Statistical Timing Analysis with Path Reconvergence and Spatial Correlations

Lizheng Zhang, Yuhen Hu, Charlie Chung-Ping Chen
ECE Department, University of Wisconsin, Madison, WI53706-1691, USA

E-mail: lizhengz@cae.wisc.edu,{hu,chen}@engr.wisc.edu

Abstract

State of the art statistical timing analysis (STA) tools often yield less accurate results when timing variables become correlated. Spatial correlation and correlation caused by path reconvergence are among those which are most difficult to deal with. Existing methods treating these correlations will either suffer from high computational complexity or significant errors.

In this paper, we present a new sensitivity pruning method which will significantly reduce the computational cost to consider path reconvergence correlation. We also develop an accurate and efficient model to deal with the spatial correlation.

1. Introduction

when technology scales down to nanometer regime, environmental and process parameter variation will significantly affect circuit performance. Traditional corner-based timing analysis will generally be too pessimistic. Statistical timing analysis(STA) that characterizes timing delays as statistical random variables as a function of these parameter variations offers a better approach for more accurate and realistic timing prediction.

In literatures, there are two distinct approach for STA: *path-based STA* and *block-based STA*. The fundamental challenge of the path-based STA [2, 7, 9, 10] is its requirement to select a proper subset of paths whose time constraints are statistically critical. This task has a computation complexity that grows exponentially with respect to the circuit size, and hence can not be easily scaled to handle realistic circuits.

This potential difficulty has motivated the development of block-base STA [1, 3–6, 11, 13] that champions the notion of *progressive computation*. Specifically, statistical timing analysis is performed block by block in the forward direction in the circuit timing graph without looking back to the path history. As such, the computation complexity of block based STA will grow linearly with respect to the circuit size.

However, to realize the full benefit of block based STA, one must solve a difficult problem that timing variables in a circuit could be correlated due to inter-parameter dependency, chip-to-chip variation, within-chip spatial dependency, path reconvergence of the circuit. In [13], a framework called *extended pseudo-canonical timing model* has been proposed to take all these correlations into account for accurate timing: for a circuit with N gates and M global variation sources:

$$\begin{aligned} X &= \mu_X + \sum_{i=1}^{N} \alpha_{X,i} R_i + \sum_{j=1}^{M} \beta_{X,j} G_j \\ &= \mu_X + \boldsymbol{\alpha}_X^* \boldsymbol{r} + \boldsymbol{\beta}_X^* \boldsymbol{g} \end{aligned} \quad (1)$$

where $\boldsymbol{r} = [R_1, \cdots, R_N]^*$ represents gate's localized independent variations and $\boldsymbol{g} = [G_1, \cdots, G_M]^*$s are global variations which may be correlated $E\{\boldsymbol{g}\boldsymbol{g}^*\} = \boldsymbol{\Sigma}$.

By including the local variation terms R_i in the timing model, all path recovergence correlations can be automatically addressed but the increasing number of local variation terms can potentially cause significant penalty in CPU time when big circuit is analyzed. By allowing the global variations to be correlated, the timing model in equation (1) provides the capability to adapt arbitrary global parameter dependency, including spatial correlation, into consideration. But it is not clear how to use this timing model to treat the important spatial correlation since the only existing spatial correlation model in literature, *quad-tree* model, has significant errors.

1.1. Path Reconvergence

As mentioned above, the penalty to consider the path reconvergence correlation using the extended

pseudo-canonical timing model is the worst-case timing complexity will be $O[N^2]$ instead of $O[N]$ since we have to traverse all local variation terms in equation (1) for every timing step. Authors in [13] propose to reduce the computation complexity by lumping small terms in the local variation sensitivity $\boldsymbol{\alpha}_X = [\alpha_{X,1}, \cdots, \alpha_{X,N}]^*$ into an independent term. However, the worst-case timing complexity under such simple lumping approach, is still quadratic with respecting to the circuit size. For example, for the circuit shown in figure 1, the lumping mechanism discussed in [13] will not work effectively since no local sensitivities can be lumped and the overall timing complexity will still be $O[N^2]$.

Figure 1. Circuit with linear structure has timing complexity $O[N^2]$ since no small local sensitivity terms can be lumped

1.2. Spatial Correlation

The only known spatial correlation model in literature is the so-called *quad-tree* model proposed in [1]. It covers the chip with grids and a structure of quad-tree is built to connect the grids cells together and the correlations between the parameters in the grid cells are represented by number of parent nodes they shared. However, this model might cause significant error since there are always nodes which are spatially very close to each other but belong to different subtrees in the quad-tree. So the correlation between these nodes might be significantly underestimated by this model.

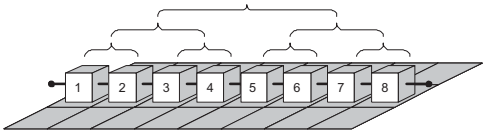


Figure 2. Quad-tree model underestimates the spatial correlation between wire segments 4 and 5

For example, if the *quad-tree* method is used to model the spatial correlation in an eight-segment straight wire, as illustrated in figure 2, the quad-tree will becomes a binary tree if the quad-partitioning is along the wire. It is obvious that the correlation between wire segments 2 and 3, 4 and 5 will be similar to that between wire segments 1 and 2 since they are similarly spatially separated. But according to the quad-tree method, the spatial correlation between 1 and 2 will be the largest, that between 3 and 4 will be second and that between 4 and 5 will be the smallest. So the *quad-tree* model fails to give similar spatial correlation when distance is similar.

1.3. Our Contribution

In this paper, we present solutions for both problems above. Specifically,

- By considering the fan-out number of gates, a fanout-based sensitivity pruning method, in addition to the original simple lumping method in [13], is developed to treat the correlation caused by path reconvergence and significant reduction in the number of local sensitivity terms is observed for benchmark circuits.
- We develop an *analytical* spatial correlation model without artificial errors as in *quad-tree* model. We propose to associate the grid size with the spatial correlation distance so as to efficiently integrate such an analytical model into statistical timing with the framework of [13].

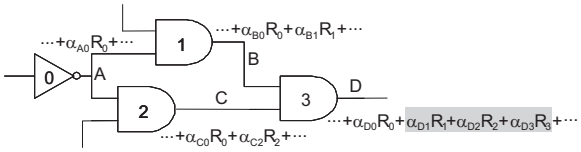
The rest of the paper is organized as following: Section 2 discusses the fanout-based sensitivity pruning method; Section 3 describes our analytical spatial correlation model and the method to integrate such model into the statistical timing analysis; Section 4 discusses the linear complexity of our timing algorithm with respecting to the circuit size and the total number of global variations; Section 5 presents a real implementation of our method in C/C++ and the testing result with ISCAS85 benchmark suites; Section 6 gives the conclusions.

2. Fanout-based Sensitivity Pruning

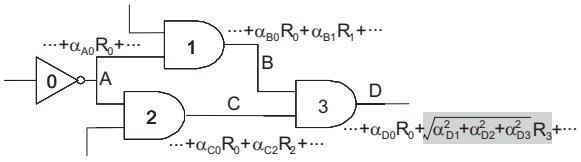
In previous section, we mentioned that the extended pseudo-canonical timing model will suffer from the quadratic computation complexity to consider the correlations caused by path recovergence even with the simple lumping method introduced in [13]. This problem, however, can be greatly alleviated by considering the fanout number of gates.

For example, for the partial circuit shown in figure 3, if all local sensitivities involved in arrival times A,B,C

and D are significant, then the simple lumping introduced in [13] will not reduce the number of the local sensitivity terms in the extended pseudo-canonical model. But if the fanout number of each gate is considered, some local sensitivity terms can still be pruned as illustrated in figure 3.



(a) Before fanout-based sensitivity pruning



(b) After fanout-based sensitivity pruning

Before fanout-based pruning
$A = \cdots + \alpha_{A0}R_0 + \cdots$
$B = \cdots + \alpha_{B0}R_0 + \alpha_{B1}R_1 + \cdots$
$C = \cdots + \alpha_{C0}R_0 + \alpha_{C2}R_2 + \cdots$
$D = \cdots + \alpha_{D0}R_0 + \alpha_{D1}R_1 + \alpha_{D2}R_2 + \alpha_{D3}R_3 + \cdots$
Average number of terms = 2.25

After fanout-based pruning
$A = \cdots + \alpha_{A0}R_0 + \cdots$
$B = \cdots + \alpha_{B0}R_0 + \alpha_{B1}R_1 + \cdots$
$C = \cdots + \alpha_{C0}R_0 + \alpha_{C2}R_2 + \cdots$
$D = \cdots + \alpha_{D0}R_0 + \sqrt{\alpha_{D1}^2 + \alpha_{D2}^2 + \alpha_{D3}^2}R_3 + \cdots$
Average number of terms = 1.75

Figure 3. Fanout-based sensitivity pruning for arrival times A,B,C and D in an example circuit

Specifically, when the arrival time D is computed, we can lump the local sensitivities α_{D1}, α_{D2} and α_{D3} together because any path reconvergence related with gates 1 and 2 will always involve gate 3 due to the fact that gates 1 and 2 only has ONE fanout to gate 3. In another word, the path information included in the local variation term R_3 will automatically imply the path information included in local variation terms R_1 and R_2. With such, lumping the local sensitivity of R_1 and R_2 into R_3 will reduce the path information redundancy and so that improve the computation efficiency. So such fanout-based sensitivity pruning will help gaining timing performance but will NOT hurt the timing accuracy since it is redundancy reduction instead of approximation.

The interesting thing we notice from the example circuit is that in the arrival time D, the local variation R_0 can also be lumped since gate 0's two fanouts have converged at gate 3. But it is difficult to decide such situation dynamically in timing analysis since it needs path trace which is computationally expensive. So the practical rule for fanout-based sensitivity pruning is:

> *If gate m has only one fanout to gate n, then perform sensitivity pruning when computing the arrival time at gate n's output as:*
>
> $$\alpha_{n,new} = \sqrt{\alpha_m^2 + \alpha_{n,old}^2}$$

It is also interesting to look back to section 1. For circuit shown in figure 1, our fanout-based sensitivity pruning will work most efficiently since every gate in the circuit will have only one fanout and so that they are all right pruning candidates. So such worst case for the original simple lumping method actually becomes the best case for the proposed fanout-based sensitivity pruning method.

3. Analytical Spatial Correlation Model

It has been widely known that the global parameters affecting the gate delays, such as gate length L, voltage supply V, temperature T etc., are not independent to each other. They might, on the other hand, correlate spatially, [8] i.e., devices nearby will have similar value of global parameters. So the fundamental property of the spatial correlation is that the correlation between the global parameters for gate at different positions will be a function of the distance between positions: the longer the distance, the smaller the correlation.

3.1. Exponential Spatial Correlation

Our approach for spatial correlation is to assume the correlation follows an analytical function of the distance. Such function can fundamentally be any analytical function that fits real calibration on silicon. As an illustrative example, here it is assumed to be an exponentially decay on the distances although the methodology here is not restricted to such exponential form. For a global parameter G, as illustrated in figure 4(a),

the covariance between the global variations at positions i and j will be:

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

where σ_G^2 is the variance of the considered global parameter, r_{ij} is the distance between positions i and j; constant r_c is the characteristic *spatial correlation distance* of the considered global parameter. The longer the r_c is, the stronger the spatial correlation.

3.2. Spatial Correlation Resolution

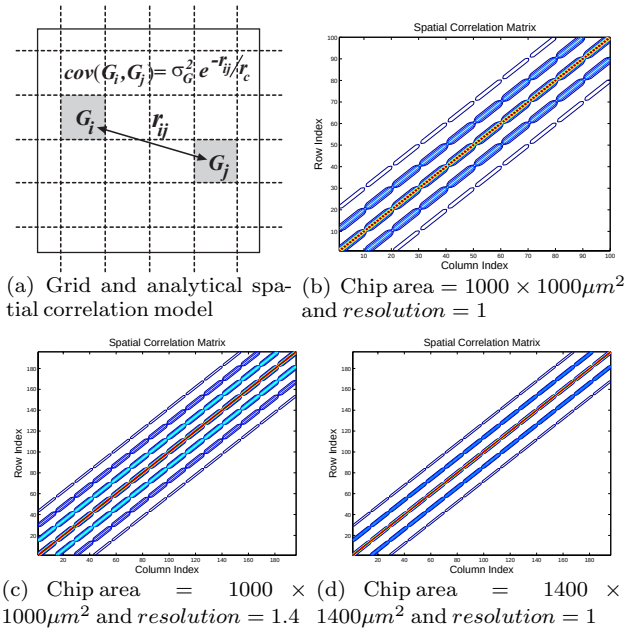
To incorporate such analytical spatial correlation model into statistical timing analysis, the chip of the circuit is covered by grids and each grid cell is assigned an individual random global variation for the considered global parameter as shown in figure 4(a). All gates in the grid cell share the same global variation as assigned if the considered global parameter affects the gate delay. Different global parameter may be associated with different grids since they may have significant different spatial correlation distance.

Intuitively, it is desired for high modeling accuracy to have very fine grid. But fine grid will result in large covariance matrix so that it is not beneficial for it will significantly degrade the performance of the timing analysis. The key strategy we proposed to make a good trade-off between such accuracy and performance is to decide the grid size based on the spatial correlation distance of the considered global parameter through a user-defined parameter of *resolution*:

$$correlation\_distance = resolution \times grid\_cell\_size$$

So fine grid is only applied when the correlation distance is short or a high resolution is demanded. Although it seems that fine grid is inevitable since there is no guarantee that correlation distance is always large, performance will still be reasonable because the sparsity of the covariance matrix will be controlled by the *resolution* parameter.

As shown in figure 4, the covariance matrix will always have a "band" structure where the number of bands in the matrix is decided by the user-defined parameter *resolution*. The higher the resolution, the more bands and the less sparsity of the matrix. With the same resolution, the number of significant elements in the covariance matrix is proportional to the number of global variations, i.e. the size of the covariance matrix.



(a) Grid and analytical spatial correlation model

(b) Chip area $= 1000 \times 1000\mu m^2$ and $resolution = 1$

(c) Chip area $= 1000 \times 1000\mu m^2$ and $resolution = 1.4$

(d) Chip area $= 1400 \times 1400\mu m^2$ and $resolution = 1$

Figure 4. Spatial correlation matrix for a global parameter with spatial correlation distance $r_c = 100\mu m$

4. Statistical Timing Complexity

If time variables are with the extended pseudo-canonical timing model as in equation (1), the variation and covariance between such time variables will be [13]:

$$\sigma_X^2 = \boldsymbol{\alpha}_X^* \boldsymbol{\alpha}_X + \boldsymbol{\beta}_X^* \boldsymbol{\Sigma} \boldsymbol{\beta}_X \quad (3)$$

$$cov(X, Y) = \boldsymbol{\alpha}_X^* \boldsymbol{\alpha}_Y + \boldsymbol{\beta}_X^* \boldsymbol{\Sigma} \boldsymbol{\beta}_Y \quad (4)$$

Consider the "band" structure of the covariance matrix $\boldsymbol{\Sigma}$ as show in figure 4, the complexity to compute the variance and covariance will be $O[\Gamma + M]$ where Γ is the average number of sensitivity terms in $\boldsymbol{\alpha}$ and M is the size of the $\boldsymbol{\Sigma}$ or the total number of global variations. Since such variance/covariance evaluation has to be done at each timing step, the overall complexity of statistical timing will be $O[(\Gamma + M)N]$ which is still linear the circuit size N since $\Gamma << N$ usually.

Since the number of bands in $\boldsymbol{\Sigma}$ is controlled by the user-defined parameter of resolution, the higher the resolution, the more bands in the matrix and the longer the variance/covariance computation time. On the other hand, the higher the resolution, the finer the grid, the more accurate of the correlation model. So the user-defined parameter *resolution* provides a good way to trade off accuracy and complexity in considering spatial correlation for statistical timing.

5. Simulation Results and Discussions

The above described algorithm has already been implemented in C/C++ and tested by ISCAS85 benchmark circuits. All tested circuits are synthesized using Design Compiler® from Synopsys®. Library cells used are characterized by Spice monte carlo simulation with Cadence®tools of Spectra® in $0.18\mu m$ technology and a $10\%(\sigma/\mu)$ variation is assumed in those process parameters. To model the spatial correlations, all tested circuits are placed by Dragon [12].

For illustration purpose, only three parameter variation are considered global: channel length(L), supply voltage(Vdd) and temperature(T). Although it is not necessary, their correlation distances are all assumed to be the same as $100\mu m$. All other variation sources specified in the $0.18\mu m$ technology file are assumed to be localized in the considered gate only.

Circuit Name	c432	c499	c880	c1335	c1908
Gate Counts	186	399	330	454	383
Γ without FBSP	22.7	2.6	3.7	4.1	4.1
Γ with FBSP	11.8	2.0	2.7	3.4	2.9
Γ reduction rate	48%	23%	27%	17%	29%
Circuit Name	c2670	c3540	c5315	c6288	c7552
Gate Counts	501	820	1237	2363	1777
Γ without FBSP	7.2	8.4	4.1	29.6	5.6
Γ with FBSP	4.4	5.0	2.7	20.9	3.8
Γ reduction rate	39%	40%	34%	29%	32%

Table 1. Average number of local sensitivity terms(Γ) w/o fanout-based sensitivity pruning(FBSP). The average Γ reduction rate = 32%

To test the effect of the proposed fanout-based sensitivity pruning method, all benchmark circuits are analyzed by out statistical timing engine with without the sensitivity pruning method and the average number of local sensitivity terms are summarized in table 1. In average, 32% of sensitivity reduction is achieved with the fanout-based pruning method. As discussed in section 2 and verified in table 2, such sensitivity reduction will not affect the accuracy of the timing analysis although we can expect significant performance for large circuits.

It has been mentioned in Section 4 that the timing complexity will be linear with respecting to the number of global variations if the spatial correlation is considered using our analytical model. This conclusion is clearly shown in figure 5 where different resolution is applied for circuit c6288 to get different number of spatially correlated global variations.

Circuit	Before FBSP		After FBSP	
	$\mu[ps]$	$\sigma[ps]$	$\mu[ps]$	$\sigma[ps]$
c432	1285	221	1287	219
c499	625	127	625	127
c880	802	158	802	158
c1355	790	189	790	189
c1908	931	207	931	207
c2670	973	138	974	139
c3540	1214	171	1214	172
c5315	920	125	919	125
c6288	4137	444	4137	444
c7552	1316	188	1317	187

Table 2. Fanout-based sensitivity pruning(FBSP) will not affect timing accuracy

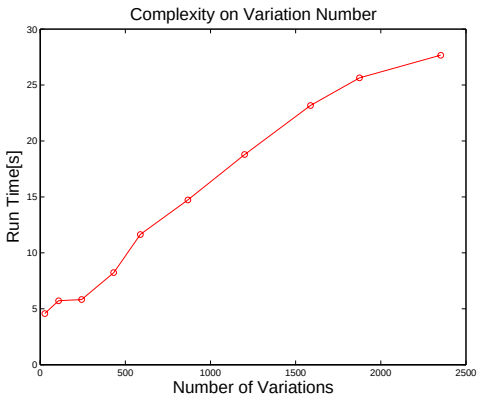


Figure 5. Timing Complexity with respecting to the number of global variations

6. Conclusions

To consider the correlation caused by path reconvergence, this paper presents a novel method to additionally reduce the timing complexity with no accuracy penalty by introducing a fanout-based local sensitivity pruning method,. In average, 32% of sensitivity term reduction is observed in ISCAS85 benchmark circuits.

An analytical spatial correlation model is also proposed to avoid the artificial error in the existing quadtree model. By constructing the grid covering the circuit according to the spatial correlation distance, the overall complexity of the timing analysis becomes linear with respecting to the total number of grid cells. A convenient parameter named *resolution* is used to make trade-off between the timing performance and spatial correlation accuracy.

7. Acknowledgement

This work was partially funded by TSMC, UMC, Faraday, SpringSoft, National Science Foundation under grants CCR-0093309 & CCR-0204468 and National Science Council of Taiwan, R.O.C. under grant NSC 92-2218-E-002-030. Also great thanks to professor Barry D. Van Veen for the great discussions.

References

[1] A. Agarwal, D. Blaauw, and V. Zolotov. Statistical timing analysis for intra-die process variations with spatial correlations. *Computer Aided Design, 2003 International Conference on. ICCAD-2003*, pages 900 – 907, Nov 2003.

[2] A. Agarwal, D. Blaauw, V. Zolotov, S. Sundareswaran, M. Zhao, K. Gala, and R. Panda. Statistical delay computation considering spatial correlations. *Design Automation Conference, 2003. Proceedings of the ASP-DAC 2003. Asia and South Pacific*, pages 271 – 276, Jan 2003.

[3] A. Agarwal, V. Zolotov, and D. Blaauw. Statistical timing analysis using bounds and selective enumeration. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 22(9):1243 –1260, Sept 2003.

[4] S. Bhardwaj, S. B. Vrudhula, and D. Blaauw. τau: Timing analysis under uncertainty. *ICCAD'03*, pages 615–620, Nov 2003.

[5] H. Chang and S. S. Sapatnekar. Statistical timing analysis considering spatial correlations using a single pert-like traversal. *ICCAD'03*, pages 621–625, Nov 2003.

[6] A. Devgan and C. Kashyap. Block-based static timing analysis with uncertainty. *ICCAD'03*, pages 607–614, Nov 2003.

[7] J.-J. Liou, A. Krstic, L.-C. Wang, and K.-T. Cheng. False-path-aware statistical timing analysis and efficient path selection for delay testing and timing validation. *Design Automation Conference, 2002. Proceedings. 39th*, pages 566 – 569, June 2002.

[8] S. R. Nassif. Modeling and analysis of manufacturing variations. *CICC*, pages 223–228, 2001.

[9] M. Orshansky. Fast computation of circuit delay probability distribution for timing graphs with arbitary node correlation. *TAU'04*, Feb 2004.

[10] M. Orshansky and K. Keutzer. A general probabilistic framework for worst case timing analysis. *Design Automation Conference, 2002. Proceedings. 39th*, pages 556 – 561, June 2002.

[11] C. Visweswariah, K. Ravindran, and K. Kalafala. First-order parameterized block-based statistical timing analysis. *TAU'04*, Feb 2004.

[12] M. Wang, X. Yang, and M. Sarrafzadeh. Dragon2000: standard-cell placement tool for large industry circuits. *ICCAD '00: Proceedings of the 2000 IEEE/ACM international conference on Computer-aided design*, pages 260–263, 2000.

[13] L. Zhang, W. Chen, Y. Hu, and C. C. Chen. Statistical timing analysis with extended pseudo-canonical timing model. *Design, Automation and Test in Europe, DATE'2005*, pages 952–957, 2005.