Departmental Technical Reports (CS)   Department of Computer Science

1-1-2006

Static Timing Analysis Based on Partial Probabilistic Description of Delay Uncertainty

Wei-Shen Wang

Vladik Kreinovich
University of Texas at El Paso, vladik@utep.edu

Michael Orshansky

Static Timing Analysis Based on Partial and Distribution-Free Probabilistic Descriptions of Delay Uncertainty

Abstract

The existing approaches to timing analysis under uncertainty are based on fundamentally restrictive assumptions. Statistical STA techniques assume that the full probabilistic distribution of timing uncertainty is available. In reality, the complete probabilistic distribution information is often unavailable. The existing alternative of treating uncertainty as interval-based, or affine, is also limited since it cannot handle probabilistic information in principle. In this paper, a fundamentally new paradigm for timing uncertainty description is proposed as a way to consistently and rigorously handle partially available descriptions of timing uncertainty. The paradigm is based on a formal theory of interval probabilistic models that permit handling parameter uncertainty that is described in a distribution-free mode - just via the range, the mean, and the variance.

This permits effectively handling multiple real-life challenges, including imprecise and limited information about the distributions of process parameters, parameters coming from different populations, and the sources of uncertainty that are too difficult to handle via full probabilistic measures (e.g. on-chip supply voltage variation). Specifically, analytical techniques for bounding the distributions of probabilistic interval variables are proposed. Also, a provably correct strategy for fast Monte Carlo simulation based on probabilistic interval variables is introduced. A path-based timing algorithm implementing the novel modeling paradigm, as well as handling the traditional variability descriptions, has been developed. The results indicate the proposed techniques can improve the upper bound of the 95th-percentile circuit delay, on average, by 9.2% and 4.8% across the ISCAS'85 benchmark circuits, compared to the worst-case timing analysis that uses only the interval information of the partially specified parameters.

1. Need for New Models of Uncertainty: Probabilistic Interval Analysis

The area of statistical timing analysis (SSTA) has recently made substantial progress in terms of algorithmic and modeling advances. Efficient block-based and incremental computation techniques based on the first-order delay model are now well developed [1][2]. Extensions of the basic framework of SSTA to higher-order models have been recently investigated to capture non-linear effects and non-Gaussian process parameter distributions [3][4][5]. Statistical delay computation for interconnect based on affine interval arithmetic has been studied in [6]. These developments in the theory of SSTA came in response to the increased variability in the process parameters, the inadequacy of the corner models, and the need to use explicit probabilistic descriptions of key process parameters.

The fundamental assumption behind all of the above techniques is that the probabilistic descriptions are readily available. In all the algorithms for SSTA [1-5], the complete knowledge about the distributions of process and environmental parameters is given, e.g. it is assumed that the process parameters are normally distributed, with the known mean and variance. Then, first-order models link delay variability with process parameters, allowing delay to be normally distributed as well [1][2]. If linear delay models are not

sufficiently accurate, higher-order models can be used, at the cost of the resulting non-Gaussian distribution of delay. The non-Gaussianity of process parameters or timing can be handled by numerical processing leading to a substantial (3-10X) increase in the run-time of the algorithm [4].

In this paper we argue that in a practical setting of cutting-edge IC design the full probabilistic information about parameter uncertainty is not available. The process characterization data is often incomplete and of limited nature, especially at the ramp-up phase of the industrial manufacturing. With limited number of measurements and characterization lots, there may be a large uncertainty in the statistic metrics (the mean and the variance) of the process parameters. Some sources of on-chip uncertainty cannot be described probabilistically: supply voltage (V_{dd}), temperature, and systematic variation sources with the unit of repeatability larger than a single chip (e.g. aberration-caused L_{gate} variation).

Interval and affine methods, which tremendously improve on the conservatism of the traditional interval techniques, can be used [6]. However, in many instances, *some but not full* probabilistic information is available. For example, variation of supply voltage in *time* depends on the input vectors applied to the chip. Because of the difficulty of performing temporal input-dependent analysis, the uncertainty about supply voltage is most typically represented by the range information [7], however, the mean and, possibly, the variance of the distribution can be estimated more easily. For example, the supply voltage may vary between 90-100% of the nominal value, with the mean equal to 97% of the nominal value. The distribution is unknown because its characterization is too expensive [8]. Statistical STA cannot meaningfully handle such a realistic scenario. The affine methods are fundamentally non-probabilistic and their extensions to handling statistics are heuristic in nature [6].

Thus, in addition to the existing techniques, a new way of treating uncertain variables with partial probabilistic information is needed to enable practical design under uncertainty. This paper develops a solution of timing analysis under uncertainty based on the principles of probabilistic interval models. These models have been developed over the last decade in the field of robust statistics, reliable computing, and computer science [9]. They are based on the generalization of classical random variables to variables described by families of distributions.

Conceptually, the most general description of an uncertain variable is an interval, e.g. $x \in [\underline{x}, \overline{x}]$. Such descriptions form the basis of interval arithmetic and its enhancement in terms of affine arithmetic [10][11]. An interval description does not permit making statements about which values of the variable are more likely. Thus, if in addition to the range, the statistic metrics, such as mean and variance, are known the interval methods are incapable of utilizing this additional information in computing the arithmetic operations (+, -, *, /, max, min). *Probabilistic interval analysis* is a natural synergy of pure interval arithmetic and probabilistic analysis. It permits the use of partial statistic information (e.g. range, mean, and variance) to quantify the likelihood of the variable in the range. The estimates are guaranteed to be conservative regardless of the precise form of the distribution. For the fully specified random variable (e.g. Gaussian) the most general representation is its cumulative distribution function (*cdf*) [12]. For a partially-specified

random variable, the most general representation is a set of cumulative distribution functions, which can be represented as bounds on the *cdf*, forming a so-called probability box.

Following the above philosophy, this paper develops timing analysis techniques that produce reliable timing estimates even if the characterization data is incomplete. The essential contribution of this paper is in handling *incomplete* and *imprecise* uncertainty description. Compared to affine methods, the developed techniques can handle both the interval and probabilistic descriptions consistently and formally. The paper describes in detail how the probability boxes can be computed effectively. Importantly, the proposed techniques are compatible with the existing SSTA tools and can handle both full and partial probabilistic descriptions simultaneously.

This paper is organized as follows. Section 2 describes the paradigm of modeling non-probabilistic uncertainty based on probabilistic interval analysis, which enables us to use partial statistic metrics for timing analysis. The computation of path delay due to Gaussian variables and probabilistic interval variables is derived. Besides, statistical techniques of robustly estimating circuit delay distribution are proposed. The experimental results are presented in Section 3.

2. Timing Analysis with Partial Probabilistic Information

In this section, an application of the new probabilistic interval techniques to timing analysis is introduced. First, the construction of the path-delay probability box is described. Second, the bound of the circuit delay distribution is constructed. Finally, a method to combine the results of the traditional SSTA with the above derivations is described.

2.1 Path Delay Computation

The timing model used in this work is based on the additive delay model containing both the uncertainty due to classical random variables and the newly introduced probabilistic interval variables. The probabilistic interval variables (as opposed to random variables) are variables for which only partial statistic metrics, mean and variance, are available in addition to the known range, or interval. The delay model can be expressed as:

$$d_i = \mu_i + \sum_{j=1}^{n} a_{i,j} \Delta x_{i,j} + \sum_{k=1}^{m} b_{i,k} \Delta y_{i,k} \qquad (1)$$

where μ_i is the mean value of the gate delay, $\Delta x_{i,j}$ is a zero-mean Gaussian random variable, and $\Delta y_{i,k}$ is a zero-mean probabilistic interval variable. The coefficients $a_{i,j}$ and $b_{i,k}$ are the sensitivities of gate delays, which are the first-order derivatives of gate delays with respect to the variables. Note that this delay model can be easily transformed into an affine arithmetic representation if variables are scaled such that the variables are limited within [-1, 1].

A concise representation of the gate delay model can be obtained by resorting to the matrix form:

$$d_i = \mu_i + A_i^T X_i + B_i^T Y_i \qquad (2)$$

where the matrices $A_i = [a_{i,1} \cdots a_{i,n}]^T$, $B_i = [b_{i,1} \cdots b_{i,m}]^T$, $X_i = [\Delta x_{i,1} \cdots \Delta x_{i,n}]^T$, and $Y_i = [\Delta y_{i,1} \cdots \Delta y_{i,m}]^T$.

The variation of parameters can be further decomposed into the linear sum of die-to-die (X_{dd}, Y_{dd}) and independent within-die components ($X_{i,wd}$, $Y_{i,wd}$):

$$d_i = \mu_i + A_i^T X_{i,wd} + A_i^T X_{dd} + B_i^T Y_{i,wd} + B_i^T Y_{dd} \qquad (3)$$

The path delay of a path P_j can be represented by:

$$D^j = \sum_{i \in P_j} (\mu_i + A_i^T X_{i,wd} + A_i^T X_{dd} + B_i^T Y_{i,wd} + B_i^T Y_{dd})$$
$$= \sum_{i \in P_j} \mu_i + \sum_{i \in P_j} g_i + \sum_{i \in P_j} u_i \qquad (4)$$

where $g_i = A_i^T X_{i,wd} + A_i^T X_{dd}$, and $u_i = B_i^T Y_{i,wd} + B_i^T Y_{dd}$

It is convenient to separate the contributions of random delay uncertainty (D_R) and probabilistic interval uncertainty (D_{PI}):

$D_R^j = \sum_{i \in p_j} g_i$ and $D_{PI}^j = \sum_{i \in P_j} \mu_i + \sum_{i \in P_j} u_i$. Computing the path delay

distribution when the gate delays are normal random variables is straightforward. Therefore, we focus on the *delay variation resulting from probabilistic interval variables* i.e. D_{PI}^j. The range of the gate delay variation, u_i, is:

$$u_i \in \left[\sum_{k=1}^{m} b_{i,k} \underline{\Delta y_{i,k}}, \sum_{k=1}^{m} b_{i,k} \overline{\Delta y_{i,k}} \right] \qquad (5)$$

where $\underline{\Delta y_{i,k}}$ and $\overline{\Delta y_{i,k}}$ are the lower and upper bound of $\Delta y_{i,k}$. Then we can compute the range of D_{PI}^j.

Because the mean values of probabilistic interval variables are zero, the mean of the path delay is:

$$E[D_{PI}^j] = \sum_{i \in P_j} \mu_i \qquad (6)$$

The variance of the path delay can be computed by:

$$Var\left\{D_{PI}^j\right\} = \sum_{i \in P_j} B_i^T \Sigma_{i,wd} B_i + \left(\sum_{i \in P_j} B_i^T\right) \Sigma_{dd} \left(\sum_{i \in P_j} B_i\right) \qquad (7)$$

where $\Sigma_{i,wd}$ and Σ_{dd} are the covariance matrices of $Y_{i,wd}$ and Y_{dd}, respectively. Since different kinds of parameters are uncorrelated, the covariance matrices are actually diagonal matrices, with the diagonal elements equal to the variance of variables.

While the ultimate objective of the paper is to derive the circuit delay distribution, being able to describe individual path delay distributions is also essential. Now that the range, the mean and the variance of D_{PI}^j are known, the challenge is to compute the probability box that contains the family of distributions satisfying the partial statistical information that is available. Actually, the computation of the probability bound can be formulated as an optimization problem:

Let $F_i : \Re \to [0,1]$ ($1 \le i \le n$) be a possible cumulative distribution function of a random variable X, and F_i satisfies the partial statistical information: $E[X] = \mu$, $Var[X] = \sigma^2$, and $X \in [\underline{X}, \overline{X}]$. The lower bound for the cumulative probability of X at a specific value x, can be computed by solving the optimization problem considering all possible F_i:

$$\max p \text{ s.t. } F_i(x) \ge p, \ 1 \le i \le n.$$

Similarly, the upper bound can be computed by:

$$\min p \text{ s.t. } F_i(x) \le p, \ 1 \le i \le n.$$

However, because we seek a fast analytical solution, we prefer to use an inequality, which is a sophisticated generalization of the Chebyshev inequality, and is known as Cantelli inequality [13]. This inequality applies when, in addition to the first two moments

of the variable, its support (range) is also known, resulting in a much tighter bound on the *cdf*. The upper bound of the cumulative probability of a random variable X is given by [14]:

$$P(X \le x) = 0 \qquad\qquad x < \underline{X}$$
$$P(X \le x) \le 1/(1+(\mu-x)^2/\sigma^2) \qquad \underline{X} \le x < \mu + \sigma^2/(\mu - \overline{X})$$
$$P(X \le x) \le 1 - (m^2 - my + s^2)/(1-y) \qquad \mu + \sigma^2/(\mu - \overline{X}) \le x \qquad (8)$$
$$\qquad\qquad \text{and } x < \mu + \sigma^2/(\mu - \underline{X})$$
$$P(X \le x) = 1 \qquad\qquad \mu + \sigma^2/(\mu - \underline{X}) \le x$$

where μ denotes the mean, σ^2 denotes the variance, $\underline{X}$ is the lower bound, $\overline{X}$ is the upper bound, $y = (x - \underline{X})/(\overline{X} - \underline{X})$, $m = (\mu - \underline{X})/(\overline{X} - \underline{X})$, and $s^2 = \sigma^2/(\overline{X} - \underline{X})^2$.

Similarly, the lower bound of the cumulative probability is:

$$P(X \le x) = 0 \qquad\qquad x < \mu + \sigma^2/(\mu - \overline{X})$$
$$P(X \le x) \ge 1 - (m(1+y) - s^2 - m^2)/y \quad \mu + \sigma^2/(\mu - \overline{X}) \le x$$
$$\qquad\qquad \text{and } x < \mu + \sigma^2/(\mu - \underline{X}) \qquad (9)$$
$$P(X \le x) \ge 1/(1 + \sigma^2/(x-\mu)^2) \qquad \mu + \sigma^2/(\mu - \underline{X}) \le x < \overline{X}$$
$$P(X \le x) = 1 \qquad\qquad \overline{X} \le x$$

Thus, expressions (8) and (9) can be used to compute the bound for the path delay cumulative probability. An example of applying this set of inequalities is shown in Figure 3(a).

The same analytical structure can be used when the mean and variance are known only with certain accuracy [15]. First, the maximum of the variance should be used in the generalized Chebyshev inequality because it primarily determines the span of the *cdf*. Second, the upper bound of the mean should be used when computing the lower (right-side) bound of the probability using (9), because it leads to the *worst* lower bound of the probability. Similarly, the lower bound of the mean should be used in (8).

Having computed the distribution of path delay variation due to probabilistic interval variables, now we combine it with the delay variation resulting from Gaussian variables. Since parameters of different categories are independent, it means that the delay variations D_R^j and D_{PI}^j are independent, and the bound for the *cdf* of the sum can be computed by convolution:

$$CDF(D^j) = CDF(D_{PI}^j) \otimes f(D_R^j) \qquad (10)$$

where $f(D_R^j)$ is the probability density function of D_R^j. We use the lower and upper bounds of $CDF(D_{PI}^j)$ in convolution respectively, and then obtain the bounds of $CDF(D^j)$. Finally, we have the bound for the path delay distribution, which enables computing the bound of delay at any quantile.

2.2 Circuit Timing Computation

In this section, we develop techniques for efficient construction of probability boxes on the distribution of circuit delay, i.e. the maximum of all path delays. New techniques are proposed to perform this task efficiently and robustly. From (4), the bound of the circuit delay can be computed by:

$$D_{\max} = \max(D^1, ..., D^N)$$
$$= \max\left(\sum_{i \in P_1}(\mu_i + g_i + u_i), ..., \sum_{i \in P_N}(\mu_i + g_i + u_i)\right) \quad (11)$$
$$\le \max(D_R^1, ..., D_R^N) + \max(D_{PI}^1, ..., D_{PI}^N)$$

Let $D_{R\max} = \max(D_R^1, ..., D_R^N)$ be the term due to random probabilistic variability, and the second term

$D_{PI\max} = \max(D_{PI}^1, ..., D_{PI}^N)$ be the term due to interval-probabilistic variability. In deriving the probability box for $D_{\max}$, we adopt a strategy in which the sources of uncertainty described probabilistically are separated from interval probabilistic uncertainty. The distribution of $D_{R\max}$ can be computed by the statistical timing analysis algorithm based on the first-order delay models (e.g. [1][2][17][18]). Therefore, in the remainder, we concentrate on the computation of $D_{PI\max}$. The two terms are then combined to generate the bounds on the full distribution of circuit delay.

In constructing the probability box for the circuit delay distribution, ideally, we would like to use analytical means as was done in Section 2.1. The generalized Chebyshev inequality can be used to find the bounds on the distribution of $D_{PI\max}$, once the mean, the variance, and the range are known. However, in general functions of probabilistic interval variables, $f(u_1..u_N)$, finding the bounds on the variance is NP-hard [16]. We show below that for *convex* functions the exact bound on the variance can be computed. Let us first establish the convexity of the term $D_{PI\max}$. The path delay $D_{PI}^j = \sum u_i, i \in P_j$ is a linear and thus convex function of u_i. The circuit delay is given by $D_{PI\max} = \max(D_{PI}^1, ..., D_{PI}^N)$ which is also a convex function of probabilistic interval variables [19]. Convexity is essential to our efficient analysis strategy, since as the theorem below shows determining the probability bound and moments of distributions of convex functions is much easier.

Our strategy is essentially based on the development of the robust (guaranteed) approach to Monte Carlo sampling from an unknown distribution [21]. The Monte-Carlo simulation is a widely-used technique to solve complex numerical problems [22]. It can be used as a powerful tool for estimating the timing performance of integrated circuits when the distributions are known [23][24]. Without the full distributional knowledge of the parameters, a possible way to perform the simulation is to heuristically generate a variety of distributions that correspond to the given partial information. However, this method is not mathematically robust because it is impossible to enumerate all possible distributions. Besides, the high run time accounting for numerous distributions prevents this method from practical use. We show that for convex functions the *robust Monte Carlo* simulation can be rigorously and efficiently performed. Compared to the traditional approach to Monte-Carlo simulation, the selection of distribution is *justified* in our simulation strategy; only distributions that cause the extreme value of the target function need to be considered. Therefore, this selective strategy is guaranteed to produce a bounding distribution, and achieves high efficiency in terms of the run time. Theorem 1 effectively defines the algorithm for such robust Monte Carlo [21].

Theorem 1. Let $\{v_1, ..., v_M\}$ be a set of *independent* random variables, where $v_i \in [\underline{v_i}, \overline{v_i}]$, and $E[v_i] = E_i$ for i=1 to M. Let $f(v_1, ..., v_M)$ be a non-negative convex function of the random variable v_i, for i=1 to M. The probabilistic bound of $f(v_1, ..., v_M)$, at a confidence level α, is defined as:

$$D^\alpha = \min\{D \in \mathbb{R} : P(f(v_1, ..., v_M) \le D) \ge \alpha\} \qquad (12)$$

Assume D^α decreases if any interval $[\underline{v_i}, \overline{v_i}]$ is narrowed down.

Then, among all possible *cdfs* of $\{v_i : i = 1..M\}$ that correspond to the partial statistical information of the range and the mean, the bound D^α achieves the maximum value when each random variable v_i follows the 2-point distribution,

$$P(v_i = \underline{v_i}) = \underline{p_i}$$
$$P(v_i = \overline{v_i}) = \overline{p_i}$$
(13)

where $\underline{p_i} = \dfrac{\overline{v_i} - E_i}{\overline{v_i} - \underline{v_i}}$, and $\overline{p_i} = \dfrac{E_i - \underline{v_i}}{\overline{v_i} - \underline{v_i}}$.

Effectively, Theorem 1 reduces the number of possible distributions that must be considered in order to find the bounding distribution, which will result in a sought probability box for the function of probabilistic interval variables. Thus, Theorem 1 permits an algorithm for the robust Monte Carlo simulation, as shown in Figure 1.

The above approach, the robust Monte Carlo simulation, will suffer from the common problems of Monte Carlo - the slow decrease of the estimation error, especially, at high percentiles. To address this concern, we have developed a fast *hybrid* approach, *the fast robust Monte Carlo simulation*, in which robust Monte Carlo is used to get a quick estimate of the moments (a much faster computation) and then analytical techniques are used for establishing bounds. The justification of the technique is based on the corollary to Theorem 1.

Corollary. The kth moment of the function, $E[y^k]$, where $y = f(v_1, ..., v_M)$, achieves the maximum value when each random variable v_i follows the above 2-point distribution. Furthermore, $E[y^k]$ achieves the minimum when $P(v_i = E_i) = 1$.

Therefore, using the above sampling procedure also guarantees that the bounds of $E[f(v_1, ..., v_M)]$ are accurately estimated.

In the fast Monte Carlo simulation, a limited number of random samples are drawn using the algorithm following Theorem 1. The corollary guarantees that we will get an accurate estimate of the range of the mean circuit delay. As for the variance of the circuit delay, it can also be bounded by the sample variance because the 2-point distribution in (13) results in the maximum variance of gate delays thus maximizes the variance of path delays and the circuit delay. Therefore, the generalized Chebyshev inequality can be then used to compute the bound of the distribution analytically.

Figure 2 illustrates the algorithm of the fast Monte Carlo simulation. This proposed strategy estimates the upper bound of sample mean and sample variance with only a limited number of runs. In practice, a few hundred runs are sufficient to generate an estimate with reasonable accuracy. This can be verified by considering the standard error of the sample mean and the confidence level of the true mean i.e. the mean of the population. From [26], the 99% confidence interval of the true mean (μ) for a variable X is $\overline{X} \pm 2.575\, \sigma_X / \sqrt{N}$, where $\overline{X}$ is the sample mean, σ_X is the true standard deviation, and N is the number of samples. For example, consider a circuit with extremely large span in the delay domain: the 3σ value of circuit delay is 45% of the mean. Then we estimate the confidence level:

$$P(|\overline{X} - \mu| \leq 2.575 \cdot 0.15\mu / \sqrt{N}) = 0.99 .$$

The error of the sample mean for $N = 500$ is less than 1.7% with probability equal to 0.99, which has a very limited impact on the result of using the generalized Chebyshev inequality. Thus, the accuracy of Monte Carlo for such a sample size is acceptable for our analysis.

Once the lower bound on the distribution of $D_{PI\max}$ is generated, the overall circuit delay distribution $D_{\max}$ can be obtained by combining $D_{PI\max}$ and $D_{R\max}$. Since these two components of delay variation are independent, the distribution of the sum can be computed by convolution, similar to (10). The lower bounds of the

cdf (i.e. the upper bound of the delay) are used in the convolution because it is a more important metric for circuit timing.

3. Experiments

The algorithms for timing analysis using partial description of uncertainty described in Section 2 have been implemented in C++, and have been tested on a set of combinational ISCAS'85 benchmark circuits. Variability of process parameters (L, V_{th}, and T_{ox}) and the environmental fluctuation (V_{dd}) are taken into account. The 3σ values for process parameters are set at 20% of the mean, including 50% die-to-die variations. The standard deviation of V_{dd} is 4% of the maximum, and the range is 16% of the maximum. In the experiments, V_{th}, T_{ox} and V_{dd} are modeled as probabilistic interval variables. Sensitivities of parameters are from SPICE simulations for a cell library of BPTM 0.13um technology [28].

The proposed timing analysis algorithms separately handle the contributions of the random probabilistic uncertainty and the interval probabilistic uncertainty. Thus, the comparison of our algorithms and the worst-case timing analysis i.e. only using the range (interval) of the interval uncertainty, should be done in two phases. We first compare the bounds of $D_{PI}{}^j$ computed by the proposed algorithm and the worst-case timing analysis, then compare the bound of the total delay, which is the sum of $D_{PI}{}^j$ and $D_R{}^j$. Note that the sum of the bound from the worst-case timing analysis for interval uncertainty and $D_R{}^j$ can be computed by simply shifting the *cdf* of $D_R{}^j$ by the worst-case delay value. A similar comparison is also made for the bounds on circuit delay distribution.

Figure 3(a) illustrates the importance of probabilistic interval analysis in path delay analysis. The upper bound of the 95th-percentile path delay ($D_{PI}{}^j$) from the proposed algorithm for the

---

for $i = 1..N$
    Generate a sample for each die-to-die parameter.
    for each gate
        Generate a sample for each within-die parameter.
        Compute gate delay.
    end
    Use static timing analysis to compute the circuit delay, D_i.
end

Sort the circuit delay D_i, $i = 1..N$ such that $D_{(1)} \leq ... \leq D_{(N)}$.

The α-percentile delay can be estimated by $D_{(k)}$, where $k = \min\left\{ n \in \mathbb{N} : n/N \geq \alpha \right\}$.

Figure 1. Algorithm of the robust Monte Carlo simulation

Compute the mean and the variance of samples:

$$\overline{D} = \sum_{i=1}^{N} D_i \Big/ N$$

$$s_D^2 = \sum_{i=1}^{N} (D_i - D_{avg})^2 \Big/ N - 1 \, .$$

With $\overline{D}$, s_D^2, and the range of the circuit delay, use (9) to compute the lower bound of the *cdf*.

Figure 2. Algorithm of the fast robust Monte Carlo simulation

critical path of circuit c6288 is only 8.4% over the mean path delay, while the worst-case timing estimate is 16.2% over the mean. Therefore, the proposed path timing analysis algorithm reduces the worst-case timing estimate by 6.7%. Similarly, the 95th-percentile total path delay ($D_R^j + D_{PI}^j$) is 20.2% over the mean for the proposed algorithm, which is superior to the worst-case delay (32.1% over the mean) in Figure 3(b). Thus, the proposed strategy improves the worst-case estimate by 9.0% for the overall path delay at the 95th percentile.

For circuit delay distribution, the proposed statistical techniques have been run on a Sun workstation with 1280 MHz CPU and 8GB memory. The robust Monte Carlo simulation (RMC) was run for 10,000 iterations; the run time ranges from 135 to 1467 seconds. Across the benchmark circuits RMC reduces the worst-case circuit delay due to interval uncertainty ($D_{PI\max}$) by 5.8-7.6% at the 95th percentile, as shown in Table I.

We also ran the fast robust Monte Carlo simulation (FRMC) to estimate the sample mean and the variance using 1,000 samples, and then analytically computed the lower bound of the cumulative probability. The run time of the fast robust Monte Carlo ranges from 12 to 114 seconds. Figure 4 shows the circuit delay variation due to probabilistic interval variables of circuit c7552, from the proposed statistical strategies and the worst-case timing analysis. The upper bound of the normalized circuit delay at the 95th percentile from RMC is 1.08, which is a better bound than the worst-case delay (1.16). Besides, FRMC is able to provide a superior bound to the worst-case delay at lower than the 87th percentile.

For the total circuit delay ($D_{\max}$), RMC and FRMC improve the estimates from the worst-case timing analysis by 9.2% and 4.8% across the benchmark circuits, for the 95th percentile delay. Table II shows the upper bound of the total circuit delay at high percentiles (90th and 95th percentiles) for both statistical techniques, and the worst-case timing analysis. Figure 5 shows an example of the total circuit delay for the circuit c7552, in which the RMC and FRMC reduce the worst-case delay estimate by 8.7% and 4.5% at the 95th percentile, respectively. Indeed, the joint use of SSTA and our statistical techniques for probabilistic interval variables is a promising synergy, and it can be easily extended to incorporate more circuit parameters, to fully assess the impact on timing performance.

Another important feature of the proposed techniques is the capability of handling skewed distributions. Some environmental parameters are not symmetrically distributed (e.g. V_{dd}); however, the normal assumption implies the distribution is symmetrical to the mean, which may cause inaccurate estimation of the circuit delay. Figure 6(a) compares path delay distributions of two cases with the same interval and variance of V_{dd} uncertainty: the right-skewed V_{dd} uncertainty and the symmetrical case. Because the voltage drop increases delay, the right-skewed V_{dd} uncertainty decreases the upper bound of delays, compared to the center-meaned V_{dd} distribution. From Figure 6(b), the similar trend can be also observed in the distribution of the total circuit delay. Thus, our timing analysis algorithm can be used to handle asymmetrical distributions (e.g. non-Gaussian), and provide a more accurate timing estimate.

As for the quality of the bound for cumulative probability, the robust Monte Carlo is superior to the fast robust Monte Carlo; however, the fast robust Monte Carlo simulation achieves higher efficiency in terms of the run time because it needs fewer runs, which allows us to have a robust delay estimate in a reasonable amount of time. Provided that the quality of the bound is important, we can resort to the robust Monte Carlo simulation.

4. Conclusions

In this paper, we propose a set of statistical techniques for estimating the path and circuit delay distributions. Given partial statistic metrics of the uncertainty, the proposed algorithm is able to analytically compute the bounds of the path delay. A robust Monte Carlo simulation technique and a hybrid approach are proposed to assess the impact of the uncertainty, and estimate the upper bound of the circuit delay. With justified selection of the distribution used in the simulation, the proposed techniques can efficiently provide a guaranteed bound of the circuit delay distribution.
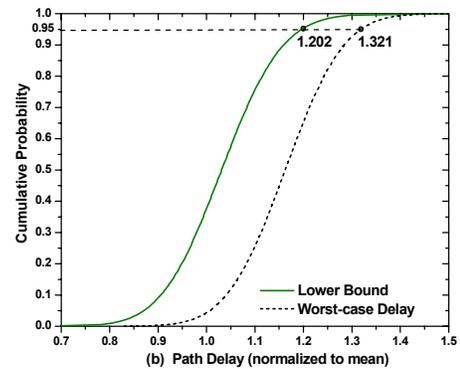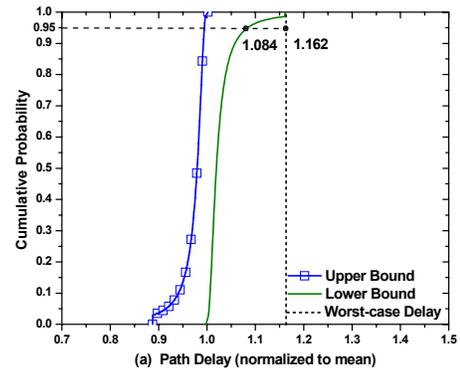

Figure 3. The path delay analysis algorithm improves the worst-case path delay by 9.0% for the critical path of circuit c6288 at the 95th percentile. a) Delay due to probabilistic interval variables; b) Total path delay.
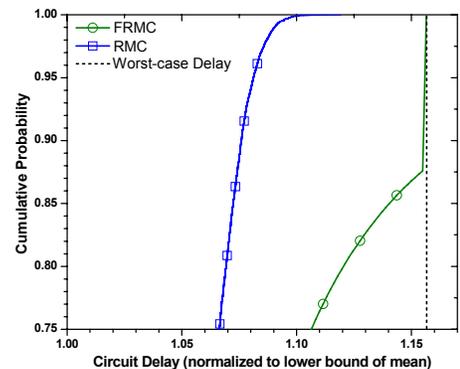

Figure 4. Upper bounds for circuit delay due to probabilistic interval variables for circuit c7552. The fast Monte Carlo simulation provides a bound superior to the worst-cast timing estimate at lower than the 87th percentile.
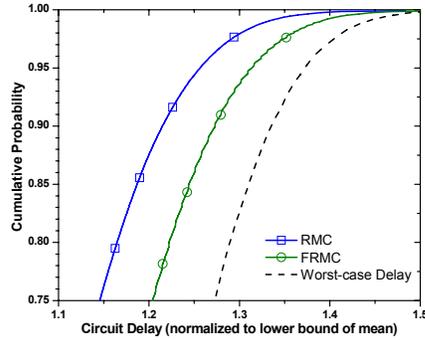
Figure 5. Upper bounds for the overall circuit delay of c7552. RMC improves the worst-case delay estimate by 8%.
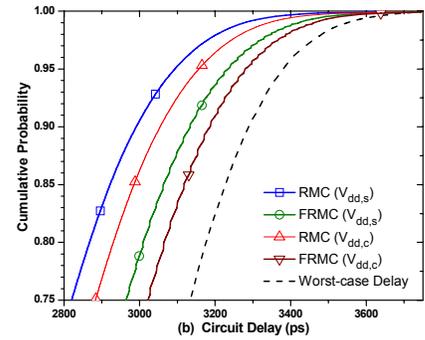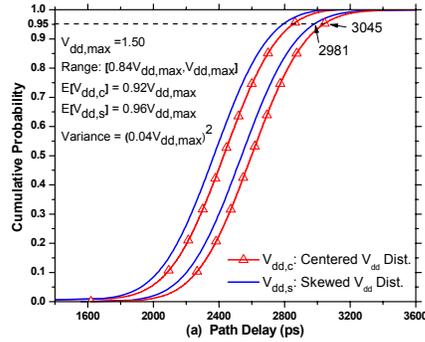

Figure 6. The right-skewed V_{dd} distribution decreases bounds of (a) path delay; and (b) circuit delay of the center-meaned V_{dd} distribution.

References

[1] C. Visweswariah *et al*, "First-order incremental block-based statistical timing analysis," *Proc. of DAC*, 2004.

[2] H. Chang *et al*, "Statistical timing analysis considering spatial correlations using a single PERT-like traversal," *Proc. of ICCAD*, 2003.

[3] Y. Zhan *et al*, "Correlation-aware statistical timing analysis with non-Gaussian delay distributions," *Proc. of DAC*, 2005.

[4] H. Chang *et al*, "Parameterized block-based statistical timing analysis with non-Gaussian parameters and nonlinear delay functions," *Proc. of DAC*, 2005.

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

[6] J. D. Ma *et al*, "Interval-valued reduced order statistical interconnect modeling," *Proc. of ICCAD*, 2004.

[7] D. Ernst *et al*, "Razor: circuit-level correction of timing errors for low-power operation," *IEEE Micro*, 24(6):10-20, November 2004.

[8] D. Kouroussis *et al*, "Incremental partitioning-based vectorless power grid verification," *Proc. of ICCAD*, 2005.

[9] V. P. Kouznetsov, *Interval Statistical Models*, Radio i Svyaz, Moscow, 1991 (In Russian).

[10] R. E. Moore, *Interval Analysis*, Prentice-Hall, 1966.

[11] J. Stolfi *et al*, "An introduction to affine arithmetic," *TEMA Tend. Mat. Apl. Computing*, 4, No. 3 (2003), 297-312.

[12] W. Feller, *An Introduction to Probability Theory and Its Applications*, Wiley and Sons, 3rd Edition, 1968.

[13] H. Godwin, *Inequalities on Distribution Functions*, Hafner, 1964.

[14] S. Ferson *et al*, "Constructing probability boxes and Dempster-Shafer structures," Sandia Report, 2002.

[15] S. Ferson, *RAMAS Risk Calc 4.0 Software: Risk Assessment with Uncertain Numbers*, CRC Press, 2002.

[16] S. Ferson *et al*, "Computing variance for interval data is NP-hard," *ACM SIGACT News*, Vol. 23(2), pp. 108-118, June 2002.

[17] A. Agarwal *et al*, "Path-based statistical timing analysis considering inter- and intra-die correlations," *TAU*, 2002.

[18] M. Orshansky *et al*, "Fast statistical timing analysis handling arbitrary delay correlations," *Proc. of DAC*, 2004.

[19] S. Boyd *et al*, *Convex Optimization*, Cambridge University Press, 2004.

[20] R. Barmish *et al*, "Monte Carlo analysis of resistive networks without apriori probability distributions," *Proc. of ISCAS*, 2000.

[21] Reference omitted.

[22] G. Fishman, *Monte Carlo: Concepts, Algorithms, and Applications*, Springer-Verlag, 1995.

[23] R. Hitchcock, "Timing verification and the timing analysis program," *Proc. of DAC*, 1982.

[24] H.-F. Jyu *et al*, "Statistical timing analysis of combinational logic circuits," *Trans. on VLSI Systems*, vol.1, (no.2), pp. 126-37, 1993.

[25] A. Lemke *et al*, "Analog circuit sizing based on formal methods using affine arithmetic," *Proc. of ICCAD*, 2002.

[26] J. Rice, *Mathematical Statistics and Data Analysis*, Wadsworth & Brooks, 1988.

[27] Y. Cao *et al*, "New paradigm of predictive MOSFET and interconnect modeling for early circuit design," *Proc. of CICC*, pp. 201-204, 2000.

[28] PTM: http://www.eas.asu.edu/~ptm.

Table I. Upper bounds for circuit delay due to probabilistic interval variables $(D_{PI\max})$. The robust Monte Carlo simulation reduces the worst-case estimates by 5.8-7.6%.

Circuit	Worst-case Delay (ps)	RMC (ps) (95th Percentile)	Reduction (%)
c880	2185	2031	7.05
c1355	2028	1910	5.82
c1908	2584	2400	7.12
c2670	2860	2650	7.34
c3540	3773	3495	7.37
c6288	16456	15201	7.63
c5315	3303	3071	7.02
c7552	2849	2663	6.53

Table II. Upper bounds for circuit delay at high percentiles and run time of the proposed techniques.

Circuit	Number of Gates	Robust Monte Carlo Simulation					Fast Robust Monte Carlo Simulation					Worst-case Delay	
		90th Percentile		95th Percentile		Run Time (s)	90th Percentile		95th Percentile		Run Time (s)	90th Percentile	95th Percentile
		Delay (ps)	Reduction (%)	Delay (ps)	Reduction (%)		Delay (ps)	Reduction (%)	Delay (ps)	Reduction (%)		Delay (ps)	Delay (ps)
c880	456	2268	10.18	2349	9.51	135	2383	5.62	2467	4.97	12	2525	2596
c1355	605	2159	9.02	2235	8.36	203	2264	4.59	2335	4.26	18	2373	2439
c1908	975	2691	9.88	2786	9.22	302	2820	5.56	2919	4.89	26	2986	3069
c2670	1544	2971	10.27	3077	9.63	442	3124	5.65	3232	5.08	38	3311	3405
c3540	1787	3890	10.27	4030	9.58	624	4097	5.49	4237	4.94	52	4335	4457
c6288	2448	16624	10.27	17173	9.60	1094	17547	5.28	18081	4.82	87	18526	18996
c5315	2600	3412	9.90	3532	9.27	984	3579	5.49	3703	4.88	79	3787	3893
c7552	3874	2992	9.25	3094	8.65	1467	3136	4.88	3236	4.46	114	3297	3387