

PARADE: PARAMetric Delay Evaluation Under Process Variation^{*}

(Revised Version)

Xiang Lu[†], Zhuo Li[†], Wangqi Qiu[‡], D. M. H. Walker[‡], Weiping Shi[†]

[†]*Dept. of Electrical Engineering
Texas A&M University
College Station, TX 77843-3124
wshi@ee.tamu.edu*

[‡]*Dept. of Computer Science
Texas A&M University
College Station, TX 77843-3112
walker@cs.tamu.edu*

Abstract

Under manufacturing process variation, the circuit delay varies with process parameters. For delay test and timing verification under process variation, it is necessary to model the variational delay as a function of process variables. However, conventional methods to generate such functions are either slow or inaccurate. In this paper, we present a number of new methods for fast parametric delay evaluation under process variation. Our methods are based on explicit delay formulae or characterized lookup tables, and are significantly faster than conventional RSM method of comparable accuracy. Due to the efficiency of our method, we can accurately model any path delay as a function of multiple interconnect and device process variables in large circuits. Experimental results on ISCAS85 circuits show that the path delay error predicted by our methods is less than 3% of that computed by the RSM using SPICE, where the path delay variation is within $\pm 10\%$.

1. Introduction

With the shrinking feature size in VLSI technology, the impact of process variation is increasingly felt. To address the effect, great amount of research has been done recently, such as the clock skew analysis under process variation [1, 2, 3], statistical performance analysis [4, 5, 6], worst case performance analysis [7, 8], parametric yield estimation [10], impact analysis on micro architecture [10] and delay fault test under process variation [11, 12, 13, 14].

In all the above research, one important task is to compute variational path delay under process variation, either as functions of process variables [1, 2, 4, 8, 9, 14] or as random variables of certain distribution [3, 6, 12, 15]. However, the conventional methods to compute path delay are either slow or inaccurate. The response surface method (RSM), which performs multiple simulations and

curve-fittings, is used in [4, 8, 9, 15]. To achieve high accuracy, the RSM method must perform multiple parasitic extractions and delay evaluations under different process conditions. Due to the large number of metal layers in the modern technology, there are many interconnect process variables. For example, for a k -layer technology, there are $3k$ process variables related to interconnect, corresponding to the metal width, metal thickness and inter-layer dielectric thickness of each layer. As a result, the traditional RSM becomes prohibitive for large circuits. Orshansky *et al.* [7] derived delay sensitivity to gate length variation based on a simple model, and expressed delay as a function of gate length. Their method does not automatically apply to interconnect process variation due to the lack of a similar model for the interconnect. For methods that treat the path delay as a random variable, it is also necessary to extract parasitic RCs and compute the path delay under different process conditions [6, 12], resulting in too much timing cost.

In this paper, we present a new method PARADE for fast PARAMetric Delay Evaluation using analytical formulae and pre-characterized lookup tables. We first model variational path delays as linear functions of process variables. By analyzing a small sample of parasitic capacitance extracted from any commercial parasitic extraction tools, we derive an efficient method to compute the effect due to process variation for parasitic capacitance. Then the variational path delay is evaluated efficiently, based on the lumped C delay model and based on the effective capacitance delay model respectively. No multiple parasitic extractions and multiple delay evaluations are needed for both methods, resulting in a significant speedup over the traditional RSM. The efficiency of our methods makes it possible to comprehensively analyze circuit performance on all interconnect and device process variables for large circuits. We do experiments on ISCAS85 circuits under TSMC 180nm 1.8V 5-metal layer technology. Circuit layout generation is done by Cadence Silicon EnsembleTM and parasitic extraction is done by Assura HyperExtract parasitic extractor. Experiments show that our methods achieve high accuracy and efficiency. Compared to the

^{*}This research was supported in part by the SRC grant 000-TJ-844, NSF grants CCR-0098329, CCR-0113668, EIA-0223785, and ATP grant 512-0266-2001.

traditional RSM, the delay error is within 5% using analytical methods, and is within 3% using the table lookup method.

The paper is organized as follows. In Section 2, we present the linear delay model and propose the method to compute parametric delay based on the lumped C delay model and based on effective capacitance delay model respectively. In Section 3, we compare the performance of the new methods with RSM. The conclusion is given in Section 4.

2. Parametric Delay Evaluation

There are many forms of process variation, see for example Stine *et al.* [16] and Nassif [17]. In this paper, we consider the systematic process variation, such as the variation on gate length, and the variation of metal width, metal thickness, and inter-layer-dielectric (ILD) thickness related to each interconnect layer. Our methods can be extended to include other process variation such as the threshold voltage, the supply voltage and the temperature, as long as the approximated delay can be expressed as a linear function of the process variables within their variation ranges.

In order to calculate the path delay under process variation, we first compute the *buffer-to-buffer* delay. The *buffer-to-buffer* delay is defined as the delay from the input pin of a cell to the input pin of a downstream cell. After each *buffer-to-buffer* delay in the circuit is computed, the delay of any path can be easily obtained by adding up *buffer-to-buffer* delays along the path.

We approximate the *buffer-to-buffer* delay as a linear function of process variables:

$$d(\mathbf{x}, s) \approx d_0(s) + b_1(s)x_1 + b_2(s)x_2 + \dots + b_p(s)x_p, \quad (1)$$

where $d_0(s)$ is the nominal delay, $\mathbf{x}=(x_1, x_2, \dots, x_p)$ is the vector of process variables, each representing the deviation from the nominal value, s is the input signal slew, and $b_i(s)=\partial d/\partial x_i$ is the delay sensitivity to process variable x_i . We assume both the nominal delay and delay sensitivities are functions of input signal slew s .

The validity of the linear model is supported by extensive simulation. We performed multiple parasitic extraction and SPICE simulation under different process conditions. It is found that for any single process variation variable, its effect on delay is approximately linear within its small variation range. In Fig. 1 we show the SPICE simulation result on a *buffer-to-buffer* segment in the circuit for several typical process variation variables. Each variable changes within its typical range (metal width $\pm 5\%$, metal thickness $\pm 20\%$, ILD thickness 40%, and gate length $\pm 5\%$). In addition, since the systematic process variables in our consideration are determined at different stages of the manufacturing process, we can assume they are independent of each other. Furthermore, within the small variation range of

each variable, the effect of each variable is additive. For example, considering width variation on metal 2 and metal 3, we denote the delay variation under metal 2 width variation and under metal 3 width variation as Δd_{w2} and Δd_{w3} respectively, and denote delay variation under both variations happening as Δd_{w1+w2} . The width changes on both metal 2 and metal 3 are 5% of the nominal metal width. Then we use $\Delta d_{w1}+\Delta d_{w2}$ to approximate Δd_{w1+w2} . In Fig. 2, we show the error distribution over 160 *buffer-to-buffer* segments in circuit c432. From the figure, for most of *buffer-to-buffer* segments, the error is considerably small. Because of the effect of layer overlapping between metal 2 and metal 3 in the layout, the error is over 20% for few *buffer-to-buffer* segments. It is interesting to study more complex model to compensate for such segments and keep the additive property. Nevertheless, for most *buffer-to-buffer* segments the effect of metal 2 and metal 3 width variation can be considered as additive. The similar result is found in other process variables.

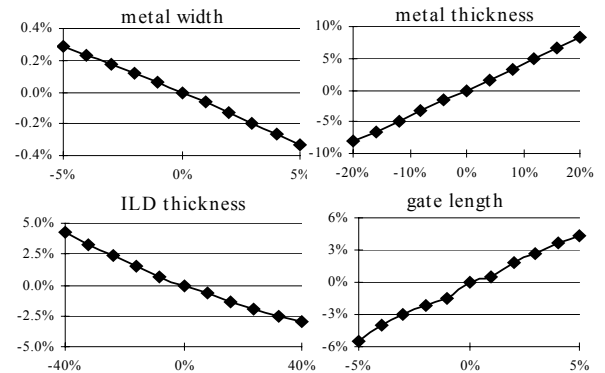


Fig 1. Delay variations due to process variation are linear in SPICE simulation. The x-axis indicates process variation and the y-axis indicates the percentage deviation from the nominal delay.

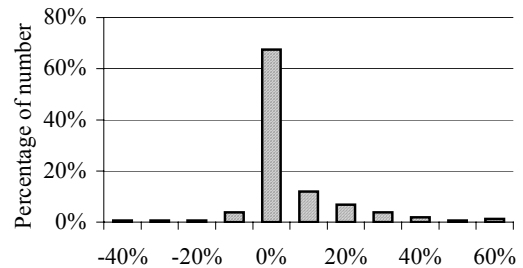


Fig 2. The delay effect of process variation is additive, which is demonstrated by the error distribution of approximating Δd_{w1+w2} with $\Delta d_{w1}+\Delta d_{w2}$ over 160 *buffer-to-buffer* segments in circuit c432.

The effect of signal slew has been studied in previous research, for example, in variational delay evaluation [17] and in static timing analysis [18]. In this paper, we

assume the effect of process variation on output signal slew is small and we propagate signal slews under the nominal process condition. The computation of nominal delay and signal slew can be done by any commercial tool, and is not the focus of this paper. The key issue is to efficiently compute delay sensitivities b_1, b_2, \dots, b_p .

A buffer-to-buffer segment in a circuit is represented by a cell driving an RC circuit, which consists of distributed R_1, \dots, R_n and distributed C_1, \dots, C_n on interconnect and sink capacitance C_s for each downstream cell. The RC circuit can be a tree-like structure or a path-like structure. Parasitic RCs are generated by commercial parasitic extraction tools, and each pair of parasitic (R_i, C_i) is related to one metal segment or a contact/via on interconnect.

In following sections, we first present an efficient method to compute variations of parasitic RCs and C_s due to process variation in section 2.1. Then in section 2.2, we delay sensitivity computation method based on lumped C delay model and based on effective capacitance delay model respectively.

2.1. Computation on RC Variations

The sink capacitance C_s is only related to device parameters of the downstream cell. In this paper, we ignore the variation of C_s . Thus $\partial C_s / \partial x_i$ is zero for all process variables.

Parasitic RCs on interconnect vary in different process conditions. The value of $\partial R_j / \partial x_i$ can be easily derived from the basic resistance computation formula $R = \rho L / (WT)$, where ρ is the resistive constant, L , W and T is the length, width and thickness the metal segment respectively.

However, it is more difficult to compute $\partial C_j / \partial x_i$. This is because the parasitic capacitance of a metal wire depends not only on the wire itself, but also on the neighboring condition. Formula-based methods for parasitic extraction are no longer used and are replaced by more accurate 2.5D/3D tools. For these tools, there is no explicitly capacitance formula we can use. To make our method widely applicable to different design flows, the computation of $\partial C_j / \partial x_i$ must be independent of any particular parasitic extraction tools. At the same time, we need to avoid multiple extractions on the whole circuit for different process variable.

To get $\partial C_j / \partial x_i$ for any process variable x_i efficiently and accurately under any complex neighboring condition, we introduce the concept of *unit capacitance variation* u_{ik} , which is an estimate of the percentage variation of parasitic capacitance on metal k , with respect to process variable x_i . In practice, we randomly choose n parasitic capacitance on metal k in a circuit, and calculate u_{ik} by:

$$u_{ik} = \frac{1}{n} \sum_j \frac{\Delta C_{jk} / \Delta x_i}{C_{jk}}, \quad (3)$$

where Δx_i is a small change of process variable x_i , C_{jk} indicates a parasitic capacitance on metal k under the nominal condition, and ΔC_{jk} is the variation of C_{jk} due to Δx_i .

For a given process technology, the value of $(\Delta C_{jk} / \Delta x_i) / C_{jk}$ is in a considerable small range. In Fig. 3 we show the distribution of $(\Delta C_{jk} / \Delta x_i) / C_{jk}$ due to the wire width variation on metal 2 in ISCAS85 circuit c432 for 406 sample capacitance. From the figure, we can see that for most parasitic capacitance C_{jk} , the value of $(\Delta C_{jk} / \Delta x_i) / C_{jk}$ is around 0.61 with small deviations.

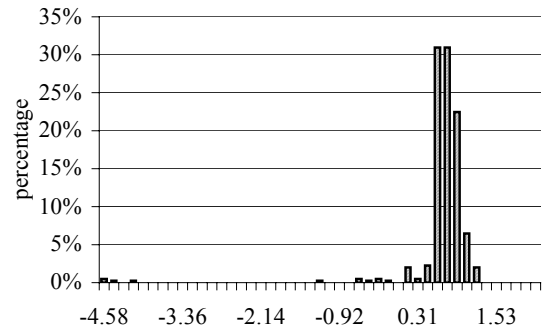


Fig 3. The distribution of $(\Delta C_{jk} / \Delta x_i) / C_{jk}$ due to metal 2 width variation on 406 samples in ISCAS85 circuit c432. The x-axis indicates values of $(\Delta C_{jk} / \Delta x_i) / C_{jk}$.

The unit capacitance variation u_{ik} is pre-computed for each metal layer with respect to each interconnect process variable, and is used to estimate the variation for any parasitic capacitance on metal k under a small change of x_i . For any C_j is on metal k , we have:

$$\partial C_j / \partial x_i = u_{ik} \cdot C_j. \quad (4)$$

2.2. Computation on Delay Sensitivity

We assume a k -factor table of delay with respect to input slew s and load C_L is given. If such a table is not available, we construct one using existing technology. The delay table under the nominal gate length is named as *nominal table*. We then build another k -factor table with the same indices of the first table, where each entry is the delay under a small change of gate length. The change of gate length ΔL_g is 3% of the nominal gate length in our experiments. This table is named as *variational table*.

We apply two delay models in delay sensitivity computation. One is lumped C delay model, and the other is effective capacitance delay model.

Lumped C delay model

In lumped C delay model, all parasitic resistance on

interconnect are removed, and all parasitic capacitance and the sink capacitance are lumped into one single load capacitance C_L . Then we have $C_L = \sum C_j$. Then we refer to the nominal table and generate delay d according to s and C_L .

For delay sensitivity to gate length variation, we refer to the variational table according to s and C_L , and generate delay d' . Then we calculate $\partial d / \partial x_i = \partial d / \partial L_g = (d - d') / \Delta L_g$.

For delay sensitivity to interconnect process variable x_i , we first calculate variation of C_L under a small change of x_i as $\Delta C_L = \Delta x_i \sum (u_{ik} C_j)$, where Δx_i is 5% of the nominal value of x_i . Then we refer to the variational table and calculate delay d' according to $C_L + \Delta C_L$. Therefore, we calculate $\partial d / \partial x_i = (d - d') / \Delta x_i$.

Effective capacitance delay model

For each buffer-to-buffer segment in the circuit, effective capacitance C_{eff} rather than lumped capacitance C_L is used to refer to the table. Thus we can consider the interconnect resistance shielding effect more accurately. There are several effective capacitance methods can be used, such as iterative method [19] and non-iterative method [20][21]. For the speed concern, we use non-iterative method here. The method proposed in [21] is used for RC interconnect and is difficult to be applied in buffer-to-buffer segment. Thus we apply the method proposed in [20], which evaluates effective capacitance by matching the delay of a cell with a Π load and the delay of a cell with a single effective capacitance load. The delay under s and C_{eff} is named as d .

For delay sensitivity to gate length variation, we first compute the effective capacitance under the variational gate length. Under the gate length change ΔL_g , effective capacitance C'_{eff} is recalculated using the method proposed in [20]. Note here the Π load does not change with ΔL_g . Then we use C'_{eff} and s to refer to the variational table, and generate delay d' . Then the delay sensitivity to gate length variation is calculated by $\partial d / \partial x_i = \partial d / \partial L_g = (d - d') / \Delta L_g$.

For delay sensitivity to interconnect process variables, we need variational effective capacitance to refer to the nominal table. Thus we have to compute the new effective capacitance C'_{eff} under process variation Δx_i . However, it costs too much to derive a new Π load and compute C'_{eff} accordingly. Instead we use $\Delta C_{eff} = C_{eff} \cdot \Delta C_L / C_L$ to approximate the change of C_{eff} due to Δx_i , where C_L is the lumped capacitance, and ΔC_L is variation of C_L and is calculated by $\Delta C_L = \Delta x_i \sum (u_{ik} C_j)$. Therefore $C'_{eff} = C_{eff} + \Delta C_{eff}$ is used to refer to the nominal table and generate delay d' , then the delay sensitivity to x_i is calculated by $\partial d / \partial x_i = (d - d') / \Delta x_i$.

3. Experiment Results

We apply our methods to ISCAS85 circuits using a UNIX server running on Solaris 2.7. The systematic process variation variables considered in our paper are variations of the transistor gate length, the width of 5 metal layers, the thickness of 5 metal layers and the thickness of 5 inter-layer-dielectrics (ILD). We apply the following manufacturing ranges of these variables: gate length $\pm 6\%$, metal width $\pm 5\%$, metal thickness $\pm 20\%$, and ILD thickness $\pm 40\%$. The range of delay variation is about $\pm 10\%$ of the nominal delay.

We first show the running time comparison between the traditional RSM and new method in Table 1. For each circuit we perform RSM and our new method respectively to generate the parasitic delay model for all buffer-to-buffer segments in the circuit. RSM is implemented by SPICE simulation with its running time listed in the third column. The path delay is computed by summing buffer-to-buffer delays. The running time of our method is listed in followed columns. Compared to RSM, our method achieves significant speedup. The running time of the method based on lumped C delay model is faster than the method based on effective capacitance delay model by 2-5 times. The reason is that the method based on effective capacitance method spends more cost on C_{eff} computation.

Table 1. Running time comparison between the traditional RSM and new methods for ISCAS85 circuits.

Circuit	# of buffer-to-buffer delays	Running time		
		RSM (hh:mm)	New Methods (s)	
			Lumped C	Effective C
c432	343	0:41	0.014	0.020
c499	440	1:03	0.017	0.026
c880	755	1:30	0.014	0.053
c1355	1096	2:13	0.044	0.084
c1908	1523	2:48	0.075	0.304
c2670	2292	4:19	0.108	0.456
c3540	2961	5:39	0.143	0.466
c5315	4509	>8 hr	0.196	0.785
c6288	4832	>9 hr	0.200	0.846
c7552	6253	>10 hr	0.308	1.600

To evaluate the accuracy of our method, we perform RSM and our method on the longest path of each circuit. Results are compared under the corner condition. In our experiments, the path delay under the nominal process condition d_0 is computed by SPICE simulation. Under the corner condition, the parametric variational delay computed by the traditional RSM is denoted as d' and the parametric variational delay calculated by our method is denoted as d'' using function (1). Then the delay error under the corner condition is computed by $(d'' - d') / (d_0 +$

d'). This value indicates the result of our method is how close to the result of RSM.

The results are shown in Table 2, where the number of cells in the longest path is listed in the second column, the path delay computed by RSM is listed in the third column and the delay variation under the worst case corner condition is listed in the fourth column. From the table, we can conclude that the method based on effective capacitance delay model is more accurate. Its delay error is less than 3% and for most circuits the error is around 1% of the path delay, where the delay error of the method based on lumped C model is less than 5%.

Table 2. Accuracy comparison between the traditional RSM and new methods for ISCAS85 circuits.

Circuit	# of cells in path	Worst case delay computed by RSM (ps)	Delay Var. (%)	Delay error under worst case corner (%)	
				Lumped C	Effective C
c432	17	698.5	10.80	-4.38	-0.01
c499	11	464.6	9.83	-2.09	-0.60
c880	24	530.3	9.54	-2.00	-0.36
c1355	24	609.1	10.37	-3.64	-2.98
c1908	40	724.5	11.02	-2.46	-1.55
c2670	32	947.6	10.83	-2.84	-0.65
c3540	47	1103.1	9.97	-0.33	-1.24
c5315	49	994.5	10.02	-2.50	-1.15
c6288	124	2853.4	9.53	-0.11	-1.71
c7552	41	690.9	10.28	-2.86	-1.32

4. Conclusions

In this paper, we present fast parametric delay evaluation under process variation by PARADE. Our method avoids multiple parasitic extractions and multiple delay evaluations as did in the traditional RSM, and result in significant speedup. The method based on effective capacitance delay model achieves higher accuracy. Experiments on ISCAS85 circuits show that our methods are effective and accurate for the parametric delay evaluation under process variation. And our new estimation method for capacitance sensitivity computation is applicable for any commercial parasitic extraction tools.

5. Acknowledgments

The authors thank Dr. Sani R. Nassif of IBM for important suggestions.

6. References

[1] Y. Liu, S. R. Nassif, L. T. Pileggi and A. J. Strojwas, "Impact of interconnect variations on the clock skew of a gigahertz microprocessor," *DAC* 2000, pp. 168–171.
 [2] V. Mehrotra, S. L. Sam, D. Boning, A. Chandrakasan, R. Vallishayee and S. Nassif, "A methodology for modeling the

effects of systematic within-die interconnect and device variation on circuit performance," *DAC* 2000, pp. 172–175.
 [3] E. Malavasi, S. Zanella, C. Min J. Uschersohn, M. Misheloff and C. Guardiani, "Impact analysis of process variability on clock skew," *ISQED* 2002, pp. 129–132.
 [4] R. B. Brawhear, N. Menezes, C. Oh, L. T. Pillage and M. R. Mercer, "Predicting circuit performance using circuit-level statistical timing analysis," *DATE* 1994, pp. 332–337.
 [5] H. Chang and S. S. Sapatnekar, "Statistical timing analysis considering spatial correlations using a single PERT-like traversal," *ICCAD* 2003, pp. 621–625.
 [6] A. Agarwal, D. Blaauw and V. Zolotov, "Statistical timing analysis for intra-die process variations with spatial correlations," *ICCAD* 2003, pp. 271–276.
 [7] M. Orshansky, L. Milor, P. Chen, K. Keutzer and C. Hu, "Impact of systematic spatial intra-chip gate length variability on performance of high-speed digital circuits," *ICCAD* 2000, pp. 62–67.
 [8] E. Acar, S. N. Nassif, L. Ying and L. T. Pileggi, "Assessment of true worst case circuit performance under interconnect parameter variations," *ISQED* 2001, pp. 431–436.
 [9] A. Gattiker, S. Nassif, R. Dinakar and C. Long, "Timing yield estimation from static timing analysis," *ISQED* 2001, pp. 437–442.
 [10] S. Borkar, T. Kamik, S. Narendra, J. Tschanz, A. Keshavarzi and V. De, "Parameter variations and impact on circuits and microarchitecture," *DAC* 2003, pp. 338–342.
 [11] G. M. Luong and D. M. H. Walker, "Test generation for global delay faults," *ITC* 1996, pp. 433–442.
 [12] 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," *DAC* 2002, pp. 566–569.
 [13] A. Krstic, L. C. Wang, K. T. Cheng and J. J. Liou, "Diagnosis of delay defects using statistical timing models," *VTS* 2003, pp. 339–344.
 [14] X. Lu, Z. Li, W. Qiu, D. M. H. Walker and W. Shi, "Longest path selection for delay test under process variation," *ASP-DAC* 2004.
 [15] A. D. Fabbro, B. Franzini, L. Croce and C. Guardiani, "An assigned probability technique to derive realistic worst-case timing models of digital standard cells," *DAC* 1995, pp. 702–706.
 [16] B. Stine, D. Boning and J. Chung, "Analysis and decomposition of spatial variation in integrated circuit process and devices," *IEEE Trans. on Semiconductor Manufacturing*, 10(1), 1997, pp. 24–41.
 [17] S. R. Nassif, "Modeling and analysis of manufacturing variations," *CICC* 2001, pp. 223–228.
 [18] D. Blaauw, V. Zolotov and S. Sundaeswaran, "Slope propagation in static timing analysis," *IEEE Trans. CAD*, 21(10), 2002, pp. 1180–1195.
 [19] J. Qian, S. Pullela and L. Pillage, "Modeling the "Effective capacitance" for the RC interconnect of CMOS gates," *IEEE Trans. CAD*, 13(12), 1994, pp. 1526–1535.
 [20] A. B. Kahng and S. Muddu, "Improved effective capacitance computations for use in logic and layout optimizations," *VLSI DESIGN* 1999, pp. 578–582.
 [21] C. V. Kashyap, C. J. Alpert and A. Devgan, "An "Effective" capacitance based delay metric for RC interconnect," *ICCAD* 2000, pp. 229–235.