On Increasing Signal Integrity with Minimal Decap Insertion in Area-Array SoC Floorplan Design

Chao-Hung Lu
Department of Electrical Engineering
National Central University
Taoyuan, Taiwan, R.O.C.
Email: chlu@ee.ncu.edu.tw

Hung-Ming Chen
E.E. Department and SoC Research Center
National Chiao Tung University
Hsinchu, Taiwan
Email:hmchen@mail.nctu.edu.tw

Chien-Nan Jimmy Liu
Department of Electrical Engineering
National Central University
Taoyuan, Taiwan, R.O.C.
Email: jimmy@ee.ncu.edu.tw

Abstract— With technology further scaling into deep submicron era, power supply noise become an important problem. Power supply noise problem is getting worse due to serious IR-drop and simultaneous switching noise, and decoupling capacitance (decap) insertion is commonly applied to alleviate the noise. There exist some approaches to addressing this issue, but they suffer either from over-design problem or late decap insertion during design stage. In this paper, we propose a methodology to insert decap in a more efficient and effective way during early design stage in area-array designs. The experimental results are encouraging. Compared with other approaches in [15] and [12], we have inserted enough decap to meet supply noise constraint while others employ more area.

I. Introduction

As VLSI technology enters the nanometer era, signal integrity become an important issue in the modern VLSI design. Basically, the integrity problem can be categorized in the signal noise issue and the power supply noise issue. In a modern low-power chip design, the supply voltage is continually dropped to reduce the power consumption. As a result, power supply noise becomes first order effect. Since the resistance of the interconnect would consume the energy of the power, power/ground (P/G) topology [8] [10], wire sizing [13] [3] could influence the voltage drop in the chip. In [10], Singh et al. use locally regular and globally irregular grids to optimize P/G design.

Employing decoupling capacitances (decap) has been a commonly used approach to reduce this kind of noise. Since manufacturing techniques continuously develop into the next generation, the gate oxide thickness is continually reduced. This results in the importance of the decap placement and sizing in digital chip design. In [15] and [12], they use decap insertion methods to improve power supply at floorplan level. Because some physical factors, such as the current of VDD pins would be durably supplied when the chip is turned on, are not considered, the decap budget could be conservatively computed and the floorplan area would be overly increased. Besides [15], [14] and [11] propose decap insertion method to improve power supply noise in placement level. In [14], the author method includes one prediction method and one correction method. In prediction step, the required decap is pessimistically estimated. In correction step, the decap size is continually reduced until an appropriate size is obtained. We believe that it is better to insert decaps during floorplanning since the floorplanning stage is early for flexible changes in layout, while a better model to calculate decap budget is needed.

Due to recent high-performance IC manufacturing, the flip-chip and area-array architectures are applied, therefore the packaging issue should be accounted for in floorplan design. In [2], authors provide a power bump allocation method to effectively reduce the distance with from the supply voltage nodes to modules, thus reducing power supply noise. However, without decap insertion, resultant floorplans will still suffer from supply noise violations. Therefore, one simple and effective way to insert appropriate size of decap is to place the high current consumption module away from another high current consumption module in floorplanning. For the floorplan representation, Guo et al. [4] first propose an O-tree representation to maintain the adjacent relation for each block. To obtain a better floorplan result and less runtime program, [1] proposes a B*-tree floorplan representation to represent a floorplan structure. Recently, [12] develops a DBL representation to plan all modules and also integrate the decap insertion into floorplanning.

In this paper, we develop O-tree-based floorplan [4] operations with a SA (Simulated Annealing)[6] approach to suppress the power supply noise. The reason for using the O-tree representation in our framework is that the horizontal relation and the vertical relation for each block could be better recorded, althouth [1] can obtain smaller floorplans. Our contributions presented in this paper are summarized as follows.

- We have developed a better noise estimation model and obtained less decap area, compared with previous approaches.
- We have adopted strong adjacent module relation O-tree representation as our engine for supply noise driven floorplanning, and successfully modified the primary operations $Delete$ and $Insert$ to be used in our framework.
- We have presented an algorithm to insert minimal decaps into a noise-guided resultant floorplan, with blocks and decaps legalization.

The rest of the paper is organized as follows. Section II describes our noise estimation method, decap budget computation and problem formulation. The floorplanning approach and decap insertion algorithm are presented in Section III. Experimental results are shown in Section IV. We conclude the paper in Section V.

Acknowledgements : This work was supported in part by the National Science Council under Contract NSC 95-2220-E-008-006 and NSC 95-2220-E-009-007. Their supports are greatly appreciated.

1-4244-0630-7/07/$20.00 ©2007 IEEE.

II. Power Delivery and Signal Integrity Issues in Area-Array Design

In this section, we describe our power delivery model and noise estimation model used in this paper and formulate our problem.

A. Power Delivery Model and Noise Estimation

In this paper, the power source distribution is based on the area-array architecture, as illustrated in Fig 1. If the area-array architecture is used in the chip, the distance from the I/O location to the connection point of the model would be substantially decreased and the performance would be significantly improved. Therefore, the area-array architecture is extensively used in the high-performance chip.

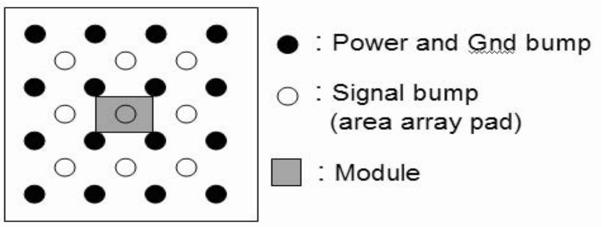

Fig. 1. Area-array footprint SoC. The Vdd and Gnd bumps are uniformly distributed across the die with signal bumps in fixed interspersed locations.

Based on the area-array architecture, the power source of each module only considers four VDD sources – right-top, right-down, left-top and left-down – and other VDD sources are ignored. We use [15] model to compute our power noise and the inaccuracy ratio(compared with SPICE results) is less than 10%. According to [15], the noise calculation of each module can use Kirchhoff's voltage law to represent as follows:

$$V_{noise}^{(k)} = \sum_{P_j \epsilon T^k} i_j R_{P_{jk}} + L_{P_{jk}} \frac{di_j}{dt} \tag{1}$$

where $V_{noise}^{(k)}$ denotes the power supply noise at module k , P_j denotes the path from VDD to node j, P_{jk} denotes the path from node j to node k ,T^k denotes the union of shortest paths and the second shortest paths, $R_{P_{jk}}$ denotes the resistance of P_{jk}, $L_{P_{jk}}$ denotes the inductance of P_{jk} and i_j is the current flowing along path P_j.

B. Decap Budget Computation

Here we introduce the computation methodology for the suitable decap budget of each circuit module. Our computation methodology is based on the difference between the maximum and general current consumption for each module. We use a simple example to explain our computation method. The current consumption of module k in the general condition is defined as I_{gen}^k, $k = 1, 2, ...M$ and the maximum switching current of module k is I_{max}^k. For simplicity, the switching current waveforms are modeled as triangles as shown in Fig.2.

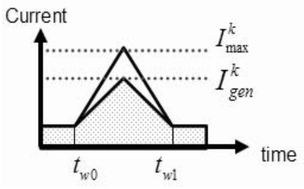

Fig. 2. Switching current consumption profile of module k. I_{gen}^k is only supplied by VDD pins and I_{max}^k is supplied by VDD pins and a decap budget.

TABLE I
Capacitance needed from different approaches. The supply voltage is 2.5V and the drop voltage constraint is set to be 0.04V. If we use equation(2) to compute the required capacitance, it is small than [15]

Method	Capacitance	Drop Voltage
Initial	0pF	2.44V
[15]	112pF	2.4715V
Equation(2)	96pF	2.4609V

The current consumption of the module in the general condition is only supplied by VDD pins, the maximum switching current can be supplied by VDD pins and vicinal decaps in a correct chip. Therefore, let C^k be the decap required for circuit k and Q^k is the amount of electric charge of the C^k. Q^k is given by the following equation (based on the model shown in Fig. 2):

$$Q^k = \int_{t_{w0}}^{t_{w1}} I_{max}^k(t)dt - \int_{t_{w0}}^{t_{w1}} I_{gen}^k(t)dt \tag{2}$$

where t_{w0} is the start time and t_{w1} is the finish time when the target module is in operational mode. The electric charge can be converted to the slicion area of the capacitance fabrication as follows:

$$C^k = \frac{Q^k}{V_{con}} \tag{3}$$

$$S_{decap} = \frac{C_{decap}}{C_{ox}} \tag{4}$$

$$C_{ox} = \frac{\epsilon_{ox}}{t_{ox}} \tag{5}$$

where V_{con} is the noise constraint of the voltage, C_{decap} is the decap budget and S_{decap} is the silicon area of C_{decap}. We use SPICE to verify our model (2) and compare with [15]. In our experiment, the supply voltage is set to be 2.5V and the power supply noise limit is set to be 0.04V. The experiment result is shown in Table I. It is obvious that the model we propose can obtain less required decap, while [15] conservatively estimate the decap budget.

C. Problem Formulation

In traditionally, the decap insertion method is applied to solve the power supply noise after the routing level. In this paper, we propose to solve this problem in early design stage. Given a set of blocks, $B_1, B_2, ...B_m$, current consumption, I_{gen}^i and I_{max}^i, of each block $B_i, 1 \leq i \leq m$, the locations for each VDD source and the noise constraint for each module, find a feasible solution such that each block B_i obtains appropriate and minimal decap budget size DBS_i, and minimum penalty area when DBS_i is inserted.

III. Minimal Decap Allocation in Power Supply Noise Aware Floorplanning

We have developed a two-step methodology to reduce and suppress noise at the floorplan level, as shown in Fig.3. First, we use a noise-driven floorplan algorithm to suppress the noise. For each block, we consider the worst case current consumption. The power supply noise would be substantially increased when two high current consumption blocks are placed at the adjacent locations. Therefore, high current consumption blocks are not abut in our floorplan algorithm.

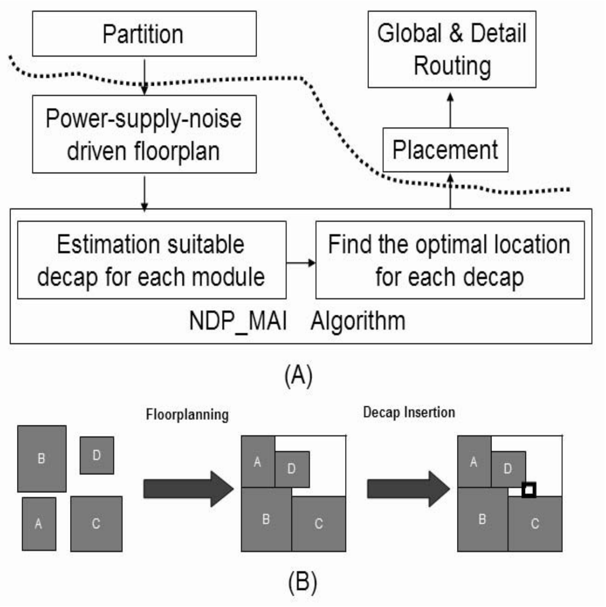
Partition
Global & Detail Routing
Power-supply-noise driven floorplan
Placement
Estimation suitable decap for each module
Find the optimal location for each decap
NDP_MAI Algorithm
(A)
Floorplanning
Decap Insertion
B
D
A
C
A
D
B
C
A
D
B
C
(B)

Fig. 3. Using a two-step approach that includes a power supply noise(PSN) driven floorplanning algorithm and a decap insertion algorithm to improve the power supply noise before routing level. (A) The flow chart of our methodology. (B) The illustration of steps.

The second step is the decap insertion method. We use a Noise-driven Decap Planning with Minimum Area Insertion(NDP_MAI) algorithm to reduce the noise after floorplanning. In this step, given an initial compacted floorplan and the current consumption for all blocks, the NDP_MAI algorithm use is to use less decap budget and extensive floorplan area to determine the locations of the inserted decaps for each block to satisfy the noise constraints.

A. O-Tree Based Power Supply Noise Aware Floorplanning

If two high current consumption blocks are placed at the adjacent location, the power supply noise problem will get worse, and the performance of the chip will be reduced. Among various floorplan representations in Sequence Pair [9], B*-tree [1], O-tree [4], TCG [7], CBL [5], and DBL [12], we have adopted O-tree representation as our engine for supply noise driven floorplanning due to its strong adjacent module relation. The main reason is : (1) O-tree representation[4] can immediately view the adjacent relation; (2) the tree structure can easily control each block location and the cost is small then the graph-based floorplan representation. In O-tree representation, it uses a horizontal tree structure and a vertical tree structure to express the floorplan result. We will briefly review the O-tree representation in next paragraph. In our approach, two transformation operations, *Delete* and *Insert*, are applied to obtain a new floorplan result. Those operations are different from the operations in original O-tree floorplanning algorithm. We have modified them so that they can be more useful in our framework. The detail discuss is shown in Fig. 4

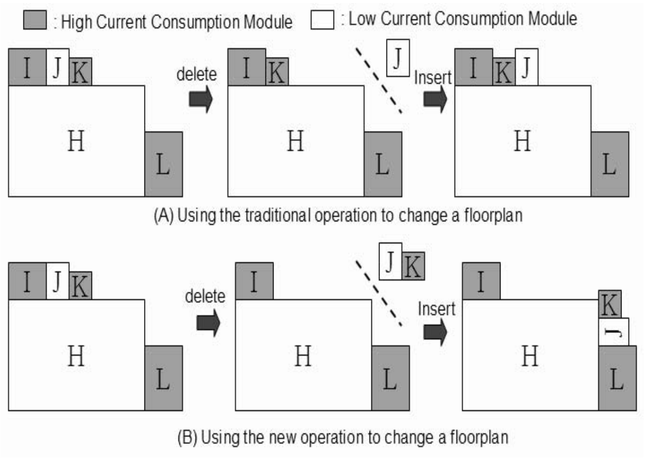
▨ : High Current Consumption Module
□ : Low Current Consumption Module
I
J
K
H
L
delete
I
K
H
L
J
Insert
I
K
J
H
L
(A) Using the traditional operation to change a floorplan
I
J
K
H
L
delete
I
H
L
J
K
Insert
I
K
J
H
L
(B) Using the new operation to change a floorplan

Fig. 4. The illustration between the usage of operations in original O-tree and our approach. If we use the original operations to change the floorplan in (A), high current consumption modules are possibly placed in the adjacent location, while new operations can improve this problem, shown in (B), where high current consumption modules are not placed together. Here we assume that we plan to delete module J.

To be more specific, we use an example to explain the *Delete* operation. In Fig.4, the horizontal tree of O-tree is set as (0011000111,HLIJK) and the vertical tree is set as (0010101101,HIJKL), where the first half of the representation denotes the horizontal(vertical) relation and the second half of the representation is the order list of blocks. If the module J is selected to delete, we will obtain the delete module list from the parts of the second half representation, using the intersection operation in two trees, from the delete module to the last module, such as JK ∩ JKL. Then we modify the original O-tree representation according to the result of the intersection operation, until the list is not shown in the O-tree representation. The illustratin is shown in Fig.5.

The *Insert* operation can be divided into three parts: (1) find all possible locations; (2) compute cost; (3) choose optimal location. We use a simple example to explain the *Insert* operation. First, all possible locations are found as shown in Fig. 6(A). We define the possible locations as the left-down corner of the empty space in current floorplan. As shown in Fig.7, the area of the floorplan is possibly increased when the module J is inserted in the floorplan. Hence, in our cost function, it considers the floorplan area and the local current consumption. The cost function can be represented as follows:

$$C_a = D_1(A_{new} - A_{original}) + D_2(I_a + I_b + I_c - I_{th}) \qquad (6)$$

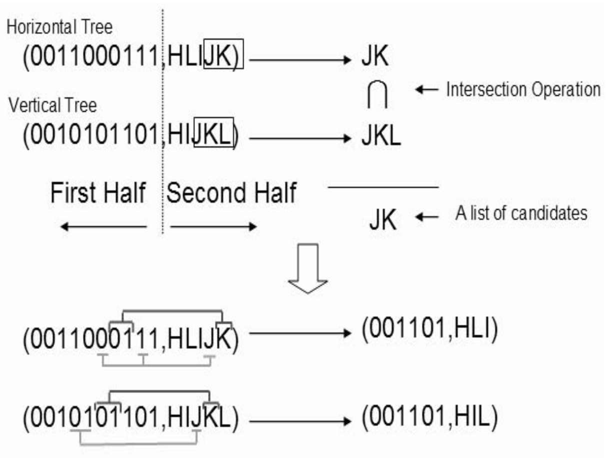

Fig. 5. Using new *delete* method to delete the module J. First, the candidate list can be obtained by applying intersection operation. Second, the original O-tree representation is modified until the list is not shown in the O-tree representation.

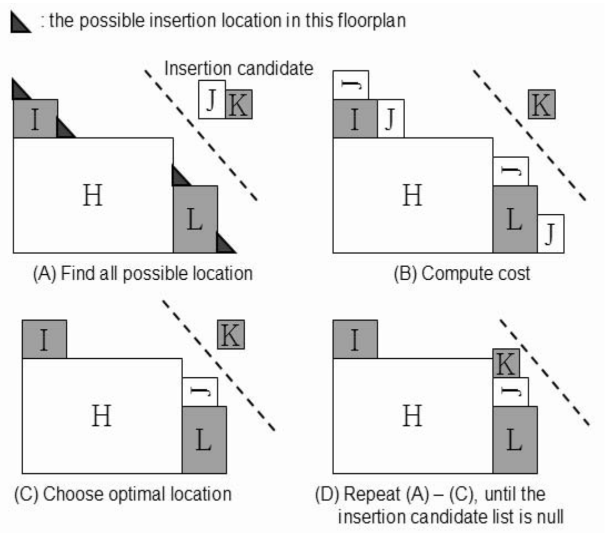
: the possible insertion location in this floorplan
Insertion candidate
J
K
I
H
L
(A) Find all possible location
(B) Compute cost
(C) Choose optimal location
(D) Repeat (A) – (C), until the insertion candidate list is null

Fig. 6. Using new *Insert* operation to insert all candidate modules. First, all possible insertion locations in one floorplan are found as shown in (A). Second, the cost of the location is computed when the module is inserted in one of the possible locations. In (B), all location costs are computed when module J is inserted in the candidate location. Then the minimum cost location is chosen as shown in (C).

where C_a denote the cost when the module A is inserted in this location, D_1 and D_2 are the weights, A_{new} is the area of the floorplan after the module is inserted, $A_{original}$ is the original area, $I_{a(b,c)}$ denotes the current consumption of the module $a(b,c)$, and I_{th} denotes the threshold value for local current consumption. D_2 is to set a large value for penalizing high local current consumption.

B. Identification of Space Priority for Decap Insertion

The space in a floorplan could be divided into three types: the extensive space, the empty space, and the available space. The edge of the extensive space and the critical channel are the same. In Fig. 8, the horizontal critical path is $B- > C$, the critical channel is the edge between block B and C. If one capacitance is inserted in the extensive space, the insertion operation will increase floorplan area as shown in Fig.8 and the topology of the floorplan would be change. In Fig. 8(A), the hollow square denotes one capacitance block and the channel space among B and C is the edge of the extensive space. If one capacitance block is inserted in this space, the circuit block and capacitance block must be moved as Fig 8(B). The area of the floorplan is then increased.

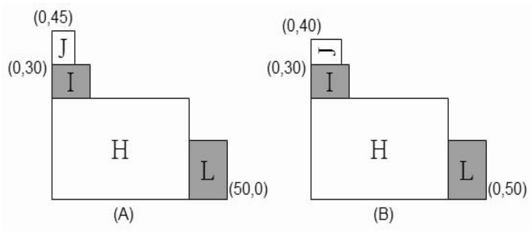

Fig. 7. One module must be inserted for two times at the same possible locations since the costs may not be the same for rotation.

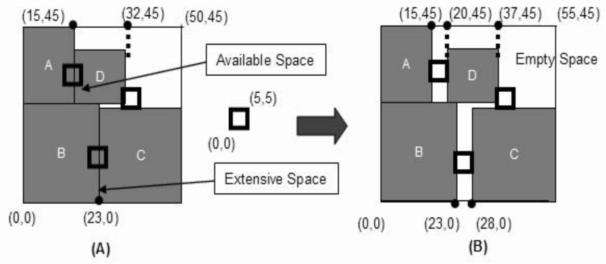

Fig. 8. When the decap is inserted into the different space in a floorplan, the area of the floorplan is possibly increased. (A) The area of the floorplan is $2250\mu m^2$ before the decap insertion. (B) The area of the floorplan is $2475\mu m^2$ after the decap insertion. The area is not increased when the decap is inserted into the empty space. If the decap is inserted into the available space, the area will be increased sometimes.

A channel space with the overlap edge of the empty room is called the empty space. If one capacitance is inserted in this space, all the blocks will be located on the original positions and the floorplan area will not be increased. The space in a floorplan is called the available space if it is not empty space or extensive space. If a capacitance is inserted in this space, the area of the floorplan would not be extended, but the location of some blocks could be changed. If one capacitance is inserted between the adjacent blocks A and D as in Fig. 8(A), the block D would be horizontally moved to its right. The result is shown in Fig. 8(B). Initially, the block A is the right block of the block D. After inserting decap, the right block of the block D is decap block and the area of the floorplan is not extended.

In summary, if the decap would be inserted in one floorplan and the candidate location of the decap has three type space could be selected – Available Space, Extensive Space and Empty Space, the priority of spaces must be defined. In this paper, we set the priority rule as follows :

Empty Space $>$ *Available Space* $>$ *Extensive Space*

C. Decap Insertion for Power Integrity

After obtaining one resultant floorplan from the first step, we will calculate required decap size for insertion. In order not to increase the floorplan area excessively, we insert the decap with four smaller decaps instead of inserting one big decap in the extensive space. Given floorplan shown in Fig. 9, block D needs decap to supply the current consumption. First, we use equation (2)-(5) to compute the optimal decap sizing when the location of the decap and the VDD source of the module is separated. Second, the decap is cut apart into four smaller decap by the Manhattan distance from the VDD source to the power bump and the feasible region of each smaller decap is a rectangle from the connection point to the VDD pin. We compute the area cost of each space in the feasible region when the decap is inserted. If decap location and the connection point location are not close, the decap must add charge to compensate the consumption of the wire. The compensation methodology is shown as follows:

$$Q_{com}^k = Q^k + l \times c \tag{7}$$

where l is the distance from the space to the connection point and the c is wire capacitance per unit length. Finally, the minimum cost space be selected to inserted the decap. We called this methodology NDP_MAI Algorithm and illustrated as follows:

NDP_MAI Algorithm

```
begin
  while $The\ power\ supply\ noise\ for\ block\ B_k\ not\ estimated$
    if $I_{max}^k > I_{gen}^k$
      $Using\ equation\ (2)-(5)\ to\ compute\ decap\ DBS_k;$
      $The\ original\ decap\ is\ partitioned\ into\ four$
      $smaller\ decap\ based\ on\ the\ distance\ from\ VDD$
      $pins\ to\ the\ connection\ point\ of\ module\ k;$
      for $each\ smaller\ decap$
        $Find\ the\ feasible\ region\ of\ the\ decap;$
        $Fix\ all\ the\ space\ in\ the\ feasible\ region;$
        for $each\ space\ in\ the\ feasible\ region$
          $Compute\ area\ cost\ when\ decap\ insert\ in$
          $this\ space;$
        end
        $Select\ the\ minimum\ cost\ space\ to\ insert$
        $decap;$
      end
    end
  end
end
```

IV. Experimental Results

The proposed power supply noise aware floorplanning methodology and NDP_MAI algorithm have been implemented using C++ language on an AMD 3200 machine with 1G memory. The set of useful parameters of the formulation in the power delivery model is based on the $0.25\mu m$ technology and listed in Table II. Five MCNC benchmark circuits, apte, hp, xerox, ami33 and ami49, are used to test the performance of proposed methodology. The power supply voltage is 2.5V and the distance between two continuous VDD is $1000\Omega/\mu m$ and the power supply mesh is $333.3\Omega/\mu m$. The I_{gen}^k for the module k is $A_k \times D_c$, where A_k is area of module k, D_c is the worst case current density. The I_{max}^k is assigned as a random value to be $1.05I_{gen}^k \sim 2I_{gen}^k$. Since the MCNC benchmark include no noise constraint, the noise constraint is set to be 0.25V. In our experiments, the operation time t_{w0} and t_{w1} of the switching current waveform are set to be 0.3ns and 0.8ns.

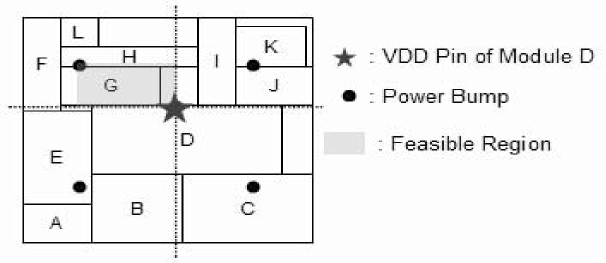

Fig. 9. The original decap budget is partitioned into four smaller decap. For each smaller decap, the feasible insertion region is from a power bump to the the VDD source. By this consideration, the smaller capacitance could be inserted into the trivial empty space and the charge time of the capacitance could be substantially decreased.

TABLE II
Parameters used in this paper, based on $0.25\mu m$ technology.

Parameters	Description	Value
r	Unit-length wire resistance	0.0125 ($\Omega/\mu m$)
c	Unit-length wire capacitance	0.4 ($fF/\mu m$)
l	Unit-length wire inductance	20 ($pH/\mu m$)
L_p	Package inductance per VDD pin	0.2 (nH)
R_p	Package resistance per VDD pin	0.5 (Ω)
j_s	Worst case current density	0.2 ($\mu A/\mu m^2$)

TABLE III
Decap obtained from our model and [15] on the same resultant floorplan. This shows our estimation needs less decap to suppress noise.

Circuit	Decap Budget (nF)	Decap Budget [15](nF)	Decrease Ratio
apte	7.01	13.46	47%
hp	1.65	2.75	40%
xerox	3.09	5.71	46%
ami33	0.08	0.27	70%
ami49	3.61	9.08	60%

To compare our method with [15], the decap budgets are listed in Table III. In [12], the experimental results only show the floorplan area and runtime. So we can not compare the decap budget with [12]. In Table III, it is clear that our computation methodology obtain more suitable decap budgets. The experimental results show that our decap budget obtained average improvement of 52.6%. The main reason is that [15] computed method ignore the supply current from VDD pins and the total charge is supplied by the decap when the power supply noise is occurred.

Table IV and V show the comparison of the experimental results between the other method and our proposed decap insertion

TABLE V
EXPERIMENTAL RESULT FOR MCNC BENCHMARK CIRCUITS. THE POWER SUPPLY NOISE CONSTRAINT IS SET TO BE 0.25V. THIS SHOWS THAT OUR APPROACH INSERTS LESS REQUIRED DECAP WHILE STILL MEETING DESIGN CONSTRAINTS.

Circuit	Modules	Area(μm^2) (Floorplan)	Area(μm^2) (Insert Decap)	Area Increase (μm^2)	[15] Area Increase (μm^2)	[12] Area Increase (μm^2)
apte	9	47761324	47780360	19036	469916	354000
hp	11	9940140	10097780	157640	317503	67000
xerox	10	20630210	20705216	75006	269374	144000
ami33	33	1241440	1245266	3824	390	11000
ami49	49	37504880	37659870	154990	218000	217000

TABLE IV
THE PEAK NOISE FOR ALL MODULES IN OUR APPROACH AND IN [15]. THE SOURCE OF THE DECAP BUDGET COMES FROM TABLE III. [12] DOES NOT PROVIDE PEAK NOISE INFORMATION.

Circuit	Our Peak Noise (V)	[15] Peak Noise (V)	[12] Peak Noise (V)
apte	0.25	0.24	N/A
hp	0.24	0.23	N/A
xerox	0.25	0.21	N/A
ami33	0.23	0.19	N/A
ami49	0.24	0.20	N/A

algorithm. In Table IV, [12] does not provide the peak noise information in the experiment. In Table V, the experimental result of each benchmark circuit includes the area after floorplanning (Area(Floorplan)), the area after the decap budget insertion and the increase area after the decap insertion. The experimental results show that our proposed decap insertion algorithm use less penalty area to finish the decap insertion job in most cases and the noise constraint could be conformed in all MCNC benchmark at the same time. The aspect ratio of the floorplan is the main reason why the result of the ami33 benchmark is worse than [15].

V. CONCLUSION

The techniques of floorplanning and decap insertion can be used to reduce the power supply noise in early design stage. Based on our framework, the suitable decap size is found. Furthermore, based on the analysis of the floorplan and space priority, the decap insertion algorithm is proposed to insert decap budget. Experimental results show that the proposed method can improve the increase ratio of the floorplan area when inserting decap.

REFERENCES

[1] Y.C. Chang, Y.W. Chang, G.M. Wu, and S.W. Wu. “B*-trees: a new representation for nonslicing floorplans”. In *IEEE/ACM Design Automation Conference*, pages 458–463, 2000.

[2] H.M. Chen, L.D. Hung, I Min. Liu, and D.F. Wong. “Simultaneous Power Supply Planning and Noise Avoidance in Floorplan Design”. In *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, pages 578–587, 2005.

[3] R. Dutta and M.M. Sadowska. “Automatic Sizing of Power/Ground (P/G) Networks in VLSI”. In *IEEE/ACM Design Automation Conference*, pages 783–786, 1989.

[4] P.N. Guo, C.K. Chen, and T. Yoshimura. “An O-tree Representation of Nonslicing Floorplan and Its Application”. In *IEEE/ACM Design Automation Conference*, pages 268–273, 1999.

[5] X. Hong, G. Huang, Y. Cai, J. Gu, S. Dong, and C.K. Cheng. “Corner block list: an effective and efficient topological representation of non-slicing floorplan”. In *IEEE/ACM Design Automation Conference*, pages 764–769, 2001.

[6] S. Kirkpatrick, C.D. Gelatt, and M.P. Vecchi. ”Optimization by Simulated Annealing”. In *”Science”*, pages 671–680, 1983.

[7] J.M. Lin and Y.W. Chang. “TCG: A transitive closure graph-based representation for general floorplans”. In *IEEE/ACM Design Automation Conference*, 2001.

[8] T. Mitsuhashi and E.S. Kuh. “Power and ground network topology optimization for cell based VLSIs”. In *IEEE/ACM Design Automation Conference*, pages 524–529, 1992.

[9] H. Murata, K. Fujiyoshi, S. Nakatake, and Y. Kajitani. “Rectangle-packing based module placement”. In *IEEE/ACM International Conference on Computer-Aided Design*, pages 472–479, 1995.

[10] J. Singh and S.S. Sapatnekar. “Congestion-aware topology optimization of structured power/ground networks”. In *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, pages 683–695, 2005.

[11] H. Su, S.S. Sapatnekar, and S.R. Nassif. “An algorithm for optimal decoupling capacitor sizing and placement for standard cell layouts”. In *”International Symposium on Physical Design ”*, pages 68–73, 2002.

[12] J.T. Yan, K.P. Lin, and Y.H. Chen. “Decoupling capacitance allocation in noise-aware floorplanning based on DBL representation”. In *Internationl Symposium on Circuits and Systems*, pages 23–26, 2005.

[13] J.T. Yan, C.H. Lu, and C.W. Wu. “Simultaneous Wiring and Buffer Block Planning with Optimal Wire-Sizing for Interconnect-Driven Floorplanning”. In *Proceedings IEEE Midwest Symposium on Circuits and Systems*, pages 25–28, 2004.

[14] C.Y. Yeh and M.S. Malgorzata. “Timing-aware power noise reduction in layout”. In *IEEE/ACM International Conference on Computer-Aided Design*, pages 627–634, 2005.

[15] S. Zhao, K. Roy, and C.K. Koh. “Decoupling Capacitance Allocation and Its Application to Power-Supply Noise-Aware Floorplan”. In *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, volume 21, pages 81–92, 2002.