

# Statistical Timing Analysis Considering Spatial Correlations

Hong Li, Cheng-Kok Koh, Venkataramanan Balakrishnan and \*Yiran Chen  
School of Electrical and Computer Engineering  
Purdue University, West Lafayette, IN 47907-1285  
\*Synopsis Inc., Mountain View, CA 94043  
{li73,chengkok,ragu}@ecn.purdue.edu  
Yiran.chen@synopsis.com

## Abstract

*In this paper, we present an efficient algorithm to predict the probability distribution of the circuit delay while accounting for spatial correlations. We exploit the structure of the covariance matrix to decouple the correlated variables to independent ones in linear-time, as opposed to conventional techniques which have a cubic-time complexity. Furthermore, we present a closed-form expression for the probability distribution of the max operation, based on which we propose a fast and accurate approximation technique. Experiments show that the proposed method is both accurate and efficient.*

## 1 Introduction

With the increasing impact of process and environmental variations on deep-submicron designs, prediction of circuit performance is becoming a challenging task. Many statistical timing analysis (STA) techniques have been proposed. These techniques model delays as statistically distributed random variables, and timing analysis is performed using these distribution functions. Most of the earlier work has ignored spatial correlations by assuming zero correlations among different gate delays. However, as shown in [4, 7], considering correlation information is essential for an accurate timing analysis. Recently, several STA methods have been proposed to consider spatial correlations. The method in [9] computes path correlations from pair-wise gate delay covariances, and derives lower and upper bounds for circuit delays. The approach in [2] computes the delay distribution of a specific critical path considering spatial correlations. In both of the above path-based methods, the number of critical paths can be exponential. The approach proposed in [4] decomposes the correlated process parameters into a possibly smaller set of uncorrelated principal components by

using eigen-decomposition and circuit delay is calculated by a PERT-like traversal based on these fixed bases. This method avoids the exponential number of paths. A large sized covariance matrix is needed to achieve a high accuracy. Eigen-decomposition of such a covariance matrix requires a high computation complexity (cubic time complexity). Moreover, the reduction in the number of variables depends on the problem in consideration.

In this paper, we develop a linear-time complexity technique to decompose the correlated random variables and derive a closed-form expression for the distribution of the max function. Numerical experiments show that the proposed technique is both accurate and efficient. The remainder of this paper is organized as follows. Section 2 describes the fast decomposition algorithm. Section 3 derives a closed-form expression for the probability distribution of the max operation. Experimental results are presented in Section 4.

## 2 Decomposition of Correlated Variables

As a consequence of process variations, the process parameters are modeled as random variables. Therefore, the delays, which are functions of the parameters, become random variables. While in general, delays can have arbitrary distributions, practical considerations necessitate simpler approximations. The studies in [2, 4, ?] demonstrated that circuit delays can be accurately approximated by normal distributions. Moreover, the authors in [7] have shown in a detailed study that approximating the gate delays and circuit delays by normal distributions can greatly simplify the computation with negligible loss of accuracy, especially when compared with the error incurred by the loss of correlation information. Therefore, modeling delay as spatial correlated variables is essential for achieving an accurate analysis. In this section, we propose a linear-time technique to decompose the correlated random variables by using the structure of inverse of the covariance matrix, with the im-

plication that statistical timing analysis considering spatial correlations can be implemented very efficiently.

## 2.1 Inverse of covariance matrix

The fundamental property of spatial correlation is that the correlation is a function of distance. In particular, it is typical that gates in close proximity exhibit stronger correlations [11].

To model the spatial correlations, the chip of the circuit is covered by a grid [11] so that the parameter variations within a cell are identical and the parameter variations in different grid cells exhibit spatial correlations. Thus, the finer the grid, the higher the accuracy. If all the grid cells are ordered in such a way that cells close in space are also close in the ordering, strong correlations are close to the diagonal entries in the covariance matrix. When the correlations decrease rapidly as the distance increases, the corresponding covariance matrix has a “band” structure [11].

Here, we shall reveal that in timing analysis problems, the inverse of the covariance matrix is actually much sparser than the covariance matrix. The reason is that while covariance matrix gives the information about joint distribution, the inverse of covariance matrix provides the information about conditional distribution [3], i.e., the correlations between well-separated random variables when nearby random variables are given. As gates are more correlated with gates nearby, given the information about the gates nearby, the dependence between this gate and the other gates will be reduced significantly. Hence, off diagonal entries in the inverse of covariance matrix, which correspond to the conditional correlations, decrease much faster than those in the covariance matrix, resulting in a sparser inverse.

To illustrate the sparsity of inverses of covariance matrices, we take for example the covariance model proposed in [11],

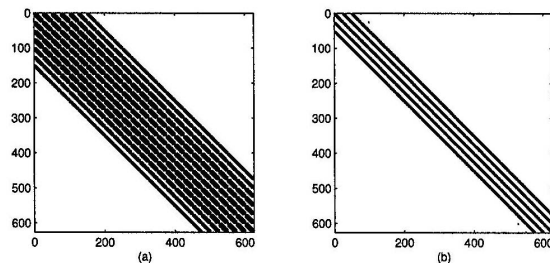
$$\text{cov}(G_i, G_j) = \sigma_G^2 \exp\left(-\frac{r_{ij}}{r_c}\right), \quad (1)$$

where  $r_{ij}$  is the distance between cell  $i$  and cell  $j$ , and  $r_c$  is called spatial correlations distance.  $r_c$  depends on the user defined resolution. As shown in Fig. 1(a) where  $\frac{r_c}{r_{i,i+1}} = 1$  and in Fig. 2(a) where  $\frac{r_c}{r_{i,i+1}} = 1.5$ , the corresponding matrices are denser than their inverses, which are shown in Fig. 1(b) and Fig. 2(b) respectively.

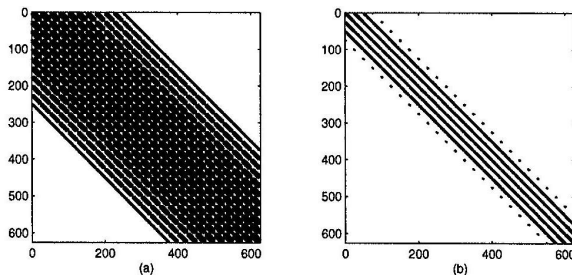
Consider another example of the covariance model,

$$\text{cov}(G_i, G_j) = \frac{1}{\alpha \cdot r_{ij}},$$

where  $\alpha$  is a constant scaler and  $r_{ij}$  is the same as above. The bigger the  $\alpha$  is, the faster the correlation decreases with the distance. As shown in Fig 3, even though the covariance matrix is dense, its inverse is sparse with a band structure.



**Figure 1. Covariance matrix with relative correlation distance being 1. (a) Sparsity pattern for covariance matrix. (b) Sparsity pattern for inverse of covariance matrix.**



**Figure 2. Covariance matrix with relative correlation distance being 1.5. (a) Sparsity pattern for covariance matrix. (b) Sparsity pattern for inverse of covariance matrix.**

In the following subsection, we will illustrate a novel method that uses the sparsity of the inverse of the covariance matrix to decouple the correlated variables. To reduce computation further, the inverse of covariance matrix can be approximated with a smaller bandwidth.

## 2.2 Cholesky factorization

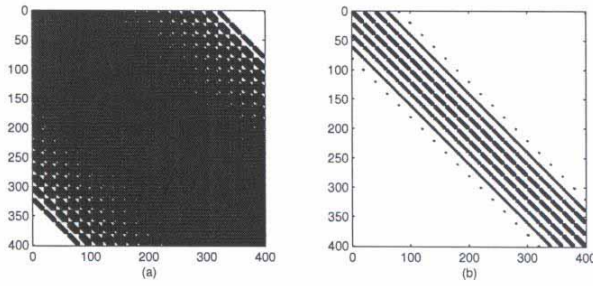
The Cholesky factorization of a symmetric positive definite matrix  $A$  computes the upper triangular matrix  $U$  such that  $A = U^T U$ . Denote  $U = \text{chol}(A)$ .

Consider mutually independent random variables  $x_i$ ,  $i = 1 \dots n$  with zero means and unit variances. Denote  $X$  as a vector whose  $i$ th entry is  $x_i$ . Then the covariance matrix for  $Y = U^T X$  is

$$\text{cov}(Y) = E(YY^T) = E(U^T X X^T U) = U^T E(X X^T) U = U^T U,$$

where  $E(\cdot)$  denotes expectation and  $\text{cov}(X) = E(X X^T)$  is an identity matrix due to the independence of  $x_i$ .

Therefore, given a set of random variables  $y_i$  with a covariance matrix  $A$ , Cholesky factorization of  $A$  transforms  $y_i$



**Figure 3. (a) Sparsity pattern for covariance matrix. (b) Sparsity pattern for inverse of covariance matrix.**

into a set of mutually independent random variables  $x_i$  with zero means and variances 1. Cholesky factorization is always feasible here, because covariance matrix is symmetric positive definite [10].

We will now present a fast algorithm to parameterize the Cholesky factorization of matrices with banded inverses. As a banded matrix can be treated as a block-tridiagonal matrix, we focus on the Cholesky factorization of the latter matrix here. Due to limited space, we skip the proof here. Refer to author's web [1] for the detailed proof for the following theorem.

**Theorem 2.1** Let  $[A_{ij}]$ ,  $i, j = 1, \dots, n$  be a symmetric positive definite block matrix with a block tridiagonal inverse, then

$$\text{chol}(A) = \begin{bmatrix} D_1 & D_1 R_1 & D_1 R_1 R_2 & \cdots & D_1 R_1 \cdots R_{n-1} \\ & D_2 & D_2 R_2 & \cdots & D_2 R_2 \cdots R_{n-1} \\ & & \ddots & \cdots & \cdots \\ & & & \ddots & \cdots \\ & & & & D_n \end{bmatrix} \quad (2)$$

and

$$D_1 = \text{chol}(A_1),$$

$$R_i = A_{ii}^{-1} A_{i,i+1}, \quad i = 1, \dots, n-1,$$

$$D_i = \text{chol}(A_{ii} - R_{i-1}^T A_{i-1,i-1} R_{i-1}), \quad i = 2, \dots, n.$$

Hence, the Cholesky factorization can be used to decompose the correlated delay variables  $y_i$ ,  $i = 1, \dots, n$  into a combination of a set of independent random variables, denoted as  $x_j$ , such that

$$y_i = \text{mean}_{x_i} + \sum_{j=1}^n a_{ij} x_j. \quad (3)$$

Let  $m$  be the block size of  $A_{ii}$ . From Theorem 2.1, sequences  $\{D_i\}_{i=1}^n$  and  $\{R_i\}_{i=1}^{n-1}$  can represent the Cholesky factorization for matrix  $A$ . The total number of parameters in  $D_i$  and  $R_i$  is  $(2n-1)m^2$ , which is the same as the number of nonzero entries in  $A^{-1}$ . By Theorem 2.1, calculating each  $D_i$  or  $R_i$  takes  $O(m^3)$  operations. Therefore, the total number of operations required for Cholesky factorization is  $O((2n-1)m^3)$ . When compared to the standard eigen-decomposition technique for PCA [4] which takes  $O((nm)^3)$ , it is evident that the exploitation of the sparsity of the inverse of the covariance matrix offers computational saving of two order of magnitude when  $n \gg m$  (i.e., when the band width is small).

### 3 Statistical Timing Analysis

After the application of the fast decomposition technique in Section 2, all the nodes and edges of a statistical timing graph are represented by a set of mutually independent normal random variables. This graph is then traversed in a breadth-first manner to find the statistically longest path. Propagating the arrival times through the gates involves sum and max operations. As multi-variable cases are easily broken down in to multiple two-variable operations, we consider only two variable operations in the sequel.

Computing the mean and variance for two normal random variables is straightforward. If  $X$  and  $Y$  are normally distributed, then  $Z = X + Y$  is also normal with mean  $\text{mean}(Z) = \text{mean}(X) + \text{mean}(Y)$  and  $\text{var}(Z) = \text{var}(X) + \text{var}(Y) + 2\text{cov}(X, Y)$ .

#### 3.1 Max of two normal random variables

The maximum of two normally distributed random variables is no longer normal. In the following, we derive a closed-form expression for the distribution of  $Z = \max(X, Y)$ .

Let  $X$  and  $Y$  be normally distributed with means  $\mu_x, \mu_y$ , variances  $\sigma_x^2$  and  $\sigma_y^2$ , and correlation coefficient  $\rho$ . We will use the notation  $\varphi(x) = (2\pi)^{-1/2} \exp(-x^2/2)$  to denote the probability density function (PDF) of the zero-mean unit-variance normal random variable and  $\Phi(x) = \int_{-\infty}^x \varphi(t) dt$  to denote its cumulative distribution function (CDF). The joint PDF of  $X$  and  $Y$  is

$$f_{XY}(x, y) = \frac{1}{2\pi\sigma_x\sigma_y(1-\rho^2)^{1/2}} \times \exp\left(\frac{-1}{2(1-\rho^2)} \left[ \left(\frac{x-\mu_x}{\sigma_x}\right)^2 - 2\rho \frac{(x-\mu_x)(y-\mu_y)}{\sigma_x\sigma_y} + \left(\frac{y-\mu_y}{\sigma_y}\right)^2 \right]\right). \quad (4)$$

With  $Z = \max(X, Y)$ , the probability that  $Z \leq z$  is

$$P(Z \leq z) = P(X \leq z, Y \leq z) = \int_{-\infty}^z \int_{-\infty}^z f_{XY}(x, y) dx dy.$$

Hence, the PDF of  $Z$  is given by

$$f_Z(z) = \frac{dP(Z \leq z)}{dz} = \int_{-\infty}^z f(z, y) dy + \int_{-\infty}^z f(x, z) dx. \quad (5)$$

Using straightforward manipulations, it can be shown that

$$\int_{-\infty}^z f(z, y) dy = f_X(z) \phi \left( \frac{z - (\mu_y - \frac{\sigma_y}{\sigma_x} \rho(z - \mu_x))}{\sqrt{(1 - \rho^2) \sigma_y}} \right),$$

where  $f_X(z) = \phi \left( \frac{z - \mu_x}{\sigma_x} \right)$ , i.e, the PDF of  $X$ .

Similarly,

$$\int_{-\infty}^z f(x, z) dx = f_Y(z) \phi \left( \frac{z - (\mu_x - \frac{\sigma_x}{\sigma_y} \rho(z - \mu_y))}{\sqrt{(1 - \rho^2) \sigma_x}} \right),$$

where  $f_Y(z) = \phi \left( \frac{z - \mu_y}{\sigma_y} \right)$ , i.e, the PDF of  $Y$ .

Upon further simplification,

$$f_Z(z) = \lambda_1(z) f_X(z) + \lambda_2(z) f_Y(z), \quad (6)$$

where

$$\lambda_1(z) = \phi \left( \frac{Z(1 + \frac{\sigma_2}{\sigma_1} \rho) - (\mu_y + \frac{\sigma_2}{\sigma_1} \rho \mu_x)}{\sqrt{(1 - \rho^2) \sigma_2}} \right), \quad (7)$$

$$\lambda_2(z) = \phi \left( \frac{Z(1 + \frac{\sigma_1}{\sigma_2} \rho) - (\mu_x + \frac{\sigma_1}{\sigma_2} \rho \mu_y)}{\sqrt{(1 - \rho^2) \sigma_1}} \right).$$

When  $\rho \neq 0$ ,

$$\lambda_1(z) = \phi \left( \frac{z - \mu_y + \frac{\sigma_y / \sigma_x}{1/\rho + \sigma_y / \sigma_x} (\mu_y - \mu_x)}{\frac{\sigma_y}{(1/\rho + \sigma_y / \sigma_x)} \sqrt{1/\rho^2 - 1}} \right), \quad (8)$$

$$\lambda_2(z) = \phi \left( \frac{z - \mu_x + \frac{\sigma_x / \sigma_y}{1/\rho + \sigma_x / \sigma_y} (\mu_x - \mu_y)}{\frac{\sigma_x}{(1/\rho + \sigma_x / \sigma_y)} \sqrt{1/\rho^2 - 1}} \right).$$

Thus,  $f_Z$ , the PDF is a linear combination of two PDFs  $f_X$  and  $f_Y$ , with weights  $\lambda_1$  and  $\lambda_2$ . With this observation, we can analyze  $Z = \max(X, Y)$ , considering a few special cases:

- $\rho = 0$ . In this case,  $X$  and  $Y$  are independent, and we have

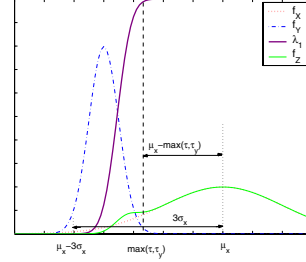
$$\lambda_1(z) = \phi \left( \frac{z - \mu_y}{\sigma_y} \right), \quad \text{and} \quad \lambda_2(z) = \phi \left( \frac{z - \mu_x}{\sigma_x} \right).$$

Suppose  $\mu_y \gg \mu_x$ . As  $\phi$  is a CDF, it is monotonically increasing, and it follows that  $1 > \lambda_2(z) \gg \lambda_1(z) > 0$ , and thus  $f_Z \approx f_Y$ . Similarly, when  $\mu_x \gg \mu_y$ ,  $f_Z \approx f_X$ . This conclusion makes intuitive sense: as  $X$  and  $Y$  are uncorrelated with well-separated means, the PDF of the one with the larger mean serves as a good approximation to the PDF of the maximum of the two.

- $\rho \neq 0$ ,  $\sigma_y \ll \sigma_x$ ,  $\mu_y < \mu_x$ . Then,  $\frac{\sigma_y / \sigma_x}{1/\rho + \sigma_y / \sigma_x} \approx 0$ , and therefore

$$\lambda_1(z) = \phi \left( \frac{z - \mu_y}{\frac{\sigma_y}{(1/\rho + \sigma_y / \sigma_x)} \sqrt{1/\rho^2 - 1}} \right).$$

Recalling that  $\phi$  is the CDF of the standard normal random variable, we note that as  $\sigma_y$  is small,  $\lambda_1(z)$  behaves like a unit-step function, with the zero-to-one transition occurring around  $\mu_y$ . Then, as  $\mu_y < \mu_x$ , we have, for  $z > \mu_y$ ,  $\lambda_1(z) \approx 1$ , or  $f_Z(z) \approx f_X(z)$ . Intuitively, the  $Y$  has a smaller mean with its PDF sharply peaked around the mean  $\mu_y$ , and therefore,  $\max(X, Y)$  has a PDF that follows  $f_X(z)$  for  $z > \mu_y$ .



**Figure 4.** Plot of  $z = \max(x, y)$  and the weight function.

We shall now propose a criterion for approximating max function. Suppose  $\mu_x > \mu_y$ . Denote the 99th percentile, i.e.,  $\mu + 3\sigma$ , for the distribution with CDF function  $\lambda_1$  as  $\tau$ . Hence, from (8),

$$\tau = \mu_y - \frac{\sigma_y / \sigma_x}{1/\rho + \sigma_y / \sigma_x} (\mu_y - \mu_x) + 3 \frac{\sigma_x}{(1/\rho + \sigma_x / \sigma_y)} \sqrt{1/\rho^2 - 1}.$$

For  $z > \tau$ ,  $\lambda_1 > 0.99$ , the weight function for  $f_X$  is very close to 1. Denote the 99th percentile for  $y$  as  $\tau_y$ ,

$$\tau_y = \mu_y + 3\sigma_y.$$

Hence,  $\int_{\tau_y}^{\infty} f_Y(z) < 0.1$ . As  $f_Y(z)$  is a PDF function that is always positive, it follows that  $f_Y(z)$  is very small when  $z > \tau_y$ . Therefore  $f_Z \approx f_X$  for  $z > \max(\tau, \tau_y)$ . We can also observe from Fig. 4 that when  $\max(\tau, \tau_y) < \mu_x$ ,  $f_Z$  matches  $f_X$  very well in the interval  $[\mu_x - \max(\tau, \tau_y), \infty]$ . Therefore, we use the ratio  $\frac{\mu_x - \max(\tau, \tau_y)}{3\sigma_x}$  as a measure to quantify how well  $f_Z$  matches  $f_X$ . When the ratio is larger than a threshold value, we approximate  $f_Z$  by  $f_X$  in our proposed method.

- For the remaining cases,  $\max$  function is approximated by a combination of the independent bases obtained from fast Cholesky factorization.

For example,  $x_i = \sum_{k=1}^n a_{ik} v_k$  where  $x_i$  is the delay for gate  $i$  and  $v_j, j = 1, \dots, n$  are independent bases. Due to the independence among the bases,  $\text{cov}(x_i, v_k) = a_{ik}$ ,  $\text{var}(x_i) = \sum_{k=1}^n a_{ik}^2$ . Similarly,  $x_j = \sum_{k=1}^n a_{jk} v_k$  and  $\text{cov}(x_j, v_k) = a_{jk}$ ,  $\text{var}(x_j) = \sum_{k=1}^n a_{jk}^2$ . Coefficients  $a_{ik}, a_{jk}$  are obtained from the Cholesky factorization of the covariance matrix for delays. The covariance between  $x_i$  and  $x_j$  can be calculated by  $\text{Cov}(x_i, x_j) = \sum_{k=1}^n a_{ik} a_{jk}$  since  $E(v_{k_1} v_{k_2}) = 0$  when  $k_1 \neq k_2$ . Given means, variances, and the covariance for two normal random variables, the exact mean and variance of  $z = \max(x_i, x_j)$  can be calculated using the formula in [5]. Moreover, knowing  $\text{cov}(x_i, v_k) = a_{ik}$  and  $\text{cov}(x_j, v_k) = a_{jk}$ , Coefficients  $a_k$  in  $z = \sum_{k=1}^n a_k v_k$  can also be calculated from the formula provided in [5] as  $a_k = \text{cov}(z, v_k)$ .

Based on all above analysis for the distribution of  $\max$ , our procedure to approximate  $\max(x_i, x_j)$  is as follows:

Calculating $z = \text{Max}(x_i, x_j)$
if $ \mu_x - \mu_y  > \epsilon_1$ (i.e., 3) && $\rho == 0$
$\max(x_i, x_j)$ = the one with larger mean ;
elseif $\frac{\mu_x - \max(\tau_x, \tau_y)}{3\sigma_x} > \epsilon_2$ (i.e., 0.5)
$\max(x_i, x_j)$ = the one with larger mean
else
calculate $\text{mean}_z$ and $\text{var}_z$ using [5];
calculate the coefficients $a_k = \text{cov}(z, v_k)$
from $\text{cov}(x_i, v_k) = a_{ik}$ , $\text{cov}(x_j, v_k) = a_{jk}$

## 4 Experimental results

The proposed algorithm was implemented in Matlab, and tested on ISCAS89 benchmark. All experiments were run on a PC with 1.5GHz CPU and 256MB memory. We use 250nm technology. For illustration purpose,  $L$  and  $W$  are random variables with the deviation 25% and 20% [8]. We take the finest grid so that every gate is contained in a different grid cell. The comparisons are made against 10,000-run Monte Carlo simulation.

In this experiment, we use the covariance matrix obtained from [11], which fits well with the measurement data on manufactured chips. Table 1 shows the means and variances from both methods. More precisely, we use two measures to illustrate the simulation results. First, we use normalized  $L^1$  norm [6],  $E_{pdf}$ , to measure the distance between PDF function  $f_{MC}(x)$  obtained from Monte Carlo simulation, and  $f_{FC}(x)$  obtained from our proposed methods. Secondly, we use the relative error in 99% quantile estimation  $E_\tau$ . In (9),  $\tau_{MC}$  and  $\tau_{FC}$  are denoted as 99% quantiles for

the distributions obtaining from Monte Carlo method and our proposed method.

$$E_{pdf} = \frac{\int |f_{MC}(x) - f_{FC}(x)| dx}{\int f_{MC}(x) dx} \%, \quad E_\tau = \frac{|\tau_{MC} - \tau_{FC}|}{\tau_{MC}} \%. \quad (9)$$

The results of the proposed method can be seen to be very close to the Monte Carlo results: the average distance of two PDF functions is 4%, and the average error in 99% quantile estimation is 0.3%. Table 1 shows run time comparison. We can see the proposed method is very fast. To show the importance of fine grid, we consider the difference between performing statistical timing analysis using fine grid and coarse grid. The error of coarse grid are also shown in Table 1. The average distance for PDF functions and the average error in 99% quantiles are 32% and 5% respectively. Both of them are about 10 times worse than those obtaining by fine grid. Fig 4 plots the CDF and PDF comparisons

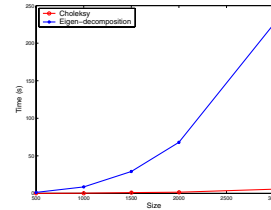


Figure 5. Run time comparison between Cholesky and Eigen-decomposition.

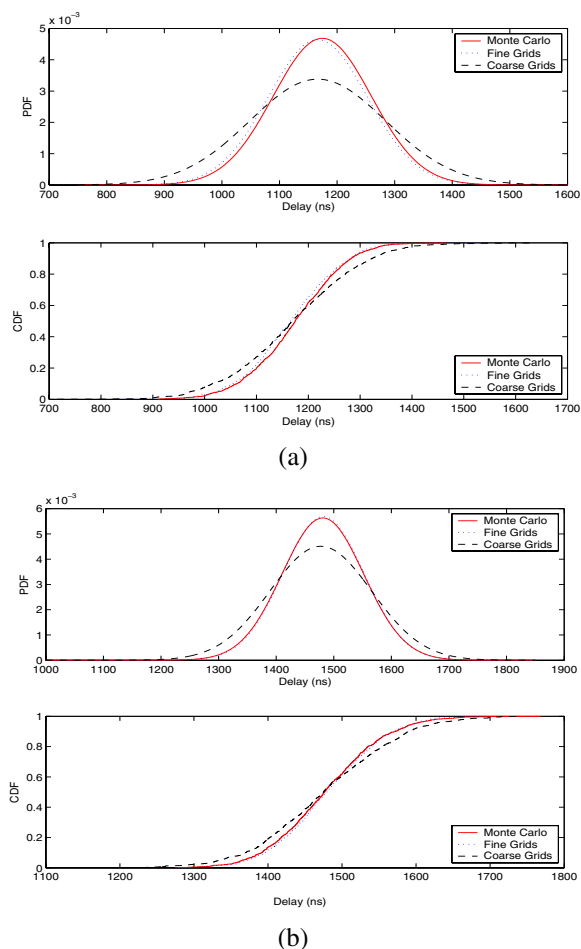
between using fine grids, coarse grids and Monte Carlo simulations for benchmark s1196 and s641. It can be observed that the distributions obtained using our method with fine grids match the distributions obtained from Monte Carlo simulations almost perfectly. For Coarse grid, although the mean value are close, the variance deviate significantly from those of Monte Carlo. This underlines the importance of developing fast and accurate STA tools that can handle large covariance matrix. Fig. 5 shows the improvement in run time to decompose the correlated random variables by using cholesky based decomposition method comparing with eigen-decomposition based method. For covariance matrix of size  $3000 \times 3000$ , the existing method takes 228 seconds to find the orthogonal basis whereas the proposed Cholesky decomposition only takes 5 seconds.

## References

- [1] <http://web.ics.purdue.edu/~li73/proofchol.htm>.
- [2] A. Agarwal, D. Blaauw, S. Sundareswaran, V. Zolotov, M. Zhou, K. Gala, and R. Panda. Path-based statistical timing analysis considering inter- and intra-die correlations. In *Proc. TAU*, pages 16–21, 2002.

**Table 1. Simulation comparison between Monte-Carlo method, and proposed method applied on fine and coarse grids.**

Bench name	Monte-Carlo (MC)			Fast Cholesky (FC)			Fine Grid Mismatch		Cells	Coarse Grid Mismatch	
	mean (ps)	SD (ps)	time (s)	mean (ps)	SD (ps)	time (s)	$E_{pdf}$	$E_{\tau}$	number	$E_{pdf}$	$E_{\tau}$
s27	777.1	49.6	92	777.6	48.4	0.3	2.4%	0.3%	4	38.5%	8%
s298	1949.4	93.6	7328	1947.6	93.1	11	1.5%	0.1%	50	32.8%	5%
s444	1788.7	75.5	11557	1788.5	74.38	21	1.4%	0.1%	65	38.8%	6%
s641	1174.3	85.1	13852	1166.4	85.9	53	7.4%	0.3%	100	37.8%	8.0%
s820	1413.4	69.1	10709	1403.2	68.4	15	11.8%	0.7%	55	37.4%	6%
s1196	1481.0	70.8	9223397	1482.1	70.1	41	1.5%	0.06%	100	21.7%	3%
s1238	1094.8	42.1	9220870	1092.2	41.8	40	4.9%	0.2%	100	12.0%	1%
s1423	1882.4	80.4	9289780	1881.0	79.6	162	1.6%	0.1%	100	36.9%	6%



**Figure 6. (a) Comparison of PDF and CDF by using fine and coarse grids for S641. (b) Comparison of PDF and CDF by using fine and coarse grids for S1196.**

- [3] J. A. Bilmes. Factored sparse inverse covariance matrices. In *International conference on Acoustics, speech and signal processing*, pages 1009–1012, 2000.
- [4] H. Chang and S. S. Sapatnekar. Statistical timing analysis considering spatial correlations using a single PERT-like traversal. In *Proc. Int. Conf. on Computer Aided Design*, pages 621–625, 2003.
- [5] C. E. Clark. Greatest of a finite set of random variables. *Operations Research*, 9:85–91, 1961.
- [6] R. E. Edwards. *Functional analysis: theory and applications*. Holt, Rinehart and Winston, 1965.
- [7] J. Le, X. Li, and L. T. Pileggi. STAC: Statistical timing analysis with correlation. In *Proc. Design Automation Conf.*, pages 343–348, 2004.
- [8] S. Nassif. Delay variability: Sources, impact and trends. In *Proc. ISSCC*, pages 368–369, 2000.
- [9] M. Orshansky and K. Keutzer. A general probabilistic framework for worst case timing analysis. In *Proc. Design Automation Conf.*, pages 556–561, 2002.
- [10] A. Papoulis. *Probability, random variables, and stochastic processes*. New York : McGraw-Hill, 1991.
- [11] L. Zhang, Y. Hu, and C. C. Chen. Statistical timing analysis with path reconvergence and spatial correlations. In *Proc. Design, Automation, and Test in Europe*, 2006.