Microelectronics Journal 41 (2010) 99–108

Contents lists available at ScienceDirect

Microelectronics Journal

journal homepage: www.elsevier.com/locate/mejo

Process variation-aware performance analysis of asynchronous circuits

Mohsen Raji, Behnam Ghavami *, Hossein Pedram, Hamid R. Zarandi

Department of Computer Engineering and Information Technology, Amirkabir University of Technology (Tehran Polytechnic), Tehran, Iran

ARTICLE INFO

Article history:
Received 17 October 2009
Received in revised form
21 December 2009
Accepted 22 December 2009
Available online 18 January 2010

Keywords:
Asynchronous circuits
Process variation
Spatial correlation
Timed Petri-Net

ABSTRACT

Current technology trends have led to the growing impact of process variations on performance of asynchronous circuits. As it is imperative to model process parameter variations for sub-100nm technologies to produce a more real performance metric, it is equally important to consider the correlation of these variations to increase the accuracy of the performance computation. In this paper, we present an efficient method for performance evaluation of asynchronous circuits considering inter- and intra-die process variation. The proposed method includes both statistical static timing analysis (SSTA) and statistical Timed Petri-Net based simulation. Template-based asynchronous circuit has been modeled using Variant-Timed Petri-Net. Based on this model, the proposed SSTA calculates the probability density function of the delay of global critical cycle. The efficiency for the proposed SSTA is obtained from a technique that is derived from the principal component analysis (PCA) method. This technique simplifies the computation of mean, variance and covariance values of a set of correlated random variables. In order to consider spatial correlation in the Petri-Net based simulation, we also include a correlation coefficient to the proposed Variant-Timed Petri-Net which is obtained from partitioning the circuit. We also present a simulation tool of Variant-Timed Petri-Net and the results of the experiments are compared with Monte Carlo simulation-based method.

© 2009 Elsevier Ltd. All rights reserved.

1. Introduction

In asynchronous circuits, local signaling eliminates the need for global synchronization which exploits some potential advantages in comparison with synchronous ones [1–5]. They have shown potential specifications in low power consumption, design reuse, improved noise immunity and electromagnetic compatibility. Asynchronous circuits are more tolerant to process parameter variations [1].

One of the interesting features of asynchronous circuits comparing with their clocked counterparts is better average-case performance [4,5]. However, the performance analysis of asynchronous circuits is a complicated problem and without an effective performance analysis method, one cannot easily take advantage of the properties of asynchronous systems to achieve optimal performance [6]. There are two major technical difficulties involved in the performance analysis of asynchronous systems. First, unlike clocked circuits where clock boundaries form natural partitions for logic between stages to be analyzed individually, an asynchronous circuit is inherently non-linear, meaning there is no easy way to partition the system into independent sub-systems. The systems have to be analyzed as a whole. Second, as the functionality of the system is dependent on the concurrent events, variations in delays mostly caused by process variations in individual components can have considerable effect on the performance of the system. As a result, performance analysis based on the average delay is not accurate. As process variations become a significant problem in deep sub-micron technology, it is really necessary to shift from deterministic timing analysis to statistical timing analysis for high-performance asynchronous circuit designs similarly to what is done in synchronous ones [7,8].

In the first part of this paper, a novel performance analysis of asynchronous circuits is introduced which basically considers process variation as it is one of the most important issues in the recent methods of performance analysis in synchronous circuits [9]. In our approach a synthesized template-base asynchronous circuit is modeled as a Variant-Timed Petri-Net that captures concurrency between interactive components in decision-free [10] systems. Delay variations in component delays caused by process variations are captured in a probabilistic delay model. As the variability of process parameters is naturally random, in order to analyze the effect of the process variations, we applied a statistical approach to the proposed timing analysis. The main goal of the proposed statistical performance analysis is to calculate the probabilistic density function of a performance metric. The proposed method is implemented and tested on the

* Corresponding author. Tel.: +98 2164542701.
E-mail addresses: raji@aut.ac.ir (M. Raji), ghavamib@aut.ac.ir (B. Ghavami), pedram@aut.ac.ir (H. Pedram), h_zarandi@aut.ac.ir (H.R. Zarandi).

0026-2692/$ - see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.mejo.2009.12.013

benchmark circuits modeled in Variant-Timed Petri-Net and its results are compared with the Monte Carlo simulation-based results. The difference between the two methods was shown to be small, ranging from 1.7% to 5.5%. But it is possible to introduce a more accurate method. In fact in order to make the problem simpler, the correlation between the process parameters is not considered in the first step of the process variation-aware performance analysis of asynchronous circuits.

Fundamentally, process variations can be classified into inter-die (also called chip-to-chip) variations and intra-die (also called within chip) variations. Process parameters that change from die to die are called inter-die variations while process parameters that have different values at different points on a die are called intra-die variations. It is noticeable that the intra-die process variations of a gate are spatially correlated with other gates found in its neighborhood [11]. It means that the process parameters values of the gates that are close to each other have a higher probability of being alike than devices that are placed far apart. In order to have a more accurate analysis, it is important to consider the spatial correlation of intra-die parameter variations in addition of incorporating the process variation issue into the analysis.

The second part of this paper includes a more efficient and accurate performance analysis method for asynchronous circuits considering process variation and also incorporating the correlation between process parameters into the analysis [12]. Again a synthesized template-base asynchronous circuit is modeled as a Variant-Timed Petri-Net and also a probabilistic delay model is used to capture the delay variations in component delays which are caused by process variations. As previously we have a statistical approach to the proposed performance analysis to analyze the effects of the process variations. In the second step of the process variation-aware performance analysis of asynchronous circuits, it seems that the calculations will be more complex and therefore a more complicated model and analysis method will be a necessity.

A multi-level quad-tree partitioning was used to extract the correlation between the process parameters during the analysis [11]. The main goal of the proposed statistical timing analysis is to calculate the probabilistic density function of a performance metric. The random variables for each process parameter are correlated to each other with different amounts of correlation. Principal component analysis (PCA) is used to make the analysis tractable [13,14]. PCA transforms a set of correlated variables into smaller set of principal components that are independent and orthogonal. In addition, a Variant-Timed Petri-Net simulator which supports probability density function as the delay of its places is developed to evaluate the performance of the circuit. The correlation coefficients obtained from the mentioned multi-level quad-tree partitioning are applied in the analysis to consider the spatial correlation in Variant-Timed Petri-Net based simulation. Both the proposed SSTA and Petri-Net simulation-based method are implemented and tested on the benchmark circuits modeled in Variant-Timed Petri-Net and its results are compared with the Monte Carlo simulation-based results again. The difference between the proposed methods and MC simulation results was shown to be small, ranging from 0.22% to 3.62%.

Comparing the results of the proposed methods with each other and with Monte Carlo simulation method, we show that considering spatial correlation in process variation-aware performance analysis reduces the error in the analysis and makes the method more accurate.

The remaining part of the paper is organized as follows; Section 2 provides a survey of previous works. In Section 3, some knowledge about process variation and statistical static timing analysis is presented. Section 4 describes Variant-Timed Petri-Nets as the dominant performance analysis model. Section 5 introduces the primary statistical performance analysis method and shows the relevant results. Section 6 shows the multi-level quad-tree partitioning used to compute the coefficient correlations that are applied in the analysis to consider the spatial correlation. Section 7 discusses the statistical static performance evaluation framework. This considers the spatial correlation, in detail. Section 8 introduces the Variant-Timed Petri-Net Simulator. Section 9 presents a comparison of the accuracy of the two methods and shows that the second step improves the process variation-aware analysis. In Section 10, we explain about the modeling and its issues. In the last section, we conclude the paper and probable future works in this scope are proposed.

2. Previous works

There are few works that consider probabilistic delay models in the performance analysis of asynchronous systems [5,7,15,16]. In [5], a method for analyzing the asymptotic performance of asynchronous systems considering delay variation was presented. However, it focused on modeling at system architecture level and did not consider circuit modeling issues such as delay variations due to the process variation. Recently Yahya and Renaudin [17] proposed a performance model for asynchronous linear-pipeline with time variable delays. But in practical designs, asynchronous pipelines can be linear or non-linear and their method cannot be applied to non-linear models easily.

Moreover, none of the above approaches considered the correlation of parameters variation. However, to realize the full benefit of variation-aware performance analysis, one must solve a difficult problem that timing variables in a circuit could be correlated due to inter-parameter dependency, chip-to-chip variation, and within chip spatial dependency.

3. Background

This section reviews technical backgrounds on modeling necessary for the development of this paper.

3.1. Process variation

One of the most important features of deep sub-micron scale CMOS technology is the increasing magnitude of variability of the key parameters which affects performance of integrated circuits. As process variations become relatively large due to technology scaling, uncertainty in performance evaluation is becoming increasingly significant at 90 nm technologies and beyond. Therefore, considering the effects of process variation is an unavoidable part of a design flow for manufacturability. For the circuit designer's sake, the distinction is between inter-chip and intra-chip variability [18]. Intra-die variations are variations in device features that exist within a single chip, meaning that a device feature varies between different locations on the same die. Often, intra-die variations exhibit spatial correlations, where devices that are close to each other have a higher probability of being alike than devices that are placed far apart. Intra-die variation also exhibit structural correlations, meaning that devices that are structurally similar have an increased likelihood of having similar device features, for instance, devices oriented in the same direction tend to be more alike. Inter-chip variations are variations that occur from one die to the next, meaning that the same device on a chip has different features among different die of one wafer, from wafer to wafer, and from wafer lot to wafer lot [18].

Historically, intra-chip variation of parameters could be ignored safely in digital circuit design. But, the patterns of variability are changing by shrinking the scale of the circuits. For example, for 130 nm CMOS technologies, the percentage of the total variation of the effective MOS channel length that can be attributed to intra-chip variation can be up to 35% [19].

From statistical point of view, process variations can be classified into systematic and random variations. As systematic variations are deterministic in nature and are caused by the structure of a particular gate and its topological environment, the process engineer can predict the value of a variable deterministically (and thus analyze, correct, and compensate for it). However, predicting systematic variations may not be feasible for many practical designs because of the following two reasons. First, accurate process simulations can be extremely expensive and may not be applicable to large-size circuits. Second, a lot of design information may not be available at the earlier design stages, further limiting the accuracy of process simulation. For example, placement and routing information is not available at schematic design phase and, therefore, it is difficult to predict systematic variations for schematic design [13]. Now suppose that the process engineer cannot correct the variant process parameter non-uniformity. To a circuit designer, this source of variability will appear statistical. As the placement of each die on the wafer is unspecified and cannot be utilized, the circuit designer cannot deterministically describe the values of parameter affecting each die and thus only a statistical description is possible. Therefore, considering process variation in timing analysis of a circuit requires a statistical approach to the analysis.

3.2. Normal delay distribution approximations

The process parameter variations can be described as normal distributions [13]. The delays, however, can be theoretically modeled as arbitrary distributions for timing analysis. Arbitrary distributions provide the most generality for static statistical analysis. However, they often result in exponential complexity when characterizing the joint probability density function between different random variables.

In contrast, we propose that the gate delays and delays of Timed Petri-Net model can be approximated by normal distributions, even though they are not strictly normal random variables. One significant advantage of treating the delays as nearly normal distributions is that the joint distribution characterization (which includes the correlation between different variables) is greatly simplified. Through some experimental results (Fig. 1), we conclude that the accuracy penalty incurred by this near normal modeling assumption is negligible compared with the error incurred by the loss of correlation information.

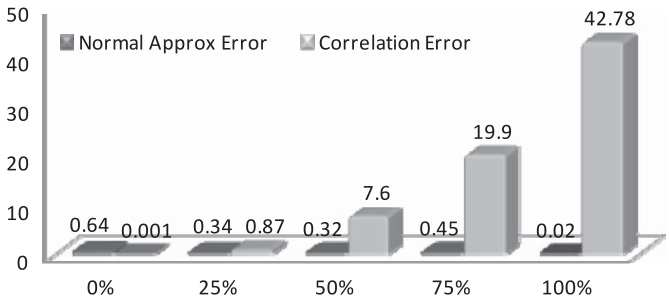

Fig. 1. Delay error compare for 99% in CDF.

3.3. Timed Petri-Net model

Petri-Nets are used as an elegant modeling formalism to model concurrency, synchronization, and choice in many applications including asynchronous circuit modeling [20]. A Petri-Net is a triple $N=(P, T, F)$ where P is the finite set of places, T the finite set of transitions, and $F \subseteq (P \times T) \cup (T \times P)$ is the flow relation. The pre-set of an element $x \subseteq (P \cup T)$ is defined as $\bullet x = \{y \in P \cup T | (y,x) \in F\}$ and its post-set is defined as $x \bullet = \{y \in P \cup T | (x,y) \in F\}$.

A marking is a token assignment for the places and represents the state of the system. Formally, a marking is a mapping M: $P \rightarrow \{0,1,2,\ldots\}$ where the number of tokens in place p under marking M is denoted by $M(p)$. If $M(p) > 0$, then the place p has tokens within. All the places in our simulation system are 1-bounded which means that they maximally can contain only one token. A transition t is enabled at marking M if $M(p) \geq 1$, $\forall p \in \bullet t$. An enabled transition may fire eventually in a feasible asynchronous circuit specification. The firing of t removes one token from each place in its pre-set and inserts one token to each place in its post-set. Timed Petri-Net is a Petri-Net in which some transitions or places can be annotated with delays. In traditional deterministic performance analysis method, places were annotated with deterministic delays as process variation was not an important problem in designs. But as variations in process parameters became considerable in new designs, the designers were motivated to analyze their designs considering process variation.

As a result, it is necessary to apply a performance model which supports the required statistical approach in the modeling of asynchronous circuits.

4. A performance model using Variant-Timed Petri-Net (VTPN) model

Asynchronous circuits after decomposition can be considered as a set of fine grained concurrent modules each one is responsible for producing a single variable. We model the network of templates with a novel Variant-Timed Petri-Net model. The main advantage of this model is that it can be used for simulation of circuits in addition to static performance analysis. In this model, the detailed structures of the original circuit including the handshaking channels are preserved. We have developed a class of models that fully supports full buffer templates [21].

The simplest form of a full buffer is a simple buffer that only reads a value from its input and writes it to its output. This behavior can be modeled simply as shown in Fig. 2. Transition *tW* is analogous to the write statement while place *pWa* emulates the write acknowledge. Similarly *pF* can be seen as the dual for read statement while *tRa* is the corresponding acknowledge. This model is very similar to FCBN model presented in [22] and the only difference is that we added *tRa*. The reason for this is that the used definition of the hierarchical Petri-Nets has a restriction on the input and output ports; all outputs must be transitions and all inputs must be places. This convention ensures that unwanted choices or merge constructs cannot be formed when connecting Petri-Net modules to each other.

We have considered delays on the places, therefore forward delay and backward delay can be put on *pF* and *pB*. In this model, the $d(f)$ represents the forward latency of a channel, while the corresponding $d(p)$ represents the backward latency of channel. We define these values as follows:

- The forward latency represents the delay through an empty channel (and associated cell).
- The backward latency represents the time it takes the handshaking circuitry within the neighboring cells to reset, enabling a second token to flow.

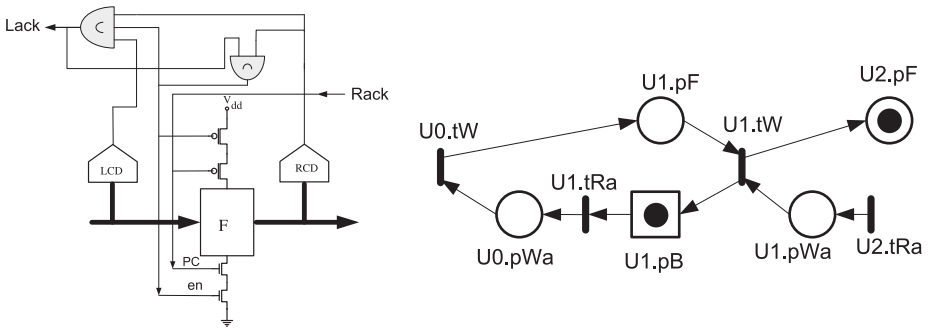

Fig. 2. A simple Full Buffer Template and its Corresponding Timed Petri-Nets model.

The values of these parameters are the normalized delays which are back annotated from the associated cell layout. In other words, the proposed template model exploits the normalized delays model for the sake of accurate performance estimation. Considering process variation in timing modeling necessitate applying probability distribution functions as delay models which will be more explained in further sections.

Performance of any computation modeled with a VTPN is dictated by the cycle time of the VTPN and thus the largest cycle metric. A cycle C in a VTPN is a sequence of places $P_1, P_2, \ldots, P_1$ connected by arcs and transitions whose first and last places are the same. The statistical cycle metric, ($SCM(c)$), is the statistical sum of the delays of all associated places along the cycle C, $d(c)$, divided by the number of tokens that reside in the cycle, $m_0(c)$, defined as

$$SCM(c) = d(c)/m_0(c) \tag{1}$$

The cycle time of a VTPN is defined as the largest cycle metric among all cycles in the Variant-TPN which must be computed statistically, i.e. $\max(SCM(c))\forall c \in C$, where C is the set of all cycles in the Variant-TPN.

In deterministic circuits, the cycle time is captured by the maximum cycle metric of the corresponding TPN model which can be resolved using traditional *Maximum Mean-Cycle* (MMC) algorithms (the *throughput* of the circuit is the reciprocal of this value). Karp's [23] algorithm is one of the fastest and common algorithms for this problem. Dasdan and Gupta [24] introduced a more efficient algorithm for MMC analysis.

One of the advantages of MMC method is its efficiency. With the result of cycle time, we can find the bottleneck of a circuit. However, the solution is different in VTPN as the delays are modeled statistically due to considering process variation.

5. The primary statistical performance analysis method

In this section, we define the modeling assumptions and our proposed SSTA method and also the required operations used in the method.

As mentioned, the delays must be modeled statistically. We model the delays as random variables with normal distributions. So each place in VTPN has a mean delay value, μ, and a standard deviation, σ, which models its variation. In the proposed method, we find all the cycles of the marked graph first. After that, we calculate the statistical cycle metric of each cycle, and then, we find the maximum SCM as the critical metric and the relevant cycle will be supposed as the critical cycle. In all steps of the method, particular operations must be applied based on the model used for modeling the delays in the graph.

As each delay is modeled as a random variable with normal distribution, it is noteworthy to explain about the statistical operations first. The three operations used in our method are SUM, DIV and MAX.

5.1. Sum operation

The sum of two random variables with normal distribution results in a random variable with normal distribution. The two parameters of the result, i.e. the mean, μ, and the standard deviation, σ, are calculated as follows:

$$\begin{aligned} &C = SUM(A,B) = A + B \\ &\mu_C = \mu_A + \mu_B \\ &\sigma_C^2 = \sigma_A^2 + \sigma_B^2 + 2\,\mathrm{cov}(A,B) \end{aligned} \tag{2}$$

5.2. DIV operation

In calculating the SCM of a cycle, the sum of delay values of the cycle will be divided by the number of the tokens in the cycle. As the sum of the delays modeled by normal random variable is still a normal random variable, the parameters of the division are calculated as follows:

$$\begin{aligned} &\mu_{A/n} = \frac{\mu_A}{n} \\ &\sigma_{A/n}^2 = \frac{\sigma_A^2}{n^2} \end{aligned} \tag{3}$$

5.3. MAX operation

The maximum of two normal random variables does not necessarily result in a normal random variable. The MAX of two random variables with normal distributions A and B can be approximated to another normal random variable C using the relationship proposed in [25], that is as follows:

$$\begin{aligned} &C = MAX(A,B) \\ &\alpha = \frac{\mu_A + \mu_B}{2}\,\beta^2 = \sigma_A^2 + \sigma_B^2 - 2\sigma_A\sigma_B\rho \\ &v_1 = (\mu_A^2 + \sigma_A^2)\phi(\alpha) + (\mu_B^2 + \sigma_B^2)\phi(-\alpha) + (\mu_A + \mu_B)\beta\varphi(\alpha) \\ &v_2 = \mu_A\phi(\alpha) + \mu_B\phi(-\alpha) + \beta\varphi(\alpha) \\ &\mu_C = v_2 \\ &\sigma_C^2 = v_1 - \mu_C^2 \end{aligned} \tag{4}$$

Here, ρ represents the correlation coefficient between A and B, and φ and ϕ are the cumulative density function, CDF, and the probability density function, PDF, of a standard normal (i.e., mean 0, STD 1) distribution, respectively.

In our method, as it is supposed that process variables do not increase or decrease simultaneously and we have no assumption on this factor, the correlation coefficients are assumed to be equal

to zero. Because of this and considering (5), the elements of the covariance matrices are also equal to zero:

$$\mathrm{cov}(A,B) = \sigma_A \sigma_B \rho \tag{5}$$

After introducing the necessary functions, it seems to be interesting to present the proposed method more perfectly:

Algorithm SSTA
Input: Variant-Time Petri-Net model of the Circuit
Output: Probability function of the performance metric (μ, σ)

1. Findall the cycles of the VT Petri-Net [26].
2. Calculate the sum of the place delays along the cycle using the SUM operation employing Eq. (2).
3. Divide the result of the previous step by the number of the tokens in the cycle using the DIV operation, employing Eq. (3).
4. Find the maximum SCM between the calculated SCMs in the previous step using the MAX operation employing Eq. (4).

As a result, we have the basic parameters of a normal random variable as a metric to evaluate the performance of the VTPN.

6. Spatial correlation model

At chip level, the spatial correlated variations are most difficult to model. The challenging problem here is how to use the silicon data measured by test structures to accurately extract the spatial correlation for the entire chip that consists of millions of transistors.

One useful approach for modeling correlated intra-die variations is to partition the entire die into a number of grids [27]. The intra-die variations in the same grid are fully correlated, while those in close (far-away) grids are strongly (weakly) correlated. The authors in [11] proposed another hierarchical approach for modeling correlated intra-die variations (this model used here). The key idea is to hierarchically divide the entire chip into a number of regions using a multi-level quad-tree partition, as shown in Fig. 3. At each level i, the die area is partitioned into $2i$ by $2i$ rectangles. The 0th level, for example, contains one rectangle only that covers the entire chip. An independent random variable ε_{ij} is assigned to each region (i,j) to model a portion of the total intra-die variations. The overall variation of a template k is expressed as the sum of the individual components ε_{ij} over all levels of the regions that overlap with the location of the template k.

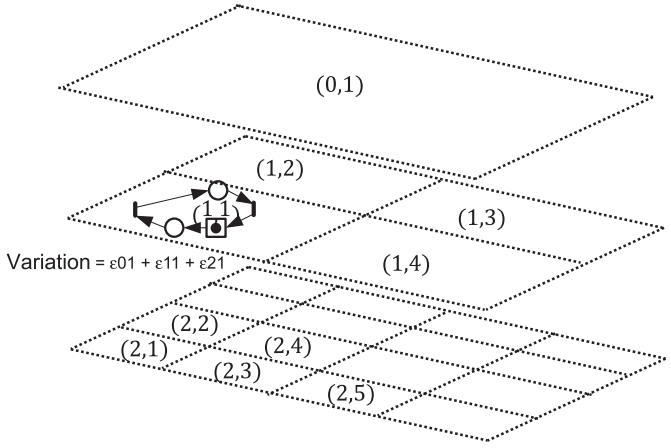

Fig. 3. Hierarchical model for spatial correlation.

Applying this model, we can consider the spatial correlation in the performance analysis of asynchronous circuits which are modeled with VTPN.

7. The improved statistical performance method considering spatial correlation

Again in this step, we model the delays as random variables with normal distributions. As mentioned, it is necessary to use a more complicated model. So each place in VTPN has a mean delay value, μ, and a set of parameter variation. The linear model used to approximate delay in the analysis is as follows:

$$d = \mu + \sum_{i=1}^{m} s_i \Delta p_i \tag{6}$$

where d is the delay of a gate, μ is the mean value for the delay; s_i is the delay sensitivity of process parameter p_i, Δp_i is the parameter variation in p_i for this gate, and m is the number of process parameters.

As mentioned earlier, the process parameters are correlated. To model the intra-die spatial correlations of parameters, we partition the die into a number of grids. Gates placed in close proximity of each other will have many common intra-die variation components resulting in a strong intra-die parameter variation correlation. Gates that lie far apart on a die share few common components and therefore have weaker correlation.

Under this model, a parameter variation in a single grid at location (x_0,y_0) can be modeled using a single random variable $p(x_0,y_0)$. For each type of parameter, L random variables are needed, each representing the value of a parameter in one of the grids in the level L. In addition, we assume that correlation exists only among the same type of parameters and there is no correlation between different types of 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 T_{ox} or W_g in any grid.

Correlation is a challenging problem for statistical timing analysis. When these relationships among process parameters are taken into consideration, the correlation structure becomes even more complicated. To make the problem tractable, we use the principal component analysis (PCA) technique [14,13,1] to transform the set of correlated parameters into an uncorrelated set.

Given N process parameters $X=[x_1,x_2,\ldots,x_N]^T$, the process variation $\Delta X = X - \mu_x$, where μ_x is the mean value of X, is often approximated as a zero-mean *multivariate Normal distribution*. The correlation of ΔX can be represented by a symmetric, positive semi-definite covariance matrix R. PCA decomposes R as follows:

$$R = V \tau V^T \tag{7}$$

where $\tau = diag(\lambda_i) \forall i = 1,2,\ldots,N$, contains the eigenvalues of R, and $V=[V_1,V_2,\ldots,V_N]$ contains the corresponding eigenvectors that are orthonormal, i.e., $V^TV=I$ (I is the identity matrix). Based on τ and V, PCA defines a set of new random variables:

$$\Delta Y = \tau^{-0.5} V^T \Delta X \tag{8}$$

These new random variables in ΔY are called the principal components. It is easy to show that all elements in $\Delta Y=[\Delta y_1,\Delta y_2,\ldots,\Delta y_N]^T$ are uncorrelated and satisfy the standard Normal distribution $N(0,1)$ [13].

Superposing the set of uncorrelated random variables of parameters on the random variables in gate delay as in Eq. (1), the expression of gate delay is then changed to the linear combination of principal components of all parameters:

$$d = \mu + \sum_{i=1}^{m} a_i pc_i \tag{9}$$

while pc_i are the principle components or factors of the delay. The variance of d can be calculated as the sum of the squares of the coefficients, i.e.

$$\sigma^2 = \sum_{i=1}^{m} a_i^2 \tag{10}$$

It is interesting to notice that the covariance between paths (here between path 1 and 2) can be calculated easily through the equation below:

$$\begin{aligned} d_1 &= \mu_1 + \sum_{i=1}^{m} a_{1,i}\, pc_{1,i} \\ d_2 &= \mu_2 + \sum_{i=1}^{m} a_{2,i}\, pc_{2,i} \\ \mathrm{cov}(1,2) &= \sum_{i=1}^{m} a_{i,1} a_{i,2} \end{aligned} \tag{11}$$

As each delay is modeled as a random variable with normal distribution, it is noteworthy to explain about the statistical operations first. The three operations used in our method are SUM, DIV and MAX.

7.1. Sum operation

The sum of two random variables with normal distribution results in a random variable with normal distribution. The SUM operation along each cycle ($d_{sum}=d_1+d_2$) is computed as follows:

$$\begin{aligned} d_1 &= \mu_1 + \sum_{i=1}^{m} a_{1,i} pc_{1,i} \\ d_2 &= \mu_2 + \sum_{i=1}^{m} a_{2,i} pc_{2,i} \\ d_{sum} &= \mu_{sum} + \sum_{i=1}^{m} a_{sum,i} pc_{sum,i} \end{aligned} \tag{12}$$

where $\mu_{sum}=\mu_1+\mu_2$, $a_{sum,I}=a_{1,i}+a_{2,i}\ \ \forall i=1,2,\ldots,m$. The standard deviation of d_{sum} can be calculated using Eq. (12) on the new set of coefficients.

7.2. DIV operation

In calculating the SCM of a cycle, the sum of delay values of the cycle will be divided by the number of the tokens in the cycle. As the sum of the delays modeled by normal random variable is still a normal random variable, the parameters of the division are calculated as follows:

$$\begin{aligned} \mu_{A/n} &= \frac{\mu_A}{n} \\ \sigma^2_{A/n} &= \frac{\sigma_A^2}{n^2} \\ a_{(A/n),i} &= \frac{a_{A,i}}{n} \end{aligned} \tag{13}$$

7.3. MAX operation

The maximum distribution (μ_{max}, σ_{max}) of two normal distributions with means (μ_A, μ_B) and standard deviations (σ_A, σ_B) and a correlation factor of (ρ_{AB}) between the distributions is calculated as follows:

The maximum distribution takes the form:

$$d_{max} = \mu_{max} + \sum_{i=1}^{m} a_{max,i} pc_{max,i} \tag{14}$$

where a_i's are the coefficients of principal components pc_i's, respectively.

Case 1: Standard deviations are equal ($\sigma_A=\sigma_B$) and correlation factor is 0($\rho_{A,B}=0$),

$$d_{max} = \begin{cases} \mu_A & \text{if } \mu_A \geq \mu_B, \\ \mu_B & \text{else} \end{cases} \tag{15}$$

Case 2: Standard deviations are not equal ($\sigma_A \neq \sigma_B$) or correlation factor is not equal to 0 ($\rho_{A,B} \neq 0$), We define two constants (α and β) as follows:

$$\begin{aligned} \alpha &= \frac{\mu_A - \mu_B}{2} \\ \beta^2 &= \sigma_A^2 + \sigma_B^2 - 2\sigma_A\sigma_B\rho_{A,B} \end{aligned} \tag{16}$$

The first moment (d'_{max} or $E(d_{max})$) and the second moment (d''_{max} or $E(d^2_{max})$) of the max distribution are calculated as follows:

$$\begin{aligned} E(d_{max}) &= \mu_A\phi(\alpha) + \mu_B\phi(-\alpha) + \beta\varphi(\alpha) \\ E(d^2_{max}) &= (\mu_A^2+\sigma_A^2)\phi(\alpha) + (\mu_B^2+\sigma_B^2)\phi(-\alpha) + (\mu_A+\mu_B)\beta\varphi(\alpha) \end{aligned} \tag{17}$$

Here, φ and ϕ are the cumulative density function, CDF, and the probability density function, PDF, of a standard normal (i.e., mean 0, STD 1) distribution, respectively. We know that the mean and the standard deviation of the distribution can be calculated using the first and second moments as follows:

$$\mu_{max} = E(d^2_{max})\sigma^2_{max} = E(d^2_{max}) - E(d_{max})^2 \tag{18}$$

The coefficients of the principal components of the new normal distribution are calculated as follows:

$$a_{max,i} = \frac{\sigma_A a_{A,i}\phi(\alpha) + \sigma_B a_{B,i}\phi(-\alpha)}{\sigma_{max}} \tag{19}$$

But as there is a potential for mismatch between the standard deviation calculated using the coefficients and the standard deviation calculated using the closed-form formulae in Eq. (12), the coefficients (a_r) are normalized to reduce the standard deviation potential errors in further calculations using the coefficients and standard deviation:

$$s_0 = \sqrt{\textstyle\sum_{i=1}^{m} a^2_{max,i}}\, a_{max,i} = a_{max,i}\frac{\sigma_{max}}{s_0} \tag{20}$$

After introducing the necessary functions, it seems to be interesting to present the proposed method more perfectly:

Algorithm **SSTA**

Input: Variant-Time Petri-Net model of the Asynchronous Circuit

Output: Probability function of the performance metric (μ, σ)

1. Findall the cycles of the VT Petri-Net[26].
2. Calculate the sum of the place delays along the cycle using the SUM operation employing Eq. (12).
3. Divide the result of the previous step by the number of the tokens in the cycle using the DIV operation, employing Eq. (13).
4. Find the maximum SCM between the calculated SCMs in the previous step using the MAX operation employing Eq. (15) to Eq. (18).
5. Find all coefficients of principle components of the maximum delay calculated in the previous step in order to calculate the path covariance in the next iteration of the algorithm.

Table 1
Comparison results of the proposed method and monte-carlo simulation method.

The circuit	No. of the nodes	No. of the cycles	The proposed SSTA		Monte Carlo		Error (SSTA–MC)/MC%	
			Mu (μ)	Sigma (σ)	Mu (μ)	Sigma (σ)	Mu (μ)	Sigma (σ)
A	6	17	15.448	0.3804	14.9887	0.5359	3.12	−29.02
B	10	51	17.445	0.2478	16.9978	0.3947	2.63	−37.23
C	16	1389	23.210	0.298	21.997	0.620	5.5	−51
D	26	1864	29.982	0.431	29.300	0.662	2.3	−34
F	20	276	17.671	0.150	17.244	0.264	2.4	−43
G	22	5605	21.374	0.184	20.522	0.436	4.1	−51.3
I	56	812	24.392	0.575	23.980	0.708	1.7	−18

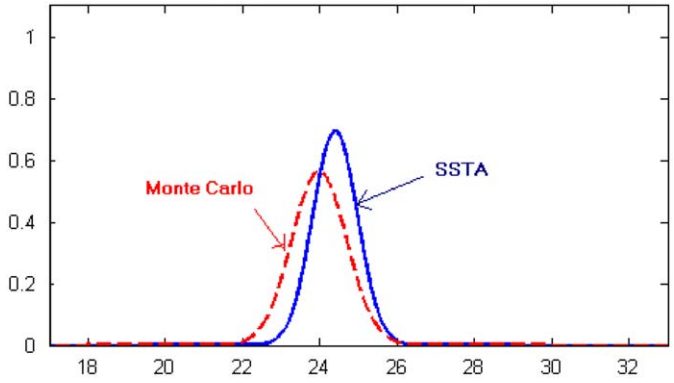

Fig. 4. Comparison of SSTA and MC methods: PDF curves.

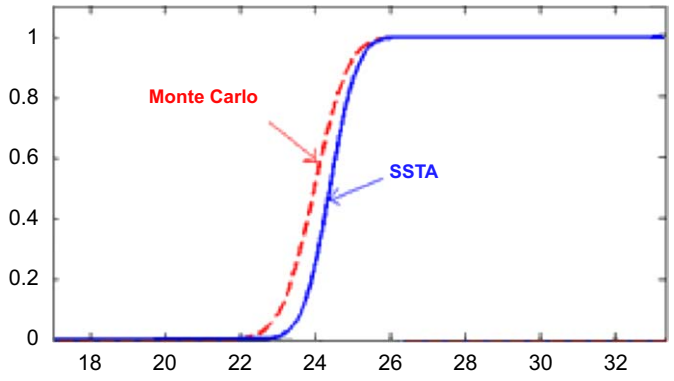

Fig. 5. Comparison of SSTA and MC methods: CDF curves.

6. `Normalize the calculated coefficients using Eq. (19) and Eq. (20) and go to step 4.`

As a result, we have the basic parameters of a normal random variable as a metric to evaluate the performance of the VTPN.

8. Evaluation and experimental results

The proposed SSTA methods have been implemented in C++, and have been tested on a set of benchmark circuits. An asynchronous synthesis toolset (PERSIA [28]) is employed to synthesis benchmarks. Then the developed tool automatically translates the decomposed circuits to its Variant-Timed Petri-Net equivalents. Inputs and outputs of the circuit are connected to each other in Petri-Net structure to form a closed loop system. Initially, all tokens are placed in input nodes.

Variability of process parameters (L, V_{th}, and T_{ox}) and the environmental fluctuation (Vdd) are taken into account. The 3σ values for process parameters are set at 20% of the mean. The standard deviation of V_{dd} is 4% of the maximum, the mean is 96% of the maximum, and the range is 84–100% of the maximum value. In the experiments, V_{th}, T_{ox} and V_{dd} are modeled as probabilistic interval variables. The range of V_{th} and T_{ox} is 80–120% of the mean. Sensitivities of parameters are from SPICE simulations for a cell library of BPTM 0.06 um technology [29].

8.1. Evaluation of primary SSTA method

The proposed method has been run on SUN Ultra Sparc 10 workstation with 1 gigabyte of memory. The sizes of VTPNs range from 6 nodes to 56 nodes. The runtime for our benchmark ranges from 5 to 100 s, depending on circuit sizes and the structure of VTPN model of circuit. To verify the results of our statistical method, we used Monte Carlo (MC) simulation for comparison. To balance the accuracy, we chose to run 10,000 iterations for the MC simulation. The runtime for the MC simulation ranges from 20 to 110 min, depending on circuit sizes and its complexity. A comparison of MC analysis results with those from statistical approach is shown in Table 1. For each test case, the mean and standard deviation (SD) values for both methods are listed. The results of SSTA can be seen to be close to the MC results: the average error is 3.1% and 37.65% for the mean and the variance value of the delays, respectively.

In Figs. 4 and 5, for the largest test case named G, we show the plots of the PDF and CDF of the circuit delay for both SSTA and MC methods. It is observable that the curves almost match each other. It will be shown that a source of the difference between them is the correlation of the process parameters which are not considered in the primary SSTA. In the next section, we consider the spatial correlation between the process parameters and the results show that it will optimize the primary SSTA.

8.2. Evaluation of SSTA method with spatial correlation

The proposed method has been run on SUN Ultra Sparc 10 workstation with 1 gigabyte of memory. The sizes of VTPNs range from 6 nodes to 56 nodes. The runtime for our benchmark ranges from 15 to 200s, depending on circuit sizes and the structure of VTPN model of circuit. To verify the results of our statistical method with Spatial Correlation, we used Monte Carlo (MC) simulation for comparison. To balance the accuracy, we chose to run 10,000 iterations for the MC simulation. The runtime for the MC simulation ranges from 30 min to 12 h, depending on circuit sizes and its complexity. A comparison of these results with those from statistical approach is shown in Table 2. For each test case,

Table 2
Comparison results of the SSTA method (with correlation) and Monte-Carlo simulation method.

The circuit	No. of the nodes	No. of the cycles	SSTA with correlation		Monte Carlo		Error (SSTA–MC)/MC%	
			Mu (μ)	Sigma (σ)	Mu (μ)	Sigma (σ)	Mu (μ)	Sigma (σ)
A	6	17	15.840	0.441	15.752	0.624	0.5	−29.32
B	10	51	17.355	0.386	17.749	0.423	−2.21	−8.74
C	16	1389	22.022	0.386	23.661	0.509	−6.92	−24.1
D	26	1864	30.093	0.547	30.652	0.721	−1.8	−24.1
F	20	276	17.955	0.299	17.964	0.270	−0.05	10.74
G	22	5605	21.0868	0.2899	20.5925	0.2428	−2.40	−19.37
I	56	812	24.000	1.200	24.663	1.006	−2.68	19.28

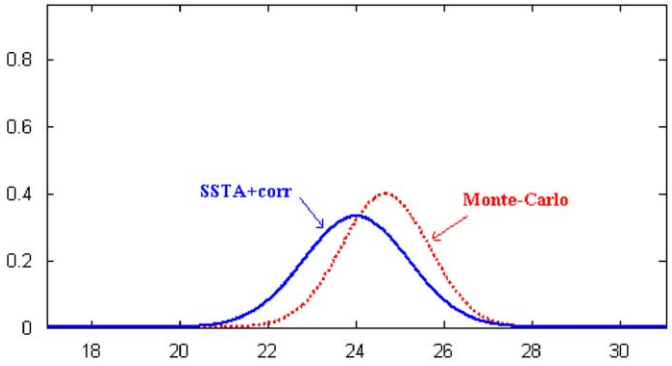

Fig. 6. Comparison of SSTA (with Correlation) and MC methods: PDF curves.

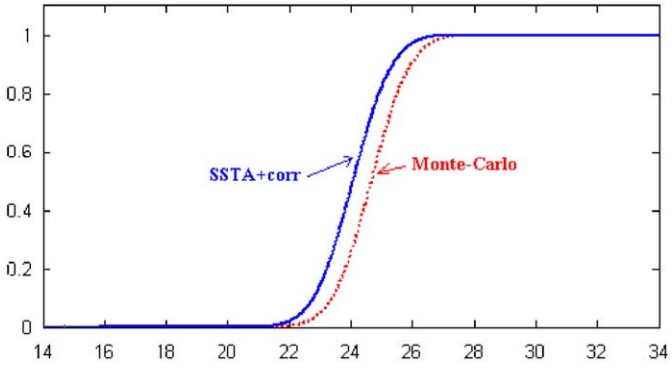

Fig. 7. Comparison of SSTA (with Correlation) and MC methods: CDF curves.

the mean and standard deviation (SD) values for both methods are listed. The results of SSTA with Spatial Correlation can be seen to be close to the MC results: the average error is 2.36% and 19.38% for the mean and the variance value of the delays, respectively.

In Figs. 6 and 7, for the largest test case named *G*, we show the plots of the PDF and CDF of the circuit delay for both SSTA with Spatial Correlation and MC methods. It is observable that the curves almost match each other. The mean values are almost equal although the standard deviations have differences yet. The main source of the remained difference between the curves is the correlation of the process parameters which are considered in SSTA with Spatial Correlation but not in the Monte Carlo simulation. Since the samples in the Monte Carlo simulation are created randomly, the correlation between the samples cannot be considered and the error appears in the comparison stage, especially in the variance part of the results as the correlation is applied in calculating the variance value of the delay more than calculating the mean value.

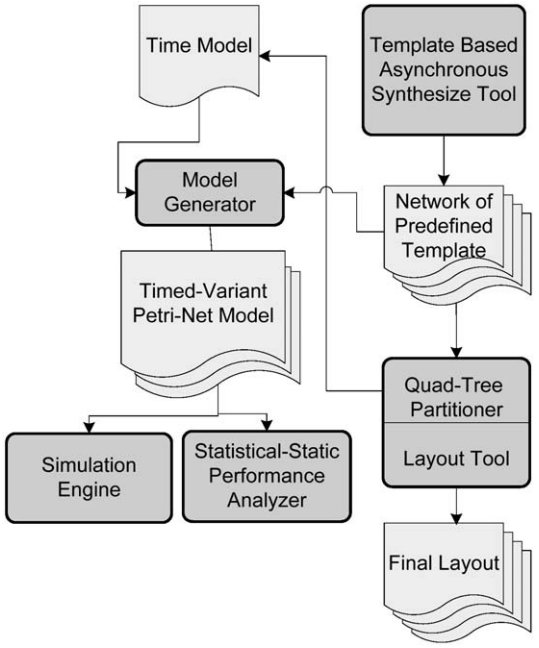

Fig. 8. Variant Timed Petri-Net Simulator.

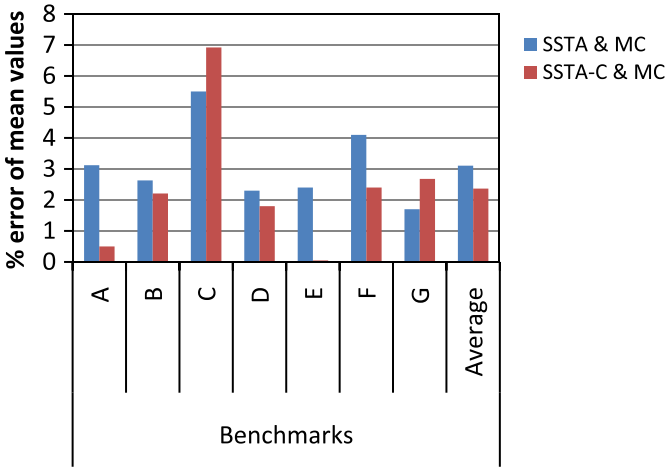

Fig. 9. The Error between the mean values of the proposed methods and the corresponding MC methods.

9. Variant-Time Petri-Net simulator

We have developed a Variant-Timed Petri-Net simulator which supports probability density function as the delay of its places to

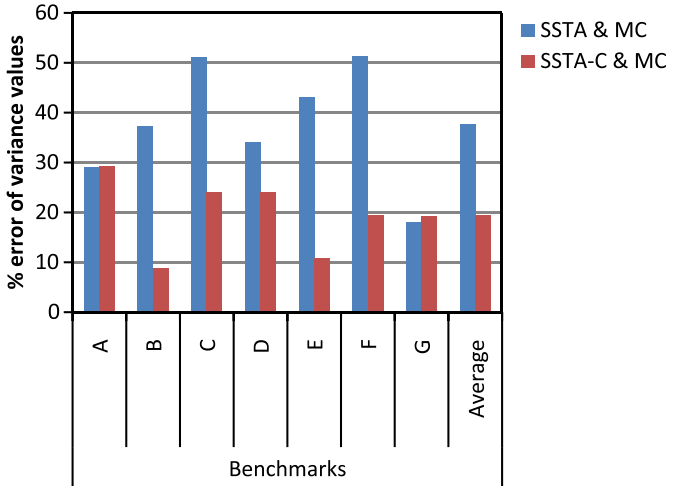

Fig. 10. The Error between the Variance Values of the Proposed Methods and the MC Methods.

evaluate the performance of the asynchronous circuit. Simulation engine supports different delay models including randomly generated delays with fixed, uniform, exponential and normal probability distribution functions. The core simulator is based on systemC [30].

SystemC models for transition and places are developed and the tool is also able to automatically elaborate a detailed systemC model for each input Petri-Net. To consider spatial correlation, the correlation coefficients obtained from the mentioned partitioning are applied in the analysis to consider the spatial correlation in Variant-Timed Petri-Net based simulation.

Fig. 8 shows the general structure of the proposed performance evaluation scheme and its interface with a generic asynchronous synthesis flow.

10. A comparison between the proposed methods

In this section, we compare the proposed methods using a graph. Figs. 9 and 10 show the error between the results of the Monte Carlo and the primary SSTA and the error between the results of the Monte Carlo and the correlation aware SSTA using two graphs. Fig. 9 shows that in almost every case, the former error is more than the latter one. The left most case, which presents the average of the other cases, shows that on average the results from the correlation aware SSTA are more close to the Monte Carlo analysis method than the primary SSTA. Fig. 10 shows that in almost every case, the correlation aware SSTA is more accurate than the SSTA which does not consider the spatial correlation; otherwise they are very close in accuracy.

11. Modeling issues and discussion

This section addresses two limitations of the model used in this paper: the use of a normal delay distribution, and the restriction to decision-free systems. In order to make a system amenable to analytical methods, the delay distributions for components are currently restricted to be normal. This restriction, however, should not have a significant impact on results in practice. Only the mean and the variance of the delay have a significant impact on the result.

Another apparent drawback of our approach is the restriction to decision-free systems, as systems with complicated choice decisions and conflicts cannot be properly modeled as marked graphs. With proper modeling, however, systems with simple choices can be analyzed under our framework. For example, a choice between a slow mode and a fast mode of operation of a certain component can be modeled using a discrete delay distribution. In some cases, more complicated choices can be handled hierarchically. Earlier, we mentioned this limitation and solved this problem by proposing a novel static performance analysis method for non-deterministic asynchronous circuit using the Probabilistic Timed Petri-Net model [21]. The proposed method had the advantage of exploiting a new performance metric and presented an efficient methodology for static estimation of average performance of non-deterministic asynchronous circuits with choices at the template level. Using the average performance analysis method [2], the proposed statistical static timing analysis method can be extended to practical asynchronous circuits which can be considered as a future work.

12. Conclusion and future work

Even though asynchronous circuits are highly tolerant to process variation, it seemed to be necessary to present a method to analyze the performance of asynchronous circuits considering the variation in process parameters. This view has facilitated us to derive an efficient method to analyze system performance, and to define meaningful performance metrics for optimization. In this paper, we present a method to analyze the performance of template-based asynchronous circuits statistically. Asynchronous circuit has been modeled using Variant-Timed Petri-Net. Based on this model, the probability density function of the delay of global critical cycle is calculated. We also present a simulation tool of Variant-Timed Petri-Net and the results of the experiments are compared with Mont Carlo simulation results. We demonstrated our method via a tool. Results show that it is possible to consider the process variation and analyze the performance while there is ignorable error between the results of the proposed method and the general Monte Carlo simulation method. We considered the spatial correlation to increase the accuracy of the primary proposed analysis method and a comparison between the results shows that it reduces the differences between the results of the proposed SSTA and the Monte Carlo simulation method.

We see many avenues for further investigation. Research goals in the immediate future include extensions to analyze asynchronous systems with choice, the development of performance/power optimization algorithms [31,19] for asynchronous systems driven by our analysis technique, and the application of our method to a broader class of concurrent systems, such as GALS systems.

References

[1] C.K. Tang, C.Y. Lin, Y.C. Lu, An asynchronous circuit design with fast forwarding technique at advanced technology node, in: Proceedings of the Ninth International Symposium of Quality Electronic Design, 2008, pp. 769–773.
[2] Peter A. Beerel, Asynchronous circuits: an increasingly practical design solution, in: Proceedings of the Third International Symposium of Quality Electronic Design, 2002, pp. 367–372.
[3] Alain J. Martin, et al., The Lutonium: a sub-nanojoule asynchronous 8051 microcontroller, in: Proceedings of the Ninth International Symposium on Asynchronous Circuits and Systems, 2003, pp. 14–23.
[4] K.Y. Yun, P.A. Beerel, V. Vakilotojar, A.E. Dooply, J. Arceo, A low-control-overhead asynchronous differential equation solver, in: Proceedings of the Third International Symposium on Asynchronous Circuits and Systems, 1997, pp. 140–155.
[5] O. Garnica, J. Lanchares, R. Hermida, Fine-grain asynchronous circuits for low-power high performance DSP implementations, in: Proceedings of the IEEE Computer Society Workshop on VLSI, 2000, pp. 111–118.
[6] P.B. McGee, S.M. Nowick, E.G. Coffman, Efficient performance analysis of asynchronous systems based on periodicity, in: Proceedings of the Third

International Conference on Hardware/Software Codesign and System Synthesis, 2005, pp. 225–230.
[7] C. Visweswariah, et al., First-order incremental block-based statistical timing analysis, in: IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2004, pp. 2170–2180.
[8] W. Wang, V. Kreinovich, M. Orshansky, Statistical timing based on incomplete probabilistic descriptions of parameter uncertainty, in: Proceedings of the 43rd Design Automation Conference, 2006, pp. 161–166.
[9] M. Raji, B. Ghavami, H. Pedram, Statistical static performance analysis of asynchronous circuits considering process variation, in: Proceedings of the 10th International Symposium of Quality Electronic Design, 2009, pp. 291–296.
[10] F. Commoner, A. Holt, S. Even, A. Pnueli, Marked directed graphs, Journal of Computer and System Sciences 5 (1971) 511–523.
[11] A. Agarwal, D. Blaauw, V. Zolotov, Statistical timing analysis for intra-die process variations with spatial correlations, in: Proceedings of the International Conference on Computer Aided Design, 2003, pp. 900–907.
[12] M. Raji, B. Ghavami, H. Pedram, Hamid-Reza Zarandi, Process variation aware performance analysis of asynchronous circuits considering spatial correlation, in: Proceedings of the Ninth International Workshop on Power and Timing Modeling, Optimization and Simulatio, 2009.
[13] X. Li, J. Le, L.T. Pileggi, Statistical performance modeling and optimization, Foundation and Trends in Electronic Design Automation 1 (4) (2003) 331–480.
[14] G. Seber, Multivariate Observations, Wiley Series, 1984.
[15] A. Xie, S. Kim, P.A. Beerel, Bounding average time separations of events in stochastic timed Petri nets with choice, in: Proceedings of the Fifth International Symposium on Asynchronous Circuits and Systems, 1999, p. 94.
[16] P.B. Pang, M. Greenstreet, Self-timed meshes are faster than synchronous, in: Proceedings of the Third International Symposium on Asynchronous Circuits and Systems, 1997, pp. 30–39.
[17] E. Yahya, M. Renaudin, Performance modeling and analysis of asynchronous linear-pipeline with time variable delays, in: Proceedings of the 14th International Conference on Electronics, Circuits and Systems, 2007, pp. 1288–1291.
[18] M. Orshansky, S.R. Nassif, D. Boning, Design for Manufacturability and Statistical Design, A Constructive Approach, Springer Press, 2008 pp. 11–15.
[19] B. Ghavami, H. Pedram, Design of dual threshold voltages asynchronous circuits, in: Proceeding of the 13th International Symposium on Low Power Electronics and Design, 2008, pp. 185–188.
[20] Sangyun Kim, Pipeline optimization for asynchronous circuits, Ph.D. Thesis, University of Southern California, August 2003.
[21] B. Ghavami, H. Pedram, High performance asynchronous design flow using a novel static performance analysis method, Computers and Electrical Engineering 35 (6) (2009) 920–994.
[22] P.A. Beerel, N. Kim, A. Lines, M. Davies, Slack matching asynchronous designs, in: Proceedings of the 12th IEEE International Symposium on Asynchronous Circuits and Systems, 2006, p. 184.
[23] R.M. Karp, A characterization of the minimum cycle mean in a diagraph, Discrete Mathematics 23 (1978) 309–311.
[24] A. Dasdan, R.K. Gupta, Faster maximum and minimum mean cycle algorithms for system performance analysis, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems 17 (10) (1998) 889–899.
[25] C.E. Clark, The greatest of a finite set of random variable, Operations Research 9 (1961) 85–91.
[26] H. Liu, J. Wang, A new way to enumerate cycles in graph, in: Proceedings of the Advanced International Conference on Telecommunications and International Conference on Internet and Web Applications and Services, 2006, p. 57.
[27] H. Chang, S. Sapatnekar, Statistical timing analysis under spatial correlations, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems 24 (no. 9) (2005) 1467–1482.
[28] Persia: A QDI Asynchronous Synthesis Tool, ASYNC, 2008.
[29] PTM: 〈http://www.eas.asu.edu/~ptm〉.
[30] Braham Lane, SystemC Language Reference Manual, Copyright © Open SystemC Initiative, San Jose, CA, 2003.
[31] P. Prakash, A.J. Martin, Slack matching quasi delay-insensitive circuits, in: Proceedings of the Seventh International Symposium on Asynchronous Circuits and Systems, 2006, pp. 195–204.