Correlation-Preserved Statistical Timing with Quadratic Form of Gaussian Variables

Lizheng Zhang, *Student Member, IEEE,* Weijen Chen, *Student Member, IEEE,*
Yuhen Hu, *Fellow, IEEE,* John A. Gubner, *Member, IEEE,* Charlie, Chung-ping Chen, *Member, IEEE,*

Abstract

Recent study shows that the existing first order canonical timing model is not sufficient to represent the dependency of the gate/wire delay on the processing and operational variations when these variations become more and more significant. Due to the nonlinear mapping from variation sources to the gate/wire delay, the distribution of the delay will no longer be Gaussian even if variation sources are normally distributed.

A novel *quadratic timing model* is proposed to capture the non-linearity of the dependency of gate/wire delays and arrival times on the variation sources. Systematic methodology is also developed to evaluate the correlation and distribution of the quadratic timing model. Based on these, a statistical static timing analysis(SSTA) algorithm is proposed which retains the complete correlation information during timing analysis and has linear computation complexity with respect to both the circuit size and the number of variation sources.

Tested on the ISCAS circuits, the proposed algorithm shows significant accuracy improvement over the existing first order algorithm with small amount of computational cost.

I. Introduction

The timing performance of deep-submicron micro-architecture will be dominated by several factors. IC manufacturing process parameter variations will cause device and circuit parameters to deviate from their designed value. [1] Low supply voltage for low-power applications will reduce noise margin, causing increased timing delay variations. Due to dense integration and non-ideal on-chip power dissipation, rising temperature of substrate may lead to hot spot, causing excessive timing variations. Classical worst case timing analysis produces timing predictions that are often too pessimistic and grossly conservative. On the other hand, statistical static timing analysis (SSTA) that characterizes timing delays as statistical random variables offers a better approach for more accurate and realistic timing prediction.

Existing SSTA methods can be categorized into two distinct approaches: **path based SSTA** [2]–[5] and **block based SSTA** [6]–[12]. The path based approach seeks to estimate timing statistically on selected *critical paths*. However, the task of selecting a subset of paths whose time constraints are statistically critical has a worst-case complexity that grows exponentially with respect to the circuit size. Hence it is not easily scalable to handle realistic circuits. The block based approach, on the other hand, champions the notion of *progressive computation*. Specifically, each gate/wire is treated as a timing block and the timing analysis is performed block by block in the

circuit timing graph without looking back to the path history. As such, the computation complexity would grow linearly with respect to the circuit size.

However, to realize the full benefit of block based SSTA, one must address a challenging issue that gate/wire delays in a circuit could be correlated since two delays might be affected by the same source of *global variations* such as voltage supply uncertainties, gate channel length variations, wire geometry variations, etc. In [7], [8], [11] the delay D is explicitly related with these global variations G_i by the *canonical timing model*:

$$D = \mu + \alpha R + \sum_i \beta_i G_i \tag{1}$$

where R, called *local variation*, accounts the cumulative effect of all variation sources other than global variations.

The global variations are generally assumed to follow Gaussian distributions [1], [13]. The delay computed from the above canonical form will be Gaussian since it is a linear combination of Gaussian random variables. This may be acceptable for cases when the variation is small and the nonlinear relationship between the gate/wire delay and the global variation sources is not significant. However, when the variation becomes larger as technology scales down to nano-meter, the non-linearity of the gate/wire delay as a function of the global variations will be more and more significant and can not be accurately approximated by the *linear* canonical timing model. In these cases, even the global variations are modeled as Gaussian random variables, the gate/wire delays, in general, will not be Gaussian random variables.

To mitigate this deficiency, in this paper, we propose a *quadratic timing model* that augments the linear canonical timing model with second order terms:

$$D = m + \alpha R + \sum_i \beta_i G_i + \sum_{i,j} \Gamma_{ij} G_i G_j \tag{2}$$

where Γ_{ij} are quadratic coefficients and m is a constant term which may be different from the mean value of the delay timing variable.

Preliminary work reported in [14] indicated that a quadratic timing model delivered $4\times$ accuracy improvement over a first order canonical model. Nevertheless, [14] does not address the important question of *how to systematically develop a quadratic timing model to perform accurate SSTA of large scale circuits*. The main objective of this paper is to develop such a practical, efficient solution to this question. To this aim, we have made a number of tangible contributions:

- A novel quadratic timing model is formulated for both gate/wire delay and signal arrival time to represent the correlation between them. Systematic methodology is also developed to evaluate the correlations and to compute distributions for the quadratic timing model.
- To treat the important spatial correlation and avoid the possible error of existing *quad-tree* model proposed in [8], an analytical spatial correlation model is developed and the method controlling the computational complexity is introduced.
- A novel SSTA algorithm is developed based on the quadratic timing model which successfully retains the complete correlation information among arrival times during timing analysis while has the same computation

complexity as algorithms based on canonical timing model.

- To reduce the error introduced by forcing the nonlinear MAX/MIN to be approximated by linear operators, a tuple-based MAX/MIN evaluation method is proposed. Using a data structure, *max tuple*, the MAX/MIN evaluation is deferred when it is significantly nonlinear.

The rest of the paper is organized as following: Section II presents the quadratic timing model for gate/wire delay and the analytical spatial correlation model; Section III introduces the mathematical tools used for correlation and distribution evaluation for quadratic timing model; Section IV describes the SSTA algorithm based on the quadratic timing model and the tuple-based MAX/MIN evaluation method; Section V presents the C/C++ implementation and testing results; Section VII gives the conclusions.

II. Quadratic Model of Timing Variables

There are two ways to treat the case when timing variables of a circuit become non-Gaussian: (1) to directly find the distribution of the non-Gaussian timing variable; (2) to express the non-Gaussian timing variable as a non-linear function of Gaussian random variables. The first approach is straightforward but it is difficult to maintain the possible correlations among timing variables. With the second approach, the correlation among the non-Gaussian timing variables can be well expressed in terms of the correlations among the underneath Gaussian random variables.

A. Taylor Expansion of Delay Function

Since the gate/wire delay's dependency on the global variation sources is usually non-linear, Taylor expansion will be a common option to analyze such a nonlinear function systematically. Assuming $G_1, G_2, ..., G_p$ are p Gaussian random variables with zero mean and unit variations, the general Taylor expansion of the delay will be:

$$D(G_1, G_2, ..., G_p) = m + \alpha R + \sum_{j=1}^{\infty} \left\{ \frac{1}{j!} \left(\sum_{i=1}^{p} G_i \frac{\partial}{\partial G'_i} \right)^j D(G'_1, G'_2, ..., G'_p) \right\}_{G'_1 = G'_2 = ... = G'_p = 0} \quad (3)$$

where $m = D(0, 0, ..., 0)$ and R is the local variation with zero mean and unit variance. If the Taylor expansion is truncated at the first order, then we will get the well known canonical timing model and the m value will be the same as the mean value of the delay $m = \mu_D$. For higher-order truncations, $m \neq \mu_D$ in general. There will usually be many local variations that are not global variations but localized in the gate/wire and only affecting the gate/wire delay to which they belong. Since the local variation R represents the overall effect of all these localized variations, the assumption of R to be Gaussian is reasonable according to the *law of large number*.

Obviously, more timing accuracy can be got with more computation cost if the Taylor expansion is truncated at higher-order terms. So a reasonable trade-off has to be made between complexity and accuracy. While working on the real circuits, we found, for variation up to 10% of the nominal value, i.e. $\sigma/\mu \leq 10\%$, truncation at first-order terms is accurate enough. For cases where variation is larger than the 10% of the nominal value, significant error will be observed for the canonical timing model which means higher-order terms are needed for accurate timing. For variations up to 30%, experiments show that truncation at the second order terms will be sufficient to get

reasonable accuracy. Since it is quite rare to have higher variations than 30%, we will only discuss the timing model with those quadratic terms in the following sections.

B. Quadratic Gate Delay Model

The gate delay D_g is a nonlinear function of the global variations. Truncating equation (3) up to the second order, we formulate the quadratic gate delay model as:

$$D_g \approx m_g + \alpha R + \frac{\partial D_g}{\partial L}L + \frac{\partial D_g}{\partial V}V + ... + \frac{1}{2}\frac{\partial^2 D_g}{\partial L^2}L^2 + \frac{\partial^2 D_g}{\partial L \partial V}LV + \frac{1}{2}\frac{\partial^2 D_g}{\partial V^2}V^2 + ... \tag{4}$$

where m_g is a constant and $L, V...$ are global variations. The coefficients in this Taylor expansion can be analytically extracted from the Spice model of the gate delay using method of finite difference. More specifically, we make many designs of the gate, each of which has a different set of predefined parameters. By doing Spice simulation for all these gates, we will get a hyper-plane of the gate delay versus all parameters. The Taylor expansion coefficients can then be obtained by fitting the hyper-plane with the polynominal equation (4).

Assume that there are p global variation variables, one may define a $p \times 1$ Gaussian *variation vector*

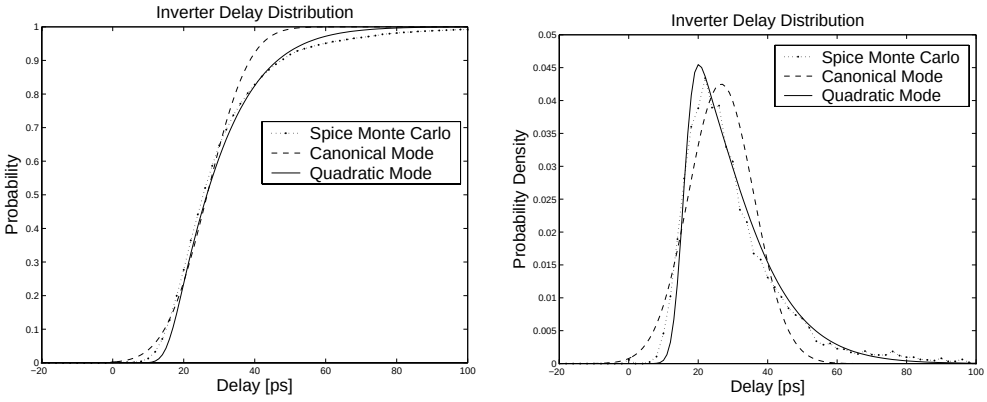
$$\boldsymbol{\delta}_g = [G_1, G_2, ..., G_p]^* \sim N(\mathbf{0}, \boldsymbol{\Sigma}_g)$$

where "*" represents the transpose operation. $\mathbf{0}$ is a zero vector. The *correlation matrix* ($\boldsymbol{\Sigma}_g = E\{\boldsymbol{\delta}_g \boldsymbol{\delta}_g^*\}$) is a $p \times p$ matrix. Generally it is not a unit matrix $\boldsymbol{I}$ since these global variations may be correlated. Consolidate equations (2) and (4) into a compact form as:

$$D_g = m_g + \alpha R + \boldsymbol{\beta}_g^* \boldsymbol{\delta}_g + \boldsymbol{\delta}_g^* \boldsymbol{\Gamma}_g \boldsymbol{\delta}_g \tag{5}$$

where the vector $\boldsymbol{\beta}_g$ and th matrix $\boldsymbol{\Gamma}_g$ are have elements as:

$$\beta_g(i) = \frac{\partial D_g}{\partial G_i} \quad and \quad \Gamma_g(i,j) = \frac{1}{2}\frac{\partial^2 D_g}{\partial G_i \partial G_j} \tag{6}$$



(a) $c.d.f.$ of Inverter Delay (b) $p.d.f.$ of Inverter Delay

Fig. 1: Distributions of Inverter Delay with parameter variations $\sigma/\mu = 30\%$

To demonstrate the advantage of the quadratic gate delay model over the first order canonical model, the probability distribution of an inverter delay is estimated using the Monte Carlo method where the timing delay of each trial is evaluated using SPICE circuit simulator. Using the quadratic form (5) extracted from the SPICE model, the delay distributions are also computed using both the quadratic timing model and the linear canonical timing model shown in figure 1. Results are compared to that obtained using the Monte Carlo simulation. Shown in figure 1(b), the "true" distribution from Monte Carlo simulation is significantly non-symmetric and non-Gaussian and can not be approximated by any canonical timing model. The quadratic model shows great accuracy improvement over the existing canonical model.

C. Quadratic Wire Delay Model

As shown in Figure 2, the distributive wire delay model separates a long wire into N equal segments with length of L. Wire segment i will have width of W_i and thickness of T_i. These width and thickness will then be considered as global variations with identical Gaussian distributions as $W_i \sim N(\mu_W, \sigma_W^2)$ and $T_i \sim N(\mu_T, \sigma_T^2)$.

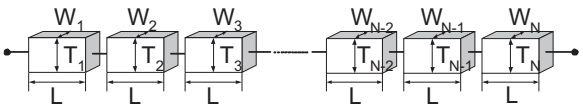


Fig. 2: Distributed Wire Delay Model

Elmore's wire delay model states that the total wire delay will be

$$D_w = \sum_{i=1}^{N}\sum_{j=i}^{N} R_i C_j = \sum_{i=1}^{N}\sum_{j=i}^{N} \frac{r_s L^2 (c_s W_j + c_f T_j)}{W_i T_i} \tag{7}$$

where R_i and C_j are the resistance and capacitance of the wire segment; r_s is the resistivity of the wire; c_s and c_f are the sheet and fringe unit capacity of the wire.

Applying the Taylor expansion to the Elmore's delay and truncating it until the second order, the quadratic wire delay model can be formulated similarly as that in the case of gate delay:

$$D_w \approx m_w + \alpha R + \boldsymbol{\beta}_w^* \boldsymbol{\delta}_w + \boldsymbol{\delta}_w^* \boldsymbol{\Gamma}_w \boldsymbol{\delta}_w \tag{8}$$

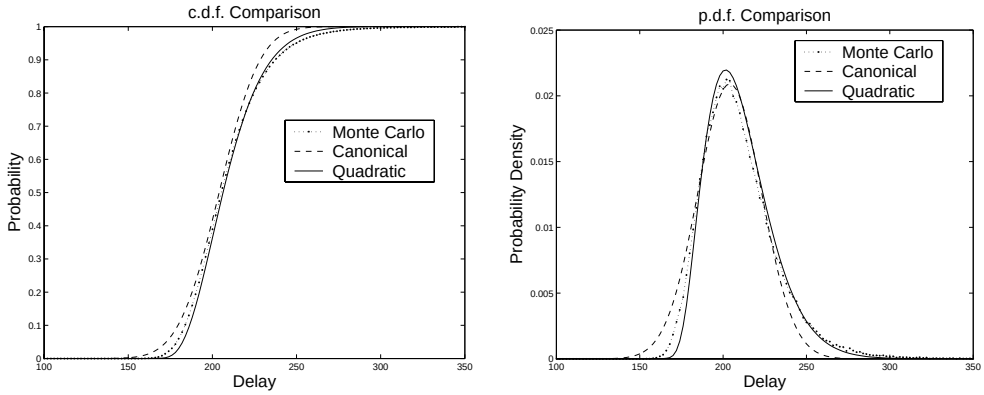
where $\boldsymbol{\delta}_w$ is a $2N \times 1$ global variation vector:

$$\boldsymbol{\delta}_w = [W_1', W_2', ..., W_N', T_1', T_2', ..., T_N'] \sim N(\mathbf{0}, \boldsymbol{\Sigma}_w)$$

with $W_i' = (W_i - \mu_W)/\sigma_W$ and $T_i' = (T_i - \mu_T)/\sigma_T$.

It is important to notice that width and thickness random variables are generally not statistically independent to each other since the wire usually spans a long distance and these random variables may be spatially correlated.

Due to the non-linear dependency of the wire delay on width and thickness variations shown in equation (7), the wire delay distribution will not be Gaussian even if the width and thickness are considered to be Gaussian.([1], [13]) This fact has been clearly shown in Figure 3 and the quadratic wire delay model again shows significant accuracy improvement over the existing canonical model when compared with the Monte Carlo simulation.



(a) $c.d.f.$ from Three Methods (b) $p.d.f.$ from Three Methods

Fig. 3: Wire Delay Distribution Comparison from Three Approaches

D. Analytical Model of Spatial Correlation

Global parameters affecting the gate delays, such as gate length L, voltage supply V, temperature T etc., are not independent to each other. Their variations might be spatially correlated, i.e., devices nearby will have similar global parameter variations. The fundamental property of the spatial correlation is that the statistical correlation between global parameters of gates at different positions will be a monotonic decreasing function with respect to the distance between the positions: the longer the distance, the smaller the correlation.

To model such kind of spatial correlation, the circuit chip area will be divided into grids and each grid cell will be assigned an individual global variation for the considered global parameter. All gates in a grid cell share the same global variation as assigned if the considered global parameter affects the gate delay. Different global parameter may be associated with different grids since they may have significantly different spatial correlation characteristics.

1) Error in Existing Quad-Tree Model: A so-called *quad-tree* model has been proposed in [8] to treat the spatial correlation within the SSTA framework. A hierarchical tree structure named quad-tree is built to connect the grid cells together and the correlations between the parameters of different grid cells are computed by counting the number of parent tree nodes they share. However, this model might cause significant error since there are always nodes which are spatially close to each other but belong to different subtrees in the quad-tree. The correlation between these nodes might be significantly underestimated by this model.

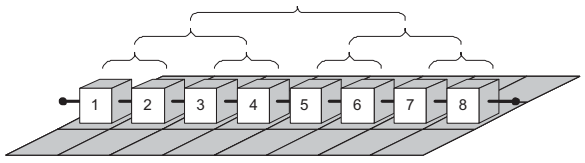
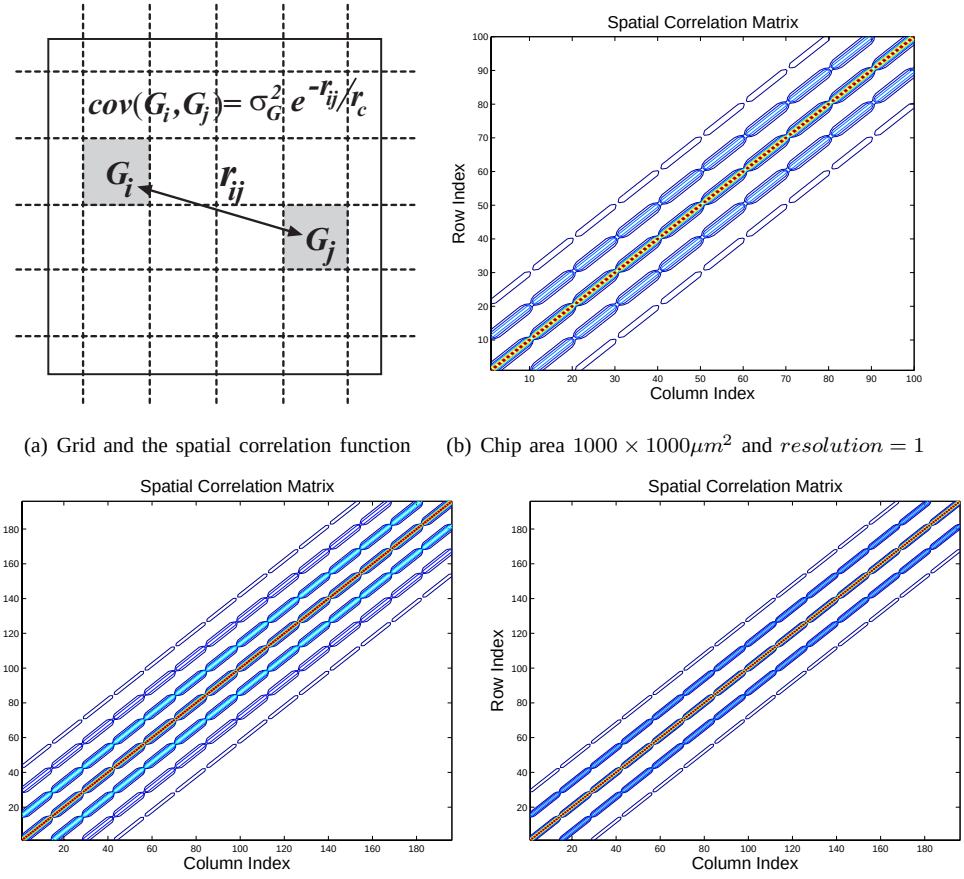


Fig. 4: Quad-tree model underestimates the spatial correlation between wire segments 4 and 5

For example, if the *quad-tree* method is used to model the spatial correlation in an eight-segment straight wire,

as illustrated in the figure 4, the quad-tree will becomes a binary tree if the quad-partitioning is along the wire. It is obvious that the correlation between wire segments 2 and 3, 4 and 5 will be similar to that between wire segments 1 and 2 since they are similarly spatially separated. But according to the quad-tree method, the spatial correlation between 1 and 2 will be the largest, that between 3 and 4 will be second and that between 4 and 5 will be the smallest. So the *quad-tree* model fails to give similar spatial correlation when distance is similar.



(a) Grid and the spatial correlation function (b) Chip area $1000 \times 1000\mu m^2$ and $resolution = 1$

(c) Chip area $1000 \times 1000\mu m^2$ and $resolution = 1.4$ (d) Chip area $1400 \times 1400\mu m^2$ and $resolution = 1$

Fig. 5: Exponential spatial correlation function and the spatial correlation matrix contour plots at the correlation distance of $r_c = 100\mu m$ and different chip area and resolutions.

2) Exponential Spatial Correlation: Our approach for spatial correlation is to assume the correlation follows an analytical function of the distance which fits the measurement data on manufactured chips. As an illustrative example, here it is assumed to be an exponential decaying function although the methodology is not restricted to such an exponential form.

For a global parameter G, as illustrated in figure 5(a), the covariance between the global variations at positions i and j will have the following expression:

$$cov(G_i, G_j) = \sigma_G^2 exp(-\frac{r_{ij}}{r_c}) \tag{9}$$

where σ_G^2 is the variance of the considered global parameter, r_{ij} is the distance between positions i and j; constant r_c is the characteristic *spatial correlation distance* of the considered global parameter. Obviously, the longer the r_c is, the stronger the spatial correlation.

3) Spatial Correlation Resolution: Intuitively, it is desired for high accuracy to have very fine grid. But fine grid will result in the large covariance matrix. It is then not beneficial since large covariance matrix will significantly degrade the performance of the timing analysis. The key strategy we proposed to make a good trade-off between accuracy and performance is to decide the grid size based on the spatial correlation distance of the considered global parameter through a user-defined parameter of *resolution*:

$$correlation\_distance = \boldsymbol{resolution} \times grid\_cell\_size$$

With such, fine grid is only applied when the correlation distance is short or high resolution is demanded.

Although it seems that fine grid is still inevitable since there is no guarantee that correlation distance is always large, performance will still be reasonable because the sparsity of the covariance matrix can be controlled by the $resolution$ parameter.

As shown in figure 5, the covariance matrix will always have a "band" structure where the number of bands in the matrix is decided by the user-defined parameter $resolution$. The higher the resolution, the more bands and the less sparsity of the matrix as shown in figures 5(b) and 5(c). With the same resolution, the number of bands will be the same as shown in figures 5(b) and 5(d) and the number of total significant elements in the covariance matrix is then proportional to the number of global variations, i.e. the dimension of the covariance matrix.

III. Correlations and Distributions for Quadratic Timing Model

During timing analysis for the given circuit, the signal arrival time at each net is the cumulative effect of all gate/wire delays at its input cone. If all gate/wire delays are expressed as the quadratic form and if only linear operations are involved during timing analysis, then the arrival time D_a will also have the quadratic form as:

$$D_a = m_a + \boldsymbol{\alpha}_a^* \boldsymbol{r}_a + \boldsymbol{\beta}_a^* \boldsymbol{\delta}_a + \boldsymbol{\delta}_a^* \boldsymbol{\Gamma}_a \boldsymbol{\delta}_a \tag{10}$$

assuming there are q gate/wire delays in the input cone of the net's arrival time D_a and p global variations are involved in these q gate/wire delays. The random variation vector $\boldsymbol{r}_a = [R_1, R_2, ..., R_q]^* \sim N(\mathbf{0}, \boldsymbol{I})$ is assumed to be independent on $\boldsymbol{\delta}_a = [G_1, G_2, ..., G_p]^* \sim N(\mathbf{0}, \boldsymbol{\Sigma}_a)$.

It is important to comment here that the above conclusion is based on that all operations during timing analysis are linear operations. This assumption is merely an approximation since there will always be non-linear operations of MAX/MIN involved to compute the arrival time.

Mathematically, the gate/wire delay quadratic equations (5) and (8) are only special cases of the arrival timing quadratic form (10), so

Theorem 1: *If every arrival time in a circuit is approximated as a linear combination of its input gate/wire delays, and all gate/wire delays have the quadratic delay format as equation* (5) *and* (8)*, then all timing variables*

in the circuit, including gate/wire delays and arrival times, will have the **quadratic timing model***:*

$$D \sim Q(m, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\Gamma}) = m + \boldsymbol{\alpha}^* \boldsymbol{r} + \boldsymbol{\beta}^* \boldsymbol{\delta} + \boldsymbol{\delta}^* \boldsymbol{\Gamma} \boldsymbol{\delta} \tag{11}$$

where $\boldsymbol{r} \sim N(\mathbf{0}, \boldsymbol{I})$ *and* $\boldsymbol{\delta} \sim N(\mathbf{0}, \boldsymbol{\Sigma})$ *are mutually independent local variations and global variations.*

A. Correlations between Quadratic Timing Variables

Both linear and quadratic dependencies are involved in the quadratic timing model (11). In order to evaluate the correlations between timing variables with quadratic forms, three types of correlation need to be computed: (1) correlation between linear terms; (2) correlations between linear and quadratic terms; (3) correlations between quadratic terms. The following theorem summarizes all these correlations:

Theorem 2: *For random vector* $\boldsymbol{\delta} \sim N(\mathbf{0}, \boldsymbol{\Sigma})$*, sensitivity vectors* $\boldsymbol{\alpha}$ *and* $\boldsymbol{\beta}$*, and quadratic coefficient matrices* $\boldsymbol{\Gamma}$, $\boldsymbol{\Gamma}_1$, $\boldsymbol{\Gamma}_2$

$$E\{\boldsymbol{\delta}^* \boldsymbol{\Gamma} \boldsymbol{\delta}\} = tr\{\boldsymbol{\Sigma} \boldsymbol{\Gamma}\} \tag{12}$$

$$cov(\boldsymbol{\alpha}^* \boldsymbol{\delta}, \boldsymbol{\beta}^* \boldsymbol{\delta}) = \boldsymbol{\alpha}^* \boldsymbol{\Sigma} \boldsymbol{\beta} \tag{13}$$

$$cov(\boldsymbol{\delta}^* \boldsymbol{\Gamma} \boldsymbol{\delta}, \boldsymbol{\beta}^* \boldsymbol{\delta}) = 0 \tag{14}$$

$$cov(\boldsymbol{\delta}^* \boldsymbol{\Gamma}_1 \boldsymbol{\delta}, \boldsymbol{\delta}^* \boldsymbol{\Gamma}_2 \boldsymbol{\delta}) = 2tr\{\boldsymbol{\Sigma} \boldsymbol{\Gamma}_1 \boldsymbol{\Sigma} \boldsymbol{\Gamma}_2\} \tag{15}$$

where "$tr\{\cdot\}$*" means "trace" and is the sum of the diagonal elements of the matrix.*

Proof: Equation (12) and (13):

$$E\{\boldsymbol{\delta}^* \boldsymbol{\Gamma} \boldsymbol{\delta}\} = E\{tr\{\boldsymbol{\Gamma} \boldsymbol{\delta} \boldsymbol{\delta}^*\}\} = tr\{\boldsymbol{\Gamma} E\{\boldsymbol{\delta} \boldsymbol{\delta}^*\}\} = tr\{\boldsymbol{\Gamma} \boldsymbol{\Sigma}\}$$

$$cov(\boldsymbol{\alpha}^* \boldsymbol{\delta}, \boldsymbol{\beta}^* \boldsymbol{\delta}) = E\{\boldsymbol{\alpha}^* \boldsymbol{\delta} \boldsymbol{\delta}^* \boldsymbol{\beta}\} = \boldsymbol{\alpha}^* E\{\boldsymbol{\delta} \boldsymbol{\delta}^*\} \boldsymbol{\beta} = \boldsymbol{\alpha}^* \boldsymbol{\Sigma} \boldsymbol{\beta}$$

For equation (14), after the vector format is expanded, $E\{\boldsymbol{\delta}^* \boldsymbol{\Gamma} \boldsymbol{\delta} \boldsymbol{\beta}^* \boldsymbol{\delta}\} = \sum_{i,j,k} c_{i,j,k} E\{G_i G_j G_k\} = 0$, since all summation terms will be moments with odd order and vanish given that $\boldsymbol{\delta} \sim N(\mathbf{0}, \boldsymbol{\Sigma})$. Since $E\{\boldsymbol{\beta}^* \boldsymbol{\delta}\} = 0$, then

$$cov(\boldsymbol{\delta}^* \boldsymbol{\Gamma} \boldsymbol{\delta}, \boldsymbol{\beta}^* \boldsymbol{\delta}) = E\{\boldsymbol{\delta}^* \boldsymbol{\Gamma} \boldsymbol{\delta} \boldsymbol{\beta}^* \boldsymbol{\delta}\} - E\{\boldsymbol{\delta}^* \boldsymbol{\Gamma} \boldsymbol{\delta}\} E\{\boldsymbol{\beta}^* \boldsymbol{\delta}\} = 0$$

For equation (15), with eigenvalue decomposition,

$$\boldsymbol{\delta}^* \boldsymbol{\Gamma} \boldsymbol{\delta} = \boldsymbol{u}^* \left(\boldsymbol{\Sigma}^{0.5}\right)^* \boldsymbol{\Gamma} \boldsymbol{\Sigma}^{0.5} \boldsymbol{u} = \boldsymbol{u}^* \boldsymbol{P}^* \boldsymbol{\Lambda} \boldsymbol{P} \boldsymbol{u}$$

where $\boldsymbol{u} \sim N(\mathbf{0}, \boldsymbol{I})$ is an independent random vector and $\boldsymbol{\Sigma}^{0.5*} \boldsymbol{\Gamma} \boldsymbol{\Sigma}^{0.5} = \boldsymbol{P}^* \boldsymbol{\Lambda} \boldsymbol{P}$ is another step of eigenvalue decomposition. It is easy to prove that random vector $\boldsymbol{x} = \boldsymbol{P} \boldsymbol{u} = [X_1, X_2, ...X_p]^*$ is also an independent random vector:

$$E\{\boldsymbol{x} \boldsymbol{x}^*\} = E\{\boldsymbol{P} \boldsymbol{u} \boldsymbol{u}^* \boldsymbol{P}^*\} = \boldsymbol{P} E\{\boldsymbol{u} \boldsymbol{u}^*\} \boldsymbol{P}^* = \boldsymbol{P} \boldsymbol{P}^* = \boldsymbol{I}$$

So $\boldsymbol{\delta}^* \boldsymbol{\Gamma} \boldsymbol{\delta} = \sum_i \lambda_i X_i^2$ where λ_is are diagonal elements in matrix $\boldsymbol{\Lambda}$. So the covariance

$$\begin{aligned} cov(\boldsymbol{\delta}^* \boldsymbol{\Gamma}_1 \boldsymbol{\delta}, \boldsymbol{\delta}^* \boldsymbol{\Gamma}_2 \boldsymbol{\delta}) &= \sum_{i,j} \lambda_{1i} \lambda_{2j} \left(E\{X_i^2 X_j^2\} - E\{X_i^2\} E\{X_j^2\}\right) \\ &= 2tr\{\boldsymbol{\Lambda}_1 \boldsymbol{\Lambda}_2\} = 2tr\{\boldsymbol{\Sigma} \boldsymbol{\Gamma}_1 \boldsymbol{\Sigma} \boldsymbol{\Gamma}_2\} \end{aligned}$$

where $E\{X_i^2X_j^2\} = E\{X_i^2\}E\{X_j^2\}$ when $i \neq j$ and when $i = j$, $E\{X_i^4\} = 3E^2\{X_i^2\}$. ■

Applying the above theorem, it is then easy to compute correlations between quadratic timing variables as:

Theorem 3: *For quadratic timing variable $D \sim Q(m, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\Gamma})$, its mean μ_D and variance σ_D^2 are*

$$\mu_D = E\{D\} = m + tr\{\boldsymbol{\Sigma\Gamma}\} \tag{16}$$

$$\sigma_D^2 = \boldsymbol{\alpha}^*\boldsymbol{\alpha} + \boldsymbol{\beta}^*\boldsymbol{\Sigma\beta} + 2tr\{\boldsymbol{\Sigma\Gamma\Sigma\Gamma}\} \tag{17}$$

and for quadratic random variables $D_1 \sim Q(m_1, \boldsymbol{\alpha}_1, \boldsymbol{\beta}_1, \boldsymbol{\Gamma}_1)$ and $D_2 \sim Q(m_2, \boldsymbol{\alpha}_2, \boldsymbol{\beta}_2, \boldsymbol{\Gamma}_2)$, the correlation between them is:

$$cov(D_1, D_2) = \boldsymbol{\alpha}_1^*\boldsymbol{\alpha}_2 + \boldsymbol{\beta}_1^*\boldsymbol{\Sigma\beta}_2 + 2tr\{\boldsymbol{\Sigma\Gamma}_1\boldsymbol{\Sigma\Gamma}_2\} \tag{18}$$

B. Distributions of Quadratic Timing Variables

To compute the distribution of the quadratic timing variable D defined in equation (11), we use a statistics technique called *characteristic function*:

Definition 1: *For random variable X, its characteristic function is defined as:*

$$C_X(\xi) = E\{e^{j\xi X}\} = \int_{-\infty}^{+\infty} e^{j\xi x} f_X(x)dx \tag{19}$$

where $f_X(x)$ is the $p.d.f.$ of X.

Since the characteristic function is actually an inverse fourier transform of the the $p.d.f.$, the $p.d.f.$ of the random variable can easily computed from its characteristic function:

Theorem 4: *If the random variable X has a characteristic function of $C_X(\xi)$, then the $p.d.f.$ of the random variable X will be:*

$$f_X(x) = \frac{1}{2\pi}\int_{-\infty}^{+\infty} e^{-j\xi x} C_X(\xi)d\xi \tag{20}$$

The formal proof of this theorem can be found in textbooks of probabilistic theory such as [15].

For the quadratic timing variable $D \sim Q(m, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\Gamma})$ defined in equation (11), its *exact* characteristic function can be analytically derived by substituting it into equation (19):

$$\begin{aligned} C_D(\xi) &= E\{e^{j\xi D}\} = \iint_{-\infty}^{+\infty} e^{j\xi(m+\boldsymbol{\alpha}^*\boldsymbol{r}+\boldsymbol{\beta}^*\boldsymbol{\delta}+\boldsymbol{\delta}^*\boldsymbol{\Gamma\delta})} f_R(\boldsymbol{r}) f_G(\boldsymbol{\delta}) d\boldsymbol{r} d\boldsymbol{\delta} \\ &= |\boldsymbol{\Omega}|^{-\frac{1}{2}} exp\{j\xi m - \frac{1}{2}\xi^2(\boldsymbol{\alpha}^*\boldsymbol{\alpha} + \boldsymbol{\beta}^*\boldsymbol{\Sigma}^{\frac{1}{2}}\boldsymbol{\Omega}^{-1}\boldsymbol{\Sigma}^{\frac{1}{2}}\boldsymbol{\beta})\} \end{aligned} \tag{21}$$

where $f_R(\boldsymbol{r})$ and $f_G(\boldsymbol{\delta})$ are joint $p.d.f.$s for Gaussian random vectors $\boldsymbol{r}$ and $\boldsymbol{\delta}$ respectively; $|\boldsymbol{\Omega}|$ is the determinant of matrix $\boldsymbol{\Omega} = \boldsymbol{I} - 2j\xi\boldsymbol{\Sigma}^{\frac{1}{2}}\boldsymbol{\Gamma}\boldsymbol{\Sigma}^{\frac{1}{2}}$. So the $p.d.f.$ of the quadratic time variable, $f_D(x)$, can then be computed from theorem 4.

Clearly, there will be one step of eigenvalue decomposition (computing $\boldsymbol{\Sigma}^{\frac{1}{2}}$) and one step of fourier transformation in order to analytically compute the distribution of a quadratic timing variable. This means the distribution computation may be computation intensive. Fortunately, it is not required to compute the distribution at every step

of timing analysis. Distribution will usually only be requested once at the primary output. So this possible intensive computation is still durable and is not dependent on the size of the circuit.

IV. SSTA WITH QUADRATIC TIMING MODEL

In block based timing analysis, the arrival time random variable propagation involves two elemental operations:

- ADD: When an input arrival time X propagates through a gate delay Y, the output arrival time will be $Z = X + Y$
- MAX: When two arrival times X and Y merge in a gate, a new arrival time of $Z = max(X, Y)$ will be formulated before the gate delay is added.

Before analysis, the quadratic parameters of individual gate/wires are extracted from their Spice models and a gate/wire library is then formed. Using the gate/wire delay information in the library, the circuit being analyzed will then be translated into a timing graph which is represented by a file called *standard delay variance correlation format(sdvcf)* where both quadratic gate/wire delays and the gate/wire connections are specified. The overall data flow of the algorithm is summarized in figure 6.

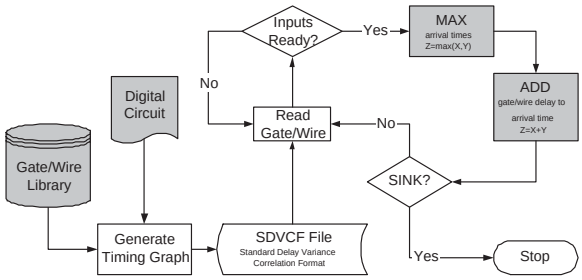


Fig. 6: Block-Based Algorithm with Quadratic Timing Model

A. ADD Operation

If both X and Y are expressed in the quadratic form of (11) $X \sim Q(m_X, \boldsymbol{\alpha}_X, \boldsymbol{\beta}_X, \boldsymbol{\Gamma}_X)$ and $Y \sim Q(m_Y, \boldsymbol{\alpha}_Y, \boldsymbol{\beta}_Y, \boldsymbol{\Gamma}_Y)$, then the output of the ADD operator is very straightforward as:

$$Z = X + Y \sim Q(m_Z, \boldsymbol{\alpha}_Z, \boldsymbol{\beta}_Z, \boldsymbol{\Gamma}_Z)$$

where the quadratic parameters are computed as:

$$\begin{aligned} m_Z &= m_X + m_Y \quad ; \quad \boldsymbol{\alpha}_Z = \boldsymbol{\alpha}_X + \boldsymbol{\alpha}_Y \\ \boldsymbol{\beta}_Z &= \boldsymbol{\beta}_X + \boldsymbol{\beta}_Y \quad ; \quad \boldsymbol{\Gamma}_Z = \boldsymbol{\Gamma}_X + \boldsymbol{\Gamma}_Y \end{aligned} \tag{22}$$

B. Linear Approximation of MAX Operation

MAX operator, however, is more complicated since it is generally a non-linear operator and error will happen if we approximate it with a linear operator. In the cases when MAX behaves linear, however, it is good to compute the MAX output with an equivalent linear operator for the sake of the simplicity.

One reasonable error function for MAX approximation is:

Definition 2: *If function $\hat{Z} = \psi(X, Y)$ is an approximation of $Z = max(X, Y)$ with random variables X and Y, then the error function is defined as:*

$$\begin{aligned} \Delta &= \iint_{-\infty}^{+\infty} \left[max(x, y) - \psi(x, y)\right]^2 f_{XY}(x, y)dxdy \\ &= E\{Z^2\} + E\{\hat{Z}^2\} - 2E\{Z\hat{Z}\} \end{aligned} \tag{23}$$

where $f_{XY}(x, y)$ is the joint distribution of X and Y.

If function $\hat{Z} = \psi(X, Y)$ includes parameters of $\lambda_1, \lambda_2, ...$, the optimum value of λ_i will then be determined by the following *optimal condition*:

$$E\{\hat{Z}\frac{\partial \hat{Z}}{\partial \lambda_i}\} = E\{Z\frac{\partial \hat{Z}}{\partial \lambda_i}\} \qquad (i = 1, 2, ...) \tag{24}$$

to minimize approximation error function Δ defined in (23).

Apply such optimal condition to the case of linear approximation, we will get the following theorem:

Theorem 5: *If $\hat{Z} = aX + bY + c$ is an approximation function of $Z = max(X, Y)$, the optimal values of a, b and c which minimize the error function defined in equation* (23) *will be the solution of equations:*

$$\begin{cases} E\{Z\} &= a\ \mu_X + b\ \mu_Y + c \\ cov(Z, X) &= a \cdot \sigma_X^2 + b \cdot cov(X, Y) \\ cov(Z, Y) &= a \cdot cov(X, Y) + b \cdot \sigma_Y^2 \end{cases} \tag{25}$$

Given quadratic timing variables of $X \sim Q(m_X, \boldsymbol{\alpha}_X, \boldsymbol{\beta}_X, \boldsymbol{\Gamma}_X)$, and $Y \sim Q(m_Y, \boldsymbol{\alpha}_Y, \boldsymbol{\beta}_Y, \boldsymbol{\Gamma}_Y)$, $E\{Z\}$, $cov(X, Z)$ and $cov(Y, Z)$ can be analytically computed using equations in [16] by additionally accepting some accuracy penalty and treating $Z = max(X, Y)$ as if MAX is operating on Gaussian random variables. So the linear approximation parameters, a, b and c, can then be solved from theorem 5 as:

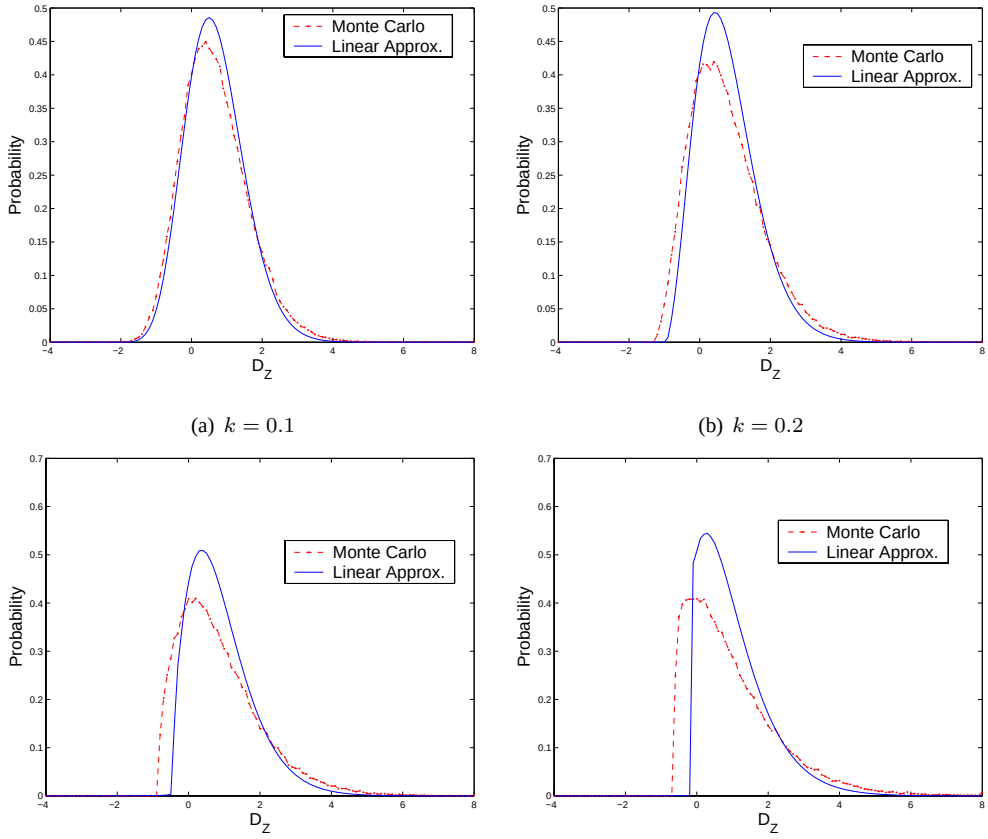
$$a = \Phi \quad b = 1 - \Phi \quad c = \varphi\sigma_{X-Y} \tag{26}$$

where Φ and φ are $c.d.f.$ and $p.d.f.$ of standard Gaussian distribution evaluated at μ_{X-Y}/σ_{X-Y}.

With such linear approximation of MAX operator, the quadratic timing variable $Z = max(X, Y) \sim Q(m_Z, \boldsymbol{\alpha}_Z, \boldsymbol{\beta}_Z, \boldsymbol{\Gamma}_Z)$ can then be easily computed as

$$\begin{aligned} \boldsymbol{\alpha}_Z = a\boldsymbol{\alpha}_X + b\boldsymbol{\alpha}_Y \quad ; \quad m_Z = am_X + bm_Y + c \\ \boldsymbol{\beta}_Z = a\boldsymbol{\beta}_X + b\boldsymbol{\beta}_Y \quad ; \quad \boldsymbol{\Gamma}_Z = a\boldsymbol{\Gamma}_X + b\boldsymbol{\Gamma}_Y \end{aligned} \tag{27}$$

There are two sources of error for such kind of linear approximation for MAX operation. The first one is that we assume MAX inputs are Gaussian when we compute the linear mixing coefficients. The second one come from the non-linearity of the MAX operator.



(a) $k = 0.1$ (b) $k = 0.2$

(c) $k = 0.3$ (d) $k = 0.4$

Fig. 7: $p.d.f.$ comparison between Monte Carlo and our linear approximation for $D_Z = max(X + kX^2, Y + kY^2)$ at different quadratic coefficients k where X and Y are independent standard Gaussian random variables

MAX inputs are expressed as quadratic format during timing analysis and are fundamentally non-Gaussian random variables. But in our linear approximation, we compute those linear combination coefficients as if MAX inputs were Gaussian random variables. Such approximation will inevitably involve some error. But we argue here that the error is tolerable due to the fact that the non-Gaussianity of MAX inputs is not severe in practical timing analysis.

The non-Gaussianity of the MAX inputs are actually decided by the relative magnitude of the quadratic coefficients versus the first order coefficients if timing variables are expressed as quadratic timing model as (11). For example shown in figure 7, we can compute the MAX result of two quadratic timing variables $D_X = X + kX^2$ and $D_Y = Y + kY^2$ as $D_Z = max(D_X, D_Y)$ using both Monte Carlo simulation and our linear approximation at different quadratic coefficients $k = 0.1, 0.2, 0.3, 0.4$. From figure 7, it is clear that the larger the quadratic coefficient is, the worse our linear approximation. However, even for large quadratic coefficient as $k = 0.4$, the

linear approximation still performs well. Moreover, the error introduced by the non-Gaussianity of MAX inputs will only make the linear approximation give more pessimistic estimation than the Monte Carlo result. In another word, our linear approximation is safe even if it will have some errors when non-Gaussianity of the input timing variables is very large.

The second error source for our linear approximation is that MAX operator is fundamentally a non-linear operator. But for the purpose of timing analysis, it mostly behaves like a linear operator and can be well approximated by a linear operator shown above.

Of course, there will be some very "bad" case happening during timing analysis where MAX operator is very non-linear. If MAX is forced to be approximated by the above linear operator, error could possibly accumulate to be out of control. As we notice that these non-linear cases of MAX operator happen with relatively small amount, we propose to use the method of *Max Tuple* as described in the following. Basically, we detects when MAX becomes non-linear and defer the MAX evaluation using a data structure of max tuple.

C. Conditional Linear MAX Approximation

For the purpose of timing analysis, it is not always necessary to explicitly compute the MAX output at every step.

If during a propagation step of MAX, $max(X, Y)$, the MAX operation is determined to be significantly non-linear, no actual computation will be done and the output of the MAX will be simply recorded as a data structure of *max tuple*: $Mt\{X, Y\}$. With such max tuples, the arrival time propagation will have the following computations:

- ADD: a gate/wire delay, D, is added into a max tuple $Mt\{X, Y\}$ as:

$$Mt\{X, Y\} + D = Mt\{X + D, Y + D\}$$

- aMAX: an arrival time, A, is MAXed with a max tuple $Mt\{X, Y\}$ as:

$$max(A, Mt\{X, Y\}) = Mt\{A, X, Y\}$$

- tMAX: two max tuples are MAXed together:

$$max(Mt\{X, Y\}, Mt\{U, V\}) = Mt\{X, Y, U, V\}$$

To practically implement such tuple-based MAX evaluation, the number of arrival times in the max tuple, i.e. the tuple size, has to be maintained as small as possible. This is realized by the obvious associative rule of max tuple as:

$$Mt\{A, X, Y\} = Mt\{max(A, X), Y\} = Mt\{A, max(X, Y)\} = Mt\{X, max(A, Y)\}$$

so if a MAX operator on any two random variables in the max tuple behaves linear, then the two random variables will be replaced by their linear combination as shown in equation (27) and so that the size of the max tuple is reduced. This reduction process will be done iteratively to minimize the tuple size.

To realize such tuple-based MAX evaluation, it is necessary to establish a method to determine the non-linearity of the MAX operator analytically. Although skewness is not a Gaussianity index for a general random variable since there are distributions which are symmetric but non-Gaussian. However, to measure the linearity of the MAX operator with Gaussian inputs, skewness is a good choice due to the fact that the non-linearity of the MAX operator will always change the symmetry of the distribution of the MAX output. However, there is no simple analytical equations to compute the skewness of the MAX output if its inputs are with quadratic forms. For the purpose to decide the non-linearity of a MAX operator, we again assume that the MAX inputs are Gaussian and a parameter of *Gaussian-Input Skewness(GIS)* can then be computed to decide the linearity of the MAX operator.

So the tuple size reduction can be realized by associating each max tuple with a matrix which stores the GIS of pairs of random variables in the tuple. And also a threshold of skewness is set to indicate if the MAX operation is non-linear. Finally, to prevent the explosion of the tuple size, a safe-guard maximum allowed size for max tuple is also set and if any of the tuple size exceeds the maximum size, the skewness threshold will be increased to activate more tuple size reduction.

Finally, in the primary output of the circuit, the circuit delay is reported as max tuple which can be easily evaluated by Monte Carlo simulation to get the requested $p.d.f.$ and/or $c.d.f.$. For limited size of max tuple, such evaluation is efficient and accurate.

D. Extra Computation Complexity

When compared with the SSTA algorithm based on first order canonical timing model, the extra computation complexity of the methods based on quadratic timing model will come from two parts:(1)compute the moment of quadratic form using equations (16) to (18); (2)update the quadratic coefficient matrix $\mathbf{\Gamma}$ using equations (22) and (27).

Assuming for a circuit with N gates, there are t types of global variation sources. And there are totally q global variation variables considering spatial correlation. Both covariance matrix $\mathbf{\Sigma}$ and quadratic coefficient matrix $\mathbf{\Gamma}$ are sparse matrix. The number of non-zero elements in $\mathbf{\Gamma}$will be $O(t^2)$. Matrix $\mathbf{\Sigma}$ will have a "band" structure as shown in figure 5 and will have $O(q)$ significant non-zero elements. Since the types of global variation sources are usually very limited for a particular technology, $q >> t$, the additional computation needed to switching from first order timing model to quadratic timing model will be mainly contributed from the moment evaluation equations from (16) to (18).

Again, in equations (16) to (18), the trace evaluation of the product of two matrix can be done in linear time, $O(q)$ considering the "band" structure of the covariance matrix $\mathbf{\Sigma}$. Since such moment evaluation has to be done at every timing step, so the overall additional timing complexity required will be $O(q \times N)$ which is linear to both the number of global variations q and the size of the circuit N.

Since the number of bands in $\mathbf{\Sigma}$ is controlled by the user-defined parameter of resolution, the higher the resolution, the more bands in the matrix and the longer the variance/covariance computation time. On the other hand, the higher the resolution, the finer the grid, the more accurate of the correlation model. So the user-defined parameter

$resolution$ provides a good way to trade off accuracy and complexity in considering spatial correlation for statistical timing.

E. Application in Path Based SSTA

Although we propose above a block based SSTA method because path based SSTA will have potential difficulty to select statistically critical paths in complex circuits, nothing prevents us applying the proposed quadratic timing model in path based SSTA.

As long as the statistically critical paths are correctly selected, the overall delay distribution of the circuit can be computed straightforwardly. For the i^{th} critical path cpi, its path delay will be the sum of all gate/wire delays in the path: $D_{cpi} = \sum_{g \in cpi} D_g$. When all gate/wire delays are quadratically represented, the path delay will also have quadratic format as:

$$D_{cpi} \sim Q(\sum_{g \in cpi} m_g, \sum_{g \in cpi} \boldsymbol{\alpha}_g, \sum_{g \in cpi} \boldsymbol{\beta}_g, \sum_{g \in cpi} \boldsymbol{\Gamma}_g) \tag{28}$$

So if there are n statistically critical paths, the overall delay distribution will be:

$$D_{all} = max(D_{cp1}, D_{cp2}, ..., D_{cpn}) \tag{29}$$

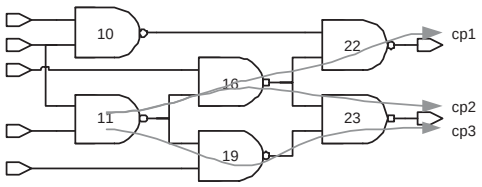


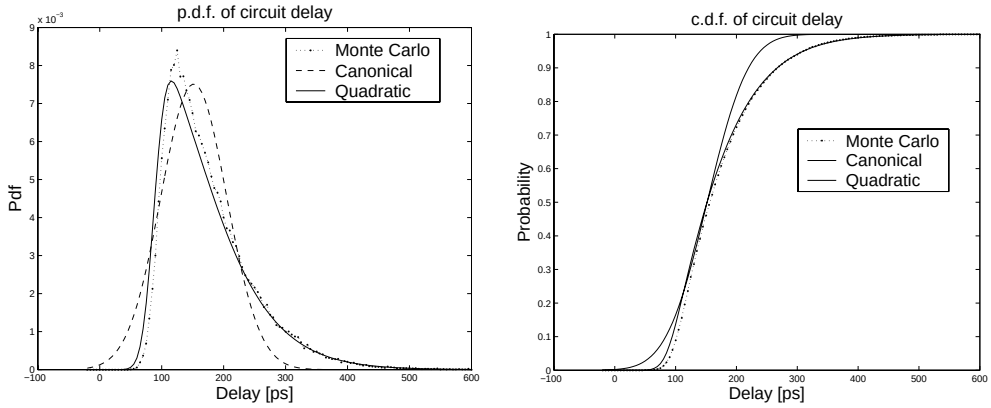
Fig. 8: Example Circuit for Path-Based Timing Analysis

For example, the statistically critical paths for the circuit shown in figure 8 will be: $cp1 : (11, 16, 22)$; $cp2 : (11, 16, 23)$; $cp3 : (11, 19, 23)$. So the overall delay distribution will be: $D_{all} = max(D_{cp1}, D_{cp2}, D_{cp3})$ where each path delay can be computed from equation (28). The computed overall delay distribution for the above circuit is graphically shown in figure 9. When compared with Monte Carlo results, by using quadratic model, significant accuracy improvement is clearly shown in both $c.d.f$ and $p.d.f$..

V. SIMULATIONS AND DISCUSSIONS

The proposed block based SSTA with quadratic timing model has been implemented in C/C++ with the name of *QuadStat* and tested on the ISCAS'85 benchmark circuits. For comparison, we also implement the SSTA based on first order canonical timing model, named *CanoStat*, is also implemented and tested. Monte Carlo simulation with 10,000 repetitions is used as a comparison standard.

Before experiment, a simple standard gate library with gates of *not, nand2, nand3, nor2, nor3, aoi22 and oai22* are constructed. The deterministic delay and statistical delay sensitivities of these library cells are extracted from



(a) $p.d.f.$ (b) $c.d.f.$

Fig. 9: Distribution Comparison of Path Based Timing Analysis: (1)Monte Carlo; (2)Canonical Model Based; (3)Quadratic Model Based

their Spice model as described in section II-B. These library cells are also laid out using Cadence® tools and their physical sizes are measured from the layout. With such a simplified library, all ISCAS circuits are firstly synthesized by Synopsys® design compiler. To obtain the placement information needed for spatial correlation, all circuits are placed with a public placement tool, Dragon®.

All parameters are assumed to have $\sigma/\mu = 30\%$ of variations. Three parameters, gate length(L), supply voltage(V) and temperature(T) are considered to be global variation sources and their correlation distance are all assumed to be $100\mu m$ for illustrative purpose. The spatial correlation resolution is set to be 3 for all three global parameters so that the size of the grid cells covering the circuit will be $33\mu m$. All other variation sources specified in the technology file are assumed to be localized and their effect on the gate delay is lumped into a single term of local variation.

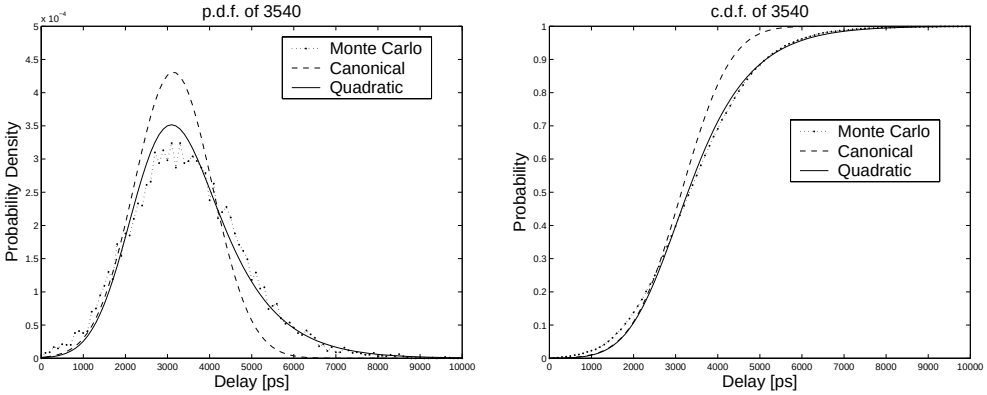
Timing results from both *QuadStat* and *CanoStat* are shown in Table I and compared with that from Monte Carlo simulation. Since the time variables, either gate/wire delays or arrival times, are modeled as non-Gaussian random variables, the mean(μ) and std(σ), used for canonical delay cases, are not sufficient to characterize the distributions of the time random variables. So we also show the 97.7% quantile of the output arrival time distribution, τ_{97}, in the table I.

The estimation error is also shown in the table from which it is clear that there is a significant accuracy improvement just by switching the delay model from canonical to quadratic. Measured by the critical parameter of the 97.7% quantile of the circuit's delay distribution, significant accuracy improvement is achieved: the average error of CanoStat is 24% while that of QuadStat is as small as 2.3%.

To graphically illustrate the accuracy improvement, the delay distributions of ISCAS circuit c3540 are show in figure 10. It is clear that the accuracy improvement of the QuadStat is mostly due to the high probability region of the distribution which is actually more critical for circuit performance. CanoStat will clearly underestimate the

Circuit	mean: μ ($\Delta\mu$)			std: σ ($\Delta\sigma$)			97.7% quantile: τ_{97}($\Delta\tau_{97}$)		
	M.C.	CanoStat	QuadStat	M.C.	CanoStat	QuadStat	M.C.	CanoStat	QuadStat
C432	1828	1653(9.5%)	1853(1.4%)	779	537 (31%)	734 (5.8%)	3490	2650(24%)	3450(1.2%)
C880	1843	1671(9.4%)	1867(1.3%)	753	448 (40%)	689 (8.5%)	3440	2500(27%)	3360(2.3%)
C1355	1811	1636(9.7%)	1828(0.9%)	746	485 (35%)	697 (6.6%)	3280	2530(23%)	3360(2.4%)
C1908	2437	2190(10%)	2432(0.2%)	914	695 (24%)	880 (3.8%)	4410	3490(20%)	4370(0.9%)
C2670	2666	2404(10%)	2738(2.7%)	1019	618 (39%)	860 (16%)	4840	3560(26%)	4620(4.6%)
C3540	3468	3136(10%)	3499(0.9%)	1344	936 (30%)	1309(2.6%)	6230	4870(22%)	6370(2.3%)
C6288	8798	7950(10%)	9393(6.8%)	3785	2661 (30%)	3535(6.6%)	16799	12919(23%)	16639(1.0%)
C7552	2440	2202(10%)	2489(2.0%)	981	599 (39%)	828 (15%)	4510	3310(27%)	4270(5.3%)
Average Error	-	9.7%	2%	-	34%	8.1%	-	24%	2.3%

TABLE I: Distribution Parameters for ISCAS Circuits with three Approaches: (1)Monte Carlo(M.C.); (2)Canonical Model(CanoStat); (3)Quadratic Model(QuadStat). Errors in parenthesis for CanoStat and QuadStat are computed using M.C. as standard.



(a) $p.d.f.$ of ISCAS c3540 (b) $c.d.f.$ of ISCAS c3540

Fig. 10: Distribution Comparison of ISCAS c3540 from Three Approaches: (1)Monte Carlo; (2)Canonical Model; (3)Quadratic Model

delay in the high probability region. This underestimation, in reality, will result in optimistic design and excessive chip failure. This example clearly shows the necessary to use quadratic timing model when variations become large in nowadays technology.

The CPU time of the three approaches is shown in Table II. Although several times of CPU penalty is observed between QuadStat and CanoStat, the overall run time is still very small for circuit with thousands of gates. So by using our quadratic model for timing, we achieved significant accuracy improvement by relatively small run time penalty.

Also, the size of the MAX tuple at the sink node and the average tuple size of the circuit are also listed. The small tuple size shows that the MAX operation, in most time during circuit timing, behaves linearly and be safely approximated by a linear operator.

Circuit	C432	C880	C1355	C1908	C2670	C3540	C6288	C7552
Gate Counts	280	641	717	1188	2004	2485	2704	5355
Grid Cells	3x3	5x3	5x4	5x4	6x5	7x6	12x10	10x8
QuadStat	0.36s	0.52s	0.81s	0.63s	0.83s	1.32s	7.23s	4.32s
CanoStat	0.07s	0.18s	0.25s	0.18s	0.23s	0.61s	4.20s	1.98s
CPU Penalty	5.1x	2.9x	3.2x	3.5x	3.6x	2.2x	1.7x	2.2x
Sink's TupleSize	1	8	1	2	2	4	3	4
Average TupleSize	1.2	1.2	1.5	1.3	1.2	1.3	1.1	1.3

TABLE II: Sink node's max tuple size, average tuple size and CPU time of CanoStat and QuadStat

VI. Acknowledgment

This work was partially funded by TSMC, UMC, Faraday, SpringSoft, National Science Foundation under grants CCR-0093309 & CCR-0204468 and National Science Council of Taiwan, R.O.C. under grant NSC 92-2218-E-002-030. Also great thanks to professor Barry D. Van Veen for the great discussions.

VII. Conclusions

A novel *quadratic timing model* is defined for time variables in SSTA and its advantages over the existing *canonical timing model* are demonstrated for both gate and wire delays.

Based on this quadratic timing model, the correlations and distribution between those non-Gaussian time variables can be elegantly evaluated. Furthermore, a novel block based SSTA algorithm is formulated using the quadratic timing model. Testing results on the benchmark circuits show that the new algorithm can significantly improve the timing accuracy without degrading performances. Finally, the advantage of using quadratic timing model in path based timing analysis is also demonstrated by a simple example.

References

[1] S. Nassif, "Within-chip variability analysis," *Electron Devices Meeting, 1998. IEDM '98 Technical Digest., International*, pp. 283 – 286, Dec 1998.

[2] J.-J. Liou, A. Krstic, L.-C. Wang, and K.-T. Cheng, "False-path-aware statistical timing analysis and efficient path selection for delay testing and timing validation," *Design Automation Conference, 2002. Proceedings. 39th*, pp. 566 – 569, June 2002.

[3] M. Orshansky, "Fast computation of circuit delay probability distribution for timing graphs with arbitary node correlation," *TAU'04*, Feb 2004.

[4] M. Orshansky and K. Keutzer, "A general probabilistic framework for worst case timing analysis," *Design Automation Conference, 2002. Proceedings. 39th*, pp. 556 – 561, June 2002.

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

[6] A. Agarwal, V. Zolotov, and D. Blaauw, "Statistical timing analysis using bounds and selective enumeration," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 22, no. 9, pp. 1243 –1260, Sept 2003.

[7] C. Visweswariah, K. Ravindran, and K. Kalafala, "First-order parameterized block-based statistical timing analysis," *TAU'04*, Feb 2004.

[8] A. Agarwal, D. Blaauw, and V. Zolotov, "Statistical timing analysis for intra-die process variations with spatial correlations," *Computer Aided Design, 2003 International Conference on. ICCAD-2003*, pp. 900 – 907, Nov 2003.

[9] S. Bhardwaj, S. B. Vrudhula, and D. Blaauw, "τau: Timing analysis under uncertainty," *ICCAD'03*, pp. 615–620, Nov 2003.

[10] A. Devgan and C. Kashyap, "Block-based static timing analysis with uncertainty," *ICCAD'03*, pp. 607–614, Nov 2003.

[11] H. Chang and S. S. Sapatnekar, "Statistical timing analysis considering spatial correlations using a single pert-like traversal," *ICCAD'03*, pp. 621–625, Nov 2003.

[12] S. Tsukiyama, M. Tanaka, and M. Fukui, "A statistical static timing analysis considering correlations between delays," *Proceedings of the 2001 conference on Asia South Pacific design automation*, January 2001.

[13] S. R. Nassif, "Modeling and analysis of manufacturing variations," *CICC*, pp. 223–228, 2001.

[14] X. Li, J. Le, P. Gopalakrishnan, and L. T. Pileggi, "Asymptotic probability extraction for non-normal distributions of circuit," *ICCAD'04*, pp. 2–9, Nov 2004.

[15] B. V. Gnedenko, *Theory of Probability*, 6th ed. Gordon and Breach Science Publishers, 1997, translated from Russian by Igor A. Ushakov.

[16] C. Clark, "The greatest of a finite set of random variables," *Operations Research*, pp. 145–162, March 1961.