On Power Supply Pads Planning for Wire-bonded IC

Hui-Zhong Leong*, Ming-Yu Huang*, Hung-Ming Chen*, Chang-Tzu Lin†
*Institute of Electronics, National Chiao Tung University, Hsinchu, Taiwan
†Information and Communications Research Laboratories, Industrial Technology Research Institute, Taiwan
Email: huizhong.ee05g@g2.nctu.edu.tw, yu0750265.ee07g@nctu.edu.tw, hmchen@mail.nctu.edu.tw, ctzulin@itri.org.tw

***Abstract*—In wire-bonding technology, Input/Output (I/O) pads are located along the peripheral of integrated circuit (IC) and power pad placement is limited by available I/O pad candidates. Power pads supply voltage to the IC through power delivery network (PDN), hence insufficient power pads may cause IC failure. To overcome this problem, we propose a power pad placement algorithm for wire-bonding technology. Experimental results show that the proposed algorithm determines both power pad counts and power pad locations effectively for a given power delivery network. In addition, the worst voltage drop for the IC is guaranteed to be less than 3% of the supply voltage.**

I. Introduction

As complementary-symmetry metal-oxide-semiconductor (CMOS) technology increasingly advances, the number of transistors on an integrated circuit (IC) increases every year. Denser IC means higher power dissipation, so low-power design techniques are introduced to overcome this problem. Among these techniques, the most direct approach for reducing power dissipation is to scale down the supply voltage as power is proportional to the square of the supply voltage [1]. Since lower supply voltage leads to lower noise margin, power supply noise may cause severe impact on the ICs. As a consequence, a robust power delivery network (PDN) and optimal counts and locations of power pads are important to guarantee stable supply voltage within an IC. PDN is a network of interconnected metal wires, and they are connected to the power pads to supply sufficient voltage to the inner core of the IC. Fig. 1 shows how PDN is connected to power pads in a wire-bonding IC.

An ideal PDN should deliver reference voltage across the IC, due to the resistance of long metal wires, supply voltage drops below the desired reference voltage. This voltage drop is also known as IR drop. Many approaches have been proposed to design a robust PDN preventing high IR drop [2]–[8]. Besides that, a sufficient number of power pads is also a major aspect that will affect IR drop as PDN requires

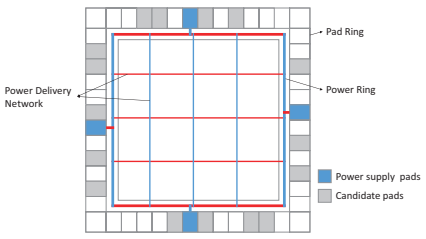


Fig. 1. An example of I/O pad ring and PDN in a wire-bonding IC.

reference voltage provided by power pads. Previous works that addressed power pads placement to minimize IR drop are proposed in [9]–[13].

We propose a power pad placement algorithm to determine power pad counts and power pad locations, such that the worst voltage drop for the IC is guaranteed to be less than 3% of the supply voltage. First, we model the given PDN as a resistive grid model. Then voltage drop analysis model proposed in [14] is utilized to calculate voltage drop of nodes on the PDN. Since a closed-form expression is used in this voltage drop analysis model, we can update voltage of nodes of PDN efficiently. Therefore, given a set of candidate pads and designer-defined PDN, our algorithm is able to find an optimal power pads configuration in a reasonable computational time.

The rest of the paper is organized as follows. First, previous works that addressed power pads placement are introduced in Section II. Section III explains details of the proposed algorithm. Section IV shows the experimental results. Final conclusions are presented in Section V.

II. Previous Works

In this paper, the focus is to decide power pad counts and power pad locations. There are many approaches that have been proposed to solve power pads placement problem. Authors in [9] adopted a mixed-integer linear program (MILP) to solve power pads placement problem. IC is first divided into regions. Worst IR drop nodes in all regions are selected as observation nodes. Power pad locations are then decided by the MILP model. [10] proposed a successive pad assignment (SPA) algorithm combined with incremental matrix inversion (IMI). The drawback of MILP and SPA is computational complexity as many matrix calculations are required. Runtime increases rapidly as the problem size increases. A force-directed function is introduced in [11] to calculate direction and the distance of power pads. Power pads are then moved into calculated locations. In [12], new locations of power pads are calculated by a weighted function. Authors in [12] mentioned that, local voltage drops can be reduced by moving power pads into new calculated locations but the global voltage drop may not be reduced. To solve this problem, power pads that located near to a moved power pad are fixed. However, methods proposed in [11] and [12] are not suitable for power pad placement problem in wire-bonding technology because power pads are only allowed to place at peripheral of IC. We use algorithm proposed by [13] to compare with our proposed method. The objective of [13] is to find optimal power pads

configuration with a fixed number of power pads and a set of candidate pad locations. First, power pads are distributed evenly then a random power pad is moved to a new location at each iteration. IR drop of the power grid is updated after each power pad movement and localized node-based iterative method is adopted to speed up voltage computation.

We propose an efficient power pads placement algorithm to determine power pad counts and power pad locations. Our contributions are stated as follows:

- Given locations of candidate pads and designer-defined PDN, our algorithm is able to determine optimal count and locations of power pads.
- The proposed algorithm is validated on four cases and the experimental results show the worst IR drops for four cases are less than 3% of the supply voltage.

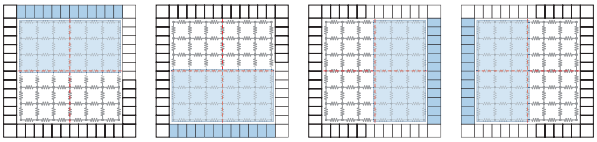
III. Power Pad Assignment Framework

Given a PDN design, candidate pad locations and IR drop constraint V_{IR}, our objective is to find optimal count and locations to assign power pads.

If we have a large number of candidate pads, finding all possible combinations of candidate pads is very time consuming. Hence, we break down our problem into smaller subproblems to speed up computational time. Candidate pads are divided into four groups according to their locations at pad ring. Then we find possible pad combination for each group G with minimum $cost$ in their corresponding $effective\ region$. The term $effective\ region$ is explained in Section III-A and a cost function is formulated to calculate the cost for pads combination in Section III-B. Final algorithm flow is shown in Section III-C.

A. Effective Region

A set of candidate pads S are divided into four groups G_{north}, G_{south}, G_{east} and G_{west} according to their locations at pad ring. Candidate pads located at north side of pad ring are grouped into G_{north}, same for G_{south}, G_{east} and G_{west}. The static IR drop increases with the distance between two grid nodes because longer distance has larger resistance. So, if a grid node ($node_i$) is located far from a power supply pad (pad_A), we can say that pad_A has minor effect on $node_i$. In other words, power pads have significant effect on grid nodes that are located at region near them, and we called that region as $effective\ region$. First, PDN grid is divided into four partitions and they are named r_{north}, r_{south}, r_{east} and r_{west} respectively. r_{north} indicates effective region that is located at upper part of PDN grid, r_{south} indicates effective region that is located at lower part of PDN grid, r_{east} indicates effective region that is located at right part of PDN grid and r_{west} indicates effective region that is located at left part of PDN grid. Candidate pads from the group G has significant effect on grid nodes located in the corresponding effective region. Fig. 2 shows four candidate pads groups G and their corresponding effective region r.



(a) (b) (c) (d)

Fig. 2. Schematic diagram for candidate pads locate at four sides of pad ring and their corresponding effective regions. Regions that are covered by blue box indicate effective regions. 2(a) G_{north} and r_{north}. 2(b) G_{south} and r_{south}. 2(c) G_{east} and r_{east}. 2(d) G_{west} and r_{west}.

B. Cost Function

A combination of candidate pads are selected from candidate pads group G and stored in a list c. The cost function Eq. 1 is adopted to calculate the $coverage$ of c.

$$cost = \sum_{i=1}^{j} V_i \ in\ effective\ region\ r, \tag{1}$$

where j is the number of grid nodes in effective region r and V_i is the voltage of grid node i located in effective region r. c with higher cost has higher $coverage$ which means that power is delivered to more grid nodes if power pads are placed in the selected candidate pad locations. Voltages of grid nodes located within an effective region are calculated because power pads have minor effects on grid nodes far away and these effects are considered to be negligible.

C. Power Pad Placement Algorithm

In this section, we proposed a power pad placement algorithm that determines optimal number and locations of power pads efficiently. Followings are definitions of variable names for power pad placement problem.

- There is a set of candidate pads S and the size of S is n.
- There are four groups of candidate pads G_{north}, G_{south}, G_{east}, G_{west} where $\{G_{north}, G_{south}, G_{east}, G_{west}\} \subset S$. $G = \{CandidatePad_1, ..., CandidatePad_m\}$, where m is the number of candidate pads in group G.
- c is the selected candidate pads with highest cost from G where $c \subset G$. All selected candidate pads c in four directions are then grouped in C, such that $C = c_{north} \cup c_{south} \cup c_{east} \cup c_{west}$.
- There exists four effective regions r_{north}, r_{south}, r_{east} and r_{west}.
- V_{IR} is the IR drop constraint.

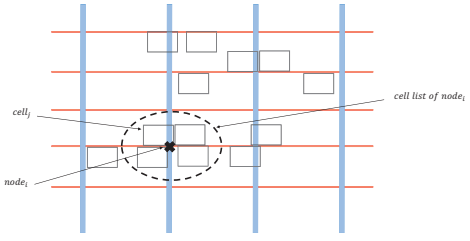


Fig. 3. The figure shows that cells are grouped in a cell list and $node_i$ is the nearest grid node to all cells in that cell list.

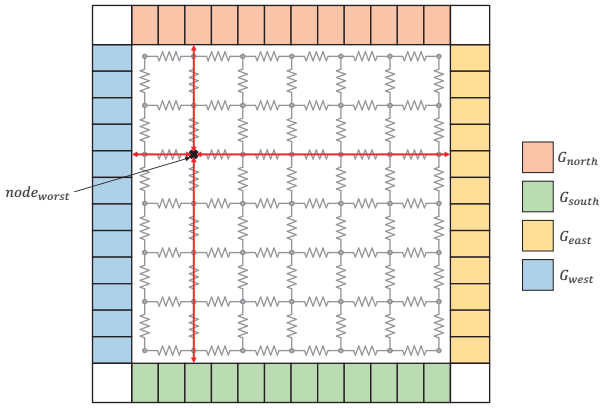


Fig. 4. An example shows that the closest candidate pad to $node_{worst}$ belongs to G_{west}.

First, a resistive grid model of PDN is built with the information provided by DEF (Design Exchange Format) and LEF (Library exchange Format) files such as locations and widths of power stripes, locations of standard cells and macros, sheet resistance and via resistance. Then for each standard cell and macro, find the closest grid node to them and these cells are added into corresponding cell list of grid nodes as shown in Fig. 3. Standard cells and macros draw power from the nearest grid node and equation below is adopted to calculate total current delivered by each grid node.

$$I_i = \frac{1}{V_{supply}} \sum_{j=1}^{k} P_j, \tag{2}$$

where V_{supply} is the reference supply voltage, P_j is the total power consumption of j-th cell and k is the number of cells in cell list of $node_i$.

Current of all grid nodes are calculated using Eq. 2, then set the grid node with largest current as $node_{worst}$. Because $node_{worst}$ needs to provide more current to cells connected to it and we can assume that IR drop at $node_{worst}$ may be the worst. Then find candidate pad that is nearest to the $node_{worst}$. Since candidate pads are divided into four groups according to their locations at pad ring, so we want to find out which group G that the closest candidate pad belongs as shown in Fig. 4. After that, we find k candidate pads from the group G with maximum cost and the cost is calculated using Eq. 1. The selected k candidate pads are stored in the list c. The closest candidate pad is not chosen directly because we want to find k candidate pads that provide sufficient voltage not only to $node_{worst}$ but also other grid nodes in the effective region of G. Next, calculate voltages of all grid nodes using IR drop analysis model proposed by [14]. Repeat the steps to find candidate pads at other directions and all selected candidate pads in four directions are placed in C. Repeat all steps until worst voltage drop is less than V_{IR}.

Pseudo code of the algorithm to find optimal count and locations of power pads that meet IR drop constraint is summarized in Algorithm 1. Our algorithm chooses a set of candidate pads from each group with maximum cost in corresponding effective region. The cost is the summation of total voltage of grid nodes in effective region. However, voltages of some grid nodes might be counted twice for two neighboring effective region and this might cause over design of power pads. To overcome this problem, we can reduce the overlapping area between two neighboring effective regions to less than one third of the area of effective region.

Algorithm 1: Power pad placement algorithm

Input: DEF, LEF, Power report, Supply voltage (V_{supply}), IR drop constraint (V_{IR}), candidate groups (G_{north}, G_{south}, G_{east}, G_{west}).
Output: Number of power pads required and locations of power pads.

```
for each cell c do
    Find closest node_i;
    Add cell c into cell list of node_i;
end
for each node do
    Calculate total current using Eq. (2);
end
Set grid node with max(I) as node_worst;
Set initial pad count, k = 1;
while drop_worst ≥ IR_constraint do
    for each direction in {north, south, east, west} do
        Find k candidate pads from candidate group G_dir;
        Calculate cost using Eq. (1);
        if cost > cost_max then
            Candidate pads are selected and stored in c;
            cost_max = cost;
        end
        else
            Find another k pads combination;
        end
        C = c_north ∪ c_south ∪ c_east ∪ c_west;
        Calculate voltages of all grid nodes;
        Set grid node with max IR drop as node_worst;
        drop_worst = voltage drop of node_worst;
        Terminates if drop_worst < V_IR;
    end
    k = k + 1;
end
```

IV. Experimental Results

TABLE I
Information of each case

Case	Tech.	Size (um*um)	# of nodes	# of cells	# of CP
Case1	Nangate45	907.63*907.2	16848	227k	51
Case2	Nangate45	971.09*970.2	18018	341k	64
Case3	Nangate45	1787.33*1786.4	33176	893k	176
Case4	TSMC18	6569.52*6569.52	132200	172k	228

The validity of our proposed algorithm to determine optimal count and locations of power pads for different cases is

TABLE II
EXPERIMENTAL RESULTS OF PROPOSED METHOD COMPARED WITH [13]

Case	[13]		Proposed Method		Runtime Speedup
	# of pads needed	Runtime (s)	# of pads needed	Runtime (s)	
Case1	9	283561.56	7	878.11	322.92x
Case2	5	223583.72	4	245.4	911.10x
Case3	10	1307907.93	10	382429.22	3.41x
Case4	12	202662.27	10	154289.21	1.31x

TABLE III
RESULTS OF STATIC IR DROP ANALYSIS PERFORMED BY VOLTUS USING CANDIDATE PADS CHOSEN BY BOTH PROPOSED ALGORITHM AND [13].

Case	Supply voltage (V)	V_IR (mV)	Worst drop of [13] (mV)	Worst drop of algo (mV)
Case1	1.1	33	25.99	27.47
Case2	1.1	33	21.59	19.25
Case3	1.1	33	32.14	30.79
Case4	1.98	59.4	55.69	51.26

presented in this section. The proposed algorithm is implemented in C++ language and compiled with g++ (GCC) 4.8.2 compiler. Our experiments are performed on an Intel Xeon E5420 machine at 2.50GHz.

To evaluate our proposed algorithm, solution quality and computational runtime are compared with SA algorithm proposed by [13]. Since the objective of the algorithm is to find optimal locations for a set of power pads only, we modified the SA algorithm to fit our problem to determine both count and locations of power pads. We use the IR drop analysis model proposed by [14] to compute voltage values at each SA iteration.

Four cases are used to demonstrate our proposed algorithm and the information of each case are shown in Table I. The core size, number of grid nodes and standard cells/macros count of all cases are shown in column 3, 4, 5 respectively. Last column shows the number of candidate pads. Table II shows the number of chosen power pads and computational runtime of our proposed algorithm compared with [13]-based method.

To validate our proposed algorithm, we performed static IR drop analysis using VOLTUS [15]. Table III shows the results of static IR drop analysis for four cases. Second column shows the supply voltages required by each case. V_{IR} is the IR drop constraint such that voltages of all grid nodes must less than V_{IR} and $V_{IR} = 0.03 * supply\ voltage$. Static IR analysis are performed on four cases with power pads selected by our proposed algorithm and [13]. The worst voltage drops for both methodologies are shown in column 4 and 5 respectively.

V. CONCLUSIONS

In this paper, we propose an algorithm that is able to determine optimal count and locations of power pads, such that the worst IR drop is less than 3% of the supply voltage. An IR drop analysis model [14] is adopted to calculate the voltages on grid nodes after new combination of power pads are selected at each iteration. We also introduced a cost function to choose candidate pads and a set of candidate pads with highest cost is selected. Our algorithm is applied on different cases to verify correctness and experimental results show that worst voltage drop of all cases are less than 3% of the supply voltage. Besides that, our algorithm is able to address optimal power pads count and locations in reasonable computational time.

REFERENCES

[1] R. Gonzalez, B. M. Gordon, and M. A. Horowitz, "Supply and threshold voltage scaling for low power cmos," *IEEE Journal of Solid-State Circuits*, vol. 32, no. 8, pp. 1210–1216, Aug 1997.

[2] X. D. S. Tan and C. J. R. Shi, "Fast power/ground network optimization based on equivalent circuit modeling," in *Proceedings of the 38th Design Automation Conference*, June 2001, pp. 550–554.

[3] C. Huang, C. Lin, W. Liao, C. Lee, H. Chen, C. Lee, and D. Kwai, "Improving power delivery network design by practical methodologies," in *IEEE 32nd International Conference on Computer Design*, Oct 2014, pp. 237–242.

[4] X. Wu, X. Hon, Y. Ca, C. K. Cheng, J. Gu, and W. Dai, "Area minimization of power distribution network using efficient nonlinear programming techniques," in *IEEE/ACM International Conference on Computer Aided Design*, Nov 2001, pp. 153–157.

[5] K. Wang and M. Marek-Sadowska, "On-chip power-supply network optimization using multigrid-based technique," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 24, no. 3, pp. 407–417, March 2005.

[6] P. Du, S.-H. Weng, X. Hu, and C. Cheng, "Power grid sizing via convex programming," in *9th IEEE International Conference on ASIC*, Oct 2011, pp. 337–340.

[7] T. Hayashi, M. Fukui, and S. Tsukiyama, "A new power grid optimization algorithm based on manufacturing cost restriction," in *European Conference on Circuit Theory and Design*, Aug 2009, pp. 703–706.

[8] M. Fukui, H. Miki, M. Yoshikawa, and S. Tsukiyama, "A power grid optimization algorithm considering via reliability," in *20th European Conference on Circuit Theory and Design*, Aug 2011, pp. 809–812.

[9] M. Zhao, Y. Fu, V. Zolotov, S. Sundareswaran, and R. Panda, "Optimal placement of power-supply pads and pins," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 25, no. 1, pp. 144–154, Jan 2006.

[10] T. Sato, H. Onodera, and M. Hashimoto, "Successive pad assignment algorithm to optimize number and location of power supply pad using incremental matrix inversion," in *Proceedings of Asia and South Pacific Design Automation Conference*, Jan 2005, pp. 723–728.

[11] K. Wang, B. H. Meyer, R. Zhang, K. Skadron, and M. Stan, "Walking pads: Fast power-supply pad-placement optimization," in *19th Asia and South Pacific Design Automation Conference*, Jan 2014, pp. 537–543.

[12] T. Yu and M. D. F. Wong, "A novel and efficient method for power pad placement optimization," in *International Symposium on Quality Electronic Design*, March 2013, pp. 158–163.

[13] Y. Zhong and M. D. F. Wong, "Fast placement optimization of power supply pads," in *Asia and South Pacific Design Automation Conference*, Jan 2007, pp. 763–767.

[14] S. Kse and E. G. Friedman, "Fast algorithms for ir voltage drop analysis exploiting locality," in *48th ACM/EDAC/IEEE Design Automation Conference*, June 2011, pp. 996–1001.

[15] "Cadence Voltus IC Power Integrity Solution," https://www.cadence.com/tw/Pages/Default.aspx.