PAPER *Special Section on VLSI Design and CAD Algorithms*

Successive Pad Assignment for Minimizing Supply Voltage Drop

Takashi SATO[†∗a)], **Masanori HASHIMOTO**[††], *and* **Hidetoshi ONODERA**[†], *Members*

SUMMARY   An efficient pad assignment methodology to minimize voltage drop on a power distribution network is proposed. A combination of successive pad assignment (SPA) with incremental matrix inversion (IMI) determines both location and number of power supply pads to satisfy drop voltage constraint. The SPA creates an equivalent resistance matrix which preserves both pad candidates and power consumption points as external ports so that topological modification due to connection or disconnection between voltage sources and candidate pads is consistently represented. By reusing sub-matrices of the equivalent matrix, the SPA greedily searches the next pad location that minimizes the worst drop voltage. Each time a candidate pad is added, the IMI reduces computational complexity significantly. Experimental results including a 400 pad problem show that the proposed procedures efficiently enumerate pad order in a practical time.
key words: successive pad assignment (SPA), incremental matrix inversion (IMI), voltage drop, power distribution network

1. Introduction

As predicted by Rents rule, the number of I/O pins of a LSI package has been increasing rapidly as more devices are packed into a chip [1]. Typical pin count for recent LSI's has reached several hundreds or over a thousand and is expected to increase even more. However, the number of pins allotted to the power supply and ground are usually limited. Recent technology advances have allowed us to include several millions of transistors on a chip, which significantly increases power consumption. Therefore, inefficient assignment of the power supply pad may cause a severe voltage drop. Optimizing the number and location of power supply pads is becoming critically important. One of the difficulties for pad optimization is the size of the power distribution network (PDN) to analyze. Even in the early planning stage, the PDN becomes quite large. Nevertheless, a methodology that realizes quick what-if analysis is required. Another difficulty is the topological change in the PDN. Designers try to optimize pad number and location by connecting and/or disconnecting power supplies to pads choosing among more than hundred pads. In order to explore a vast combination

of the pads, a simulation model must be reused to exploit analysis efficiency.

Much work on voltage drop analysis have been published since it has already become a real-life concern for industrial chip designs [2], [3]. The use of various original and efficient numerical techniques is proposed to accelerate computational time [4]–[6]. These methods are suitably used once a PDN is fixed for analysis. However, when the designer needs to optimize number and location of power supply pads, network topology will be modified trial by trial. Therefore, full procedures such as setting up a new circuit matrix, symbolic factorization, numerical factorization, and forward-backward substitution, are required for every trial. Those processes incur significant overhead when there are many pad combinations. Pad layout optimizations for signal and power supply integrity, such as the simultaneous switching output noise and crosstalk noise are discussed in [7] but their objectives do not include voltage drop minimization. Other works related to this subject present power network planning from a reliability standpoint [8] and the minimum number pad assignment with planar layout realization problem [9], but unfortunately there is no mention about the voltage drop. A Min-Forest heuristic is proposed in [10] to achieve uniform ground bounce on a power/ground tree of predetermined number of pads. The authors in [11] propose to solve a pad optimization problem utilizing a linear programming framework with divide and conquer approach and candidate pruning. However, the computational complexity required to solve a dense equivalent admittance matrix may still limit solvable problem size when the PDN is too tightly coupled to partition adequately.

In this paper, we first propose to build an equivalent resistance matrix to adopt for slight topology modification, then we utilize an incremental matrix inverse approach to enhance computational complexity. The contributions of this paper are 1) an efficient voltage drop calculation applicable for a set of circuits with different number or location of power supply pads, and 2) an efficient calculation to greedily optimize pad number and location.

2. Problem Formulation

In the following sections, static voltage drop is discussed—a PDN is modeled using resistors only. The following are the assumptions for supply pad assignment problem.

- There is a set S_p of pads P_i of which we assign ideal

power supplies represented by ideal voltage sources. The size of S_p is n_p, i.e. $\{S_p \ni P_i | i = 1 \ldots n_p\}$.

- There exists a set S_q of current sink points Q_i which represents power consumption of each circuit block. The size of S_q is n_q, i.e. $\{S_q \ni Q_i | i = 1 \ldots n_q\}$. At all or some of the points in S_q, independent current sources $j_i (i = 1 \ldots n_q)$ are connected to represent power consumption around the points Q_i.

- Voltage drop of a chip is observed at all points in S_q. Absolute supply voltage measured at Q_i is defined as V_{Q_i}. Drop voltage ΔV_{Q_i} equals $V_{dd} - V_{Q_i}$.

Figure 1 depicts a chip image including pads, circuit blocks, and current sink points. Inside the circuit blocks there are the current sink points Q_i that represent power consumption and are used as drop voltage observation points. Larger block requires more number of representative points in general especially when power consumption in a block is distributed unevenly. The PDN is modeled by resistors although they are not illustrated in Fig. 1 for clarity. This example shows the peripheral I/O pad configuration, although the proposed algorithm is applicable to other types of I/O's, such as area pads, without any modification. The algorithm can also be applied to pin assignment problems for a circuit block, but in this paper we concentrate only on the pad assignment problem.

Pads P_i with hatching are candidates for making connections to external power supplies through bonding wires or bumps. Remaining pads drawn without hatching are for other purposes, such as for signals. A larger number of pads can be included in the set S_p than the maximum number allowed for power supplies since increasing the possible combinations enlarge the solution space and possibly yields a better solution than starting from a limited choice. If the algorithm reveals that the number of power supply pads could be reduced, unused pads can be assigned to other signals. Or, the use of a less expensive package, which usually has a smaller outline and a smaller number of pins, may become possible. Therefore, choosing a subset of S_p that gives the minimum drop voltage at the observation points is our objective.
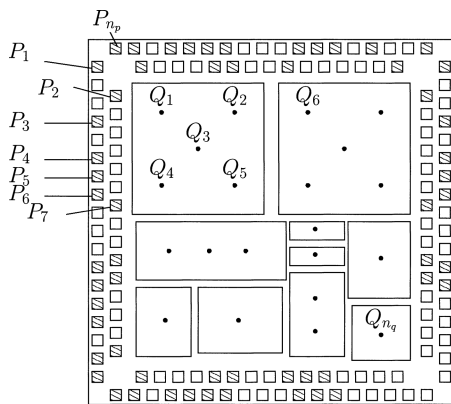
Based on the above presupposition, we define the following pad assignment problems.

Problem 1 Given a PDN, power consumption distribution, and allowable number of power supply pads n_a ($n_a \leq n_p$), determine the location of pads that minimize the worst voltage drop V_{worst}, defined as $V_{worst} = \max_i(\Delta V_{Q_i})$.

Problem 2 Given a PDN, power consumption distribution, power supply pad candidate set S_p, and voltage drop target V_t, determine the minimum number of pads and their locations so that $V_{worst} < V_t$ is satisfied.

3. Circuit Model for the PDN

The largest difference between ordinary drop voltage calculation and the pad assignment problem is whether circuit topology, specifically the source point set, is finalized or not. In the pad assignment problem, all pads that are initially included in S_p may not be necessarily connected to the power supplies—pads that are not connected to power supplies must be treated as open. Figure 2 is an abstracted chip model for the pad assignment problem. On the left of the PDN there are pads and on the right are current sink connections. As seen in the figure, the nodes in sets S_p and S_q are defined as external ports. Once voltage source connections to the pads are determined, voltage distribution is calculated by solving a linear equation constructed using the modified nodal approach for v_q:

$$\begin{pmatrix} Y & E \\ E^T & 0 \end{pmatrix} \begin{pmatrix} v_q \\ i_p \end{pmatrix} = \begin{pmatrix} J_q \\ v_p \end{pmatrix}. \tag{1}$$

The use of simpler nodal analysis formulation which replaces ideal voltage supply sources to Thevenin's equivalent current sources is also possible. Here, Y is a conductance matrix, E is an incident matrix for voltage sources, v_q is a nodal voltage vector for observation points, v_p is a pad voltage vector, i_p is a voltage source current vector, and J_q is a current source vector. Equation (1) is solved using direct methods, iterative methods [3], or various techniques such as in [4]–[6].

Direct decomposition methods are efficient for repeatedly solving a set of voltage drop problems. As long as the circuit topology is fixed, recalculation of the same PDN for
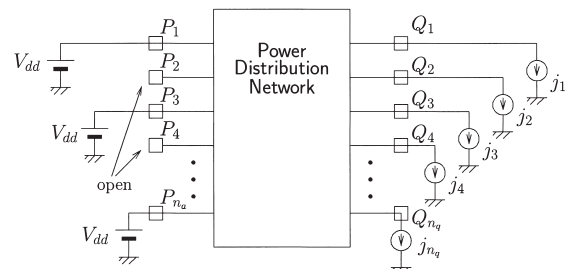

Fig. 1 Overview of candidate pad (P_i) and power consumption point (Q_i) distribution for a chip with peripheral pads. Q_i also serve as observation points for voltage drop.

Fig. 2 Abstracted chip model for Fig. 1.

a different current vector $\boldsymbol{J}_q$ is efficient since the decomposed matrix, whose calculation consumes most of the total calculation time, can be reused. However, to solve for the problems defined in the previous section, Eq. (1) has to be repeatedly constructed and solved for slightly different connections of the supply sources. In this case, every time the circuit topology changes, the most time-consuming process of matrix decomposition is required.

4. Efficient Voltage Drop Calculation for Exploring Pad Assignment Combinations

In this section, we propose the SPA algorithm that efficiently assigns pads to minimize voltage drop. Distinguishing characteristic of the SPA is to create an equivalent resistance matrix preserving all possible voltage supply pads and current source points as terminals, and to solve it using the IMI.

4.1 Step 1: Determination of the First Source Point

To calculate observation node voltages in Fig. 2, at least one node voltage has to be defined. Otherwise, the conductance matrix in Eq. (1) becomes singular. Step 1 determines the first supply point that will be used as the voltage reference in the following steps. We call this pad as the reference pad. If any supply pads are already fixed as supply pads, this step can be skipped since one of the fixed supply pads can be used as the reference pad.

The following is the procedure to determine the reference pad.

1. Connect all pads P_i in set S_p to ground.
2. Connect all circuit currents j_i to sinks Q_i in set S_q.
3. Calculate currents at all pads $i_{P_1}, i_{P_2}, \ldots i_{P_{n_p}}$.
4. Find a pad P_k ($0 \le k \le n_p$) in which the largest current flows $i_k = \max_\ell(i_{P_\ell})$ and use P_k as the reference P_{ref}.

From its derivation, P_{ref} has the lowest combined equivalent resistance for all current points Q_i. Thus the pad location for this one pad case is obtained.

4.2 Step 2: Equivalent Resistance Matrix Calculation

Once P_{ref} is determined, we then derive the equivalent resistance matrix R_{eq}. Each entry r_{ij} in R_{eq} represents combined resistance between ports i and j, which is derived by a column through the following procedure.

1. Connect reference pad P_{ref} to ground.
2. For each pad P_ℓ in set $S_r = S_p \cup S_q$ except for the reference pad $P_\ell \neq P_{ref}$ ($\ell = 1 \ldots (n_p+n_q-1)$), connect a 1 ampere current source between P_ℓ and P_{ref}. Leave all other ports open including the ones in S_q.
3. Measure all port voltages as a column vector $\boldsymbol{v}_{p_\ell}$. Here, $\boldsymbol{v}_{p_\ell}$ is arranged as

$$\boldsymbol{v}_{p_\ell} = (V(Q_1), \ldots, V(Q_{n_q}),$$
$$V(P_1), \ldots, V(P_{n_p}))^T. \qquad (2)$$

Then set the voltage vector as the ℓ-th column in R_{eq}.

$$R_{eq} = (\boldsymbol{v}_{p_1}, \ldots, \boldsymbol{v}_{p_{n_q}},$$
$$\boldsymbol{v}_{p_{(n_q+1)}}, \ldots, \boldsymbol{v}_{p_{(n_q+n_p-1)}}) \qquad (3)$$

R_{eq} can be efficiently obtained using direct methods. As mentioned earlier, once decomposition of the circuit matrix in Eq. (1) is found, it can be reused for different current connections by one forward and backward substitution.

4.3 Step 3: Calculation of Voltage Drop with Different Pad Combinations

Using R_{eq}, the voltage drop at the observation points are obtained as follows.

1. Partition equivalent resistance R_{eq} by observation points and number of candidate pads.

$$\begin{pmatrix} \boldsymbol{v}_q \\ \boldsymbol{v}_P \end{pmatrix} = R_{eq} \begin{pmatrix} \boldsymbol{J}_q \\ \boldsymbol{i}_P \end{pmatrix} = \begin{pmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \end{pmatrix} \begin{pmatrix} \boldsymbol{J}_q \\ \boldsymbol{i}_P \end{pmatrix} \qquad (4)$$

2. Because currents flowing into the pads that are not connected to the power supply are all zero, the corresponding rows and columns in R_{eq} can be eliminated. Let $\boldsymbol{v}_{P_s}, \boldsymbol{i}_{P_s}$ be voltage and current vectors of size k for voltage source connecting pads only.

$$\begin{pmatrix} \boldsymbol{v}_q \\ \boldsymbol{v}_{P_s} \end{pmatrix} = \begin{pmatrix} R_{11} & H_{12} \\ H_{21} & H_{22} \end{pmatrix} \begin{pmatrix} \boldsymbol{J}_q \\ \boldsymbol{i}_{P_s} \end{pmatrix} \qquad (5)$$

where H_{12}, H_{21}, and H_{22} are matrices made up of the elements of R_{12}, R_{21}, and R_{22} which are related to the pads connected to the power supplies. H_{12} is size $n_q \times (n_a-1)$, H_{21} is size $(n_a-1) \times n_q$, and H_{22} is size $(n_a-1) \times (n_a-1)$ matrix, and each is a sub-matrix of R_{12}, R_{21}, and R_{22}, respectively. n_a is the number of pads connecting to power supply as defined in problem 1.

3. Solving Eq. (5) for the observation point voltages yields

$$\boldsymbol{v}_q = (R_{11} - H_{12} H_{22}^{-1} H_{21}) \boldsymbol{J}_q + H_{12} H_{22}^{-1} \boldsymbol{v}_{P_s}, \qquad (6)$$

Since $\boldsymbol{v}_{P_s}$ is defined as relative supply voltage to the reference node, $\boldsymbol{v}_{P_s} = \boldsymbol{0}$ for the cases with single supply voltage. Then $\boldsymbol{v}_q$ becomes

$$\boldsymbol{v}_q = (R_{11} - H_{12} H_{22}^{-1} H_{21}) \boldsymbol{J}_q. \qquad (7)$$

The first term in Eq. (6) is the voltage drop due to the combination of chip power consumption and given PDN. The second term is the voltage changes due to the different supply voltages other than V_{dd}. $\boldsymbol{v}_{P_s}$ becomes non-zero only when the power supplies at the pads are non-ideal such as the case where voltage drop in packages or printed circuit boards are included. Equation (7) also consists of two terms. $R_{11}\boldsymbol{J}_q$ is the voltage drop when the supply voltage is provided through the reference node only. The second term $H_{12} H_{22}^{-1} H_{21} \boldsymbol{J}_q$ is the voltage recovery by other pads that are used as additional power supply pads to the reference pad.

Algorithm 1 Successive_pad_assignment()

---

determine_reference_pad()
R_{eq} =*construct_equivalent_resistance_matrix()*
for $i = 1..n_p$ **do**
   pad_order(i) = i;
end for
for $j = 1..n_p$ except for reference pad **do**
   for $i = j..n_p$ **do**
      $H_{12,(j)=}$ append column i of R_{eq} to $H_{12,(j-1)}$
      $H_{21,(j)=}$ append row i of R_{eq} to $H_{21,(j-1)}$
      $H_{22,(j)}^{-1}$ =*incremental_matrix_inverse*$(H_{22,(j-1)}^{-1})$
      $\boldsymbol{v_q} = (R_{11} - H_{12}H_{22}^{-1}H_{21}) \, \boldsymbol{J_q}$
      store $(i_{min} = i, u, H)$ if $u = \max(\boldsymbol{v_q})$ is the smallest
   end for
   best_index = i_{min}
   store $(H_{22,(j-1)}^{-1}, H_{12,(j-1)}, H_{21,(j-1)})$
   swap_matrix$(R_{eq}, n_q + best\_index, n_q + j)$
   swap_vector(pad_order, best_index, j)
   *best_pad*=pad_order(*best_index*)
end for

4.4 Successive Pad Assignment Algorithm

Combining the above procedures, we construct a pad assignment algorithm. Pad assignment is a combinatorial optimization problem in which the combinations explode for problems with a large number of pads. Exhaustive search of all combinations is prohibitive for this kind of problem. Therefore, we propose an algorithm which adopts a greedy approach for assigning pad locations (Algorithm 1). The input of the SPA algorithm is a set of pad candidates that are renumbered as 1 to n_p in the vector *pad_order*. The SPA algorithm sorts *pad_order* in the pad assignment order, in which index j of the vector gives the j-th pad selection.

A procedure *Successive_pad_assignment* (SPA) starts by determining the reference pad, then the equivalent resistance matrix is determined. After that, local optimal pads are determined one by one. The inner loop of i calculates the local worst voltage drop for all possible pad assignments. A pad that yields the smallest voltage drop will be selected, after which the resistance matrix is reordered for the IMI.

4.5 Efficiency Improvement through Incremental Matrix Inverse

The SPA gives an approximation solution. It preserves previously selected pads, which prunes a significant number of trials. However, as shown in Eq. (7), the SPA process includes inverting a matrix $H_{22} \equiv A_{11,r}$ which is of size r when the user determines the r-th pad. Since H_{22} is generally a dense matrix from its construction, calculation complexity is in order of $O(r^3)$ through direct methods. When n_p or n_a is much smaller than the number of original circuit nodes, which is usually the case, the use of the equivalent resistance matrix is efficient. However, when the number of candidate pads is large, it again becomes a bottleneck. In the SPA, as it preserves already selected pads, the matrix $H_{22}^{-1} \equiv A_{11,(r-1)}^{-1}$ of size $(r - 1)$ has already been calculated
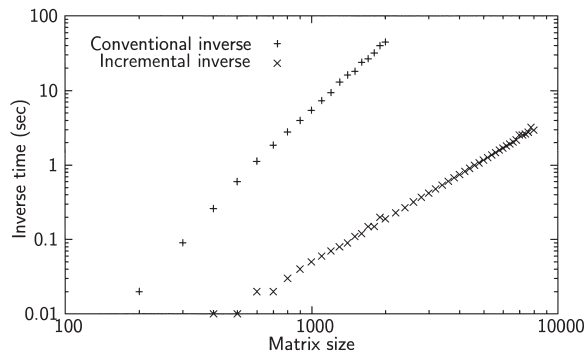
Fig. 3 CPU time required for R_{eq} inverse.

when trying to add the r-th pad. Here $A_{11,(r-1)}$ is a common sub-matrix of $A_{11,r}$. We utilize this matrix for an incremental inverse calculation of H_{22} to pursue further efficiency of the SPA algorithm.

Let A_{11} be a size r square matrix whose inverse is already calculated. A_{11} and its inverse are both symmetric by construction. We calculate a matrix B which is an inverse of A.

$$AB \equiv \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & a_{22} \end{pmatrix} \begin{pmatrix} B_{11} & B_{12} \\ B_{21} & b_{22} \end{pmatrix} = \begin{pmatrix} E_{11} & 0 \\ 0 & 1 \end{pmatrix} \quad (8)$$

Here, A_{12}, A_{21}, a_{22} are row and column vectors corresponding to the next supply pad candidate, respectively. A_{12} is a $(r-1) \times 1$ column vector, $A_{21} = A_{12}^T$, and E_{11} is a size $(r-1)$ identity matrix. Solving each component in B being A_{11}^{-1} as known yields the following set of equations.

$$B_{21} = B_{12}^T = (C_{21}A_{12} - a_{22})^{-1}C_{21} \quad (9)$$
$$B_{11} = A_{11}^{-1} - C_{21}^T B_{21} \quad (10)$$
$$b_{22} = (1 - A_{21}B_{12})/a_{22}. \quad (11)$$

Here, $C_{21} = A_{21}A_{11}^{-1}$. Computational complexity for above equations can be evaluated by the number of multiplications required and is $O(r^2)$ at the maximum which is an order of magnitude faster than conventional matrix inverse of $O(r^3)$. Figure 3 shows CPU time comparison between the conventional matrix inverse and the IMI both measured using the same Octave [12] program on a 2.8 GHz Linux workstation. Here, the conventional matrix inverse is an optimized Gaussian elimination for a dense matrix ported from LINPACK [13]. A single dense matrix inverse for a greater than 1500 port problem was calculated in less than 0.1 second, which is more than 100x faster than the conventional method.

4.6 Discussions

Since construction of Eq. (7) is independent of $\boldsymbol{J_q}$, any current distribution can be applied to ports once the R_{eq} is constructed. This is a very good property. It enables what-if analysis on different current distributions because a chip usually has several power consumption scenarios depending on its operational modes. We understand that the SPA does not necessarily give a very good solution but its

efficiency enables multiple analysis of different scenarios quickly. Choosing commonly assigned pads over different combinations of J_q should achieve a more robust solution. There are also many occasions that a designer has a solid power estimate for particular blocks but is uncertain for other blocks. Using the proposed approach, the designer can try different power consumption of the blocks by changing J_q to see the impact on the voltage drop.

Also, in this paper, the assignment is described using a power supply network as an example assuming that the locations of ground pads are already available. However, there are no technical difficulties in applying the SPA for ground networks. Therefore, a sequential assignment of ground pads and power supply pads is possible. Simultaneous assignment of the power supply and ground pads, which should result in better solution than doing it in series, is one of our future works.

5. Experimental Results

Figure 4 shows example PDN models. There are 16 supply pad candidates ($P_1, \ldots, P_{16}$; $n_p = 16$) and 4 observation points ($Q_1, \ldots, Q_4$). Representative currents connected at the observation points are: $(j_1, j_2, j_3, j_4) = (0.6, 0.5, 0.6, 0.3)$ amperes, respectively. Power supply voltages are 1.0 V for both circuits. Locations of the pads and current sink points are also the same. The difference between the two circuits is the PDN. Circuit 1 has uniform grid while circuit 2 contains obstacle blocks, such as analog circuits or power-gated circuits, to which the PDN in interest is prohibited to connect. Resistors used in the example are 100–200 mΩ.

Table 1 shows the pad selection order calculated for circuit 1 and 2. For both circuits, the reference pad is determined using the procedure in step 1, '*determine_initial_power_pad*' and the selected pad happened to be the same P_{12}. Next, the second pad is assigned by calculating the voltage drop v_q using all remaining pad candidates. The best choice as the second pad for circuit 1 is P_2. P_2 is located on the opposite edge to the reference pad, which is quite reasonable. The first four pads are selected from four different edges and located about the center of the edges since the resistor network structure used in circuit 1 is regular and power consumption spreads almost diagonally over the chip. For the circuit 2 with a different resistor network but the same current position, the first four pads are the same. Figure 5 plots the worst voltage drop as a function of number of selected pads. The sharp rise for the first four pads in the best pad selection means that these pads are very critical for circuit 1. Table 1 also shows the example of ineffective pad selection order. Instead of picking up the best possible pad, we choose the worst pad that improved the worst voltage drop the least when adding a power supply pad. Compared with the best pad selection, inefficiency is obvious by looking at the first four selections by noticing that neighboring pads at the chip corner are selected in sequence. The worst voltage drop is accordingly large. The
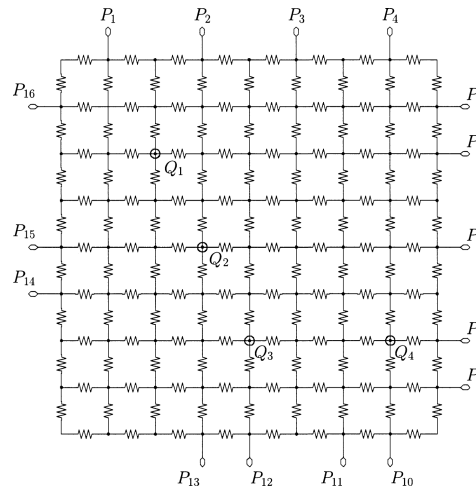
(a) Circuit 1.
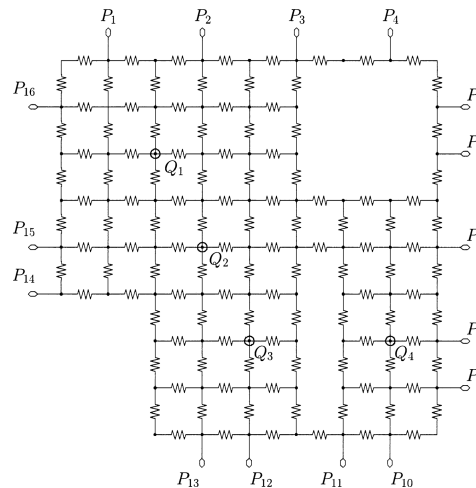
(b) Circuit 2.

Fig. 4 Example circuits with different PDN. P_1 to P_{15} represent power supply candidates and Q_1 to Q_4 represent current sink and voltage observation points.

Table 1 Efficient (the best) and inefficient (the worst) pad assignments. Numbers correspond to the pad numbers in Fig. 4.

Assign. order		1	2	3	4	5	6	7	8
	ckt	9	10	11	12	13	14	15	16
best	#1	12	2	8	15	6	11	16	14
		3	7	13	1	4	9	10	5
	#2	12	2	8	15	11	16	7	13
		3	14	1	10	9	4	5	6
worst	#1	5	4	6	3	2	1	16	7
		8	9	10	11	15	14	13	12
	#2	5	6	4	7	8	9	10	11
		13	12	3	14	15	16	1	2

voltage drop differences between the best and the worst pad selection are 43 mV (110%) and 88 mV (200%) for circuit 1 and 2, respectively, with 3 pads. Here the error is defined as (voltage drop difference)/(V_{dd}−voltage drop when all candidate pads are used).

If the target voltage V_t of the worst voltage drop were set to 50 mV, 5 pads are sufficient in the best pad selection
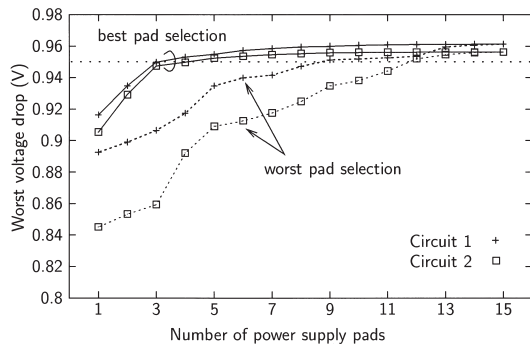
Fig. 5 The voltage drop for different pad combinations. Pin selections correspond to Table 1. Poorly chosen pad set increases voltage drop significantly.

Fig. 6 Solution quality verification. The difference between pad assignment results using SPA and the optimal results by exhaustive search is small.

for both circuits. On the other hand, the poor pad selection requires 10 pads for circuit 1, and 12 pads for circuit 2. Required pad number difference is much larger for circuits which are larger, are more irregular in shape, or have more uneven current distribution. Consulting Table 1 and Fig. 5 also provides us with the critical pads for the combination of this PDN and J_q. Pads P_{12}, P_2, P_8, and P_{15} are critically important in these examples. Selecting those pads compensates voltage drop significantly. We also find that assigning 6 power supply pads including critical pads is sufficient for these examples in the best pad selection because the worst voltage drop is almost saturated beyond this point. Saturation of the curves in Fig. 5 suggests that there may exist many redundant power supply pads if assignments are based on human intuition only. Through this experiment, we understand that the pad assignment is important and it has to be well considered when an on-chip PDN is constructed.

We verified the SPA's solution quality by comparing the assignment with an exhaustive search (AES) solution. Even for these small examples, the AES for all possible combination is very expansive since the number of trials is the sum of the combination and it has an exponential dependence on n_p, which is $\sum_i^{n_p} \binom{n_p}{i} = 2^{n_p}$. In circuit 1 and 2, the total number of trials required for AES is 65,536 while SPA reduces it to 120. Figure 6 compares maximum voltage drop between the solutions obtained by the SPA and the AES. The largest difference is 1 mV and 0.4 mV, respectively, for circuit 1 and 2. This result confirms that the SPA calculates a practically good solution in a significantly reduced time.

As more practical examples, Table 2 compares computational time of the SPA algorithm. Circuit 3 to 5 differ in modeling granularity. In circuit 3, the PDNs in the lower metal layers are reduced in order to make the resistor network smaller. The number of observation points and candidate pads are equal for these three circuits. The column #trials indicates the total number of simulations to find the local optimum node required in the SPA. Setup is time required to read the PDN netlist and to formulate circuit matrices used in the following steps. Steps 1, 2, and 3 in
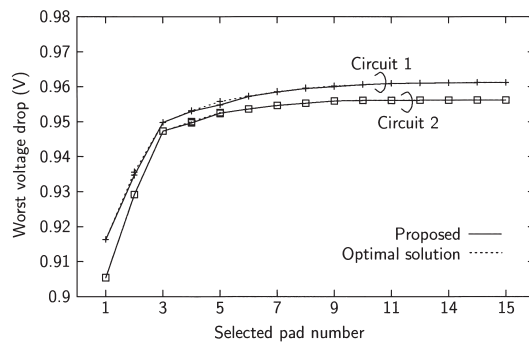
Table 2 correspond to the steps of the SPA algorithm described in Sect. 4. Step 1 and step 2 are the time required for DC analysis to determine the reference pad and the equivalent resistance matrix. In these steps, we used a general asymmetrical version of a sparse linear solver working on a 2.8 GHz processor [14], [15]. Step 3 is the time required for the complete pad order enumeration. Even for a circuit with more than 400 K resistors, preparation for R_{eq} required only a few minutes. Once R_{eq} is constructed, finding pad assignment order requires only 3 seconds and is independent of the original circuit size. Exploration of the pad assignment can be conducted very efficiently.

Circuits 6 through 9 are various examples with different observation points and pads. We see that the number of observation points has weak impact on the calculation time of the SPA but the number of pads has a strong effect. This comes from the fact that the SPA tries to find all pad orders—the number of trials to find a local optimum pad is $n_p(n_p - 1)/2 \simeq O(n_p^2)$. Being that the IMI lies in the innermost loop, overall calculation complexity becomes $O(n_p^4)$. Although calculation for a few hundred pads are sufficient for the pad assignment problems in current generation LSI, further heuristics such as the divide and conquer approach described in [11] may be required to solve full pad ordering problem of size $n_p > 500$. The IMI can be combined with any of these heuristics.

The last column in Table 2 shows estimated total calculation time for the conventional pad order selection. We define the following heuristics as the conventional method: starting from an initial power supply pad to the n_p-th pad, a designer enumerates pad order one by one from the remaining candidates through his or her intuition. Each time a new supply pad is added to the selection, the worst voltage drop is recalculated to obtain the worst voltage drop in order to check if the selected pads satisfies target voltage drop constraint or not.

In the conventional pad assignment, CPU time required for each voltage drop recalculation is equivalent to the sum of 'setup' and 'step 1' since circuit topology changes. Therefore, the required simulation time for conventional pad assignment can be estimated as $n_p \cdot (T_{setup} + T_{step1})$. The

Table 2 Computational time comparison for various circuit examples.

circuit	#resistor	#node	# ports		SPA	SPA: CPU time (sec.)					Conv.: CPU
			n_q	n_p	# trials	setup	step 1	step 2	step 3	total	total (sec.)
3	34080	22519	81	68	2278	3.1	1.4	12.8	3.0	20.3	306.0
4	170027	111767	81	68	2278	28.4	4.8	64.6	3.0	100.2	2257.6
5	417462	273896	81	68	2278	128.6	13.4	166.7	3.0	311.7	9656.0
6	12043	8340	200	100	4950	1.2	0.9	8.9	25.5	36.5	210.0
7	201000	121352	50	100	4950	39.2	6.4	77.8	6.0	129.4	4560.0
8	71097	43498	104	292	42486	9.8	2.4	71.0	360.5	443.7	3562.4
9	139288	91850	200	400	79800	28.7	4.4	215.6	2388.2	2636.9	13240.0

SPA algorithm achieves up to 40x speed improvement in these examples. In addition, because the SPA generally provides better pad selection order, the SPA process can be terminated with less number of selected pads than the conventional method, which further accelerates the calculation time.

6. Conclusion

A pad assignment procedure called SPA which minimizes voltage drop on a given power supply network is proposed. The proposed procedure enumerates both location and number of power supply pads to satisfy voltage drop specification. The PDN reduction using equivalent resistance matrix which preserves both pads and on-chip current sinks as ports efficiently serves for on-chip voltage drop calculation and eliminates re-generations and re-decompositions of the circuit matrix due to topological change. By the incremental use of already obtained matrix inverse, SPA accelerates voltage drop calculation time significantly. Experimental results showed significant speedup compared with the conventional method. Experiments also revealed that inefficient pad selection increases the number of power supply pads substantially.

References

[1] SIA, International Technology Roadmap for Semiconductors, 2003 Edition, Assembly and Packaging; Interconnect.

[2] A. Dalal, L. Lev, and S. Mitra, "Design of an efficient power distribution network for the UltraSPARC-I^{TM} microprocessor," Proc. ICCD, pp.118–123, Oct. 1995.

[3] A. Dharchoudhury, R. Panda, D. Blaauw, and R. Vaidyanathan, "Design and analysis of power distribution networks in PowerPC^{TM} microprocessors," Proc. DAC, pp.738–743, June 1998.

[4] J. Kozhaya, S. Nassif, and F.N. Najm, "A multigrid-like technique for power grid analysis," IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst., vol.21, no.10, pp.1148–1160, Oct. 1998.

[5] Y.M. Lee and C.C.P. Chen, "The power grid transient simulation in linear time based on 3D alternating-direction-implicit method," Proc. DATE, pp.1020–1025, 2003.

[6] H. Qian, S.R. Nassif, and S.S. Sapatnekar, "Random walks in a supply network," Proc. DAC, pp.93–98, June 2003.

[7] M. Pedram, K. Chaudhary, and E. Kuh, "I/O pad assignment based on the circuit structure," Proc. ICCD, pp.314–318, Oct. 1991.

[8] A.H. Ajami, K. Banaerjee, A. Mehrotra, and M. Pefram, "Analysis of IR-drop scaling with implications for deep submicron P/G network designs," Proc. ISQED, pp.35–40, May 2003.

[9] M. Marek-Sadowska, "Pad assignment for power nets in VLSI circuits," IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst., vol.6, no.4, pp.550–560, July 1987.

[10] J. Oh and M. Pedram, "Multi-pad power/ground network design for uniform distribution of ground bounce," Proc. DAC, pp.287–290, June 1998.

[11] M. Zhao, Y. Fu, V. Zolotov, S. Sundareswaran, and P. Panda, "Optimal placement of power supply pads and pins," Proc. DAC, pp.165–170, June 2004.

[12] "GNU octave," http://www.octave.org/

[13] J.J. Dongarra, C.B. Moler, J.R. Bunch, and G. Stewart, LINPACK Users' Guide, SIAM, Philadelphia, PA, USA, 1979.

[14] O. Schenk, M. Hagemann, and S. Röllin, "Recent advances in sparse linear solver technology for semiconductor device simulation matrices," Proc. SISPAD, pp.103–108, Sept. 2003.

[15] O. Schenk and K. Görtner, "Solving unsymmetric sparse systems of linear equations with PARDISO," J. Future Generation Computer Systems, vol.20, no.3, pp.475–487, 2004.

Takashi Sato received the B.E., M.E. degrees from Waseda University, Tokyo, Japan and the Ph.D. degree from Kyoto University, Kyoto, Japan, respectively. From 1991 to 2003, he worked for Hitachi, Ltd., where he was engaged in the design and development of analog circuit simulator, high-speed processor – memory interface circuits. Since 2003, he is with Renesas Technology Corp., where he has been engaged in signal and power supply integrity analysis and related design methodologies. He was a visiting industrial fellow at the University of California, Berkeley, from 1998 to 1999. His research interests include analog circuit simulation techniques, on-chip and on-board interconnect modeling, signal and power supply integrity analysis, and their application to high speed interface circuits. Dr. Sato is a member of the IEEE. He received the Beatrice Winner Award at ISSCC 2000 and the Best Paper Award at ISQED 2003.

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 2001, he was an Instructor in Department of Communications and Computer Engineering, Kyoto University. 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. He is a member of IEEE, ACM and IPSJ.

Hidetoshi Onodera received the B.E., and M.E., and Dr. Eng. degrees in Electronic Engineering from Kyoto University, Kyoto, Japan, in 1978, 1980, 1984, respectively. Since 1983 he has been a Research Associate (1983–1991), an Associate Professor (1992–1998), and a Professor (1999-) in the Department of Communications and Computer Engineering, Graduate School of Informatics, Kyoto University. His research interests include EDA and design methods for digital/analog/RF circuits, design for high speed and low power, and design for manufacturability. He is a member of the Information Processing Society of Japan, IEEE, and ACM.