

Accuracy Enhancement of Grid-Based SSTA by Coefficient Interpolation

Shinyu NINOMIYA[†], Nonmember and Masanori HASHIMOTO^{†a)}, Member

SUMMARY Statistical timing analysis for manufacturing variability requires modeling of spatially-correlated variation. Common grid-based modeling for spatially-correlated variability involves a trade-off between accuracy and computational cost, especially for PCA (principal component analysis). This paper proposes to spatially interpolate variation coefficients for improving accuracy instead of fining spatial grids. Experimental results show that the spatial interpolation realizes a continuous expression of spatial correlation, and reduces the maximum error of timing estimates that originates from sparse spatial grids. For attaining the same accuracy, the proposed interpolation reduced CPU time for PCA by 97.7% in a test case.
key words: statistical timing analysis, manufacturing variability

1. Introduction

For coping with aggravation of manufacturing variability, stochastic performance estimation before fabrication has been eagerly demanded, and statistical static timing analysis (SSTA) has been intensively studied [1]–[4]. Delay times are expressed in statistical distributions, and signal arrival times are computed statistically. With SSTA, a relation between performance and yield can be predicted before fabrication.

For implementing SSTA, variability models of gates and interconnects are necessary. Manufacturing variability is often decomposed into die-to-die, within-die spatially-correlated, and random components [5]. Among these components, the within-die spatially-correlated component is the least tractable, since a large number of random variables and their covariance matrix are necessary to take it into account, which severely limited the analyzable circuit size in the past [4]. Later on, reference [2] proposed SSTA that models spatially-correlated component using a 2-D grid. The authors translate correlated random variables into uncorrelated random variables by PCA (principal component analysis), and improve SSTA efficiency. Another approach is a modeling with quad-tree proposed in [6]. Both approaches have different advantages and disadvantages [7], however they share the same issue, that is the trade-off between accuracy and computational time required for modeling. When the number of spatial division increases, the spatially-correlated component is well reproduced, however it involves unwanted increase in computational time.

This paper presents a technique that mitigates dis-

cretization error by spatially interpolating coefficients of principal components in PCA-based SSTA [2]. The interpolation enables a continuous expression of correlation even when the grid-based modeling is used for spatially-correlated variability.

This paper is organized as follows. Section 2 discusses the conventional grid-based model of manufacturing variability, and points out its problem. Section 3 presents accuracy enhancement of the grid-based model using coefficient interpolation. Experiments in Sect. 4 demonstrate improvement of SSTA accuracy. The discussion is concluded in Sect. 5.

2. Problem of Grid-Based Variability Model

This section explains a grid-based spatially-correlated variability modeling used in SSTA, and demonstrates its problem.

2.1 Grid-Based Modeling of Spatially-Correlated Manufacturing Variability

Variation of a parameter that affects delay (F), e.g. gate length and threshold voltage, is often expressed as [5]

$$F = f_0 + X_g + X_s + X_r, \quad (1)$$

where f_0 is the average value of F , and X_g , X_s , and X_r are random variables whose average is 0. X_g represents die-to-die variability, and it fluctuates uniformly within a chip. All elements on a single chip have the same value in terms of die-to-die variability. In contrast, X_s and X_r represent within-chip variability, and elements within a chip have different values.

The within-chip variability consists of spatially-correlated variation X_s and random variation that differs element by element X_r (Fig. 1). X_s has stronger correlation between neighboring elements, and the correlation decreases as the distance increases, whereas X_r fluctuates randomly

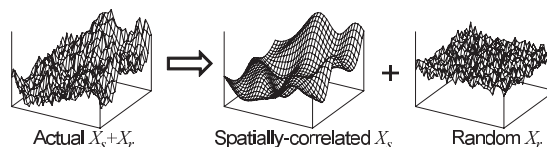


Fig. 1 A conceptual illustration of within-chip variability.

Manuscript received March 11, 2010.

Manuscript revised June 21, 2010.

[†]The authors are with the Department of Information Systems Engineering, Osaka University, Suita-shi, 565-0871 Japan.

a) E-mail: hasimoto@ist.osaka-u.ac.jp

DOI: 10.1587/transfun.E93.A.2441

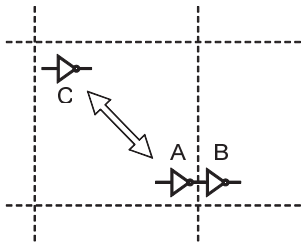


Fig. 2 An example that discretization causes a significant error.

independent of other elements. With X_s component, relative placement of elements affects correlation between the element delays.

To take the spatially-correlated variability into SSTA, a model that can reproduce the variability with a reasonable accuracy is necessary. Reference [2] proposes a SSTA that takes into consideration the spatially-correlated manufacturing variability using PCA. We here explain how the variability is modeled in [2]. We first divide a chip spatially. Spatially-correlated component X_s is discretized in a 2-D grid, and a random variable is assigned to each region. Within a region, the variability is assumed to be identical. After the variable assignment, a correlation coefficient matrix is constructed, and PCA is applied to the matrix. Random variable p_i in region i is expressed as a sum of orthogonalized variable p'_j .

$$p_i = \mu_i + \sigma_i \sum_{j=1}^m \sqrt{\lambda_j} v_{ij} p'_j, \quad (2)$$

where μ_i is the average of p_i , σ_i is the standard deviation of p_i , λ_j is the j -th largest eigenvalue of the correlation coefficient matrix, v_{ij} is the i -th value of the eigenvector corresponding to λ_j , and m is the number of the principal components.

Applying the above grid-based modeling to F in Eq. (1), F is expressed as a linear summation of uncorrelated random variables.

$$F = f_0 + \sum_{j=1}^n k_{ij} x_j + \delta, \quad (3)$$

where x_j is the uncorrelated random variable whose average and standard deviation are 0 and 1, respectively, and it includes p'_j in Eq. (2) and die-to-die variation component. k_{ij} is the coefficient of x_j in region i , and δ corresponds to random component X_r .

2.2 Error due to Discretization

Correlation coefficient between two elements continuously changes in space by nature, while the grid-based model incurs discretization error inevitably. Figure 2 illustrates two examples of the discretization error. When elements A and B are placed adjacently, they often have a strong correlation. However, there is a grid boundary between them, and hence

Table 1 CPU time required for PCA.

Grid	CPU time (sec)
20 × 20	0.33
30 × 30	4.31
40 × 40	24.2
50 × 50	92.86
60 × 60	306.09

different random variables are assigned for them. Consequently, the modeled correlation becomes weaker than the actual one. On the other hand, though elements A and C are placed distantly, they belong to the same region, and hence the correlation coefficient between them is modeled as one. In this case, the modeled correlation is stronger than the actual one.

These errors due to discretization are significant especially when the number of discretized regions is small, because the size of each region becomes larger, and the correlation between adjacent two regions becomes weaker. In fact, to model spatially-correlated variability accurately, finer spatial discretization is necessary, which will be shown in Sect. 4.

2.3 Computational Cost

The finer grid improves the modeling accuracy. The larger number of random variables, however, increases CPU times of PCA and SSTA.

The computational complexity of PCA used in modeling is $O(n^3)$, where the number of regions is n [2]. Table 1 lists CPU time for PCA that was performed using R [9] on a computer with Opteron 2.4 GHz processor and 16 GB memory. As the number of discretized regions increases, the required CPU time increases drastically. Memory usage is also a problem, since the memory space of $O(n^2)$ is necessary to store the correlation coefficient matrix. The CPU time of SSTA is proportional to the number of principal components [2].

This CPU time problem becomes severer especially when the chip area is large and the correlation distance is small. CPU time increases when more accurate analysis is necessary, and the accuracy degrades when CPU time is saved vice versa. Taking into account the trade-off between CPU time and accuracy, we have to model the spatially-correlated variability with a reasonable discretization.

3. Accuracy Enhancement by Interpolating Principal Component Coefficients

Grid-based modeling of spatially-correlated variability necessarily involves the discretization error as pointed out in Sect. 2. To solve this problem, we propose to use spatial interpolation for expressing continuous change of correlation and mitigating the discretization error. We interpolate principal component coefficient k_{ij} in Eq. (3) using two interpolation techniques that are often used for image processing; bilinear and bicubic interpolations [8].

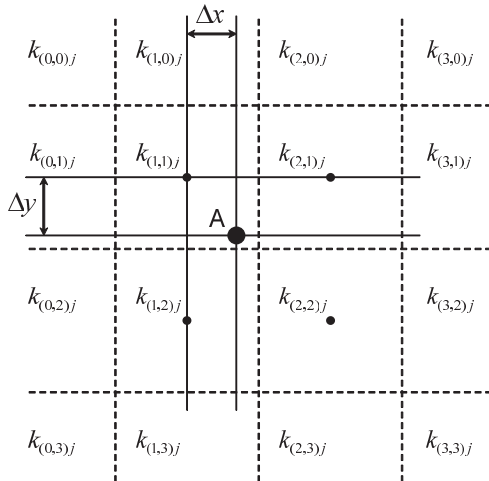


Fig. 3 Interpolation for point A.

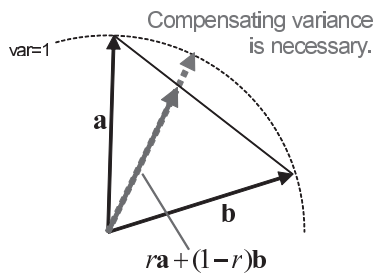


Fig. 4 Variance compensation.

We here explain the coefficient interpolation via bilinear interpolation as an example. Dotted lines in Fig. 3 represent grid boundaries, and here let us compute coefficient k_j corresponding to an element at point A. Bilinear interpolation uses values at neighboring four points.

$$k_j = (1 - \Delta x)(1 - \Delta y)k_{(i_x, i_y)j} + \Delta x(1 - \Delta y)k_{(i_x+1, i_y)j} \\ + (1 - \Delta x)\Delta y k_{(i_x, i_y+1)j} + \Delta x\Delta y k_{(i_x+1, i_y+1)j}. \quad (4)$$

Δx and Δy represent horizontal and vertical distances from the center of region $(i_x, i_y) (= (1, 1))$ in Fig. 3 where point A is included, respectively. The distance between adjacent regions is normalized to one, and then $0 \leq \Delta x, \Delta y \leq 0.5$.

When simply interpolating coefficients using Eq. (4), the variance after the interpolation is underestimated. Figure 4 illustrates the reason. Suppose \mathbf{a} and \mathbf{b} are coefficient vectors $(k_{i0}, k_{i1}, \dots, k_{in})$ and $(k_{(i+1)0}, k_{(i+1)1}, \dots, k_{(i+1)n})$, and we interpolate \mathbf{a} and \mathbf{b} for simplicity, though bilinear interpolation uses four vectors. The norm of the interpolated vector $\|\mathbf{r}\mathbf{a} + (1-r)\mathbf{b}\|$, where r is a weighting factor determined by distance, becomes smaller than the correct value, which leads to underestimation of variance.

We therefore compensate the underestimation by multiplying a constant to the coefficient interpolated by Eq. (4).

$$k'_j = k_j \cdot \frac{\sigma_{org}}{\sqrt{\sum_i^n k_i^2}}. \quad (5)$$

σ_{org} is the standard deviation before the interpolation, and

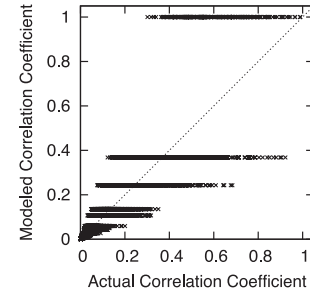


Fig. 5 Correlation coefficient in conventional grid-based model.

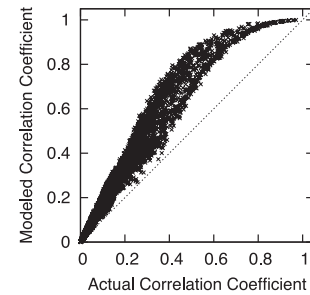


Fig. 6 Correlation coefficient with bilinear interpolation.

k'_j is the coefficient after compensating the variance.

In the case of using bicubic interpolation, values at neighboring 16 points are used for interpolation [8]. The expression of the interpolation is described in Appendix. The compensation of the variance (Eq. (5)) is similarly performed after the interpolation.

From now, we will demonstrate the error of correlation coefficients is reduced by the proposed coefficient interpolation. Referring to a variability in a 90 nm technology [10], we assumed that the correlation coefficient of spatially-correlated variability was dependent on distance, and expressed as e^{-2x} , where x mm is the distance between two elements, referring to [10]. We chose two points in a chip randomly, and compared the correlation coefficients estimated by conventional grid-based model and the proposed model to the correct value. The results are depicted in Figs. 5, 6 and 7. The chip size and the grid size were assumed to be $5 \times 5 \text{ mm}^2$ and 10×10 , respectively.

Figure 5 shows the correlation coefficients expressed by the conventional grid-based model. Due to the discretization, only a few discrete values are expressible. There are many dots where the estimated correlation coefficient is larger than the actual one in the upper part. On the other hand, Figs. 6 and 7 depict the correlation coefficients estimated with bilinear and bicubic interpolations. With the interpolation, the continuous expression of correlation coefficients is attained. Large errors of the original grid-based model found in Fig. 5 are improved both in Figs. 6 and 7. In the case of bilinear interpolation, the modeled correlation coefficients tend to be larger than the actual, because the function $f(x) = e^{-2x}$ is concave and the linear interpolation between any two points overestimates the correlation.

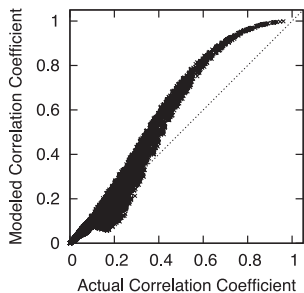


Fig. 7 Correlation coefficient with bicubic interpolation.

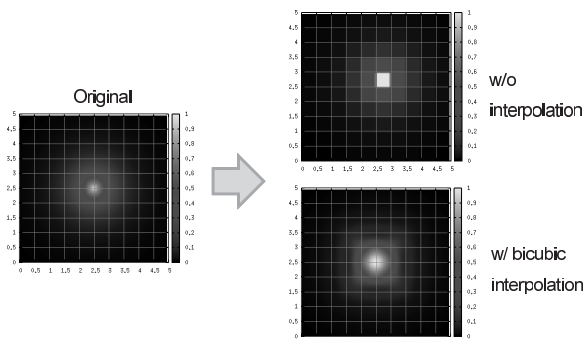


Fig. 8 Distribution of correlation coefficient to the center point.

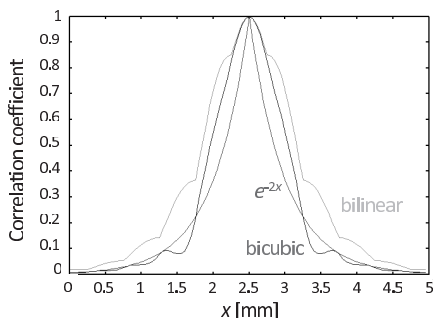


Fig. 9 Correlation coefficient to the center point. The distributions at $y=2.5$ are extracted from Fig. 8.

On the other hand, this tendency is suppressed in the case of bicubic interpolation. Figure 8 shows the distribution of correlation coefficients to the center point. The interpolation smooths the distribution. Figure 9 shows the distribution of correlation coefficient at $y=2.5$ to the center point extracted from Fig. 8. We can see that the bilinear interpolation overestimates the correlation coefficient in all x region, whereas the bicubic interpolation reduces the amount of overestimation in the range of correlation coefficient from 0.25 to 0.8 and underestimates the correlation coefficient below the range. Figure 9 also suggests that the appropriate interpolation could depend on the function of the spatial correlation.

Table 2 lists RMS (root mean square) error of the correlation coefficients, that is RMS value of (modeled correlation coefficient) – (actual correlation coefficient). The bicubic interpolation archived the lowest error whereas the RMS error was increased by bilinear interpolation. In the case

Table 2 RMS error of correlation coefficient.

Model	RMS error
grid	0.0893
bilinear	0.1031
bicubic	0.0573

that the spatial correlation is expressed by an exponential function, the bilinear interpolation is not appropriate.

4. Experimental Results

In this section, the proposed modeling with interpolation is applied to SSTA, and the SSTA accuracy is discussed.

4.1 Experimental Conditions

We implemented SSTA proposed in [2] with C++ language, and evaluated the accuracy. We assumed a $5 \times 5 \text{ mm}^2$ chip in a 90 nm technology. Supposing V_{th} had spatial correlation just as an example, V_{th} variation of $\sigma=25 \text{ mV}$ was given. Note that the proposed interpolation is independent of variation parameters of the spatial correlation. The correlation coefficient was assumed to be dependent on distance $x \text{ mm}$, and be expressed as e^{-2x} . Other variability components, such as random and die-to-die, were not considered here. We used a benchmark circuit c1355 included in IS-CAS85 benchmark set. The number of cells was 329. We obtained a cell placement using a commercial P&R tool, and scaled the placement to two sizes; $0.25 \times 0.25 \text{ mm}^2$ and $0.05 \times 0.05 \text{ mm}^2$. When a circuit is placed in a smaller area, more accurate model of correlation coefficient is necessary for estimating timing. The grid size was varied from 2×2 to 20×20 .

When we use the grid-based model, the analyzed result varies depending on the circuit placement, even though the assumed variability is uniform, as pointed out using Fig. 2. This is because the relative position between grid boundaries and cells fluctuates the timing estimates. We therefore placed the circuit at 8×8 positions within a single region, as depicted in Fig. 10, and evaluated the timing estimates at each position. Thus, the total number of timing estimation for each circuit is 64.

4.2 Without Interpolation

Figures 11 and 12 show the timing estimates when the conventional grid-based model was used. The left and right graphs represent the mean and standard deviation (SD) of the circuit delay with respect to the 64 timing estimates, respectively. The error bars indicate the maximum and minimum values. The horizontal axis is the number of discretization per side. We can see the estimates converge to a value as the number of discretized regions becomes large. However, when the circuit area is small, the difference between the maximum and minimum is still large in Fig. 12. In this case, the 20×20 grid is not sufficient. The timing analysis of a circuit in a small area is sensitive to the discretization

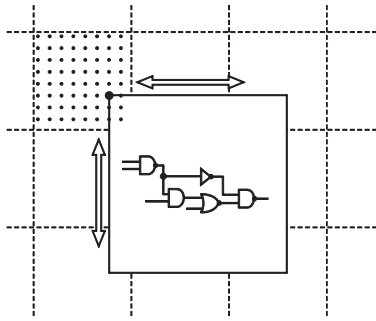


Fig. 10 Circuit placement for evaluation.

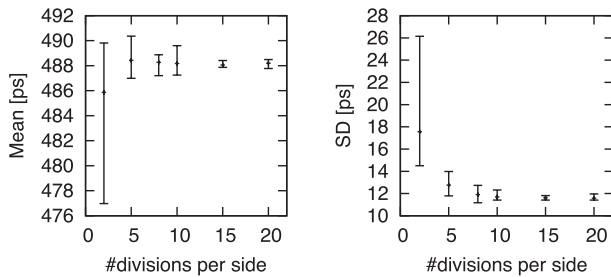


Fig. 11 Mean and standard deviation (SD) of circuit delay (w/o interpolation, placed area $0.25 \times 0.25 \text{ mm}^2$).

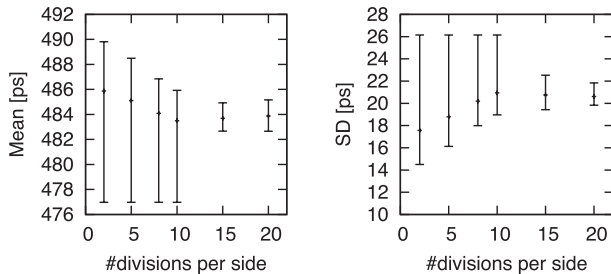


Fig. 12 Mean and standard deviation (SD) of circuit delay (w/o interpolation, placed area $0.05 \times 0.05 \text{ mm}^2$).

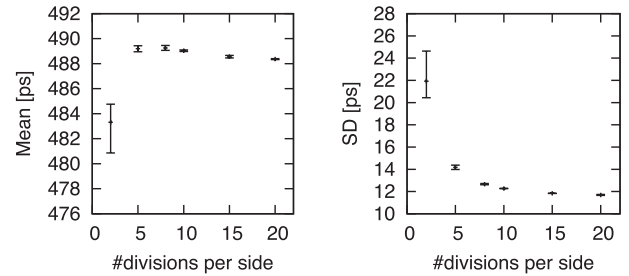


Fig. 13 Mean and standard deviation (SD) of circuit delay (w/bicubic interpolation, placed area $0.25 \times 0.25 \text{ mm}^2$).

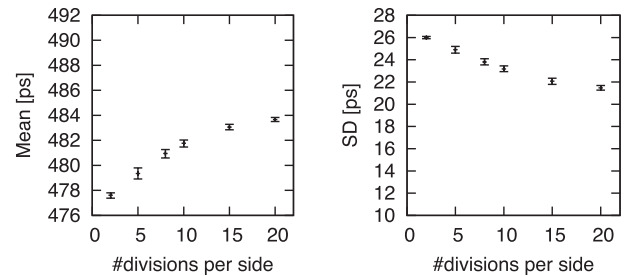


Fig. 14 Mean and standard deviation (SD) of circuit delay (w/bicubic interpolation, placed area $0.05 \times 0.05 \text{ mm}^2$).

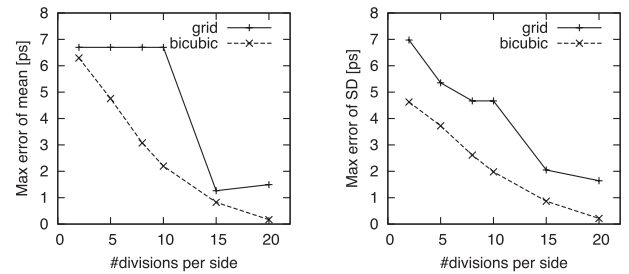


Fig. 15 Max. error vs. #discretization (placed area $0.05 \times 0.05 \text{ mm}^2$).

error.

Let us examine the above result from a viewpoint of chip designers. The timing analysis of a small block requires finer spatial discretization. In contrast, the above result suggests that in the case of timing verification for inter-block signals, sparser discretization is sufficient. Therefore, the required fineness of spatial discretization, that is the physical dimension of a grid, is determined by the error of the small blocks, and it is independent of the entire circuit size.

4.3 With Interpolation

Figures 13 and 14 show the estimates when the bicubic interpolation was applied. The difference between the maximum and minimum values becomes small, which means the estimation is not sensitive to the relative placement between cells and grid boundaries.

Figure 15 shows the maximum errors of the average and standard deviation in the case of $0.05 \times 0.05 \text{ mm}^2$

placement. The timing analysis with bicubic interpolation achieved more accurate estimation than that with the conventional grid-based model. Although the conventional model could reduce the error by increasing the number of discretized regions, the proposed model attained the same accuracy with the smaller number of discretized regions. For example, when the maximum acceptable error of the average and standard deviation is 3ps, the proposed model requires only 8×8 grid, whereas the conventional model needs 15×15 grid. In this case, the proposed model can reduce the CPU cost for PCA by 97.7%.

In the experimental setup in this paper in terms of chip size and function of correlation coefficient, the CPU time of PCA for 15×15 grid is affordable. On the other hand, the correlation distance of the spatial correlation becomes shorter and the chip size becomes larger, the number of grids increases and the CPU time of PCA explodes due to $O(n^3)$ complexity. The SSTA run time also increases because the SSTA run time is proportional to the number of principal components [2]. In this situation, the proposed interpolation

becomes more effective in the PCA run time. In addition, SSTA with the proposed interpolation has fewer principal components, though the runtime of SSTA increases due to the interpolation. Therefore, the overhead of SSTA run time is expected to be insignificant.

5. Conclusion

This paper discussed modeling for spatially-correlated variability, and presented an accuracy enhancement technique with spatial interpolation of principal component coefficients. We experimentally demonstrated that the interpolation enabled the continuous expression of correlation even though the grid-based modeling was adopted. We also verified the accuracy improvement of SSTA. Even when analyzing a circuit placed in a small area, the proposed modeling provided accurate timing estimates with reasonable grid fineness. From another aspect, the proposed model attained the same accuracy even when the number of discretization was reduced. In the test case, CPU time for PCA was reduced by 97.7%. Future work includes further investigation supposing more advanced technologies.

Acknowledgement

This work was supported by STARC and NEDO. The authors thank Mr. Takashi Enami and Mr. Ken-ichi Shinkai of Osaka University for technical discussions with them and their reviews.

References

- [1] D. Blaauw, K. Chopra, A. Srivastava, and L. Scheffer, "Statistical timing analysis: From basic principles to state-of-the-art," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol.27, no.4, pp.589–607, April 2008.
- [2] H. Chang and S. Sapatnekar, "Statistical timing analysis under spatial correlations," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol.24, no.9, pp.1467–1482, 2005.
- [3] C. Visweswariah, K. Ravindran, K. Kalafala, S.G. Walker, S. Narayan, D.K. Beece, J. Piaget, N. Venkateswaran, and J.G. Hemmett, "First-order incremental block-based statistical timing analysis," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol.25, no.10, pp.2170–2180, Oct. 2006.
- [4] S. Tsukiyama, M. Tanaka, and M. Fukui, "A statistical static timing analysis considering correlations between delays," *Proc. ASP-DAC*, pp.353–358, 2001.
- [5] J. Xiong, V. Zolotov, and L. He, "Robust extraction of spatial correlation," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol.26, no.4, pp.619–631, April 2007.
- [6] A. Agarwal, D. Blaauw, V. Zolotov, S. Sundareswaran, M. Zhao, K. Gala, and R. Panda, "Statistical delay computation considering spatial correlations," *Proc. ASP-DAC*, pp.271–276, 2003.
- [7] B. Cline, K. Chopra, D. Blaauw, and Y. Cao, "Analysis and modeling of CD variation for statistical static timing," *Proc. ICCAD*, pp.60–66, 2006.
- [8] R. Keys, "Cubic convolution interpolation for digital image processing," *IEEE Trans. Acoust. Speech Signal Process.*, vol.ASSP-29, no.6, pp.1153–1160, 1981.
- [9] <http://www.r-project.org/>
- [10] H. Masuda, S. Ohkawa, A. Kurokawa, and M. Aoki, "Challenge:

Variability characterization and modeling for 65- to 90-nm processes," *Proc. CICC*, pp.593–599, 2005.

Appendix

Coefficient k_j is obtained by bicubic interpolation as follows.

$$k_j = \begin{pmatrix} -\Delta y^3 + 2\Delta y^2 - \Delta y \\ \Delta y^3 - 2\Delta y^2 + 1 \\ -\Delta y^3 + \Delta y^2 + \Delta y \\ -\Delta y^3 - \Delta y^2 \end{pmatrix}^T \cdot \begin{pmatrix} k_{(i_x-1,i_y-1)j} & k_{(i_x,i_y-1)j} & k_{(i_x+1,i_y-1)j} & k_{(i_x+2,i_y-1)j} \\ k_{(i_x-1,i_y)j} & k_{(i_x,i_y)j} & k_{(i_x+1,i_y)j} & k_{(i_x+2,i_y)j} \\ k_{(i_x-1,i_y+1)j} & k_{(i_x,i_y+1)j} & k_{(i_x+1,i_y+1)j} & k_{(i_x+2,i_y+1)j} \\ k_{(i_x-1,i_y+2)j} & k_{(i_x,i_y+2)j} & k_{(i_x+1,i_y+2)j} & k_{(i_x+2,i_y+2)j} \end{pmatrix} \cdot \begin{pmatrix} -\Delta x^3 + 2\Delta x^2 - \Delta x \\ \Delta x^3 - 2\Delta x^2 + 1 \\ -\Delta x^3 + \Delta x^2 + \Delta x \\ -\Delta x^3 - \Delta x^2 \end{pmatrix} \quad (\text{A} \cdot 1)$$



Shinyu Ninomiya received the B.E. and M.E. degrees in information systems engineering from Osaka University in 2007 and 2009, respectively. He is currently with Renesas Electronics Corporation. His research interest includes variability modeling and statistical timing analysis.



Masanori Hashimoto received the B.E., M.E. and Ph.D. degrees in Communications and Computer Engineering from Kyoto University, Kyoto, Japan, in 1997, 1999, and 2001, respectively. Since 2004, he has been an Associate Professor in Department of Information Systems Engineering, Graduate School of Information Science and Technology, Osaka University. His research interest includes computer-aided-design for digital integrated circuits, and high-speed circuit design. Dr. Hashimoto served on the technical program committees for international conferences including DAC, ICCAD, ISPD, ASP-DAC, ICCD and ISQED. He is a member of IEEE and IPSJ.