

# Signal Probability Based Statistical Timing Analysis

Bao Liu

Computer Science and Engineering Department

University of California, San Diego

La Jolla, CA 92093-0114

Email: bliu@cs.ucsd.edu

## Abstract

We observe that Monte Carlo (SPICE) simulation provides the most accurate and trustable statistical timing analysis, while the existing SSTA method has completely ignored the effect of input statistics on chip timing performance, and provides either accurate estimate nor pessimistic bound of the actual chip timing performance statistics. We propose signal probability (i.e., the logic one occurrence probability for a signal) based statistical timing analysis for improved accuracy and reduced pessimism over the existing SSTA methods, and improved efficiency over Monte Carlo (SPICE) simulation. Our experimental results show that our proposed SPSTA computes mean (standard deviation) of signal arrival times within 6.2% (18.6%), while SSTA computes mean (standard deviation) of signal arrival times within 13.40% (64.3%) of Monte Carlo simulation results; SPSTA also provides signal probability estimation within 14.28% of Monte Carlo simulation results for the ISCAS'89 benchmark circuits.

## 1 Introduction

VLSI manufacturing process today observes increased variations on layout geometries and circuit performance. Limitations on manufacturing equipments include: lithographic issues, e.g., optical proximity, defocus, and lens aberration, which affect feature dimensions, e.g., wire width or transistor channel length; the chemical mechanical polishing (CMP) process which varies feature thickness for different local layout density; and dopant variations in chemical processes. On the other hand, aggressive VLSI designs induce increased variations on system performance. Integration of increased number of components in a single chip results in increased supply voltage drop and temperature variation; higher operating frequencies observe increased capacitive and inductive couplings on silicon surface and through substrate; aggressive performance optimization could result in a large number of near-critical paths with increased probability of timing failure.

VLSI timing analysis techniques have been developed to capture the increasingly significant timing performance variation. (1) Traditional minimum/maximum based timing analysis, which computes minimum and maximum delays separately, and verifies timing requirements between either minimum or maximum path delays, captures only die-to-die variations. (2) Corner based timing analysis, which computes minimum and maximum delays simultaneously, and allows timing verification between minimum and maximum path delays, captures intra-die variations. (3) Statistical static timing analysis (SSTA), which computes delay distribution for each pin (block-based) or path (path-based), provides "timing yield" or probability for a chip to meet timing requirements.

Block-based SSTA [1, 25] represents signal arrival time variation at each pin as a probability distribution function (pdf). As-

suming symmetry or normality of signal arrival time distributions, these probability distribution functions are computed based on simple formulas in a breadth-first netlist traversal, which is efficient, incremental, and suitable for optimization. Path-based SSTA [18, 19] provides more accurate statistical analysis on a set of near-critical paths, e.g., in corner-based or Monte Carlo analysis, signal arrival time at a pin could have different distributions in different paths, where correlations due to path-sharing can be better captured. Timing criticality probabilities and correlations of the near-critical paths are computed for signoff analysis.

Correlations come from (1) path-sharing in the presence of re-convergent fanouts, and (2) dependence on common variational parameters. Correlated pdf's can be propagated in conditional probabilities [1]. Correlated parameters can be broke down into uncorrelated random variables in principle component analysis (PCA) [14]. Layout geometrical variations are translated into performance variations by sensitivity-based[3], interval-valued [10], or matrix-perturbation-theory-based [9] analysis through interconnect model order reduction [17].

Including more sources of variation into consideration has significantly improved accuracy of statistical timing analysis. For example, a statistical gate delay is subject to significant variation when multiple inputs of the gate are switching at the same time [2]. Neglecting this multiple-input switching effect could underestimate the mean delay of a gate by up to 20% and overestimate the standard deviation of the delay of a gate by up to 26%. A signal switching at a neighboring cross-coupled net injects a noise waveform and affects the interconnect delay. It also alters the driver gate delay of the net due to effective load capacitance variation. Neglecting this crosstalk aggressor alignment effect could lead to dramatic differences in interconnect/gate delay means and standard deviations [6, 7].

Through these multiple-input switching and cross-coupling effects, as well as other effects such as supply voltage drop and inductance coupling, we observe that delay of a gate or an interconnect in a VLSI circuit is strongly affected by the occurrence and the timing of switching activities in other neighboring or distant nets. This strongly suggests that input statistics and circuit status have a significant effect on chip timing performance and performance variation. Existing SSTA approaches have completely ignored this effect, which need serious review.

Static timing analysis seeks to bound the statistical timing performance of a chip by finding a best case and a worst case. The current SSTA method includes process variation effects and finds a best case timing distribution and a worst case timing distribution (Fig. 1). Such a method has serious problems. (1) The purpose of statistical timing analysis is to reduce the pessimism in static timing analysis through improved accuracy. (2) The input statistics has the most significant effect on timing performance variation. The process variations are secondary effects on chip timing performance. (3) The effect of process variations on chip timing performance has to be computed for a given input vector, because for each input vec-

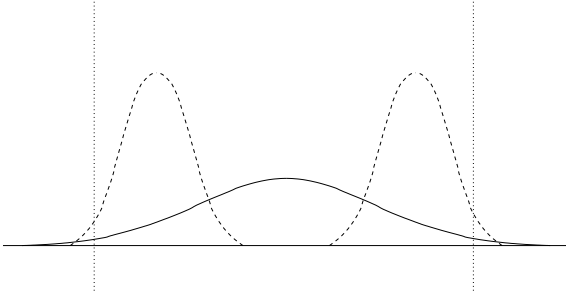


Figure 1: A VLSI timing performance distribution (solid curve) resulting from input statistics and process variations, which is captured by STA in two bounds (dotted lines), and by SSTA in two best/worst case distributions (dashed curves). STA and SSTA estimates are pessimistic if false paths are not excluded. STA bounds may represent the  $\pm 3\sigma$  points of SSTA distributions.

tor the effect of process variations is different. (4) SSTA computes the effect of process variations on chip timing performance based on an artificial input vector which may not exist. For example, the probability for two signals to arrive at about the same time to activate the crosstalk coupling effect cannot be accurately estimated in SSTA, it can only be assumed, e.g., that it always happens in worst case analysis. The resultant timing distribution deviates from and does not guarantee to bound the actual timing performance distribution (Fig. 1).

In real life, manufactured chips are tested dynamically, i.e., by given test vectors for a required fault coverage. Dynamic simulation gives more accurate results than static methods, especially for VLSI circuits in the presence of significant crosstalk coupling, supply voltage drop, and other DSM/nanometer effects. Such accuracy improvement is much needed for statistical timing analysis. As a result, we strongly suggest that statistical timing analysis cannot be static. Any accurate and meaningful statistical timing analysis has to be dynamic, by including the effect of input statistics.

Monte Carlo (SPICE) simulation provides the most accurate and trustable statistical timing analysis results. For efficiency improvement over Monte Carlo (SPICE) simulation, we look into the power estimation literature, where a spectrum of probabilistic techniques can be found, which compute signal transition occurrence probabilities in a circuit, e.g., based on signal probabilities [13], transition densities [12], probabilistic waveform simulation [15], or BDD based computation with potential correlation consideration [5].

In this paper, we introduce signal probability in probabilistic power estimation to statistical timing analysis, and propose signal probability based statistical timing analysis (SPSTA) for improved accuracy and reduced pessimism over the existing SSTA methods, and improved efficiency over Monte Carlo (SPICE) simulation. We propose to compute signal transition temporal occurrence probability (top) function, which is a generalization of signal arrival time probability density function and signal toggling rate, and replace the MAX operation in SSTA by a WEIGHTED SUM operation. We present two statistical timing analysis methods: moment and correlation computation based and polynomial of variational variable based. We summarize the advantages of SPSTA over SSTA. Our experimental results show that our proposed SPSTA computes mean (standard deviation) of signal arrival times within 6.2% (18.6%), while SSTA computes mean (standard deviation) of signal arrival times within 13.40% (64.3%) of Monte Carlo simulation results; SPSTA also provides signal probability estimation within 14.28% of Monte Carlo simulation results for the ISCAS'89 benchmark circuits.

The rest of the paper is organized as follows. We introduce statistical static timing analysis, and statistical power estimation as background in Section 2. We present signal probability based statis-

tical timing analysis, including its variables, operations, two abstraction methods, and summary of advantages in Section 3. We present our experimental results in Section 4, and conclude in Section 5 with ongoing research directions.

## 2 Background

### 2.1 Statistical Static Timing Analysis (SSTA)

SSTA takes into account process and environmental variations in VLSI designs, and represents timing properties of a circuit as statistical variables, e.g., in the form of probability density functions.<sup>1</sup> There are two basic operations in SSTA (Fig. 2).

#### 2.1.1 SUM

The output signal arrival time  $t_0$  of an interconnect is given by the summation of the input signal arrival time  $t_1$  and the delay  $d$  of the interconnect.

$$\begin{aligned} t_0 &= t_1 + d \\ cdf(t_0 = \tau) &= \int_{\tau_1 + \tau_2 \leq \tau} pdf(t_1 = \tau_1) pdf(d = \tau_2) d\tau_1 d\tau_2 \\ pdf(t_0 = \tau) &= \int_0^{\infty} pdf(t_1 = \tau_1) pdf(d = \tau - \tau_1) d\tau_1 \end{aligned} \quad (1)$$

For  $pdf(t_1)$  and  $pdf(d)$  in normal distributions,  $pdf(t_0)$  is also a normal distribution, which mean and standard deviation are given as follow.

$$\begin{aligned} \mu(t_0) &= \mu(t_1) + \mu(d) \\ \sigma^2(t_0) &= \sigma^2(t_1) + \sigma^2(d) + 2cov(t_1, d) \end{aligned} \quad (2)$$

#### 2.1.2 MIN/MAX

The MIN(MAX) operation is required for the output signal arrival time of a multiple-input gate, when the output signal arrival time  $t_0$  of a gate is given by the earliest/latest of input signal arrival times  $t_1$  and  $t_2$  (plus gate delay). Whether a MIN or a MAX operation needs to be taken depends on the logic of the gate and the direction of the input signal transitions. For example, the output rising(falling) signal arrival time of an AND gate is given by the MAX(MIN) of the rising(falling) input signal arrival times, and the output rising(falling) signal arrival time of an OR gate is given by the MIN(MAX) of the rising(falling) signal arrival times.

For a MAX operation, the output signal arrival time pdf and cdf are given as follow.

$$\begin{aligned} t_0 &= MAX(t_1, t_2) \\ cdf(t_0 = \tau) &= cdf(t_1 = \tau) cdf(t_2 = \tau) \\ pdf(t_0 = \tau) &= cdf(t_1 = \tau) pdf(t_2 = \tau) \\ &\quad + pdf(t_1 = \tau) cdf(t_2 = \tau) \end{aligned} \quad (3)$$

For  $pdf(t_1)$  and  $pdf(t_2)$  in normal distributions, the result of the MAX operation  $pdf(t_0)$  is not a normal distribution. Its mean and standard deviation are given as follow.

$$\begin{aligned} \mu(t_0) &= \mu(t_1)Q + \mu(t_2)(1-Q) + \theta P \\ \sigma^2(t_0) &= (\mu^2(t_1) + \sigma^2(t_1))Q + (\mu^2(t_2) + \sigma^2(t_2))(1-Q) \\ &\quad + (\mu(t_1) + \mu(t_2))\theta P - \mu^2(t_0) \end{aligned} \quad (4)$$

<sup>1</sup>Probability density functions extend to joint probability density functions in the presence of correlations, e.g., for the equations in this section.

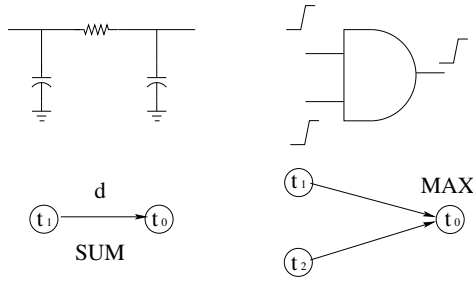


Figure 2: Two basic operations SUM and MAX in SSTA.

where

$$\begin{aligned}\theta^2 &= \sigma^2(t_1) + \sigma^2(t_2) - 2cov(t_1, t_2) \\ \lambda &= (\mu(t_1) - \mu(t_2)) / \theta \\ P &= \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{\lambda^2}{2}\right) \\ Q &= \frac{1}{2} \left( \operatorname{erf}\left(\frac{\lambda}{\sqrt{2}}\right) + 1 \right)\end{aligned}$$

Taking the MIN operation is similar, e.g., by  $MIN(t_1, t_2) = -MAX(-t_1, -t_2)$ .

SSTA have such MIN/MAX operations inherent from static timing analysis, i.e., input vector oblivious worst case timing analysis. They are the most computationally expensive operations in SSTA, and introduce pessimism. In the following, we propose a new operation, WEIGHTED SUM, by taking into account input signal statistics, e.g., in terms of signal probabilities as in power estimation literature.

## 2.2 Signal Toggling Analysis in Statistical Power Estimation

Power consumption estimation techniques include two categories: simulation based and statistical power estimation [13]. Statistical power estimation techniques achieve significant efficiency improvement than simulation based methods based on statistical characterization of circuit switching activities, e.g., by signal probability.

### 2.2.1 Signal Probability

**Definition 1** Signal probability  $P(y)$  of a net  $y$  is the occurrence probability for the net  $y$  to be of logic one.

Given the signal probabilities for the inputs of a Boolean function, one can compute the signal probability for the output of the function. For example, for a two-input AND gate  $y = x_1 \cdot x_2$ , applying basic probability theory gives  $P(y) = P(x_1)P(x_2)$  (Fig. 3). In general, for a Boolean function  $y = f(x_1, \dots, x_n)$ , if the inputs  $x_i$  are independent, the signal probability  $P(y)$  of the output  $y$  is given by

$$P(y) = P(x_1)P(f_{x_1}) + P(\bar{x}_1)P(f_{\bar{x}_1}) \quad (5)$$

where  $f_{x_1} = f(1, x_2, \dots, x_n)$  and  $f_{\bar{x}_1} = f(0, x_2, \dots, x_n)$  are cofactors of  $f$  with respect to  $x_1$ . By representing a Boolean function in a binary decision diagram (BDD), such computation takes linear time in term of the BDD size. Computing signal probabilities for all nodes in a netlist takes a single traversal of the netlist.

### 2.2.2 Signal Toggling Rate

**Definition 2** Signal toggling rate  $\rho_y$  is the expected number of signal togglings per unit time for a net  $y$ .

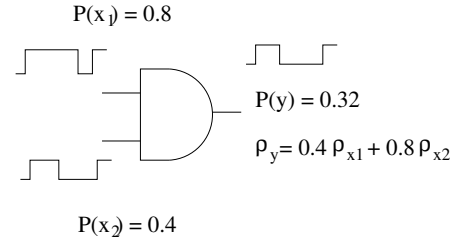


Figure 3: Signal probability and signal toggling rate computation for an AND gate.

Statistical power estimation computes signal toggling rate  $f_y$  at the output of a logic gate based on the Boolean difference function as follows (Fig. 3) [12].

$$\rho_y = \sum_i P\left(\frac{\partial y}{\partial x_i}\right) \rho_{x_i} \quad (6)$$

The Boolean difference function enables a signal propagation path from an input  $x_i$  to the output  $y$  and is given by

$$\frac{\partial y}{\partial x_i} = y|_{x_i=1} \oplus y|_{x_i=0} \quad (7)$$

where  $\oplus$  denotes the exclusive-or operation. Given signal probabilities in (5) in a netlist, Boolean difference probabilities in (7) and signal toggling rates in (6) can be computed efficiently in a single traversal of the netlist.

## 3 Signal Probability Based Statistical Timing Analysis (SPSTA)

In this section, we introduce signal probability in statistical power estimation to statistical timing analysis and compute signal transition temporal occurrence probabilities for more accurate statistical timing analysis.

### 3.1 Variable: Signal Transition Temporal Occurrence Probability

**Definition 3** Signal transition temporal occurrence probability (t.o.p.)  $\Phi(y) = f(t)$  is the time domain occurrence probability for a signal to take transition for net  $y$ .

Signal transition temporal occurrence probability gives time domain distribution of signal toggling rate, while the signal toggling rate concept in power estimation is the average of signal transition temporal occurrence probability in time domain.

Signal transition temporal occurrence probability also differs with signal arrival time probability density function, in that the time domain integral of signal arrival time probability density function equals one, based on the assumption of static timing analysis that there is always a signal transition occurrence in a net in a clock cycle, while the integral of signal transition temporal occurrence probability gives signal toggling rate, which may not be one (e.g., see Fig. 4). A signal transition temporal occurrence probability function can be normalized to unit integral to give a signal arrival time pdf. Therefore, the results of signal probability based statistical timing analysis, in the form of signal transition temporal occurrence probability, include both signal arrival time pdf's, and signal toggling rates. This implies that signal probability based statistical timing analysis gives more information than static timing analysis, and the additional signal toggling rate estimates help in power consumption and supply voltage degradation estimation (which in turn could affect timing).

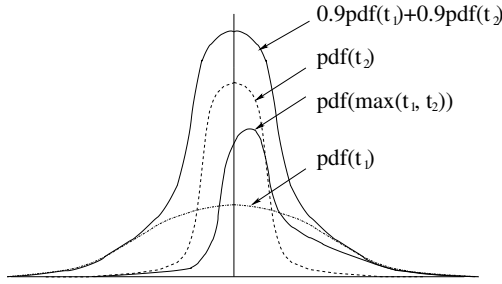


Figure 4: The results of the MAX and the WEIGHTED SUM operations for an AND gate with two inputs both of 0.9 signal probability, and of signal arrival times  $t_1$  and  $t_2$  in symmetric distributions with the same mean but different deviations.

### 3.2 Operation: WEIGHTED SUM

Signal transition temporal occurrence probability functions are computed in the same way as signal toggling rates, i.e., by weighted summation based on signal probabilities.

$$\varphi(y) = \sum_i P\left(\frac{\partial y}{\partial x_i}\right)\varphi(x_i) \quad (8)$$

The result of a WEIGHTED SUM operation would be significantly different than that of a MAX operation. For symmetric input signal arrival time distributions, the result of a WEIGHTED SUM operation is still symmetric, while the result of a MAX operation is non-symmetric (Fig. 4).

It turns out that the MAX operation is applicable only for the signal arrival time of a non-controlled value at the output of a gate, e.g., for the arrival time of a logic one at the output of an AND gate. A MIN operation is needed for the signal arrival time of a controlled value at the output of a gate, e.g., for the arrival time of a logic zero at the output of an AND gate. Combining the results of the MIN and the MAX operations gives a symmetric signal arrival time distribution at the output of the gate, as is given by a WEIGHTED SUM operation. Furthermore, the MIN and the MAX operations give accurate output signal arrival times only when multiple inputs are switching at the same time, in other cases their results are bounds of the actual signal arrival times. A four-value logic based statistical timing analysis gives more accurate results as follows.

### 3.3 Extension to Four-Value Logic

We extend SPSTA to the four-value logic, i.e., based on logic one '1', logic zero '0', rising signal transition 'r', and falling signal transition 'f' (Table 1). This is needed because we need to separate rising signal transition arrival time distributions with falling signal transition arrival time distributions, which are propagated by different MIN/MAX computations for a multiple-input gate, as in static timing analysis. Different MIN/MAX computations for the rising/falling signal transitions spread the rising and the falling signal transitions in different directions, and lead to increased signal transition time distribution deviation. One must separate the rising and the falling signal transitions to take this spreading effect into account in statistical timing analysis. Separating the rising and falling signal transitions also filters out the glitches which result from simultaneous rising and falling signal transitions at the inputs of a gate. Glitches are not counted by a MIN/MAX operation, but are included in a WEIGHTED SUM operation based on the two-value logic. Moving to four-value logic allows identification of glitches.

In a four-value logic, we extend (5) and compute signal probabilities for logic one '1', logic zero '0', rising signal transition 'r', and falling signal transition 'f', respectively, as follow.

$$P_{cd} = 1 - P_{ncd} - P_r - P_f$$

Table 1: Four-value logic AND and OR operations with MIN/MAX signal arrival time computation.

AND	0	1	r	f	OR	0	1	r	f
0	0	0	0	0	0	0	1	r	f
1	0	1	r	f	1	1	1	1	1
r	0	r	r (MAX)	0	r	r	1	r (MIN)	1
f	0	f	0	f (MIN)	f	f	1	1	f (MAX)

$$\begin{aligned} P_{ncd} &= \Pi_i P_{nc}(i) \\ P_r &= \Pi_i (P_{nc}(i) + P_{inv(f)}(i)) - P_{ncd}(i) \\ P_f &= \Pi_i (P_{nc}(i) + P_{inv(r)}(i)) - P_{ncd}(i) \end{aligned} \quad (9)$$

where  $cd$  is the controlled value,  $ncd$  is the non-controlled value,  $nc$  is the non-controlling value, and  $inv$  is the inversion of the gate. For example, for an AND gate, 0 is the controlling and the controlled value, and 1 is the non-controlling and the non-controlled value, there is no inversion  $inv(r) = r$ ,  $inv(f) = f$ , and the four-value logic signal probabilities are given as follow.

$$\begin{aligned} P_0(y) &= 1 - P_r(y) - P_f(y) - P_1(y) \\ P_1(y) &= \Pi_i P_1(x_i) \\ P_r(y) &= \Pi_i (P_1(x_i) + P_r(x_i)) - P_1(y) \\ P_f(y) &= \Pi_i (P_1(x_i) + P_f(x_i)) - P_1(y) \end{aligned} \quad (10)$$

In the four-value logic, the output signal transition temporal occurrence probability is given by a combination of WEIGHTED SUM and MAX operations, e.g., applying a WEIGHTED SUM operation based on the input signal probabilities, and applying the MAX operation for a multiple input switching scenario.

$$\begin{aligned} \varphi_r(y) &= \sum_R \Pi_{x_i \in R} P_r(x_i) \Pi_{x_i \notin R} P_{nc}(x_i) \varphi_r(\text{Max}_{x_i \in R}(x_i)) \\ \varphi_f(y) &= \sum_F \Pi_{x_i \in F} P_f(x_i) \Pi_{x_i \notin F} P_{nc}(x_i) \varphi_f(\text{Max}_{x_i \in F}(x_i)) \end{aligned} \quad (11)$$

where  $R \subseteq \{x_i\}$  is a set of rising inputs, and  $F \subseteq \{x_i\}$  is a set of falling inputs for a non-inverting gate. For example, for a two-input AND gate, the rising signal transition top function is given by

$$\begin{aligned} \varphi_r(y) &= P_r(x_1)P_1(x_2)\varphi_r(x_1) \\ &+ P_1(x_1)P_r(x_2)\varphi_r(x_2) \\ &+ P_r(x_1)P_r(x_2)\varphi_r(\text{MAX}(x_1, x_2)) \end{aligned} \quad (12)$$

The runtime is  $O(2^k)$  to compute signal transition temporal occurrence probabilities for a  $k$ -input gate, and is linear to the circuit size, i.e., the computation can be done in a single netlist traversal.

### 3.4 Computing Moments and Correlations

Statistical properties of signal transition temporal occurrence probability in a circuit can be represented by statistical moments, e.g., means, standard deviations, skewnesses, etc., and correlations. Statistical theory gives the moments and the correlations of a linear function as follows.

$$\begin{aligned} \bar{\varphi}_y &= \sum_i P\left(\frac{\partial y}{\partial x_i}\right)\bar{\varphi}_{x_i} \\ \sigma^2(\varphi_y) &= \sum_i P^2\left(\frac{\partial y}{\partial x_i}\right)\sigma^2(\varphi_{x_i}) \\ &+ 2 \sum_{i,j} P\left(\frac{\partial y}{\partial x_i}\right)P\left(\frac{\partial y}{\partial x_j}\right)cov(\varphi_{x_i}, \varphi_{x_j}) \end{aligned}$$

$$\begin{aligned} \text{cov}(\varphi_y, \varphi_k) &= \sum_i P\left(\frac{\partial y}{\partial x_i}\right) \text{cov}(\varphi_{x_i}, \varphi_k) \\ \text{corr}(\varphi_y, \varphi_k) &= \frac{\text{cov}(\varphi_y, \varphi_k)}{\sigma(\varphi_y)\sigma(\varphi_k)} \end{aligned} \quad (13)$$

The runtime of these moments and correlations computation is linear to the circuit size and the computation can be finished in a single netlist traversal.

### 3.5 Higher-Order Correlations and Exact Signal Probability

A higher-order covariance, e.g., a  $n$ -th order covariance of  $n+1$  variables is defined as follows.

$$\text{cov}(x_0, \dots, x_n) = E(\Pi_i(x_i - \bar{x}_i)) \quad (14)$$

Given these higher-order covariances, the signal probability of a Boolean function is computed based on simple Boolean and polynomial derivations. For example,

$$P(x_1 x_2) = P(x_1)P(x_2) + \text{cov}(x_1, x_2) \quad (15)$$

based on the definition of covariance

$$\text{cov}(x_1, x_2) = E((x_1 - \bar{x}_1)(x_2 - \bar{x}_2)) \quad (16)$$

and

$$\begin{aligned} P(\bar{x}_1 x_2) &= P(x_2) - P(x_1 x_2) \\ P(x_1 + x_2) &= P(x_1) + P(x_2) - P(x_1 x_2) \end{aligned} \quad (17)$$

based on set theory.

Combined with symbolic simulation which computes Boolean functions for each node in the circuit, e.g., based on Binary Decision Diagrams (BDDs), signal probabilities are then computed for each node in a circuit. This method takes into account the correlations due to reconvergent fanout nets, which is not taken into account by the method in Section 2.2.1.

However, a large number, e.g.,  $O(n^2)$  first-order correlations, and much more higher-order correlations are needed to compute exact signal probabilities in a netlist of  $n$  nodes. Truncating higher-order moments and correlations gives an accuracy-efficiency trade-off.

### 3.6 Symbolic Analysis

The second category of methods compute circuit properties (e.g., signal probabilities, signal arrival times) in closed form expressions of design parameters and variational parameters which represent process and environmental variabilities. Such methods include: polynomial computation [8], affine arithmetics [10], and probabilistic interval analysis [20].

Variational delays are obtained by derivation of closed-form formulas [10, 20], or by sampling analysis and regression [8, 6].

This approach avoids the run time and memory space consuming moment and correlation computation. However, another dimension of accuracy-efficiency tradeoff is introduced in truncating the higher-order polynomial terms.

### 3.7 Comparing SPSTA and SSTA

There are several significant advantages of signal probability based statistical timing analysis (SPSTA) over statistical static timing analysis (SSTA).

1. SPSTA aims to capture the exact circuit performance statistics. SSTA is a mixture of accurate characterization and worst case analysis, which is inconsistent, and provides neither accurate analysis nor bounds of performance statistics.

2. SPSTA takes into account statistical characterization of circuit inputs and compute signal probabilities for statistical circuit performance, which as we observe depends on the inputs and the operation mode of the circuit. SSTA does not take any input statistics or circuit operation mode into account.
3. SSTA is significantly pessimistic. Current SSTA tools compute two signal arrival time pdf's as min/max signal arrival time distributions for each timing node, as in static timing analysis. However, such min/max signal arrival time pdf's are corners in essence, corners cannot be enumerated, and the resultant probabilities are not timing yield.
4. SSTA pessimism due to worst case analysis does not guarantee any bound of the actual signal arrival time distribution. Pessimistic timing analysis could lead to larger deviations and less correlations of signal arrival times. Less correlations lead to optimistic estimation, for example, underestimation of supply voltage variation, crosstalk coupling and multiple input switching effects. These effects eventually offset SSTA results and do not guarantee any bound of timing performance. SPSTA is more accurate in computing correlations hence circuit performance statistics.
5. SPSTA computes occurrence probability for a rising or falling signal transition as well as its timing distribution. Such an occurrence probability is an integral part in statistical timing analysis, e.g., in estimating the probability for a chip to meet its performance requirement.
6. In summary, SPSTA achieves more accurate, consistent, and complete statistical timing analysis than SSTA.

## 4 Experiment

In the following experiments, we implement three timing analyzers: SPSTA, SSTA, and a logic simulator for Monte Carlo simulation, and compare their resultant signal arrival time means and standard deviations for ISCAS'89 benchmark circuits. We implement SPSTA based on the four-value logic and the equations in Section 3.3. Our implemented SSTA separates rising and falling signal transition arrival times, and computes their means and standard deviations for each timing node by either the MIN or the MAX operation, based on the logic of the gate and the input signal transition directions. For Monte Carlo simulation, we implement a four-value logic (i.e., logic one 1, logic zero 0, rising signal transition  $r$ , and falling signal transition  $f$ ) simulator. We assign the four logic values and signal arrival times for the rising and the falling signal transitions to the primary inputs and the flip-flop outputs, and propagate them through the netlists. We do not count glitch, so a rising and a falling signal transition for an AND gate give logic zero at the output. We take the MIN or the MAX computation for signal arrival times based on the logic of the gate and the signal transition directions.

The ISCAS'89 benchmark circuits are assumed to have unit delay for each gate, and zero delay for each net. In the first part of the experiment, we assign logic one, logic zero, rising signal transition, and falling signal transition with equal occurrence probabilities to the primary inputs and the flip-flop outputs of the benchmark circuits. The rising and the falling signal transitions are in standard normal distributions. Therefore, the primary inputs have 0.5 signal probability, 0.5 mean signal toggling rate, and 0.25 variance of signal toggling rate. In the second part of the experiment, we assign 15% logic one, 75% logic zero, 2% rising signal transition, and 8% falling signal transition to the primary inputs and the flip-flop outputs. The rising and the falling signal transitions are in standard normal distributions. Therefore, the primary inputs have 0.2 signal probability, 0.1 mean signal toggling rate, and 0.09 variance of signal toggling rate. In both cases, we keep the primary inputs and the flip-flop outputs independent to each other, such that there is zero covariance of signal toggling rate between any two primary inputs, and we assign to each primary input and flip-flop output a

mean signal transition temporal occurrence probability in the standard normal distribution  $\bar{\varphi}_i(t) = N(0, 1)$ .

Table 2 shows the means, the standard deviations, and the occurrence probabilities of signal transitions in the most critical timing paths, and Table 3 gives the runtimes, for SPSTA, SSTA, and 10,000 runs of Monte Carlo logic simulation, respectively. We have the following observations.

1. SSTA conducts the MIN and MAX operations respectively for the rising and falling signal transitions in a signal propagation path, and does not provide a bound of critical path delay. SSTA results are also independent of primary inputs and flip-flop outputs statistics.
2. SPSTA takes primary inputs and flip-flop outputs statistics into account, and gives more accurate mean delay estimation than SSTA in most of the test cases, compared with Monte Carlo logic simulation results.
3. SPSTA estimates of signal arrival time variations are much more close to Monte Carlo simulation results. SSTA estimates of signal arrival time variations are much smaller, e.g., the MIN and MAX operations result in smaller standard deviations for the output than the inputs of a gate.
4. SPSTA gives accurate signal probabilities and signal transition occurrence probabilities. SSTA does not provide such estimations.
5. We implemented SPSTA without consideration of signal correlations. Signal correlations as well as the MIN/MAX computations contribute to the difference between the SPSTA results and the Monte Carlo simulation results in Table 2.
6. SPSTA and SSTA are far more efficient than Monte Carlo simulation.

## 5 Conclusion

Input vector oblivious worst case analysis in SSTA is inconsistent with statistical analysis and leads to pessimism and inaccuracy. We propose signal probability based statistical timing analysis, by taking circuit inputs statistics into account. Our experimental results show that our proposed SPSTA computes mean (standard deviation) of signal arrival times within 6.2% (18.6%), while SSTA computes mean (standard deviation) of signal arrival times within 13.40% (64.3%) of Monte Carlo simulation results. SPSTA also provides signal probability estimation within 14.28% of Monte Carlo simulation results for the ISCAS'89 benchmark circuits.

## References

- [1] A. Agarwal, D. Blaauw, V. Zolotov, and S. Vrudhula, "Statistical Timing Analysis Using Bounds," in Proc. *Design, Automation, and Test in Europe*, 2003, pp. 62-68.
- [2] A. Agarwal, F. Dartu and D. Blaauw, "Statistical Gate Delay Model Considering Multiple Input Switching," in Proc. *Design Automation Conference*, 2004, pp. 658-663.
- [3] K. Agarwal, D. Sylvester, D. Blaauw, F. Liu, S. Nassif and S. Vrudhula, "Variational Delay Metrics for Interconnect Timing Analysis," in Proc. *Design Automation Conference*, 2004, pp. 381-384.
- [4] S. Bhanja and N. Ranganathan, "Switching Activity Estimation of VLSI Circuits Using Bayesian Networks," *IEEE Trans. on VLSI*, 11(4), 2003, pp. 558-567.
- [5] A. Ghosh, S. Devadas, K. Keutzer, and J. White, "Estimation of Average Switching Activity in Combinational and Sequential Circuits," in Proc. *Design Automation Conference*, 1992, pp. 253-259.
- [6] A. B. Kahng, B. Liu and X. Xu, "Statistical Crosstalk Aggressor Alignment Aware Interconnect Delay Calculation," in Proc. *System Level Interconnect Prediction*, 2006.
- [7] A. B. Kahng, B. Liu and X. Xu, "Statistical Gate Delay Calculation with Crosstalk Alignment Consideration," in Proc. *Great Lake Symposium on VLSI*, 2006.

Table 2: The means and standard deviations of the rising and the falling signal arrival times on the most critical path given by (1) 4-value logic based SPSTA, (2) min-max separated SSTA, and (3)  $10K \times$  Monte Carlo simulation. The primary inputs and the flip-flop outputs are of (I) 0.5 signal probability, 0.5 mean signal toggling rate, and 0.25 variance of signal toggling rate, or (II) 0.2 signal probability, 0.1 mean signal toggling rate, and 0.09 variance of signal toggling rate.

(I)									
test case		SPSTA			SSTA		$10K \times$ Monte Carlo		
		$\mu$	$\sigma$	$P$	$\mu$	$\sigma$	$\mu$	$\sigma$	$P$
s208	r	6.49	1.39	0.01	7.23	0.84	6.65	1.56	0.03
s298	r	5.40	1.28	0.13	6.23	0.70	5.15	1.58	0.19
s344	r	8.96	2.83	0.25	8.95	0.69	8.92	2.86	0.25
s349	r	8.96	2.80	0.25	8.95	0.69	8.92	2.86	0.25
s382	r	6.76	0.76	0.20	6.82	0.68	6.62	1.13	0.20
s386	r	8.18	1.76	0.01	8.96	0.60	7.90	1.89	0.01
s526	r	5.40	1.28	0.13	6.23	0.70	5.15	1.58	0.19
s1196	r	15.92	2.21	0.00	14.44	0.47	15.55	2.07	0.00
s1238	r	14.92	2.29	0.00	13.44	0.47	15.66	0.85	0.00
s208	f	7.44	1.40	0.00	8.23	0.84	6.98	1.59	0.00
s298	f	5.48	1.58	0.25	5.79	0.54	5.14	1.82	0.25
s344	f	9.04	2.99	0.25	8.20	0.81	8.72	3.27	0.25
s349	f	9.04	2.99	0.25	8.20	0.81	8.72	3.27	0.25
s382	f	6.81	1.40	0.21	6.77	0.57	6.23	1.77	0.13
s386	f	7.18	1.77	0.01	7.96	0.60	6.90	1.89	0.01
s526	f	5.48	1.49	0.25	5.79	0.54	5.16	1.82	0.25
s1196	f	14.92	2.21	0.00	13.44	0.47	14.55	2.07	0.00
s1238	f	12.92	2.30	0.00	10.73	0.61	13.10	0.00	0.00
(II)									
test case		SPSTA			SSTA		$10K \times$ Monte Carlo		
		$\mu$	$\sigma$	$P$	$\mu$	$\sigma$	$\mu$	$\sigma$	$P$
s208	r	6.45	1.81	0.02	7.23	0.84	6.33	1.97	0.02
s298	r	5.16	1.63	0.06	6.23	0.70	4.98	1.71	0.05
s344	r	9.23	3.47	0.13	8.95	0.69	8.93	3.51	0.13
s349	r	9.24	3.39	0.13	8.95	0.69	8.93	3.51	0.13
s382	r	6.49	0.84	0.03	6.82	0.68	6.35	1.16	0.03
s386	r	6.86	2.23	0.00	8.96	0.60	7.02	2.27	0.00
s526	r	5.17	1.60	0.06	6.23	0.70	4.71	1.68	0.15
s1196	r	13.12	2.74	0.00	14.44	0.47	16.24	0.00	0.00
s1238	r	11.14	3.58	0.00	13.44	0.47	14.24	0.00	0.00
s208	f	6.18	2.42	0.00	8.23	0.84	5.64	1.60	0.06
s298	f	5.24	1.49	0.11	5.79	0.54	5.16	1.77	0.11
s344	f	9.51	3.16	0.10	8.20	0.81	9.35	3.37	0.10
s349	f	9.51	3.24	0.10	8.20	0.81	9.35	3.38	0.10
s382	f	6.41	0.62	0.07	6.77	0.57	6.02	1.04	0.07
s386	f	7.36	1.32	0.05	7.96	0.60	6.62	1.80	0.05
s526	f	5.24	1.53	0.11	5.79	0.54	4.85	1.85	0.11
s1196	f	12.12	2.74	0.00	13.44	0.47	15.24	0.00	0.00
s1238	f	10.12	2.59	0.00	10.73	0.61	13.24	0.00	0.00

Table 3: CPU runtime (in seconds) of (1) 4-value logic based SPSTA, (2) min-max separated SSTA, and (3)  $10K \times$  Monte Carlo simulation.

test	SPSTA	SSTA	$10K \times$ Monte Carlo
s208	0.78	0.20	11.96
s298	0.93	0.25	14.37
s344	1.39	0.44	25.63
s349	1.41	0.45	25.06
s382	1.37	0.45	23.89
s386	1.43	0.40	21.49
s526	2.13	0.70	33.88
s1196	13.12	6.62	289.33
s1238	12.94	6.68	250.64

- [8] V. Khandelwal and A. Srivastava, "A General Framework for Accurate Statistical Timing Analysis Considering Correlations," in Proc. *Design Automation Conference*, 2005.
- [9] Y. Liu, L. T. Pileggi and A. J. Strojwas, "Model Order-Reduction of RC(L) Interconnect Including Variational Analysis," in Proc. *Design Automation Conference*, 1999, pp. 201-206.
- [10] J. D. Ma and R. A. Rutenbar, "Interval-Valued Reduced Order Statistical Interconnect Modeling," in Proc. *Intl. Conference on Computer Aided Design*, 2004, pp. 460-467.
- [11] R. Marculescu, D. Marculescu, and M. Pedram, "Probabilistic Modeling of Dependencies During Switching Activity Analysis," *IEEE Trans. CAD*, 17, 1998, pp. 73-83.
- [12] F. N. Najm, "Transition Density: A New Measure of Activity in Digital Circuit," *IEEE Trans. on Computer-Aided Design*, 12(2), 1993, pp.310-323.
- [13] F. N. Najm, "A Survey of Power Estimation Techniques in VLSI Circuits," *IEEE Trans. On VLSI Systems*, 2(4), 1994, pp. 446-455.
- [14] F. N. Najm and N. Menezes, "Statistical Timing Analysis Based on a Timing Yield Model," in Proc. *Design Automation Conference*, 2004, pp. 460-465.
- [15] F. N. Najm, R. Burch, P. Yang, and I. Hajj, "Probabilistic Simulation for Reliability Analysis of CMOS VLSI Circuits," *IEEE Trans. on Computer-Aided Design*, 9(4), 1990, pp. 439-450.
- [16] S. R. Nassif and J. N. Kozhaya, "Fast Power Grid Simulation," in Proc. *Design Automation Conference*, pp. 156-161, 2000.
- [17] A. Odabasioglu, M. Celik and L. T. Pileggi, "PRIMA: Passive Reduced-Order Interconnect Macromodeling Algorithm," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, vol. 17, no. 8, August 1998, pp. 645-654.
- [18] M. Orshansky, A. Bandyopadhyay, "Fast statistical timing analysis handling arbitrary delay correlations," in Proc. *Design Automation Conference*, 2004, pp. 337-342.
- [19] M. Orshansky, K. Keutzer, "A general probabilistic framework for worst case timing analysis," in Proc. *Design Automation Conference*, 2002, pp. 556-561.
- [20] M. Orshansky, W.-S. Wang, G. Xiang, and V. Kreinovich, "Interval-Based Robust Statistical Techniques for Non-Negative Convex Functions with Applications to Timing Analysis of Computer Chips," in Proc. *Reliable Engineering Computing*, 2006.
- [21] S. Pant, D. Blaauw, V. Zolotov, S. Sundareswaran, and R. Panda, "A Stochastic Approach to Power Grid Analysis," in Proc. *Design Automation Conference*, 2004, pp. 171-176.
- [22] C.-Y. Wang and K. Roy, "COSMOS: A Continuous Optimization Approach for Maximum Power Estimation of CMOS Circuits", in Proc. *International Conference of Computer-Aided Design*, 1997, pp. 52-55.
- [23] C.-Y. Wang, K. Roy and T.-L. Chou, "Maximum Power Estimation for Sequential Circuits Using a Test Generation Based Technique," in Proc. *IEEE Custom Integrated Circuits Conference*, 1996, pp. 229-232.
- [24] Q. Wu, Q. Qiu and M. Pedram, "Estimation of Peak Power Dissipation in VLSI Circuits Using the Limiting Distributions of Extreme Order Statistics", *IEEE Trans. on Computer-Aided Design*, 20(8), 2001, pp. 942-956.
- [25] C. Visweswariah, K. Ravindran, K. Kalafala, S. G. Walker, and S. Narayan, "First-Order Incremental Block-Based Statistical Timing Analysis," in Proc. *DAC*, 2004.
- [26] L. Zhang, W. Chen, Y. Hu and C. C. Chen, "Statistical Static Timing Analysis with Conditional Linear MAX/MIN Approximation and Extended Canonical Timing Model," *IEEE Trans. on CAD*, 2005.