

# TOPOLOGY OPTIMIZATION OF STRUCTURED POWER/GROUND NETWORKS

Jaskirat Singh Sachin S. Sapatnekar

Department of Electrical & Computer Engineering  
University of Minnesota  
Minneapolis, MN 55455  
{jsingh,sachin}@ece.umn.edu

## ABSTRACT

This paper presents an efficient method for optimizing the design of power/ground (P/G) networks by using locally regular, globally irregular grids. The procedure divides the power grid chip area into rectangular sub-grids or tiles. Treating the entire power grid to be composed of many tiles connected to each other enables the use of a hierarchical circuit analysis approach to identify the tiles containing the nodes having the greatest drops. Starting from an initial equal number of wires in each of the rectangular tiles, wires are added in the tiles using an iterative sensitivity based optimizer. A novel and efficient table lookup scheme is employed to provide gradient information to the optimizer. Experimental results on test circuits of practical chip sizes show that the proposed P/G network topology after optimization saves 16% to 28% of the chip wiring area over other commonly used topologies.

### Categories and Subject Descriptors

B.7.2 [Hardware]: INTEGRATED CIRCUITS-Design Aids

### General Terms

Design, Optimization

### Keywords

Power, ground, routing

## 1. INTRODUCTION

Technology scaling in VLSI chips has led to an increase in chip densities and hence, an increase in the currents drawn from the P/G networks. On the other hand, the resistances of the interconnects have also increased due to the decrease in wire widths. As a result, the IR drop in the supply grid wires has become significant and this can lead to logic failures. Even if the voltage drop in the P/G nets is not large enough to cause logic inversion, it may still affect the performance of the circuit by increasing the delays of the logic gates. Specialized techniques for the design and optimization of P/G networks have thus become essential to tackle the problem of signal integrity for the current and future VLSI chips. The key constraints in the design of power supply network are those of IR drop, electromigration and ground bounce due to inductive effects. To meet these constraints, the typical techniques available to the designers of supply networks are (i) wire sizing [1–4], (ii) adding decoupling capacitors (decaps) [5, 6], and (iii) using appropriate topologies for the P/G network [7–9].

The wire sizing and decap placement methods of [1–6] all assume that the topologies of P/G networks are fixed, and only the widths of the wire segments and the positions of decaps need to be determined. The wire sizing and decap placement solutions to

P/G network designs have a significant cost of over-utilization of the chip area. Furthermore, if the wire widths of the supply network vary throughout the chip, the routing of signal nets becomes much more difficult as a lot of book-keeping must be done to keep track of the locations and widths of P/G wires. In the works on topology optimization, the emphasis is on optimal assignment of the pins to the pads and placement of pads on the power grid. The fact that the topology has a significant influence on the final layout area is recognized, but the quest for a good topology design technique remains an open problem.

In general, it is desirable to have as much regularity as possible in the power grid in order to permit the locations of power grid wires to be easily accounted for during signal routing. A highly irregular grid (for example, one that has been sized irregularly, or one in which wires have been selectively removed) may well provide an excellent solution if the power grid design problem is viewed in isolation. However, if we consider the *entire* design flow, a high degree of irregularity can be an impediment to the design methodology as it may require a large amount of book-keeping to keep track of the precise locations of the power wires, and to determine which regions have excessive wiring congestion. Moreover, the number of optimizable parameters for such a problem can be very high, which may make the optimization highly computational. At the other extreme, a fully regular grid has few optimizable parameters and is ideal for the signal router. However, the constraint of full regularity can be overly limiting, since the requirement of regularity may cause the design to use excessive wiring resources. For instance, if all wire widths were to be required to be identical, the wires could be over-sized in regions that have relatively low current densities.

Our work uses structured regularity in the power grid, with a topology that is intermediate to fully regular grids and highly irregular grids. These grids can be thought of as being a piecewise uniform grid that is *globally irregular* and *locally regular*. This structure combines the best of both worlds: it has the advantages for routing that is afforded by fully regular nets, while offering the flexibility in optimization and better resource utilization permitted by irregular topologies.

The design of P/G network as a structured but non-uniform mesh has not been previously explored. The main features of our method are as follows:

- We present a topology optimization heuristic algorithm to design a supply grid, having a non-uniform mesh-like structure, that meets the IR drop constraints. The optimization that determines the grid structure is carried out under DC conditions using a sensitivity-based heuristic.

---

This work was supported in part by the SRC under contract 2003-TJ-1804 and the NSF under award CCR-0205227.

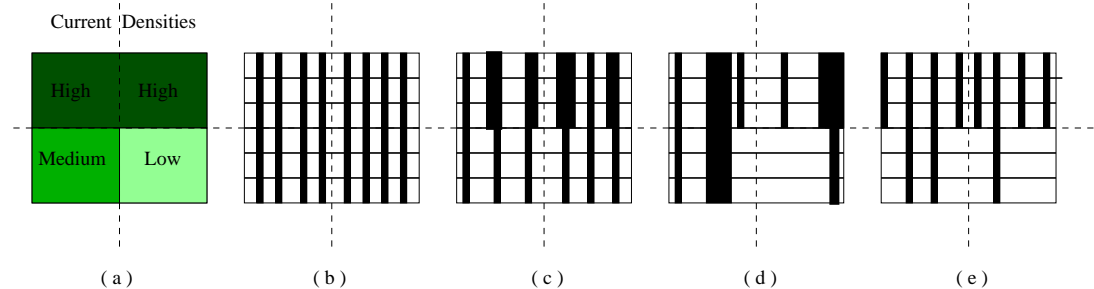


Figure 1: (a) Current densities for the chip. Possible Designs of P/G Network (b) Grid with 4 wires in each tile and regular wire sizing. (c) Grid with 3 wires in each tile and irregular wire sizing. (d) A nonuniform grid with variable pitches throughout and irregular sizing. (e) A piecewise uniform grid with 4 wires in the upper half tiles, 2 wires in bottom-left tile and 1 wire in bottom-right tile and uniform sizing.

- The macromodeling approach proposed in [10] is used to solve the circuit and determine the most critical node. We introduce reasonable approximations, specific to our scheme, that can be used to further speed up this method.
- Starting from an initial uniform grid, having the same number of wires in all the tiles in both horizontal and vertical directions, we iteratively add wires to the grid until the voltage drop constraints are met. These wires are added so as to maximize the alignments between wires in different grids.
- We guide the optimization using a novel table look-up based sensitivity calculation method to generate the gradient.

## 2. PROBLEM DESCRIPTION

Our task is to optimize the design of the proposed P/G network given the worst case currents drawn by P/G nets in different regions of the chip, the position of  $V_{dd}$  and ground pads on the chip and the voltage drop specifications.

We use a toy example to illustrate the possible design choices. For simplicity, let us assume that for the given example problem we can divide the chip into four rectangular regions or tiles having different current densities as shown in Figure 1(a).

The design in Figure 1(b) corresponds to the case where the voltage drop constraints are met by constructing a regularly structured grid with regularly sized elements with the same number of wires in the four tiles, i.e., four wires in each tile. This design uses more wire resources than required since the wire sizes and the minimum number of wires in each tile are chosen according to the region with the worst-case voltage drop. The design in Figure 1(c) meets the design constraints by employing three wires in each tile but the wires in the upper half are sized individually and irregularly to decrease the resistance and reduce the voltage drop. Such a design makes the task of the signal router more difficult, since it must keep track of the variable amount of space available in each region. The design in Figure 1(d) utilizes the lowest wiring area by using variable pitches and by sizing individual wires separately. However, besides the fact that this design would make the routing of signal nets very difficult, the optimization itself involves numerous design variables and is therefore computationally intensive. The design in Figure 1(e) is essentially the design we propose and optimize in this paper. This design is piecewise uniform as within a tile it employs a near-constant pitch and uses the same wire sizing throughout the chip. The wires are sized uniformly throughout the chip so as to maintain regularity and meet the IR drop needs. Such a design is more economical in utilization of wiring resources than designs in Figure 1(b) and (c), and has the desirable property of regularity that Figure 1(d) lacks, and does not

aggravate the congestion problem for signal nets. Moreover, due to an inherent structure in the design, it is easy to optimize.

## 3. SOLUTION TECHNIQUE

The sequence of entire optimization procedure is summarized in the following steps:

1. The P/G network is abstracted with an equivalent circuit model. Section 3.1 explains the P/G net circuit model.
2. The chip is divided into  $k$  rectangular tiles. An imaginary *skeleton grid* (to be defined in Section 3.2) is superimposed on the chip area on which the actual supply grid is built, to maintain wire alignments across tile boundaries. Starting with an equal number of wires in all tiles in both horizontal and vertical directions, an initial sparse actual grid is formed on the skeleton grid.
3. The grid is analyzed using the macromodeling technique after doing the required port approximations, as described in Section 3.3, and the most critical node  $x$  in tile  $i$ , having the maximum voltage drop from  $V_{dd}$ , is determined.
4. The voltage sensitivity of the most critical node  $x$ , with respect to increase in area in tile  $i$ , by addition of  $l$  wires, is computed using the gradient calculation method, as described in Section 3.4.
5. The number of horizontal or vertical wires in the tile, which produces the maximum voltage sensitivity is increased by  $l$ . The current sources to internal nodes of the tile are re-assigned, so that the sum of the current sources at all internal nodes is the total current drawn by the P/G buses in that tile.
6. Steps 3, 4 and 5 are repeated until the voltage of the most critical node is greater than a specified value.

### 3.1. The P/G Circuit Model

Our work is applicable in a typical design flow, after the floor-planning or placement stage, where some information about the switching patterns of various blocks is available.

Figure 2(a) shows a layout with the placed functional blocks, labeled as letters A to E. The dashed lines show the super-imposition of a skeleton grid, described in the next section, over the chip area.

Figure 2(b) describes the circuit model used for the P/G net. The entire layout is divided into a regular grid, with power grid wires, shown as the darkest wires in Figure 2(b), at the boundaries between each grid region. Each sub-grid or tile has  $m_i$  horizontal and  $n_i$  vertical wires, shown with thin dark lines.

It is possible to determine the maximum current drawn by a VLSI circuit by using the current estimation techniques such as

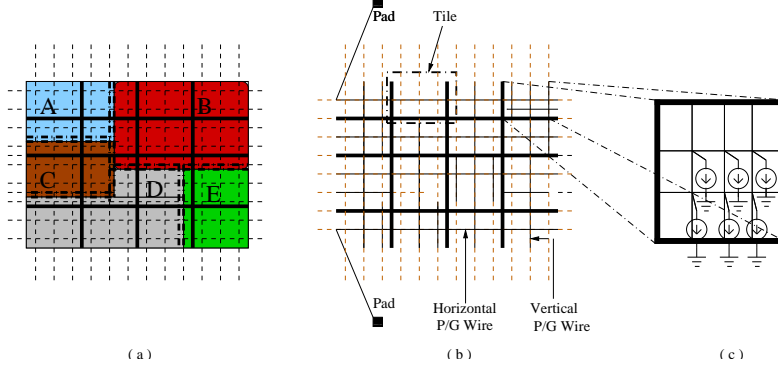


Figure 2: (a) An illustration of the layout with functional blocks A to E. The dashed lines represent the skeleton grid. (b) An abstraction of the grid into tiles with orthogonal wires. The thick dark lines represent the edge boundaries, along which our model assumes the P/G wires that are always present. The thin dark lines constitute the P/G wires inside a tile, and their density can be different in different tiles. The dashed lines represent the skeleton grid. (c) An internal view of a single tile showing the placement of current sources at internal nodes.

[11]. Given the worst case currents drawn by P/G wires in rectangular tiles, the power grid is modeled as a resistive mesh, with constant current sources to ground placed at each node inside the tiles as shown in Figure 2(c). The value of each current source within a tile is identical and is equal to the total current that flows through the P/G buses in the tile, divided by the number of nodes inside a tile. In other words :

$$c_i = \frac{I_i}{m_i n_i}$$

where  $c_i$  is the value of each current source in tile  $i$ ,  $I_i$  is the total current drawn by P/G buses of tile  $i$ ,  $m_i$  is the number of wires in the horizontal direction in tile  $i$  and  $n_i$  is the number of wires in the vertical direction in tile  $i$ .

Clean  $V_{dd}$  and ground pads are assumed to be distributed either on the periphery of the chip (two such pads are shown in Figure 2(b)) or uniformly distributed over the chip area.

### 3.2. Building the Power Grid

Our optimization procedure builds a non-uniform grid, with different densities of wires in different tiles. We use the concept of a *skeleton grid* to ensure that the wires in the adjacent tiles are not misaligned. The skeleton grid is an imaginary uniform grid superimposed over the layout area. The wires of the non-uniform real grid are placed on the skeleton grid. The pitches of the skeleton grid are chosen such that if the real grid were to completely occupy the skeleton grid, it would have the minimum number of wires to meet the voltage drop constraints. Intuitively, a skeleton grid is just a regular grid as depicted in the example of Figure 1(a).

Figure 3 depicts the actual non-uniform grid built on a uniform skeleton grid. In this example, the layout is divided into four tiles. The thick shaded lines are the power grid wires that demarcate the tile boundaries. The wires inside a tile are distributed to maintain a near-constant pitch, as per the idea of local regularity.

Our method maximizes wire alignment across tile boundaries by adding the wires on the skeleton grid in the same pre-determined order in all tiles in each iteration. For example, in Figure 3, the wires are added in the tiles in the following order: wire 1 first, and then wire 2 and wire 3. Hence wires in adjacent tiles, e.g., tile 1 and tile 2 are always aligned. Such a structure aids the routing of signal nets as the only book-keeping that is required is related to the presence or absence of wires on the skeleton grid in the given tile.

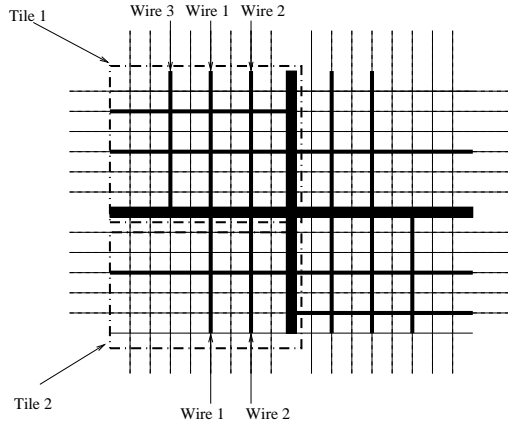


Figure 3: Illustration of a skeleton grid, shown with light lines. The P/G wires must lie on this grid, and the positions of actual P/G wires are shown with dark lines.

### 3.3. Circuit Analysis by Macromodeling Approach

#### 3.3.1. The Hierarchical Method of [10]

For the purposes of determining the most critical nodes in the circuit, our work uses the hierarchical circuit analysis technique of [10]. This section discusses the adaptation of the macromodeling method to this work.

Referring to our model of the power grid in Figure 2, the hierarchical modeling idea can be efficiently applied to our system since we have partitioned the entire power network into a set of rectangular tiles. As we are dealing with a resistive mesh model and DC excitations, each rectangular tile can be abstracted as a multiport electrical element which has a linear current-voltage relationship. These multiport elements are referred to as *macromodels*. If  $m$  is the number of wires in the horizontal direction in a tile and  $n$  is the number of vertical wires in a tile then each macromodel is a  $q$ -port linear element, where  $q = 2(m + n)$ , with the transfer characteristic given by the following equation:

$$\mathbf{I} = \mathbf{A}\mathbf{V} + \mathbf{S} \quad (1)$$

where  $\mathbf{I} \in \mathbf{R}^q$ ,  $\mathbf{A} \in \mathbf{R}^{q \times q}$ ,  $\mathbf{V} \in \mathbf{R}^q$ ,  $\mathbf{S} \in \mathbf{R}^q$ ,  $\mathbf{A}$  is the port admittance matrix,  $\mathbf{V}$  is the vector of voltages at the ports corresponding to the voltages at the nodes on edges of the tiles,  $\mathbf{I}$  is the current through the interface between the tiles, and  $\mathbf{S}$  is a vector of current sources between each port and ground.

We omit the mathematics involved in computing the macro-model parameters  $(A, S)$  as it is explained in detail in [10]. The macromodel parameters  $(A, S)$  of each tile are stamped into the MNA equation in the global system given by

$$MX = \mathbf{b} \quad (2)$$

- $M$  is the matrix containing the conductance links between global nodes and the tiles, the conductance links between the tiles and the stamps of  $A$  for each tile.
- $X$  is the vector of voltages of global nodes and ports.
- $\mathbf{b}$  is the vector of current sources at global nodes and stamps of  $S$  for each tile.

The global system corresponding to the example of Figure 3, where the chip is partitioned into four tiles, is described by:

$$M = \begin{bmatrix} G_{00} & G_{01} & G_{02} & G_{03} & G_{04} \\ G_{01}^T & A_1 & G_{12} & G_{13} & G_{14} \\ G_{02}^T & G_{12}^T & A_2 & G_{23} & G_{24} \\ G_{03}^T & G_{13}^T & G_{23}^T & A_3 & G_{34} \\ G_{04}^T & G_{14}^T & G_{24}^T & G_{34}^T & A_4 \end{bmatrix} \quad \text{and} \quad \mathbf{b} = \begin{bmatrix} \mathbf{I}_0 \\ -S_1 \\ -S_2 \\ -S_3 \\ -S_4 \end{bmatrix}$$

The entries corresponding to the index 0 are associated with the global nodes and the other indices to the tiles.

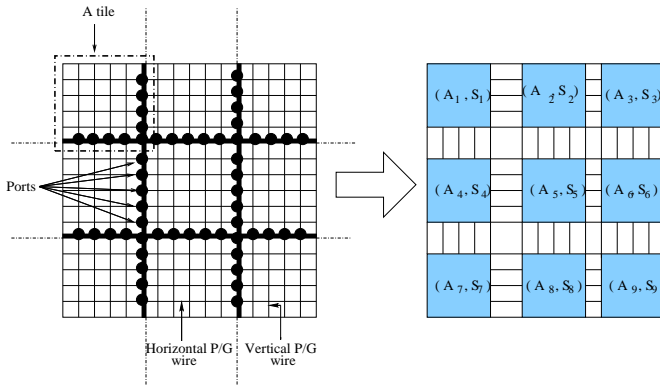


Figure 4: Converting the tiles to macromodels.

Figure 4 shows the conversion of tiles of the power grid into macromodels. In this example, the power grid which is divided into 9 tiles, is reduced to a system of nine multiport elements given by the macromodel parameters  $(A, S)$ . The macromodels are connected to each other through port connections.

For the purposes of efficiency, we also implement the step of sparsification of  $A$  matrix as proposed in [10]. This increases the sparsity of the global matrix  $M$  at the cost of reasonable simulation errors.

### 3.3.2. Port Approximation: Approach and Validation

As mentioned in Section 3.1, the nodes on the edges of the tiles are on the same conducting wire, and so it is reasonable to make an approximation that collapses some of the nodes<sup>1</sup> at the edge<sup>2</sup>. We use this observation to reduce the size of the macromodels, making an approximation to reduce the number of ports of a tile from  $2(m_i + n_i)$  to  $2(k + p)$  ports where  $k \leq m_i$  and  $p \leq n_i$ .

<sup>1</sup>This is a coarser and faster approximation than that used in multigrid-based methods.

<sup>2</sup>Even if the grid boundaries do not have these wires, the assumption is likely to be valid for reasonably dense grids.

Each of the rectangular tiles is considered as a  $2(k + p)$ -port linear element by making an approximation that the voltage variation of some nodes on the edges of the rectangular tiles is small. To further maintain the accuracy of the circuit solution, during the reduction of the edge nodes of the tiles, any  $V_{dd}$  and ground pad nodes that lie on the tile edges are always preserved and so are their immediate neighbors. Also for every removed node, its immediate neighbor is always preserved, thus approximating for small voltage variations on the edges but not for large ones. The corner nodes of the tiles are never removed. Figure 5 illustrates this process where a 20 port tile is reduced to a 14 port tile. To validate that this re-

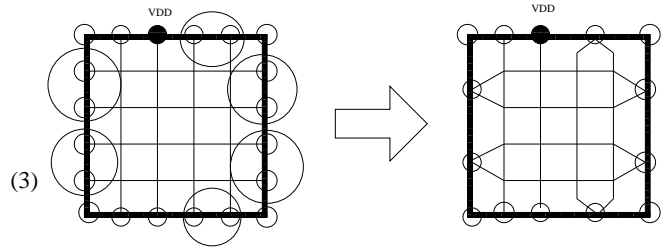


Figure 5: Reducing the number of ports of a tile.

duction from  $2(m + n)$  ports to  $2(k + p)$  ports is a reasonable approximation, we perform simulations of power grid circuits and empirically choose the value of  $k$  and  $p$  so that there is a reasonable upper bound on the error in the solution of the original and the approximated systems.

# of Ports Kept	Avg Error %	Max Err %	Runtime (sec)
4	9.21%	13.24%	0.54
8	7.52%	10.02%	0.67
10	5.89%	9.50%	0.78
12	5.38%	8.08%	0.87
20	3.19%	6.58%	1.88
24	2.60%	5.08%	2.79
28	2.41%	5.06%	3.95
32	1.57%	3.51%	5.46
36	1.27%	2.71%	7.19
40	0.07%	1.02%	9.49
44	0.00%	0.00%	11.60

Table 1: Errors for port approximations for a  $10 \times 10$  grid

Table 1 shows the the orders of errors and run time improvements for the approximations. The table shows the data for the runs for power delivery to a  $2\text{cm} \times 2\text{cm}$  chip with a  $V_{dd}$  value of 1.2 Volts and pads distributed throughout the chip. The chip is divided into  $10 \times 10$  power grid, i.e., 100 tiles with each tile having 10 horizontal and 10 vertical wires. The exact number of ports without any port approximations is 44 and this corresponds to the simulation data on the last row. The range of voltages for the exact simulation without any port approximations is 0.88V - 1.20 V.

The table shows the orders of the average and maximum errors for simulations with the port approximations as compared to a simulation without removing any ports. As seen in the table we gain significant runtime improvement from these approximations while ensuring that the errors remain within reasonable bounds. For a power grid design problem in which the bound on the worst case voltage drop is to be kept within 8%-10% of  $V_{dd}$ , the level of accuracy given by an average error of about 1%-3% is adequate.

### 3.4. Gradient Calculation

For our problem, the sensitivity calculations in matrix form provide the gradient information to guide the direction of optimization. The analysis in Section 3.3 determines the most critical node that has the greatest voltage drop. The task of sensitivity computation is to determine to which tile would the addition of a  $l$  horizontal or vertical wires would have the greatest impact on improving the worst case voltage drop. To explain the sensitivity calculation method, the following notations will be used:

- $k$  : Number of rectangular tiles into which the power grid is divided.
- $V_{xi}$  : Voltage of the most critical node  $x$  which is found in tile  $i$ .
- $p_i$  : Number of wires in tile  $i$ .
- $m_i$  : Number of wires in tile  $i$  in the horz direction.
- $n_i$  : Number of wires in tile  $i$  in the ver direction.
- $W$  : The wire width for the power grid which is assumed to be constant for the entire layout.
- $L_{hi}$  : The length of the horz wire in tile  $i$ .
- $L_{vi}$  : The length of the ver wire in tile  $i$ .
- $Ar_i$  : Wiring area used in tile  $i$ .
- $V_i$  : The vector of port voltages of tile  $i$  which contains the most critical node  $x$ .
- $t$  : Numbers of ports kept for tiles after doing the port approximations.
- $l$  : The number of wires added in any tile in each iteration.

The following relations exist between the above defined terms.

$$Ar_i = W(m_i L_{hi} + n_i L_{vi}) \quad (4)$$

$$p_i = m_i + n_i \quad (5)$$

Using the chain rule we can make the following calculations:

$$\frac{\delta \mathbf{V}_{xi}}{\delta Ar_i} = \frac{\delta \mathbf{V}_{xi}}{\delta p_i} \frac{\delta p_i}{\delta Ar_i} \quad (6)$$

From equations (4) and (5)

$$\frac{\delta Ar_i}{\delta p_i} = \frac{\delta Ar_i}{\delta m_i} + \frac{\delta Ar_i}{\delta n_i} \quad (7)$$

$$\frac{\delta Ar_i}{\delta p_i} = W(L_{hi} + L_{vi}) \quad (8)$$

To calculate the numerator of the right hand side (RHS) of (6), i.e.,  $\frac{\delta \mathbf{V}_{xi}}{\delta p_i}$ , we use the following procedure. Consider the global system,  $M\mathbf{X} = \mathbf{b}$  as described by (2). Let us assume that by the process of adding wires in tile  $i$ ,  $M$  changes to  $M + \delta M$  and  $\mathbf{b}$  changes to  $\mathbf{b} + \delta \mathbf{b}$ . It can be easily seen from equation (3) that the changes  $\delta M$  in the global matrix  $M$  are in the entries corresponding to the stamp of the first macromodel parameter  $A_i$ , and the changes  $\delta \mathbf{b}$  in the vector  $\mathbf{b}$  take place in the entries corresponding to stamps of second macromodel parameter  $\mathbf{S}_i$ . The sensitivities of particular response variables, i.e., the port voltages entries in the  $\mathbf{X}$  vector in (2), can be calculated by the following equations [12]:

$$\frac{\delta \mathbf{X}_j}{\delta \mathbf{b}_k} = \xi_{jk} \quad (9)$$

where  $\xi_j$  is the column vector corresponding to the  $j^{th}$  row of  $M^{-1}$ , and  $\xi_{jk}$  is the  $k^{th}$  component of the  $\xi_j$ , and

$$\frac{\delta \mathbf{X}_j}{\delta M_{mn}} = -\xi_{jm} \mathbf{X}_n \quad (10)$$

where  $\xi_{jm}$  is the negative of  $m^{th}$  component of column vector  $\xi_j$  multiplied by  $n^{th}$  component of the original solution vector  $\mathbf{X}$ . Using the equations (9) and (10) and applying the chain rule, we can find out the sensitivities of voltages at the ports of the tile which contains the most critical node  $x$ , with respect to an addition of  $l$  wires in tile  $i$ , by the following equation :

$$\frac{\delta \mathbf{X}_j}{\delta p_i} = \sum_{m=1}^t \sum_{n=1}^t \frac{\delta \mathbf{X}_j}{\delta M_{mn}} \frac{\delta M_{mn}}{\delta p_i} + \sum_{k=1}^t \frac{\delta \mathbf{X}_j}{\delta \mathbf{b}_k} \frac{\delta \mathbf{b}_k}{\delta p_i} \quad (11)$$

where  $t$  is the number of kept ports for the tiles after the port approximations and the summation indices in the terms  $\sum_{m=1}^t \sum_{n=1}^t$  and  $\sum_{k=1}^t$  arise due to the change in the macromodel ( $A_{t \times t}, \mathbf{S}_{t \times 1}$ ) stamps for tile  $i$  where the extra wires are added. For the purposes of efficiency, the two unknown quantities in equation (11),  $\left(\frac{\delta M_{mn}}{\delta p_i}\right)$  and  $\left(\frac{\delta \mathbf{b}_k}{\delta p_i}\right)$ , are calculated using a table lookup scheme described in the following section.

#### Table Lookup

The number of wires in a tile in horizontal and vertical directions can assume a finite set of values starting from the initial number of wires, to a maximum number that corresponds to the number of wires on the skeleton grid inside a tile.

The macromodel matrix  $A$  depends only on the number of wires in a tile and the vector  $\mathbf{S}$  depends on the number of wires and the value of current sources in a tile. However, since our model assumes equal valued current sources placed at the internal nodes of the tile as described in Section 3.1, the  $\mathbf{S}$  vector can be calculated for a current source of unit value at the internal nodes, and then a new  $\tilde{\mathbf{S}}$  vector can be computed as a scalar multiple of the  $\mathbf{S}$  vector by the following relation:

$$\tilde{\mathbf{S}} = c\mathbf{S} \quad (12)$$

where  $\tilde{\mathbf{S}}$  is the vector for the tile with a current source of magnitude  $c$  placed at the internal nodes and  $\mathbf{S}$  is the vector for the tile with a current source of unit magnitude placed at the internal nodes.

Index	# of Horizontal Wires	# of Vertical Wires	$A_{t \times t}$	$\mathbf{S}_{t \times 1}$
:	:	:	:	:
q	3	3	$A_1$	$\mathbf{S}_1$
:	:	:	:	:
p	3	5	$A_1$	$\mathbf{S}_1$
:	:	:	:	:
100	10	10	$A_{100}$	$\mathbf{S}_{100}$

Table 2: Table lookup example

Given that macromodel parameters ( $A, \mathbf{S}$ ) depend on only the number of wires in a tile, we construct, in advance, a table that contains the corresponding macromodel parameters for a set of values of horizontal and vertical wires in a tile.

The structure of the table is shown in Table 2. We can now make the following computations from the table lookup:

$$\frac{\delta M_{mn}}{\delta p_i} \approx \frac{\Delta M_{mn}}{\Delta p_i} = (A_{uv})_p - (A_{uv})_q \quad (13)$$

$$\frac{\delta \mathbf{b}_k}{\delta p_i} \approx \frac{\Delta \mathbf{b}_k}{\Delta p_i} = c((\mathbf{S}_r)_p - (\mathbf{S}_r)_q) \quad (14)$$

where  $(u, v)$  are the rows and columns of  $A_{t \times t}$  and  $(m, n)$  are the rows and columns of  $M$  corresponding to the stamp of  $A$ . Similarly  $r$  is the row of  $\mathbf{S}_{t \times 1}$  and  $k$  is the row in  $\mathbf{b}$  corresponding to the stamp of  $\mathbf{S}$ . The index  $q$  is the index in the table corresponding to

the current numbers of horizontal and vertical wires in a tile and  $p$  is the index in the table corresponding to the number of horizontal and vertical wires after an addition of  $l$  wires in either of the directions. These equations hold because any change in the  $M$  matrix and the  $\mathbf{b}$  vector due to an addition of a wire in the tile would only be in the stamps of  $A$  and  $S$ . Hence we can now easily evaluate (13) and (14).

All the terms in the RHS of (11) are now known from the solutions of Equations (9), (10), (13) and (14). The evaluation of equation (11) yields the sensitivities of the port voltages with respect to the wire addition in the tile.

We now need to calculate the sensitivity of the most critical (i.e. the left hand side(LHS) of (6)) node which could be an internal node of a tile. Since our model of the power grid as shown in Figure (2) is purely linear, we can relate the port voltages to the most critical node by the following equation:

$$\mathbf{V}_{xi} = \mathbf{C}^T \mathbf{V} + D_x \quad (15)$$

where  $\mathbf{V}$  is the vector of port voltages,  $\mathbf{C}^T$  is a row vector and  $D_x$  is a constant. Referring to [10], these terms are calculated as:  $\mathbf{C}^T = x^{th}$  row (for the internal node  $x$ ) of matrix  $-G_{11}^{-1}G_{12}$ .  $D_x = x^{th}$  component of vector  $G_{11}^{-1}\mathbf{J}_1$ .

If the most critical node is not in the same tile in which a wire is being added then the terms  $\mathbf{C}^T$  and  $D_x$  are constant. So from equation (15)

$$\frac{\delta \mathbf{V}_{xi}}{\delta p_j} = \mathbf{C}^T \frac{\delta \mathbf{V}}{\delta p_j} \quad (16)$$

If the most critical node is indeed in the same tile in which a wire is being added,

$$\frac{\delta \mathbf{V}_{xi}}{\delta p_i} = \mathbf{C}^T \frac{\delta \mathbf{V}}{\delta p_i} + \mathbf{V} \frac{\delta \mathbf{C}^T}{\delta p_i} + \frac{\delta D_x}{\delta p_i} \quad (17)$$

Thus from equations (16) and (17) we get the LHS of equation (6), which is nothing but the change in voltage of the most critical node by making an addition of  $l$  wires in tile  $i$ . This is the gradient information we require to guide the optimization process described in the next section. The table-lookup scheme is implemented as a file read and thus does not increase the memory requirements of the optimization procedure.

We empirically choose to keep the values of  $l$  to be from 5-7 wires added in a tile in every iteration. For larger values of  $l$ , the orders of errors in the sensitivity computations increase, as the assumptions made to neglect the second order variations, i.e.,  $\delta M \delta \mathbf{b}$  in the derivation of equations (9) and (10) no longer hold.

### 3.5. Optimization Heuristic

We use a greedy optimization heuristic based on the information obtained from the sensitivity computations. Using some of the definitions in the previous section, the optimization problem is formulated as follows:

$$\begin{aligned} \text{Minimize Wiring Area} &= W \sum_{i=1}^k (n_i L_{vi} + m_i L_{hi}) \\ \text{Subject to} & \\ \text{Min}(\mathbf{V}_x) &\geq V_{spec1} \text{ For Power grid} \\ \text{Max}(\mathbf{V}_x) &\leq V_{spec2} \text{ For Ground grid} \\ P_{vi} &= c_1 \lambda_v, \forall i : 1 \text{ to } k \\ P_{hi} &= c_2 \lambda_h, \forall i : 1 \text{ to } k \end{aligned}$$

where  $k$  is the number of tiles in the P/G grid,  $P_{hi}$  and  $P_{vi}$  are the wire pitches in the horizontal and vertical direction in tile  $i$ ,  $\lambda_h$  and  $\lambda_v$  are the pitches in the horizontal and vertical directions of the *skeleton grid*,  $c_1$  and  $c_2$  are some integer constants. These constraints enforce the wires in a tile to be placed on the skeleton grid with a near-constant pitch within a tile  $i$ . The wire pitches are related to the line resistances by the following relations

$$R_{vi} = \frac{\rho_s P_{hi}}{W} \quad \text{and} \quad R_{hi} = \frac{\rho_s P_{vi}}{W} \quad (18)$$

where  $R_{vi}$  and  $R_{hi}$  are the line resistances of the vertical and horizontal wires in tile  $i$ ,  $\rho_s$  is the sheet resistivity and  $W$  is the wire width which is assumed to be constant for the entire layout.

### 3.6. Speedup Techniques

In the optimization procedure we can generate the following savings in the some of the steps to achieve significant speed up.

- Instead of calculating the voltage sensitivities for all the tiles we can define an *active sensitivity window* around the most critical node so that there are enough pads within the window and then make the sensitivity calculations for only the tiles inside that window.
- By adding a few wires in only one of the tiles, we need to Cholesky-factor the  $G$  matrix for only the changed tile. The factors of other unchanged tiles are re-used.
- In any analysis step after the initial one, we do not necessarily have to solve all the tiles to determine the most critical node. After solving for the tile  $j$ , that had the worst drop, all the tiles having greater minimum voltages in the previous iteration than the minimum of voltage of tile  $j$  in the current iteration, need not be solved.

### Extension to Multiple Metal Layers

The proposed optimization technique can be easily extended to design a power grid for chips having multiple layers of metal. Typically, the upper metal layers use wider wire widths than the lower layers. Also, adding wires in upper layers would yield greater reduction in the IR drop, since the upper layer wires affect the voltages at larger number of sinks. The chip area can be divided into rectangular tiles in each of the layers. In any iteration, to determine the layer and the tile for wire additions that would result in the greatest IR drop reduction, with minimum wire area increase, we need to calculate the gradient, i.e.,  $\frac{\partial \mathbf{V}_{xi}}{\partial A r_i}$  for a few tiles in each layer. Hence, the power grid is designed by iteratively adding wires in the most sensitive tile and layer combination.

## 4. EXPERIMENTAL RESULTS

The proposed optimization scheme was implemented in C using a sparse matrix library [13], and results on several P/G networks were tested. Due to the unavailability of benchmark circuits for P/G networks, the circuits were randomly generated but with real circuit parameter values. The circuit parameter values, sheet resistivity ( $\rho_s$ ), wire width ( $W$ ) and the worst case current sources were taken from [14] and [15] for power delivery to a 2cm $\times$ 2cm chip in 130nm technology with  $V_{dd} = 1.2V$ . The voltage constraints for the power grids, i.e.,  $V_{spec1}$  was 1.08V, i.e., 90% of  $V_{dd}$ . The experiments were performed on Pentium-4 processor Linux machines with the clock speed of 2.4 GHz.

Figure 6 illustrates the non-uniform grids obtained at the end of the optimization heuristic for a 2cm $\times$ 2cm chip divided into 100 tiles. Figure 6(a) refers to the case when the  $V_{dd}$  pads are distributed at the periphery of the chip, as for a wire-bonded package.



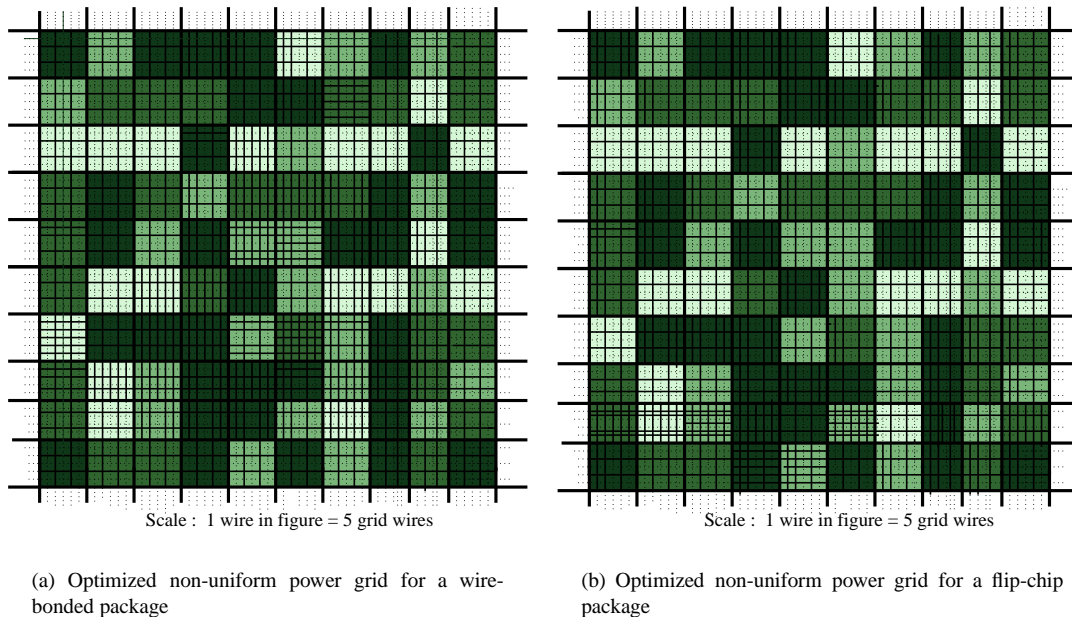


Figure 6: Power grids constructed by the proposed optimization for a chip divided into 100 tiles, superimposed over the current density patterns. The regions with darker shades have higher current densities.

Ckt	# of Tiles	$V_{init,x}$	$V_{opt,x}$	# of Ports Per Tile	# of Wires Per Iteration	Wire Area	Wire Area	% Reduction in Wire Area	Iterations	CPU Time (mins)
		(V)	(V)			Regular Grid ( $cm^2$ )	Proposed Design ( $cm^2$ )			
1	80	0.852	1.083	52	5	0.962	0.689	28.38%	201	84.4
2	100	0.847	1.086	44	5	0.840	0.677	19.41%	171	65.7
3	144	0.852	1.081	36	6	0.922	0.711	22.88%	133	54.1
4	160	0.858	1.082	32	7	0.984	0.819	16.78%	102	44.5

Table 3: Wiring area comparisons with regular grid and constant width topology.

Figure 6(b) shows the grid obtained when the pads are distributed throughout the chip, as for a flip-chip package. The grids are superimposed on the current density patterns, where darker colors represent higher current density regions. The light dashed lines represent the skeleton grid and the dark solid lines represent the actual grid. According to the scale chosen, one solid line is a substitute for five wires in the actual grids. As seen from the figure, it is not always that a higher current density tile would have a denser grid. Factors such as the pad locations also affect the density of the power grids in a given region.

Two sets of experiments were conducted that intended to compare our approach with :

1. The wire area utilized by a uniformly structured grid and constant wire width throughout the chip. Table 3 refers to this comparison.
2. The wire area utilized by a uniform grid for which wires are sized differently in different regions of the chip. Table 4 refers to this comparison. The number of wires for the uniform grids of Table 4 is less than that of the uniform grids of Table 3, but the wire widths are greater for wires in some of the high current density tiles of the uniform grids of Table 4.

For the power grids of Table 3 and Table 4, the  $V_{dd}$  pads were distributed throughout the chip. As listed in Table 3, the number of tiles into which the chip was divided for optimization are given in

the second column. The third column ( $V_{init,x}$ ) refers to the voltage of the most critical node before the optimization, when starting from an initial grid with equal number of wires in all tiles. Column four ( $V_{opt,x}$ ) indicates the voltage at the end of the optimization process. This voltage is measured using an exact simulation of the power grid circuit, i.e., without the approximation of reducing the number of ports and sparsification of  $A$  matrix. The optimization is run for some more iterations even after meeting the voltage drop constraint, so that the errors introduced by the port approximations and sparsification can be accounted for by a bit of over-design. The number of ports retained for each tile after the process of *Port Approximation* is shown in the fifth column. Column six shows the number of wires that were added in every iteration in the most sensitive tile to meet the voltage drop constraints. The seventh column shows the amount of wiring area that would be utilized by a P/G network topology having a constant pitch of wires throughout the chip and constant wire width. By enumeration, a minimum number of wires that satisfy the voltage drop constraint, was chosen to construct this regular grid. The eighth column (Wire Area Proposed Design) shows the amount of wire area used by the proposed design at the end of the optimization. The wire widths used by the proposed optimization is the same as the one used for the regular grid for a fair comparison. As seen from the table, there is a saving of about 16% to 28% in wire area by the proposed optimization scheme over the topology having a regular grid and constant sizing

with same number of wires in all tiles.

Table 4 shows the comparison of wire area of the proposed P/G network structure with a design that employs wire width sizing, starting from an initial uniform grid with same number of wires in all tiles and constant wire widths. It should be emphasized that the number of wires in the tiles for the regular grid of Table 4 is less than that (about 0.6-0.7 $\times$ ) of the regular grid design that is listed in column seven of Table 3 and the minimum wire width of the regular grid design of Table 4 is the same as that of the proposed design.

Ckt	# of Tiles	# of Tiles Sized	Wire Area with Sizing ( $cm^2$ )	Wire Area Proposed Design ( $cm^2$ )	% Reduction in Wire Area
1	80	32	0.843	0.689	18.26%
2	100	38	0.883	0.677	23.38%
3	144	48	0.894	0.711	20.46%
4	160	62	1.096	0.819	25.27%

Table 4: Wiring area comparisons with the wire sizing method.

The third column in Table 4 refers to the number of tiles in which the wire widths were incrementally sized. The tiles with high current densities are identified and the wire widths in those tiles are incrementally sized until the voltage drop constraints are met. We compare the sizing solution against the four optimized power networks of Table 3. The fourth column (Wiring Area with Sizing) lists the wire area used of various chips by the wire sizing solution. The fifth column (Wire Area Proposed Design) lists the wire area used by the proposed optimization. The last column indicates the percentage change in the wire area. There is a reduction of 18% to 25% in wire area by the proposed optimization scheme over the design with the wire width sizing technique.

The runtime of the proposed design of the power grid is a function of the number of wires added in each iteration step and the number of ports that are removed to make the macromodels. To better understand the runtime, accuracy and area-reduction trade-offs, we perform a series of experiments in which we construct a grid using the proposed optimization scheme for power delivery to 2cm $\times$ 2cm chip divided into 100 tiles. The runtime, accuracy and wire area reduction trade-off is explored by varying the number of ports kept for the tiles and the number of wires added per iteration.

Design	# of Ports	# Wires per Iteration	% Reduction in Wire Area	Avg Err in Opt Design	CPU time (mins)
1	44	5	19.41%	2.14%	63.4
2	44	6	16.23%	2.32%	60.6
3	44	7	13.34%	2.64%	58.9
4	40	5	19.09%	3.67%	56.2
5	40	6	16.50%	3.95%	54.5
6	40	7	14.41%	4.23%	53.4
7	36	5	19.52%	5.81%	41.6
8	36	6	17.12%	6.04%	40.4
9	36	7	13.21%	6.23%	39.2
10	28	5	18.98%	9.02%	19.7
11	28	6	16.47%	9.55%	18.6
12	28	7	13.59%	10.17%	17.1

Table 5: Various trade-offs in power grid design

As shown by Table 5, on one hand, the runtime significantly reduces by removing more ports while forming the macromodels, but on the other hand, the errors in measurement of voltages increase. If we add more wires per iteration for the same number

of kept ports, the runtime slightly improves but at the cost of accuracy and savings in the wire area. By adding more wires per iteration, we are increasing the amount of over design and hence reducing the improvement in area reduction over other topologies. Thus according to the level of accuracy and the order of runtime required, we can choose the appropriate parameters to guide the optimization.

## 5. CONCLUSION

In this paper, we have proposed a new optimization scheme for the design of structured P/G networks that are locally regular and global irregular. Our approach maximizes alignment between wires in each tile using the idea of the skeleton grid. Experimental results on randomly generated P/G networks of practical chip size and circuit parameters show significant reduction in the wire area used by the proposed scheme when compared to the area consumed by the wire sizing solution and the uniform grid topologies. This design also aids in an easier routing scheme for the signal nets later in the design as minimal book-keeping needs to be done for the proposed P/G architecture. Including a simulator in the optimization loop ensures the accuracy of the optimized solution but our results show that the run times are reasonable.

## 6. REFERENCES

- [1] X. D. S. Tan *et al*, "Reliability-Constrained Area Optimization of VLSI Power/Ground Networks Via Sequence of Linear Programming," in *Proc. ACM/IEEE DAC*, pp. 156–161, 1999.
- [2] X. D. S. Tan and C. J. R. Shi, "Fast Power/Ground Network Optimization Based on Equivalent Circuit Modeling," in *Proc. ACM/IEEE DAC*, pp. 550–554, 2001.
- [3] X. Wu *et al*, "Area Minimization of Power Distribution Network Using Efficient Nonlinear Programming Techniques," in *Proc. IEEE/ACM ICCAD*, pp. 153–157, 2001.
- [4] T. Wang and C. C. Chen, "Optimization of the Power/Ground Network Wire-Sizing and Spacing Based on Sequential Network Simplex Algorithm," in *Proc. ISQED*, pp. 157–162, 2002.
- [5] H. Su, K. H. Gala, and S. S. Sapatnekar, "Fast Analysis and Optimization of Power/Ground Networks," in *Proc. IEEE/ACM ICCAD*, pp. 477–480, 2000.
- [6] H. H. Chen and D. D. Ling, "Power Supply Noise Analysis Methodology for Deep-Sub-micron VLSI Chip Design," in *Proc. ACM/IEEE DAC*, pp. 638–643, 1997.
- [7] T. Mitsuhashi and E. S. Kuh, "Power and Ground Network Topology Optimization for Cell Based VLSIs," in *Proc. ACM/IEEE DAC*, pp. 524–529, 1992.
- [8] H. Cai, "Multi-pads Single Layer Power Net Routing in VLSI Circuit," in *Proc. ACM/IEEE DAC*, pp. 183–188, 1988.
- [9] J. Oh and M. Pedram, "Multi-pad Power/Ground Network Design for Uniform Distribution of Ground Bounce," in *Proc. ACM/IEEE DAC*, pp. 287–290, 1998.
- [10] M. Zhao *et al*, "Hierarchical Analysis of Power Distribution Networks," in *Proc. ACM/IEEE DAC*, pp. 482–486, 2000.
- [11] H. Kriplani, F. Najm, and I. Hajj, "Pattern Independent Maximum Current Estimation in Power and Ground Buses of CMOS VLSI Circuits : Algorithms, Signal Correlations, and Their Resolution," in *Proc. IEEE/ACM ICCAD*, pp. 998–1012, 1995.
- [12] L. T. Pillage, R. A. Rohrer, and C. Visweswariah, *Electronic Circuit and System Simulation Methods*. McGraw-Hill, New York, NY, 1995.
- [13] <http://www.netlib.org/c/meschach>.
- [14] Semiconductor Industry Association, "International Technology Roadmap for Semiconductors," 2001. Available at <http://public.itrs.net>.
- [15] J. Cong, "An Interconnect-Centric Design Flow for Nanometer Technologies," in *Proc. IEEE*, vol. 89, pp. 477–480, 2000.