Probabilistic Transitive-Closure Ordering and Its Application on Variational Buffer Insertion

Jinjun Xiong and Lei He

***Abstract*—We propose a provably transitive-closure ordering rule with theoretical foundations to prune suboptimal design solutions in the presence of process variations. As an example, this probabilistic ordering rule is applied to develop an efficient variational buffering algorithm. Compared to the conventional deterministic approach, variational buffering improves the parametric timing yield by 15.7% on average. This transitive-closure ordering rule may be leveraged to solve other computer-aided-design problems considering process variation effects.**

***Index Terms*—Buffering, closure, probabilistic ordering, process variation, transitive ordering, variation metric comparison.**

I. Introduction

As we move into the ultradeep submicrometer era, integrated circuits exhibit substantial performance variability because of manufacturing process variations. To combat this phenomenon, statistical design analysis and optimization has thus become a major research focus in recent years, such as in [1]–[3].

In the presence of process variation, the conventional design metrics, such as timing and power, are no longer a constant number but a random variable. This makes it challenging to compare the quality of different design solutions, as under different process space, the same design may result in different design metrics. How do we compare different metrics so that one solution is always better than the other in the statistical sense? Is it always possible to compare two solutions with its design metric represented as a distribution? Does the variational comparison guarantee the transitive-closure property as usual deterministic comparison? In other words, if solution one is better than solution two, solution two is better than solution three, can we say for sure that solution one is also better than solution three? Although there are some works in literature [4]–[7] that have tried to compare different variational solutions, they are all based on heuristics and no definite answer is available yet.

The major contribution of this paper is to provide some theoretical foundations to answer the above questions. Based on the normal distribution assumption, we propose a probabilistic transitive-closure ordering rule that makes variational evaluation of different solutions possible. To show the usefulness of this rule, we apply it to the variation-aware-buffer-insertion (vawBuf) problem, the same as in [4] and [5], and show an efficient implementation of the algorithm. Compared to the conventional deterministic buffering, our buffering algorithm considering correlated process variations improves the parametric timing yield by 15.7%, on average. We believe, although not verified, that this transitive-closure ordering rule can be leveraged to solve other computer-aided-design problems in the presence of process variation effects.

Manuscript received May 2, 2006; revised August 9, 2006 and October 1, 2006. This work was supported in part by the NSF under CAREER award CCR-0401682 and by a UC MICRO grant sponsored by Actel and Intel. This paper was recommended by Guest Editor J. Hu.

J. Xiong is with the IBM Thomas J. Watson Research Center, Yorktown Heights, NY 10589 USA (e-mail: jinjun@us.ibm.com).

L. He is with the University of California, Los Angeles, CA 90096 USA.

Digital Object Identifier 10.1109/TCAD.2007.891360

II. Probabilistic Transitive-Closure Ordering Rule

We observe that for deterministic ordering, the following two properties hold: 1) For any two given deterministic design metrics corresponding to two different design solutions, there exists an ordering property between them, i.e., T_1 is either greater than T_2 or less than T_2. 2) There exists a transitive ordering property between different design metrics, i.e., if $T_1 > T_2$ and $T_2 > T_3$, then $T_1 > T_3$.

The above two properties, however, are not necessarily true when design metrics are given as random variables. But it is easy to see that any ordering rule that ensures the above two properties can be used to evaluate different solutions in the presence of process variations. Toward this goal, we first extend the deterministic ordering relation between T_1 and T_2 by enforcing

$$P(T_1 > T_2) = 1 \qquad (1)$$

i.e., solution T_1 has 100% probability (almost always) to result in a larger value when compared to solution T_2. We have the following lemma.

Lemma 1: Given T_1, T_2, and T_3 as three correlated random variables with arbitrary distributions, if $P(T_1 > T_2)=1$, $P(T_2 > T_3)=1$, then $P(T_1 > T_3) = 1$.

Proof: Let $X = T_1 - T_2$ and $Y = T_2 - T_3$ and the joint probability density function (JPDF) of X and Y be $f(x, y)$. As $P(T_1 > T_2) = P(X > 0) = \int_0^{+\infty} dx \int_{-\infty}^{+\infty} f(x, y)dy = 1$, we have $\int_{-\infty}^{0} dx \int_{-\infty}^{+\infty} f(x, y)dy = 1 - \int_0^{+\infty} dx \int_{-\infty}^{+\infty} f(x, y)dy = 0$. Because $f(x, y) \geq 0$ for all x and y, we have $f(x, y) = 0$ for $x < 0$.

Similarly, from $P(T_2 > T_3) = P(Y > 0) = 1$, we have $f(x, y) = 0$ for $y < 0$. Therefore, we have $\int_0^{+\infty} dy \int_0^{+\infty} f(x, y)dx = 1$. Then, $P(T_1 > T_3)=P(X + Y > 0) = \int_{-\infty}^{+\infty} dy \int_{-y}^{+\infty} f(x, y)dx = \int_0^{+\infty} dy \int_{-y}^{+\infty} f(x, y)dx \geq \int_0^{+\infty} dy \int_0^{+\infty} f(x, y)dx = 1$. As we know $P(T_1 > T_{3)} \leq 1$, we must have $P(T_1 > T_3) = 1$. ■

Lemma 1 shows that comparison between T_1 and T_2 based upon $P(T_1 > T_2) = 1$ enforces the transitive ordering property between solutions. However, for any two given solutions, it is not always possible to compare them. Moreover, in practice, such a 100% probability requirement is too restrictive. Therefore, we relax such a requirement so that T_1 is said to be greater than T_2 if the following condition holds:

$$P(T_1 > T_2) \geq \overline{p_T} = 0.5. \qquad (2)$$

In other words, it is likely that T_1 is greater than T_2 in the probabilistic sense. We have the following lemma.

Lemma 2: Given T_1 and T_2 as two correlated but different random variables with arbitrary distributions, we have either $P(T_1 > T_2) \geq 0.5$ or $P(T_1 < T_2) \geq 0.5$.

Proof: The proof follows directly from the fact that $P(T_1 > T_2) + P(T_1 < T_2) = 1$. ■

Lemma 2 shows that comparison based upon $P(T_1 > T_2) \geq 0.5$ results in a proper ordering between two design metrics. But for arbitrary distributions, we can show that it does not preserve the transitive ordering property, in general. In the following, we prove that when the random solutions follow a joint normal distribution, both properties indeed hold simultaneously.

Lemma 3: Given T_1, T_2, and T_3 as three correlated but different random variables with joint normal distributions, if $P(T_1 > T_2) \geq 0.5$, $P(T_2 > T_3) \geq 0.5$, then $P(T_1 > T_3) \geq 0.5$.

Proof: According to [8], the probability of $T_1 > T_2$ is given by $P(T_1 > T_2) = \Phi[(\mu_1 - \mu_2)/\sigma_{1,2}]$, where Φ is the cumulative density function (CDF) of a standard normal distribution; μ_1 and μ_2 are the mean values of T_1 and T_2, respectively. $\sigma_{1,2}$ can be computed by

$$\sigma_{1,2} = \left(\sigma_1^2 - 2 \cdot \rho_{1,2}\sigma_1\sigma_2 + \sigma_2^2\right)^{1/2} \tag{3}$$

where σ_1^2 and σ_2^2 are variance of T_1 and T_2, respectively, and $\rho_{1,2}$ is the correlation coefficient of T_1 and T_2. Because any CDF function is a nondecreasing function, and for the standard normal distribution $\Phi(0) = 0.5$, then we have $\Phi(x) \geq 0.5$ for any $x \geq 0$. Therefore, to have $P(T_1 > T_2) \geq 0.5$ is equivalent to have $(\mu_1 - \mu_2)/\sigma_{1,2} \geq 0$. Because $\sigma_{1,2}$ is positive according to (3), hence we have $\mu_1 \geq \mu_2$. Knowing $P(T_1 > T_2) \geq 0.5$ and $P(T_2 > T_3) \geq 0.5$, we have $\mu_1 \geq \mu_2$ and $\mu_2 \geq \mu_3$. Therefore, we have $\mu_1 \geq \mu_3$, which is equivalent to $P(T_1 > T_3) \geq 0.5$. ■

Next, we discuss the extension of the above transitive-closure ordering rule for other choices of $\overline{p_T}$. We have the following theorem which proves that for $\overline{p_T}$ between 0.5 and 1, the transitive ordering property always holds.

Theorem 1: Given T_1, T_2, and T_3 as three correlated but different random variables with joint normal distributions, if $P(T_1 > T_2) > \overline{p_T}$, $P(T_2 > T_3) > \overline{p_T}$, then $P(T_1 > T_3) > \overline{p_T}$ for any $\overline{p_T}$ between 0.5 and 1.

Proof: Define $X = T_1 - T_2$ and $Y = T_2 - T_3$, then we have $X + Y = T_1 - T_3$. Therefore, $P(T_1 > T_2) = P(X > 0)$, $P(T_2 > T_3) = P(Y > 0)$, and $P(T_1 > T_3) = P(X + Y > 0)$. Because T_1, T_2, and T_3 are joint normal, then X and Y are also normal. Denote the probability density function (pdf) of X as $N(\mu_x, \sigma_x)$ and the pdf of Y as $N(\mu_y, \sigma_y)$, where μ_x and σ_x (similarly μ_y and σ_y) are the mean and standard deviation for X (similarly Y), respectively. Hence, we can obtain the pdf of $X + Y$, which is also a normal distribution, as $N(\mu_x + \mu_y, \sqrt{\sigma_x^2 + \sigma_y^2 + 2\rho\sigma_x\sigma_y})$ with ρ being the correlation coefficient between X and Y. We have

$$P(X > 0) = P\left(\frac{X - \mu_x}{\sigma_x} > -\frac{\mu_x}{\sigma_x}\right) = 1 - \Phi\left(-\frac{\mu_x}{\sigma_x}\right). \tag{4}$$

According to the property of the standard normal distribution $\Phi(-t) = 1 - \Phi(t)$ we have

$$P(T_1 > T_2) = P(X > 0) = \Phi\left(\frac{\mu_x}{\sigma_x}\right). \tag{5}$$

As we already know $P(T_1 > T_2) > \overline{p_T}$, we hence have $\Phi(\mu_x/\sigma_x) > \overline{p_T}$. Since any CDF function is also a nondecreasing function, we have

$$\frac{\mu_x}{\sigma_x} > \bar{t} \tag{6}$$

where $\Phi(\bar{t}) = \overline{p_T}$. Moreover, for $0.5 \leq \overline{p_T} \leq 1$, we have $\bar{t} > 0$. Similarly, we have

$$\frac{\mu_y}{\sigma_y} > \bar{t}. \tag{7}$$

From (6) and (7), we have

$$\mu_x + \mu_y > (\sigma_x + \sigma_y)\bar{t} \tag{8}$$

$$\frac{\mu_x + \mu_y}{\sqrt{\sigma_x^2 + \sigma_y^2 + 2\rho\sigma_x\sigma_y}} > \frac{(\sigma_x + \sigma_y)\bar{t}}{\sqrt{\sigma_x^2 + \sigma_y^2 + 2\rho\sigma_x\sigma_y}}. \tag{9}$$

Because $-1 \leq \rho \leq 1$, it is easy to show that

$$\frac{(\sigma_x + \sigma_y)}{\sqrt{\sigma_x^2 + \sigma_y^2 + 2\rho\sigma_x\sigma_y}} \geq 1. \tag{10}$$

As $\bar{t} > 0$, by multiplying both sides of (10) by $\bar{t} > 0$ and then combining it with (9), we have

$$\frac{\mu_x + \mu_y}{\sqrt{\sigma_x^2 + \sigma_y 2 + 2\rho\sigma_x\sigma_y}} > \frac{(\sigma_x + \sigma_y)\bar{t}}{\sqrt{\sigma_x^2 + \sigma_y 2 + 2\rho\sigma_x\sigma_y}} \geq \bar{t}. \tag{11}$$

Therefore, by the fact that Φ is a nondecreasing function and (11), we finally have

$$\begin{aligned} P(T_1 > T_3) &= P(X + Y > 0) \\ &= \Phi\left(\frac{\mu_x + \mu_y}{\sqrt{\sigma_x^2 + \sigma_y^2 + 2\rho\sigma_x\sigma_y}}\right) \\ &> \Phi(\bar{t}) = \overline{p_T}. \end{aligned} \tag{12}$$

■

III. Application on Buffer Insertion

The above transitive-closure ordering rule has many potential applications in design automation in the presence of process variation. As a proof of concept, we illustrate its use by solving the vawBuf problem [4]–[6].

Formulation 1—vawBuf Problem: Given a routing tree with parasitic capacitance and resistance, legal buffer positions, and required arrival times and loading capacitances specified at all sinks, determine the placement of buffers in the routing tree such that the probability of the required arrival time at the root meeting the design specification is maximized with the consideration of process variations for both interconnect and devices. When there are multiple valid solutions at the root, as a secondary objective, we choose the one with minimum number of buffers.

Two figures-of-merit are associated with every legal buffer position t in the tree, i.e., the downstream loading capacitance C_t and the required arrival time T_t. We characterize a device (buffer) in terms of its gate capacitance (C_b), intrinsic delay (T_b), and output resistance (R_b). For a given interconnect segment, we characterize it by its lumped resistance R_w and capacitance C_w.

As in [2], we employ the first-order canonical form to model all characteristics of interests as a random variable. Take T_b as an example, we have $T_b = T_{b0} + \gamma_b^T X$, where T_{b0} is the mean value of T_b; X is a normalized random vector that includes the interchip variation, spatial correlation, and uncorrelated random variation; γ_b is the corresponding coefficient vector of X. All random variables in X are mutually independent and follow a standard normal distribution, i.e., $X \sim N(0, I)$. In the following, we denote $C_b = C_{b0} + \eta_b^T X$, $R_b = R_{b0} + \zeta_b^T X$, $C_w = C_{w0} + \eta_w^T X$, and $R_w = R_{w0} + \zeta_w^T X$.

We follow the same deterministic buffering algorithm (detBuf) based on dynamic programming [9] to solve the vawBuf problem. Under the Elmore delay model, at each node t, new solutions are generated by the following three key operations: 1) adding a wire; 2) adding a buffer; and 3) merging two solutions. The atomic operations involved are addition (subtraction), minimum, and multiplication.

It has been shown in block-based statistical timing analysis [1], [2] that addition (subtraction) and minimum operations on two canonical forms can be easily represented as a new canonical form. Hence, the new C_t and T_t are still in canonical form when only those operations are involved. But T_t after adding a wire or buffer also involves multiplication operation. In the following, we propose an approximation technique that keeps T_t after multiplication operations still in the first-order canonical form. For example, T_t after adding a wire for solution (T_n, C_n) with $T_n = T_{n0} + \beta^T X$ and $C_n = C_{n0} + \alpha_n^T X$ is computed by

$$\begin{aligned} T_t &= T_n - R_w \cdot C_n - \frac{1}{2} \cdot R_w \cdot C_w \\ &= T_{t0} + \delta_t^T \cdot X - X^T \Gamma X \end{aligned} \tag{13}$$

where $T_{t0} = T_{n0} - R_{w0} \cdot C_{n0} - (1/2) \cdot R_{w0} \cdot C_{w0}$, $\delta_t = \beta_n - C_{n0} \cdot \zeta_w - R_{w0} \cdot \alpha_n - (1/2)(\cdot R_{w0} \cdot \eta_w + C_{w0} \cdot \zeta_w)$, and $\Gamma = \zeta_w \cdot \alpha_n^T + (1/2)\zeta_w \cdot \eta_w^T$. It is obvious that T_t in (13) is not a canonical form because of the quadratic term $X^T \Gamma X$. To represent T_t as a canonical form, we introduce the following theorems.

Theorem 2: Given random variables in vector form X that follow a standard multivariate Gaussian distribution as N(0,I), for any vector δ and matrix Γ, we have [10]

$$E(X^T \Gamma X) = \mathrm{tr}(\Gamma) \tag{14}$$

$$E(X^T \Gamma X \delta^T X) = 0 \tag{15}$$

$$E\left((X^T \Gamma X)^2\right) = 2\mathrm{tr}(\Gamma^2) + \mathrm{tr}(\Gamma)^2 \tag{16}$$

where $E(\cdot)$ is the expectation operation of a random variable, and $\mathrm{tr}(\cdot)$ is the trace operation of a matrix which takes the sum of diagonal elements of the matrix.

Theorem 3: Given $\Gamma = \alpha\beta^T + \zeta\eta^T$, we have

$$\mathrm{tr}(\Gamma) = \beta^T \alpha + \eta^T \zeta \tag{17}$$

$$\mathrm{tr}(\Gamma^2) = (\beta^T \alpha)^2 + (\eta^T \zeta)^2 + 2(\beta^T \zeta)(\eta^T \alpha) \tag{18}$$

where α, β, ζ, and η are all vectors.

With the aid of the above two theorems, the mean and variance of T_t in (13) can be computed as follows:

$$\mu(T_t) = E(T_t) = T_{t0} - \mathrm{tr}(\Gamma) \tag{19}$$

$$\sigma^2(T_t) = E(T_t^2) - E(T_t)^2 = \delta_t^T \delta + 2\mathrm{tr}(\Gamma^2). \tag{20}$$

We then approximate (13) by the following canonical form that matches its exact mean and variance[1]:

$$T_t = (T_{t0} - \mathrm{tr}(\Gamma)) + \sqrt{1 + \frac{2\mathrm{tr}(\Gamma^2)}{\delta_t^T \delta}} \cdot \delta_t^T \cdot X. \tag{21}$$

Knowing the above three key operations, the dynamic programming-based buffer insertion can be solved by recursively applying the above three operations to obtain new solutions as we traverse the routing tree bottom-up. Moreover, because we always keep solutions in first-order canonical form after each operation, we can apply the same technique recursively to compute all new solutions while traversing the routing tree bottom up.

To make the dynamic programming-based buffering algorithm tractable, [9] proposed to define the dominance relationship (or pruning rule) between two solutions such that solution (C_1, T_1) dominates solution (C_2, T_2) if condition $C_1 < C_2$ and $T_1 > T_2$ are satisfied.

In the presence of process variation, we propose the following variation-aware pruning rule, i.e., solution (C_1, T_1) is said to dominate solution (C_2, T_2) if the following two conditions hold:

$$P(C_1 < C_2) \geq 0.5, \quad P(T_1 > T_2) \geq 0.5. \tag{22}$$

[1]By simply ignoring the quadratic term, [6] approximates (13) as a canonical form, thus losing accuracy even on the second-order term.

TABLE I
RUNTIME COMPARISON IN SECONDS

Bench	sinks	detBuf	T2P [5]	vawBuf
p1	269	0.003	25.4	1.0
p2	603	0.012	-	4.3
r1	267	0.004	-	3.6
r2	598	0.004	-	15.0
r3	862	0.023	-	27.5
r4	1903	0.044	-	88.9
r5	3101	0.083	-	195.8

In other words, it is likely that C_1 is less than C_2 and T_1 is greater than T_2 in the probabilistic sense. We call the pruning rule as defined by (22) as transitive-closure-based pruning rule.

According to Lemmas 2 and 3, we can compare and sort random solutions based on their respective mean values. Hence, for two sets of sorted random solutions, the pruning can be done in linear time based on a merge-sort like operation. Following similar arguments as in [9] and [11], we conclude that our variation aware buffer insertion algorithm has the same complexity as the deterministic algorithm as $O(B \cdot N^2)$ with B types of buffers and N number of legal buffer locations.

IV. EXPERIMENT RESULTS

Two sets of benchmarks are obtained from the public domain for our experiments [12]. Without loss of generality, the 65-nm BSIM technology is assumed with one buffer type in the library. We budget the 3-sigma random device variation, interdie global variation, intradie spatial variation, and interconnect variation all to be 5% of its nominal value, respectively.

We compare our approach with [5], in which a two-side threshold-based pruning rule (denoted as T2P) is proposed and it relates the dominance relationship of solutions to designers' willingness of accepting uncertainty. A threshold value π_α gives a measure of a designer's preference for certainty in choosing the design parameter x in the presence of variations, such that the final design would have x less than π_α with $(100\alpha)\%$ certainty, i.e., $\alpha = \int_{-\infty}^{\pi_\alpha} f(x)dx$, where $f(x)$ is the pdf of x. Given two different thresholds for either C_t and T_t, for example π_{α_l} and π_{α_u} for C_t, and π_{β_l} and π_{β_u} for T_t, such that $0 \leq \alpha_l < \alpha_u \leq 1$ and $0 \leq \beta_l < \beta_u \leq 1$, solution (C_1, T_1) is said to dominate solution (C_2, T_2) if the following conditions are satisfied:

$$\pi_{\alpha_u}^{(1)} < \pi_{\alpha_l}^{(2)}, \quad \pi_{\beta_l}^{(1)} > \pi_{\beta_u}^{(2)}. \tag{23}$$

In other words, C_1's upper threshold $\pi_{\alpha_u}^{(1)}$ is smaller than C_2's lower threshold $\pi_{\alpha_l}^{(2)}$, while T_1's lower threshold $\pi_{\beta_l}^{(1)}$ is larger than T_2's upper threshold $\pi_{\beta_u}^{(2)}$.

We compare the efficiency of the two different pruning rules[2] in Table I. We see that the T2P algorithm only finishes the first benchmark and fails for the rest of the benchmarks due to exceeding either memory capacity $(2G)$ or tolerable time limit (4 h in our setting). This observation is expected, because the two-side threshold-based pruning rule only imposes partially ordering between solutions, rendering the

[2]To compute the JPDF of C_t and T_t explicitly with the assumption of independence between device and interconnect variations, [5] employed a numerical integration method. For a fair comparison, we reimplemented the algorithm of [5] under the same process variation model as used in this paper with $\alpha_l = \beta_l = 0.2$ and $\alpha_u = \beta_u = 0.8$.

TABLE II
COMPARISON BETWEEN DETBUF AND VAWBUF

Bench	detBuf			vawBuf	
	Buffer	$D_{3\sigma}$	Yield Loss	Buffer	$D_{3\sigma}$
p1	58	2403	0%	60 (3.3%)	2403
p2	149	3203	0%	156 (4.5%)	3204
r1	59	790	0%	65 (9.2%)	790
r2	112	1128 (1.5%)	35.3%	135 (17%)	1111
r3	173	1147 (0.5%)	1.6%	188 (8%)	1142
r4	320	1723 (1.4%)	54.9%	374 (14.4%)	1699
r5	544	1986 (1.0%)	17.9%	608 (10.5%)	1966
Avg		0.6%	15.7%	9.6%	

complexity of merging and pruning very high. In contrast, by using the transitive-closure pruning rule, our vawBuf algorithm can easily run through all benchmarks and for the largest benchmark $r5$, the runtime is about 3 min. This significant runtime speedup is achieved because the transitive-closure pruning rule as discussed in Section II enforces a relatively strict ordering between solutions, thus enabling an efficient implementation for both merging and pruning. Compared to the runtime for the conventional detBuf with projected mean plus 3-sigma values, we find that our vawBuf algorithm runs slower than detBuf. But this is expected because of the additional computation needed to handle correlated process variations.

We compare the solution quality between detBuf and vawBuf algorithms in Table II. Monte Carlo simulation is used as a golden test to obtain the delay distribution at the root for both cases. We employ the 3-sigma delay in the distribution as a figure-of-merit to compare the results. We further define the 3-sigma delay $(D_{3\sigma})$ of vawBuf as 100% timing yield point and use it to find the timing yield for the deterministic design. The difference between these two indicates the potential timing yield loss. We observe that compared to the deterministic buffering, our variation aware buffer insertion improves, on average, the 3-sigma timing by 0.6%, and the parametric timing yield by 15.7%, respectively. This highlights the importance of developing efficient algorithms for IC designs to actively attack process variation effects.

Interestingly, we observe that, for some relatively small benchmarks, the improvement for 3-sigma delay and yield is almost negligible, while for some large benchmarks, the improvement is quite significant. These observations to some degree agree with what has been reported in [13] for infinity long two-pin nets. There is a need, however, to look into the theoretical explanation to the observations.

We also report the number of buffers inserted for both algorithms in Table II. We see that our variation aware buffering algorithm tends to put more buffers into the design to combat the correlated process variations than the deterministic design.

V. CONCLUSION AND DISCUSSION

We have proposed a provably transitive-closure ordering rule and applied it to buffer insertion considering process variations. We conclude that process variation must be considered to achieve high parametric timing yield. This ordering rule is based on normal distribution assumption, and we plan to extend it to other distributions (such as exponential distribution) in the future. We envision that the transitive-closure ordering rule may be applicable to other design automation algorithms in the presence of process variations.

ACKNOWLEDGMENT

The authors would like to thank L. Cheng at the University of California, Los Angeles, for his helpful discussion and suggestions on this paper.

REFERENCES

[1] H. Chang and S. S. Sapatnekar, “Statistical timing analysis considering spatial correlations using a single PERT-like traversal,” in *Proc. Int. Conf. Comput.-Aided Des.*, Nov. 2003, pp. 621–625.
[2] C. Visweswariah, K. Ravindran, K. Kalafala, S. Walker, and S. Narayan, “First-order incremental block-based statistical timing analysis,” in *Proc. Des. Autom. Conf.*, Jun. 2004, pp. 331–336.
[3] M. Mani, A. Devgan, and M. Orshansky, “An efficient algorithm for statistical minimization of total power under timing yield constraints,” in *Proc. Des. Autom. Conf.*, Jun. 2005, pp. 309–314.
[4] V. Khandelwal, A. Davoodi, A. Nanavati, and A. Srivastava, “A probabilistic approach to buffer insertion,” in *Proc. Int. Conf. Comput.-Aided Des.*, Nov. 2003, pp. 560–567.
[5] J. Xiong, K. Tam, and L. He, “Buffer insertion considering process variation,” in *Proc. Des. Autom. Test Eur.*, 2005, pp. 970–975.
[6] A. Davoodi and A. Srivastava, “Variability-driven buffer insertion considering correlations,” in *Proc. IEEE Int. Conf. Custom Integr. Circuits*, Oct. 2005, pp. 425–430.
[7] ——, “Probabilistic evaluation of solutions in variability-driven optimization,” in *Proc. Int. Symp. Phys. Des.*, Apr. 2006, pp. 17–24.
[8] M. Cain, “The moment-generating function of the minimum of bivariate normal random variables,” *Amer. Stat.*, vol. 48, no. 2, pp. 124–125, May 1994.
[9] L. P. P. P. van Ginneken, “Buffer placement in distributed RC-tree networks for minimal Elmore delay,” in *Proc. IEEE Int. Symp. Circuits Syst.*, 1990, pp. 865–868.
[10] L. Zhang, W. Chen, Y. Hu, J. A. Gubner, and C. C.-P. Chen, “Correlation-preserved non-Gaussian statistical timing analysis with quadratic timing model,” in *Proc. Des. Autom. Conf.*, Jun. 2005, pp. 83–88.
[11] Z. Li and W. Shi, “An o(bn^2) time algorithm for optimal buffer insertion with b buffer types,” in *Proc. Des. Autom. Test Eur.*, Mar. 2005, pp. 1324–1329.
[12] W. Shi and Z. Li, “An o(nlogn) time algorithm for optimal buffer insertion,” in *Proc. Des. Autom. Conf.*, Jun. 2003, pp. 580–585.
[13] L. Deng and M. D. Wong, “Buffer insertion under process variations for delay minimization,” in *Proc. Int. Conf. Comput.-Aided Des.*, 2005, pp. 317–321.