A Framework for Statistical Timing Analysis using Non-Linear Delay and Slew Models

Sarvesh Bhardwaj, Praveen Ghanta, Sarma Vrudhula[†]
Department of Electrical Engineering, [†]Department of Computer Science and Engineering
Arizona State University, Tempe, AZ 85281.

ABSTRACT

In this paper[1] we propose a framework for Statistical Static Timing Analysis (SSTA) considering intra-die process variations. Given a cell library, we propose an accurate method to characterize the gate and interconnect delay as well as slew as a function of underlying parameter variations. Using these accurate delay models, we propose a method to perform SSTA based on a quadratic delay and slew model. The method is based on efficient dimensionality reduction technique used for accurate computation of the max of two delay expansions. Our results indicate less than 4% error in the variance of the delay models compared to SPICE Monte Carlo and less than 1% error in the variance of the circuit delay compared to Monte Carlo simulations.

1. INTRODUCTION

With the CMOS technology reaching 45nm node, the parametric yield loss due to large variations in delay and leakage has become the dominant contributing factor to the total yield loss. This has resulted in a significant amount of work in the area of statistical methods for parametric yield analysis and optimization of digital circuits in recent years. A number of techniques for statistical static timing analysis (SSTA) [5, 14, 2] and statistical leakage analysis [10] have been proposed. Previously proposed SSTA approaches can be classified into either path-based [8] or block-based [5, 12, 16, 14].

Initial block based SSTA methods [5, 12] were based on modeling the delay as a linear canonical function of Gaussian random variables and then approximating the max operation using Clarke's approximation. The intra-die correlations were captured by de-correlating the random variables modeling device parameters using Principal Component Analysis. However, the approximation of max of two delays as a linear function can result in significant errors. To solve this problem methods to propagate quadratic delay models were proposed [16, 14]. The authors in [16] use a *conditional linear max* operator to propagate the delay functions. [14] proposed an efficient moment matching based technique for propagating quadratic delay functions. The authors in [11] described a method to propagate linear delay models of non-Gaussian and Gaussian sources of variations. The max of two delays is also modeled as a linear function of non-Gaussian and Gaussian random variables.

To develop accurate models for gate and interconnect delays, we propose a novel technique based on *Polynomial Chaos*. In order to reduce the number of random variables in the analysis used for modeling the correlations due to intra-die variations, we use Karhunen-Loéve expansion (KLE) expansion. KLE provides a much more compact representation of the process variations and can provide the same degree of accuracy as the grid based approach with up to 4-5 times less number of random variables [4].

Once we have obtained accurate delay and slew models, we propose an approach to propagate delay and slew functions across the circuit graph to perform SSTA. Our method is based on an efficient dimensionality reduction technique for obtaining the coefficients of the max of two delays. The representation of the delay models using orthogonal polynomials allows us to independently compute the coefficients of the max of two delay expansions instead of solving a system of equations obtained using moment matching [14]. We also demonstrate how to account for non-Gaussian sources of variations in our proposed SSTA algorithm.

The outline of the rest of the paper is as follows: Section 2 provides the background material on *Polynomial Chaos*. Given a cell library, the library characterization for obtaining accurate delay and slew models is described in Section 3. Section 4 describes the process of de-correlating the random variables using KLE. Our method for performing SSTA is outlined in Section 5 and results are given in Section 7.

2. BACKGROUND

Let $\mathcal{H}$ be a complete metric space with an inner product $\langle\cdot,\cdot\rangle$ defined. The *norm* $\|f\|$ of any function f in $\mathcal{H}$ is defined as $\|f\| = \sqrt{\langle f, f\rangle}$. Using the results of polynomial chaos [6], a *second-order stochastic process* (a process whose second moment is finite) can be represented as

$$f = \sum_{i=0}^{\infty} \langle f, \psi_i\rangle \psi_i = \sum_{i=0}^{\infty} a_i \psi_i. \tag{1}$$

The equality in (1) is such that the series on the right converges to f in the norm. The functions, ψ_i's are *orthonormal basis functions*, that is

$$\langle \psi_i, \psi_j\rangle = E[\psi_i \cdot \psi_j] = \delta_{ij} = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

where $E[\cdot]$ is the expectation over the concerned probability space. In our problem, the basis functions, ψ_i's are functions of the random variables (RVs) modeling the underlying process variations. For example, if the process variations are modeled using *normal* RVs $\boldsymbol{\xi} = (\xi_1, \xi_2, \ldots, \xi_r)$, the resulting basis functions are known as Hermite polynomials. The first few uni-variate Hermite polynomials are given by

[1] This work was supported by NSF through ITR grant #CCR-0205227 and grant #CNS-0509540. An extended version of this paper is available at: http://www.public.asu.edu/~sbhardw1/papers/bhardwaj-iccad06.pdf

ICCAD'06, November 5-9, 2006, San Jose, CA
Copyright 2006 ACM 1-59593-389-1/06/0011 ...$5.00.

$$\begin{array}{ll} \psi_0(\xi) = 1 & \psi_1(\xi) = \xi \\ \psi_2(\xi) = \xi^2 - 1 & \psi_3(\xi) = \xi^3 - 3\xi \end{array} \qquad (3)$$

The multi-variate Hermite polynomials are simply products of lower order uni-variate Hermite polynomials. Previously, [13] used the PCE in Hermite polynomials to model the voltage response of an interconnect.

2.1 Obtaining the PCE for a System Response

The expansion in (1) is an infinite series and in practice has to be truncated to a finite number of terms, say n. As shown in (1) the i-th coefficient a_i in the PCE of a random function f is simply the inner product of f with the i-th basis function ψ_i. Thus representation of f in the form of (1) requires an accurate computation of this inner product. In the case where we have some implicit functional relation between the system response f and the excitation (as in the case of [13]), it is possible to obtain the coefficients of the truncated expansion $\hat{f}$ by minimizing the norm of the error between f and $\hat{f}$ [13] (also known as the Galerkin method).

But what happens when we do not have any functional relation describing f? In such a case, we have to resort to numerical techniques for computation of the inner products $\langle f, \psi_i \rangle$. The inner product is defined as

$$\langle f, \psi_i \rangle = E[f \cdot \psi_i] = \int f(\boldsymbol{\xi}) \cdot \psi_i(\boldsymbol{\xi}) \cdot w(\boldsymbol{\xi}) \partial \boldsymbol{\xi} \qquad (4)$$

where $w(\boldsymbol{\xi})$ is the weight function (probability density function) corresponding to the distribution of $\boldsymbol{\xi}$. For any general distribution of $\boldsymbol{\xi}$ and any arbitrary f, the integral in (4) can be estimated using a number of numerical techniques such as Monte Carlo or generalized quadrature methods. These techniques approximate the integral in (4) using a sum as

$$\int f(\boldsymbol{\xi}) \cdot \psi_i(\boldsymbol{\xi}) \cdot w(\boldsymbol{\xi}) \partial \boldsymbol{\xi} = \sum_{k=1}^{m} f(\boldsymbol{\xi}_k) \cdot \psi_i(\boldsymbol{\xi}_k) \cdot w_k \qquad (5)$$

where $\boldsymbol{\xi}_k = (\xi_{1k}, \xi_{2k}, \ldots, \xi_{rk})$ is a point in the r-dimensional parameter space and w_k is the weight of the k-th point. However, for these approaches the number of samples m at which the integrand needs to be evaluated for high accuracy can be prohibitively large. Instead, for some specific distributions such as Gaussian, Uniform etc., and smooth functions f, the integral can be evaluated with a very high accuracy using $N+1$ order Gaussian quadrature in each dimension, where N is the order of the polynomial that can accurately approximate f.

For our library characterization problem, we observed that the delay and slew can be modeled accurately using a second-order expansion. Thus we use a third-order Gaussian quadrature to estimate the inner products and thus the coefficients of the expansion. In practice, the number of parameters used for modeling gate and interconnect delay and slew is typically not more than 6-7 ($V_{tn}, V_{tp}, L, t_{ox}$ etc.) which results in 3^7 or ~2000 evaluations. Since, the characterization has to be done only once for a given library, this is a one time cost. Moreover we show in section 3 that a method known as Smolyak quadrature can be used to reduce the number of points to ~100.

3. DELAY AND SLEW MODELING

The delay and slew of the logic gates of a CMOS library vary with both the process variations (e.g., V_{tn}, V_{tp}, T_{ox}, L) that are typically modeled as Gaussian in nature and also deterministic parameters like load capacitance C_{eff} and power supply V_{dd}. The deterministic variations are typically handled through the use of look up tables. The process variations are random (with/with-out correlations) in nature and are generally handled as perturbations in a $\pm 3\sigma$ region around the nominal values of the parameters.

In this work, we develop a model for delay & output slew as a function of all these variables and input slew S_{in}. We first model the delay as a multi-variate function of all these variables, treating all these variables as deterministic quantities, through orthogonal polynomial interpolation (SPICE runs) on a Smolyak grid of quadrature nodes [3, 7]. We model the delay over the natural range of variations in C_{eff}, S_{in} that is typically encountered in analysis and the variables V_{tn}, V_{tp}, T_{ox} and L in their $\pm 4\sigma$ range. We then project this deterministic model on to a second order Hermite polynomial basis in the process variables and input slew. The coefficients of the second order Hermite polynomial expansion, which are now functions of the variable C_{eff}, can be readily obtained for various values of C_{eff}.

3.1 Smolyak Grid Interpolation

A widely used method in functional approximation is orthogonal polynomial based interpolation that ensures mean-square optimality in convergence w.r.t. the order of expansion. The response of interest is expressed as an N^{th} order series in orthogonal polynomial basis (e.g., Cheybshev, Legendre, Laguerre etc.,) and is then interpolated on the zeros of the $(N+1)^{th}$ order polynomials. These are called the quadrature nodes and are the exact same nodes that are also used in integration using the Gaussian quadrature method. However, for multi-variate interpolation or integration in r variables, the number of quadrature nodes increases as $\propto (N+1)^r$, which is well known as the curse-of-dimensionality.

To address this issue, we use an efficient approach based on interpolation (SPICE runs) on a Smolyak grid [3, 7]. This approach ensures some optimality in convergence while reducing the number of interpolation points for a given r as compared to full quadrature integration. Let's suppose that the variables under consideration are $\vec{Z} = (Z_1, ..., Z_r)$ deterministic variables that are normalized to the range $(-1, 1)$ and that the response of interest is a $(p)^{th}$ order polynomial and $(p+1)^{th}$ order interpolation is being used.

- The Smolyak grid is defined as
$$\Theta(r, p+1) = \bigcup_{p+1 \le |\boldsymbol{i}| \le p+r} \left(W_1^{i_1} \times W_2^{i_2} ... \times W_r^{i_r} \right) \qquad (6)$$
where $\boldsymbol{i} = (i_1 + ... + i_r)$ is the order of Smolyak interpolation and $W_j^{i_1}$ indicates a set of zeros of orthogonal polynomials of order i_1 in the dimension of variable j, $j = 1, .., r$. In this work, we use the zeros of the Chebyshev polynomials. *If $i_j = 1$, then W_j is taken as a set with only one point $\{0\}$. This significantly reduces the total number of interpolation points.*

- It has been shown that interpolation on a Smolyak grid assures an error bound for the mean-square error [3, 7] $\epsilon \le c_{r,k}\ n^{-k}\ (log\ n)^{(k+1)(r-1)}$ where n is the number of interpolation points and k is the order of the maximum derivative that exists for the function $d(\vec{Z})$. For functions like delay that are sufficiently smooth ($K > 1$), the rate of convergence is faster with increase in n (which follows from increase in the order i of interpolation).

- The number of Smolyak grid points increase as $\propto \frac{r^{p+1}}{(p+1)!}$. This cost increases much slowly as compared to full grid based interpolation which increases as r^{p+1}.

3.2 Delay & Slew as a deterministic function

Let's suppose that we would like to model the delay of a CMOS library gate in the ranges $[p_{-4\sigma}, p_{4\sigma}]$ for each parameter p, where p can be any of $\{L, T_{ox}, V_{tn}, V_{tp}\}$. Also, for C_{eff} and input slew S_{in}, the range is $[C_1, C_2]$ and $[S_{in_1}, S_{in_2}]$ respectively. We normalize them to range [-1,1] using

$$\begin{aligned} p &= (p_{-4\sigma} + p_{4\sigma})/2 + Z_i \cdot (p_{4\sigma} - p_{-4\sigma})/2 \\ C_{\mathit{eff}} &= (C_1 + C_2)/2 + Z_5\,(C_2 - C_1)/2 \\ S_{in} &= (S_{in_1} + S_{in_2})/2 + Z_6\,(S_{in_2} - S_{in_1})/2 \qquad (7) \end{aligned}$$

where $Z_i, i = 1, 2, 3, 4$ correspond to L, T_{ox}, V_{tn} and V_{tp} respectively. We then express the delay as a 2^{nd} order Chebyshev polynomial series in the variables $\vec{Z}$ and obtain the coefficients based on 3^{rd} interpolation on the Smolyak grid of Chebyshev zeros using 84 interpolation points (SPICE runs) for each CMOS gate in the library. We thus have,

$$\begin{aligned} d(\vec{Z}) = \sum_{i=0}^{N} \alpha_i \Psi_i(\vec{Z}) &= \alpha_0 + \sum_{i=1}^{6} \alpha_i Z_i + \sum_{i=1}^{6} \alpha_{6+i}(2Z_i^2 - 1) + \\ &+ \alpha_{13} Z_1 Z_2 + \cdots + \alpha_N Z_5 Z_6 \\ &= f(L, T_{ox}, V_{tn}, V_{tp}, C_{\mathit{eff}}, S_{in}) \qquad (8) \end{aligned}$$

3.3 Statistical model using Galerkin

In the presence of process variations, for a particular value of C_{eff}, we can express the process and slew variables as a function of normalized (zero mean, unit variance) random variables $\boldsymbol{\xi} = (\xi_1, \xi_2, \xi_3, \xi_4, \xi_5)$ as

$$\begin{aligned} V_{tn} &= \bar{V}_{tn} + \sigma_{V_{tn}}\xi_3, \quad V_{tp} = \bar{V}_{tp} + \sigma_{V_{tp}}\xi_4, \quad L = \bar{L} + \sigma_L \xi_1, \\ T_{ox} &= \bar{T}_{ox} + \sigma_{T_{ox}}\xi_2, \quad S_{in} = \bar{S}_{in} + \sigma_{S_{in}}\xi_5 \qquad (9) \end{aligned}$$

From (8) and (9), we have $d(\vec{Z}) = f(L, T_{ox}, V_{tn}, V_{tp}, C_{\mathit{eff}}, S_{in})$ $= g(\boldsymbol{\xi})$. The delay now needs to be projected on to a second order orthogonal polynomials in $\xi_1, \xi_2, \xi_3, \xi_4$ and a linear function in ξ_5. *The linear projection for slew variable ξ_5 is necessary to be able to propagate the delay expansions in the circuit.* For Gaussian variables, Hermite polynomials form the basis. We then have

$$\begin{aligned} d(\boldsymbol{\xi}) = \sum_{i=0}^{N} \beta_i H_i(\boldsymbol{\xi}) + \beta_{N+1}\xi_5 &= \beta_0 + \sum_{i=1}^{4} \beta_i \xi_i + \sum_{i=1}^{4} \beta_{4+i}(\xi_i^2 - 1) \\ &+ \beta_9 \xi_1 \xi_2 + \cdots + \beta_{N+1}\xi_5. \qquad (10) \end{aligned}$$

The coefficients β_i can be then obtained optimally using the Galerkin technique as $\beta_i = E(d(\boldsymbol{\xi}).H_i(\xi_p\, \xi_q))$.

The above procedure is repeated for each logic gate in the CMOS library. We compared our delay models for all the gates with those against 5000 Monte Carlo simulations and on average, our approach lead to an error of $< 1\%$ in mean and $< 4\%$ in variance. The major cost of the above procedure is the interpolation (SPICE runs), which is a one time cost. And mere substitutions are required from then onwards for delay uncertainty propagation through the circuit. *A model for output slew is obtained in a similar fashion as discussed above for the delay.*

Interconnect delay and slew are modeled as a second order Hermite polynomial series in the random variables. The coefficients of the series are obtained through sampling (similar to interpolation in gate delay modeling) on a variational interconnect model obtained from a Galerkin minimization procedure similar to that in [13].

4. PROCESS CHARACTERIZATION

Due to manufacturing variations, the parameter L and V_{th} of each gate on a die is a random variable. Also for a particular manufactured die, the gate length and threshold voltage are functions of location of the gate on the die. Thus these random parameters can be modeled as a stochastic process $p(\boldsymbol{x}, \theta)$, where $\boldsymbol{x} = (x, y)$ is the location on the die and θ belongs to the space of manufactured outcomes. Given two gates on a die located at $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$, the random variables $p(\boldsymbol{x}_1, \theta)$ and $p(\boldsymbol{x}_2, \theta)$ are correlated. Let the covariance function of $p(\boldsymbol{x}_2, \theta)$ be represented by $C(\boldsymbol{x}_1, \boldsymbol{x}_2)$. Hence, ideally for each parameter, there are as many random variables as the number of gates on a die.

In order to reduce the number of random variables in the analysis, the process $p(\boldsymbol{x}, \theta)$ can be represented using a Fourier-series type representation

$$p(\boldsymbol{x}, \theta) = \sum_{n=1}^{\infty} \sqrt{\lambda_n}\,\xi_n(\theta)\phi_n(\boldsymbol{x}), \qquad (11)$$

where $\{\xi_n(\theta)\}$ is a set of uncorrelated random variables, λ_n are the eigenvalues, and $\{\phi_n(\boldsymbol{x})\}$ are orthonormal eigenfunctions of $C(\boldsymbol{x}_1, \boldsymbol{x}_2)$. That is they are solution of the equation

$$\int_{\boldsymbol{D}} \boldsymbol{C}(\boldsymbol{x}_1, \boldsymbol{x}_2)\phi_n(\boldsymbol{x}_1) d\boldsymbol{x}_1 = \lambda_n \phi_n(\boldsymbol{x}_2). \qquad (12)$$

The expansion (11) is known as the Karhunen-Loéve expansion. In addition to treating the random variable corresponding to the parameter of each gate as a separate random variable, it can do so with a significantly small set of random variables $\{\xi_n(\theta)\}$ [4]. For example, consider the isotropic covariance function $C(\boldsymbol{r}_1, \boldsymbol{r}_2) = \exp(-c_r|r_1 - r_2|) = C(r_1, r_2)$, where c_r is the inverse of the *correlation length* in the radial direction. The general solution of (12) for this covariance function has the form [6]

$$\phi_n(r) = a_1 \cos(\omega_n r) + a_2 \sin(\omega_n r), \quad \lambda_{r,n} = \frac{2c}{\omega_n^2 + c^2}, \quad (13)$$

where ω_n is the solution of $c - \omega \tan(\omega a) = 0$ and $\omega + c\tan(\omega a) = 0$ for odd and even n respectively. Also, $a_2 = 0$ for odd n and $a_1 = 0$ for even n. The eigenvalues λ_n determine the contribution of the n-th random variable to the variance of $p(\boldsymbol{r}, \theta)$. Since we can always order the eigenvalues such that $\lambda_n > \lambda_{n+1}$, we truncate the expansion by finding the smallest M such that $\lambda_M (\sum_{n=1}^{M} \lambda_n)^{-1} \le \epsilon$, where ϵ is a threshold decided by the designer. In this work, we choose $\epsilon = 0.005$. Using this criteria, the KLE for $p(\boldsymbol{x}, \theta)$ having a radial covariance function was obtained and truncated to obtain $M = 9$.

The delay expansion of each gate i is obtained in terms of a common set of random variables by substituting the KLE corresponding to each random parameter of the gate i in its delay expansion d_i. Once delays of all the gates have been obtained, we perform SSTA to compute the circuit delay in terms of the common set of variables.

5. STATISTICAL TIMING ANALYSIS

Let d_i and T_i represent the delay and output arrival time respectively of gate i. Let $s_{i,in}$ and $s_{i,out}$ represent the input and output slew respectively for gate i. Also, let $\mathcal{L}(m)$ be the set of nodes at m-th level in the circuit. As discussed in section 3, for each gate we have two expansions corresponding to delay and output slew as a quadratic function of the random variables and a linear function of the input slew. The input slew at the primary inputs is either assumed to be available as quadratic function of the *same* random variables which model the parameters of the gates in the circuit or is modeled as a deterministic quantity. Thus the output slew as well as the output arrival times of all the gates in $\mathcal{L}(1)$ are absolutely determined by functions of the underlying process variables. That is

$$T_i = \sum_{k=1}^{n} \alpha_{ik}\psi_k(\boldsymbol{\xi}), \quad s_{i,out} = \sum_{k=1}^{n} \beta_{ik}\psi_k(\boldsymbol{\xi}) \qquad (14)$$

where $i \in \mathcal{L}(1)$. Let node $j \in \mathcal{L}(2)$ have fan-ins $i_1, i_2, \ldots i_m \in \mathcal{L}(1)$. Now, for each input i_k of node j the pin-to-pin delay $d_{i_k j}$ depends on the output slew $s_{i_k,out}$ of node i_k. The output arrival time T_j for node j can then be written as

$$T_j = \max\{T_{i_k} + d_{i_k j} : i_k \in \mathit{FI}(j)\} \qquad (15)$$

where $d_{i_k j} = \sum_{\ell=1}^{n} \gamma_{j\ell}\psi_\ell(\boldsymbol{\xi}) + a_j \cdot s_{i_k,out}$. Similarly, the output slew $s_{j,out}$ of node j is modeled as

$$s_{j,out} = \max\left\{ \sum_{\ell=1}^{n} \beta_{j\ell}\psi_\ell(\boldsymbol{\xi}) + s_{i_k,out} : i_k \in FI(j) \right\}. \quad (16)$$

Since $i_k \in \mathcal{L}(1)$, its output slew $s_{i_k,out}$ is available as a PCE from (14). Thus, for all the nodes in $\mathcal{L}(2)$, the slew and output arrival times are only functions of the underlying random variables. Using the same reasoning as above, the output arrival times of all the nodes the subsequent levels can be represented as functions of the process random variables.

The next two sections discuss the computation of these two operations for the case where the basis functions in the delay and slew expansions are from the set $\{1, \xi_i, \xi_j, \xi_i^2 - 1, \xi_j^2 - 1, \xi_i\xi_j : i, j = 1, \ldots, m\}$, where $m = rp$ is the number of random variables in the analysis.

Given two delay expansions $d_1 = \sum_{i=1}^{n} \alpha_i\psi_i(\boldsymbol{\xi})$ and $d_2 = \sum_{i=1}^{n} \beta_i\psi_i(\boldsymbol{\xi})$, their sum $d_1 + d_2$ can be obtained as

$$d_1 + d_2 = \sum_{i=1}^{n} \alpha_i\psi_i(\boldsymbol{\xi}) + \sum_{i=1}^{n} \beta_i\psi_i(\boldsymbol{\xi}) = \sum_{i}^{n} (\alpha_i + \beta_i)\psi_i(\boldsymbol{\xi}) \quad (17)$$

Compared to *sum*, computation of the *max* operation is not that straightforward. Since the set of random variables and thus the basis functions is the same for *all* delay expansions in our analysis, each delay expansion can be written in the following canonical form.

$$d(\boldsymbol{\alpha}) = \alpha_0 + \sum_{i=1}^{r} \alpha_{1i}\xi_i + \sum_{i=1}^{r} \alpha_{2i}(\xi_i^2 - 1) + \sum_{\substack{i,j=1 \\ i<j}}^{r} \alpha_{3ij}\xi_i\xi_j. \quad (18)$$

where $\boldsymbol{\alpha} = (\alpha_0, \alpha_{11}, \ldots, \alpha_{1r}, \alpha_{21}, \ldots, \alpha_{2r}, \alpha_{312}, \alpha_{313}, \ldots, \alpha_{3(r-1)r})$ and $\alpha_{3ij} = 0$ for $i, j = 1 \ldots, r,\ j < i$. Thus, each expansion is uniquely determined by the set of the coefficients of the basis functions. Let this mapping from the coefficients to the expansion be denoted by $d(\boldsymbol{\alpha})$ as defined in (18). Thus given two expansions $d(\boldsymbol{\alpha})$ and $d(\boldsymbol{\beta})$, we want to find an expansion for $d_{max} = \max\{d(\boldsymbol{\alpha}), d(\boldsymbol{\beta})\}$. Noting that $d_{max} = d(\boldsymbol{\alpha}) + \max\{0, d(\boldsymbol{\beta}) - d(\boldsymbol{\alpha})\}$, we need to find coefficients $\boldsymbol{\gamma}$ such that $d(\boldsymbol{\gamma}) = \max\{0, d(\boldsymbol{\beta}) - d(\boldsymbol{\alpha})\}$. Also, note that $d(\boldsymbol{\beta}) - d(\boldsymbol{\alpha}) = d(\boldsymbol{\beta} - \boldsymbol{\alpha})$, where $\boldsymbol{\delta} = \boldsymbol{\beta} - \boldsymbol{\alpha}$ is the component-wise difference of the tuples for $\boldsymbol{\beta}$ and $\boldsymbol{\alpha}$.

From (4), we see that the coefficients $\boldsymbol{\gamma}$ can be obtained by computing the following inner products

$$\begin{aligned}
\gamma_0 &= \langle d(\boldsymbol{\gamma}), 1\rangle = \langle \max\{0, d(\boldsymbol{\delta})\}, 1\rangle & (19)\\
\gamma_{1i} &= \langle d(\boldsymbol{\gamma}), \xi_i\rangle = \langle \max\{0, d(\boldsymbol{\delta})\}, \xi_i\rangle & (20)\\
\gamma_{2i} &= \langle d(\boldsymbol{\gamma}), \xi_i^2 - 1\rangle = \langle \max\{0, d(\boldsymbol{\delta})\}, \xi_i^2 - 1\rangle & (21)\\
\gamma_{3ij} &= \langle d(\boldsymbol{\gamma}), \xi_i\xi_j\rangle = \langle \max\{0, d(\boldsymbol{\delta})\}, \xi_i\xi_j\rangle. & (22)
\end{aligned}$$

and $\gamma_{3ii} = 0$ for $i, j = 1, \ldots, r$ and $j < i$. We now describe how to compute the inner product given by (20). The expansion $d(\boldsymbol{\delta})$ is first written by separating the terms dependent on ξ_i and those not dependent on ξ_i as follows

$$\begin{aligned}
d(\boldsymbol{\delta}) &= \delta_0 + \delta_{1i}\xi_i + \delta_{2i}(\xi_i^2 - 1) + \xi_i \cdot \sum_{j=1}^{r} \delta_{3ij}\xi_j + \\
&\quad + \sum_{\substack{j=1 \\ j\neq i}}^{r} \left(\delta_{1j}\xi_j + \delta_{2j}(\xi_j^2 - 1)\right) + \sum_{\substack{j,k=1 \\ j\neq i<k}}^{r} \delta_{3jk}\xi_j\xi_k \quad (23)
\end{aligned}$$

The above equation is in r-dimensional variability space. It is a second order equation and we only need 3 quadrature points in every dimension using the rule of Gaussian quadrature. Thus an exact quadrature based approach would require 3^r points, which can be computationally very expensive for large r. Thus we propose a moment matching based dimensionality reduction technique that reduces this complexity to 3^3. This method is based on mapping the last three terms in (23) to two functions of new Gaussian random variables ζ_1 and ζ_2 using moment matching by defining $X = \sum_{j=1}^{r} \delta_{3ij}\xi_j = a\,\zeta_1$ and

$$\begin{aligned}
Y &= \sum_{\substack{j=1 \\ j\neq i}}^{r} \left(\delta_{1j}\xi_j + \delta_{2j}(\xi_j^2 - 1)\right) + \sum_{\substack{j,k=1 \\ j\neq i<k}}^{r} \delta_{3jk}\xi_j\xi_k \\
&= b_1\zeta_2 + b_2(\zeta_2^2 - 1) \quad (24)
\end{aligned}$$

The mean of both X (Gaussian) and Y above is zero and $a = \sqrt{\sum_{j=1}^{r} \delta_{3ij}^2}$. In order to obtain b_1 and b_2, we minimize the absolute difference between the *skewness* of the quantity on the LHS and the RHS of (24) subject to their variance being equal. The variance σ^2 and skew κ of the quantity on the left of (24) are $\sigma^2 = \sum_{j=1, j\neq i}^{r} (2\delta_{2j}^3 + \delta_{1j}^2)$ and

$$\begin{aligned}
\kappa &= \sum_{j=1, j\neq i}^{r} \Big(8\delta_{2j}^3 + 6\delta_{1j}^2\delta_{2j} + 6\delta_{3jk}^2(\delta_{2j} + \delta_{2k}) + \\
&\quad + 6\delta_{3jk}(\delta_{1j} + \delta_{1k}) + 6 \cdot \sum_{\sum i_{jk} = 3} \prod_{\substack{j,k=1 \\ j\neq i<k}}^{r} \delta_{3jk}^{i_{jk}}\Big) \quad (25)
\end{aligned}$$

The last term in (25) above corresponds to the summation of the product of the coefficients δ_{3jk} of the cross-product terms in (24) taken three at a time ($\sum i_{jk} = 3$). The skew of the reduced form on the RHS of (24) is $\hat{\kappa} = 8b_2^3 + 6b_1^2 b_2$. Thus the skew minimization problem can be formulated as

$$\begin{aligned}
\min_{b_1, b_2} \quad & |8b_2^3 + 6b_1^2 b_2 - \kappa| \\
\text{subject to} \quad & 2b_2^2 + b_1^2 = \sigma^2 \qquad (26)\\
& b_1, b_2 \in \mathcal{R}
\end{aligned}$$

In [15], the authors give a method to solve the above problem when $|\kappa| \leq 2\sqrt{2}\sigma^3$. We extend their method to analytically solve the above minimization problem for any κ and σ. Replacing b_1^2 from the constraint of (26) in the objective function, we can rewrite the objective function as $|g(b_2)|$, where $g(b_2) = 4b_2^3 - 6\sigma^2 b_2 + \kappa$.

Theorem 5.1. *The optimal solution of the skew minimization problem (26) is*

$$b_2^\star = \begin{cases} \frac{\sigma}{\sqrt{2}} & \text{if } g(\frac{\sigma}{\sqrt{2}}) \geq 0 \\ \hat{b}_2 & \text{if } g(\frac{\sigma}{\sqrt{2}}) \leq 0 \text{ and } g(\frac{-\sigma}{\sqrt{2}}) \geq 0 \\ \frac{-\sigma}{\sqrt{2}} & \text{if } g(\frac{-\sigma}{\sqrt{2}}) \leq 0 \end{cases} \quad (27)$$

where $\hat{b}_2 \in [\frac{-\sigma}{\sqrt{2}}, \frac{\sigma}{\sqrt{2}}]$ is one of the real solutions of $g(b_2) = 0$.

Proof. Since we want real solutions for b_1 and b_2, $b_2 \in [\frac{-\sigma}{\sqrt{2}}, \frac{\sigma}{\sqrt{2}}]$ for b_1 to be real. Now consider the derivative of g, $g'(b_2) = 12b_2^2 - 6\sigma^2$. Over the domain of b_2, $g'(b_2) \leq 0$ and thus $g(b_2)$ is a decreasing function. Hence $g(b_2)$ can be either positive (Case I), have only one real root (Case II) or is negative in the range $[\frac{-\sigma}{\sqrt{2}}, \frac{\sigma}{\sqrt{2}}]$. For Case I and Case III, the optimal solution lies on the boundary of the domain of b_2. For Case III, $g(\frac{\sigma}{\sqrt{2}}) \leq 0$ and $g(\frac{-\sigma}{\sqrt{2}}) \geq 0$. These conditions translate to $|\kappa| \leq 2\sqrt{2}\sigma^3$ [15]. This condition is also equivalent to the condition for the cubic $g(b_2)$ to have all three real roots. Thus the roots for the cubic equation can be obtained and let the one that lies in $[\frac{-\sigma}{\sqrt{2}}, \frac{\sigma}{\sqrt{2}}]$ be $\hat{b}_2$. □

Now that we have obtained a, b_1 and b_2, we can rewrite the expansion $d(\boldsymbol{\delta})$ in (23) as

$$d(\boldsymbol{\delta}) = \delta_0 + \delta_{1i}\xi_i + \delta_{2i}(\xi_i^2 - 1) + a\,\xi_i\zeta_1 + b_1\zeta_2 + b_2(\zeta_2^2 - 1). \quad (28)$$

However, the inner product $\langle \max\{0, d(\boldsymbol{\delta})\}, \xi_i\rangle$ cannot be evaluated at this time because ζ_1 and ζ_2 are correlated. In order to de-correlate them, we first need to obtain their covariance $cov(\zeta_1,\zeta_2)$. To do this, we use the functional relation between $cov(\zeta_1,\zeta_2)$ and $cov(X,Y)$, which is $cov(X,Y) = a\,b_1 cov(\zeta_1,\zeta_2)$ [15]. Since X and Y are zero mean, their covariance is same as the expectation of their product. Also, as X and Y are functions of orthogonal polynomials, the expectation of their product can be written as $cov(X,Y) = E[X\cdot Y] = \sum_{j=1,j\neq i}^{r} \delta_{1j}\delta_{3ij}$. Hence

$$cov(\zeta_1,\zeta_2) = \left(\frac{1}{a\,b_1}\right)\cdot \sum_{j=1,j\neq i}^{r} \delta_{1j}\delta_{3ij}. \tag{29}$$

Two correlated Gaussian variables ζ_1 and ζ_2 with correlation ρ can be transformed into two new random variables χ_1 and χ_2 using a linear transformation [9]. Thus, we have the delay expansion $d(\boldsymbol{\delta})$ as function of only 3 independent Gaussian random variables ξ_1, χ_1 and χ_2. Hence, the inner product $\langle \max\{0, d(\boldsymbol{\delta})\}, \xi_i\rangle$ can be computed using Gauss-Hermite quadrature to obtain the coefficient γ_{1i}.

The coefficients for the other basis vectors can also be obtained in a similar manner. While computing the coefficients of the cross product terms $\xi_i\xi_j$, we need to keep the terms containing ξ_i and ξ_j and reduce the dimensionality of the remaining set of random variables. For the cross product case, the complexity for Gaussian quadrature will be 3^5 as there will be two existing variables and 3 additional variables (one each corresponding to the product terms with ξ_i and ξ_j and another one for the rest of the terms). In this case we use Smolyak quadrature to reduce the complexity to ~ 70 quadrature points. In practice, using our gate delay models and SSTA on benchmark circuits, we found that neglecting cross product terms does not result in significant error (less than 1% error in both mean and standard deviation for all benchmarks). Neglecting the cross terms results in the use of only 27 Gauss-Hermite quadrature nodes and results in extremely fast runtimes. Thus these terms can be neglected to trade-off accuracy for performance. Thus after computing $d(\boldsymbol{\gamma})$, we can obtain $d_{max} = d(\boldsymbol{\alpha}) + d(\boldsymbol{\gamma})$.

For a gate with multiple fan-ins, the output arrival time can be computed by successively performing the max of the current max and the next fan-in.

6. NON-GAUSSIAN VARIATIONS

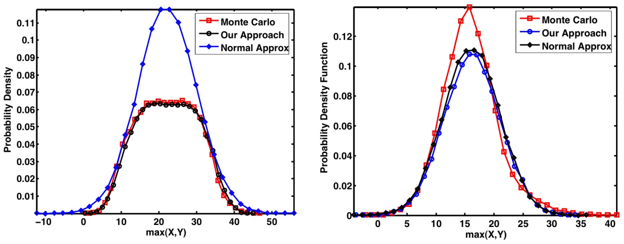
The method described in Section 5 can also be used to propagate linear delay expansions of non-Gaussian variables. Once we have independent non-Gaussian (say uniform) and Gaussian variables $\boldsymbol{\zeta}$ and $\boldsymbol{\xi}$ respectively, we can construct the orthonormal basis functions corresponding to the distributions of the random variables $\boldsymbol{\zeta}$ and $\boldsymbol{\xi}$. The inner product in this case is also defined by (4), with the weight function $w(\cdot)$ being the joint distribution of $\boldsymbol{\zeta}$ and $\boldsymbol{\xi}$. Since the random variables are independent, their joint probability distribution will be the product of their marginal distributions. For any distribution, the first order basis functions are of the form $\psi(x) = a(x+b)$. Thus the first order delay expansion can be written as

$$d(\boldsymbol{\alpha}) = \alpha_0 + \sum_{i=1}^{r_1} \alpha_{1i}\cdot a(\zeta_i + b) + \sum_{i=1}^{r_2} \alpha_{2i}\xi_i \tag{30}$$

We can then follow a procedure similar to the case of Gaussian variables to perform the max operation. While computing the inner product $\langle \max\{0, d(\boldsymbol{\delta})\}, a(\zeta_i+b)\rangle$, $d(\boldsymbol{\delta})$ in (23) can be written as

$$d(\boldsymbol{\delta}) = \delta_0 + \delta_{1i}\cdot a(\zeta_i+b) + \sum_{\substack{j=1,\\ j\neq i}}^{r_1} \delta_{1j}\cdot a(\zeta_j+b) + \sum_{\substack{j=1,\\ j\neq i}}^{r_2} \delta_{2j}\xi_j \tag{31}$$

Now the last two terms in (31) are sums of *independent* Gaussian and non-Gaussian random variables. Hence, using the Central Limit Theorem [9], they can be accurately modeled using a Gaussian random variable using moment matching. Thus the coefficients of the max can be computed efficiently in this case as well. If the number of non-Gaussian random variables is small 3, we do not need to approximate them using a Gaussian RV and can keep them in the expansion as they are during the computation of the inner product. As an extreme illustrative example, Figure 1 (a) shows the PDF of the max of two expansions having one Gaussian and one uniform random variable. As can be seen from the figure, our method provides an accurate estimation of the max even for non-Gaussian RVs. Even though the PDF shows significant deviation from the Gaussian, as the number of non-Gaussian variables increases, the non-Gaussian variables become less dominant. In such a case, the non-linearity of the max operation becomes a major source of error as compared to the error induced by neglecting the non-Gaussian component (shown in Figure 1 (b)). Hence, even in the presence of non-Gaussian sources of variations a linear model will not suffice. Thus further study is required as to what model (linear in non-Gaussian RVs and quadratic in Gaussian RVs or quadratic in both types of RVs) provides the best accuracy/performance trade-off.



(a) (b)

Figure 1: Max of two delay expansions with (a) one non-Gaussian and a Gaussian RV in each, and (b) two non-Gaussian and two Gaussian RVs in each

7. EXPERIMENTAL RESULTS

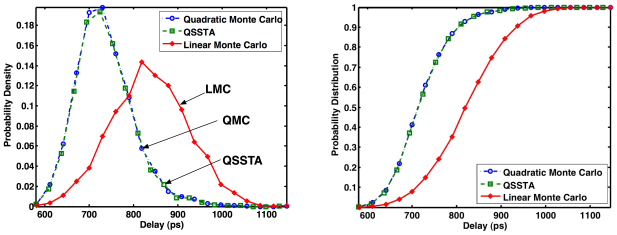
We implemented the proposed framework in C++. The ISCAS89 benchmark circuits were mapped to a cell library in SIS. They were then placed using UMpack [1]. The delay and slew models for the cells in the library were obtained for a 90-nm technology. The random variables considered in the analysis were gate length L, threshold voltage V_{th} and oxide thickness t_{ox}. Correlations for each of these variables were modeled using a radial exponential covariance function. The experiments were run on a 1.25GHz machine with 1.25GB RAM.

Table 1 gives the comparison of our approach with 5000 Monte Carlo (MC) simulations on the ISCAS89 benchmark circuits. QSSTA corresponds to our methodology based on a quadratic delay model. During our library characterization, we found that the cross terms were not significant even for 30% 3σ variations. Hence, we ignore the cross terms while performing SSTA. However, to make a fair comparison, the cross terms are kept while performing the MC simulations. Thus MC computes the delay based on the *exact* delay function consisting of the cross terms. Since we are using non-linear delay models, one measure of the non-linearity (or deviation from Gaussianity) of a random variable is its skew. Hence in addition to the mean and variance, we also compare the skew of the circuit delay with that of the MC samples. For each sample in the space of random variables, we evaluate the delay expansion for the arrival time at the sink node of the circuit to obtain the delay given for that sample by our SSTA. For MC, we obtain

Circuit	# Gates	μ_d Comparison			σ_d Comparison			Skew Comparison			Runtime (sec.)	
		QSSTA	MC	Diff. (%)	QSSTA	MC	Diff. (%)	QSSTA	MC	Diff. (%)	QSSTA	MC
C499	290	568	568	0.000	47.1	47.57	0.988	0.79	0.82	3.66	6	203
C880	227	564.6	565.1	0.088	47.8	47.6	-0.420	0.86	0.88	2.27	11	282
C1355	514	748.6	749.2	0.080	63.4	64.2	1.246	0.82	0.86	4.65	20	521
C2670	427	514.8	515.2	0.078	42.2	42.3	0.236	0.69	0.72	4.17	19	389
C3540	743	971.1	971.8	0.072	79.2	79.19	-0.013	0.76	0.8	5.00	20	402
C5315	946	898.8	899.8	0.111	74.7	75.1	0.533	0.79	0.83	4.82	21	2165
C6288	1688	2555	2556	0.039	209.7	210.2	0.238	0.72	0.75	4.00	55	2287
C7552	1262	741.6	742.1	0.067	61.61	61.67	0.097	0.79	0.82	3.66	41	1262
Average				0.067			0.363			4.028		

Table 1: Results of SSTA for ISCAS89 benchmark circuits with a total of 27 random variables

the delay for each node in the gate corresponding to that sample and perform a run of Dijkstra's algorithm to obtain the true circuit delay. From the table, we can see that the average error in the mean and standard deviation obtained using our approach is less than 1% compared to 5000 MC samples. In addition, the average error between the skew obtained using our approach and that obtained using MC is ∼ 5%. Compared to our approach an approach based on linear models resulted in significant errors in the mean and variance with average errors being ∼ 2% and ∼ 8% respectively. It should be noted that while performing linear MC, we do not simply ignore the higher order terms. Instead the variance corresponding to the higher order terms is added to the variance of the first order coefficients for each random variable. Figure 2(a) and (b) compare the sample PDFs and CDFs respectively for our approach with the MC using a *exact* quadratic model and with the MC performed assuming a linear model. As shown in the figure, our approach matches extremely well with the exact MC approach. The linear model based MC shows significant deviation from the exact MC, which shows the importance of using second order delay models. Again from the PDF and CDF for C7552, we see that the cross-terms between the random variable do not contribute much to the circuit delay. This is similar to trend observed by [14].



(a) (b)

Figure 2: Comparison of the (a) PDF and (b) CDF obtained using our approach with exact Monte Carlo and linear Monte Carlo

The last two columns in Table 1 show the runtime in seconds for our approach as well as the Monte Carlo approach. All the benchmarks using our approach were solved in less than 60 seconds. Whereas, MC completed in ∼ 2400 sec. for some of the cases. Thus compared to MC approach provides ∼ 40× improvement in the runtime.

8. CONCLUSIONS

We propose an accurate approach for modeling the delay using orthogonal polynomials. The intra-die correlations are captured using Karhunen-Loéve Expansion which can reduce the dimensionality of the variability space by ∼ 4× for similar accuracy. We also propose a novel algorithm to propagate our second order delay expansions through the circuit to perform SSTA. Our method can take non-Gaussian sources of variations into account. The experimental results show less than 1% error in the mean and variance of the circuit delay compared to MC simulations.

9. REFERENCES

[1] http://vlsicad.eecs.umich.edu/bk/pdtools/.

[2] A. Agarwal, D. Blaauw, and V. Zolotov. Statistical timing analysis for intra-die process variations with spatial correlations. In *Proc. of ICCAD*, 2003.

[3] V. Barthelmann, E. Novak, and K. Ritter. High dimensional polynomial interpolation on sparse grids. *Advances in Comp. Math.*, pages 273–288, 2000.

[4] S. Bhardwaj, S. Vrudhula, P. Ghanta, and Y. Cao. Modeling of intra-die process variations for accurate analysis and optimization of nano-scale circuits. In *Proc. of IEEE/ACM Design Automation Conference*, 2006.

[5] H. Chang and S. Sapatnekar. Statistical timing analysis considering spatial correlations using a single pert-like traversal. In *Proc. of ICCAD*, 2003.

[6] R. G. Ghanem and P. Spanos. *Stochastic Finite Elements: A Spectral Approach*. Springer-Verlag, 1991.

[7] A. Keese and H. G. Matthies. Numerical methods and smolyak quadrature for nonlinear stochastic partial differential equations. Technical report, Institute of Scientific Computing, Brunswick, 2003.

[8] M. Orshansky and K. Kuetzer. A General Probabilistic Framework for Worst Case Timing Analysis. In *Proc. of DAC*, 2002.

[9] A. Papoulis. *Probability, Random Variables and Stochastic Processes*. McGraw-Hill, 3rd edition, 1991.

[10] R. Rao, A. Devgan, D. Blaauw, and D. Sylvester. Parametric yield estimation considering leakage variability. In *Proc. of DAC*, 2004.

[11] J. Singh and S. Sapatnekar. Statistical Timing Analysis with Correlated Non-Gaussian Parameters using Independent Component Analysis. In *IEEE TAU Workshop*, February 2006.

[12] C. Visweswariah et al. First-order incremental Block-Based Statistical Timing Analysis. In *IEEE/ACM Design Automation Conference*, pages 331–336, 2004.

[13] J. Wang, P. Ghanta, and S. Vrudhula. Stochastic Analysis of Interconnect Performance in the Presence of Process Variations. In *Proc. of ICCAD*, 2004.

[14] Y. Zhan et al. Correlation-aware statistical timing analysis with non-gaussian delay distributions. In *Proc. of ICCAD*, Nov 2005.

[15] L. Zhang, J. Shao, and C. C.-P. Chen. Non-Gaussian Statistical Parameter Modeling for SSTA with Confidence Interval Analysis. In *International Symposium on Physical Design*, 2006.

[16] L. Zhang et al. Correlation-Preserved Non-Gaussian Statistical Timing Analysis with Quadratic Timing Model. In *Proc. of DAC*, 2005.