Statistical Timing Analysis Under Spatial Correlations

Hongliang Chang and Sachin S. Sapatnekar, *Fellow, IEEE*

***Abstract*— Process variations are of increasing concern in today's technologies, and can significantly affect circuit performance. We present an efficient statistical timing analysis algorithm that predicts the probability distribution of the circuit delay considering both inter-die and intra-die variations, while accounting for the effects of spatial correlations of intra-die parameter variations. The procedure uses a first-order Taylor series expansion to approximate the gate and interconnect delays. Next, principal component analysis techniques are employed to transform the set of correlated parameters into an uncorrelated set. The statistical timing computation is then easily performed with a PERT-like circuit graph traversal. The run-time of our algorithm is linear in the number of gates and interconnects, as well as the number of varying parameters and grid partitions that are used to model spatial correlations. The accuracy of the method is verified with Monte Carlo simulation. On average, for 100nm technology, the errors of mean and standard deviation values computed by the proposed method are 1.06% and -4.34% respectively, and the errors of predicting the 99% and 1% confidence point are -2.46% and -0.99% respectively . A testcase with about 17,800 gates was solved in about 500 seconds, with high accuracy as compared to a Monte Carlo simulation that required more than 15 hours.**

***Index Terms*— Circuit, Deep_submicron, Timing_analysis, VLSI**

I. Introduction

PROCESS variations have become an increasing concern in integrated circuits as circuit sizes continue to increase and feature sizes continue to shrink. As device and interconnect parameters such as physical dimensions show variability, the prediction of circuit performance is becoming a challenging task. Conventional static timing analysis (STA) handles the problem of variability by analyzing a circuit at multiple process corners. However, it is generally accepted that such an approach is inadequate, since the complexity of the variations in the performance space implies that if a small number of process corners is to be chosen, these corners must be very conservative and pessimistic. For true accuracy, this can be overcome by using a larger number of process corners, but then the number of corners that must be considered for an accurate modeling will be too large for computational efficiency.

The limitations of traditional STA techniques lie in their deterministic nature. An alternative approach that overcomes these problems is statistical STA, which treats delays not as fixed numbers, but as probability density functions (PDF's), taking the statistical distribution of parametric variations into consideration while analyzing the circuit.

Process variations can be classified into the following categories: *inter-die variations* are the variations from die to die, while *intra-die variations* correspond to variability within a single chip. Inter-die variations affect all the devices on same chip in the same way, e.g., making the transistor gate lengths of devices on the same chip all larger or all smaller, while the intra-die variations may affect different devices differently on the same chip, e.g., making some devices have smaller transistor gate lengths and others larger transistor gate lengths.

It used to be the case that the inter-die variations dominated intra-die variations, so that the latter could be safely neglected. However, in modern technologies, intra-die variations are rapidly and steadily growing and can significantly affect the variability of performance parameters on a chip [1]. The increase in intra-chip parameter variations is due to the effects such as micro-loading in the etch, variation in photoresist thickness, optical proximity effects and stepper within-field aberrations as the manufacturing sizes approach the optical resolution limit [2]. Intra-die variation is spatially correlated: it is locally layout-dependent and circuit-specific, i.e., devices with similar layout patterns and proximity structures tend to have similar characteristics; it is globally location-dependent, i.e., devices located close to each other are more likely to have the similar characteristics than those placed far away.

Due to the increasing effect of intra-die variations, several commercial flows have begun to include intra-die variations in the last few years, e.g., the OCV (On-Chip Variation) analysis in Synopsys's PrimeTime and the LCD (Linear Combination of Delay) mode of IBM's EinsTimer. In literature, a number of studies on statistical timing analysis have focused on circuit performance prediction considering intra-die variation. Continuous methods [3]–[6] use analytical approaches to find closed-form expressions for the PDF of the circuit delay. For simplicity, these methods often assume a normal distribution for the gate delay, but even so, finding the closed-from expression of the circuit distribution is still not an easy task. Discrete methods [7]–[9] are not limited to normal distributions, and can discretize any arbitrary delay distribution as a set of tuples, each corresponding to a discrete delay and its probability. The discrete probabilities are propagated through the circuit to find a discrete PDF for the circuit delay. However, this method is liable to suffer from the problem of having to propagate an exponential number of discrete point probabilities. In [10], an efficient method was proposed by modeling arrival times as cumulative density functions and delays as probability density functions and by defining operations of *add* and *max* on these

This work was supported in part by the NSF under award CCR-0205227 and by the SRC under contract 2003-TJ-1092.

Hongliang Chang is with Department of Computer Science and Engineering and Sachin S. Sapatnekar is with Department of Electrical and Computer Engineering, both in the University of Minnesota.

functions. Alternatively, instead of finding the distribution of circuit delay directly, several attempts have been made to find upper and lower bounds for the circuit delay distribution [5], [7], [11].

Although many prior works have dealt with intra-chip variations, most of them have ignored intra-chip spatial correlations by simply assuming zero correlations among devices on the chip. The difficulty in considering spatial correlations between parameters is that it can result in complicated path correlation structures that are hard to deal with. The authors of [6] consider correlation between delays among the transistors inside a single gate (but not correlations between gates). The work in [12] uses a Monte Carlo sampling-based framework to analyze circuit timing on a set of selected sensitizable true paths. Another method in [5] computes path correlations on the basis of pair-wise gate delay covariances and used an analytic method to derive lower and upper bounds of circuit delay. The statistical timing analyzer in [13] takes into account capacitive coupling and intra-die process variation to estimate the worst case delay of critical path. Two parameter space techniques, namely, the parallelepiped method and the ellipsoid method, and a performance-space procedure, the binding probability method, were proposed in [14] to find either bounds or the exact distribution of the minimum slack of a selected set of paths. The approach in [3] proposes a model for spatial correlation and a method of statistical timing analysis to compute the delay distribution of a specific critical path. However, the PDF for a critical path may not be a good predictor of the distribution of the circuit delay (which is the maximum of all path delays), as explained in Section II. Moreover, the method may be computationally expensive when the number of critical paths is too large. In [15], the authors further extended their work in [3], [7] to compute an upper bound on the distribution of exact circuit delay.

In this paper, we will propose an algorithm for statistical STA that computes the distribution of circuit delay while considering spatial correlations. We will model the circuit delay as a correlated multivariate normal distribution, considering both gate and wire delay variations. In order to manipulate the complicated correlation structure, the Principal Component Analysis (PCA) technique is employed to transform the sets of correlated parameters into sets of uncorrelated ones. The statistical timing computation is then performed with a PERT-like circuit graph traversal. The complexity of the algorithm is $O(p \times n \times (N_g + N_I))$, which is linear in the number of gates N_g and interconnects N_I, and also linear in the number of varying parameters p and the number of grid squares n that are used to model variational regions. In other words, the cost is, at worst, $p \times n$ times the cost of a deterministic STA. We believe that this is the first method that can fully handle spatially correlated distributions under reasonably general assumptions, with a complexity that is comparable to traditional deterministic STA. This work can also be extended, using the same framework of maximum of delays (Section IV-C), to find the distribution of minimum of delays which can be applied to analysis such as computing minimum delay distributions for short-path analysis (to check for hold time violations), for required arrival time (RAT) analysis, etc.

The remainder of the paper is organized as follows. Section II formally formulates the problem to be solved in this work. Section III explains the model used for process variation and spatial correlation of intra-die variation. The algorithm is presented in Section IV and its run time complexity analysis is given in the following section. The extension to handle inter-chip variation and spatially uncorrelated intra-chip components is introduced in Section VI. The extension to compute minimum of delays is also presented in Section VI. Finally, a list of experimental results and their analysis are shown in Section VII.

II. Problem formulation

Under process variations, parameter values such as the gate length, the gate width, the metal line width and the metal line height are random variables. Some of these variations such as across-chip linewidth variations (ACLV) are deterministic, while others are random: this work will focus on the effects of random variations, and will model these parameters as random variables. The gate and interconnect delays, as functions of these parameters, also become random variables. Given appropriate modeling of process parameters or gate and interconnect delays, the task of statistical STA is to find the PDF of the circuit delay.

The static timing analysis works with the usual translation from a combinational circuit to a timing graph [16]. The nodes in this graph correspond to the circuit primary inputs/outputs and gate input/output pins. The edges are of two types: one set corresponds to the pin-to-pin delay arcs within a gate, and the other set to interconnections from the drivers to receivers. The edges are weighted by the pin-to-pin gate delay, and interconnect delay, respectively. The primary inputs of the combinational circuit are connected to a virtual source node, and the primary outputs to a virtual sink node with directed virtual edges. In the case that primary inputs arrive at different times, the virtual edges from the virtual source to the primary inputs are assigned weights of the arrival times. Likewise, if the required times at the primary outputs are different, the weights of the edges from the outputs to the virtual sink are appropriately chosen.

For a combinational logic circuit, the problem of static timing analysis is to compute the longest path delay in the circuit from any primary input to any primary output, which corresponds to length of the longest path in the timing graph. In static timing analysis, the technique that is commonly referred to in the literature as PERT (Program Evaluation and Review Technique) is commonly used[1]. This procedure starts from the source node to traverse the graph in a topological order and uses a *sum* operation or *max* operation (at a multi-fanin node) to find the longest path at the sink node. For details, the reader may refer to [16], [17].

Since we will employ a PERT-like traversal to analyze the distribution of circuit delay, we define a statistical timing graph of a circuit, as in the case of deterministic STA.

[1]In reality, this is actually the critical path method (CPM) in operations research. However, we will persist with the term "PERT," which is widely used in the static timing analysis literature.

Definition 2.1: Let $G_s = (V, E)$ be a timing graph for a circuit with a single source node and a single sink node, where V is a set of nodes and E a set of directed edges. The graph G_s is called a statistical timing graph if each edge i is assigned a weight d_i, where d_i is a random variable, where the random variables may be uncorrelated or correlated. The weight associated with an edge corresponds to gate delay or interconnect delay. For a virtual edge, the weight is random variables with mean of its deterministic value and standard deviation of zero and it is independent from any other edges.

Definition 2.2: Let a path p_i be a set of ordered edges from the source node to the sink node in G_s, and D_i be the path length distribution of p_i, computed as the sum of the weights d_k for all edges k on the path. Finding the distribution of $D_{max} = max(D_1, \ldots, D_i, \ldots, D_{n_{paths}})$ among all paths (indexed from 1 to n_{paths}) in the graph G_s is referred to as the problem of statistical static timing analysis (SSTA) of a circuit.

Note that for the same nominal design, the identity of the longest path may change, depending on the random values taken by the process parameters. Therefore, finding the delay distribution of one critical path at a time is not enough, and correlations between paths must be considered in finding the max of the PDF's of all paths. Such an analysis is essential for finding the probability of failure of a circuit, which is available from the cumulative density function (CDF) of the circuit delay.

For an edge-triggered sequential circuit, the statistical timing graph can be constructed similarly by breaking the circuit into a set of combinational blocks between latches, and the analysis includes statistical checks on setup and hold time violations. The former requires the computation of the distribution of the maximum arrival time at the latches, which requires the solution of the SSTA problem as defined above. On the other hand, the latter problem needs the distribution of the minimum arrival time at the latches to be computed, and this can be solved by a trivial extension of the framework for the SSTA problem proposed in the paper, using minimum operators, as will be mentioned in Section VI-C, instead of maximum operators.

Our approach to solve the SSTA problem is based on the following assumption on the distribution of the process parameter values:

Assumption 1: The process parameter values are assumed to be normally distributed random variables.

The gate and interconnect delays, being functions of the fundamental process parameters, are approximated using a first-order Taylor series expansion. We will show that as a result of this, all edges in graph G_s are normally distributed random variables. Since we consider spatial correlations of the process parameters, it turns out that some of the delays are correlated random variables. Furthermore, the circuit delay D_{max} is modeled as a multivariate normal distribution. Although the closed form of circuit delay distribution is not normal, we show that the loss of accuracy is not significant under this approximation.

III. Modeling parameter variations

In this section, we will introduce the model used for intra-chip variations with spatial correlation. Although we consider only intra-die variations of parameters at this point, the extension of this work to handle inter-die variations will be introduced later in Section VI-A.

A. Components of Intra-Chip Variations

The intra-chip parametric variation δ_{intra} can be decomposed into three components, a deterministic global component δ_{global}, a deterministic local component δ_{local} and a random component ϵ [18]

$$\delta_{intra} \quad = \quad \delta_{global} + \delta_{local} + \epsilon. \tag{1}$$

The global component, δ_{global}, is location-dependent. Across the die or reticle field, it can be modeled by a slanted plane and expressed as a simple function of its location

$$\delta_{global}(x, y) \quad = \quad \delta_0 + \delta_x x + \delta_y y, \tag{2}$$

where (x, y) is its die location, δ_x and δ_y are gradients of parameter indicating the spatial variations of parameter along the x and y directions respectively.

The local component, δ_{local}, is proximity-dependent and layout-specific. The random component, ϵ, stands for the random intra-chip variation and the vector of all random components across the chip or reticle field has a correlated multivariate normal distribution due to spatial correlation of the intra-chip variation

$$\vec{\epsilon} \sim N(0, \Sigma), \tag{3}$$

where Σ is the covariance matrix of parameters. The detailed model for this covariance matrix will be described in the next section. For spatially uncorrelated parameters, Σ becomes a diagonal matrix where the entries represent variances. If the variances of the parameters described by this matrix are assumed to be uniform across the chip, then Σ is a multiple of the identity matrix.

In this paper, we will only consider the impact of global and random components. However, the local component can also be included in the model, given, for instance, the chip layout and pre-characterized spatial maps of parameters as in [19].

Under intra-die variation, the value of parameter p located at (x, y) can be modeled as

$$p \quad = \quad \bar{p} + \delta_x x + \delta_y y + N(0, \sigma), \tag{4}$$

where $\bar{p}$ is the nominal design parameter value at die location $(0, 0)$.

In this way, all parameter variations are modeled as location-dependent normally distributed random variables.

In this work, for transistors, we consider the following process parameters [20] as random variables: transistor length L_g and width W_g, gate oxide thickness T_{ox}, doping concentration density N_a; for interconnect, at each metal layer, we consider the following parameters: metal width W_{int_l}, metal thickness T_{int_l} and ILD thickness H_{ILD_l}, where the subscript l represents that the random variable is of layer l, where $l = 1 \ldots n_{layers}$. Among all the parameters listed above, L_g

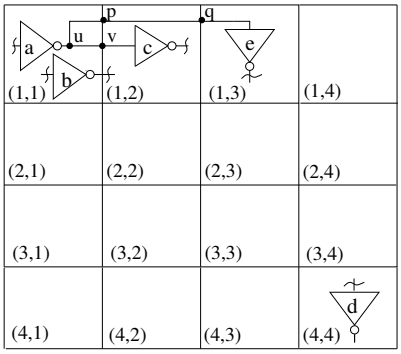


Fig. 1. Grid model for spatial correlations

is observed to exhibit largest parameter variability and also has the most important impact on circuit performance when it shows variations [20]. We believe that this framework is general enough that it can be applied to handle variations of other parameters as well.

B. Spatial Correlations

To model the intra-die spatial correlations of parameters, we partition the region of die or reticle field[2] into $nrow \times ncol = n$ grids. Since devices [wires] close to each other are more likely to have more similar characteristics than those placed far away, we assume perfect correlations among the devices [wires] in the same grid, high correlations among those in close grids and low or zero correlations in far-away grids. For example, in Figure 1, gates a and b (whose sizes are shown to be exaggeratedly large) are located in the same grid square, and it is assumed that their parameter variations (such as the variations of their gate length), are always identical. Gate a and c lie in neighboring grids, and their parameter variations are not identical but highly correlated due to their spatial proximity (for example, when gate a has a larger than nominal gate length, it is highly probable that gate c will have a larger than nominal gate length, and less probable that it will have a smaller than nominal gate length). On the other hand, gates a and d are far away from each other, their parameters may be uncorrelated, (e.g., when gate a has a larger than nominal gate length, the gate length for d may be either larger or smaller than nominal).

Our algorithm makes a second assumption on the distribution of process parameters:

Assumption 2: It is assumed that nonzero correlations may exist only among the same type of process parameters in different grids, and there is no correlation between different types of process parameters[3].

[2]The same model can be used to model the parameter variations across a reticle field containing multiple chips, in which case, these multiple chips can be analyzed simultaneously and the maximum of the delays at the POs of all chips is the distribution of chip delay. This does not change the complexity of the algorithm, since the number of dies in a reticle field is a small integer.

[3]This assumption is not critical to the correctness of our procedure, but is used in our experimental results. In case the assumption is not strictly true [21], our method is still general enough to handle correlations between parameters of different types, either by decomposing the correlated parameters into an uncorrelated set using an orthogonal transformation such as the principal component analysis (PCA) technique, or by constructing a covariance matrix for all correlated parameters.

For example, the L_g values for transistors in a grid are correlated with those in nearby grids, but are uncorrelated with other parameters such as W_g or W_{int_l} in any grid. (Note here that we consider interconnect parameters in different layers to be "different types of parameters," e.g., W_{int_1} and W_{int_2} are uncorrelated.)

Under this model, the parametric variation for a spatially correlated parameter in a single grid at location (x, y) can be modeled using a single random variable $p(x, y)$. In total, this representation requires n random variables, each representing the value of a parameter in one of the n grids, and a covariance matrix of size $n \times n$ representing the spatial correlations among the grids. The covariance matrix could be determined from data extracted from manufactured wafers. For example, a test structure methodology was developed to support the evaluation of process parameter variations in [22]. The number of grid regions divided can be also determined using the test structure methodology by refining the number of grids until delay distribution of test structure converges or changes only within a small tolerance range. In this work, due to the lack of access to real wafer data, we use the correlation matrix derived from the spatial correlation model in [3]. However, we believe that our model is more general than the model used in [3], since it is purely based on neighborhood. For example, consider again the case in Figure 1, by our model, the parameter in grid $(1, 2)$ has equal correlations with that in grid $(1, 1)$ and $(1, 3)$. While by the model of [3], it will have higher correlation with grid $(1, 1)$ than grid $(1, 3)$, i.e., the correlations are uneven at the two neighbors of grid $(1, 2)$.

For clarity of presentation, we here assume that all types of parameters have spatial correlations. In manufacturing, due to effects such as random dopant fluctuations, the intra-chip variations of some parameters are truly uncorrelated from transistor to transistor. The extension of this work to incorporate the effect of spatially uncorrelated parameters will be shown in Section VI.

IV. Statistical Timing Analysis Algorithm

The core statistical STA method is described in this section, and its description is organized as follows. At first, in section IV-A, we will describe how we model the distributions of gate and interconnect delays as normal distributions, given the PDF's that describe the variations of various parameters. In general, these PDF's will be correlated with each other. In section IV-B, we will show how we can simplify the complicated correlated structure of parameters by orthogonal transformations. Section IV-C will describe the PERT-like traversal algorithm on the statistical timing graph by demonstrating the procedure for the computation of *max* and *sum* functions. Finally, Section IV-D will explain why orthogonal transformations are important in our method.

A. Modeling Gate/Interconnect Delay PDF's

In this section, we will show how the variations in the process parameters are translated into PDF's that describe the variations in the gate and interconnect delays that correspond to the weights on edges of the statistical timing graph.

In section III, the geometrical parameters associated with the gate and interconnect are modeled as normally distributed random variables. Before we introduce how the distributions of gate and interconnect delays will be modeled, let us first consider an arbitrary function $d = F(\vec{P})$ that is assumed to be a function on a set of parameters $\vec{P}$, where each $p_i \in \vec{P}$ is a random variable with a normal distribution given by $p_i \sim N(\mu_{p_i}, \sigma_{p_i})$.

We can approximate the function d linearly using a first order Taylor expansion

$$d = d_0 + \sum_{\forall \text{ parameters } p_i} \left[\frac{\partial F}{\partial p_i}\right]_0 \Delta p_i, \quad (5)$$

where d_0 is the nominal value of d, calculated at the nominal values of parameters in $\vec{P}$, $\frac{\partial F}{\partial p_i}$ is computed at the nominal values of p_i, $\Delta p_i = p_i - \mu_{p_i}$ is a normally distributed random variable and $\Delta p_i \sim N(0, \sigma_{p_i})$.

In this approximation, d is modeled as a normal distribution, since it is a linear combination of normally distributed random variables. Its mean μ_d, and variance σ_d^2 are

$$\mu_d = d_0 \quad (6)$$

$$\sigma_d^2 = \sum_{\forall i} \left[\frac{\partial F}{\partial p_i}\right]_0^2 \sigma_{p_i}^2 + 2 \sum_{\forall i \neq j} \left[\frac{\partial F}{\partial p_i}\right]_0 \left[\frac{\partial F}{\partial p_j}\right]_0 cov(p_i, p_j), \quad (7)$$

where $cov(p_i, p_j)$ is the covariance of p_i and p_j.

It is reasonable to ask whether the approximation of d as a normal distribution is valid, since the distribution of d may, strictly speaking, not be Gaussian. We can say that when Δp_i has relatively small variations, the first order Taylor expansion is adequate and the approximation is acceptable with little loss of accuracy. This is generally true of intra-chip variations, where the process parameter variations are relatively small in comparison with the nominal values. For this reason, as functions of process parameters, the gate and interconnect delays can be approximated as a sum of normal distributions (which is also normal) applying equation (5).

Computing the PDF of interconnect delay: In this work, we use the Elmore delay model for simplicity to calculate the interconnect delays[4]. Under the Elmore model, the interconnect delay is a function of the vector of resistances, $\vec{R}_w$, the vector of capacitances, $\vec{C}_w$, of all wire segments in the interconnect tree, and the vector of input load capacitances, $\vec{C}_g$, of the fanout gates, or receivers:

$$d_{int} = D_{int}(\vec{R}_w, \vec{C}_w, \vec{C}_g). \quad (8)$$

Since the resistances and capacitances above are determined by the process parameters $\vec{P}$ of the interconnect and the receivers, such as W_{int_l}, T_{int_l}, H_{ILD_l}, W_g, L_g and T_{ox}, the sensitivities of the interconnect delay to a parameter p_i can be found by using the chain's rule

$$\frac{\partial d_{int}}{\partial p_i} = \sum_{\forall R_{w_k} \in \vec{R}_w} \frac{\partial D_{int}}{\partial R_{w_k}} \frac{\partial R_{w_k}}{\partial p_i} + \sum_{\forall C_{w_k} \in \vec{C}_w} \frac{\partial D_{int}}{\partial C_{w_k}} \frac{\partial C_{w_k}}{\partial p_i} + \sum_{\forall C_{g_k} \in \vec{C}_g} \frac{\partial D_{int}}{\partial C_{g_k}} \frac{\partial C_{g_k}}{\partial p_i}. \quad (9)$$

The distribution of interconnect delay can then be approximated on the computed sensitivities.

We will now specifically consider the factors that affect the interconnect delay associated with edges in the statistical timing graph. Recall that under our model, we divide the chip area into grids so that the parameter variations within a grid are identical, but those in different grids exhibit spatial correlations. Now consider an interconnect tree with several different segments that reside in different grids. The delay variations in the tree are affected by the parameter variations of wires in all grids that the tree traverses. For example, in Figure 1, consider the two segments uv and pq in the interconnect tree driven by gate a. Segment uv passes through the grid $(1,1)$ and pq through the grid $(1,2)$. Then the resistance and capacitance of segment uv should be calculated based on the process parameters of grid $(1,1)$, while the resistance and capacitance of segment pq should be based on those of grid $(1,2)$. Hence, the distribution of the interconnect tree delay is actually a function of random variables of interconnect parameters in both grid $(1,1)$ and grid $(1,2)$, and should incorporate any correlations between these random variables. Similarly, if the gates that the interconnect tree drives reside in different grid locations, the interconnect delay to any sink is also a function of random variables of gate parameters of all grids in which the receivers are located.

In summary, the distribution of interconnect delay function can be approximated by

$$\begin{aligned} d_{int} = {} & d_{int}^0 + \sum_{i \subset \Gamma_g} \left[\frac{\partial D_{int}}{\partial L_g^i}\right]_0 \Delta L_g^i + \sum_{i \subset \Gamma_g} \left[\frac{\partial D_{int}}{\partial W_g^i}\right]_0 \Delta W_g^i \\ & + \sum_{i \subset \Gamma_g} \left[\frac{\partial D_{int}}{\partial T_{ox}^i}\right]_0 \Delta T_{ox}^i + \sum_{l=1}^{n_{layer}} \left\{ \sum_{i \subset \Gamma_{int}} \left[\frac{\partial D_{int}}{\partial W_{int_l}^i}\right]_0 \Delta W_{int_l}^i \right. \\ & \left. + \sum_{i \subset \Gamma_{int}} \left[\frac{\partial D_{int}}{\partial T_{int_l}^i}\right]_0 \Delta T_{int_l}^i + \sum_{i \subset \Gamma_{int}} \left[\frac{\partial D_{int}}{\partial H_{ILD_l}^i}\right]_0 \Delta H_{ILD_l}^i \right\}, \end{aligned} \quad (10)$$

where d_{int}^0 is the interconnect delay value calculated with nominal values of parameters, Γ_g is the set of indices of grids that all the receivers reside in, Γ_{int} is the set of indices of grids that the interconnect tree traverses, and $\Delta L_g^i = L_g^i - \mu_{L_g^i}$ where L_g^i is the random variable representing transistor length in the i^{th} grid. The parameters ΔW_g^i, ΔT_{ox}^i, $\Delta W_{int_l}^i$, $\Delta T_{int_l}^i$ and $\Delta H_{ILD_l}^i$ are similarly defined. As before, the subscript "$_0$" next to each sensitivity represents the fact that it is evaluated at the nominal point.

Computing the PDFs of gate delay and output signal transition time: The distribution of gate delay and output signal transition time at the gate output can be approximated in a similar manner as described above, given the sensitivities of the gate delay to the process parameters.

Consider a multiple-input gate, let $d_{gate}^{pin_i}$ be the gate delay from the i^{th} input to the output and $S_{out}^{pin_i}$ be the corresponding

[4]However, it should be emphasized that any delay model may be used, and all that is needed is the sensitivity of the delay to the process parameters. For example, through a full circuit simulation, the sensitivities may be computed by performing adjoint sensitivity analysis.

output signal transition time. In general, both $d_{gate}^{pin_i}$ and $S_{out}^{pin_i}$ can be written as a function of the process parameters $\vec{P}$ of the gate, the loading capacitance of the driving interconnect tree $\vec{C}_w$ and the succeeding gates that it drives $\vec{C}_g$, and the input signal transition time $S_{in}^{pin_i}$ at this input pin of the gate

$$d_{gate}^{pin_i} = D_{gate}(\vec{P}, \vec{C}_w, \vec{C}_g, S_{in}^{pin_i}), \quad (11)$$

$$S_{out}^{pin_i} = S_{gate}(\vec{P}, \vec{C}_w, \vec{C}_g, S_{in}^{pin_i}). \quad (12)$$

The distributions of $d_{gate}^{pin_i}$ and $S_{in}^{pin_i}$ can be approximated as Gaussians using linear expressions of parameters, where the mean values of $d_{gate}^{pin_i}$ or $S_{in}^{pin_i}$ can be found by using the mean values of $\vec{P}$, $\vec{C}_w$, $\vec{C}_g$ and $S_{in}^{pin_i}$ in functions D_{gate} or S_{gate}, and the sensitivities of either $d_{gate}^{pin_i}$ or $S_{in}^{pin_i}$ to process parameters can be computed applying the chain's rule. The derivatives of $\vec{C}_w$ and $\vec{C}_g$ to the process parameters can be easily computed, as $\vec{C}_w$ and $\vec{C}_g$ are functions of process parameters. The input signal transition time, S_{in}, is a function of the output transition time of the preceding gate and the delay of the interconnect connecting the preceding gates and this gate, where both interconnect delay (as discussed earlier) and output transition time of the preceding gate (as will be shown in the next paragraph) are Gaussian random variables that can be expressed as a linear function of parameter variations. Therefore, at a gate input, the input signal transition time S_{in} is always given as a normally distributed random variable with a mean and first-order sensitivities to the parameter variations.

To consider the effect of non-ideal input signal on gate delay, the output signal transition time S_{out} at each gate output needs to be computed in addition to pin-to-pin delay of the gate. In conventional static timing analysis, S_{out} is set to $S_{out}^{pin_i}$ if the path ending at the output of the gate traversing the i^{th} input pin has the longest path delay d_{path_i}. In statistical static timing analysis, each of the paths through different gate input pins has a certain probability to be the longest path. Therefore, S_{out} should be computed as a weighted sum of the distributions of $S_{out}^{pin_i}$, where the weight equals the probability that the path through the i^{th} pin is the longest among all others:

$$S_{out} = \sum_{\forall input\ pin\ i} \{Prob[d_{path_i} > max_{\forall j \neq i}(d_{path_j})] \times S_{out}^{pin_i}\}, \quad (13)$$

where d_{path_i} is the random path delay variable at the gate output through the i^{th} input pin. The result of $max_{\forall j \neq i}(d_{path_j})]$ is a random variable representing for the distribution of maximum of multiple paths. As will be discussed later in Section IV-C, d_{path_i} and $max_{\forall j \neq i}(d_{path_j})$ can be approximated as Gaussians using *sum* and *max* operators, and their correlation can easily be computed. Therefore, finding the value of $Prob[d_{path_i} > max_{\forall j \neq i}(d_{path_j})$, i.e. $Prob[d_{path_i} - max_{\forall j \neq i}(d_{path_j} > 0)$ becomes computing the probability of a Gaussian random variable greater than zero, which can easily be found from a look-up table. As each $S_{out}^{pin_i}$ is a Gaussian random variable in linear combination of parameter variations, S_{out} is therefore also a Gaussian distributed random variable and its sensitivities to all process parameters $\frac{\partial S_{out}}{\partial p_i}$ can easily be found from its linear expression of parameters.

B. Orthogonal Transformation of Correlated Variables

In statistical timing analysis without spatial correlations, correlations due to reconvergent paths has long been an obstacle. When the spatial correlation of process parameters is also taken into consideration, the correlation structure becomes even more complicated. To make the problem tractable, we use the Principal Component Analysis (PCA) technique [23] to transform the set of correlated parameters into an uncorrelated set.

PCA is a method that can be employed to examine the relationship among a set of correlated variables. Given a set of correlated random variables $\vec{X}$ with a covariance matrix R, PCA can transform the set $\vec{X}$ into a set of mutually orthogonal random variables, $\vec{X'}$, such that each member of $\vec{X'}$ has zero mean and unit variance. The elements of the set $\vec{X'}$ are called principal components in PCA, and the size of $\vec{X'}$ is no larger than the size of $\vec{X}$. Any variable $x_i \in \vec{X}$ can then be expressed in terms of the principal components $\vec{X'}$ as follows:

$$x_i = (\sum_j \sqrt{\lambda_j} \cdot v_{ij} \cdot x'_j)\sigma_i + \mu_i, \quad (14)$$

where x'_j is a principal component in set $\vec{X'}$, λ_j is the j^{th} eigenvalue of the covariance matrix R, v_{ij} is the i^{th} element of the j^{th} eigenvector of R, and σ_i and μ_i are, respectively, the mean and standard deviation of x_i.

Since we assume that different types of parameters are uncorrelated, we can group the random variables of parameters by types and perform principal component analysis in each group separately, i.e., we compute the principal components for $\vec{L}_g$, $\vec{W}_g$, $\vec{T}_{ox}$, $\vec{N}_a$, $\vec{W}_{int_l}$ and $\vec{T}_{int_l}$ individually. Clearly, not only are the principal components of the same type of parameters independent, but so are the principal components of different type of parameters.

For instance, let $\vec{L}_g$ be a random vector representing transistor gate length variations in all grids and it is of multivariate normal distribution with covariance matrix R_{L_g}. Let $\vec{L'}_g$ be the set of principal components computed by PCA. Then any $L_g^i \in \vec{L}_g$ representing the variation of transistor gate length of the i^{th} grid can then be expressed as a linear function of the principal components

$$L_g^i = \mu_{L_g^i} + a_{i1} \times l_g^{'1} + \cdots + a_{it} \times l_g^{'t}, \quad (15)$$

where $\mu_{L_g^i}$ is the mean of L_g^i, $l_g^{'i}$ is a principal component in $\vec{L}'_g$, all $l_g^{'i}$ are independent with zero means and unit variances, and t is the total number of principal components in $\vec{L'}_g$.

In this way, any random variable in $\vec{W}_g$, $\vec{T}_{ox}$, $\vec{N}_a$, $\vec{W}_{int_l}$, $\vec{T}_{int_l}$ and $\vec{H}_{ILD_l}$ can be expressed as a linear function of the corresponding principal components in $\vec{W}'_g$, $\vec{T}'_{ox}$, $\vec{N}'_a$, $\vec{W}'_{int_l}$, $\vec{T}'_{int_l}$ and $\vec{H}'_{ILD_l}$. Superposing the set of rotated random variables of parameters on the original random variables in gate or interconnect delay in equation (10), the expression of gate or interconnect delay is then changed to the linear combination of principal components of all parameters

$$d = d_0 + k_1 \times p'_1 + \cdots + k_m \times p'_m, \quad (16)$$

where $p'_i \in \vec{P'}$ and $\vec{P'} = \vec{L}'_g \cup \vec{W}'_g \cup \vec{T}'_{ox} \cup \vec{N}'_a \cup \vec{W}'_{int_l} \cup \vec{T}'_{int_l} \cup \vec{H}'_{ILD_l}$ and m is the size of $\vec{P'}$.

Note that all of the principal components p'_i that appear in equation (16) are independent. Equation (16) has the following properties:

Property 1: Since all p'_i are orthogonal, the variance of d can be simply computed as

$$\sigma_d^2 = \sum_{i=1}^{m} k_i^2. \quad (17)$$

Property 2: The covariance between d and any principal component p'_i is given by

$$cov(d, p'_i) = k_i \sigma_{p'_i}^2 = k_i. \quad (18)$$

In other words, the coefficient of p'_i is exactly the covariance between d and p'_i.

Property 3: Let d_i and d_j be two random variables:

$$d_i = d_i^0 + k_{i1} \times p'_1 + \cdots + k_{im} \times p'_m, \quad (19)$$
$$d_j = d_j^0 + k_{j1} \times p'_1 + \cdots + k_{jm} \times p'_m. \quad (20)$$

The covariance of d_i and d_j, $cov(d_i, d_j)$, can be computed by

$$cov(d_i, d_j) = \sum_{r=1}^{m} k_{ir} k_{jr}. \quad (21)$$

In comparison, without an orthogonal transformation, the value of $cov(d_i, d_j)$ has to be computed by a more complicated formula as will be described in section IV-D.

C. PERT-like Traversal of Statistical STA

Using the techniques discussed up to this point, all edges of the statistical timing graph may be modeled as normally distributed random variables. In this section, we will describe a procedure for finding the distribution of the statistical longest path in the graph.

In conventional deterministic STA, the PERT algorithm can be used to find the longest path in a graph by traversing it in topological order using two types of functions:

- the *sum* function, and
- the *max* function.

In our statistical timing analysis, a PERT-like traversal is employed to find the distribution of circuit delay. However, unlike deterministic STA, the *sum* and *max* operations here are functions of a set of correlated multivariate Gaussian random variables instead of fixed values:

1) $d_{sum} = \sum_{i=1}^{l} d_i$, and
2) $d_{max} = max(d_1, \cdots, d_l)$.

where d_i is a Gaussian random variable representing either gate delay or wire delay expressed as linear functions of principal components in the form of equation (19), and l is the number of random variables that *sum* or *max* function is operating on.

Computing the distribution of the sum *function:* The computation of the distribution of *sum* function is simple. Since the $d_{sum} = \sum_{i=1}^{l} d_i$ is a linear combination of normally distributed random variables, d_{sum} is a normal distribution. The mean μ_{dsum} and variance σ_{dsum}^2 of the *sum* are given by

$$\mu_{d_{sum}} = \sum_{i=1}^{l} d_i^0, \quad (22)$$

$$\sigma_{d_{sum}}^2 = \sum_{j=1}^{m} \sum_{i=1}^{l} k_{ij}^2. \quad (23)$$

Computing the distribution of the max *function:* The *max* function of n normally distributed random variables $d_{max} = max(d_1, \cdots, d_l)$ is, strictly speaking, not Gaussian. However, we have found that, in practice, it can be approximated closely by a Gaussian. This idea is similar in spirit to Berkelaar's approach in [4], [24], although it is more general since Berkelaar's work restricted its attention to delay random variables that were uncorrelated[5]. In this work, we use the Gaussian distribution to approximate the result of a *max* function, so that $d_{max} \sim N(\mu_{d_{max}}, \sigma_{d_{max}})$. We also approximate d_{max} as a linear function of all the principal components $p'_1 \cdots p'_m$

$$d_{max} = \mu_{d_{max}} + a_1 p'_1 + \cdots + a_m p'_m. \quad (24)$$

Therefore, determining this approximation for d_{max} is equivalent to finding the values of $\mu_{d_{max}}$ and all a_i's.

From *Property 2* of Section IV-B, we know that the coefficient a_r equals $cov(d_{max}, p'_r)$. Then the variance of the expression on the right hand side of equation (24) is computed as $s_0^2 = \sum_{r=1}^{m} a_r^2 = \sum_{r=1}^{m} cov^2(d_{max}, p'_r)$. Since this is merely an approximation, there may be a difference between the value s_0^2 and the actual variance $\sigma_{d_{max}}^2$ of d_{max}. To diminish the difference, we can normalize the value of a_r by setting it as

$$a_r = cov(d_{max}, p'_r) \cdot \frac{\sigma_{d_{max}}}{s_0}. \quad (25)$$

We can see now that to find the linear approximation for d_{max}, the values of $\mu_{d_{max}}$, $\sigma_{d_{max}}$ and $cov(d_{max}, p_i)$ are required. In the work of [6], similar inputs were required in their algorithm and the results from [25] were applied and seen to provide good results. In this work, we have borrowed the same analytical formula from [25] for the computation of the *max* function.

According to [25], if ξ and η are two random variables, $\xi \sim N(\mu_1, \sigma_1)$, $\eta \sim N(\mu_2, \sigma_2)$, with a correlation coefficient of $r(\xi, \eta) = \rho$, then the mean μ_t and the variance σ_t^2 of $t = \max(\xi, \eta)$ can be approximated by

$$\mu_t = \mu_1 \cdot \Phi(\beta) + \mu_2 \cdot \Phi(-\beta) + \alpha \cdot \varphi(\beta), \quad (26)$$
$$\sigma_t^2 = (\mu_1^2 + \sigma_1^2) \cdot \Phi(\beta) + (\mu_2^2 + \sigma_2^2) \cdot \Phi(-\beta) + (\mu_1 + \mu_2) \cdot \alpha \cdot \varphi(\beta) - \mu_t^2, \quad (27)$$

where

$$\alpha = \sqrt{\sigma_1^2 + \sigma_2^2 - 2\sigma_1\sigma_2\rho}, \quad (28)$$
$$\beta = \frac{(\mu_1 - \mu_2)}{\alpha}, \quad (29)$$
$$\varphi(x) = \frac{1}{\sqrt{2\pi}} exp\left[-\frac{x^2}{2}\right], \quad (30)$$
$$\Phi(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} exp\left[-\frac{y^2}{2}\right] dy. \quad (31)$$

[5]Many researchers in the community were well aware of Berkelaar's results as early as 1997, though his work did not appear as an archival publication.

The formula will not apply if $\sigma_1 = \sigma_2$ and $\rho = 1$. However, in this case, the *max* function is simply identical to the random variable with largest mean value.

Moreover, from [25], if γ is another normally distributed random variable and the correlation coefficients $r(\xi, \gamma) = \rho_1$, $r(\eta, \gamma) = \rho_2$, then the correlation between γ and $t = \max(\xi, \eta)$ can be obtained by

$$r(t, \gamma) = \frac{\sigma_1 \cdot \rho_1 \cdot \Phi(\beta) + \sigma_2 \cdot \rho_2 \cdot \Phi(-\beta)}{\sigma_t}. \quad (32)$$

Using the formula above, we can find all the values needed. As an example, let us see how this can be done by first starting with a two-variable *max* function, $d_{max} = max(d_i, d_j)$. Let d_{max} be of the form of equation (24). We can find the approximation of d_{max} as follows:

1) Given the expressions of d_i and d_j each as linear combinations of the principal components, compute their mean and standard deviation values μ_{d_i}, σ_{d_i} and μ_{d_j}, σ_{d_j} respectively as described in *Property 1* of Section IV-B.
2) Find the correlation coefficient between d_i and d_j where $cov(d_i, d_j)$, the covariance of d_i and d_j can be computed using *Property 3* in Section IV-B.
   Now if $r(d_i, d_j) = 1$ and $\sigma_{d_i} = \sigma_{d_j}$, set d_{max} to be identical to d_i or d_j, whichever has larger mean value and we can stop here; otherwise, we will continue to the next step.
3) Calculate the mean $\mu_{d_{max}}$ and variance $\sigma^2_{d_{max}}$ of d_{max} using equations (26) and (27).
4) Find all coefficients a_r of p'_r. According to *Property 2*, $a_r = cov(d_{max}, p'_r)$, also, $cov(d_i, p'_r) = k_{ir}$ and $cov(d_j, p'_r) = k_{jr}$. Applying equation (32), the values of $cov(d_{max}, p'_r)$ and thus a_r can be calculated.
5) After all of the a_r's have been calculated, determine $s_0 = \sqrt{\sum_{r=1}^{m} {a_r}^2}$. Normalize the coefficient by resetting each $a_r = a_r \frac{\sigma_{d_{max}}}{s_0}$.

The calculation of the two-variable *max* function can easily be extended to a multi-variable *max* function by repeating the steps of the two-variable case recursively.

As mentioned at the beginning of this section, max of two Gaussian random variables is not strictly Gaussian. This approximation can sometimes introduce serious error, e.g. when the two Gaussian random variables have the same mean and standard deviation and correlation value of -1, and the distribution of the maximum is a half Gaussian. During the computation of multi-variable *max* function, some inaccuracy could be introduced since we approximate the *max* function as normal even though it is not really normal, and proceed with further recursive calculations. To the best of our knowledge, there is no theoretical analysis available in literature that quantifies the inaccuracies when a normal distribution is used to approximate the maximum of a set of Gaussian random variables. However, a numerically based analysis was provided in [25] which suggests that in some situations the errors can be great, but for many applications this approximate is quite satisfactory. We will show results in Section VII that suggest that such inaccuracies are not significant in the circuit context, and we will see that our results match very well with the simulation results from a Monte Carlo analysis.

Input: *Process parameter variations*
Output: *Distribution of circuit delay*

1) *According to the size of the chip, partition the chip region into $n = nrow \times ncol$ grids.*
2) *For each type of parameter, determine the n jointly normally distributed random variables and the corresponding covariance matrix.*
3) *Perform an orthogonal transformation to represent each random variable with a set of principal components.*
4) *For each gate and net connection, model their delays as linear combinations of the principal components generated in step 3.*
5) *Map the circuit into a statistical timing graph by adding one virtual-source node, one virtual-sink node and corresponding edges.*
6) *Using* sum *and* max *functions on Gaussian random variables, perform a PERT-like traversal on the graph to find the distribution of the statistical longest path. This distribution achieved is the circuit delay distribution.*

Fig. 2. Overall flow of our statistical timing analysis.

Also, recall that we have a "normalization" step to diminish the difference between the variance computed from the linear form of *max* approximation and the real variance of the *max* function. As in the case of approximating the *max* as normal distribution, there is no theoretical proof about how this "normalization" step can affect the accuracy of the approximation. Another option to diminish the difference is to move it into an independent random Gaussian component, and it is difficult to state definitively which of these options is better. In our work, we choose the former option and find that it provides excellent accuracy, as will be shown in Section VII, where the statistics of the "normalization" ratio for several test circuits are provided.

At this point, not only the edges, but also the results of *sum* and *max* functions are expressed as linear functions of the principal components. Therefore, using a PERT traversal by incorporating the computation of *sum* and *max* functions described above, the distribution of arrival time at any node in the timing graph becomes a linear function of principal components, and so the distribution of circuit delay can be computed at the virtual sink node.

The overall flow of our algorithm is shown in Figure 2. It is noticed that this work is in some sense parallel to the work of [14]: in [14], delays are represented as linear combinations of global random variables, while in our work, they are linear functions of principal components; in [14], the max of delays are reexpressed as linear functions using binding probabilities, while in our work, the linear functions are found by an analytical method from [25].

To further speed up the process, the following technique may be used: During the *max* operation of statistical STA, if the value of $\mu + 3 \cdot \sigma$ of one path has a lower delay than the value of $\mu - 3 \cdot \sigma$ of another path, we can simply calculate the

max function ignoring the former path.

D. The Utility of Principal Components

The previous sections described our statistical STA algorithm. The purpose of this section is to elaborate why the orthogonal transformation is needed to transform the set of correlated process parameters to an uncorrelated set, and how it can simplify the problem of statistical STA considering spatial correlations.

Let d_i and d_j be the distributions of two gate delays. For simplicity, we assume that the gate lengths $\vec{L}_g$ are the only spatially correlated parameters. We also assume that d_i and d_j are sensitive to the same set of correlated random variables of gate lengths $L_g^1, \ldots, L_g^n$. Using equation (10), d_i and d_j can be expressed as

$$d_i = d_i^0 + c_{i1} L_g^1 + \ldots + c_{in} L_g^n, \tag{33}$$

$$d_j = d_j^0 + c_{j1} L_g^1 + \ldots + c_{jn} L_g^n. \tag{34}$$

Obviously, the covariance of d_i and d_j is decided by the covariance structure of $\vec{L}_g$. The direct calculation of $cov(d_i, d_j)$ is of a complicated form as in the work of [5]

$$cov(d_i, d_j) = \sum_{a=1}^{n} \sum_{b=1}^{n} c_{ia} c_{jb} cov(L_g^a, L_g^b). \tag{35}$$

In contrast, in our method, we first perform orthogonal transformations on $\vec{L}_g$. Any element $L_g^l \in \vec{L}_g$ is expressed as

$$L_g^l = L_{g0}^l + a_{l1} l_g^{'1} + \ldots + a_{lm} l_g^{'m}. \tag{36}$$

Next, by superposition we transform d_i and d_j to:

$$d_i = d_i^0 + k_{i1} l_g^{'1} + \ldots + k_{im} l_g^{'m}, \tag{37}$$

$$d_j = d_j^0 + k_{j1} l_g^{'1} + \ldots + k_{jm} l_g^{'m}. \tag{38}$$

The value of $cov(d_i, d_j)$ can be simply computed using the coefficients of $\vec{L}_g'$ by $cov(d_i, d_j) = \sum_{r=1}^{m} k_{ir} k_{jr}$ in linear time $O(m)$. The advantage in this computation is that we do not need know which specific parameters in d_i and d_j are correlated. In fact, consider the coefficients of $l_g^{'1}$ in both d_i and d_j, $k_{i1} = \sum_{r=1}^{n} c_{ir} a_{r1}$ and $k_{j1} = \sum_{r=1}^{n} c_{jr} a_{r1}$. It can be seen that the covariance of gate lengths have all been incorporated in the coefficient of the principal components $l_g^{'1}, \ldots, l_g^{'n}$. For this reason, we ensure that the computation of $cov(d_i, d_j)$ can actually take the correlations of gate lengths into consideration correctly.

The direct computation of the covariance of path delays is in a similar form. In general, the path delays are correlated when the gate delays on these paths are correlated. As shown in the work of [5], the path covariances can be computed on the basis of pair-wise gate delay covariances; however, the number of paths is numerous which makes it computationally difficult to apply such a path-based method to large circuits.

In our method, with the orthogonal transformation, the covariances of path delays are manifested as the coefficients of the independent principal components as in the case of correlated gate delays. The covariances of the paths can then be simply computed in linear time based on these coefficients only, and there is no need to worry about how the gates on the paths are correlated or which parts are correlated. For the same reason, in this algorithm, besides the spatial correlations, path correlations due to reconvergence (structural correlations) can also be accounted for automatically by using the orthogonal transformation on the spatially correlated parameters. However, when spatially uncorrelated parameters are involved in the computation, the structural correlations due to these independent parameters can not be dealt with by this methodology. The extension of the work for handling spatially uncorrelated parameters will be given in Section VI-B.

V. Computational Complexity

We present a run time complexity analysis here to show which factors most greatly affect the CPU time of the algorithm.

The flow shown in Figure 2 can be divided into two parts: model pre-characterization (steps 1, 2 and 3) and statistical static timing analysis (SSTA) (steps 4, 5 and 6). Model pre-characterization consists of construction of parameter variations and grid-based spatial correlation models, and the computation of Principal Components (PC) for spatially correlated parameters. The computation of PCs requires calculations of eigenvectors and eigenvalues of the covariance matrix and its time complexity is $O(p \cdot n^3)$, where n is total number of grids divided and p is the number of parameters considered. While this step may seem to be a bottleneck of the algorithm, it is a only one-time process. Once the models of parameter variations are constructed, they can be repeatedly used to analyze any design. Meanwhile, for spatial correlated parameters, the PCs computed from the covariance matrix are only model-dependent, so that for different designs analyzed with the same parameter model, the same set of PCs can be applied. In other words, the step of model pre-characterization is in fact a one-time library construction at early stage and therefore can be excluded from the run time complexity analysis of the algorithm.

The run-time of the SSTA algorithm can be divided into:

1) The time required to find the delay distribution of the gate and interconnect[6]: This run time depends on how many different grids the interconnect passes through and how many grids the gates are located in, and in general these numbers are bounded by constant numbers. The run time is also proportional to the total number of principal components, since we perform orthogonal transformation at each wire segment of interconnect. For each random variable, the number of principal components is no more than the total number of grids n partitioned on the chip. The total number of principal components is no more than $p \cdot n$. Thus, the time required to find the distribution of a single gate or wire can be estimated as $O(p \cdot n)$. If N_g is the total number of gates and N_I the number of net connections in the circuit, the time of this part can be estimated as $O(p \cdot n \cdot (N_g + N_I))$.
2) The time required to evaluate the *max* function: The cost of this operation is proportional to the number of random

[6]The time to characterize the sensitivities of delay on parameter variations is excluded from this analysis.

variables involved in the max operation and the number of principal components of each random variable. The *max* operation is used at all multi-input gates and at the last level (sink node) where the maximum circuit delay is computed. This number can be upper bounded by the total number of net connections N_I in the circuit. Thus, the run time of this part is $O(p \cdot n \cdot N_I)$.

3) The time required to compute output transition time at each gate output: For a gate with $k > 2$ inputs, it requires k^2 *max* operations and $k - 1$ *sum* operations, which are constant numbers of *max* and *sum* operations. The computation is needed for all gates and thus the total cost is $O(p \cdot n \cdot N_g)$.
4) The time required to evaluate the *sum* function: The *sum* operation must be performed at all gates and interconnects encountered during the PERT-like traversal. A single *sum* operation requires $O(n)$, and therefore, the total complexity for this part is $O(p \cdot n \cdot (N_g + N_I))$.

Therefore, the run time complexity of the algorithm is $O(p \cdot n \cdot (N_g + N_I))$, which is $p \cdot n$ times that of deterministic STA.

VI. Extending the Method to Handle Inter-Chip Variations, Spatially Uncorrelated Intra-Chip Parameters, and Min-delay Computations

In this section, we will first describe how this work can be extended to include the effect of inter-chip variations in addition to intra-chip variations. Subsequently, we will explain how spatially uncorrelated parameters can be incorporated into the current proposed algorithm. Finally, we will show how minimum delay computations can easily be incorporated into this framework.

A. Inter-Chip Variations

In general, the process parametric variation can be modeled as

$$\delta_{total} \quad = \quad \delta_{inter} + \delta_{intra}, \tag{39}$$

where δ_{inter} is the inter-chip variation and δ_{intra} is the intra-chip variation. As for δ_{intra}, δ_{inter} is also modeled as a Gaussian random variable.

As introduced in Section I, inter-chip variation has a global effect on all the transistors [wires] within a single chip, and therefore a single random variable, δ_{inter}, can be applied to all transistors [wires] to model the effect of inter-die variation. Consequently, the covariance matrix for each type of spatially correlated parameter is changed by adding to all entries a value of $\sigma^2_{\delta_{inter}}$, the variance of inter-chip parametric variation. Based on the new covariance matrices, the same statistical STA methodology can still be applied to compute distribution of chip delay.

B. Spatially Uncorrelated Parameters

In practice, it is observed that not all process parameters are spatially correlated. For example, the variations of T_{ox} or N_a are independent from transistor to transistor. To model the intra-die variation of spatially uncorrelated parameter, a separate random variable has to be used for each gate [wire] to represent such independence, instead of a single random variable for all gates [wires] in the same grid for the spatial correlated parameters. Consequently, the timing analysis framework introduced in previous sections must be further extended to accommodate the spatially uncorrelated parameters.

As an example, let us consider the case that gate oxide thickness T_{ox} is the only spatially uncorrelated parameter. The idea described here can easily be extended to the case where there is more than one uncorrelated parameter. With inter- and intra-chip variations, the variation of T_{ox} for the i_{th} transistor can be expressed as $\delta^{inter}_{T_{ox}} + \Delta T^i_{ox}$, where $\delta^{inter}_{T_{ox}}$ is the random variable representing for the inter-chip variation of T_{ox}, and ΔT^i_{ox} the intra-chip variation of T_{ox} of the i_{th} transistor. Accordingly, the expressions for device [wire] delays are reformulated by substituting $\delta^{inter}_{T_{ox}} + \Delta T^i_{ox}$ for where ΔT_{ox} of the i_{th} transistor appears. Since the orthogonal transformations of parameters are performed only on spatially correlated parameters, the variables $\delta^{inter}_{T_{ox}}$ and ΔT^i_{ox} are preserved in the delay expressions of linear combination of principal components and either variable is independent from the principal components and any other random variables in the delay expressions. The timing propagation using the *sum* and *max* operators remains the same, except that after each *sum* or *max* operation, the random variables for intra-die variations of spatially uncorrelated parameters, ΔT^i_{ox}'s, are merged into one random variable, so that, at each arrival time, only one independent random variable is kept for all intra-die variations of spatially uncorrelated parameters. It is observed that the way of adding this independent random variable to the standard form of the representation of arrival times is similar to the "residual" variance's lumping into the independently random part in [26].

Although structural correlations can be automatically taken into account using orthogonal transformation on spatially correlated parameters as explained in Section IV-D, the structural correlations due to spatially uncorrelated parameters cannot be handled with the same technique because of the merging of these random variables during the propagation. To reduce the inaccuracies caused, one can appeal to the available literature on handling structural correlations in statistical STA [7], [9], [10]. In this work, we have ignored the structural correlations caused by the spatially uncorrelated parameters. However, since the structural correlations from spatially correlated parameters are considered, the inaccuracies introduced from this assumption are not significant, as will be demonstrated in Section VII.

C. Distribution of the Minimum of a Set of Gaussians

In circuit performance analysis, computations such as finding the required arrival time (RAT) for long-path analysis, and minimum delay computations for short-path analysis (to check for hold time violations) require the computation of the minimum of a set of delays, which becomes finding the distribution of the minimum of a set of random variables under process variations.

The procedure for calculation of maximum of a set of Gaussians can be utilized to compute the minimum of a set of Gaussian random variables, $d_1 \cdots d_l$. Specifically, $d_{min} = min(d_1, ..., d_l)$ can be computed as

$$d_{min} = -max(-d_1, ..., -d_l), \tag{40}$$

where d_i is a normally distributed random variable and *max* is the operator introduced is Section IV-C.

VII. Experimental Results

The proposed algorithm was implemented in C++ as the software package *"MinnSSTA*," and tested on the edge-triggered ISCAS89 benchmark circuits by working on the combinational logic blocks between the latches. All experiments were run on a Linux PC with a 2.0GHz CPU and 256MB memory. We experimented with parameters of 100nm technologies on a 2-metal layer interconnect model. The process parameters (Table I) used here are based on predictions from [20], [27].

Since the computation requires physical information about the locations of the gates and interconnects, all cells in the circuit were first placed using the placement tool, Capo [28]. Global routing was then performed to route all the nets in the circuits. Depending on the size of circuit, we divided the chip area into different sizes of grids, so that each grid contains no more than a hundred cells. Again, due to the lack of access to real wafer data, the covariance matrix for intra-die variations used in this work were derived from the spatial correlation model used in [3] by equally splitting the variance into all levels.

To verify the results of our method *MinnSSTA*, we used Monte Carlo (*MC*) simulations based on the same grid models for comparison. To balance the accuracy and run time, we chose to run 10,000 iterations for the Monte Carlo simulation.

We first present the experimental results assuming that all parameters are spatially correlated while using fixed values for the spatially uncorrelated parameters (T_{ox} and N_a). Table II shows a comparison of the results of *MC* with those from *MinnSSTA*. For each test case, the mean and standard deviation (SD) values for both methods are listed. The results of *MinnSSTA* can be seen to be very close to the *MC* results: the average error is -0.23% for the mean and -0.32% for the standard deviation. In Figure 3, for the largest test case s38417, the plots of the PDF and CDF of the circuit delay for both *MinnSSTA* and *MC* methods are provided. It is observed that the curves almost perfectly match each other. This demonstrates the accuracy of the PCA approach for correlated parameters, including its ability to account for structural correlations.

Next, the results for considering the variations of the spatially uncorrelated parameters (T_{ox} and N_a) are given in Table III. On average, the error is 1.06% for the mean value and -4.34% for the standard deviation. In Table VIII, the 99% and 1% confidence points achieved by *MC* and *MinnSSTA* are also provided and the average errors are -2.46% and -0.99% respectively. Again, for the largest test case s38417, the PDF and CDF curves of the circuit delay for both *MinnSSTA* and

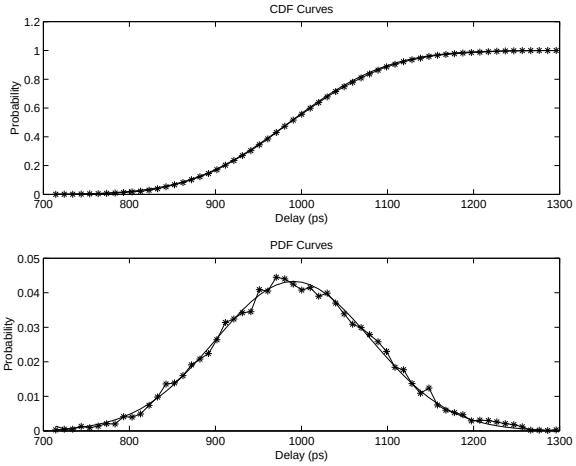


Fig. 3. A comparison of *MinnSSTA* and *MC* methods (assuming fixed values of T_{ox} and N_a) for circuit s38417. The curve marked by the solid line denotes the results of *MinnSSTA*, while the plot marked by the starred lines denotes the results of *MC*.

MC methods are plotted in Figure 4, It can be seen that, at the range of lower and higher circuit delay values, the circuit delay distribution computed from *MinnSSTA* matches well with that of the Monte-Carlo simulation, although there are some deviations in the central portion. As mentioned in Section VI-B, some error may be introduced from the structural correlations, which are not handled exactly in the presence of uncorrelated intra-die components. Based on our analysis of the experiments, we find that the cause for the small error that is introduced here is primarily because our implementation does not handle structural correlations between the uncorrelated variables. We believe that, by appending into the existing framework an algorithm that handles structural correlation [7], [9], [10], the error of the results in Table III can be further reduced.

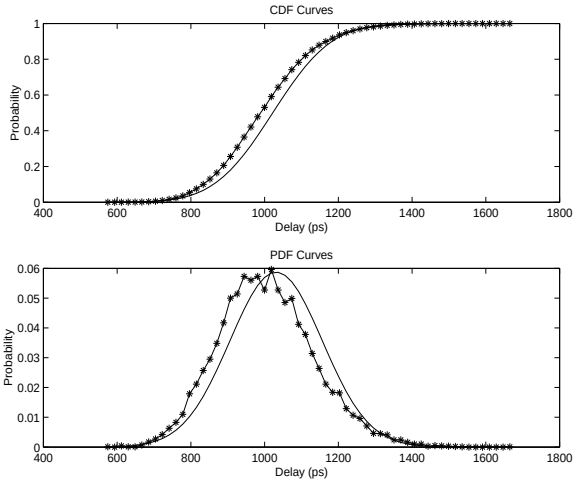


Fig. 4. A comparison of *MinnSSTA* and *MC* methods for circuit s38417, considering all sources of variation, some of which are spatially correlated and some of which are not. The curve marked by the solid line denotes the results of *MinnSSTA*, while the plot marked by the starred lines denotes the results of *MC*.

In Table III, the CPU times for both methods are provided.

TABLE I

PARAMETERS USED IN THE EXPERIMENTS

Parameters	L_g (nm)	W_g (nm)	T_{ox} (nm)	N_a ($\times 10^{17} cm^{-3}$) nmos/pmos	W_{int} (nm)	T_{int} (nm)	H_{ILD} (nm)
$\bar{p}$	60.0	150.000	2.500	9.70000/10.04000	150.0	500.0	300.0
$3\sigma_{inter}$	9.0	11.250	0.250	0.72750	15.0	25.0	22.50
$3\sigma_{intra}$	4.5	5.625	0.125	0.36375	7.5	12.5	11.25
$\delta_x x_{max} + \delta_y y_{max}$	4.5	5.625	0.125	0.36375	7.5	12.5	11.25

TABLE II

COMPARISON RESULTS ASSUMING FIXED VALUES OF T_{ox} AND N_a

Benchmark	Monte-Carlo (MC)		MinnSSTA		$\frac{(MinnSSTA-MC)}{MC}$%	
Name	Mean(ps)	SD(ps)	Mean(ps)	SD(ps)	Mean	SD
s38417	988.6	91.0	985.8	90.8	-0.28%	-0.22%
s38584	1726.9	153.1	1720.9	151.6	-0.35%	-0.98%
s35932	1165.5	101.6	1162.7	101.3	-0.24%	-0.30%
s15850	1370.2	131.1	1367.2	129.6	-0.22%	-1.14%
s13207	1219.9	116.1	1217.3	116.2	-0.21%	0.09%
s9234	674.6	65.4	673.7	64.8	-0.13%	-0.92%
s5378	413.1	38.5	411.8	38.4	-0.31%	-0.26%
s1196	499.9	45.8	499.3	46.2	-0.12%	0.87%
s27	102.5	9.9	102.3	9.9	-0.20%	0.00%

TABLE III

COMPARISON RESULTS OF THE PROPOSED METHOD AND MONTE-CARLO SIMULATION METHOD

Benchmark			Monte-Carlo (MC)			MinnSSTA				$\frac{(MinnSSTA-MC)}{MC}$%	
Name	#Cells	#Grids	Mean(ps)	SD(ps)	CPU-time(s)	Mean(ps)	SD(ps)	CPU-time(s)	PCA-time(s)	Mean	SD
s38417	23815	256	995.6	130.3	21005	1022.0	125.4	406.11	0.15	2.65%	-3.76%
s38584	20705	256	1738.4	226.4	24039	1798.2	215.6	460.36	0.15	3.44%	-4.77%
s35932	17793	256	1214.7	161.8	53922	1251.2	144.7	505.71	0.15	3.00%	-10.57 %
s15850	10369	256	1388.2	178.9	8856	1397.8	172.1	175.96	0.15	0.69%	-3.80%
s13207	8260	256	1230.7	158.8	9060	1239.7	154.9	172.62	0.15	0.73%	-2.46%
s9234	5825	64	688.6	90.6	5346	690.6	85.2	32.23	0.02	0.29%	-5.96%
s5378	2958	64	421.1	54.3	3907	420.8	51.8	27.41	0.02	-0.07%	-4.60%
s1196	547	16	505.9	66.0	781	502.7	64.4	1.51	0.01	-0.63%	-2.42%
s27	13	4	103.6	13.7	9	103.0	13.6	0.00	0.00	-0.58%	-0.73%

To show that the PCA steps require very little run time, the run time for this part is also listed; however, as pointed out earlier, this can be considered a preprocessing step that is carried out once for each technology, and its cost need not be considered in the computation. We can see that the CPU time of *MinnSSTA* on all test cases is very fast. The circuit with the longest run time, s35932, was analyzed in only about 500 seconds, while the *MC* simulation required over 15 hours.

In the proposed approach, in order to make the computed value of standard deviation of d_{max} the same as that of the approximated linear expression, the coefficients of parameters in the linear expression are normalized by the ratio of the standard deviation of d_{max} (namely, $\sigma_{d_{max}}$) to that of the linear expression s_0. In Table IV, the statistics of this ratio for all testcases are listed, including the mean, standard deviation, minimum and maximum values of the ratio and the probability of the ratio falls into each given range. In general, the higher the ratio, the larger the error for estimating d_{max}, and thus the less accurate for estimating the circuit delay distribution using the proposed approach. For example, the testcase $s35932$ has the highest probability of 0.045 for the ratio to be greater than 1.1, and also has the largest errors predicting the circuit mean and standard deviation. Over all testcases, the average value of the ratio is 1.003, which is a reasonably small number so that the accuracy of the proposed statistical SSTA should not be affected significantly by this normalization step.

To further verify the applicability of the proposed algorithm, we have demonstrated it on a path-balanced circuit whose topology is a binary tree of depth 10. Table V lists the results achieved by *MinnSSTA* and (*MC*). The errors obtained are -0.54% for the mean and -6.26% for the standard deviation; -4.56% and -1.65% for the 99% and 1% confidence point, respectively. This shows that the proposed approach can predict the timing yield well, even for path-balanced circuits.

One may ask what happens if a Monte-Carlo approach was run for the same amount of time as the proposed algorithm. In Table VI, we show the data achieved from Monte-Carlo runs in the equivalent CPU-time of the proposed method "MinnSSTA". Since this Monte-Carlo simulation can only run a small number of iterations and samples the soluton space insufficiently, it does not meet any of the usual convergence criteria used for Monte Carlo analysis. Therefore, what is achieved is not the genuine distribution of circuit delay, but merely the distribution from an incomplete number of runs. The table shows the minimum values and maximum values of the circuit delay from this insufficient number of Monte-Carlo runs, and, for purposes of comparison, the results with the 1% and 99% confidence points, respectively, from the 10,000 iterations of Monte-Carlo simulation. It can be seen that the accuracy is highly variable: in some cases, Monte Carlo analysis comes close to the action value, while in others, it is very far away. Most notably, large deviations can be seen both for a small circuit (s27) and a large circuit (s38584),

TABLE IV

STATISTICS OF RATIO OF STANDARD DEVIATION OF ACCURATE VALUE $\sigma_{d_{max}}$ TO s_0 OF THE LINEAR EXPRESSION.

Circuit	Ratio of $\sigma_{d_{max}}$ to s_0				Probability of the ratio in each range				
Name	mean	stdev	minimum	maximum	< 1	$= 1$	$(1, 1.01)$	$[1.01, 1.1]$	> 1.1
s38417	1.0031	0.0051	≈ 1	1.0262	0.0004	0.3246	0.5582	0.1168	0
s38584	1.0037	0.0054	≈ 1	1.1804	0.0023	0.4124	0.1700	0.0001	0
s35932	1.0120	0.0278	≈ 1	1.1583	0.0022	0.2883	0.4290	0.2350	0.0454
s15850	1.0018	0.0033	≈ 1	1.0233	0.0034	0.4029	0.5538	0.0398	0
s13207	1.0028	0.0048	≈ 1	1.0260	0.0008	0.3256	0.5843	0.0893	0
s9234	1.0017	0.0035	1	1.0209	0	0.3825	0.5636	0.0538	0
s5378	1.0012	0.0023	1	1.0289	0	0.4310	0.5563	0.0126	0
s1196	1.0007	0.0021	≈ 1	1.0150	0.0021	0.7068	0.2764	0.0148	0
s27	1.0006	0.0014	1	1.0030	0	0.8	0.2000	0	0

TABLE V

EXPERIMENTAL RESULTS ON A BINARY TREE CIRCUIT OF DEPTH-10

Approach	Mean(ps)	SD(ps)	99% Point(ps)	1% Point(ps)
MC	669.8	86.2	894.8	486.3
MinnSSTA	666.2	80.8	854.0	478.3
$\frac{(MinnSSTA-MC)}{MC}\%$	-0.54%	-6.26%	-4.56%	-1.65%

implying that the reliability of such an approach is suspect. Of course, this is not surprising in the least, because the artificial limitation on the run time has made the Monte Carlo analysis unreliable, by permitting only a low point of confidence for its predictions, and has not permitted it to fully sample the search space.

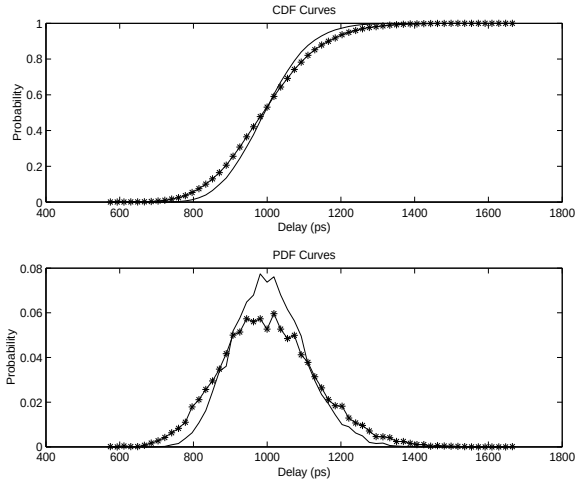


Fig. 5. A comparison of statistical STA with and without considering spatial correlations, under Monte Carlo analysis, for circuit s38417. The curve marked by the solid line denotes the case where spatial correlations are ignored, while the curve with the starred lines denotes the results of incorporating spatial correlations; this is identical to the curve in Figure 4.

To show the importance of considering spatial correlations, we consider the difference between performing statistical timing analysis while considering spatial correlation and while ignoring it. Since this is a comparison to determine why spatial correlations are important, the CPU time is not a consideration. Therefore, we run another set of Monte Carlo simulations (*MCNoCorr*) on the same set of benchmarks, this time assuming zero correlations among the devices and wires on the chip. The comparison between the data is shown in Table VII. It can be observed that although the mean values are close, the variances of the uncorrelated cases (*MCNoCorr*) are much smaller than the correlated cases (*MC*). On average, the standard deviation of the correlated case increases by 25.93%. Again, we plot the PDF and CDF curves of both simulations for circuit s38417 in Figure 5. It is seen that the CDF and PDF curves of *MCNoCorr* deviate significantly from those of *MC*. In other words, statistical timing analysis without considering correlation may incorrectly predict the real performance of the circuit and could even overestimate the performance of the circuit. This underlines the importance of developing efficient statistical STA methods that can incorporate spatial correlations.

As an alternative, we consider the option of using multiple process corners (*MPC*) for these experiments, where the circuit delays are evaluated at all possible corners of parameter values at $\mu \pm 3 \cdot \sigma$, where μ is the mean and σ the standard deviation for the parameter. Table VIII compares the worst-case and best-case delays obtained at exhaustive process corners using the *MPC* method, with the 99% and 1% confidence point delay achieved from the Monte-Carlo simulation (*MC*) accordingly. On average, the *MPC* approach overestimates the worst-case delay of circuit by 30.81% and underestimates the best-case delay by 28.08%. These results also emphasize the importance of considering spatial correlations during statistical STA, as is done by our algorithm.

VIII. CONCLUSION AND FUTURE WORK

In this paper, we have proposed an algorithm for performing statistical STA, considering spatial correlations related to intra-chip process variations. We show that performing statistical timing analysis while ignoring spatial correlations may not be adequate to predict the circuit performance correctly, and that fast and accurate statistical STA methods, such as ours, that incorporate spatial correlations are essential. An analysis of the complexity shows it to be reasonable, and like conventional STA, it is linear in the number of gates and interconnects. The penalty that is paid here is that unlike deterministic STA, it is also linear in the number of grid squares. As a trivial extension of maximum of delays, the computation for the distribution of minimum of delays is also provided.

TABLE VI

COMPARISON OF MONTE-CARLO IN EQUIVALENT CPU-TIME OF MINNSSTA WITH THAT OF 10,000 RUNS (MC)

Circuit	Minimum Delay		MC: $1\%pt$ (ps)	$\frac{mindelay-1\%pt}{1\%pt}\%$	Maximum delay		MC: $99\%pt$ (ps)	$\frac{maxdelay-99\%pt}{99\%pt}\%$
Name	$\#runs$	delay (ps)	$(10,000runs)$		$\#runs$	delay (ps)	$(10,000runs)$	
s38417	170	725.0	722.0	0.42%	170	1372.8	1333.3	2.96%
s38584	180	1205.7	1261.3	-4.41%	180	2794.3	2310.3	20.95%
s35932	100	847.3	882.3	-3.97%	100	1650.0	1635.2	0.91%
s15850	175	994.4	1012.9	-1.82%	175	1880.6	1844.8	1.94%
s13207	190	887.8	893.1	-0.59%	190	1728.1	1629.9	6.02%
s9234	60	452.3	499.7	-9.49%	60	884.1	922.6	-4.17%
s5378	50	306.0	308.9	-0.92%	50	534.3	559.9	-4.58%
s1196	16	441.7	370.4	19.26%	16	637.8	673.4	-5.29%
s27	10	350.0	74.9	367.30%	10	653.8	138.4	372.37%

TABLE VII

COMPARISON OF TIMING ANALYSIS WITH AND WITHOUT SPATIAL CORRELATIONS

Benchmark	Anal. w/ corr. (MC)		Anal. w/o corr. (MCNoCorr)		$\frac{(MC-MCNoCorr)}{MCNoCorr}\%$	
Name	Mean(ps)	SD(ps)	Mean(ps)	SD(ps)	Mean	SD
s38417	995.6	130.3	996.7	98.7	0.11%	-24.25%
s38584	1738.4	226.4	1741.9	180.5	0.20%	-20.27%
s35932	1214.7	161.8	1253.6	140.0	3.20%	-13.47%
s15850	1388.2	178.9	1393.8	121.9	0.40%	-31.86%
s13207	1230.7	158.8	1233.8	110.2	0.25%	-30.60%
s9234	688.6	90.6	691.9	61.9	0.48%	-31.68%
s5378	421.1	54.3	424.7	38.2	0.85%	-29.65%
s1196	505.9	66.0	507.6	48.8	0.34%	-26.06%
s27	103.6	13.7	103.7	10.2	0.10%	-25.55%

TABLE VIII

COMPARISON OF 99% AND 1% CONFIDENCE POINT

Bench.	MC		MinnSSTA		$\frac{(MinnSSTA-MC)}{MC}\%$		MPC		$\frac{(MPC-MC)}{MC}\%$	
Name	99% Pt.(ps)	1% Pt.(ps)	99% Pt.(ps)	1% Pt.(ps)	99% Pt.(ps)	1% Pt.(ps)	Worst-Case	Best-Case	Worst-Case	Best-Case
s38417	1333.3	722	1313.6	730.4	-1.48%	1.16%	1758.1	522.1	31.86%	-27.69%
s38584	2310.3	1261.3	2299.5	1296.9	-0.47%	2.82%	3056.0	915.4	32.28%	-27.42%
s35932	1635.2	882.3	1587.6	914.8	-2.91%	3.68%	2051.2	613.0	25.44%	-30.52%
s15850	1844.8	1012.9	1797.9	997.7	-2.54%	-1.50%	2442.9	725.2	32.42%	-28.40%
s13207	1629.9	893.1	1599.8	879.6	-1.85%	-1.51%	2175.4	646.6	33.47%	-27.60%
s9234	922.6	499.7	888.7	492.5	-3.67%	-1.44%	1207.3	359.7	30.86%	-28.02%
s5378	559.9	308.9	541.2	300.4	-3.34%	-2.75%	736.6	219.2	31.56%	-29.04%
s1196	673.4	370.4	652.4	353.0	-3.12%	-4.70%	874.2	265.8	29.82%	-28.24%
s27	138.4	74.9	134.6	71.4	-2.75%	-4.67%	179.3	55.6	29.55%	-25.77%

The current algorithm is limited by the following: it assumes that the distribution of parameter variations are Gaussian and the distribution of gate [wire] delays have linear dependency on the variation of process parameters. A good direction for future research involves solving the problem of statistical timing analysis on non-Gaussian process parameter variations and nonlinear delay dependencies.

ACKNOWLEDGMENTS

The authors would like to thank the anonymous reviewers, whose comments provided excellent feedback, resulting in an improvement in the quality of our paper.

REFERENCES

[1] S. Nassif, "Design for variability in DSM technologies," in *Proceedings of the IEEE International Symposium on Quality of Electronic Design*, San Jose, California, USA, Mar. 2000, pp. 451–454.

[2] V. Axelrad and J. Kibarian, "Statistical aspects of modern IC designs," in *Proceedings of the 28th European Solid-State Device Research Conference*, Bordeaux, France, Sept. 1998, pp. 309–321.

[3] A. Agarwal, D. Blaauw, V. Zolotov, S. Sundareswaran, M. Zhao, K. Gala, and R. Panda, "Statistical delay computation considering spatial correlations,," in *Proceedings of the Asia and South Pacific Design Automation Conference*, Kitakyushu, Japan, Jan. 2003, pp. 271–276.

[4] M. Berkelaar, "Statistical delay calculation, a linear time method," (Personal communication).

[5] M. Orshansky and K. Keutzer, "A general probabilistic framework for worst case timing analysis," in *Proceedings of the ACM/IEEE Design Automation Conference*, New Orleans, Louisiana, USA, June 2002, pp. 556–561.

[6] S. Tsukiyama, M. Tanaka, and M. Fukui, "A statistical static timing analysis considering correlations between delays," in *Proceedings of the Asia and South Pacific Design Automation Conference*, Yokohama, Japan, Jan. 2001, pp. 353–358.

[7] A. Agarwal, D. Blaauw, V. Zolotov, and S. Vrudhula, "Computation and refinement of statistical bounds on circuit delay," in *Proceedings of the ACM/IEEE Design Automation Conference*, Anaheim, California, USA, June 2003, pp. 348–353.

[8] J. Liou, K. Cheng, S. Kundu, and A. Krstic, "Fast statistical timing analysis by probabilistic event propagation," in *Proceedings of the ACM/IEEE Design Automation Conference*, Las Vegas, Nevada, USA, June 2001, pp. 661–666.

[9] S. Naidu, "Timing yield calculation using an impulse-train approach," in *Proceedings of the 15th International Conference on VLSI Design*, Bangalore, India, Jan. 2002, pp. 219–224.

[10] A. Devgan and C. Kashyap, "Block-based static timing analysis with uncertainty," in *Proceedings of the IEEE/ACM International Conference on Computer Aided Design*, San Jose, California, USA, Nov. 2003, pp. 607–614.

[11] S. Bhardwaj, S. Vrudhula, and D. Blaauw, "TAU: Timing analysis under uncertainty," in *Proceedings of the ACM/IEEE International Conference on Computer Aided Design*, San Jose, California, USA, Nov. 2003, pp. 615–620.

[12] J. Liou, A. Krstic, L. Wang, and K. Cheng, "False-path-aware statis-

tical timing analysis and efficient path selection for delay testing and timing validation," in *Proceedings of the ACM/IEEE Design Automation Conference*, New Orleans, Louisiana, USA, June 2002, pp. 566–569.

[13] B. Choi and D. M. H. Walker, "Timing analysis of combinational circuits including capacitive coupling and statistical process variation," in *Proceedings of the IEEE VLSI Test Symposium*, Montreal, Canada, Apr. 2000, pp. 49–54.

[14] J. Jess, K. Kalafala, S. Naidu, R. Otten, and C. Visweswariah, "Statistical timing for parametric yield prediction of digital untegrated circuits," in *Proceedings of the ACM/IEEE Design Automation Conference*, Anaheim, California, USA, June 2003, pp. 932–937.

[15] A. Agarwal, D. Blaauw, and V. Zolotov, "Statistical timing analysis for intra-die process variations with spatial correlations," in *Proceedings of the IEEE/ACM International Conference on Computer Aided Design*, San Jose, California, USA, Nov. 2003, pp. 900–907.

[16] S. Sapatnekar, *Timing*. Boston, MA: Kluwer Academic Publishers, 2004.

[17] T. Kirkpatrick and N. Clark, "PERT as an aid to logic design," *IBM Journal of Research and Development*, vol. vol. 10, pp. 135–141, June 1966.

[18] Y. Liu, S. Nassif, L. Pileggi, and A. Strojwas, "Impact of interconnect variations on the clock skew of a gigahertz microprocessor," in *Proceedings of the ACM/IEEE Design Automation Conference*, Los Angeles, California, USA, June 2000, pp. 168–171.

[19] M. Orshansky, L. Milor, P. Chen, K. Keutzer, and C. Hu, "Impact of spatial intrachip gate length variability on the performance of high-speed digital circuits," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 21*, pp. 544–553, May 2002.

[20] S. Nassif, "Delay variability: Sources, impact and trends," in *Proceedings of IEEE International Solid-State Circuits Conference*, San Francisco, California, USA, Feb. 2000, pp. 368–369.

[21] B. Stine, D. Boning, and J. Chung, "Analysis and decomposition of spatial variation in integrated circuit processes and devices," *IEEE Transaction on Semiconductor Manufacturing*, vol. vol. 10, pp. 24–41, Feb. 1997.

[22] D. Boning, J. Panganiban, K. Gonzalez-Valentin, S. Nassif, C. McDowell, A. Gattiker, and F. Liu, "Test structures for delay variability," (Personal communication).

[23] D. Morrison, *Multivariate Statistical Methods*. New York, NY: McGraw-Hill, 1976.

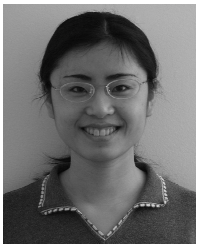
[24] E. Jacobs and M. Berkelaar, "Gate sizing using a statistical delay model," in *Proceedings of Design Automation and Test in Europe*, 2000, pp. 283–290.

[25] C. Clark, "The greatest of a finite set of random variables," *Operations Research, vol. 9*, pp. 85–91, 1961.

[26] C. Visweswariah, K. Ravindran, K. Kalafala, S. G. Walker, and S. Narayan, "First-order incremental block-based statistical timing analysis," in *Proceedings of the ACM/IEEE Design Automation Conference*, San Diego, California, USA, June 2004, pp. 331–336.

[27] J. Cong, "Challenges and opportunities for design innovations in nanometer technologies," (Available at: http://ballade.cs.ucla.edu/~cong/papers/final1.pdf), Dec. 1997, semiconductor Research Corporation Design Sciences Concept Paper.

[28] "Capo: A large-scale fixed-die placer from UCLA," Available at: http://vlsicad.ucsd.edu/GSRC/bookshelf/Slots/Placement.



Hongliang Chang received the B.E. degree from Qiqihar University, and the M. S. degree from University of Minnesota in 1996 and 2001 respectively. She is currently pursuing her Ph.D. at the Department of Computer Science and Engineering at the University of Minnesota.

Sachin S. Sapatnekar received the B.Tech. degree from the Indian Institute of Technology, Bombay in 1987, the M.S. degree from Syracuse University in 1989, and the Ph.D. degree from the University of Illinois at Urbana-Champaign in 1992. From 1992 to 1997, he was an assistant professor in the Department of Electrical and Computer Engineering at Iowa State University. He is currently the Robert and Marjorie Henle Professor in the Department of Electrical and Computer Engineering at the University of Minnesota.

He is an author of four books and a coeditor of one volume, and has published mostly in the areas of timing and layout. He has held positions on the editorial board of the IEEE Transactions on VLSI Systems, and the IEEE Transactions on Circuits and Systems II, the IEEE Transactions on CAD,and has been a Guest Editor for the latter. He has served on the Technical Program Committee for various conferences, and as Technical Program and General Chair for the Tau workshop and for the International Symposium on Physical Design. He has been a Distinguished Visitor for the IEEE Computer Society and a Distinguished Lecturer for the IEEE Circuits and Systems Society. He is a recipient of the NSF Career Award, three best paper awards at DAC and one at ICCD, and the SRC Technical Excellence award. He is a fellow of the IEEE.