

Statistical Transistor-Level Timing Analysis Using a Direct Random Differential Equation Solver

Qin Tang, *Student Member, IEEE*, Javier Rodríguez, Amir Zjajo, *Member, IEEE*, Michel Berkelaar, *Member, IEEE*, and Nick van der Meijs, *Member, IEEE*

Abstract—To improve the accuracy of static timing analysis, the traditional nonlinear delay models are increasingly replaced by more physical gate models, such as current source models and transistor-level gate models. However, the extension of these accurate gate models for statistical timing analysis is still challenging. In this paper, we propose a novel statistical timing analysis method based on transistor-level gate models. The accuracy and efficiency are obtained by using an efficient random differential equation based solver. The correlations among signals and between input signals and delay are fully accounted for. In contrast to Monte Carlo simulation solutions, the variational waveforms for statistical delay calculation are obtained by simulating only once. At the end of statistical timing analysis, both the statistical delay moments and the variational waveforms are available. The proposed algorithm is verified with standard cells and ISCAS85 benchmark circuits in a 45-nm technology. The experimental results indicate that the proposed method can capture multiple input simultaneous switching for statistical delay calculation, and can provide 0.5% error for delay mean and 2.7% error for delay standard deviation estimation on average. The proposed statistical simulation introduces a small runtime overhead with respect to static timing analysis runtime. The MATLAB implementation of the proposed algorithm has two orders of magnitude speedup, compared to Spectre Monte Carlo simulation.

Index Terms—Correlation, gate model, process variations, random differential equations, statistical timing analysis (STA).

I. INTRODUCTION

As the feature sizes in CMOS technologies continue to shrink, process variations (which influence device and interconnect properties) increase rapidly, causing uncertainty in circuit timing. Predicting the timing uncertainty is traditionally done through corner-based analysis, which performs static timing analysis (STA) at multiple corners to obtain

the extreme-case results. In each corner, process parameters are set at extreme points in the multidimensional space. As a consequence, the worst-case delay from the corner-based timing analysis is overly pessimistic, since it is unlikely for all process parameters to have extreme values at the same time. In addition, the number of process corners grows exponentially with the number of process variations. Recently, statistical STA (SSTA) has been proposed as an alternative to corner-based analysis. In contrast to STA, SSTA represents gate delays and interconnect delays as probability distributions, and provides the distribution (or statistical moments) of timing values rather than deterministic quantities. Typically, there are two types of SSTA techniques: block-based SSTA and path-based SSTA. Block-based SSTA calculates and propagates gate delay and interconnect delay in a leveled breadth-first manner. In path-based SSTA, a set of (near-)critical paths is selected and submitted to the statistical timer for detailed analysis, allowing designers to perform a quality-oriented design pass focused on key paths. Generally, N_{cp} critical paths are enumerated by using fast block-based STA [1]–[5], where N_{cp} is chosen to cover a sufficient spread of paths. Many statistical critical path selection approaches have been proposed [6]–[9].

In both block-based and path-based SSTA approaches, the gate timing models play a significant role for the accuracy-efficiency tradeoff. Many published SSTA methods can be regarded as function-based SSTA, in which the gate delay is modeled as a linear or nonlinear function [10] of process variations, similar to the traditional nonlinear delay model (NLDM) [11] in STA. The coefficients in the function-based statistical gate timing models are characterized and stored in look-up tables with input slew (S_{in}) and effective load capacitance (C_{eff}) as parameters. When calculating statistical gate delay moments, these coefficients are interpolated based on the nominal value of S_{in} and C_{eff} . However, due to process variations, both S_{in} and C_{eff} are variational as well. Not considering the statistical nature of S_{in} and C_{eff} can result in 30% delay errors and even worse for bigger circuits [12]. Also, like NLDM, function-based models do not account for resistive interconnect and nonlinear input waveforms. In addition, the function-based delay representation is entirely based on nonphysical or empirical models, which is their major source of inaccuracy [13].

A large number of more physical gate timing models have been proposed for accurate STA, such as voltage-dependent

Manuscript received October 31, 2012; revised March 7, 2013, June 10, 2013, and August 23, 2013; accepted September 23, 2013. Date of current version January 16, 2014. This work was supported by the European Union and the Dutch government as part of the ENIAC MODERN project (ENIAC-120003). This paper was recommended by Associate Editor S. Vrudhula.

Q. Tang, A. Zjajo, M. Berkelaar, and N. van der Meijs are with the Department of Microelectronics, Delft University of Technology, Delft, The Netherlands (e-mail: qintang@ieee.org).

J. Rodríguez was with the Department of Microelectronics, Delft University of Technology, Delft, The Netherlands. He is now working in Strukton Rolling Stock, Alblasserdam, The Netherlands.

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TCAD.2013.2287179

current source models (CSMs) [12]–[20] and transistor-level gate models [21]–[29]. These gate timing models, denoted as ViVo-gate models, represent every gate by current sources and capacitances with respect to input voltage (V_i) and output voltage (V_o). Most voltage-dependent CSMs target only accurate modeling of combinational gate delay with the assumptions: 1) the input signal is independent of the output signal and 2) single input switching. Hence, they fail to model internal nodes and capacitances, which leads to different undesired symptoms for sequential elements, including nonmonotonic behavior, failure to model storage behavior, etc. [18]. In contrast, the transistor-level gate models can handle sequential circuits in the same way as the combinational circuits without the limiting assumptions of CSMs. They are able to consider multiple input (near-)simultaneous switching (MISS) and to capture the input voltage change caused by the output transition. In addition, the transistor-level gate models have a better defined physical relationship with node voltages and physical parameters. It has been shown that the transistor-level gate models are more general and accurate for timing, noise, and power analysis and practical for multimillion gate STA runs [21]–[29]. Thus, the transistor-level gate models recently attract much attention for accurate static timing analysis.

The voltage-based CSMs and transistor-level gate models have been extended for statistical delay calculation [12]–[15], [25]. In [12], the current source and capacitance values in the CSM are modeled as a quadratic Hermite function of process variations. Several crossing times are characterized based on this quadratic CSM, from which other crossing time distributions are calculated by process variation sampling and linear interpolation. In [13], the variational voltages and all elements in the CSM are modeled as a stochastic first-order expression in terms of process variations, and then the output voltage is treated as a Markovian process for delay distribution calculation. Similarly, a CSM with parametric nonlinear voltage-dependent current source and parametric capacitance is used in [14] and [15]. The voltage in [14] is represented as a time-domain statistical variable and time-domain integration is performed by taking into account input voltage waveform variations. The gate output voltage distribution in [15] is obtained by Monte Carlo (MC) sampling and stored for various voltage levels via regression-based models. Transistor-level gate models are used to estimate the timing variabilities based on corner-based timing analysis in [25]. However, these methods do not take signal correlations and sequential cells into consideration, and most of them are only verified in a few simple gates considering only single input switching. In addition, the solvers proposed for these statistical delay calculations either have difficulties for other gate timing models [13], [14] or require many simulation trails [12], [15], [25].

In this paper, to obtain high accuracy (even for important scenarios such as MISS), we propose a statistical timing analysis solution based on transistor-level gate models to provide statistical delay information. To our knowledge, it is the first time that the ViVo-gate models are extended for statistical timing analysis and achieve sufficient accuracy

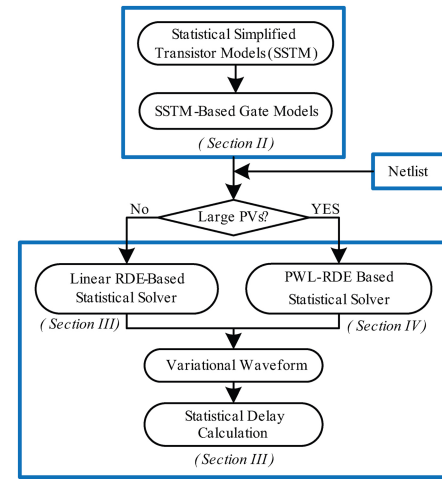


Fig. 1. Flowchart of the RDE-based statistical analysis algorithm.

for both sequential and combinational circuits. The proposed solution has the following features.

- 1) The nonlinear random MNA equation is linearized into a linear random differential equation (RDE). The resulting RDE is solved by an RDE-based solver. The linearity limitation is addressed by using a piecewise linear (PWL)-RDE solver.
- 2) By using the RDE-based solver, the results are obtained by simulating only once, which is much more efficient than MC-based and many-corner-based timing analysis.
- 3) In the proposed RDE based statistical solver, all input signals and their correlations are considered together, thus fundamentally addressing MISS in statistical timing analysis.
- 4) As we use a common format for voltage and current waveforms and passive components (resistances and capacitances) in the gate models, the correlations among input signals and between input signal and delay are preserved during statistical delay calculation.

Extending our previous work published in [30]–[32], this paper adds a more general statistical delay calculation method. Similarly, the linearity limitation encountered in [30]–[32] is overcome by using a PWL-RDE based method. In addition, the results are extended to include sequential circuits. The top-level flowchart of our algorithm is shown in Fig. 1. Measured or simulated data is collected to characterize the statistical simplified transistor models (SSTMs), which are then used to construct the SSTM-based gate model library for timing analysis. After reading the netlist of the circuit, the non-MC RDE-based statistical solver is called for analysis. If high accuracy needs to be maintained for large process variations, the PWL-RDE based statistical solver needs to be used. A variational waveform of the output is provided after using the RDE-based or PWL-RDE based solver, based on which the desired statistical moments of delay are computed.

The rest of this paper is organized according to the flowchart in Fig. 1. Section II presents the SSTM and gate model construction method. Section III introduces the mathematical derivation, calculation flow, statistical delay calculation, and

complexity analysis of the proposed RDE-based statistical timing analysis method. Section IV shows the PWL-RDE based statistical solver. Section V presents the implementation and results, and Section VI provides the conclusions.

II. STATISTICAL SIMPLIFIED TRANSISTOR MODEL

As a key issue for transistor-level gate models [21]–[26], [28], [29], the transistor model needs to capture sufficient second-order effects for accuracy, to account for the impact of process variations, while still being simple enough to be evaluated efficiently. The transistor models in the existing transistor-level timing analysis methods can be categorized into two types.

A. Detailed Transistor Models

For example, BSIM3 with efficient simulation algorithms. The timing analysis in [23] and [24] partitions the circuit into small channel-connected components and then simulates each component individually. A test waveform generation algorithm is introduced in [23], which can capture the worst case timing with only a single simulation. A worst-case waveform propagation was proposed in [27] for timing analysis using BSIM3 models.

B. Simplified Transistor Models

The transistor model for timing analysis in [29] uses look-up tables (LUTs) for drain-source current (i_{ds}) and an input-transition-dependent constant value for five intrinsic capacitances of each transistor. The transistor models in [22], [25], and [26] also use LUTs for i_{ds} , but implement SPICE's version of Meyer's model for the five intrinsic capacitances. By using a linear-centric method, in which the Jacobian matrix is constant for all iterations, the efficiency of transistor-level timing analysis is significantly improved [22], [25], [26], [28]. It is shown in [25] that the LUT-based intrinsic capacitance models are more practical than expensive analytical models.

All the methods mentioned above demonstrate practicability and accuracy of transistor-level timing analysis.

In this paper, we extend the simplified transistor model for statistical gate modeling. In the SSTM shown in Fig. 2 [33], every transistor in the gate is modeled by a current source i_{ds} and five capacitors (c_{gs} , c_{gb} , c_{gd} , c_{sb} , c_{db}). The gate channel capacitances c_{gs} , c_{gb} , and c_{gd} are modeled as a function of V_{gs} and V_{ds} . Since c_{sb} is at least one order of magnitude smaller than the other capacitances and c_{db} is normally smaller than the load capacitance, these two junction depletion capacitances are represented by constant values. In addition to the nominal values of current source and capacitance, SSTM also contains the sensitivities of these model elements to statistical process parameters of interest. These process parameters can be physical process parameters such as effective channel length (L_{eff}) and threshold voltage (V_{th}), or nonphysical parameters derived from dimension-reduction methods, such as principal component analysis (PCA), independent component analysis [34], [35], and reduced rank reduction [36].

The proposed SSTM has simpler characterization requirements and faster characterization setup and runtime, compared

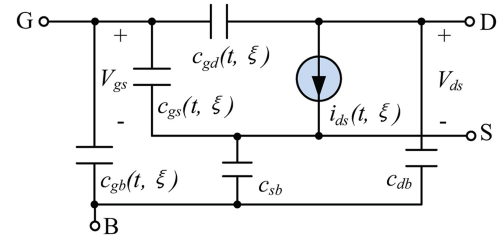


Fig. 2. Proposed SSTM. ξ denotes the process variation vector.

to the CSMs mentioned in Section I. CSMs require transient analysis or AC analysis for different combinations of S_{in} and C_{eff} or different combinations of input and output voltages at different corners. For transistor-level gate modeling, we only need to characterize the unique transistors in the standard cell library, defined by transistor width, source area, and other model parameters. The current and capacitances of SSTM are obtained by a DC sweep at the gate, drain, and source terminals. For statistical analysis, the sensitivities in SSTM are characterized by a finite-difference approximation.

The gate models are constructed by replacing every transistor in the gate by its corresponding SSTM. After RC extraction, model order reduction techniques, such as [37] and [38], can be employed to significantly reduce the complexity of the interconnect model, in which every resistance and capacitance is represented as a linear function of process variations. It should be noted that the statistical timing analysis method presented in Section III is independent of the transistor model used. The proposed SSTM can be extended to include leakage components, such as by adding a leakage current source.

The efficiency of nonstatistical transistor-level timing analysis has been shown to be practical for large digital circuits [25]–[27], [29], [39]. There are two different ways to perform transistor-level timing analysis: block-based and path-based.

For block-based analysis, every gate in the circuit is visited exactly once, in a breadth-first order, starting at the inputs. For each of the gates some amount of work is performed to determine the (maximum/minimum) delay of that gate. In transistor-level analysis, this is a simple simulation task (involving just a single gate and a few vectors) of which the run time does not depend on the circuit size, and can be considered constant. Of course, this constant will be higher for transistor-level models than for simple LUT-based delay models. This leads to a full timing analysis run time linear in the number of gates, allowing application to very large circuits.

For the more accurate path-based transistor-level timing analysis, which is the focus of this paper, typically a fast block-based analysis is performed first to determine the set of near-critical paths that need to be analyzed more carefully. As [40] shows, even statistical path selection for a multimillion gate chip design can be completed in a matter of seconds. This set is typically a few hundred paths. Each of these paths contains at most a few dozen gates, as the maximum depth of the path relates directly to the delay and hence the clock frequency of the design. The number of gates on a path has not changed much over technology generations, as

clock frequencies increase while gate delays decrease. This makes the run time of path based timing analysis mainly dependent on the number of paths to be analyzed, which increases (sub)linearly with the circuit size. All this makes path-based transistor-level timing analysis feasible even for very large circuits.

As we show in Section III-E, the above also applies to our statistical transistor-level timing analysis, as the overhead of making the simulation statistical turns out to be quite small, only a few percent of extra run time for the longest paths.

III. RDE-BASED STATISTICAL TIMING ANALYSIS

In this section, an RDE-based statistical simulation method that directly provides variational waveforms is presented. The theoretical derivation, calculation algorithm, correlation considerations, and statistical delay calculation of this method are introduced. In our mathematical notation, matrices are denoted using bold fonts, and vectors and scalars are represented using normal fonts. In this section, the following notations will be used:

- 1) x , the state variable vector containing node voltages and branch currents;
- 2) $\dot{x}$, its time derivative;
- 3) p_s , the nominal value vector of process parameter p ;
- 4) p_ξ , random process parameter vector;
- 5) δ_0 , the initial condition variation caused by process variations;
- 6) ξ , process variation vector.

A. RDE-Based Statistical Simulation

For ViVo-gate models, such as CSMs and transistor-level gate models in [12]–[15], [21]–[29], nodal analysis (NA) or modified NA (MNA) is used for gate simulation. The NA/MNA approach is deterministic with all parameter values fixed. A typical MNA equation can be written in the following compact format:

$$F(\dot{x}, x, t, p_s) = 0 \quad x(t_0) = x_0. \quad (1)$$

Let $x_s(t)$ be the solution of (1) which satisfies

$$F_s = F(\dot{x}_s, x_s, t, p_s) = 0 \quad x(t_0) = x_0. \quad (2)$$

Since all process parameters have their nominal values p_s , $x_s(t)$ in (2) is deterministic, (2) can be solved by SPICE-like engines or ViVo-gate model based STA methods, such as [23], [29], and [33]. However, the solution becomes statistical if process variations are considered.

Taking into account process variations, (1) becomes a random differential equation as

$$F_\xi = F(\dot{x}, x, t, p_\xi) = 0 \quad x(t_0) = x_0 + \delta_0 \quad (3)$$

$$\xi = p_\xi - p_s. \quad (4)$$

Our objective is to calculate the solution $x(t)$ of (3). For $x(t)$ calculation, MC simulation can be considered the golden reference since it can obtain sufficient accuracy by using several thousands of circuit simulation trials. However, the efficiency of the MC method is insufficient in a design flow,

and thus other statistical timing analysis methods that possess high speedup and similar accuracy are desired. It is desired to calculate statistical properties (e.g., statistical moments) of $x(t)$ directly by solving (3). The difficulties are: 1) the equation has random variables, which differs from ordinary differential equations, and 2) the equation is nonlinear with respect to process variations since the relationship between circuit elements (e.g., intrinsic capacitance and current of each transistor) and process variations is highly nonlinear.

In order to address these two difficulties and solve $x(t)$ from (3), the system is linearized by a first-order truncated Taylor expansion as follows:

$$F_\xi \approx F_s + \frac{\partial F_s}{\partial \dot{x}_s(t)}(\dot{x}(t) - \dot{x}_s(t)) + \frac{\partial F_s}{\partial x_s(t)}(x(t) - x_s(t)) + \frac{\partial F_s}{\partial p_s}\xi = 0 \quad (5)$$

where $F_s = 0$ has been introduced in (2). To simplify the notation, the variation of state variable x is denoted by $y(t) = x(t) - x_s(t)$. Inserting this into (5) and replacing the matrices $\partial F_s / \partial \dot{x}_s$, $\partial F_s / \partial x_s$, and $\partial F_s / \partial p_s$ with $\mathbf{C}(x_s)$, $-\mathbf{R}(x_s)$, and $-\mathbf{Q}(x_s)$, respectively, the following equation can be obtained:

$$\mathbf{C}(x_s)\dot{y}(t) = \mathbf{R}(x_s)y(t) + \mathbf{Q}(x_s)\xi \quad y_0 = \delta_0. \quad (6)$$

The nonlinear equation (3) has now been converted into a linear RDE in y with x_s -dependent coefficient matrices. If N is the number of unknown nodes and N_p is the number of process variations, $\mathbf{C}$, $\mathbf{R}$, and $\mathbf{Q}$ in (6) are $N \times N$, $N \times N$, and $N \times N_p$ matrices, respectively.

Consequently, the solution $x(t)$ is divided into two parts: obtaining $x_s(t)$ from (2) and $y(t)$ from (6). The solution procedure of $x_s(t)$ is the typical deterministic STA [23], [29], [33]. What remains is the difficulty in solving $y(t)$ from the RDE (6).

According to [41, Theorem 7.1.1], (6) has a unique mean square solution that can be represented by (7)

$$y(t) = \Phi(t, t_0)y_0 + \Theta(t)\xi \quad (7)$$

$$= \Psi(t)\xi \quad (8)$$

where $\Phi(t, t_0)$ is the homogeneous solution of (6) satisfying $\mathbf{C}(t)\dot{\Phi}(t, t_0) = \mathbf{R}(t)\Phi(t, t_0)$ and $\Theta(t)$ is an integral in the range $[t_0, t]$, which depends on Φ , $\mathbf{C}$, and $\mathbf{Q}$. If the initial condition x_0 is deterministic, y_0 is zero. On the other hand, even if the initial condition y_0 is statistical, caused by process variations, it could also be represented as a first-order function with respect to ξ . Therefore, $y(t)$ can be written as in (8) where $\Psi(t)$ is an $N \times N_p$ matrix. It is clear from (8) that the coefficient $\Psi(t)$ must be calculated efficiently for $y(t)$.

Although the theorem in [41] points out that the solution process of (7) is defined by the joint probability distributions of all the random variables ξ , solving $\Theta(t)$ directly, however, is in general difficult to achieve. Instead of directly solving $\Theta(t)$ for $\Psi(t)$, we substitute (8) into (6). Then, the equation for $\Psi(t)$ becomes

$$\mathbf{C}(x_s)\dot{\Psi}(t) = \mathbf{R}(x_s)\Psi(t) + \mathbf{Q}(x_s). \quad (9)$$

Now the random differential equation (6) becomes a linear ordinary differential equation in $\Psi(t)$, which can be solved

efficiently by well-known methods. After solving $x_s(t)$ and $\Psi(t)$, $x(t)$ can be obtained in (10) based on $y(t) = x(t) - x_s(t)$ and (8)

$$x(t) = x_s(t) + \Psi(t)\xi. \quad (10)$$

Here, the process variation vector ξ includes both global process variations and local variations. For a specific random process parameter with a global deviation and local deviations, the global deviation and correlated local deviation affect all the transistors in the same way; hence, they can be clubbed together [42]. As an extra optimization, the large number of local process deviations can be converted into a much smaller number of independent local variables by using techniques such as Karhunen–Loève expansion [43] and PCA during analysis. According to [29] and [42], the local variables can be further collapsed to a single variable, such as by treating it in a root of the sum of square fashion, to prevent an explosion in the number of variables. By using these methods, the number of process variations can be reduced significantly.

Based on (10), the time-varying moments of the output voltage can be calculated. The mean and covariance of $x(t)$ are expressed in (11) and (12), respectively

$$\mu_x = E\{x(t)\} = x_s(t) \quad (11)$$

$$\Gamma_{xx} = E\{(x(t) - \mu_x)^2\} = \Psi(t)\Gamma_{\xi\xi}\Psi^T(t) \quad (12)$$

where $z^{*2} = zz^T$. $\Gamma_{\xi\xi} = E\{\xi\xi^T\}$ is the covariance matrix of the process variations and has the following format:

$$\Gamma_{\xi\xi} = \begin{bmatrix} \sigma_1^2 & \rho_{12}\sigma_1\sigma_2 & \dots & \rho_{1N_p}\sigma_1\sigma_{N_p} \\ \rho_{12}\sigma_1\sigma_2 & \sigma_2^2 & \dots & \rho_{2N_p}\sigma_2\sigma_{N_p} \\ \dots & \dots & \dots & \dots \\ \rho_{N_p2}\sigma_{N_p}\sigma_1 & \rho_{N_p2}\sigma_{N_p}\sigma_2 & \dots & \sigma_{N_p}^2 \end{bmatrix} \quad (13)$$

where ρ_{ab} denotes the correlation coefficient between the a_{th} and b_{th} process variations.

B. Analysis Flow

The delay distribution analysis procedure is shown in Algorithm 1. The implementation details of Steps 1–5 are presented as follows.

Step 1: The initial condition x_0 of every gate is obtained from the data characterized in the library according to the switching of nominal input signals (rising, falling, or static). A DC analysis method is also included in the algorithm for generality.

Step 2: In general, for waveform evaluation STA [12]–[15], [21]–[27], [29], in order to solve the nominal waveform $x_s(t)$, modified nodal analysis leads to nonlinear ordinary differential equations or differential algebraic equation system that, in most cases, is transformed into a nonlinear algebraic system $H(x_s) = 0$ by means of numerical integration methods [44]. At every integration step, a Newton–Raphson (NR)-type method is then used to solve $H(x_s) = 0$. If $H(x_s) = 0$ is a set of nonlinear algebraic equations over the variable vector x_s and H is twice differentiable in x_s with a Lipschitz continuous [45] Jacobian matrix $\mathbf{J}_{NR} = \partial H / \partial x_s$, the NR method can be used to solve x_s with the iterations as

$$x_{s,k+1} = x_{s,k} - \mathbf{J}_{NR}^{-1}(x_{s,k}) \cdot H(x_{s,k}) \quad (14)$$

Algorithm 1 Delay distribution calculation

Initialization :

SSTM-based gate models {section II}

Input waveform data (variational or deterministic)

The number k : the k^{th} node output which needs to propagate

Analysis

1. Initial condition x_0

2. STA: solve (2) for nominal value $x_s(t)$

3. Update matrices $\mathbf{C}$, $\mathbf{R}$, $\mathbf{Q}$ based on $x_s(t)$

4. Solve (9) for $\Psi(t)$ by following iterations:

for $j = 1$ to N_p do

if $\xi_j \neq 0$ then

solve $\mathbf{C}(x_s)\dot{\Psi}_j(t) = \mathbf{R}(x_s)\Psi_j(t) + \mathbf{Q}(x_s)u$

else

$\Psi_j(t) = \mathbf{0}$ {0: empty vector}

end if

end for

$\Psi_j(t) = \Psi(t)u$ where u is a $N_p \times 1$ selection vector with only the j_{th} value one.

5. Save the output data for propagation: $v_{nom}(t)$ and $S_{v,\xi}(t)$

6. Calculate the statistical delay {section III-D}

where $x_{s,k}$ is the x_s at the kth iteration. Since $\mathbf{J}_{NR}$ depends on x_s , each iteration of the NR method usually needs to update the Jacobian matrix $\mathbf{J}_{NR}$, calculate the LU decomposition or the $\mathbf{J}_{NR}$ inversion, resolve the equation, and check errors and convergence. By using the NR method, the Jacobian matrix update and the inversion or LU decomposition dominates the runtime of the iterations. However, only the direction of the Jacobian matrix ensures convergence [22], [25], [26], which means that a simpler and less expensive $\mathbf{J}_{NR}$ can be used. In addition, the NR method requires a continuous derivative $\partial H / \partial x_s$. Although our LUT-based transistor model significantly reduces the complexity of the model itself, direct use of LUT-based models to speed up the waveform evaluation with the NR-based algorithm is less efficient, since it requires high-order spline interpolation methods for continuous and smooth partial derivatives. Therefore, the simplified chord (SC) method is used in our algorithm, which uses a constant Jacobian matrix for all iterations at each time step

$$x_{s,k+1} = x_{s,k} - \mathbf{J}_C^{-1} \cdot H(x_{s,k}). \quad (15)$$

Although the SC method has a linear convergence rate, which is slower than the quadratic convergence rate of the NR method, the runtime cost of the SC method is lower in our application.

Step 3: At every time point, once x_s is known, $\mathbf{C}$, $\mathbf{R}$, and $\mathbf{Q}$ are updated and function (9) can be solved to obtain Ψ . However, the high dimension of Ψ and $\mathbf{Q}$ poses additional difficulty, which is solved in Step 4.

Step 4: If the jth process parameter does not have variations or its variation is ignored ($\xi_j = 0$), then $\Psi_j(t) = \mathbf{0}$. Otherwise, ($\xi_j \neq 0$), $\Psi_j(t)$, the sensitivity of the variational voltage to the j_{th} process variation, must be computed. Based on (9), $\Psi_j(t)$ is calculated as

$$\mathbf{C}(x_s)\dot{\Psi}_j(t) = \mathbf{R}(x_s)\Psi_j(t) + \mathbf{Q}(x_s)u \quad j = 1 : N_p \quad (16)$$

where u is an $N_p \times 1$ selection vector whose elements are all zeros except the j th element, which has value one ($j = 1 : N_p$). After using a numerical integration method, due to x_s -dependent coefficients $\mathbf{C}(x_s)$, $\mathbf{R}(x_s)$, and $\mathbf{Q}(x_s)$, (16) becomes a linear algebraic equation with respect to the variable Ψ_j . In our algorithm, we use the trapezoidal method; then, the linear algebraic equation (LAE) is expressed as

$$\mathbf{A}\Psi_j(t_n) = \mathbf{B} \quad j = 1 : N_p. \quad (17)$$

The expressions of $\mathbf{A}$ and $\mathbf{B}$ are formulated as follows:

$$\mathbf{A} = 2 \cdot \mathbf{C}_n - h_n \cdot \mathbf{R}_n \quad (18)$$

$$\mathbf{B} = 2 \cdot \mathbf{C}_n \cdot \Psi_j(t_{n-1}) + h_n \cdot f_{n-1} + h_n \cdot \mathbf{Q}_n u \quad (19)$$

$$f_{n-1} = \mathbf{R}_n \Psi_j(t_{n-1}) + \mathbf{Q}_{n-1} u \quad (20)$$

where $\mathbf{C}_n$, $\mathbf{R}_n$, and $\mathbf{Q}_n$ denote the value of $\mathbf{C}$, $\mathbf{R}$, and $\mathbf{Q}$ at $x_s(t_n)$, respectively, and h_n is the time step that is used to solve $x_s(t_n)$. Therefore, (17) can be solved fast without the necessity of root-finding iterations. Only LU decomposition, and forward and backward substitution are needed to solve the linear algebraic equation. In addition, when j varies from 1 to N_p , only the right-hand side of (17), $\mathbf{B}$, needs to be updated. The same coefficient $\mathbf{A}$ of N_p LAEs in (17) only requires LU decomposition once to solve these N_p LAEs.

Step 5: A voltage of interest, which needs to be stored and propagated (denoted as $v_o(t, \xi)$), can be expressed as

$$\begin{aligned} v_o(t, \xi) &= \chi^T x_s(t) + \chi^T \Psi(t) \xi \\ &= v_{nom}(t) + S_{v,\xi}^T(t) \xi \end{aligned} \quad (21)$$

where χ is an $N \times 1$ selection vector whose elements are all zeros except the k th element has value one ($k = 1 : N_p$), v_{nom} is the nominal voltage, and $S_{v,\xi}$ is $N_p \times 1$ sensitivity vector of output voltage with respect to ξ .

During path-based timing analysis, each critical path can be simulated as a whole to obtain $v_{nom}(t)$ and $S_{v,\xi}^T(t)$ directly for statistical path delay calculation. Gate-by-gate propagation can also be used. For a single transition propagating from gate to gate, $v_{nom}(t)$ and $S_{v,\xi}^T(t)$ of each gate during the transition period (when v_{nom} switches from low to high or from high to low) are propagated. This expresses the voltages as linear functions of the process variables, through which the correlations between voltages are implicitly defined. Using our methods in a traditional block-based SSTA tool that abstracts from transitions and logic values would require the implementation of notions of statistical maximum and minimum voltage waveform/delay over a set of possible transitions. We have not explored this in our research, as it would introduce a lot of inaccuracy, which is at odds with the high accuracy of our transistor-level gate models.

C. Correlations of Variational Waveforms

During statistical timing analysis, the correlation of signals caused by process variations and path reconvergence should be considered and efficiently simulated. Fig. 3 shows the delay standard deviation (σ) of a NAND2 with respect to different correlation coefficients (ρ) of the input arrival times, for several nominal arrival time differences (dt). The arrival time distribution of each input signal has standard deviation $\sigma_t = 10$ ps.

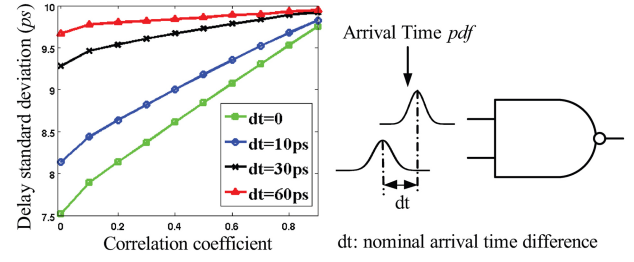


Fig. 3. Importance of correlations. *pdf* denotes probability density function.

It is clear that the correlation of input signals has significant impact on the delay σ when input arrival time distributions are close (e.g., $dt = 0$ in Fig. 3), and it cannot be ignored.

Fig. 3 also shows that when the mean values of arrival times are far away from each other (e.g., $dt = 6\sigma_t = 60$ ps), the delay σ s are almost the same for different correlation coefficients. This leads to the possibility of an extra optimization. In our algorithm, if more than one input switch in a multi-input gate, the 50% crossing times σ of every two switching inputs are calculated and checked. If the signals are not overlapping, the correlation between them will be ignored and the latest/earliest input or inputs will be propagated while the other is assumed static. On the other hand, if they are overlapping, all stochastic correlated inputs are considered.

D. Statistical Delay calculation

Besides statistical simulation, the extraction of statistical delay from variational voltages is also key to extend ViVo-gate models for statistical timing analysis. The methods of existing gate level statistical timing analysis have the following three main categories.

- 1) Interpolation-based analysis. The output waveforms at different corners are simulated in [25], and then the output waveform is characterized by linear interpolation. However, this method assumes that the results at different corners are linear with respect to the process variations and many samples are required for delay calculation.
- 2) MC simulation based on statistical CSMs. The statistical moments of several crossing times are calculated by MC simulations based on statistical CSMs in [12], [15]. However, even though the MC simulations are applied, the accuracy of statistical delay calculation is not competitive due to the over-simplified CSMs.
- 3) Direct calculation based on Markovian process assumption. After calculating the nominal voltage and voltage sensitivities with respect to process variations, the delay distribution is calculated by assuming that the voltage at every time point is a Markovian stochastic process due to the numerical integration method [13], [30], [31]. In order to calculate the distribution of a crossing time, the joint probability of voltage at different time steps is calculated by using the bivariate normal distribution formula, which is erroneous when the Gaussian distribution assumption for voltages is inaccurate.

In our algorithm, we use a more general method for statistical delay calculation. Fig. 4 shows a variational waveform

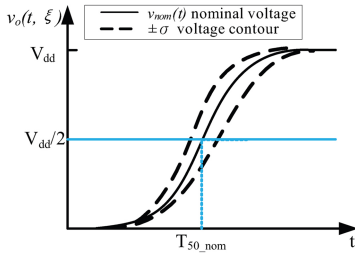


Fig. 4. Variational waveform and statistical crossing time.

where T_{50_nom} is the 50% crossing time of the nominal voltage. Since t_{50} is defined as the time for voltages to cross $V_{dd}/2$, for every possible waveform, the voltage at t_{50} can be written as

$$v_o(t_{50}, \xi) = V_{dd}/2 \quad (22)$$

where $v_o(t, \xi)$, described in (21), is the variational waveform due to the process variations ξ .

The derivative of (22) with respect to process variation ξ can be expressed as

$$\left. \frac{\partial v_o(t, \xi)}{\partial \xi} \right|_{t=T_{50_nom}} + \left. \frac{\partial v_{nom}(t)}{\partial t} \right|_{t=T_{50_nom}} \cdot \frac{\partial t_{50}}{\partial \xi} = 0 \quad (23)$$

where $v_{nom}(t)$ denotes the nominal voltage vector without process variations after using the RDE-based solver in (21). Based on (23), the sensitivity of t_{50} with respect to ξ ($S_{t,\xi}$) can be calculated as

$$S_{t,\xi} = - \frac{\frac{\partial v_o(T_{50_nom}, \xi)}{\partial \xi}}{\frac{\partial v_{nom}(T_{50_nom})}{\partial t}} = - \frac{S_{v,\xi}(T_{50_nom})}{S_{v,t}(T_{50_nom})} \quad (24)$$

where $S_{v,\xi}(t)$ is the $N_p \times 1$ sensitivity vector calculated by the RDE-based statistical solver (21) and $S_{v,t}(T_{50_nom})$ is the slope of nominal value at T_{50_nom} . Based on the sensitivity $S_{t,\xi}$, the statistical 50% crossing time can be obtained as

$$t_{50} = T_{50_nom} + S_{t,\xi}^T \xi. \quad (25)$$

Other statistical crossing times can be calculated in the same way with different threshold voltages. Equation (25) indicates that the crossing time is a first-order function of the process variations. The mean and standard deviation can be calculated based on (25) with

$$\mu_{t_{50}} = T_{50_nom} \quad (26)$$

$$\sigma_{t_{50}} = \sqrt{S_{t,\xi}^T \mathbf{\Gamma}_{\xi\xi} S_{t,\xi}} \quad (27)$$

where $\mathbf{\Gamma}_{\xi\xi}$ is the covariance matrix of the process variations which has been introduced in (13). The correlations of the process variations are considered in the $\sigma_{t_{50}}$ calculation. Since the RDE-based method is based on SSTM-based gate models, all input signals are considered for voltage sensitivity $S_{v,\xi}$ calculation that is used for t_{50} calculation. Therefore, MISS is fundamentally handled in the algorithm.

E. Complexity Analysis

In Algorithm 1, the majority of the runtime is consumed in Step 2 to calculate the nominal value x_s and in Step 4 to compute the sensitivities Ψ . Therefore, $T_{SSTA} \approx T_{STA} + T_\Psi$, where T_{SSTA} is the runtime of the whole statistical timing

analysis algorithm, T_{STA} is the runtime of Step 2 and T_Ψ is the time of Steps 3 and 4.

Step 2 can be solved by ViVo-gate model based STA procedures [12]–[15], [23], [29], [33], and its efficiency depends on the gate models and the simulation methods. The proposed LUT-based transistor model can significantly reduce the runtime of model evaluation [25], [26], [29]. In transistor-level circuit simulation methods, the solving procedure of the circuit equation (2) is mainly the LAE solution at each iteration, including the LU factorization, forward substitution, and backward substitution. By using the proposed SC method, the LU factorization only needs to be done once at every time point, rather than at every iteration. By using sparse matrix techniques, the computational complexity is usually $O(N^\alpha)$, $\alpha \approx 1.2 \sim 1.5$, which is significantly better than the cubic complexity without using the sparse matrix techniques [45].

Compared to the traditional ViVo-gate model based STA, our statistical timing analysis method requires extra runtime $T_\Psi(t)$ to calculate the statistical moments of delay. According to the computation method introduced in Step 4, no root-finding iterations (e.g., NR and SC methods) are necessary to calculate Ψ . Hence, in contrast to T_{STA} , the complexity of T_Ψ is $O(N^\alpha)$ at each time point (no iterations) rather than at each iteration. Although N_p LAEs need to be solved in Step 4, the solving procedure of these N_p LAEs in (16) shares the same coefficient $\mathbf{A}$. Therefore, solving N_p LAEs requires only one LU decomposition, which reduces the runtime of statistical timing analysis significantly.

For the circuit simulation, the runtime is mainly consumed by the following parts: 1) matrix update, including model evaluation and element stamp; 2) LAE solution after LU factorization; 3) LU factorization; and 4) others, including time step calculation, convergence check, etc. In order to analyze the runtime ratio, the differential equation of $x_s(t)$ in (2) can be considered to have an ordinary differential equation format similar to (16). The runtime of solving $x_s(t)$ at every time point is approximately $[N^2 + (2N^2 + N) \cdot N_{it}] \cdot T_{up} + N_{it} \cdot T_{LE} + T_{LU} + T_{ot}$, where T_{up} denotes the runtime for updating an entry in a matrix, T_{LE} denotes the runtime to solve an LAE after obtaining the LU factorization results, T_{LU} represents the runtime of LU factorization of an $N \times N$ matrix, T_{ot} represents the runtime of part 4 at every time point; and N_{it} is the average number of iterations at every time point using the SC method. It is worth noting that the $N \times N$ Jacobian matrix $\mathbf{J}_{SC}$ consists of the derivatives of elements (e.g., currents and capacitances of SSTM) with respect to node voltages.

In contrast, the runtime of solving $\Psi(t)$ at every time point is approximately $(\beta N^2 + NN_p) \cdot T_{up} + N_p \cdot T_{LE} + T_{LU}$, where $\beta < 1$. The β is less than one since some entries of $\mathbf{J}_{SC}$ are reused in $\mathbf{R}$ update, which reduces the runtime for matrix update at every time point in Step 4. Since Step 2 and Step 4 have the same number of time points, the runtime overhead is

$$\frac{T_\Psi}{T_{STA}} \approx \frac{(\beta N^2 + NN_p) \cdot T_{up} + N_p \cdot T_{LE} + T_{LU}}{[N^2 + (2N^2 + N) \cdot N_{it}] \cdot T_{up} + N_{it} \cdot T_{LE} + T_{LU} + T_{ot}}. \quad (28)$$

According to (28), if the circuit is large ($N \gg 1$) and T_{up} dominates the runtime, $\frac{T_\Psi}{T_{STA}} < \frac{1}{1+2N_{it}}$. We can see from $\frac{T_\Psi}{T_{STA}}$

Algorithm 2 Piecewise Linear RDE-Based Statistical Delay Calculation**Initialization :**

SSTM-based gate models, input waveform data and the number k , same as the Initialization in Algorithm 1.

$\xi_s^{(m)}$, the nominal value of process variation in the subspace $\Omega^{(m)}$.

Analysis

for $m = 1$ to M **do**

1. Initial condition $x_0^{(m)}$ when $p_s^{(m)} = p_s + \xi_s^{(m)}$
2. STA: solve (2) for nominal value $x_s^{(m)}$ when $p = p_s^{(m)}$
3. Update matrices **C**, **R**, **Q** based on $x_s^{(m)}$
4. Solve (9) for $\Psi^{(m)}$ similar like Step 4 in Algorithm 1
5. Save the output data: $v_{nom}^{(m)}(t)$ and $S_{v,\xi}^{(m)}(t)$

end for

6. Construct the output signal

7. Calculate the statistical delay {Section III-D}

that the runtime overhead of our statistical simulation is small, especially for large circuits.

IV. PIECEWISE LINEAR RDE-BASED STATISTICAL SIMULATION

In the proposed RDE-based statistical method presented in Section III-A, the stochastic MNA (3) is solved by a first-order Taylor approximation, which implies the assumption that the solution of (3) can be considered linear in the range of ξ variabilities. For instance, we can see from Fig. 5 that if the L_{eff} distribution has a small variance, the linearization is valid. In contrast, if the distribution spans [44 nm, 58 nm], the linear model is not accurate enough. The validity of linearization can be verified by checking whether the truncation error of (5) is smaller than an error threshold. However, it is nontrivial to calculate the truncation error since it involves the 2nd-order derivative of circuit elements, such as transistor intrinsic capacitances and currents. According to our experimental results, for highly nonlinear process parameters such as L_{eff} , when $3\sigma/\mu$ is larger than 10%, the piecewise linear solver is required to maintain the accuracy for statistical timing analysis.

In order to overcome the accuracy problem of nonlinear process variations ξ , we partition the ξ space such that (3) can be considered linear in each subspace. Let $\xi \subseteq \Omega$ where Ω is the domain in which ξ varies and $\Omega^{(m)}, m = 1, 2, \dots, M$ as the mutually exclusive subspaces that form a partition and satisfy i)

$$\bigcup_{m=1}^M \Omega^{(m)} = \Omega \text{ and } ii) \Omega^{(m)} \cap \Omega^{(n)} = \emptyset (m \neq n = 1, 2, \dots, M).$$

For instance, if L_{eff} distribution in Fig. 5 has standard deviation $3\sigma = 6$ nm, we can divide it into three subspaces $[-\infty, 48 \text{ nm})$, $[48 \text{ nm}, 52 \text{ nm}]$, and $(52 \text{ nm}, +\infty]$. The dividing method also depends on the nonlinearity of the process variation. If a process variation is much more nonlinear in a subspace than in other subspaces, we can divide this subspace into more subspaces.

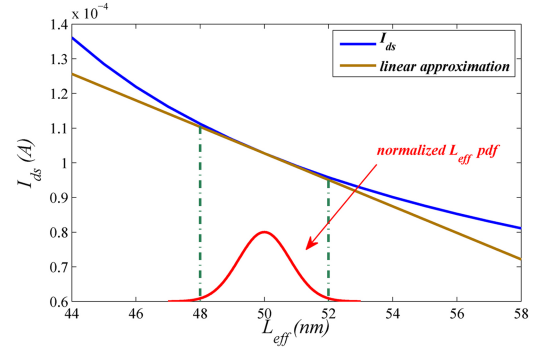


Fig. 5. I_{ds} of nMOS transistor with width 130 nm and nominal length 50 nm when $V_{gs} = V_{DD}$ and $V_{ds} = V_{DD}$. The pdf of L_{eff} variability is normalized to illustrate the position in the L_{eff} x-axis.

According to the law of total probability [46], the probability of x , denoted as $P_r(x)$, can be expressed as

$$P_r(x) = \sum_{m=1}^M P_r^{(m)} P_r(x|\Omega^{(m)}) \quad (29)$$

$$P_r^{(m)} = P_r(\xi \in \Omega^{(m)}) = \int_{\Omega^{(m)}} f_{\xi}(\xi) d\xi \quad (30)$$

where $f_{\xi}(\xi)$ is the probability density function (pdf) of ξ and $P_r^{(m)}$ is the probability of ξ in the subspace $\Omega^{(m)}$. Based on the law of total probability, $x(t)$ can be calculated from the solution in each subspace. The piecewise linear method is a common method to manage nonlinearity. Its application for device mismatch simulation was proposed in [47], and we apply it to our RDE-based statistical simulation method.

The piecewise linear RDE-based statistical delay calculation flow is listed in Algorithm 2. $\xi_s^{(m)}$, being the nominal value of process variation in each $\Omega^{(m)}$, is chosen such that the solution can be considered linear in the neighborhood spanned by the value of $\xi_s^{(m)}$. Since $\xi_s^{(m)}$ is not always zero now, the nominal process parameter in subspace $\Omega^{(m)}$ becomes $p_s^{(m)} = p_s + \xi_s^{(m)}$. Then, the proposed RDE-based statistical method is performed in each subspace. The linearization for statistical MNA (3) is performed at $x_s^{(m)}$ and $p_s^{(m)}$, where $x_s^{(m)}$ is the output curve under $p^{(m)}$. After solving $\Psi^{(m)}$, the following variational output in each $\Omega^{(m)}$ can be obtained

$$x^{(m)} = x_s^{(m)} + \Psi^{(m)} (\xi^{(m)} - \xi_s^{(m)}) \quad (31)$$

where the explicit dependence on time is omitted for notational simplicity. Based on (31), the mean of $x^{(m)}$ can be calculated as

$$\begin{aligned} \mu_x^{(m)} &= E \{ x^{(m)} \}_{\Omega^{(m)}} \\ &= x_s^{(m)} + \Psi^{(m)} E \{ \xi^{(m)} - \xi_s^{(m)} \}_{\Omega^{(m)}} \\ &= x_s^{(m)} + \Psi^{(m)} (\mu_{\xi}^{(m)} - \xi_s^{(m)}) \end{aligned} \quad (32)$$

where $E\{z(\xi)\}_{\Omega^{(m)}} = \int_{\Omega^{(m)}} z(\xi) f_{\xi}(\xi) d\xi$. The covariance of $x^{(m)}$, denoted as $\Gamma_{xx}^{(m)}$, can be derived by substituting (31) for $x^{(m)}$ and (32) for $\mu_x^{(m)}$ to (33), respectively

$$\Gamma_{xx}^{(m)} = E \left\{ \left(x^{(m)} - \mu_x^{(m)} \right)^{*2} \right\}_{\Omega^{(m)}} \quad (33)$$

$$= E \left\{ \left(x_s^{(m)} + \Psi^{(m)} \left(\xi^{(m)} - \xi_s^{(m)} \right) - x_s^{(m)} + \Psi^{(m)} \left(\mu_{\xi}^{(m)} - \xi_s^{(m)} \right) \right)^{*2} \right\}_{\Omega^{(m)}} \quad (34)$$

where $z^{*2} = zz^T$. Let $\Gamma_{\xi\xi}^{(m)}$ be the covariance of ξ in $\Omega^{(m)}$, $\Gamma_{xx}^{(m)}$ can be rewritten in a compact form as

$$\Gamma_{xx}^{(m)} = \Psi^{(m)} \Gamma_{\xi\xi}^{(m)} \Psi^{(m)T}. \quad (35)$$

In (32) and (35), the unknown items, $\mu_{\xi}^{(m)}$ and $\Gamma_{\xi\xi}^{(m)}$, are the mean and covariance of ξ in $\Omega^{(m)}$, and can be written as

$$\mu_{\xi}^{(m)} = \int_{\Omega^{(m)}} \xi^{(m)} f_{\xi}(\xi) d\xi \quad (36)$$

$$\Gamma_{\xi\xi}^{(m)} = \int_{\Omega^{(m)}} \left(\xi^{(m)} - \mu_{\xi}^{(m)} \right)^{*2} f_{\xi}(\xi) d\xi. \quad (37)$$

It is worth noting from (32) that if $\xi_s^{(m)}$ is chosen such that $\xi_s^{(m)} = \mu_{\xi}^{(m)}$, the calculation of $\mu_x^{(m)}$ is simplified to $\mu_x^{(m)} = x_s^{(m)}$. In addition, the minimization of $E \left\{ \left\| \xi^{(m)} - \xi_s^{(m)} \right\|^2 \right\}$ maximizes the chances that the linear approximation around $\xi_s^{(m)}$ has good accuracy in the variability range of $\xi^{(m)}$.

Based on the results in each subspace, the total output can be computed. According to the law of total probability shown in (29), the mean of x can be obtained from

$$\begin{aligned} \mu_x &= E\{x\} = \sum_{m=1}^M P_r^{(m)} \int_{\Omega^{(m)}} x^{(m)} f_{\xi}(\xi) d\xi \\ &= \sum_{m=1}^M P_r^{(m)} \mu_x^{(m)} \end{aligned} \quad (38)$$

where $P_r^{(m)}$ and $\mu_x^{(m)}$ have been introduced in (30) and (32). Similarly, the covariance of x can be calculated as

$$\begin{aligned} \Gamma_{xx} &= E \left\{ \left(x(\xi) - \mu_x \right)^{*2} \right\} \\ &= \sum_{m=1}^M P_r^{(m)} E \left\{ \left(x^{(m)} - \mu_x \right)^{*2} \right\}_{\Omega^{(m)}} \\ &= \sum_{m=1}^M P_r^{(m)} E \left\{ \left(x^{(m)} - \mu_x^{(m)} + \mu_x^{(m)} - \mu_x \right)^{*2} \right\}_{\Omega^{(m)}} \\ &= \sum_{m=1}^M P_r^{(m)} \left(\Gamma_{xx}^{(m)} + \left(\mu_x^{(m)} - \mu_x \right)^{*2} \right). \end{aligned} \quad (39)$$

Then, μ_x and Γ_{xx} can be used to obtain the mean and variance of $v_o(t, \xi)$ [introduced in (21)].

Different from the statistical delay calculation method introduced in Section III-D, $\Psi(t)$ is not calculated by using the PWL-RDE solver. Therefore, $S_{v,\xi}(t)$ is not available and (24) cannot be calculated. However, we can use the similar method

to calculate the mean and variance of t_{50} . According to (27), the variance of t_{50} can be expressed as

$$\sigma_{t_{50}}^2 = S_{t,\xi}^T \Gamma_{\xi\xi} S_{t,\xi}. \quad (40)$$

By substituting the expression of $S_{t,\xi}$ into (24) to (40), we can have the following expression:

$$\sigma_{t_{50}}^2 = \frac{S_{v,\xi}^T(T_{50_nom}) \Gamma_{\xi\xi} S_{v,\xi}(T_{50_nom})}{S_{v,t}(T_{50_nom})}. \quad (41)$$

As defined in (21), $S_{v,\xi}^T(t) = \chi^T \Psi(t)$. Therefore, (41) can be rewritten as

$$\sigma_{t_{50}}^2 = \frac{\chi^T \Psi(T_{50_nom}) \Gamma_{\xi\xi} \Psi^T(T_{50_nom}) \chi}{S_{v,t}(T_{50_nom})}. \quad (42)$$

According to (12), (42) is written as

$$\sigma_{t_{50}}^2 = \frac{\chi^T \Gamma_{xx}(T_{50_nom}) \chi}{S_{v,t}(T_{50_nom})}. \quad (43)$$

Therefore, based on μ_x and Γ_{xx} , the mean of t_{50} is calculated based on (26) and the variance of t_{50} is calculated based on (43).

The computational complexity of the piece-wise RDE-based method is M times the complexity of the RDE-based method presented in Section III-E, since Algorithm 1 is repeated M times in Algorithm 2. Therefore, the optimum number M depends not only on the number of process variations, 3σ of each process variation distribution and the nonlinearity of each process variation for accuracy, but also on the runtime. According to our experiments, L_{eff} is more nonlinear than V_{th} . When $3\sigma/\mu > 10\%$ for L_{eff} , the PWL-RDE solver should be used and we choose $M = 3$ for the experiments.

V. EXPERIMENTAL RESULTS

The proposed statistical timing analysis algorithm was implemented in Matlab and tested on all combinational cells and widely used sequential cells found in the standard cell library of the Nangate 45-nm Open Cell Library Package 2009 [48] and on ISCAS85 benchmark circuits. All experiments were performed on a computer with an Intel core 2 duo CPU with 3-GHz clock speed and 3 GB of memory. The SSTM is characterized by using the BSIM4 model in Spectre. Spectre can provide the necessary intrinsic capacitance values of each transistor after DC simulation.

The Verilog netlists of all ISCAS85 circuits are downloaded from [49] and then mapped to the Nangate 45-nm Technology Library with Cadence Encounter. The parasitic RC models of the wires are extracted from layout and stored in SPF and SPEF files. From each circuit we extract the most critical nonfalse path found by the timing engine in Encounter. The parser in our algorithm reads the Verilog netlist and SPF files, and then constructs simulation equations for stages, paths, and circuits. In order to check the error contributed by the SSTM only, we also implemented the SSTM model in Verilog-A and loaded it as a compiled model in Spectre [50]. The following symbols will be used in this section.

- 1) *RDEM*: The proposed RDE-based method (Section III) using the SSTM model (Section II).

TABLE I
ERRORS OF RDEM RELATIVE TO SPECTREB

average error	mean	standard deviation
delay errors	0.47%	0.28%
slew errors	0.20%	0.91%

- 2) *SpectreM*: Spectre simulation using the SSTM model.
- 3) *SpectreB*: Spectre simulation using the BSIM4 model.
- 4) *PWL-RDEM*: The proposed piece-wise linear RDE-based method (Section IV) using SSTM.

All relative errors reported in this paper are from comparison to the results from SpectreB.

A. SSTM-Based Deterministic Timing Analysis

Since the statistical simulation depends on the nominal value computation [x_s in (2)], the accuracy of the proposed SSTM-based gate models, used with our RDE-based method, for deterministic timing analysis (no process variations) is important. The accuracy of SSTM is first tested on the minimum-sized standard cells INV, NAND2, NOR2, AND2, XOR2, BUF, MUX2, AOI21, AOI211, and NAND4. Every switching input signal is a ramp with input slew varying from 7.5 ps to 600 ps and the load capacitance changes from 0.40 fF to 25.6 fF. The input slew and load capacitance ranges are the same as the ranges in the NLDM liberty file of the library. Both rising and falling inputs are simulated. In addition, the scenarios that all input signals switch at the same time (MISS) are also included in the experiments. For every gate, hundreds of simulations are performed for different input slew, output capacitance, and input switching scenarios, which result in hundreds of delay and slew errors. For accuracy comparison, the thousands of simulations of the cells listed above lead to delay and slew error statistics, as listed in Table. I. These results indicate a high accuracy of deterministic delay and slew calculation of these cells by using our SSTM-based gate model in the RDE-based engine.

The accuracy of the SSTM-based model and the deterministic simulation method are also tested on the critical paths of the ISCAS circuits. The results obtained by using RDEM and those obtained in SpectreM are both compared to the results of SpectreB and listed in Table II. The last row in Table II shows the average absolute value of the errors. The delay and slew errors of SpectreM are within 1% and 2% of SpectreB, which indicates the accuracy of our LUT-based simplified transistor model for timing analysis. The absolute value of delay and slew errors using SpectreM are slightly worse than the error magnitudes from RDEM on average. This may be caused by the combination of the LUT-based SSTM models and NR-based algorithms in Spectre, which suffer from non-smooth derivatives. Our use of the simplified chord algorithm, as explained in Step 2 of Section III-B, prevents such problems.

B. Statistical Delay Calculation Considering MISS

In order to evaluate the capability of our statistical simulation method for multiple variational inputs, we applied

TABLE II
DELAY AND OUTPUT SLEW ERRORS (%) FOR DETERMINISTIC TIMING ANALYSIS RELATIVE TO THE RESULTS OF SPECTREB

circuit	critical path		RDEM		SpectreM	
	gates	transistors	delay error	slew error	delay error	slew error
C432	27	128	0.93%	-1.63%	0.30%	1.38%
C499	20	92	-0.03%	0.05%	-0.95%	0.44%
C880	31	126	0.53%	-0.59%	-0.21%	0.57%
C1355	27	96	0.61%	0.17%	-0.12%	0.42%
C1908	39	144	-0.32%	-0.29%	-0.73%	0.20%
C2670	18	70	-0.06%	-0.52%	-0.53%	-0.29%
C3540	47	170	0.07%	0.39%	-0.59%	-1.44%
C5315	50	172	0.42%	0.32%	0.00%	0.01%
C6288	37	120	-0.09%	-0.19%	-0.99%	-0.08%
C7552	59	210	0.67%	0.13%	-0.61%	-1.78%
av. abs. error			0.37%	0.43%	0.50%	0.66%

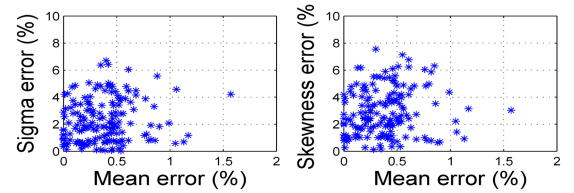


Fig. 6. All moment percentage errors comparison for MISS.

our RDEM approach on cells with up to four inputs. The multi-input cells are NAND2, NOR2, NOR3, NAND3, AOI21, AOI211, AOI22, and NAND4. All input signals of these gates are variational with variable correlation and have a high probability to switch near-simultaneously (MISS). The variational input signals are modeled as a ramp signal of 40-ps mean input transition time with voltage variations. The following two parameters are varied to obtain diverse scenarios to simulate for every cell: 1) the standard deviation of input voltages (σ_V) and 2) nominal arrival time differences between every two input signals (similar to dt in Fig. 3). The minimum and maximum σ_V are 1% and 10% of V_{dd} , respectively. The correlations among pairs of voltage variations range from 0 to 0.8. The statistical simulation results are compared to 10K SpectreB MC simulations. Fig. 6 shows that most μ errors are within 1%. Only AOI211 has errors larger than 1% when the correlation coefficient is 0.8 and σ_V is large. All the delay σ errors are lower than 6%, except two (6.42% and 6.71%) coming from NAND4 and AOI21, respectively, when all inputs have the same nominal arrival time. The third-order statistical central moment, skewness (γ), has maximum errors around 8%, which occurs when both σ_V and the correlation coefficient have the largest value. The average μ , σ , and γ errors are 0.38%, 2.30%, and 2.87%, respectively. For statistical delay calculation with MISS, these seem to be acceptable errors.

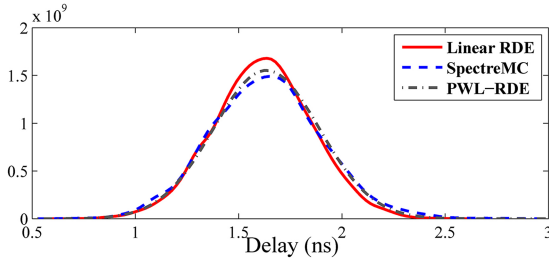
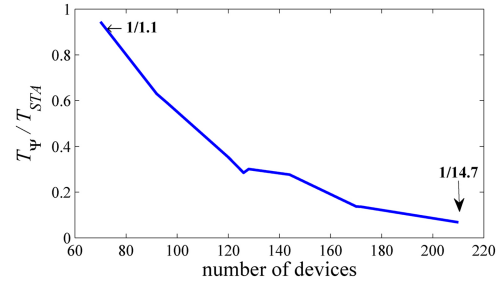
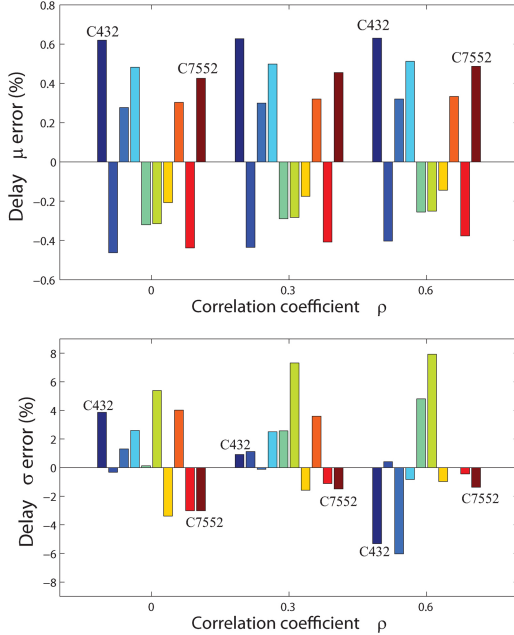
C. Statistical Delay Calculation for Sequential Circuits

The RDEM method is applied to the following three different sequential circuits with increasing level of complexity [51]: 1) an active-high transparent latch (DLH_X1) composed of 16 transistors; 2) a positive-edge triggered D flip-flop (DFF_X1) composed of 28 transistors;

TABLE IV

DELAY μ AND σ RESULTS OF ISCAS85 CIRCUITS WITH DIFFERENT PROCESS VARIATION VALUES, COMPARED WITH 10K SPECTREB MC

circuit	delay μ results								delay σ results							
	case I* low σ				case II high σ				case I low σ				case II high σ			
	μ_S°	μ_R	error	μ_S	μ_R	error	error _{pwl}	σ_S	σ_R	error	σ_S	σ_R	error	error _{pwl}	σ_S	σ_R
C432	1.33	1.34	0.62%	1.33	1.34	0.31%	1.23%	2.97	3.08	3.86%	17.87	18.00	0.72%	-1.48%	17.87	18.00
C499	0.92	0.92	-0.46%	0.93	0.92	-1.10%	0.39%	3.46	3.45	-0.32%	14.14	14.23	0.66%	-1.23%	14.14	14.23
C880	1.53	1.53	0.28%	1.53	1.53	-0.01%	0.87%	4.09	4.15	1.31%	23.66	22.45	-5.10%	-1.77%	23.66	22.45
C1355	1.14	1.15	0.48%	1.15	1.15	-0.27%	0.65%	3.14	3.22	2.60%	18.08	17.36	-3.99%	-4.31%	18.08	17.36
C1908	1.63	1.62	-0.32%	1.63	1.62	-0.88%	0.42%	6.61	6.61	0.13%	27.17	23.87	-12.15%	7.59%	27.17	23.87
C2670	1.39	1.39	-0.31%	1.40	1.39	-0.57%	0.36%	4.94	5.20	5.38%	22.67	21.32	-5.95%	-2.08%	22.67	21.32
C3540	2.13	2.12	-0.21%	2.14	2.12	-0.59%	0.43%	7.84	7.58	-3.39%	35.43	30.97	-12.60%	1.80%	35.43	30.97
C5315	1.70	1.71	0.30%	1.71	1.71	-0.35%	0.70%	4.67	4.86	4.02%	27.42	27.01	-1.51%	-4.95%	27.42	27.01
C6288	1.27	1.26	-0.44%	1.28	1.26	-0.94%	0.28%	4.65	4.51	-3.01%	20.66	18.23	-11.76%	1.42%	20.66	18.23
C7552	3.26	3.27	0.43%	3.27	3.27	0.21%	0.01%	11.76	11.40	-3.01%	54.18	47.78	-11.82%	0.86%	54.18	47.78
av. abs. error			0.39%			0.52%	0.53%			2.70%			6.63%	2.75%		

Fig. 9. Path delay $pdfs$ of C1908 obtained by using linear RDE and PWL-RDE solvers, respectively.Fig. 11. Run time comparison T_Ψ/T_{STA} .Fig. 10. Delay μ (upper) and σ errors (bottom) of ISCAS85 circuits. L_{eff} and V_{th} variable with three different correlations ρ . Different colors of the groups represent the circuits listed in Table IV from C432 to C7552.

The accuracy to estimate the delay moments considering correlation coefficient highly depends on the sensitivity characterization. According to the relationship between I_{ds} , L_{eff} , and V_{th} , the device I_{ds} decreases when L_{eff} or V_{th} increases. However, since $V_{th} \propto \sqrt{1 + \frac{LPE0}{L_{eff}}}$, the increasing L_{eff} causes decreasing V_{th} , which further results in rising I_{ds} . Therefore, these two process parameters are physically correlated

and the delay variations caused by L_{eff} and V_{th} are partly compensated. These physically correlated process variations lead to difficulties for device sensitivity characterization. The sensitivities of CSM model element to process variations are characterized based on best mean square error fit and derived from a series of SPICE MC simulations in [13]. In order to prevent an explosion of LUTs, Goel and Vrudhula [12] model the current and capacitance in gate models as second-order Hermite polynomials of process variations. These methods vary all the process variations of interest together for sensitivity characterization, which takes into account the physical correlation of process parameters. However, characterization takes thousands of times longer. In our method, simple finite differences are used for sensitivity approximation, which is much faster for characterization (only one or two extra DC analyses are required for each transistor) at the cost of some accuracy. Taking into account local correlations for sensitivity characterization would improve the accuracy of our statistical timing analysis method with high correlations.

E. Runtime

As mentioned in Section III-E, the total computation time of the proposed method is $T_{STA} = T_{STA} + T_\Psi$. T_Ψ , the computation time of the proposed RDEM algorithm, is approximately proportional to T_{STA} , as mentioned in Section III-E. Since the statistical simulation is performed with fewer matrix evaluations and without time step calculation nor NR iterations, the ratio T_Ψ/T_{STA} should be smaller for bigger circuits. Fig. 11 shows the runtime overhead of T_Ψ with respect to T_{STA} in the critical paths of the ISCAS85 circuits. The proposed statistical calculation shows lower runtime overhead for bigger circuits,

as expected. Compared to SpectreB MC runs, the proposed RDEM method, which is implemented in Matlab, achieves $200\times$ speedup on average. The speedup is smaller for larger circuits, showing the benefit of the sparse matrix techniques and efficient data loading techniques employed in Spectre. The same benefit can be obtained for our methods by using these techniques for higher efficiency.

VI. CONCLUSION

In this paper, we present a method to extend voltage-based gate models for statistical timing analysis. We construct gate models based on SSTMs for high accuracy. Correlations among input signals and between input signal and delay are fully preserved during simulation. Furthermore, the MISS problem is addressed by considering all input signals together. The variational waveform for statistical delay calculation is computed by a new RDE-based method. For high accuracy in the case of large process variations, we proposed a piecewise linear RDE-based statistical solver, which divides the process variation space into several subspaces. Since the proposed timing analysis is based on the transistor-level gate models, it is able to handle both combinational and sequential circuits. The experiments demonstrated the good combination of accuracy and efficiency of the proposed method for both deterministic and statistical timing analyses.

From the future work point of view, the proposed RDE-based method is generic for different transistor models. Hence, this method can be extended to evaluate the impact of small process variations on analog circuits. In this case, the proposed LUT-based device model with linear interpolation needs changes due to the small signal and continuousness requirements in analog circuit simulation.

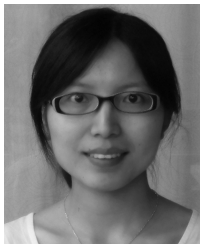
ACKNOWLEDGMENT

The authors would like to thank X. Zhen, Delft University of Technology, Delft, The Netherlands, who implemented the proposed SSTM into Spectre.

REFERENCES

- [1] A. Gattiker, S. Nassif, R. Dinakar, and C. Long, "Timing yield estimation from static timing analysis," in *Proc. ISQED*, 2001, pp. 437–442.
- [2] C. S. Amin, N. Menezes, K. Killpack, and F. Dartu, "Statistical static timing analog: How simple can we get?" in *Proc. DAC*, 2005, pp. 652–657.
- [3] A. Agarwal, D. Blaauw, V. Zolotov, S. Sundareswaran, Z. Min, K. Gala, et al., "Statistical delay computation considering spatial correlations," in *Proc. ASPDAC*, 2003, pp. 271–276.
- [4] R.-B. Lin and M.-C. Wu, "A new statistical approach to timing analysis of VLSI circuits," in *Proc. Int. Conf. VLSI Des.*, 1998, pp. 507–513.
- [5] H. Mangassarian and M. Anis, "On statistical timing analysis with inter- and intra-die variations," in *Proc. DATE*, vol. 1, 2005, pp. 132–137.
- [6] S. Tsukiyama and M. Fukui, "Accuracy of the criticality probability of a path in statistical timing analysis," in *Proc. ECCTD*, 2009, pp. 707–710.
- [7] Z. He, T. Lv, H. Li, and X. Li, "Graph partition based path selection for testing of small delay defects," in *Proc. ASPDAC*, 2010, pp. 499–504.
- [8] J. Chung, J. Xiong, V. Zolotov, and J. Abraham, "Path criticality computation in parameterized statistical timing analysis," in *Proc. ASPDAC*, 2011, pp. 249–254.
- [9] X. Lin and A. Davoodi, "Bound-based statistically-critical path extraction under process variations," *IEEE Trans. Computer-Aided Design Integr. Circuits Syst.*, vol. 30, no. 1, pp. 59–71, Jan. 2011.
- [10] S. Bhardwaj, S. Vrudhula, and A. Goel, "A unified approach for full chip statistical timing and leakage analysis of nanoscale circuits considering intradie process variations," *IEEE Trans. Computer-Aided Design Integr. Circuits Syst.*, vol. 27, no. 10, pp. 1812–1825, Oct. 2008.
- [11] J. F. Croix and D. F. Wong, "A fast and accurate technique to optimize characterization tables for logic synthesis," in *Proc. DAC*, 1997, pp. 337–340.
- [12] A. Goel and S. Vrudhula, "Statistical waveform and current source based standard cell models for accurate timing analysis," in *Proc. IEEE DAC*, 2008, pp. 227–230.
- [13] H. Fatemi, S. Nazarian, and M. Pedram, "Statistical logic cell delay analysis using a current-based model," in *Proc. IEEE DAC*, 2006, pp. 253–256.
- [14] B. Liu and A. B. Kahng, "Statistical gate level simulation via voltage controlled current source models," in *Proc. IEEE Int. Workshop Behavioral Modeling Simulation*, Sep. 2006, p. 23.
- [15] B. Liu, "Gate level statistical simulation based on parameterized models for process and signal variations," in *Proc. IEEE ISQED*, 2007, pp. 257–262.
- [16] J. F. Croix and D. F. Wong, "Blade and razor: Cell and interconnect delay analysis using current-based models," in *Proc. IEEE DAC*, 2003, pp. 386–389.
- [17] C. Amin, C. Kashyap, N. Menezes, K. Killpack, and E. Chiprout, "A multi-port current source model for multiple-input switching effects in CMOS library cells," in *Proc. IEEE DAC*, Jun. 2006, pp. 247–252.
- [18] C. Kashyap, C. Amin, N. Menezes, and E. Chiprout, "A nonlinear cell macromodel for digital applications," in *Proc. IEEE ICCAD*, 2007, pp. 678–685.
- [19] N. Menezes, C. Kashyap, and C. Amin, "A true electrical cell model for timing, noise, and power grid verification," in *Proc. IEEE DAC*, Jun. 2008, pp. 462–467.
- [20] B. Amelifard, S. Hatami, H. Fatemi, and M. Pedram, "A current source model for CMOS logic cells considering multiple input switching and stack effect," in *Proc. IEEE DATE*, 2008, pp. 568–574.
- [21] A. Devgan, "Accurate device modeling techniques for efficient timing simulation of integrated circuits," in *Proc. IEEE ICCD*, 1995, pp. 138–143.
- [22] F. Dartu, "Gate and transistor level waveform calculation for timing analysis," Ph.D. dissertation, Center for Electron. Design Automation, Carnegie Mellon Univ., Pittsburgh, PA, USA, 1997.
- [23] P. Kulshreshtha, R. Palermo, M. Mortazavi, C. Bamji, and H. Yalcin, "Transistor-level timing analysis using embedded simulation," in *Proc. IEEE ICCAD*, 2000, pp. 344–349.
- [24] P. F. Tehrani, S. W. Chyou, and U. Ekambaram, "Deep sub-micron static timing analysis in presence of crosstalk," in *Proc. IEEE ISQED*, 2000, pp. 505–512.
- [25] E. Acar, "Linear-centric simulation approach for timing analysis," Ph.D. dissertation, Dept. of ECE, Carnegie Mellon Univ., Pittsburgh, PA, USA, 2001.
- [26] E. Acar, F. Dartu, and L. Pileggi, "TETA: Transistor-level waveform evaluation for timing analysis," *IEEE Trans. Computer-Aided Design Integr. Circuits Syst.*, vol. 21, no. 5, pp. 605–616, May 2002.
- [27] L. McMurchie and C. Sechen, "WTA-waveform-based timing analysis for deep-micro circuits," in *Proc. IEEE ICCAD*, 2002, pp. 625–631.
- [28] Z. Wang and J. Zhu, "Transistor-level static timing analysis by piecewise quadratic waveform matching," in *Proc. IEEE DATE*, 2003, pp. 312–317.
- [29] S. Raja, F. Varadi, M. Becer, and J. Geada, "Transistor level gate modeling for accurate and fast timing, noise, and power analysis," in *Proc. IEEE DAC*, 2008, pp. 456–461.
- [30] Q. Tang, A. Zjajo, M. Berkelaar, and N. P. van der Meijs, "RDE-based transistor-level gate simulation for statistical static timing analysis," in *Proc. IEEE DAC*, 2010, pp. 787–792.
- [31] Q. Tang, A. Zjajo, M. Berkelaar, and N. P. van der Meijs, "Statistical delay calculation with multiple input simultaneous switching," in *Proc. IEEE ICICDT*, 2011, pp. 1–4.
- [32] Q. Tang, A. Zjajo, M. Berkelaar, and N. P. van der Meijs, "Transistor-level gate model based statistical timing analysis considering correlations," in *Proc. IEEE DATE*, 2012, pp. 917–922.
- [33] Q. Tang, A. Zjajo, M. Berkelaar, and N. P. van der Meijs, "Transistor level waveform evaluation for timing analysis," in *Proc. VARI*, 2010, pp. 1–6.
- [34] A. Hyvarinen and E. Oja, "Independent component analysis: Algorithms and applications," *Neural Netw. J.*, vol. 13, nos. 4–5, pp. 411–430, 2000.
- [35] R. Manduchi and J. Portilla, "Independent component analysis of textures," in *Proc. IEEE ICCV*, vol. 2, 1999, pp. 1054–1060.

- [36] Z. Feng, P. Li, and Y. Zhan, "Fast second-order statistical static timing analysis using parameter dimension reduction," in *Proc. IEEE DAC*, 2007, pp. 244–249.
- [37] Y. Bi, K. J. van der Kolk, J. F. Villena, L. M. Silveira, and N. P. van der Meijs, "Fast statistical analysis of RC nets subject to manufacturing variabilities," in *Proc. IEEE DATE*, 2011, pp. 32–37.
- [38] X. Li, P. Li, and L. T. Pileggi, "Parameterized interconnect order reduction with explicit-and-implicit multi-parameter moment matching for inter/intra-die variations," in *Proc. IEEE ICCAD*, 2005, pp. 806–812.
- [39] V. S. Nandakumar, D. Newmark, Y. Zhan, and M. Marek-Sadowska, "Statistical static timing analysis flow for transistor level macros in a microprocessor," in *Proc. 11th IEEE ISQED*, 2010, pp. 163–170.
- [40] V. Zolotov, J. Xiong, H. Fatemi, and C. Visweswariah, "Statistical path selection for at-speed test," *IEEE Trans. Computer-Aided Design Integr. Circuits Syst.*, vol. 29, no. 5, pp. 749–759, May 2010.
- [41] T. T. Soong, *Random Differential Equations in Science and Engineering*. New York, NY, USA: Academic, 1973.
- [42] C. Visweswariah and K. Ravindran, "First-order incremental block-based statistical timing analysis," *IEEE Trans. Computer-Aided Design Integr. Circuits Syst.*, vol. 25, no. 10, pp. 2170–2180, Oct. 2006.
- [43] S. Vrudhula, J. M. Wang, and P. Ghanta, "Hermite polynomial based interconnect analysis in the presence of process variations," *IEEE Trans. Computer-Aided Design Integr. Circuits Syst.*, vol. 25, no. 10, pp. 2001–2011, Oct. 2006.
- [44] A. Brambilla, A. Premoli, and G. Storti-Gajani, "Recasting modified nodal analysis to improve reliability in numerical circuit simulation," *IEEE Trans. Circuits Syst.*, vol. 52, no. 3, pp. 522–534, Mar. 2005.
- [45] F. N. Najm, *Circuit Simulation*. Hoboken, NJ, USA: Wiley, 2010.
- [46] Z. Peyton and J. R. Peebles, *Probability, Random Variables, and Random Signal Principles*. New York, NY, USA: McGraw-Hill, 1993.
- [47] G. Biagetti, P. Crippa, A. Curzi, S. Orcioni, and C. Turchetti, "Piecewise linear second moment statistical simulation of ICs affected by non-linear statistical effects," *Int. J. Circuit Theory Appl.*, vol. 38, no. 9, pp. 969–993, 2010.
- [48] Nangate 45nm Open Cell Library. (2009) [Online]. Available: <https://www.si2.org/openeda.si2.org/projects/nangatelib/>
- [49] X. Lu and W. P. Shi. (2004). *Layout and Parasitic Information for ISCAS Circuits* [Online]. Available: <http://dropzone.tamu.edu/xiang/iscas.html>
- [50] X. Zheng, "Implementing and evaluating a simplified transistor model for timing analysis of integrated circuits," Master's thesis, Dept. of EEMCS/ME, Delft Univ. Technol., Delft, The Netherlands, 2012.
- [51] J. Rodriguez, Q. Tang, A. Zjajo, M. Berkelaar, and N. van der Meijs, "Direct statistical simulation of timing properties in sequential circuits," in *Proc. PATMOS*, 2012, pp. 131–141.



Qin Tang (S'11) received the B.S. degree from the Department of Electronic Engineering, Southeast University, Nanjing, China, and the M.S. degree from the Department of Microelectronics and Solid State Electronics, School of Electronic Science and Engineering, Southeast University in 2006 and 2008, respectively, and the Ph.D. degree from the Delft University of Technology in 2013.

She is currently with the State Key Laboratory of Solid State Lighting, China. Her current research interests include design automation of integrated

circuits, specifically circuit simulation, transistor and gate modeling, timing analysis, and statistical performance analysis.



Javier Rodríguez received the Telecommunication Engineering degree from the Technical University of Madrid, Madrid, Spain, in 2006, and the M.Sc. degree in electrical engineering (microelectronics) from the Department of Microelectronics, Delft University of Technology, Delft, The Netherlands, in 2012, focusing on circuit design automation and verification techniques.

He is currently with Strukton Rolling Stock, Alblasterdam, The Netherlands, as a Software Engineer.



Amir Zjajo (M'02) received the M.Sc. and DIC degrees from Imperial College London, London, U.K., in 2000, and the Ph.D. degree from the Eindhoven University of Technology, Eindhoven, The Netherlands, in 2010, all in electrical engineering.

He joined Philips Research Laboratories, Eindhoven, in 2000, as a Research Staff Member with the Mixed-Signal Circuits and Systems Group. From 2006 to 2009, he was with Corporate Research, NXP Semiconductors, Eindhoven, as a Senior Research Scientist. In 2009, he joined the Department of

Microelectronics, Delft University of Technology, Delft, The Netherlands, as a Faculty Member with the Circuit and Systems Group. He has published more than 60 papers in referenced journals and conference proceedings. He holds more than ten U.S. patents or has patents pending. He has authored two books. His current research interests include mixed-signal circuit design, signal integrity and timing, and yield optimization of very large scale integration.

Dr. Zjajo serves as a member of the Technical Program Committee of the IEEE Design, Automation and Test in Europe Conference, the IEEE International Symposium on Circuits and Systems, and the IEEE International Mixed-Signal Circuits, Sensors and Systems Workshop.



Michel Berkelaar (M'96) received the Ph.D. degrees in electrical engineering from the Eindhoven University of Technology, Eindhoven, The Netherlands, in 1987 and 1992, respectively.

From 1992 to 2000, he was an Assistant and Associate Professor with the Eindhoven University of Technology. From 2000 to 2009, he was the Director of Research with Magma Design Automation, Eindhoven. Currently, he is a Researcher with the Department of Microelectronics, Delft University of Technology, Delft, The Netherlands. He has published more than 60 papers at conferences and in journals, mainly in the

areas of logic synthesis and statistical timing.

Dr. Berkelaar has served as a reviewer for many conferences and journals. He has served as the Topic Chair for ICCAD and DATE on multiple occasions. He has also served as the Program Chair for IWLS.



Nick van der Meijs (M'87) received the M.Sc. and Ph.D. degrees from the Delft University of Technology (TU Delft), Delft, The Netherlands, in 1985 and 1992, respectively.

Currently, he is an Associate Professor with the Circuits and Systems Group, Department of Micro Electronics and Computer Engineering, TU Delft. As the Director of Studies, he is responsible for the content, organization, and quality of the B.Sc. and M.Sc. curricula in electrical engineering and computer engineering at TU Delft. He has (co-)authored

some 100 papers on various topics, including design frameworks, interconnect optimization, and parasitics modeling, and was one of the lead developers of the SPACE 2-D and 3-D parasitic layouts to circuit extractors. He and his research group currently focus both on modeling of parasitic effects in advanced integrated circuits and on circuit level design methods and tools for dealing with variability.

Dr. van der Meijs is a Regular Reviewer for various EDA and design methodology conferences and journals. He has served as the Topic Chair on multiple at conferences.