

Power Distribution Network Optimization for Timing Improvement with Statistical Noise Model and Timing Analysis

Takashi ENAMI[†], Takashi SATO^{††}, *Nonmembers*, and Masanori HASHIMOTO^{†a)}, *Member*

SUMMARY We propose an optimization method for power distribution network that explicitly deals with timing. We have found and focused on the facts that decoupling capacitance (decap) does not necessarily improve gate delay depending on the switching timing within a cycle and that power wire expansion may locally degrade the voltage. To resolve the above facts, we devised an efficient sensitivity calculation of timing to decap size and power wire width for guiding optimization. The proposed method, which is based on statistical noise modeling and timing analysis, accelerates sensitivity calculation with an approximation and adjoint sensitivity analysis. Experimental results show that decap allocation based on the sensitivity analysis efficiently minimizes the worst-case circuit delay within a given decap budget. Compared to the maximum decap placement, the delay improvement due to decap increases by 3.13% even while the total amount of decaps is reduced to 40%. The wire sizing with the proposed method also efficiently reduces required wire resource necessary to attain the same circuit delay by 11.5%.

key words: power distribution network, decoupling capacitance, timing analysis, statistical static timing analysis, decap insertion, wire sizing

1. Introduction

Power supply noise has been a primary concern in modern high-performance circuit design due to increased current consumption and lowered supply voltage. Recently, power supply noise has been increasingly influential on timing. On-chip power supply noise mainly consists of IR drop and Ldi/dt noise. Expanding power wires is a common technique for reducing IR drop sacrificing of routability and wire resources. To efficiently exploit wire resource, power wire optimization has been intensively studied [2], [3]. However, it cannot reduce Ldi/dt noise originating from package and bonding wires, or rather wider wires have less damping effect and might intensify Ldi/dt noise.

In recent designs, decoupling capacitance (decap) has been intentionally placed in power distribution network (PDN) to suppress IR drop and Ldi/dt noise. Decap is often implemented with MOS gate capacitance, and a large decap consumes a large silicon area. Moreover, gate leakage current has increased along miniaturization of transistor, and hence it becomes more difficult to place a large amount of decap within a given leakage constraint. Therefore, necessary and sufficient decaps should be placed at appropriate

positions. In the past, efficient decap allocation has been studied [4]–[8]. In addition, co-optimization methods for simultaneously managing decaps and power wires were also proposed [9], [10].

However, most conventional design methods aim to reduce power supply noise, and not to directly minimize the impact on timing. Earlier studies [2]–[7], [9], [10] are classified into two major groups. The first one aims to minimize PDN resources under several constraints including electrical limitations, such as voltage drop and current density, and physical rules such as minimum wire width and decap size. The other group aims to reduce voltage drop within a limited amount of resources. The methods of the latter group first specify an allowable voltage drop, which is sometimes given by the integration of the excessive drop with respect to time and is minimized within the given resource budget. Essentially, both of these groups are mutually transformable, namely, their purposes are to make effective use of PDN resources. However, the suppression of voltage drop is not straightly correlated with timing improvement. In particular, decap insertion into on-chip PDN does not necessarily improve timing [11]. To efficiently use PDN resources and maximize timing improvement, we must consider how resources affect timing.

Pant and Blaauw [8] proposed a decap allocation method for improving timing within a decap budget. The authors computed sensitivities of an objective function that mainly consists of timing variation using adjoint sensitivity analysis, and found positions suitable for decap placement. However, the problem is that, at each position, the worst voltage drop within a specified clock cycle is computed and used for gate delay calculation without considering the switching timing window. Decap nicely reduces peak voltage drop, but does not improve supply voltage at all the timings within a whole clock cycle, which is illustrated in Sect. 2.1. The same authors [8] assume that reduction in peak voltage drop necessarily improves timing, but it is not *true*. Moreover, a noise waveform changes cycle by cycle, and the noise reduction effect of decap also varies. Spatial noise variation and within-cycle and inter-cycle temporal noise variation caused by decap insertion/removal must be considered in timing-oriented decap allocation.

This paper proposes a PDN optimization method for timing that explicitly takes spatial and temporal variation of noise due to decaps and power wire modification into account. To take noise dependence on input patterns into consideration, we statistically model dynamic power sup-

Manuscript received March 19, 2012.

Manuscript revised June 18, 2012.

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

^{††}The author is with the Department of Communications and Computer Engineering, Kyoto University, Kyoto-shi, 606-8501 Japan.

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

DOI: 10.1587/transfun.E95.A.2261

ply noise and its variation due to decap modification, and compute statistical sensitivity. For efficient sensitivity computation, we devised a performance function, which can be efficiently computed with an approximation, for adjoint sensitivity analysis [12]. The sensitivity analysis thus tightly couples timing variation and PDN parameters, which is the main contribution of this work. Guided with the computed sensitivity, we can identify PDN components that improve timing.

The rest of this paper is organized as follows. Section 2 illustrates how decaps and power wire expansion affect timing and demonstrates that decap does not necessarily improve timing. Section 3 gives an overview of the proposed method and Sect. 4 describes the details of the method. Section 5 demonstrates that the proposed method can improve timing via PDN modification, and finally Sect. 6 concludes the discussion.

2. Difficulties in Timing-Aware Design

Modifications in PDN affect the voltage waveform, which improves or degrades circuit performance. The problem is that it is difficult to predict how a PDN structure affects timing. We first examine the relation between decap and timing in Sect. 2.1. We then focus on power wire and timing in Sect. 2.2.

2.1 Decap Effect on Timing

Decoupling capacitance suppresses the peak voltage drop, while the average voltage within a clock cycle hardly improves. Decaps are discharged to supply charge to current-hungry nodes as temporal current sources. Afterwards, the decaps must be recharged. When the decap is large, a larger amount of charge must be restored, which means the charge time for recovery, which is tightly related to the RC time constant, becomes longer. This phenomenon is illustrated in Fig. 1. After adding a decap, the peak voltage drop significantly reduces. In contrast, the supply voltage in the latter half of the clock cycle is lower than that before the decap addition. This means that decap does not necessarily improve

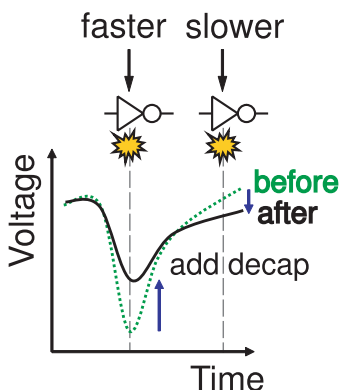


Fig. 1 Decap effect on timing.

supply voltage at all timings within a cycle. In other words, at some timings, the supply voltage improves, which results in timing improvement. On the other hand, at other timings, the supply voltage worsens and gate switching delay increases.

Suppose the left gate in Fig. 1 is in a critical path. The critical path delay in this example is improved by the addition of decap. In contrast, the circuit delay increases when the right gate of Fig. 1 belongs to a critical path. Furthermore, decreasing the decap may improve this critical path delay, which might not be widely known to designers. Without considering such decap attributes, timing cannot be efficiently improved via decap allocation.

The effect of decaps on timing improvement depends on the position of a critical path, and hence appropriate positions that maximize timing improvement should be selected. We show the impact of decap position on a worst-case delay as an example. We estimate the circuit delay using a noise aware statistical static timing analysis (SSTA) [13], which gives a circuit delay distribution considering noise dependence on input patterns. We define the worst-case delay as $\mu + 3\sigma$ for this example, other definitions of delay also can be used, where μ and σ are the average and the standard deviation of the circuit delay, respectively. In this example, a power/ground network (Fig. 2) was attached to a divider circuit [14], where both power and ground networks were separately modeled, though the figure is simplified. In this circuit, a 100 fF capacitance was initially placed between power and ground at each node. We then calculated the worst-case delay of circuit c6288 (one of ISCAS85 benchmark circuits) when the divider noise is given. We evaluated the variation of the worst-case delay when a 3 pF decap was added into each power/ground node. Figure 2 illustrates the worst-case delay variation. When a decap is placed at (0, 3) or (1, 3), the worst-case delay is minimized by more than 5 ps, whereas decaps placed at six positions increase the circuit delay. In this example, we should place decaps at (0, 3) and (1, 3), and must not place decaps at (2, 0) and (2, 1).

2.2 Effect of Wire Width on Timing

Expanding power supply wires without considering timing also may cause timing degradation or wasteful spending on wire resources. We explain such a case using Fig. 3. The upper-left node is connected to an ideal voltage source, and

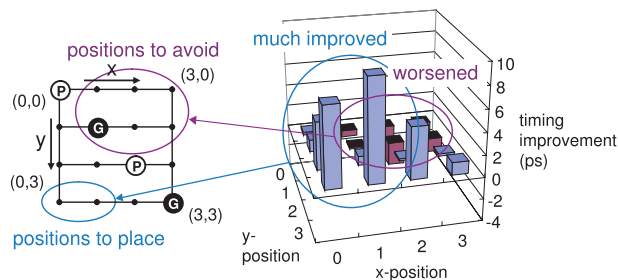


Fig. 2 Evaluating position dependence of decap on timing.

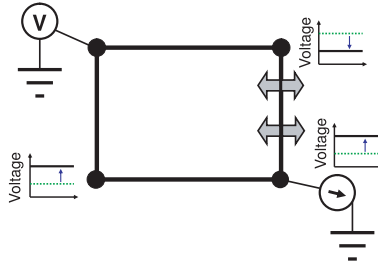


Fig. 3 Voltage variation caused by wire expansion.

the current is drawn to the bottom-right node through two routes. Now, let us expand the right-side wire and decrease the resistance. In this case, the voltages at the bottom-left and bottom-right nodes increase as we expect. Decrease in current flowing through the left wire improves the voltage at the bottom-left node. Also, decrease in impedance between the power supply and current source mitigates the voltage drop at the bottom-right node. However, the voltage at the upper-right node falls because the current through the upper-right node increases. That is to say, expanding the wire can improve the voltage at most nodes, but can also worsen the voltage at the upstream node. On the other hand, from the timing point of view, it is important to improve the voltage on the critical path.

To identify suitable positions where decap amount or wire width should be increased, we need to compute sensitivities of timing for each candidate of decap placement positions and wire segments. In Sect. 4, we discuss how to efficiently compute the sensitivities and how to determine PDN parameters.

3. Problem Formulation and Overview of Proposed Method

We formulate a PDN design problem such that PDN design is modified from an initial one to minimize the worst-case delay using a given amount of PDN resources. In this paper, PDN parameters for optimization are limited to decap amount and wire width, and the decap amount at each placeable position and the wire width of each segment are modified. The maximum decap size and wire width are specified in advance for each placeable position and wire segment respectively. This problem is transformed into the following optimization problem.

- **Objective:**
 - To minimize circuit delay.
- **Constraints:**
 - Total resources of PDN.
 - Maximum decap amount at each position.
 - Maximum wire width of each segment.
- **Optimization parameters:**
 - Decap amount at each position.
 - Wire width of each segment.

The proposed method first calculates the sensitivity of circuit delay to each PDN parameter at each position. When a PDN parameter is varied, power supply noise changes, which results in timing variation. We therefore express sensitivity as the product of two sensitivities; sensitivity of power supply noise to the PDN parameter and sensitivity of timing to power supply noise. Once one parameter is changed, its effect spreads spatially in the circuit, and the voltages at many points vary. We thus define the sensitivity of circuit delay to i -th parameter P_i as follows.

$$\frac{\partial Delay}{\partial P_i} = \sum_{V_j} \left(\frac{\partial Delay}{\partial V_j} \times \frac{\partial V_j}{\partial P_i} \right), \quad (1)$$

where $Delay$ is the circuit delay, V_j is the voltage at j -th point, and P_i corresponds to decoupling capacitance C_i or wire conductance G_i .

4. Sensitivity Calculation Using Adjoint Sensitivity Analysis

First, Sect. 4.1 explains the statistical noise modeling used in sensitivity calculation and timing analysis. Section 4.2 presents efficient computation of $\frac{\partial Delay}{\partial P_i}$. Then, Sect. 4.3 describes decap allocation and wire expansion based on the sensitivities.

4.1 Statistical Modeling of Power Supply Noise

We first briefly introduce a statistical modeling of power supply noise that we proposed [13].

Noise waveforms differ cycle by cycle depending on input patterns. Power supply noise varies continuously in space and time, and strictly speaking, every cell has different noise waveform. When analyzing power supply noise, observation points are limited because of computational cost. Furthermore, we spatially and temporally discretize power supply noise, and assign random variables. Finally, we compute statistical properties, such as average, standard deviation and correlation of the assigned random variables.

We first determine the representative voltages by spatially discretizing a chip. This spatial discretization is performed by partitioning a chip/block area into a 2-D grid. Each divided partition may have several observation points, and in this case, the representative value of a partition must be chosen. As a representative value, for example, the voltage at the center point (Fig. 4) or the average voltage in each partition is a candidate. The voltages of all nodes in the same partition are assumed to be identical.

An important property of power supply noise is its dynamic behavior. To express dynamic waveforms within a cycle, we partition a clock cycle into several time spans and compute a representative voltage (e.g. average shown in Fig. 4).

We then assign a random variable to the power supply or ground voltage at each time span and at each spatial partition. We call this assigned random variable a power

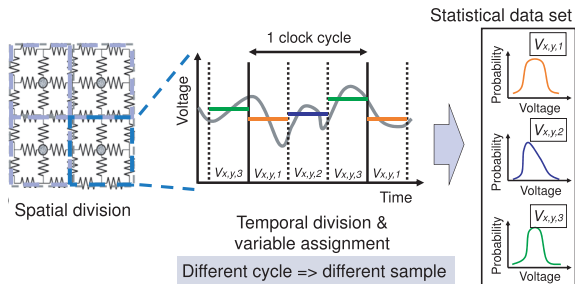


Fig. 4 Statistical modeling of dynamic power supply noise [13].

variable. We treat the voltage value at every clock cycle as a different sample. Figure 4 shows an example when the voltage at partition (x, y) is divided into three time spans. Its random variables are denoted as $V_{x,y,1}$, $V_{x,y,2}$, and $V_{x,y,3}$. The number of time spans is determined according to the modeling requirement, i.e. when we need to accurately model dynamic variation within a clock cycle, the number of spans should be increased; otherwise, several spans are sufficient.

To efficiently perform statistical timing analysis, our previous work [13] orthogonalized the variables with principal component analysis, and derived a statistical model including statistical information such as averages, standard deviations, and correlation coefficients of the variables.

4.2 Sensitivity Calculation of Circuit Delay to Decap

In this section, we propose an efficient procedure to calculate Eq. (1).

First, let us examine a straightforward sensitivity calculation. The sensitivity of circuit delay to a power variable $\frac{\partial \text{Delay}}{\partial V_j}$ can be computed using SSTA considering power supply noise [13]. The sensitivity of a power variable to a PDN parameter $\frac{\partial V_j}{\partial P_i}$ can be calculated with a circuit simulation. However, both require a large number of SSTA runs and circuit simulations. SSTA must be repeated for the number of power variables to obtain all sensitivities. Circuit simulation must be performed for the number of decaps or wire segments. Therefore, the required cost for a straightforward sensitivity computation is prohibitively expensive.

We therefore compute $\frac{\partial \text{Delay}}{\partial V_j}$ with an approximation and devise a performance function directly corresponding to the circuit delay for adjoint sensitivity analysis [12] to efficiently calculate $\frac{\partial \text{Delay}}{\partial P_i}$. The following subsections explain how to compute $\frac{\partial \text{Delay}}{\partial V_j}$ and $\frac{\partial \text{Delay}}{\partial P_i}$ in detail.

4.2.1 Sensitivity Calculation of $\frac{\partial \text{Delay}}{\partial V_j}$

We first calculate the sensitivity of circuit delay to power variable $\frac{\partial \text{Delay}}{\partial V_j}$. Every gate delay is varied by voltage fluctuation, but the delay variation of each gate does not necessarily affect the circuit delay because the gate with a large timing slack has little effect on the circuit delay. We therefore have to consider how much each gate delay variation contributes to circuit delay variation.

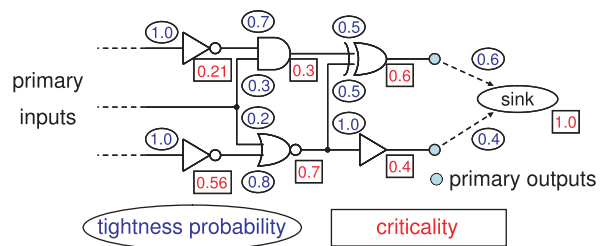


Fig. 5 Criticality computation.

To do this, we calculate “criticality”, defined as the probability that a gate belongs to the critical path of a circuit. Several methods for computing criticality have been proposed [15], [16]. We adopt the method proposed by Visweswariah et al. [15] for the sake of implementation simplicity, though other methods also can be used. Criticality is computed in two stages. The first stage is tightness probability calculation at every gate, where the tightness probability is that in which an input is selected in MAX operation for the latest arrival time computation at the gate, and is computed for every input. This calculation is already performed inside SSTA, and hence no additional computational cost is necessary. The second stage is the criticality computation. We illustrate the computational flow in Fig. 5. First, the criticality of the (virtual) sink node is set to 1, because the sink node is always included in the critical path. Then, the criticality of each gate is calculated with a backward traversal of the timing graph.

$$Cr_i = \sum_{j \in \text{outputs}(i)} Cr_j \times Tp_{j,i}, \quad (2)$$

where Cr_i is the criticality of gate i , Cr_j is the criticality of fan-out gate j , and $Tp_{j,i}$ is the tightness probability for selecting the arrival time of gate i at gate j .

We then approximate the sensitivity of a circuit delay to a power variable $\frac{\partial \text{Delay}}{\partial V_j}$ using criticality. Focusing on a single gate, the contribution of the gate to circuit delay variation depends on not only criticality but also on the sensitivity magnitude of the gate delay to power supply voltage. We thus express the contribution of a gate as the product of these two factors. We next focus on power variables. A power variable has a strong impact on circuit delay when many gates that have large contributions on circuit delay are associated with the power variable. Considering the fact that voltage variation both at a receiver and at drivers affects gate delay, the sensitivity of circuit delay to power variable $V_{x,y,t}$ is described as

$$\begin{aligned} \frac{\partial \text{Delay}}{\partial V_{x,y,t}} = & \sum_{i \in (x,y,t)} \left[Cr_i \sum_{j \in \text{inputs}(i)} \left(\left(\frac{\partial d_i}{\partial V_{re}} \right)_j \cdot Tp_{i,j} \right) \right. \\ & \left. + \sum_{k \in \text{outputs}(i)} \left\{ \sum_l \left(Cr_k \cdot \left(\frac{\partial d_k}{\partial V_{dr}} \right)_l \cdot Tp_{k,l} \right) \right\} \right], \\ & [\text{where } l \in (\text{inputs}(k) \in (x', y', t))], \quad (3) \end{aligned}$$

where $\left(\frac{\partial d_i}{\partial V_{re}} \right)_j$ is the sensitivity of gate delay d_i from in-

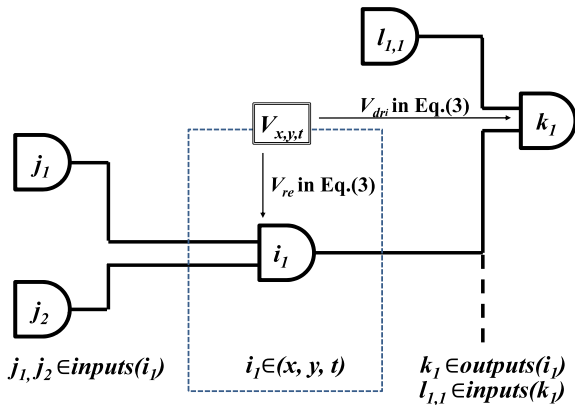


Fig. 6 An example of gate connections associated with Eq. (3).

put j to power variable V_{re} . V_{dr} and V_{re} are the voltages on the driver and receiver sides respectively, which means Eq. (3) deals with the voltage difference between driver and receiver. The former term represents the contribution of gate i as a receiver, and the latter term is the contribution as drivers. Gate i also contributes the latter term when another gate l at position (x', y') drives gate k at time span t . Gate indices of Eq. (3) are exemplified in Fig. 6. Tightness probability is used to exclude the contribution of non-critical gates. For example, if the path from gate j_2 to gate i_1 has no possibility to be included in a critical path, Tp_{i_1, j_2} becomes 0 and the effect of this path is discarded.

We thus calculate $\frac{\partial Delay}{\partial P_i}$ in Eq. (1) using Eq. (3). This approximation is experimentally validated in Sect. 5.2. With this approximation, we do not have to repeat SSTA for the number of power variables.

4.2.2 Sensitivity Calculation of $\frac{\partial Delay}{\partial P_i}$ Using Adjoint Sensitivity Analysis

We next discuss how to compute $\frac{\partial Delay}{\partial P_i}$ using $\frac{\partial Delay}{\partial V_j}$. A straightforward procedure to obtain $\frac{\partial Delay}{\partial P_i}$ in Eq. (1) is to compute $\frac{\partial V_j}{\partial P_i}$ by slightly changing P_i and simulating power supply noise in each case. The problem is that we have to perform the simulation for all decaps and wire segments, which is impractical in terms of computational cost. To solve this problem, we use adjoint sensitivity analysis. By appropriately setting a performance function required in adjoint sensitivity analysis, we can directly obtain $\frac{\partial Delay}{\partial P_i}$ instead of $\frac{\partial V_j}{\partial P_i}$. We assume that a power distribution network is modeled as a linear circuit and switching gates are expressed as current sources.

The adjoint sensitivity analysis we use simulates two circuits; the original circuit and its adjoint circuit. The adjoint circuit is constructed as follows. First, all independent voltage and current sources are removed from the original circuit. Then, current sources, which are given based on a performance function of time τ , $f(\tau)$, are inserted into voltage observation points. We then explain how to calculate

$f(\tau)$. The final value we want to obtain is $\frac{\partial Delay}{\partial P_i}$ in Eq. (1), and hence $\int_0^T f(\tau) d\tau$ should be a circuit delay function w.r.t. power variables. This means that the delay expression using first-order expansion (4) should correspond to $\int_0^T f(\tau) d\tau$.

$$Delay = Delay_{init} + \sum_t \sum_x \sum_y \frac{\partial Delay}{\partial V_{x,y,t}} (V_{x,y,t} - V_{x,y,t}^{init}), \quad (4)$$

$$= \sum_t f(t) \left(= \int_0^T f(\tau) d\tau \right), \quad (5)$$

where T is the clock period. A clock cycle is divided into several time spans as mentioned in Sect. 4.1, and hence the integral is expressed as a summation. Thus, the performance function $f(\tau)$ is expressed with power supply (ground) voltage of node n ($V_n(\tau)$) as

$$f(\tau) = D(\tau) + \left[\sum_x \sum_y \left\{ W_{x,y}(\tau) \sum_{n \in (x,y)} V_n(\tau) \right\} \right]. \quad (6)$$

Here, $D(\tau)$ is the term independent of $V_n(\tau)$, and $W_{x,y}(\tau)$ is $\frac{\partial Delay}{\partial V_j}$. Node n is one of the observation points inside partition (x, y) . If there are three temporal divisions used in the statistical noise model, $W_{x,y}(\tau)$ is also discretized and described as

$$W_{x,y}(\tau) = \begin{cases} \frac{\partial Delay}{\partial V_{x,y,1}} & (0 \leq \tau < bt_{1,2}) \\ \frac{\partial Delay}{\partial V_{x,y,2}} & (bt_{1,2} \leq \tau < bt_{2,3}) \\ \frac{\partial Delay}{\partial V_{x,y,3}} & (bt_{2,3} \leq \tau < T), \end{cases} \quad (7)$$

where $bt_{i,j}$ is the boundary time between span i and span j . Strictly speaking, in some cases, $W_{x,y}(\tau)$ must be divided by an integer depending on the number of observation points in a spatial partition and the number of time steps in a temporal span.

The value of the current source attached to node n , Φ_n , is represented as

$$\begin{aligned} \Phi_n(\zeta (= T - \tau)) &= -\frac{\partial f(\tau)}{\partial V_n}, \\ &= -W_{x,y}(\tau). \end{aligned} \quad (8)$$

Now that we can obtain the simulation result of the adjoint circuit with current sources of Eq. (8), we next calculate the sensitivity of circuit delay to PDN parameter P_i ($\frac{\partial Delay}{\partial P_i}$) by convolution. According to the theory of adjoint sensitivity analysis, in case that PDN parameter P_i is decap C_i , the sensitivity to decap C_i is calculated as follows.

$$\frac{\partial Delay}{\partial C_i} = - \int_0^T \{ \psi_{C_i}(T - \tau) v_{C_i}'(\tau) \} d\tau, \quad (9)$$

where ψ_{C_i} is the voltage difference of C_i in the adjoint circuit, and v_{C_i}' is the derivative of the voltage difference of C_i in the original circuit. Similarly, we can obtain the sensitivity to wire conductance G_i .

$$\frac{\partial Delay}{\partial G_i} = - \int_0^T \{\psi_{G_i}(T - \tau) v_{G_i}(\tau)\} d\tau. \quad (10)$$

Note that the voltage difference of G_i in the original circuit is not differentiated.

Recalling that PDN parameter P_i includes both C_i and G_i , $\frac{\partial Delay}{\partial P_i}$ corresponding to capacitance and conductance can be calculated with Eqs. (9) and (10), respectively. We thus obtain the sensitivities of circuit delay to power variables. It should be noted that though $\frac{\partial Delay}{\partial v_j}$ includes an approximation, the other computation is exact thanks to adjoint sensitivity analysis. With a single adjoint network analysis, we can obtain the sensitivities to all decaps and wire segments.

4.3 Decap Allocation and Wire Expansion

We explain how to determine decaps or wire segments for improving timing based on sensitivities. We sort the PDN parameters based on sensitivities and assign decaps or expand wires in the order of sensitivities. Co-optimizing decap allocation and wire structure is possible by appropriately setting a priority function for decap and wires as stated in previous research [9], [10]. However, for simplicity, we only discuss individual optimization. Thus, resource constraints are separately given, namely, as total decap amount and total wire area.

Noise variation due to PDN modification varies cycle by cycle because current consumption fluctuates according to input patterns. It is difficult to determine the optimum PDN just by looking at a particular clock cycle. We calculate the sensitivities for a certain amount of cycles and obtain the sensitivity distribution. We already have noise waveforms for a certain amount of cycles used in the statistical noise modeling; therefore it is easy to obtain the sensitivity distribution. The sensitivity of C_i and G_i in m th clock cycle are respectively calculated by

$$\frac{\partial Delay}{\partial C_i} = - \int_0^T \{\psi_{C_i}(T - \tau) v_{C_i}((m-1)T + \tau)\} d\tau, \quad (11)$$

$$\frac{\partial Delay}{\partial G_i} = - \int_0^T \{\psi_{G_i}(T - \tau) v_{G_i}((m-1)T + \tau)\} d\tau. \quad (12)$$

We have several choices in how to sort the sensitivities since each sensitivity has a distribution. We have tried some choices and empirically found that the ordering based on the average sensitivity gave good optimization results. Thus, we sort decaps or wires using the average sensitivity.

After sorting, we assign the size of each decap or the width of each power wire. We explain how to allocate decaps as a representative of modification. Power wire expansion is performed in a similar manner. When the timing varies nonlinearly w.r.t. the decap size, we have to carefully

choose the decap size. On the other hand, when the change in decap size hardly varies the sensitivities, it is reasonable to assign the maximum amount of decaps with a good sensitivity. We have experimentally verified that the sensitivities did not change significantly even after decap allocation, which is shown in Sect. 5.2. We therefore set all the decap sizes to zero and assign the maximum decap amount from the top in the sensitivity order until the decap budget is fully used. The algorithm is very simple, but it efficiently works, which is shown in Sect. 5.3.

5. Experimental Results

This section discusses the experimental results. We first explain the experimental conditions in Sect. 5.1 and validate the sensitivity analysis in Sect. 5.2. We then show that decap allocation with the proposed method can improve timing in Sect. 5.3. Finally in Sect. 5.4, we reveal that the proposed method can also operate in efficient wire sizing.

5.1 Experimental Conditions

For simplicity, delay times of ISCAS85 benchmark circuits, a 64-bit multiplier, and an ALU circuit for vector operation were analyzed under the power supply noise of an FPU circuit [14]. The noise generator and the timing-analyzed circuits were also different for simplicity, and there was no technical difficulty into analyzing the timing of the noise generator circuit. Both circuits were synthesized using a commercial logic synthesizer and placed and routed using a commercial tool with a 90-nm standard cell library. We attached a power/ground network, shown in Fig. 7, to the noise generator circuit and simulated the power supply noise with input vectors of 2000 clock cycles. We prepared resistive and inductive PDNs and obtained two types of voltage waveforms, as shown in Fig. 8. In the statistical modeling of power supply noise, the numbers of spatial and temporal divisions were set to 10×10 and 10, respectively.

5.2 Validation of Sensitivities

We first validate the sensitivity calculation discussed in Sect. 4. $\frac{\partial Delay}{\partial v_j}$ computation introduces an approximation using criticality and tightness probability. The other point of the proposed sensitivity computation is exact thanks to adjoint sensitivity analysis. Therefore, we verify the estimation accuracy of $\frac{\partial Delay}{\partial v_j}$. For comparison, we also calcu-

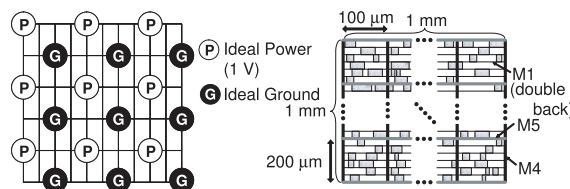


Fig. 7 Power/ground network of test circuit.

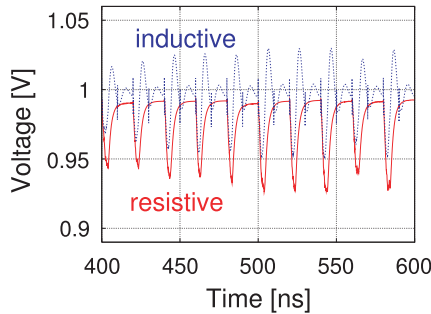


Fig. 8 Voltage waveforms at center point (VDD side).

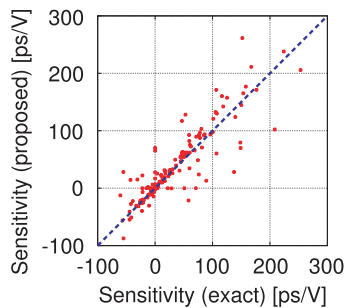


Fig. 9 Accuracy verification of $\frac{\partial \text{Delay}}{\partial V_j}$ computation of (c1908).

lated the reduction of the worst-case delay when the average of one power variable is improved by 1 mV. When the power variable corresponds to power/ground, the average increases/decreases by 1 mV. The sensitivities estimated in this exact procedure and with the proposed method are plotted in Fig. 9, and each dot corresponds to a power variable. We can see that the sensitivity is well estimated with Eq. (3), though the computational cost of the proposed method is negligibly small. The exact method must perform SSTA for the number of power variables, whereas the proposed method requires only one simple backward trace of the timing graph for criticality computation.

We then evaluated how different the sensitivities are before and after decap allocation as an example. The proposed method changes many decaps or power wires simultaneously based on the sensitivities evaluated simultaneously. Generally speaking, when a large variation occurs in a PDN, the sensitivities could significantly change. If the sensitivities are totally different before and after PDN optimization, we have to incrementally update the sensitivities.

We suppose two circuits; an initial circuit and a circuit optimized using the proposed method. Both circuits have the same amount of decaps in total, which corresponds to 25% of the total circuit area. In the optimization, we selected the top 50% of decap positions based on sensitivities, and increased the amount to the maximum, and at other positions, the decaps were removed. In the initial circuit, decaps were uniformly distributed in space. Figure 10 shows the sensitivities before and after decap allocation. The sensitivities are almost unchanged, which means it is hardly necessary to incrementally update the sensitivity. Figure 10

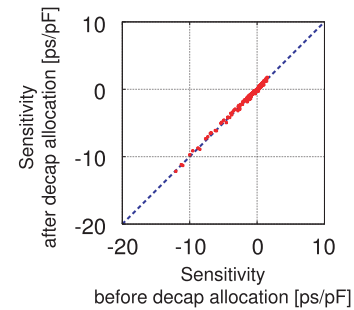


Fig. 10 Sensitivity comparison before and after decap allocation (c432).

also tells us that even though the decap size is maximized or becomes zero, the sensitivities are still acceptably accurate, and hence an intermediate size between zero and the maximum does not have to be selected.

5.3 Timing Optimization with Decap Allocation

We now demonstrate the timing optimization results of the proposed method. In Sect. 5.3.1, we evaluated the proposed decap allocation under various conditions of power supply noise. We then discuss where the decaps were inserted using the proposed method in Sect. 5.3.2. Finally, we compare the proposed method to a conventional method [8] in Sect. 5.3.3.

Before showing the experimental results, let us summarize the experimental setup. The decap per area was calculated assuming it was designed with $1\ \mu\text{m}$ gate length in the 90-nm CMOS process. Decaps could be placed in up to 50% of the circuit area at every point, for simplicity. The number of points was set to 3960. We calculated the sensitivities to decaps when 25% circuit area was occupied with decaps, i.e. 50% of the maximum amount of decaps were inserted.

5.3.1 Delay Improvement/Degradation with Various Decap Budgets

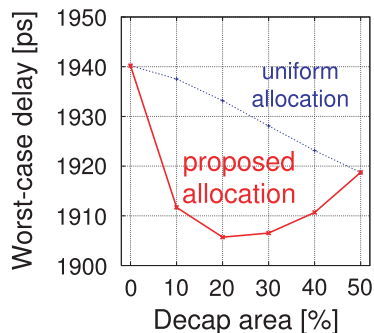
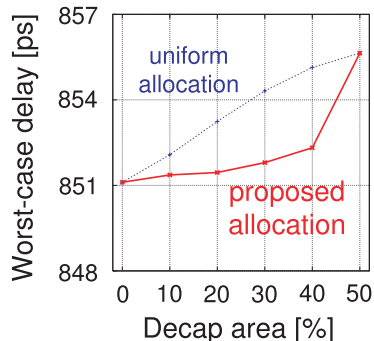
(1) Resistive Noise

First, we evaluated the relation between decap amount and circuit delay with decaps located using the proposed method. We assumed the resistive power supply noise for this evaluation. To reveal the effectiveness of the proposed method, we compared the worst-case timings of two circuits. In the first circuit, decaps were uniformly placed in the circuit, and the second circuit was optimized using the proposed method. In both circuits, the total amount of decaps was the same.

Figures 11 and 12 show the worst-case delays of the multiplier and the c7552 circuit with various decap budgets. In the horizontal axis, 50% of decap rate means that 50% of the circuit area was occupied by decaps. Note that in 0% and 50% cases, the amounts of decap at all the decap positions are zero or the maximum, which means the decap insertion results of the proposed allocation and the uniform allocation become identical. Therefore, the worst-case delay becomes

Table 1 Decap allocation results compared to maximum decap allocation (resistive noise).

circuit	# cells	Nominal delay (ps)	Delay w/o decap (ps)	Max decap allocation		Proposed decap allocation		
				delay(ps)	imp. rate(%)	decap rate(%)	delay(ps)	imp. rate(%)
c432	232	716.1	907.7	896.8	5.70	30	894.1	7.12
c1355	329	399.7	498.5	494.7	3.89	40	494.4	4.15
c1908	674	897.2	1008	1019	-9.83	20	1007	0.986
c6288	3382	2371	2953	2939	2.47	20	2929	4.27
c7552	2500	718.3	851.1	855.6	-3.41	0	851.1	0
multiplier	41629	1590	1940	1919	6.14	20	1906	9.86
ALU	17050	910	1078	1077	0.700	10	1076	1.22
average	-	-	-	-	0.809	20	-	3.94

**Fig. 11** Decap allocation results with various decap budget (multiplier, resistive noise).**Fig. 12** Decap allocation results with various decap budget (c7552, resistive noise).

the same in 0% and 50% cases.

In Fig. 11, increasing the amount of decaps did not always improve the delay. Moreover, Fig. 12 shows that increasing the amount of decaps deteriorated the delay of the c7552 circuit. In both in Figs. 11 and 12, the decaps allocated using the proposed method worked better than the uniformly placed decaps. In these experiments, the same power supply noise was given to every circuit. However, the effect of decaps on timing varied, which means the optimal decap allocation is strongly dependent on circuits and explicit consideration of timing is necessary in decap design.

Table 1 lists the worst-case delays of the delay-optimized circuits, those without decaps and those with the maximum amount of decaps. The third column represents the delay with ideal power supply voltage. The fourth column represents the delays when no decap was allocated.

The fifth and sixth columns are the results with the maximum amount of (50%) decaps. The sixth column lists the timing improvement rate over the delay increase due to power supply noise, which is defined as

$$\text{imp. rate} = \frac{\text{delay}_{w/o} - \text{delay}_{w/}}{\text{delay}_{w/o} - \text{delay}_{Nom}}. \quad (13)$$

where delay_{Nom} is the nominal delay without noise, $\text{delay}_{w/o}$ is the delay without decaps, and $\text{delay}_{w/}$ is the delay with decaps. The seventh to ninth columns are the results of the proposed method. The seventh column is the optimum amount of decaps to minimize the delay.

Compared with the maximum allocation, the optimum allocation of decaps increased the improvement rate by 3.13% (from 0.809% to 3.94%) on average. In the multiplier circuit, the improvement rate reached 9.86%. At the same time, the amount of decaps reduced by 60.0% on average. These results clearly point out that inserting decap as much as possible, which is of a common practice though, is not an appropriate design strategy. On the other hand, with the proposed method, we could effectively place decaps at timing-sensitive positions without increasing silicon area and gate leakage much. Besides, it should be noted that the delay improvement from delay without decap is not so large and the absolute and relative improvements are 24 ps and 1.5% at most. As explained in Sect. 2.1, the decoupling capacitance does not improve supply voltage at all timings within a cycle, and hence the average supply voltage does not change drastically. Therefore, the timing improvement via decap sizing is limited. By combining decap and wire sizing, decap effect on timing could be enhanced, which is a possible direction of our future work.

We finally show an example of the CPU time needed for decap allocation. The CPU times for adjoint network analysis with a fast linear circuit simulator [17], the convolution in Eq. (9), and SSTA including criticality computation in Eq. (2) were 15.0 s, 2.16 s and 2.15 s (e.g. circuit c6288), respectively. The CPU times of the other computations were negligibly small.

(2) Inductive noise

We also investigated the case where power supply noise is inductive. We compared the worst-case delays of two circuits as with the above evaluations. The worst-case delays of the c1908 circuit are shown in Fig. 13. Again, the alloca-

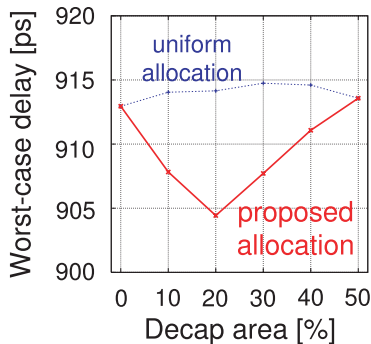


Fig. 13 Decap allocation results with various decap budgets (c1908, inductive noise).

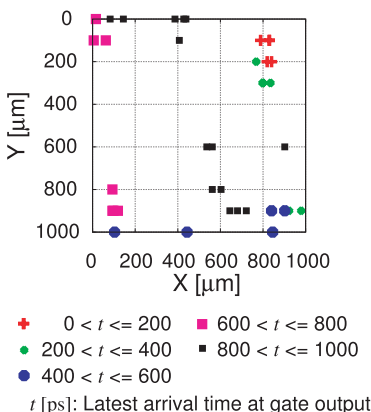


Fig. 14 Distribution of critical gates (c1908, resistive noise).

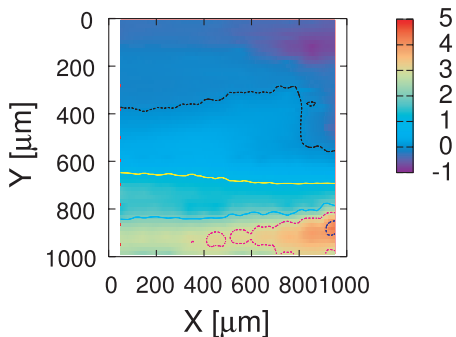


Fig. 15 Distribution of sensitivities (c1908, resistive noise).

tion with the proposed method achieved better results than uniform allocation. The proposed method gave an efficient allocation for timing even when inductive noise was dominant.

5.3.2 Discussion on Suitable Positions for Decap Insertion

We now discuss where the decaps were inserted in relation to the *critical* gates. A critical gate means that in which criticality is more than 0.3. Figure 14 shows where the critical gates exist, and positions of the gates were classified using five symbols according to the latest arrival time at each gate output. In contrast, Fig. 15 depicts the distribution of sen-

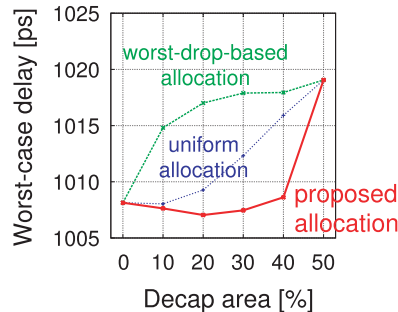


Fig. 16 Decap allocation results compared to worst drop allocation (c1908, resistive noise).

sivities. Negative sensitivity means a desirable position to insert decap, and decaps were preferentially placed at such positions. Judging from these figures, decaps, which were inserted near the critical gates whose latest arrival times were early (especially $0 < t \leq 200$ [ps]), improved the circuit delay. On the other hand, the positions near the critical gates, whose latest arrival times were in the middle of the clock cycle ($400 < t \leq 600$ [ps]), worsened circuit delay, suggesting that insertion of decaps should be avoided.

5.3.3 Comparison to Worst-Drop-Based Allocation

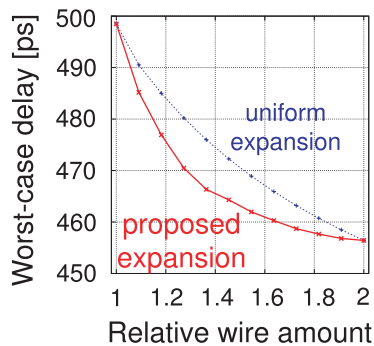
In the next experiment, we evaluated the conventional method [8], whose delay calculation is based on only the worst voltage drop. Therefore, an allocation result of the conventional method depends on the amount of change in the worst drop. As we mentioned in Sect. 2.1, decaps that suppress the worst voltage drop do not necessarily decrease the circuit delay. That is to say, decap allocation proposed by Pant and Blaauw [8] may worsen timing. We implemented the worst-drop-based allocation method maintaining the essentials of Pant and Blaauw’s [8] as described below. In this experiment, the worst voltage drop is defined as the minimum voltage at VDD side or the maximum voltage at the ground side in the first clock cycle. The worst voltage and its moment are obtained for every spatially divided partition, and adjoint sensitivity analysis is performed with stimulus current sources instead of Eq. (8). A stimulus current source is impulse current which is determined by the worst drop moment and Eq. (3). In this case, Eq. (3) is calculated using STA with the worst voltage. Figure 16 is an example that the worst-drop-based allocation increased the worst-case delay. When decaps are inserted, we must consider the effect of temporal voltage fluctuation.

5.4 Timing Optimization via Grid Wire Sizing

We finally evaluated the grid wire sizing. This method was performed based on the sensitivities calculated in Eq. (12). For simplicity, we only expanded the vertical (M4) grid wires in Fig. 7. The proposed method doubled the width of the grid wire in ascending order of sensitivity. The sensitivity of the grid wire was the sum of the sensitivities of the

Table 2 Wire sizing results (resistive noise).

circuit	Worst-case delay (ps)			Relative wire amount to achieve avg. delay		(uniform - proposed) uniform (%)
	all original	all doubled	avg. delay	uniform	proposed	
c432	907.6	836.2	871.9	1.33	1.26	5.73
c1355	498.5	456.4	477.5	1.33	1.18	11.7
c1908	1011	962.1	986.3	1.33	1.13	15.0
c6288	2953	2749	2851	1.35	1.22	9.77
c7552	851.0	799.1	825.0	1.37	1.14	17.1
multiplier	1931	1783	1857	1.35	1.27	5.62
ALU	1078	1011	1044	1.36	1.14	15.8
average	-	-	-	-	-	11.5

**Fig. 17** Wire sizing results with various wire budgets (c1355, resistive noise).

wire segments – wire segments were defined in the circuit simulation. The total number of vertical grid wires was 22 in the power/ground network. We also evaluated the circuit whose grid wires were uniformly expanded for comparison.

The sizing with the proposed method results of the c1355 circuit are illustrated in Fig. 17. The proposed sizing decreased the worst-case delay more than the uniform expansion. In cases that the relative wire amounts are 1.0 and 2.0, all the wire widths are the lower limit or the upper limit, and hence the worst-case delays of the proposed and uniform expansions are the same.

Table 2 compares the proposed expansion to the uniform expansion on the condition that the worst-case delays were the same. The second and third columns represent the worst-case delay when all wires were not expanded and fully expanded, respectively, and the fourth column represents the average of their delays. The fifth and sixth columns are the relative wire amounts necessary to achieve the delays in the fourth column. These values were calculated by linear interpolation of the nearest two points. The expansion with the proposed method saved 11.5% of wire resource on average, which clarified that the proposed method achieved efficient use of wire resources.

6. Conclusion

We proposed a timing-aware PDN modification method based on statistical noise modeling. Taking into account that a large decap could temporally lower supply voltage, and that an expanded power wire could locally lower the voltage,

the proposed method explicitly computes the timing sensitivity to decap size and wire width considering dynamic noise behavior and noise dependence on input patterns. The proposed method allocates decaps and sizes power wires based on the sensitivities efficiently computed thanks to an approximation and adjoint sensitivity analysis. The experimental results show that the maximum decap allocation does not necessarily minimize the circuit delay. The proposed method gives better allocation compared to maximum and uniform allocations. For resistive FPU noise, the proposed method decreased the delay by 3.13% on average with only 40.0% of the decap amount. As for power wire expansion, the proposed method also achieved better results than the uniform expansion. To attain the same circuit delay, the proposed method required around 11.5% less wire resources.

Acknowledgement

This work is supported in part by the Semiconductor Technology Academic Research Center (STARC), by the New Energy and Industrial Technology Development Organization (NEDO) and by the VLSI Design and Education Center (VDEC), the University of Tokyo with the collaboration with Synopsys Corporation.

References

- [1] T. Enami, M. Hashimoto, and T. Sato, "Decoupling capacitance allocation for timing with statistical noise model and timing analysis," Proc. ICCAD, pp.420–425, Nov. 2008.
- [2] T. Mitsuhashi and E.S. Kuh, "Power and ground network topology optimization for cell based VLSIs," Proc. DAC, pp.524–529, June 1992.
- [3] X.-D.S. Tan and C.-J.R. Shi, "Fast power/ground network optimization based on equivalent circuit modeling," Proc. DAC, pp.550–554, June 2001.
- [4] H. Li, Z. Qi, S.X.-D. Tan, L. Wu, Y. Cai, and X. Hong, "Partitioning-based approach to fast on-chip decap budgeting and minimization," Proc. DAC, pp.170–175, June 2005.
- [5] H. Su, S.S. Sapatnekar, and S.R. Nassif, "Optimal decoupling capacitor sizing and placement for standard-cell layout designs," IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst., vol.22, no.4, pp.428–436, April 2003.
- [6] P.-Y. Chen, C.-Y. Liu, and T.T. Hwang, "Transition-aware decoupling-capacitor allocation in power noise reduction," Proc. ICCAD, pp.426–429, Nov. 2008.
- [7] Y. Shi, J. Xiong, C. Liu, and L. He, "Efficient decoupling capacitance budgeting considering operation and process variations,"

- IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst., vol.27, no.7, pp.1253–1263, July 2008.
- [8] S. Pant and D. Blaauw, “Timing-aware decoupling capacitance allocation in power distribution networks,” Proc. ASP-DAC, pp.757–762, Jan. 2007.
- [9] K. Wang and M.M.-Sadowska, “On-chip power supply network optimization using multigrid-based technique,” Proc. DAC, pp.113–118, June 2003.
- [10] M. Zhao, R. Panda, B. Reschke, Y. Fu, T. Mewett, S. Chandrasekaran, S. Sundareswaran, and S. Yan, “On-chip decoupling capacitance and P/G wire co-optimization for dynamic noise,” Proc. DAC, pp.162–167, June 2007.
- [11] T.R.-Arabi, G. Taylor, M. Ma, and C. Webb, “Design & validation of the Pentium III and Pentium 4 processors power delivery,” Proc. VLSI Circuits, pp.220–223, June 2002.
- [12] T.L. Pillage, R.A. Rohrer, and C. Visweswariah, *Electronic Circuit & System Simulation Methods*, MC Graw-Hill, 1995.
- [13] T. Enami, S. Ninomiya, and M. Hashimoto, “Statistical timing analysis considering spatially and temporally correlated dynamic power supply noise,” IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst., vol.28, no.4, pp.541–553, April 2009.
- [14] OPENCORES.ORG, <http://www.opencores.org/>
- [15] C. Visweswariah, K. Ravindran, K. Kalafala, S.G. Walker, and S. Narayan, “First-order incremental block-based statistical timing analysis,” Proc. DAC, pp.331–336, June 2004.
- [16] J. Xiong, V. Zolotov, N. Venkateswaran, and C. Visweswariah, “Criticality computation in parameterized statistical timing,” Proc. DAC, pp.63–68, July 2006.
- [17] C. Mizuta, J. Iwai, K. Machida, T. Kage, and H. Masuda, “Large-scale linear circuit simulation with an inversed inductance matrix,” Proc. ASP-DAC, pp.511–516, Jan. 2004.



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, ASP-DAC, DATE, ICCD and ISQED. He is a member of IEEE and IPSJ.



Takashi Enami received the B.E., M.E., and Ph.D. degrees in Information Systems Engineering from Osaka University, Osaka, Japan, in 2006, 2008, and 2011 respectively. Since 2011, he has been with Fujitsu Laboratories Ltd. His research interest includes noise aware timing verification and low power design. He is a member of IEEE.



Takashi Sato received B.E. and M.E. degrees from Waseda University, Tokyo, Japan, and a Ph.D. degree from Kyoto University, Kyoto, Japan. He was with Hitachi, Ltd., Tokyo, Japan, from 1991 to 2003, with Renesas Technology Corp., Tokyo, Japan, from 2003 to 2006, and with the Tokyo Institute of Technology, Yokohama, Japan. In 2009, he joined the Graduate School of Informatics, Kyoto University, Kyoto, Japan, where he is currently a professor. He was a visiting industrial fellow at the

University of California, Berkeley, from 1998 to 1999. His research interests include CAD for nanometer-scale LSI design, fabrication-aware design methodology, and performance optimization for variation tolerance. Dr. Sato is a member of the IEEE.