs at all nodes in the netlist. To ensure the coverage of maximum process variation scenarios, we ensure that all paths recommended for each node are testable. Note that, here, we do not specifically consider robust or nonrobust sensitization. While nonrobust tests are prone to fault-masking, robust tests may fail to detect a delay fault that is detectable by a nonrobust test [42]. Our goal is to consider critical paths under most RVSs; therefore, we do not limit ourselves only to paths that can be robustly sensitized. In addition, we generate test patterns to sensitize multiple long paths through each fault site. Thus, it is likely that at least one of these long paths will lead to a robust test.

We first use a commercial synthesis tool to insert scan chains in the netlist. Two-step preprocessing is then performed to generate L_0 for the scan-inserted netlist.

1) STA Run: In the first step of the preprocessing flow, we obtain the timing characteristics of the design in the absence of any process variation using a commercial STA tool (Synopsys PrimeTime [43]). To determine the appropriate rated clock period, we first run STA with a 1-ns clock and observe the slack on the longest path. The clock period is subsequently adjusted for the rest of the flow by considering a 1% positive slack margin for the longest path. For example, suppose that the longest path has a slack sl_{long} ns when a 1-ns rated clock is used. For the rest of the analysis, we use a clock period given by $t_{\text{clk}} = 1.01 \cdot (1 - sl_{\text{long}})$ ns.

Next, we rerun the STA tool iteratively using a clock with period t_{clk} and obtain the set of longest paths. In each iteration, the following STA inputs are modified until the number of paths in the list remains constant for consecutive iterations:

1) *num_paths:* The maximum number of long paths generated by the STA tool. In our flow, the starting value of num_paths is equal to the number of instances in the netlist. In each subsequent iteration, we increase the value by 20%.

Authorized licensed use limited to: Princeton University. Downloaded on March 11,2023 at 05:34:33 UTC from IEEE Xplore. Restrictions apply.

2) *nworst:* The maximum number of paths ending at each scan flop. In our flow, the starting value of nworst is equal to (number of instances/50). For each subsequent iteration, we increase nworst by 20%.
3) *max_slack:* The maximum slack among all the paths generated. In our flow, we keep it fixed at $0.2 \times t_{\text{clk}}$.

2) Shortlisting of Testable Long Paths, L_0: After the STA run, path delay faults are added to each path, and using Synopsys TetraMAX [43], the ATPG-Untestable path-delay faults are identified, and the corresponding paths are removed from the path list. Therefore, all the paths in the final list satisfy both Criterion 1 and Criterion 2.

C. Selection of Longest Paths for Random Variation Scenarios

The propagation delay of each path in L_0 is calculated for a total of n_s RVSs. In any given variation scenario, for each gate, the CNFET process parameter p_i is randomly chosen from a Gaussian distribution with mean μ_i as its nominal value and standard deviation σ_i and the distribution truncated in the range $[(1-n_i/100)\cdot\mu_i, (1+n_i/100)\cdot\mu_i]$. The process parameters for each gate are chosen randomly independent of the placement of the gate; this ensures that both interdie and intradie variations are taken into account. As the selection procedure for paths to detect delay faults is independent of the placement, the proposed method needs to be executed only once, even if there are multiple place-and-route iterations. However, as mentioned previously, all the transistors in a gate have been assumed to have similar variations in any specific scenario.

The CNFET delay fault library is used to calculate the propagation delay of the logic gates under the different variation scenarios. Consider a gate G_k with a nominal delay d_{0k} and process parameters $p_1, p_2, \ldots, p_n$. The delay contribution of the process parameter p_i is given by $c_{ik} = d_{ik} - d_{0k}$, where d_{ik} is the gate delay in the presence of variation in p_i as obtained from the delay library. The gate delay in the presence of simultaneous variations in CNFET parameters can be approximated by $D = d_{0k} + \sum_{p_i} c_{ik}$. The total delay of a path in a variation scenario is obtained by adding the gate delays of the standard cells in the path.

Once the delays of the paths in L_0 are calculated for the n_s scenarios, a subset of paths, L_1, is selected for test generation. For each standard cell, the paths through the cell are sequentially added to L_1 and removed from L_0 in descending order of propagation delay until one of the following happens: 1) L_1 contains $\rho_{\max}$ paths through the cell and 2) there are no paths through the cell remaining in L_0. At this point, the path-selection procedure terminates. The number of iterations for long-path selection is $O(n_s \cdot |L_0|)$. The actual number of iterations required approaches this limit as $\rho_{\max}$ increases.

D. Test Generation for Selected Long Paths

We use a commercial ATPG tool in the Launch-on Shift timing mode to generate test patterns to detect delay faults through the selected paths in L_1. Based on the steps for longest path selection, the SDD ATPG flow using VADF can be divided into four phases, as shown in Fig. 5.

1) *P1:* Preprocessing step to shortlist long testable paths (see Section IV-B).
2) *P2:* Generating n_s RVS (see Section IV-C).
3) *P3:* Calculate delay of each shortlisted path in L_0 for n_s RVS and select the final list of paths L_1 (see Section IV-C).
4) *P4:* Test generation for selected long paths (see Section IV-D).

The delay-library generation (Flow A in Fig. 5) is not included in these four phases as it needs to be performed only once for each standard cell library.

V. Analysis and Simulation Results

A. Simulation Setup

Using the SDQL metric, we compare the effectiveness of the test patterns generated by VADF with the pattern set from a commercial ATPG tool and a related technique from academia [28]. In the commercial ATPG tool, test patterns are obtained using a "constrained" transition delay fault model. For effective SDD testing, the tool ensures the sensitization of long paths by allowing the user to set the maximum timing margin on any sensitized path (max_tmgn). Setting max_tmgn to a sufficiently low value increases test effectiveness; however, this results in increased ATPG effort and test pattern count. Delay defects of size greater than 20% of the rated clock period can be detected more easily using conventional transition delay patterns. Therefore, we consider max_tmgn $= 0.2 \cdot t_{\text{clk}}$, where t_{clk} is the rated clock period.

No VADF testing technique for CNFET designs has previously been proposed; therefore, for comparison, we considered a modified version of a similar method targeted toward Si CMOS circuits [28]. In this method, variations are considered in metal width, metal thickness, ILD thickness, and gate length of Si-MOSFETs. These variations have a linear impact on the propagation delay, and thus, the path delay can be expressed as a linear function in the parameter variations. The authors utilize this linearity to map the longest path selection problem to the feasibility problem in linear programming. The set of long paths is then generated in $O(n)$ time, where n is the number of testable paths through a fault site.

While the above method performs well for Si technology, the authors themselves point out several drawbacks that are especially critical for emerging technology, such as CNFETs. While calculating the path delay, the same parameter variations have been assumed in all the gates on the path, therefore ignoring intradie variations. Also, the method can be applied only to those parameter variations that have a linear impact on the delay. As we show in Fig. 2, this does not hold for all the CNFET parameters.

To apply the method proposed in [28] to CNFET circuits, we first express the gate delay as a linear function of the parameter variations. R-squared (R^2) is a well-known statistical metric used in linear regression [44]; we use it to compute the goodness-of-fit between the actual gate delays and the gate delays predicted by the linear model. Note that

Authorized licensed use limited to: Princeton University. Downloaded on March 11,2023 at 05:34:33 UTC from IEEE Xplore. Restrictions apply.

TABLE II
GOODNESS-OF-FIT OF THE LINEAR MODEL FOR VARIOUS CNFET PARAMETERS

Gate	$R^2_{d_{CNT}}$	R^2_s	$R^2_{W_g}$	$R^2_{L_g}$	$R^2_{t_{ox}}$
AND2X1	0.8126	0.9696	0.0066	0.9877	0.9996
OR2X1	0.8143	0.9684	0.0063	0.9966	0.9868
NAND2X1	0.8090	0.9696	0.0093	0.9791	0.9764
NOR2	0.8158	0.9651	0.0143	0.9664	0.9723

the R^2 metric is different from (rms Error)$_k$ defined earlier in Section III-C. The R^2 values when variations are introduced in different CNFET parameters for various gates are shown in Table II. Variations in L_g and t_{ox} have a linear impact on the delay [17]; this is reflected by the high value of $R^2_{L_g}$ and $R^2_{t_{\text{ox}}}$. Variation in CNT spacing s has a step impact that can nevertheless be satisfactorily modeled by a linear function. However, the low values of $R^2_{d_{\text{CNT}}}$ and $R^2_{W_g}$ highlight the limitation of the linear regression model. Using the linear model, path delays are predicted under different process variation scenarios, and at most $\rho_{\max}$ long paths through each node are identified. Finally, test patterns are generated for path delay faults in the shortlisted paths. Significant inaccuracies are introduced when we model CNFET variations using a linear model; this, in turn, degrades the effectiveness of the test patterns.

Thus, in the proposed VADF method, we use the CNFET delay library to model the path delay under CNFET variations. The delay library is a collection of gatewise lookup table on how each parameter affects the propagation delay; this is obtained using HSPICE simulation of each gate and, thus, minimizes the modeling error. The test patterns are generated for delay faults on the final shortlisted paths in L_1.

B. SDQL Under Parameter Variations

Effective SDD detection requires the sensitization of faults through paths with minimal timing slack. As discussed in Section I, SDQL is a computationally tractable metric that takes the timing margin on the sensitized path into account [12]. To simulate the effectiveness of test patterns under process variations, a delay-defect distribution function $F(s)$ is considered, where s is the delay fault size in nanoseconds. Therefore, $F(s)$ denotes the probability that a transition delay fault of size s exists at a random node. Note that $F(s)$ is typically obtained from empirical data or a process test chip; some examples of such distributions can be found in [12]. SDD detection becomes more relevant when more defects are of small size, i.e., $F(s)$ decreases rapidly with increasing s.

We first revisit the SDQL metric to derive some theoretical results related to VADF. Consider an SDD fault X and a corresponding test pattern TP$_X$ for it. Let the timing margin of the longest (minimum-slack) path through X in the absence of any parameter variations be given by $T^\star_m$. Similarly, let the nominal slack through the path sensitized by TP$_X$ be $T^\star_d$. Therefore, faults of size $s < T^\star_m$ are redundant, whereas all faults of size $s > T^\star_d$ can be detected be TP$_X$. The probability that an irredundant fault at X remains undetected is, therefore, given by $P^\star_X = \int_{T^\star_m}^{T^\star_d} F(s)ds$. The nominal SDQL for a test pattern set is then given by

$$\theta^\star = \sum_{j=1}^{2N} P^\star_{X_j} = \sum_{j=1}^{2N} \int_{T^\star_m}^{T^\star_d} F(s)ds \quad (3)$$

where the number of nodes (delay faults) in the netlist is $N(2N)$. The probability of SDD escape decreases with decreasing $\theta^\star$. Conventional SDD ATPG tools attempt to generate a test-pattern set that minimizes $\theta^\star$.

TABLE III
COMPARISON OF CPU TIMES FOR SDD ATPG USING VADF ($\rho_{\max} = 10$) AND THE COMMERCIAL ATPG TOOL. THE CPU TIME FOR VADF IS SEGMENTED INTO FOUR PHASES (SEE FIG. 5)

Benchmark	VADF					Commercial ATPG Tool (h)
	P1 (h)	P2 (h)	P3 (h)	P4 (h)	Total (h)	
AES	35.96	0.08	123.37	0.94	160.35	0.07
Ethernet	5.09	0.01	6.02	0.20	11.32	0.05
usb_funct	0.17	0.00	0.05	0.03	0.25	0.06
Rocketcore	4.95	0.02	8.95	0.47	14.39	0.11
vga_lcd	0.47	0.00	1.27	0.02	1.76	0.21
OR1200	4.17	0.03	5.69	0.02	9.91	0.43

We showed in Section III-B that CNFET parameter variations have a significant impact on the gate delay, which affects the timing slacks on the longest and sensitized paths, as well as the SDQL. Let the SDQL under a random parameter variation scenario be given by $\tilde{\theta}$. We have proven that $\mu(\tilde{\theta}) \geq \theta^\star$, or in other words, it is likely that the SDQL value increases under parameter variations (details are provided in the Appendix). The practical significance of this result is that a test set generated without considering CNFET process variations will lead to higher SDQL (lower test quality) than a variation-aware test. Let D_k be a random variable that denotes the propagation delay of G_k under a process variation scenario. From Fig. 2, we observe that CNFET parameter variations can result in both positive and negative deviations in the gate delays. However, in Lemma 1 (stated in the Appendix), we show that, under multiple parameter variation scenarios, the expected gate delay (and in turn path delay) increases from its nominal value. Due to this asymmetric impact, CNFET circuits are more susceptible to timing failures under random parameter variations; this necessitates the use of variation-aware SDD test patterns for yield improvement.

Theorem 1 establishes that the conventional definition of SDQL, as given by (3), underestimates (overestimates) the SDQL (effectiveness) of the SDD test patterns under CNFET process variations. For different variations, the SDQL values are distributed over a range, and therefore, we use the mean SDQL, $\mu(\tilde{\theta})$, as a more accurate measure of test effectiveness when comparing the three test generation methods in Section V-D.

C. Benchmarks

We have considered multiple IWLS'05 benchmarks and the OpenRISC 1200 (OR1200) CPU core, with CNFET logic gates in their netlists. Note that Lu *et al.* [28] used the ISCAS benchmarks; however, we have not used them because the circuits are relatively small and ATPG can sensitize the longest path through all their nodes with minimum effort [6].

The clock frequency and NAND2-equivalent gate count for each benchmark are presented in Table IV. The available

Authorized licensed use limited to: Princeton University. Downloaded on March 11,2023 at 05:34:33 UTC from IEEE Xplore. Restrictions apply.

gate-level netlists for the Opencores benchmarks (AES, Ethernet, usb_funct, and vga_lcd) are synthesized using the 180-nm GSCLib library. However, our CNFET delay library is created at the 45-nm node; therefore, we synthesized the benchmark RTLs using the 45-nm Nangate Open Cell Library. The OR1200 benchmark is used to show the applicability of the proposed method on large designs. We performed simulations on a server with 64-GB 1066-MHz RAM and two 2.53-GHz Intel E5630 Xeon CPUs with four cores each.

Table III shows the execution time for pattern generation using VADF with $\rho_{\max} = 10$. The total CPU time for VADF is divided into the four phases, P1, P2, P3, and P4, as shown in Fig. 5. The pattern generation time for the commercial ATPG tool for each benchmark is also provided for reference. From the simulation results, we observe that P1 and P3 are the major components in the total runtime. This is expected, as finding long paths in a large netlist and calculating their delay under random variations is computationally expensive. Consequently, the CPU time does not necessarily scale with the size of the design but rather with the increasing difficulty of finding long testable paths. This, in turn, depends on the design topology and is difficult to predict without extensive simulations.

Our results show that, for most benchmarks, which are of nontrivial sizes, the pattern generation time using VADF is within acceptable limits. Observe that the CPU time for AES is significantly higher compared with the other benchmarks; this is due to the large latency associated with identifying the long paths using PrimeTime in P1. Note that the run time for path selection is a one-time investment for a particular design, which can result in a significant reduction in manufacturing test time (due to the reduction in delay fault pattern count). Moreover, the CPU time reported here is for a modest university-level computational environment and software developed by university researchers; it can conceivably be reduced by an order of magnitude or more in industry settings.

D. Results and Discussion

Test sets have an inherent tradeoff between the pattern count and effectiveness (in our case, SDQL). We have performed two experiments to evaluate this tradeoff and compare the performance of VADF with a state-of-the-art academic method [28] and a commercial ATPG tool. Commercial ATPG tools today use sophisticated delay models for efficient pattern generation; to compare our method with such models, we have used Synopsys TetraMAX for the commercial SDD ATPG flow. Next, we will describe these two experiments and present the simulation results demonstrating the effectiveness of the proposed method.

1) Experiment E1—Standalone SDD ATPG: In VADF, the test set TP_A is generated for delay faults in all paths on L_1 using the path delay fault model in the ATPG tool. Similarly, the ATPG tool is used to generate the test set TP_B for path delay faults in all paths selected by [28]. The pattern set TP_C is obtained using the commercial SDD ATPG flow where a transition delay fault model is used. As discussed in Section V-A, we consider max_tmgn $= 0.2 \cdot t_{\mathrm{clk}}$, where t_{clk} is the rated clock period. This ensures SDD detection via long paths. While generating TP_B and TP_C for E1, we constrain the ATPG tool to ensure that $|\mathrm{TP}_A| = |\mathrm{TP}_B| = |\mathrm{TP}_C|$; this results in a comparison between three pattern sets of equal size. We then compute $\mu(\tilde{\theta})$ for 100 random process variation scenarios for the three pattern sets for different $F(s)$.

For each benchmark considered in E1, 100 RVSs are simulated. In each scenario, the propagation delays of all "nodes of interest" are modified in the standard delay format (SDF) file. A "node of interest" is one that has at least one path with slack $<0.2 \cdot t_{\mathrm{clk}}$ through it. Parameter variations at these nodes affect delays of the critical paths. Short (large-slack) paths are easily sensitized [45]; therefore, a delay test for a node on such paths can sensitize other near-critical paths. However, test patterns targeted toward long paths involve significant justification and backtracking effort; therefore, such patterns are fault-specific and are less likely to sensitize delay faults on other near-critical paths. This results in a degraded SDQL when parameter variations occur because the set of long paths that variation-unaware ATPG tools target may not include the actual long paths under parameter variations. Therefore, we inject variations in the "nodes of interest" to accurately evaluate VADF and the other two test-generation methods, even when the set of long paths changes under variations.

From Lemma 1, we know that the expected gate delay under CNFET parameter variations is always greater than the nominal delay. To ensure this, we increase the delay of all nodes of interest by a positive value chosen from the positive half of a Gaussian distribution with mean, $\mu = 0$, and $\sigma = 0.05 \cdot t_{\mathrm{clk}}$. SDDs can arise due to a number of issues besides CNFET parameter variations—these include resistive shorts and opens, power supply variations, voltage droop, and coupling faults. We consider a Gaussian distribution when inserting random delay faults in the design to demonstrate the performance of test generation methods when multiple sources lead to random SDDs. Fault simulation is then performed for the three test sets obtained using VADF, [28], and the commercial ATPG tool, with transition delay faults added at the nodes of interest. The mean, $\mu(\tilde{\theta})$, and standard deviation, $\sigma(\tilde{\theta})$, of the SDQL values for the 100 RVSs are then computed. In Table IV, we present the values of $\mu(\tilde{\theta})$ and $\sigma(\tilde{\theta})$ for different values of $\rho_{\max}$ for the benchmarks. Here, the SDD distribution function $F(s) = e^{-s}$, for $0 \leq s \leq \infty$, is the default distribution function considered by the ATPG tool. Note that $F(s)$ is a parameter of the process and can assume different distributions. Therefore, in addition to the default $F(s) = e^{-s}$, we compute the SDQL for the three pattern sets assuming $F(s) = 1.1e^{-1.1s}$. Table V shows the values of $\mu(\tilde{\theta}) \pm \sigma(\tilde{\theta})$ corresponding to the three test sets for $F(s) = 1.1e^{-1.1s}$. The lower value of $\mu(\tilde{\theta})$ in the proposed method shows that VADF can be used for efficient SDD testing irrespective of the probability distribution of the defect sizes.

From Tables IV and V, we observe that the test sets generated using VADF provide considerably lower SDQL compared with the other methods across all the benchmarks. This result is not surprising for CNFET designs because the commercial ATPG tool, and Lu *et al.* [28] do not consider CNFET-specific process variations. VADF can be integrated with commercial ATPG tools to make them CNFET-aware and more effective for CNFET-based designs. Note also that

Authorized licensed use limited to: Princeton University. Downloaded on March 11,2023 at 05:34:33 UTC from IEEE Xplore. Restrictions apply.

TABLE IV

SDD Pattern Effectiveness in Terms of SDQL Values With: 1) $\rho_{\max} = 2$; 2) $\rho_{\max} = 5$; and 3) $\rho_{\max} = 10$. For VADF [28] and the Commercial ATPG Tool, the Corresponding Test Sets Are TP_A, TP_B, and TP_C, Respectively

Benchmark (clock frequency)	Eqv. Gate Count	Test Pattern Count	VADF		[28]		Commercial ATPG Tool	
			$\mu(\tilde{\theta})$ ($\times 10^6$)	$\sigma(\tilde{\theta})$ ($\times 10^6$)	$\mu(\tilde{\theta})$ ($\times 10^6$)	$\sigma(\tilde{\theta})$ ($\times 10^6$)	$\mu(\tilde{\theta})$ ($\times 10^6$)	$\sigma(\tilde{\theta})$ ($\times 10^6$)
AES (1000 MHz)	357.6 K	(a) 340 (b) 570 (c) 736	(a) 17.26 (b) 5.82 (c) 7.21	(a) 0.92 (b) 0.04 (c) 0.35	(a) 137.91 (b) 58.45 (c) 22.35	(a) 8.6 (b) 0.45 (c) 0.16	(a) 60.69 (b) 7.65 (c) 9.63	(a) 2.13 (b) 0.06 (c) 0.06
Ethernet (500 MHz)	99.58 K	(a) 542 (b) 1008 (c) 1010	(a) 3.00 (b) 2.14 (c) 1.67	(a) 0.02 (b) 0.01 (c) 0.01	(a) 44.04 (b) 60.15 (c) 54.45	(a) 0.29 (b) 0.37 (c) 0.38	(a) 35.67 (b) 51.11 (c) 51.73	(a) 0.32 (b) 0.31 (c) 0.36
usb_funct (900 MHz)	22.79 K	(a) 95 (b) 171 (c) 183	(a) 13.86 (b) 8.02 (c) 8.53	(a) 0.08 (b) 0.07 (c) 0.06	(a) 14.38 (b) 10.21 (c) 9.49	(a) 0.09 (b) 0.08 (c) 0.06	(a) 21.89 (b) 12.59 (c) 10.29	(a) 0.13 (b) 0.10 (c) 0.07
Rocketcore (400 MHz)	72.99 K	(a) 107 (b) 272 (c) 290	(a) 8.32 (b) 6.61 (c) 6.69	(a) 0.04 (b) 0.03 (c) 0.02	(a) 46.28 (b) 33.73 (c) 35.01	(a) 0.21 (b) 0.16 (c) 0.18	(a) 26.69 (b) 18.55 (c) 17.44	(a) 0.13 (b) 0.09 (c) 0.11
vga_lcd (480 MHz)	159.06 K	(a) 30 (b) 117 (c) 132	(a) 7.93 (b) 6.78 (c) 6.81	(a) 0.05 (b) 0.04 (c) 0.04	(a) 30.67 (b) 53.97 (c) 49.12	(a) 0.19 (b) 0.34 (c) 0.298	(a) 44.17 (b) 76.44 (c) 68.77	(a) 0.28 (b) 0.48 (c) 0.42
OR1200 (260 MHz)	1.49 M	(a) 37 (b) 71 (c) 90	(a) 2.41 (b) 1.99 (c) 1.85	(a) 0.01 (b) 0.02 (c) 0.01	(a) 17.69 (b) 18.91 (c) 14.37	(a) 0.05 (b) 0.05 (c) 0.04	(a) 13.29 (b) 12.29 (c) 12.73	(a) 0.04 (b) 0.04 (c) 0.04

the mean SDQL decreases for all the methods when $\rho_{\max}$ increases from 2 to 5. With increasing $\rho_{\max}$, the number of paths through each node in L_0 increases, thus increasing the likelihood of sensitization via the longest path, which, in turn, decreases SDQL. With an increasing number of paths to be tested, $|TP_A|$ increases, and as a result, $|TP_C|$ is increased to maintain an equal pattern count for a fair comparison. This increase in $|TP_C|$ results in a lower SDQL value obtained using the commercial ATPG tool.

We observe that, for some benchmarks, $\mu(\tilde{\theta})$ increases when $\rho_{\max}$ is increased from 5 to 10. This happens because, for each VADF run, we consider a set of 1000 independently generated RVS to ensure maximum coverage of SDD sites. These RVS guide the path selection; therefore, it is possible that some of the paths selected with $\rho_{\max} = 5$ may not be present in the set of paths selected with $\rho_{\max} = 10$. Therefore, it is possible that, for a particular RVS used during fault simulation, the critical path through a fault site is present in the $\rho_{\max} = 5$ path set but not in the $\rho_{\max} = 10$ path set. The likelihood of such anomalies can be reduced by increasing the number of RVS during path selection. Also, the SDQL of a pattern set can vary with the RVS that is used for fault simulation; for example, SDQL will increase if, in a particular RVS, parameter variations are introduced in hard-to-detect nodes. However, in all cases, VADF test patterns consistently guarantee more efficient SDD detection compared with the other methods. The CPU time increases with increasing $\rho_{\max}$; therefore, the tradeoff between $\mu(\tilde{\theta})$ and CPU time with different values of $\rho_{\max}$ needs to be analyzed to obtain optimal VADF parameters.

The pattern count for commercial SDD ATPG test sets needs to be considerably higher than for VADF to ensure similar SDQL. In Fig. 6, we show how the SDQL ratio $\gamma = \mu(\tilde{\theta}_C)/\mu(\tilde{\theta}_A)$ changes as $|TP_C|/|TP_A|$ increases; $\mu(\tilde{\theta}_C)$ and $\mu(\tilde{\theta}_A)$ denote the mean SDQL of the commercial and VADF pattern sets, respectively. The pattern count for commercial SDD ATPG test sets needs to be considerably higher to ensure similar SDQL ($\gamma \approx 1$). Note that here we considered SDD test pattern sets; comprehensive transition delay test patterns are considered in E2.

TABLE V

SDD Pattern Effectiveness for $F(s) = 1.1e^{-1.1s}$ With $\rho_{\max} = 5$. For VADF [28] and the Commercial ATPG Tool, the Test Sets Are TP_A, TP_B, and TP_C, Respectively

Benchmark	VADF	[28]	Commercial ATPG Tool
	$\mu(\tilde{\theta}) \pm \sigma(\tilde{\theta})$ ($\times 10^6$)	$\mu(\tilde{\theta}) \pm \sigma(\tilde{\theta})$ ($\times 10^6$)	$\mu(\tilde{\theta}) \pm \sigma(\tilde{\theta})$ ($\times 10^6$)
AES	4.81 ± 0.01	47.66 ± 0.08	6.32 ± 0.03
Ethernet	2.23 ± 0.02	62.77 ± 0.07	53.33 ± 0.05
usb_funct	6.93 ± 0.05	7.74 ± 0.07	8.55 ± 0.08
Rocketcore	6.57 ± 0.02	33.55 ± 0.18	18.45 ± 0.08
vga_lcd	6.86 ± 0.01	57.84 ± 0.03	73.22 ± 0.05
OR1200	1.61 ± 0.01	13.26 ± 0.04	8.57 ± 0.02

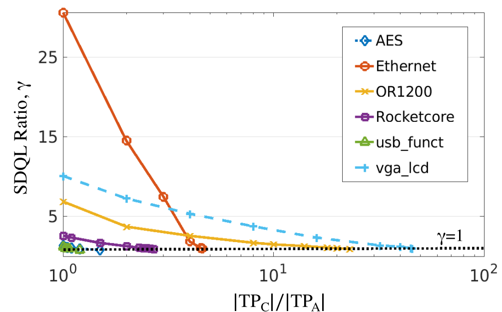


Fig. 6. Variation in $\gamma = \mu(\tilde{\theta}_C)/\mu(\tilde{\theta}_A)$ with increasing pattern count in the commercial ATPG SDD test set.

2) Experiment E2—SDD ATPG With Top-Off Timing-Unaware Transition Delay ATPG: In SDD ATPG, the conventional transition delay testing flow is constrained to specifically sensitize paths where the slack margin is less than a threshold (given by $\max_{\text{tmgn}}$, see Section V-A). Finding test patterns for such long paths is often difficult and computationally expensive. As a result, if SDD test patterns are solely used for delay testing, the fault coverage is low. To remedy this, a hybrid approach is used; SDD ATPG targets faults having relatively small slack on the longest paths through them, while timing-unaware transition delay ATPG is used to target the remaining faults along their easiest-to-sensitize paths [46]. In experiment E2, we compare the performance of VADF with the commercial SDD ATPG tool when this hybrid approach is used. We consider TP_A generated in E1. Using ATPG, we generate the top-off pattern

Authorized licensed use limited to: Princeton University. Downloaded on March 11,2023 at 05:34:33 UTC from IEEE Xplore. Restrictions apply.

TABLE VI

TRANSITION FAULTS TESTING USING VADF ($\rho_{\max} = 10$) AND CNFET VARIATION-UNAWARE COMMERCIAL ATPG TOOL. FOR VADF AND THE COMMERCIAL ATPG TOOL, THE CORRESPONDING TEST SETS ARE TP_A^* AND TP_{CA}^*, RESPECTIVELY

Benchmark	VADF		Commercial ATPG	
	$\|TP_A^*\|$	$\mu(\tilde{\theta}_A^*)$ $(\times 10^6)$	$\|TP_{CA}^*\|$	$\mu(\tilde{\theta}_{CA}^*)$ $(\times 10^6)$
AES	815	7.06	778	8.71
Ethernet	8489	1.66	8743	23.44
usb_funct	801	8.43	755	9.53
Rocketcore	3018	5.59	4027	10.78
vga_lcd	11510	5.07	12772	6.79
OR1200	68424	1.66	64434	9.87

set $\text{TP}_{A'}$ such that $\text{TP}_A^* = \text{TP}_A \cup \text{TP}_{A'}$ can detect all testable transition faults. TP_A^* is, therefore, the comprehensive pattern set generated by VADF for delay fault testing. To compare VADF with conventional CNFET variation-unaware transition delay fault testing, we first use the commercial ATPG tool to generate the test set TP_{CA}. This test set targets SDDs by sensitizing long paths with a slack of $0.2 \times t_{\text{clk}}$ or less but is unaware of CNFET variations. Again, using ATPG, we generate the top-off pattern set $\text{TP}_{CA'}$ such that $\text{TP}_{CA}^* = \text{TP}_{CA} \cup \text{TP}_{CA'}$ can detect all testable transition faults. Note that, while both TP_A^* and TP_{CA}^* can detect all transition faults, TP_A^* covers SDDs more effectively.

Table VI shows results from experiment E2, where we compare the size (pattern count) of TP_A^* with that of the commercial ATPG delay fault test set TP_{CA}^*. For the Ethernet, Rocketcore, and vga_lcd benchmarks, using VADF in the transition fault testing flow results in a smaller pattern count in addition to ensuring effective (low-SDQL) coverage of SDDs. We find that $|TP_A^*| > |\text{TP}_{CA}^*|$ for the AES, usb_funct, and OR1200 benchmarks; however, the difference in pattern counts is low. Also, test sets generated using VADF offer significantly lower mean SDQL ($\mu(\tilde{\theta}_A^*)$) compared with commercial test sets ($\mu(\tilde{\theta}_{CA}^*)$) for all the benchmarks. This shows that the combination of VADF and timing-unaware transition delay ATPG offers more effective SDD detection compared with the commercial tool with a small increase (and often a decrease) in pattern count. Recall also that, if the commercial tool is used instead of VADF, the total pattern count must be considerably higher to ensure similar SDQL (see Fig. 6).

VI. CONCLUSION

Process variations in CNFETs are different from those in Si MOSFETs. Due to the nonlinear and asymmetric nature of the impact of CNFET variations on the propagation delay, Si-based models of parameter variations cannot be extended to CNFETs. We have presented a delay testing method to detect SDDs in the presence of CNFET parameter variations. The proposed method guarantees efficient SDD detection by selecting multiple testable long paths through each fault site; this selection is performed taking various random parameter variations into account. We have shown that the conventional definition of the (nominal) SDQL metric cannot accurately measure the effectiveness of SDD test patterns under random process variations. Toward this end, we have proposed the "variation-aware" mean SDQL metric that considers various process variations scenarios while computing the effectiveness of a test pattern set. This metric is subsequently used to show that the proposed method generates higher quality test patterns compared with a state-of-the-art commercial ATPG tool and related prior work. VADF can be adapted to other emerging transistor technologies besides CNFETs, and it can be used in synergy with commercial ATPG tools to improve delay testing.

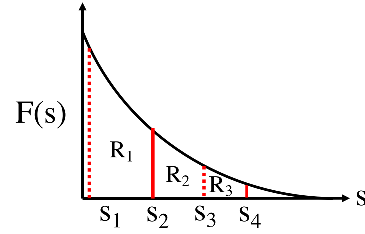


Fig. 7. Area under the curve for calculating SDQL. Here, $s_1 = \mu(\tilde{T}_m)$, $s_2 = T_m^\star$, $s_3 = \mu(\tilde{T}_d)$, and $s_4 = T_d^\star$. From (3), the regions R_2 and R_3 correspond to P_X^*. From (8), the regions R_1 and R_2 correspond to $\mu(\tilde{P}_X)$.

APPENDIX

Lemma 1: The mean propagation delay of a gate G_k under randomly generated variation scenarios is always greater than the nominal delay, i.e., $\mu(D_k) > d_{0k}$.

Proof: In Section III-C, we showed that $D_k = d_{0k} + \sum_{p_i} c_{ik}$, where p_i is a CNFET process parameter and c_{ik} is the deviation due to variation in p_i. Considering the six CNFET parameters, $\mu(D_k) = d_{0k} + \mu(c_{d_{\text{CNT}},k}) + \mu(c_{L_g,k}) + \mu(c_{W_g,k}) + \mu(c_{H_g,k}) + \mu(c_{t_{\text{ox}},k}) + \mu(c_{s,k})$.

I_{ON} is linear in L_g and t_{ox} [17]; thus, the impact of variations in these parameters on the gate delay is symmetric around the nominal (0% variation). Therefore, $\mu(c_{L_g,k}) = \mu(c_{t_{\text{ox}},k}) = 0$. The gate delay is at a local minima at the nominal value of W_g. As a result, for random variation in W_g around the nominal, $\mu(c_{W_g,k}) > 0$. In [17], we showed that the impact of H_g on I_{ON} and the gate delay is negligible. Thus, $\mu(c_{H_g,k}) = 0$. While the variation in gate delay with s is not symmetric about the origin, the impact is largely similar for positive and negative deviations from the nominal value. Therefore, $\mu(c_{s,k}) \approx 0$. On the other hand, note that the impact of variation in d_{CNT} is highly asymmetric about the origin. The increase in the gate delay with a negative deviation in d_{CNT} is significantly higher than the decrease in delay for a positive deviation of similar magnitude. As a result, for random variations, $\mu(c_{d_{\text{CNT}},k}) > 0$.

Therefore, for random variations in all CNFET parameters, $\mu(c_{d_{\text{CNT}},k}) + \mu(c_{L_g,k}) + \mu(c_{W_g,k}) + \mu(c_{H_g,k}) + \mu(c_{t_{\text{ox}},k}) + \mu(c_{s,k}) > 0$. Thus, $\mu(D_k) > d_{0k}$. $\square$

Theorem 1: The mean SDQL of a test pattern set under randomly generated variation scenarios is always greater than the nominal SDQL, i.e., $\mu(\tilde{\theta}) \geq \theta^\star$.

Proof: From Lemma 1, we know that the propagation delay of a gate is likely to increase under CNFET parameter variations. The net delay through a path is the aggregate of the individual gate delays; therefore, the path delay is also expected to increase under parameter variations. Let $\tilde{T}_m(\tilde{T}_d)$ be a random variable that denotes the timing slack on the longest (sensitized) path through a fault site under a process variation scenario. Due to parameter variations, i.e., the mean of the timing slacks of interest, $\mu(\tilde{T}_m) < T_m^\star$, and $\mu(\tilde{T}_d) < T_d^\star$. From Fig. 2, we observe that the relative impact of a particular

Authorized licensed use limited to: Princeton University. Downloaded on March 11,2023 at 05:34:33 UTC from IEEE Xplore. Restrictions apply.

parameter variation on the gate delay, given by D_{ik}/d_{0k}, is similar for various gates. Suppose that, for multiple variation scenarios in a parameter p_i, the mean relative change for any gate is given by $\mu(D_{ik})/d_{0k} = K_i$. Therefore, the mean delay contribution of the parameter p_i on gate G_k is given by $\mu(c_{ik}) = \mu(D_{ik}) - d_{0k} = (K_i - 1) \cdot d_{0k}$. The mean gate delay D_k under multiple variation scenarios is then given by

$$\mu(D_k) = d_{0k} + \sum_{p_i} \mu(c_{ik}) = d_{0k} + \sum_{p_i} (K_i - 1) \cdot d_{0k} = d_{0k}\left(1 + \sum_{p_i} (K_i - 1)\right). \quad (4)$$

Suppose that m gates, $G_1, G_2, \ldots, G_m$, are present on the longest path through the fault site X. The mean path delay of this longest path under parameter variations is, therefore, given by

$$\mu(\tilde{\mathrm{PD}}_L) = \sum_{k=1}^{m} \mu(D_k) = \left(\sum_{k=1}^{m} d_{0k}\right) \cdot \left(1 + \sum_{p_i} (K_i - 1)\right) = \mathrm{PD}_L^\star\left(1 + \sum_{p_i} (K_i - 1)\right) \quad (5)$$

where $\mathrm{PD}_L^\star = \sum_{k=1}^{m} d_{0k}$ is the nominal delay of the longest path. The mean timing margin on the longest path under multiple parameter variation scenarios is given by

$$\mu(\tilde{T}_m) = T_{\mathrm{clk}} - \mu(\tilde{\mathrm{PD}}_L) = T_m^\star - \mathrm{PD}_L^\star \sum_{p_i} (K_i - 1) \quad (6)$$

where T_{clk} is the rated system clock period. From Lemma 1, we have that, for any CNFET parameter p_i, $K_i = \mu(D_{ik})/d_{0k} \geq 1$. Therefore, $\mu(\tilde{T}_m) \leq T_m^\star$. Using a similar approach as above, it can be shown that the mean timing margin on the path sensitized by the SDD test pattern TP_X is given by

$$\mu(\tilde{T}_d) = T_d^\star - \mathrm{PD}_D^\star \sum_{p_i} (K_i - 1) \quad (7)$$

where $T_d^\star$ is the nominal timing margin on the sensitized path and $\mathrm{PD}_D^\star$ is the nominal delay of the sensitized path. Again, $\mu(\tilde{T}_d) \leq T_d^\star$. Let $\tilde{P_X}$ be a random variable that denotes the probability that an SDD at X remains undetected under a process variation scenario. Its expected value under RVSs is then given by

$$\begin{aligned}\mu(\tilde{P_X}) &= \int_{\mu(\tilde{T}_m)}^{\mu(\tilde{T}_d)} F(s)ds \\ &= \int_{\mu(\tilde{T}_m)}^{T_m^\star} F(s)ds + \int_{T_m^\star}^{\mu(\tilde{T}_d)} F(s)ds \\ &\quad + \int_{\mu(\tilde{T}_d)}^{T_d^\star} F(s)ds - \int_{\mu(\tilde{T}_d)}^{T_d^\star} F(s)ds \\ &= \int_{\mu(\tilde{T}_m)}^{T_m^\star} F(s)ds + P_X^\star - \int_{\mu(\tilde{T}_d)}^{T_d^\star} F(s)ds \end{aligned} \quad (8)$$

where $P_X^\star$ is obtained from (3). From (8)

$$\mu(\tilde{P_X}) - P_X^\star = \int_{\mu(\tilde{T}_m)}^{T_m^\star} F(s)ds - \int_{\mu(\tilde{T}_d)}^{T_d^\star} F(s)ds. \quad (9)$$

The first and second integrals in (9) correspond to the regions R_1 and R_3 in Fig. 7, respectively. Using (6) and (7)

$$s_2 - s_1 = T_m^\star - \mu(\tilde{T}_m) = \mathrm{PD}_L^\star \sum_{p_i} (K_i - 1) \quad (10)$$

$$s_4 - s_3 = T_d^\star - \mu(\tilde{T}_d) = \mathrm{PD}_D^\star \sum_{p_i} (K_i - 1). \quad (11)$$

As $\mathrm{PD}_L^\star$ is the delay of the longest path through a fault site, $\mathrm{PD}_L^\star \geq \mathrm{PD}_D^\star$. Thus, $(s_2 - s_1) \geq (s_4 - s_3)$. Moreover, $F(s)$ is a monotonically decreasing function. Therefore, the area of region R_1 is greater than the area of R_3. Using this in (10), $\mu(\tilde{P_X}) - P_X^\star \geq 0$. Therefore, for a circuit with N nodes and $2N$ SDD faults $(X_1, X_2, \ldots, X_{2N})$, the expected value of SDQL under a random parameter variation scenario is given by

$$\mu(\tilde{\theta}) = \sum_{j=1}^{2N} \mu(\tilde{P_X}) \geq \sum_{j=1}^{2N} P_{X_j}^\star. \quad (12)$$

Therefore, using (3), $\mu(\tilde{\theta}) \geq \theta^\star$, and the theorem follows. □

References

[1] G. Hills *et al.*, "Modern microprocessor built from complementary carbon nanotube transistors," *Nature*, vol. 572, no. 7771, pp. 595–602, Aug. 2019.

[2] R. Rao, "Carbon nanotubes and related nanomaterials: Critical advances and challenges for synthesis toward mainstream commercial applications," *ACS Nano*, vol. 12, pp. 11756–11784, 2018.

[3] A. Lin, N. Patil, H. Wei, S. Mitra, and H. S. P. Wong, "A metallic-CNT-tolerant carbon nanotube technology using asymmetrically-correlated CNTs (ACCNT)," in *Proc. VTS*, 2009, pp. 182–183.

[4] J. L. Garcia-Gervacio and V. Champac, "A methodology to compute the statistical fault coverage of small delays due to opens," in *Proc. 52nd IEEE Int. Midwest Symp. Circuits Syst.*, Aug. 2009, pp. 1211–1214.

[5] R. Tayade and J. Abraham, "Small-delay defect detection in the presence of process variations," *Microelectron. J.*, vol. 39, no. 8, pp. 1093–1100, Aug. 2008.

[6] M. Yilmaz, K. Chakrabarty, and M. Tehranipoor, "Test-pattern selection for screening small-delay defects in very-deep submicrometer integrated circuits," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 29, no. 5, pp. 760–773, May 2010.

[7] T. Ni, D. Liu, Q. Xu, Z. Huang, H. Liang, and A. Yan, "Architecture of cobweb-based redundant TSV for clustered faults," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 28, no. 7, pp. 1736–1739, Jul. 2020.

[8] T. Ni, D. Liu, Q. Xu, Z. Huang, H. Liang, and A. Yan, "A novel TDMA-based fault tolerance technique for the TSVs in 3D-ICs using honeycomb topology," *IEEE Trans. Emerg. Topics Comput.*, early access, Jan. 24, 2020, doi: 10.1109/TETC.2020.2969237.

[9] T. Ni, D. Liu, Q. Xu, Z. Huang, H. Liang, and A. Yan, "A cost-effective TSV repair architecture for clustered faults in 3D IC," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, early access, Sep. 25, 2020, doi: 10.1109/TCAD.2020.3025169.

[10] T. Ni *et al.*, "LCHR-TSV: Novel low cost and highly repairable honeycomb-based TSV redundancy architecture for clustered faults," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 39, no. 10, pp. 2938–2951, Oct. 2020.

[11] R. Vollertsen, "Burn-in," in *Proc. Integr. Rel. Workshop Final Rep.*, 1999, pp. 167–173.

[12] Y. Sato, S. Hamada, T. Maeda, A. Takatori, Y. Nozuyama, and S. Kajihara, "Invisible delay quality-SDQM model lights up what could not be seen," in *Proc. ITC*, Nov. 2005, p. 8.

[13] B. Ghavami, M. Raji, H. Pedram, and O. N. Arjmand, "CNT-count failure characteristics of carbon nanotube FETs under process variations," in *Proc. DFT*, Oct. 2011, pp. 86–92.

[14] Y.-M. Lin, J. Appenzeller, Z. Chen, Z.-G. Chen, H.-M. Cheng, and P. Avouris, "High-performance dual-gate carbon nanotube FETs with 40-nm gate length," *IEEE Electron Device Lett.*, vol. 26, no. 11, pp. 823–825, Nov. 2005.

[15] C. Qiu, Z. Zhang, and L.-M. Peng, "Scaling carbon nanotube CMOS FETs towards quantum limit," in *IEDM Tech. Dig.*, Dec. 2017, p. 5.

Authorized licensed use limited to: Princeton University. Downloaded on March 11,2023 at 05:34:33 UTC from IEEE Xplore. Restrictions apply.

[16] C.-S. Lee, E. Pop, A. D. Franklin, W. Haensch, and H.-S.-P. Wong, "A compact virtual-source model for carbon nanotube FETs in the sub-10-nm regime—Part I: Intrinsic elements," *IEEE Trans. Electron Devices*, vol. 62, no. 9, pp. 3061–3069, Sep. 2015.

[17] S. Banerjee, A. Chaudhuri, and K. Chakrabarty, "Analysis of the impact of process variations and manufacturing defects on the performance of carbon-nanotube FETs," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 28, no. 6, pp. 1513–1526, Jun. 2020.

[18] C.-S. Lee and H.-S. P. Wong, "Stanford virtual-source carbon nanotube field-effect transistors model," Dept. Elect. Eng., Stanford Univ., Stanford, CA, USA, Tech. User's Manual, 2015. Accessed: Aug. 10, 2020. [Online]. Available: https://nanohub.org/publications/43

[19] C. Huang, *Robust Computing With Nano-Scale Devices: Progresses and Challenges*, vol. 58. New York, NY, USA: Springer, 2010.

[20] P. Bernardi, M. S. Reorda, A. Bosio, P. Girard, and S. Pravossoudovitch, "On the modeling of gate delay faults by means of transition delay faults," in *Proc. DFT*, Oct. 2011, pp. 226–232.

[21] I. Pomeranz, "A metric for identifying detectable path delay faults," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 31, no. 11, pp. 1734–1742, Nov. 2012.

[22] I. Pomeranz and S. M. Reddy, "Transition path delay faults: A new path delay fault model for small and large delay defects," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 16, no. 1, pp. 98–107, Jan. 2008.

[23] X. Lu, Z. Li, W. Qiu, D. M. H. Walker, and W. Shi, "PARADE: Parametric delay evaluation under process variation [IC modeling]," in *Proc. ISSCS*, Mar. 2004, pp. 276–280.

[24] B. Ghavami and M. Raji, "Failure characterization of carbon nanotube FETs under process variations: Technology scaling issues," *IEEE Trans. Device Mater. Rel.*, vol. 16, no. 2, pp. 164–171, Jun. 2016.

[25] B. C. Paul, S. Fujita, M. Okajima, T. H. Lee, H.-S. Philip Wong, and Y. Nishi, "Impact of a process variation on nanowire and nanotube device performance," *IEEE Trans. Electron Devices*, vol. 54, no. 9, pp. 2369–2376, Sep. 2007.

[26] V. Mehrotra, S. L. Sam, D. Boning, A. Chandrakasan, R. Vallishayee, and S. Nassif, "A methodology for modeling the effects of systematic within-die interconnect and device variation on circuit performance," in *Proc. DAC*, Jun. 2000, pp. 172–175.

[27] A. Agarwal, D. Blaauw, and V. Zolotov, "Statistical timing analysis for intra-die process variations with spatial correlations," in *Proc. ICCAD*, Nov. 2003, pp. 900–907.

[28] X. Lu, Z. Li, W. Qiu, D. M. H. Walker, and W. Shi, "Longest-path selection for delay test under process variation," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 24, no. 12, pp. 1924–1929, Dec. 2005.

[29] R. Brawhear, N. Menezes, C. Oh, L. T. Pillage, and M. R. Mercer, "Predicting circuit performance using circuit-level statistical timing analysis," in *Proc. EDAC-ETC-EUROASIC*, 1994, pp. 332–337.

[30] N. Ahmed, M. Tehranipoor, and V. Jayaram, "Timing-based delay test for screening small delay defects," in *Proc. DAC*, Jul. 2006, pp. 320–325.

[31] A. Srivastava, V. Singh, A. D. Singh, and K. K. Saluja, "A methodology for identifying high timing variability paths in complex designs," in *Proc. IEEE 24th Asian Test Symp. (ATS)*, Nov. 2015, pp. 115–120.

[32] M. M. Shulaker, G. Hills, T. F. Wu, Z. Bao, H. S. P. Wong, and S. Mitra, "Efficient metallic carbon nanotube removal for highly-scaled technologies," in *IEDM Tech. Dig.*, Dec. 2015, pp. 32–34.

[33] N. Patil *et al.*, "VMR: VLSI-compatible metallic carbon nanotube removal for imperfection-immune cascaded multi-stage digital logic circuits using carbon nanotube FETs," in *IEDM Tech. Dig.*, Dec. 2009, pp. 1–4.

[34] N. Patil *et al.*, "Scalable carbon nanotube computational and storage circuits immune to metallic and mispositioned carbon nanotubes," *IEEE Trans. Nanotechnol.*, vol. 10, no. 4, pp. 744–750, Jul. 2011.

[35] N. Patil, A. Lin, J. Zhang, H. S. P. Wong, and S. Mitra, "Digital VLSI logic technology using carbon nanotube FETs: Frequently asked questions," in *Proc. DAC*, Jul. 2009, pp. 304–309.

[36] B. Balaji, J. McCullough, R. K. Gupta, and Y. Agarwal, "Accurate characterization of the variability in power consumption in modern mobile processors," in *Proc. HotPower*, 2012, p. 8.

[37] P. Gupta *et al.*, "Underdesigned and opportunistic computing in presence of hardware variability," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 32, no. 1, pp. 8–23, Jan. 2013.

[38] X. Liu *et al.*, "Detailed analysis of the mean diameter and diameter distribution of single-wall carbon nanotubes from their optical response," *Phys. Rev. B, Condens. Matter*, vol. 66, no. 4, 2002, Art. no. 045411.

[39] Z. Al Tarawneh, "The effects of process variations on performance and robustness of bulk CMOS and SOI implementations of C-elements," Ph.D. dissertation, School Elect., Electron. Comput. Eng., Newcastle Univ., Newcastle upon Tyne, U.K., 2011.

[40] J. Zhang, N. Patil, A. Hazeghi, and S. Mitra, "Carbon nanotube circuits in the presence of carbon nanotube density variations," in *Proc. DAC*, Jul. 2009, pp. 71–76.

[41] B. Bozorgzadeh, S. Shahdoost, and A. Afzali-Kusha, "Delay variation analysis in the presence of power supply noise in nano-scale digital VLSI circuits," in *Proc. IEEE 57th Int. Midwest Symp. Circuits Syst. (MWSCAS)*, Aug. 2014, pp. 117–120.

[42] A. Pierzynska and S. Pilarski, "Non-robust versus robust [test generation]," in *Proc. ITC*, Oct. 1995, pp. 123–131.

[43] *Web Link*. Accessed: Jul. 10, 2020. [Online]. Available: https://news.synopsys.com/2018-03-19-Synopsys-Introduces-Breakthrough-Fusion-Technology-to-Transform-the-RTL-to-GDSII-Flow

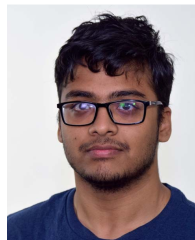
[44] N. J. D. Nagelkerke, "A note on a general definition of the coefficient of determination," *Biometrika*, vol. 78, no. 3, pp. 691–692, Sep. 1991.

[45] S. Eggersglüß and R. Drechsler, *High Quality Test Pattern Generation and Boolean Satisfiability*. New York, NY, USA: Springer, 2012.

[46] (2009). *Small-Delay-Defect Testing*. [Online]. Available: https://www.edn.com/small-delay-defect-testing-2/

Sanmitra Banerjee (Graduate Student Member, IEEE) received the B.Tech. degree from the IIT Kharagpur, Kharagpur, India, in 2018. He is currently working toward the Ph.D. degree in electrical and computer engineering at Duke University, Durham, NC, USA.

He was an Intern with Texas Instruments Inc., Bengaluru, India, and Intel Corporation, Folsom, CA, USA. His current research interests include fault modeling and design-for-testability solutions for emerging technologies, such as carbon nanotube field-effect transistors (FETs) and monolithic 3D ICs.



Arjun Chaudhuri (Graduate Student Member, IEEE) received the B.Tech. degree from IIT Kharagpur, Kharagpur, India, in 2017. He is currently working toward the Ph.D. degree in electrical and computer engineering at Duke University, Durham, NC, USA.

He was an Intern with Global Foundries Inc., Malta, NY, USA, and Alibaba Computing Technology Lab., Sunnyvale, CA, USA. His current research interests include fault modeling, design-for-testability, fault tolerance of machine learning hardware, monolithic 3-D integrated circuits, and emerging technology-driven neuromorphic computing.

August Ning (Graduate Student Member, IEEE) received the B.S.E. degree from Duke University, Durham, NC, USA, in 2020. He is currently working toward the Ph.D. degree in electrical engineering at Princeton University, Princeton, NJ, USA.

He was an intern with Microsoft Corporation, Redmond, WA, USA. His current research interests include VLSI testing and computer architecture.

Krishnendu Chakrabarty (Fellow, IEEE) received the B.Tech. degree from IIT Kharagpur, Kharagpur, India, in 1990, and the M.S.E. and Ph.D. degrees from the University of Michigan, Ann Arbor, MI, USA, in 1992 and 1995, respectively.

He is currently the John Cocke Distinguished Professor, the Department Chair of Electrical and Computer Engineering, and a Professor of computer science with Duke University, Durham, NC, USA.

Dr. Chakrabarty is also a Fellow of Association for Computing Machinery and American Association for the Advancement of Science and a Golden Core Member of the IEEE Computer Society.

Authorized licensed use limited to: Princeton University. Downloaded on March 11,2023 at 05:34:33 UTC from IEEE Xplore. Restrictions apply.