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 which verifies the accuracy of our approach.

1. Introduction

Imprecise control of lithography equipment and channel doping leads to a significant variability in device dimensions and threshold voltages. In nanometer regime, the variability in manufacturing process has not scaled commensurate with the device dimensions. Therefore, the variability of circuit performance has been rapidly increasing as we continue shrinking device dimensions. To account for the variability of circuit performance 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, statistical static timing analysis (SSTA) 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 SSTA can be classified as either path-based or block-based. Path-based SSTA 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 SSTA ([4],[5],[12]) requires only a single PERT-like traversal of the circuit graph and is more efficient than path-based SSTA.

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 variation leads to variations in delay 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 SSTA, 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 SSTA 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 SSTA 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 timing analysis tool.

A canonical first order model was used in ([5],[6]) which captures the first order sensitivities of delay to the (normally distributed) sources of variation and preserves the correlation information among all timing quantities. In this work, we show how the common 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 preserving 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 more 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

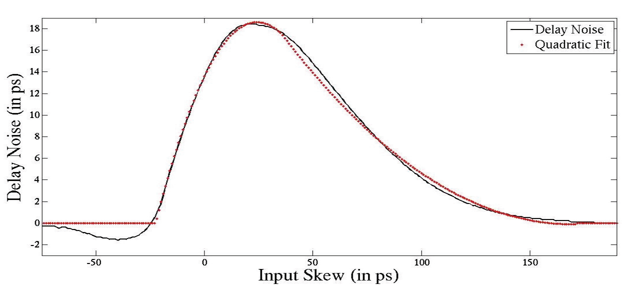

Figure 1. Delay change curve captures the dependence of delay noise on input skew

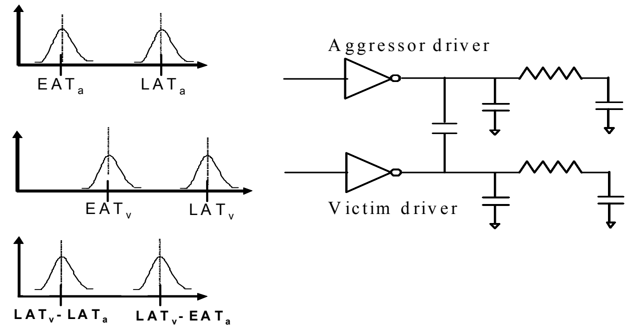

Figure 2. Skew window obtained by subtracting the early and late aggressor inp. arrival times from the late victim inp. arrival time.

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

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

where d_0 is the nominal delay, s_i is the of sensitivity of delay to the process parameter p_i which are assumed to be normally distributed and R is the random component. Principal Component Analysis (PCA) can be used to transform the set of process parameters into a set of mutually independent normal random variables. In block-based SSTA, the early and late arrival time distributions are propagated by performing statistical max and min 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 distribution. Note that the use of a skew window can lead to pessimistic bound 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 contributes to variability of the delay noise. For instance, it has been reported in [12], that interconnect variation causes about 10% ($3\sigma/\mu$) variability in the magnitude of the peak noise voltage. However, 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 in a single iteration.

3. Statistical Delay Noise as a function of input skew

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. In other words, the relative ratios' of sensitivities of delay noise distribution is identical to that of the input skew 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

$$s = s_0 + s_1 \cdot x_1 + s_2 \cdot x_2 \tag{2}$$

having a mean s_0 and sensitivities s_1 and s_2 with respect 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

$$Cov(s, x_1) = s_1 \tag{3}$$

The delay noise obtained from the DCC has a quadratic dependence on the input skew s i.e.

$$d = a \cdot s^2 + b \cdot s + c \quad . \tag{4}$$

Substituting (2) in (4), we obtain

$$d = a \cdot (s_0 + s_1 \cdot x_1 + s_2 \cdot x_2)^2 + b \cdot (s_0 + s_1 \cdot x_1 + s_2 \cdot x_2) + c \tag{5}$$

The covariance of delay noise with parameter x_1 is given by

$$Cov(d, x_1) = E[(d - d_0) \cdot x_1] \tag{6}$$

Substituting (5) in (6), we obtain

$$Cov(d, x_1) = E\begin{bmatrix} as_1^2 x_1^3 + as_2^2 x_2^2 x_1 + (2a+b)s_1 x_1^2 + \\ (2a+b) \cdot s_2 x_2 x_1 + 2as_1 s_2 x_1^2 x_2 \end{bmatrix} \tag{7}$$

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

$$Cov(d, x_1) = E\left[as_1^2 x_1^3 + (2a+b)s_1 x_1^2\right] \tag{8}$$

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

$$Cov(d, x_1) = E[as_1^2 x_1^3] + E[(2a+b)s_1 x_1^2] \tag{9}$$

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

$$Cov(d, x_1) = (2 \cdot a + b) \cdot s_1 \tag{10}$$

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

$$Cov(d, x_2) = (2 \cdot a + b) \cdot s_2 \tag{11}$$

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

$$\frac{Cov(d, x_2)}{Cov(d, x_1)} = \frac{Cov(s, x_2)}{Cov(s, x_1)} = \frac{s_2}{s_1} \tag{12}$$

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. □

Given an input skew distribution, using Theorem 1 we can obtain the correlations of the delay noise distribution. To express the delay noise in canonical normal form, we need to compute the mean and 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 μ and standard deviation σ given by

$$\begin{aligned} f_s(s) &= N(\mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(s-\mu)^2}{2\sigma^2}\right) \\ \mu &= s_0 \\ \sigma &= \sqrt{(s_1^2 + s_2^2)} \end{aligned} \quad . \tag{13}$$

Suppose that the piece-wise quadratic DCC has the following functional form

$$d_{cc} = \left\{ \begin{array}{ccc} 0 & , & s < z_0 \\ a_1 s^2 + b_1 s + c_1 & , & z_0 \le s \le z_1 \\ a_2 s^2 + b_2 s + c_2 & , & z_1 \le s \le z_2 \\ 0 & , & s > z_2 \end{array} \right\} \tag{14}$$

Since the delay noise is a function of only the input skew, and the functional dependence is captured by the DCC, we can appeal to the

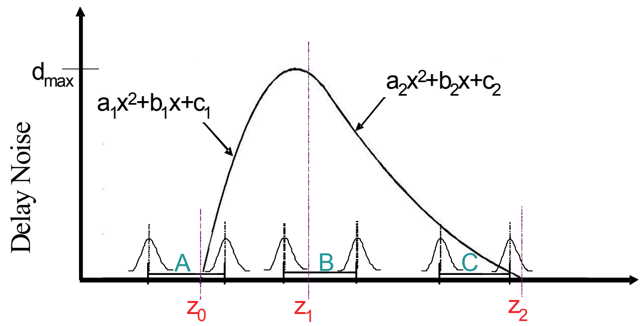

Figure 3. Delay change curve in piece wise quadratic form

basic theory of probability and statistics [14] to obtain the PDF of delay noise f_d as a function of the input skew distribution

$$f_d(s) = \frac{f_s(x_1)}{|2a_1x_1 + b_1|} + \frac{f_s(x_2)}{|2a_2x_2 + b_2|} \quad , \tag{15}$$

where x_1 and x_2 are the smaller and larger roots of the two quadratic pieces of the DCC, respectively

$$\begin{aligned} x_1 &= (-b_1 + \sqrt{b_1^2 - 4a_1(c_1 - s)})/2a_1 \\ x_2 &= (-b_2 + \sqrt{b_2^2 - 4a_2(c_2 - s)})/2a_2 \end{aligned} \quad . \tag{16}$$

Note, that the delay noise distribution obtained in (15) is not necessarily gaussian. However, as observed in [13], the delay distribution behaves like a normal distribution when the variance σ^2 of the input skew distribution is small and the mean μ falls on the "linear" part of the DCC. Using the PDF of delay distribution from (15), it is possible to compute the first and the second moment of delay noise in closed form **(**refer to **Appendix A).** The canonical form of delay noise can be constructed by matching the first two moments of delay noise obtained analytically. Correlations of the delay noise distribution are assigned by using the sensitivities of the input skew distribution to process parameters

4 Delay noise in SSTA framework

In the previous section, we obtained closed-form expressions for the mean and variance of the delay noise distribution given a single input skew distribution. It was also observed that the correlations in the input skew are preserved in the delay noise distribution. However, in block-based SSTA framework, we no longer have a single skew distribution at the input of the victim driver which can be used to compute the delay noise distributions. Instead, we have statistical timing windows at every node where the early and late arrival times are canonical distributions obtained by performing statistical min and max operations respectively. In this section, we propose the use of a statistical skew window for computing the delay noise distribution. To further reduce pessimism in our analysis, instead of using a single skew window, we propose the use of multiple smaller statistical skew windows. We now look at the computation of a skew window for an aggressor-victim pair and explain how it can be used to compute the corresponding delay noise distribution.

4.1 Delay noise from skew window

It has been shown in [15] that regardless of the aggressor transition, the worst-case delay noise occurs when the victim input transition occurs at the latest point in its timing window. In other words, adding the worst-case delay noise to the late victim input arrival time will result in the maximum slowdown (increase in the victim output 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. The arrival time distributions of end points of the skew window are referred to as early and late skew distributions.

The arrival times are normal random variables in canonical form. Note that the mean of the difference of the two normal distributions is given by the differences in their individual means. Therefore, the skew window that was obtained by using the early and late arrival times of the aggressor bounds the *mean* of any feasible input skew distribution. This is true because there exists no aggressor input arrival time distribution whose mean is greater than the mean of the late arrival time distribution, or less than the mean of early arrival time distribution. Since we always use the late victim arrival time distribution to compute the skew window, we conclude that the mean values of all feasible skew distributions must lie within the skew window.

We now look at how the skew window can be used to compute the distribution of delay noise. As shown in Figure 3, the skew window can align with the DCC in three different ways. Case *A* occurs when the mean of late skew distribution is less than the worst-case skew value of the DCC (viz. z_1). Case *B* occurs when the mean of late skew distribution is greater than z_1 and the mean of early skew distribution is less than z_1 . Lastly, Case *C* occurs when the mean of early skew distribution is greater than z_1 .

Since any skew distribution which lies within the skew window is feasible, for case *B*, the delay noise is modeled by its worst-case value d_{max} . Note that in Case *A,* the DCC is an increasing function of skew. Hence, for any feasible skew distribution with a *variable* mean and *fixed* variance, the corresponding (mean) delay noise will keep increasing as we keep shifting the mean of skew distribution to the right. Therefore, the mean delay noise will be maximized when the mean of the feasible skew distribution coincides with that of the late skew distribution. Using the above fact, we propose to use the analytical results from Section 2 and the late skew distribution to compute the delay noise distribution in canonical form. Similarly, for Case *C*, we use the early skew distribution to obtain the delay noise distribution. Thus, given the statistical timing windows from block-based SSTA, we can analytically compute the delay noise distribution. Also, since the delay noise is computed in canonical form, it can trivially be added to the late victim output arrival time distribution and propagated downstream in block-based SSTA.

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 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 current 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 (σ/μ) of 10% for the gate length variation.

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 10 ps 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 with 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.

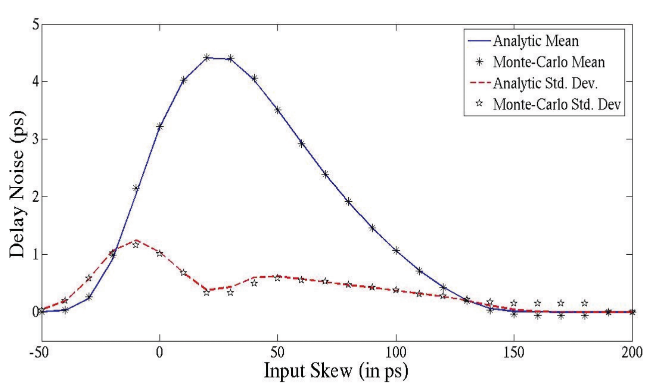

Figure 4. Comparison of analytical delay noise distribution with that obtained from Monte-Carlo simulations.

Table 1. Circuit Delay distribution accounting for Delay Noise in Block-Based SSTA.

ckt	Nominal Circuit Delay with No Delay Noise (in ps)		Circuit Delay with Worst-Case Delay Noise		Circuit Delay with Statistical Delay Noise (1 windows)		Circuit Delay with Statistical Delay Noise (10 windows)	
	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.
i1	458.56	12.59	678.7	11.9	576.1	11.3	554.6	11.50
i2	595.14	20.1	891.2	18.8	743.75	19.8	718.8	19.97
i3	445.4	3.93	659.1	3.63	564.1	2.94	543.9	3.68
i4	683.2	14.32	979.5	15.5	852.5	14.52	818.9	14.64
i5	780.9	5.48	1234.9	7.54	979.1	6.95	932.0	6.92
i6	734.4	18.5	1347.4	12.1	968.31	7.42	916.4	6.64
i7	701.5	28.7	1344.5	28.4	964.1	14.1	911.1	14.7
i8	836.2	25.4	1253.3	22.8	1052.3	23.2	998.3	21.4
i9	1047.3	44.78	1677.5	44.9	1251.7	47.8	1182.5	47.96
i10	2424.4	65.15	3136.2	58.7	2640.2	61.15	2558.8	61.25

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 *i*9, 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 *i*9 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 *i*9 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 correlations in input skew distribution are captured in the delay noise distribution and the ratio's of sensitivities of both distributions are identical. Using the correlation information, 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 expressions 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 interconnect variation while modeling delay noise.

References

[1] P. D. Gross, R. Arunachalam, K. Rajagopal and L.T. Pileggi,“Determining the Worst-Case Aggressor Alignment for Delay Calculation,” in Proc. of *International Conference on Computer Aided Design*, pp. 212-219, 1998.

[2] A. Agarwal, D. Blaauw, V. Zolotov, S. Sundareswaran, M. Zhou, K. Gala, and R. Panda,“Statistical delay computation considering spatial correlations,” in ACM/IEEE *Asia-Pacific Design Automation Conference*, 2003.

[3] M. Orshansky and K. Keutzer,“A general probabilistic framework for worst case timing analysis,” in Proc. of *Design Automation Conference*, pp. 556–561, 2002.

[4] A. Agarwal, D. Blaauw, and V. Zolotov, “Statistical timing analysis using bounds and selective enumeration,” in IEEE Trans. on *Computer-Aided Design of Integrated Circuits and Systems*, pp. 1243–1260, 2003.

[5] H. Chang and S. Sapatnekar,“Statistical timing analysis considering spatial correlations using a single pert-like traversal,” in Proc. of *International Conference on Computer Aided Design*, pp. 621-625, 2003.

[6] C. Visweswariah, K. Ravindran, K. Kalafala, S. G. Walker, S. Narayan, “First-order incremental block-based statistical timing analysis” in Proc. of *Design Automation Conference*, pp. 331-336, 2004.

[7] J. Le, X. Li and L. T. Pileggi, “STAC: Statistical Timing Analysis with Correlation,” in Proc. of *Design Automation Conference*, pp. 343-348, 2004.

[8] D. Sinha and H. Zhou, “A Unified Framework for Statistical Timing Analysis with Coupling and Multiple Input Switching,” in Proc. of *International Conference on Computer Aided Design*, pp. 837-843, 2005.

[9] R. Tayade, V. K. Kalyanam, S. Nassif, M. Orshansky and J. Abraham,“ Estimating Path Delay Distribution Considering Coupling Noise,” in Proc. of *ACM Great Lakes symposium on VLSI*, pp. 61-66, 2007.

[10]T. Sato, Y. Cao, D. Sylvester and C. Hu, “Characterization of Interconnect Coupling Noise using In-situ Delay Change Curve Measurements,” in Proc. of *IEEE ASIC/SOC Conference*, pp. 321-325, 2000.

[11]K. Agarwal, T. Sato, Y. Cao, D. Sylvester and C. Hu, “Efficient Generation of Delay Change Curves for Noise-Aware Static Timing Analysis,” in Proc. of *Design Automation Conference*, pp. 342-348, 2004.

[12]K. Agarwal, M. Agarwal, D. Sylvester and D. Blaauw, “Statistical Interconnect Metrics for Physical Design Optimization” in IEEE Trans. of *Computer Aided Design of Integrated Circuits and Systems*, pp. 1273-1288, 2006.

[13]A. B, Kahng, B. Liu, X. Xu, “Statistical Gate Delay Calculation with Crosstalk Alignment Considered” in Proc. of *ACM Great Lakes symposium on VLSI*, pp. 223-228, 2006.

[14]A. Papoulis, S. U. Pillai, “Probability, Random Variables and Stochastic Processes”, ISBN 0073660116, 4th edition, 2002.

[15]R. Gandikota, K. Chopra, D. Blaauw, D. Sylvester, M. Becer and J. Geada,“Victim Alignment in Crosstalk Aware Timing Analysis.” in Proc. of *International Conference on Computer Aided Design,* 2007

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 $a_1 s^2 + b_1 s + c$ 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

$$\begin{array}{c} z_0 = (\sqrt{b_1^2 - 4a_1(c_1 - d_{max})})/2a_1 \\ z_1 = (\sqrt{b_1^2 - 4a_1 c_1})/2a_1 \end{array},$$

where d_{max} is the peak delay noise in DCC. The first moment of the first term in the PDF of delay noise is given by

$$M_1 = I_1(z_1) - I_1(z_0) \tag{17}$$

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

$$I_1(z) = \frac{1}{2\sqrt{2\pi}} e^{-\left(\left(\frac{(b_1 + 2a_1\mu)^2}{a_1} + 4z^2\right)/8s^2\right)} \cdot \Bigg(e^{-((b_1 z + 2a_1 \mu z)/2a_1 s^2)} \cdot s \cdot (b_1 + 2a_1(\mu + z)) + e^{-\left(\frac{(b_1^2 + 4a_1 b_1 \mu + 4a_1^2(\mu^2 + z^2))}{8a_1^2 s^2}\right)} \cdot \sqrt{2\pi} \cdot (c_1 + b_1\mu + a_1(s^2 + \mu^2)) Erf\left[\frac{-b_1 - 2a_1\mu + 2a_1 z}{2\sqrt{2}a_1 s}\right]\Bigg)$$

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

$$M_2 = I_2(z_1) - I_2(z_0) , \tag{18}$$

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

$$I_2(z) = \frac{1}{8a_1\sqrt{2\pi}} e^{-\left(\left(\frac{(b_1 + 2a_1\mu)^2}{a_1^2} + 4z^2\right)/(8s^2)\right)} \cdot \Bigg(e^{-\left(\frac{(b_1 + 2a_1\mu)z}{2a_1 s^2}\right)} \cdot s \cdot \left(\begin{array}{l} -b_1^3 + 2a_1 b_1^2(\mu - z) + 4a_1 b_1(2c_1 + a_1(5s^2 + 3\mu^2 + 2\mu z + z^2)) \\ \quad + 8a_1^2(2c_1(\mu + z) + a_1(\mu^3 + \mu^2 z + \mu z^2 + z^3 + s^2(5\mu + 3z))) \end{array} \right) + 4a_1 e^{\left(\frac{(b_1^2 + 4a_1 b_1 \mu + 4a_1^2(\mu^2 + z^2))}{8a_1^2 s^2}\right)} \cdot \sqrt{2\pi} \cdot (c_1^2 + b_1^2(s^2 + \mu^2) + 2a_1 b_1 \mu(3s^2 + \mu^2) + a_1^2(3s^4 + 6s^2\mu^2 + \mu^4) + 2c_1(b_1\mu + a_1(s^2 + \mu^2))) Erf\left[\frac{-b_1 - 2a_1\mu + 2a_1 z}{2\sqrt{2}a_1 s}\right]\Bigg)$$