Modeling Crosstalk in Statistical Static Timing Analysis
Ravikishore Gandikota, David Blaauw, Dennis Sylvester
Department of EECS, University of Michigan, Ann Arbor MI. email:{gravkis,blaauw,dennis}@eecs.umich.edu

Abstract
Increasing process variation in the nanometer regime motivates the use of statistical static timing analysis tools for timing verification. As device dimensions get smaller, signal integrity effects such as crosstalk noise become more significant. Therefore, it is necessary to accurately model the impact of crosstalk noise on the circuit delay. Process variations cause variability in the crosstalk alignment which leads to the variability in the delay noise. However, most of the existing approaches model delay noise as a worst-case deterministic quantity. In this work, we capture the variability of delay noise by first deriving the closed-form expressions of mean and standard deviation of the delay noise distribution. Next, we obtain the correlation information of the delay noise and use it to represent the delay noise distribution in canonical form. Delay noise, in canonical form, can easily be integrated with existing SSTA tools. We show experimental results that verify the accuracy of our approach.

Categories and Subject Descriptors
B.7.2 [Integrated Circuits]: Design Aids; B.8.2 [Performance and Reliability]: Performance Analysis and Design Aids

General Terms
Algorithms, Design

Keywords
Crosstalk, delay noise, SSTA

1. INTRODUCTION

Imprecise control of lithography equipment and channel doping can lead to a significant variability of the device dimensions and threshold voltages. In nanometer regime, the variability in manufacturing process has not scaled commensurate with the device dimensions. Consequently, the variability of circuit performance has grown rapidly as we continue shrinking device dimensions. To account for the variability in the timing verification of the circuit, we can perform traditional static timing analysis (STA) at multiple process, voltage and temperature (PVT) corners. However, with an increase in variability, the number of corners that are needed to accurately model the circuit performance has grown rapidly. Therefore, static timing analysis (STA) which models gate delays and circuit performance as random variables, with a probability distribution function (PDF), has emerged as an efficient alternative to corner-based STA. Most of the techniques proposed in STA can be classified as either path-based or block-based. Path-based STA algorithms ([2],[3]) compute the delay distribution of the critical paths in the circuit and are accurate since they preserve paths correlation information. However, path-based approaches suffer from an explosion in the total number of paths that have to be enumerated. On the other hand, block-based STA ([4],[5],[12]) requires only one single path-like traversal of the circuit graph and is more efficient than path-based STA.

Scaling of device dimensions has also led to a considerable reduction in gate delays. However, due to less aggressive interconnect scaling, wire delays have not reduced in proportion to gate delays and wire delays, especially the global interconnect delays, now contribute significantly to the total circuit delay. Due to parasitic capacitive coupling between wires, wire delay depends on the switching activity of neighboring wires. As the spacing between wires continues to shrink, the magnitude of the coupling capacitance increases and it now dominates the wire ground capacitance. Therefore, the magnitude of noise that is coupled on a victim net due to switching transitions of aggressor nets has become significant. If the aggressor-victim pair switch in the same direction, coupling noise can speedup the victim transition and reduce the victim delay. On the other hand, if the aggressor-victim pair switch in mutually opposite directions, coupling noise can slowdown the victim transition and increase the victim delay. This change in victim delay due to coupling noise is referred to as delay noise and it contributes to a significant portion of the circuit delay. Therefore, accurate modeling of delay noise is necessary for timing signoff analysis of high performance designs.

It has been observed that delay noise strongly depends on the aggressor-victim input skew or the difference between arrival times at the inputs of aggressor-victim drivers (see Figure 1). Process variations translate into delay variations and the delay variability of upstream gates translates into uncertainty in the arrival times at the input of aggressor-victim drivers. Therefore, due to the variability in aggressor-victim input skew, delay noise can no longer be treated as a deterministic quantity. Sources such as aggressor-victim interconnect variation also contribute to the variability of delay noise. However, a majority of the timing analysis techniques used today model crosstalk induced delay noise as a deterministic quantity.

Overlap between the aggressor-victim timing windows computed in STA was used in [1] to identify whether the aggressor can couple noise onto the victim. In block-based STA, however, the end points of statistical timing windows are random variables which are obtained by performing recursive ‘max’ and ‘min’ atomic operations in a topological order. In [7], the authors extend the above idea to STA by expanding the nominal timing window by 3σ on both sides, where σ is the standard deviation of early and late arrival times. Overlap between the expanded timing windows of the aggressor-victim pair is used to identify whether noise is coupled onto the victim net. Since, the worst-case delay noise is applied whenever the expanded windows overlap, the above technique leads to a pessimistic estimation of delay noise.

The mutual dependence of delay noise and timing windows leads to a ‘chicken-and-egg’ problem. However, in [8] the authors propose an iterative approach for crosstalk aware STA as a fixpoint on a lattice and theoretically proved its convergence. In [9] the coupling capacitance is modeled as a random variable which depends on the skew between aggressor-victim arrival times. In [13] the...
authors provide a closed-form expression for computing the PDF of delay noise given the aggressor-victim input arrival time distributions. However, delay noise was assumed to be independent of input arrival time distributions and no correlation information of the delay noise was preserved. The lack of correlation information makes it difficult to integrate the delay noise distribution accurately into an statistical analysis tool.

A canonical first order model was used in ([5],[6]) which captures the first order sensitivities of delay to the (normally distributed) source of variation and preserves the correlation information among all timing quantities. In this work, we show how the first order SSTA framework can be extended to accurately capture the variation in delay noise. We model statistical delay noise as a random variable and express it in the canonical form by computing its first order sensitivities to the variation sources. Given a delay change curve which captures the dependence of delay noise on the aggressor-victim input skew and an input skew distribution in canonical form, we obtain closed-form expressions of the resulting delay noise distribution. To compute the ‘noisy’ victim output arrival time, we must add the delay noise to the ‘noiseless’ victim output arrival time. In order to do so, we express delay noise in the canonical form by matching the first two moments and computing the correlation information. Since delay noise and victim output arrival time are both expressed in canonical form, we can use the statistical ‘sum’ operation to compute the noisy victim output arrival time with delay noise.

In this work, we propose the use of a statistical skew window whose end points are obtained by subtracting the end points of the aggressor input timing window from the late victim input arrival time. Using the skew window and the delay change curve, we analytically obtain the delay noise distribution in canonical form. In order to further reduce pessimism in our analysis, we propose to fragment the skew window into smaller segments. Using the fragmented skew window and the delay change curve, we then obtain the distribution of delay noise. The proposed technique matches well with Monte-Carlo simulations and we observe a significant reduction in pessimism of delay noise when compared to prior approaches which do not model delay noise as a statistical quantity.

The rest of the paper is organized as follows: In Section 2, we analyze the problem of computing the delay noise distribution in the presence of variation in detail. In Section 3, we present an analytical technique to compute the delay noise distribution in canonical form, given a single aggressor-victim input skew distribution and a DCC. In Section 4, we extend the analytical technique such that worst-case delay noise computation can be performed within the current SSTA framework with statistical timing windows. In Section 5, we present experimental results and in Section 6, we conclude this paper.

2 PROBLEM DESCRIPTION

In this section, we examine the problem of modeling the delay noise distribution in the presence of process variations. The amount of delay noise that is coupled to a victim by an aggressor depends on several factors such as aggressor-victim slew rates, driver strengths, the ratio of coupling capacitance to ground capacitance and the input skew. Also, an aggressor can couple noise only when its transition is temporally close to the victim transition. Therefore, the magnitude of delay noise strongly depends on the aggressor-victim input skew. The HSPICE simulation plot in Figure 1 shows the delay noise as a function of the input skew and is referred to as the Delay Change Curve (DCC). The DCC can be derived either by using SPICE based methods [10] or by using analytical methods [11] in which the noise pulse coupled on the victim is approximated by a two piece model and the DCC is obtained analytically by curve-fitting. Process variation leads to uncertainty in signal arrival times at the aggressor-victim inputs. Therefore, delay noise which is a function of a variable input skew is no longer deterministic. However, a majority of statistical timing analysis techniques model the worst-case delay noise as a deterministic quantity and this can often lead to pessimistic results.

The goal of this work is to model delay noise in current SSTA framework where delays are expressed in a canonical form

\[ d = d_0 + \sum_{i=1}^{N} s_i \cdot p_i + s_{N+1} \cdot R \]  

where \(d_0\) is the nominal delay, \(s_i\) is the of sensitivity of delay to the process parameter \(p_i\) which are standard normal random variables. \(R\) is the random component of the delay noise distribution and is also a standard normal random variable. Principal Component Analysis (PCA) can be used to transform the set of process parameters into a set of mutually independent normal random variables. The early and late arrival time distributions are propagated by performing statistical ‘min’ and ‘max’ operations recursively in a topological order ([5],[6]).

In [15], it has been shown that the worst-case delay noise occurs when the victim input transition occurs at the latest point in its timing window. Therefore, for computing the worst-case delay noise, we are only interested in the distribution of late victim input arrival time. Given the statistical timing window at the input of the aggressor, we subtract the early and late aggressor input arrival time distributions from the late victim input arrival time distribution to obtain the skew window (as shown in Figure 2). In this work, using this statistical skew window and the DCC, we derive closed-form expressions for the mean and variance of the delay noise distribu.

Figure 2. Skew window obtained by subtracting the early and late aggressor inp. arrival times from the late victim inp. arrival time.
tion. Note that the use of a single skew window can lead to pessimistic bounds on the delay noise distribution. Therefore, we propose to divide the skew window into smaller segments to further reduce the amount of pessimism in our analysis.

Since delay noise strongly depends on the input skew, in this work, we model the dominant source of variation in delay noise which is the variability in the input skew. Other sources such as variation in the aggressor-victim coupled interconnect also contribute to variability of the delay noise. However, their contribution to delay noise variability can be quite small. For instance, it has been reported in [12], that interconnect variation causes only a 10% (3σ/µ) variability in the magnitude of the peak noise voltage. Also, in [13] the authors show that the variability in delay noise due to other sources of variation can be assumed to be independent of the input skew distribution, without much loss of accuracy. Therefore, in this work we focus on the dominant source of variation in delay noise which is the variation in the input skew distribution. Also, the ‘chicken-and-egg’ problem occurring due to the mutual dependence of delay noise and timing windows can be solved using iterations [8]. Hence, in this work we focus on accurately modeling the delay noise distribution on the victim within a single iteration of the delay noise computation loop.

3. STATISTICAL DELAY NOISE

In this section, we analytically compute the delay noise distribution in canonical form, given a single input skew distribution in canonical form and a quadratic model of the DCC. We first show that the relative ratios of sensitivities of delay noise distribution is identical to that of the input skew distribution. We then obtain closed form expressions for computing the mean and standard deviation of delay noise distribution. The results obtained in this section will be used later in Section 4 to compute the worst-case delay noise in SSTA framework where we have statistical skew windows.

3.1 Correlations in delay noise

Since the DCC captures the dependence of delay noise on the input skew, it is easy to see that the delay noise distribution must be correlated with the input skew distribution. However, in this subsection, given a quadratic DCC model we show that the correlations in the input skew are preserved exactly in the delay noise distribution. This fact allows us to represent the delay noise distribution in a canonical form. Delay noise distribution in canonical form preserves the necessary correlation information and can easily be integrated in traditional block-based SSTA methods.

**Theorem 1.** Given a quadratic DCC and an input skew distribution in canonical form, the relative ratios of sensitivities of delay noise distribution to process parameters is the same as that of input skew distribution.

**Proof:** Without loss of generality, we assume an input skew distribution

\[ s = s_0 + s_1 \cdot x_1 + s_2 \cdot x_2 \]

(2)

having a mean \( s_0 \) and sensitivities \( s_1 \) and \( s_2 \) with respect to two independent standard normal random variables \( x_1 \) and \( x_2 \). Since \( x_1 \) and \( x_2 \) are independent and have unit variance, it is easy to see that the covariance of the input skew \( s \) with \( x_1 \) is given by

\[ \text{Cov}(s, x_1) = s_1 \]

(3)

The delay noise obtained from the DCC has a quadratic dependence on the input skew \( s \), that is

\[ d = a \cdot s^2 + b \cdot s + c \]  

(4)

Substituting (2) in (4), we obtain

\[ d = a \cdot (s_0 + s_1 \cdot x_1 + s_2 \cdot x_2)^2 + b \cdot (s_0 + s_1 \cdot x_1 + s_2 \cdot x_2) + c \]

(5)

The covariance of delay noise with parameter \( x_1 \) is given by

\[ \text{Cov}(d, x_1) = E[(d - d_0) \cdot x_1] \]

(6)

where \( E \) is the expectation operator. Substituting (5) in (6), we get

\[ \text{Cov}(d, x_1) = E\left[ a^2 s_1^2 + 2a s_1^2 s_2 x_1 + (2a + b) s_1^2 x_1^2 + 2b s_1^2 x_1 x_2 + 2a s_1^2 s_2 x_2 \right] \]

(7)

Since \( x_1 \) and \( x_2 \) are independent, the expectation of cross-product terms containing \( x_1 \cdot x_2 \) is zero. Equation (7) reduces to

\[ \text{Cov}(d, x_1) = E[2a s_1^2 x_1] + E[(2a + b) s_1^2 x_1] \]

(8)

Using the linearity of expectation operator, we rewrite (8) as

\[ \text{Cov}(d, x_1) = E[a s_1^2 x_1^2] + E[(2a + b) s_1^2 x_1] \]

(9)

The odd moments of a standard normal random variable are zeros and the even moments evaluate to one. Therefore, the first term in (9) disappears and we finally obtain the result

\[ \text{Cov}(d, x_1) = (2a + b) \cdot s_1 \]

(10)

Note that the covariance of the delay noise obtained in (10) is the same as the covariance of input skew obtained in (3) scaled by a constant factor (i.e., \( 2a + b \)). Performing a similar analysis with process parameter \( x_2 \), we obtain

\[ \text{Cov}(d, x_2) = (2a + b) \cdot s_2 \]

(11)

Since the covariance of delay noise with respect to process parameters are a scaled version of the covariance of the input skew, from (3), (10) & (11) we obtain

\[ \frac{\text{Cov}(d, x_2)}{\text{Cov}(d, x_1)} = \frac{\text{Cov}(s, x_2)}{\text{Cov}(s, x_1)} = \frac{s_2}{s_1} \]

(12)

Since the ratios between covariance remains constant, the correlation information in the input skew is preserved in delay noise. Note that this result is independent of the number of process parameters that are considered in the input skew distribution as every covariance is scaled by the same factor. \( \Box \)

Given an input skew distribution and a quadratic DCC, using Theorem 1 we can obtain the correlations of the delay noise distribution. To express the delay noise in canonical form, we only need to compute the mean and the standard deviation of the delay noise distribution.

3.2 Canonical Delay Noise Distribution

In this sub-section, given a quadratic model of the DCC and the input skew distribution in canonical form, we analytically compute the mean and standard deviation of the delay noise distribution. Suppose that the input skew distribution is given by Equation (2). Since the process parameters are normal random variables, the input skew distribution \( f_s \) is therefore normally distributed with mean \( \mu \) and standard deviation \( \sigma \) given by

\[ f_s(s) = N(\mu, \sigma) = \frac{1}{\sigma \sqrt{2\pi}} \exp \left( \frac{(s - \mu)^2}{2\sigma^2} \right) \]

(13)

\[ \mu = s_0, \sigma = \sqrt{s_1^2 + s_2^2} \]
4.2 Multiple skew windows

Delay noise computation by using the skew window assumes a worst-case delay noise for the Case B. This worst-case assumption could prove to be pessimistic, especially when there are only a few
paths terminating on the aggressor input. For every path that terminates on the aggressor, we obtain a corresponding skew distribution by subtracting the path delay distribution from the late victim input arrival time distribution (as shown in Figure 2). As pointed out earlier, the mean values of each of these skew distributions is bounded by the skew window. Suppose we arrange the skew distributions in the order of increasing mean. If the number of paths terminating on the aggressor are small, then it is possible that there is a significant gap between the means of two consecutive paths. Under such circumstances, the probability of the occurrence of the worst-case delay noise can be reduced considerably.

Instead of using a single skew window, we propose using multiple skew windows [16] as a technique to reduce the amount of pessimism in the computation of delay noise distribution. Suppose we fragment the single skew window, which starts at the mean of early skew distribution and ends at the mean of the late skew distribution, into 5 smaller skew windows. If we have a path whose mean delay falls within any of the 5 smaller skew windows, then we assign the path to that particular skew window. In other words, the mean of the path distribution is bounded by the smaller skew window. Therefore, the end points of the smaller skew window are characterized by the path delay distribution. These smaller skew window can be used exactly in the same manner as earlier (i.e. Case A, Case B, & Case C). This approach of using multiple skew windows allows us to identify those cases where the worst-case skew is not feasible due to fewer number of paths terminating on the aggressor.

5 RESULTS

In this section, we will show experimental results that verifies the accuracy and effectiveness of our proposed approach for modeling the delay noise distribution in a block-based SSTA framework. A prototype noise analysis tool was implemented in C++ and a 0.13μm standard cell library was used for synthesis and technology mapping of MCNC benchmark circuits. The designs were placed and routed by using a commercial APR tool. The distributed RC parasitics were extracted by using a commercial parasitic extraction tool. For every aggressor-victim pair, the DCC’s were analytically generated on the fly. In our analysis, we assume a 3σ/μ variability of 30% for the gate length variations.

The accuracy of the analytical results for computing the mean and standard deviation of delay noise is verified against Monte-Carlo simulations in Figure 4. A normal input skew distribution is created whose standard deviation is fixed at 10ps and whose mean is varied from -50ps to 200ps. Using the DCC in Figure 1 and the analytical results from Section 2, we obtain the mean and standard deviation of delay noise. The accuracy of the results are verified using Monte-Carlo, where the input skew distribution is simulated using 10,000 samples. As expected, it can be seen that the mean delay noise peaks when the mean input skew is aligned at worst-case skew (around 30ps). It is interesting to note that the standard deviation of delay noise seems to be minimized at the point where the mean delay noise peaks.

In Table 1, we show the circuit delay distribution obtained by incorporating statistical delay noise in block-based SSTA. The first column shows the mean and standard deviation of the circuit delay with no noise. In the second column, we use the approach suggested in [7] and assume worst-case delay noise whenever the statistical aggressor-victim timing windows overlap. This worst-case delay noise assumption can lead to an unreasonably large amount of delay noise. For instance, for circuit i9, it can be seen that the mean circuit delay increases by about 60% when compared to the mean nominal circuit delay. In the third column, the circuit delay distribution is obtained by modeling statistical delay noise using the proposed approach with a single statistical skew window. Note that the percentage increase in mean circuit delay of circuit i9 is less than 20% of the mean nominal circuit delay. In the fourth column, we see that the use of ten smaller skew windows instead of a single skew window leads to a further reduction in the mean circuit delay. With the usage of smaller skew windows, the percentage increase in the mean circuit delay of i9 is less than 13% of the mean nominal circuit delay. On an average, by using a single statistical skew window, the mean delay noise decreases by 54.6%. Furthermore, we obtain an average reduction of 23.4% in the mean delay noise by the usage of 10 smaller skew windows for every aggressor.

6 CONCLUSIONS

In this work, we model the variability in delay noise which occurs due to the variability in the crosstalk alignment. We analytically compute the mean and standard deviation of the delay noise distribution. We also proved that, for a quadratic DCC, the correlations in input skew distribution are captured in the delay noise distribution and the ratios’ of the sensitivities of both distributions are identical. Using the correlation information and by matching the first two moments, we represent the delay noise distribution in canonical form which allows us to integrate delay noise into a standard statistical timing analysis tool. The accuracy of the analytical
results derived was verified against Monte-Carlo simulations. It was shown that the amount of pessimism in the delay noise distribution obtained is significantly less than that obtained by assuming a worst-case value. In future work, we plan on accounting the other sources of variation such as the victim slew variations and the interconnect variation while modeling delay noise.

REFERENCES


APPENDIX

A. First and Second Moments of delay noise

The delay noise distribution in Equation (15) is given by the sum of two terms. The moments of delay noise is given by the sum of the moments of both the terms in (15). In this Appendix, we derive the first and second moments only for first quadratic piece $a_1s^2 + b_1s + c_1$ and note that the derivation of the moments for the second piece is analogous.

While computing the expectation of $f_{y}$, we first perform a transformation of the variable $s$ to $z$

$$z = (\sqrt{b_1^2 - 4a_1(c_1 - s)})/2a_1.$$

The limits of the integration now become

$$z_0 = (\sqrt{b_1^2 - 4a_1(c_1 - d_{\text{max}})})/2a_1$$

and $z_1 = (\sqrt{b_1^2 - 4a_1(c_1)})/2a_1$ where $d_{\text{max}}$ is the peak delay noise in DCC. The first moment of the first term in the PDF of delay noise is given by

$$M_1 = I_1(z_1) - I_1(z_0),$$

(17)

where $I_1(z)$ is the indefinite integral given by the following,

$$I_1(z) = \frac{1}{2a_1} \int e^{-(b_1z + 2a_1\mu z)/2a_1} \cdot (c_1 + b_1\mu + a_1(s^2 + \mu^2)) \cdot \text{Erf} \left[ \frac{b_1 - 2a_1\mu + 2a_1z}{2\sqrt{2a_1s}} \right] ds.$$

Similarly, the second moment of the first term in the PDF of delay noise can be computed as

$$M_2 = I_2(z_1) - I_2(z_0),$$

(18)

where the indefinite integral $I_2(z)$ is given by

$$I_2(z) = \frac{1}{8a_1^2} \int e^{-(b_1^2 + 4a_1\mu z + 4a_1\mu^2 + 4a_1z^2)/2a_1} \cdot \left[ (b_1^2 + 4a_1\mu z + 4a_1\mu^2 + 4a_1z^2) \right] \cdot \text{Erf} \left[ \frac{b_1 - 2a_1\mu + 2a_1z}{2\sqrt{2a_1s}} \right] ds.$$