

Effective Power Network Prototyping via Statistical-Based Clustering and Sequential Linear Programming

Sean Shih-Ying Liu*, Chieh-Jui Lee*, Chuan-Chia Huang*, Hung-Ming Chen*, Chang-Tzu Lin[†] and Chia-Hsin Lee[†]

*Institute of Electronics, National Chiao Tung University, Hsinchu, Taiwan

[†]Information and Communication Research Laboratory, Industrial Technology Research Institute, Taiwan

Email: {sniealu,jerrylee2929.ee92,chuan.ee96}@gmail.com, hmchen@mail.nctu.edu.tw, {ctzulin,JerryLee}@itri.org.tw

Abstract—In this paper, we propose a framework that automatically generates a power network based on given placed design and verifies the power network by the commercial tool without IR and Electro-Migration (EM) violations. Our framework integrates synthesis, optimization and analysis of power network. A deterministic method is proposed to decide number and location of power stripes based on clustering analysis. After an initial power network is synthesized, we propose a sensitivity matrix G_s which is the correlation between updates in stripe resistance and nodal voltage. An optimization scheme based on Sequential Linear Programming (SLP) is applied to iteratively adjust power network to satisfy a given IR drop constraint. The proposed framework constantly updates voltage distribution in response to incremental change in power network. To accurately capture voltage distribution on a given chip, our power network models every existing power stripes and via resistances on each layer. Experimental result demonstrates that our power network analysis can accurately capture voltage distribution on a given chip and effectively minimize power network area. The proposed methodology is experimented on two real designs in TSMC 90nm and UMC 90nm technology respectively and achieves 9%-32% reduction in power network area, compared with the results from modern commercial PG synthesizer.

I. INTRODUCTION

The challenge in power network optimization generally involves on constructing a power network that satisfies several given constraints using minimal amount of resource. Cell gains its driving power from power pads via power grid. During the transmission, partial current is drawn off due to wire resistance which is commonly known as IR drop effect.

IR drop effect is a serious issue in nanometer design. As technology advances, less voltage is required to drive a single cell. However, driving voltage transmitted to a cell may not be sufficient due to unacceptable voltage drop. Insufficient driving voltage will deteriorate signal integrity and increase noise susceptibility of a given design. In addition, gate delay increases non-linearly as driving voltage at gate decreases which may eventually lead to a setup/hold time violation.

A. Prior Works on Power Network

In terms of power network synthesis, Chen *et. al.* [1] proposed a two level mesh structure composed of a coarser mesh and a finer mesh. A closed form representation is proposed to estimate worst case IR drop effect. In [1], cell's power consumption is neglected and power grid is modeled as infinite resistive grid.

Regarding to power network optimization, Tan *et. al.* [2] solves the problem by formulating it as sequence of linear programs. Wu *et. al.* [3] determines stripe width by applying nonlinear programming technique with penalty function. Su *et. al.* [4] incorporates decoupling capacitors into power network optimization and optimizes the power network using iterative method. Considering the variation in current, Bord *et. al.* [5] proposed a non-uniform power mesh based on minimizing a linear combination of total average power and total wire area. Wang *et. al.* [6] presents the sequential network simplex algorithm which can optimize both the width and length very efficiency.

Prior to 2011 TAU Contest [7], many previous works assume a simplified resistive model since a complete nodal analysis on entire power network is computationally expensive to be invoked. However, recent progress in [8–10] demonstrates nodal analysis is capable to handle very complex designs.

B. Our Contribution

While prior works have extensively studied techniques on power network synthesis, optimization and analysis, there still exist room for improvement on communication between power network analysis and optimization. In standard design flow, power network is generally constructed in a mesh-based fashion. Optimization of power network can be applied on adjusting width for each individual segments. However, resizing power segments poses significant difficulty on satisfying design rules and manufacturability. A common approach in industry on optimizing power network is to decide number of power stripes and determine appropriate width for power stripes. There are design rules specified on spacing and dimension of power stripes. Power stripes can not be arbitrary resized and there exists limitation on placement of stripes.

To this end, we seek for an automated framework that analyzes given placement and then simultaneously synthesizes the power network. Our framework considers practical design constraints which limits physical dimension on power stripe and rings. Our framework is validated in a industrial physical synthesis flow. Empirical result reveals the effectiveness our proposed methodology.

To satisfy a given IR drop constraint while minimizing power network area, we propose a framework that iteratively adjusts the given power network. In the proposed framework, a sensitivity matrix G_s is reconstructed in each iteration to identify region suffers most severe IR-drop in the power network and then incrementally adjusts the power stripes. In brief, our main contributions are summarized as follow.

- 1) A statistical method based on clustering analysis to determine an initial power network.
- 2) An automated framework that integrates a preconditioned conjugate gradient (PCG) power analysis with optimization method based on sequential linear programming (SLP) to refine a given power network.
- 3) The proposed framework is validated on two industrial designs and empirical result shows that our proposed method can achieve 9%-32% power network area reduction compare to conventional power network synthesis.

Optimization to power network is only effective if both power network model and analysis are accurate. To accurately analyze a given design, power consumption of every cells and resistance of power stripes, via and power rings are taken into consideration.

C. Overview of the Proposed Framework

Fig. 1 is a flow chart of the proposed algorithm. Designs are exported to our database using OpenAccess [11]. An initial power

This work is partially supported by the National Science Council of Taiwan under grant No. NSC 101-2220-E-009-045
978-3-9815370-0-0/DATE13/©2013 EDAA

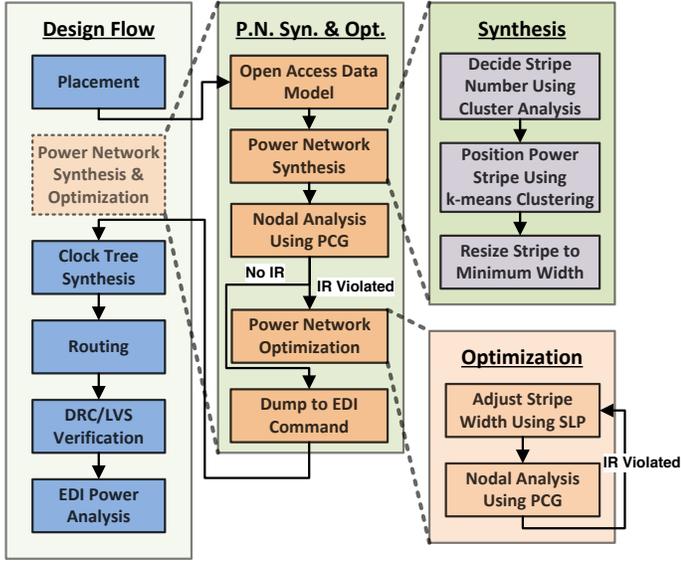


Fig. 1. Flow chart of the proposed power network optimization. Design is exported using OpenAccess [11] after placement. Our algorithm synthesizes and optimizes the exported design. Result is fed back to design flow by dumping our result in EDI command.

network is generated based on statistical method. Analysis and optimization are then iteratively performed on the synthesized power network. The topology of power network is exported to Encounter Digital Implementation (EDI) [12] commands. IR drop of the power network is analyzed using commercial power analysis tool.

The organization of the remaining paper is as follows. Section II presents a power network model in which a graph representation can be derived. Section III demonstrates [13] how number and position of power stripes are determined using clustering analysis based on Gap Statistic [14] and k-means [15] clustering. After power network is synthesized, Section IV presents an optimization methodology using Sequential Linear Programming which is tightly integrated to nodal analysis. Finally, Section V presents the experimental result and Section VI concludes the paper.

II. REPRESENTATION OF POWER MESH NETWORK

In this section, a three dimensional mesh structure is presented to model current flow and resistive model for a given power network. In a typical standard cell design, the given layout is divided into rows with row height equivalent to the height of a standard cell. Pre-planned power/ground stripes that runs horizontally are placed at top and bottom of the row. The pre-planned power/ground stripes are connected to main power branch which are connected to the power/ground pads. Thus, when a cell is placed in one of the rows, it is automatically connected to chip's power network.

The main power branch is vertical and horizontal power stripes but much wider in size. The main power branch serves to transmit power from power pad to pre-planned power/ground stripes. The topology of main power branch generally relies on designers to determine number and position of power stripes. Major EDA tools provide power analysis tool at certain stage of design. Designers will need to rely on the power analysis tool to verify and reconfigure their power network. This process generally involves several repetitions of try and error.

In this regard, to obtain a more accurate power analysis, a three dimensional mesh structure is used to represent power network for a given design. Fig. 2(a) depicts how voltage is transmitted to drive a cell. In Fig. 2(a), driving voltage travels from power pad and descends

through all metal layers to the standard cell then travels back to ground pad. Regarding to previous works which primarily focus on IR-drop analysis from power pad to standard cells, a three dimensional representation can accurately model both paths travelling from power pad to standard cell and standard cell to ground pad.

A graph representation can be derived as illustrated in Fig. 2(b). In graph representation of power mesh, each node have a coordinate corresponding to a via contact or a current source. Each node will also have 6 connected edges. Each edge has a resistance value and a current value. Power consumption of standard cells is modelled as a current source $I_i = P_i/V_{DD}$. Edge resistance is the wire resistance connecting between two nodes $r_j = \rho \cdot l_j/w_j$. Via resistance $r_k = \frac{R_{via}}{\text{num. via}}$ is calculated by dividing resistance of one via by the cross section area connecting two metal layers.

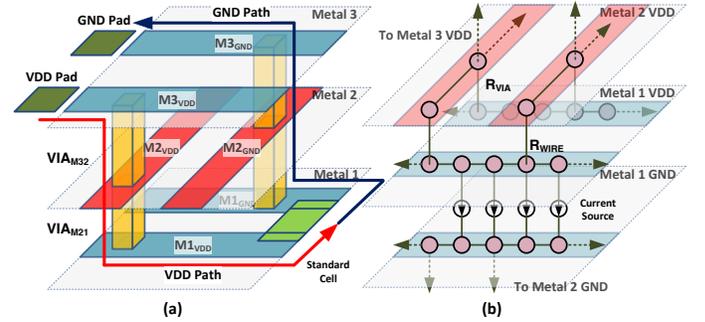


Fig. 2. Structural representation of 3D power mesh network. (a) Pre-planned power stripes lies in Metal 1 layer. Power branches planned by designers are placed in Metal 2 layer and above. Driving voltage is fed from power pad and then travels down to Metal 1 layer to drive the standard cells and then travels back to ground pad. (b) In graph representation, via contacts and standard cells are represented as nodes. Power consumption of a cell is modeled as a current source. Wire connecting between two nodes is modeled as edge resistance.

III. POWER NETWORK SYNTHESIS USING NON-UNIFORM POWER MESH NETWORK

Conventional power network is generally a mesh based network. IR drop is alleviated by increasing number of power stripes in the power mesh network until IR drop constraint is satisfied. However, such approach is likely to add redundant power stripes in regions where IR drop is not violated. Redundant power stripes creates extra routing blockages during routing stage.

In this work, construction of the initial power network is divided to two stages. First stage determines optimal location to place given n number of power stripes. Second subsection determines appropriate number of power stripes. A deterministic method based on Gap Statistics [14] is adopted to determine number of stripes. After the number of stripes is determined, k-means clustering [15] is applied to place power stripes where worst IR drop can be effectively minimized. To increase quality of k-means clustering, initial positions of power stripes are determined based on careful seeding [16].

A. Finding Location of Stripe Using k-means Clustering

In this subsection, the concept of clustering is applied in synthesizing power network. Region with higher density of cells is likely to have higher power consumption. Thus, to maximize effectiveness of a power stripe, power stripe should be placed in position where IR drop can be minimized. Fig. 3 is an illustration that depicts similarity between clustering and determining position of power stripes.

Since driving voltage needs to travel through pre-planned power stripes to reach a cell, distance from a cell to its nearest power branch is directly proportional to the IR drop of a cell. The main power stripes

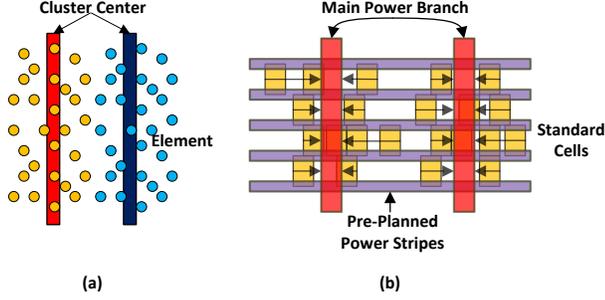


Fig. 3. Using k-means clustering to determine position of power stripes.

Algorithm 1 Determine Location of Power Stripes

- 1: Randomly choose one cell x_i from X
 - 2: Add x_i to C
 - 3: **for** 1 to $k - 1$ **do**
 - 4: **for** each cell $x_i \in X$ **do**
 - 5: calculate $D(x_i)$
 - 6: **end for**
 - 7: Randomly choose y between 0 and $\sum_{i \in n} D(x_i)^2$
 - 8: Find i such that $\sum_{j=0}^i D(x_j)^2 \geq y \geq \sum_{j=0}^{i-1} D(x_{j-1})^2$
 - 9: Add x_i to C
 - 10: **end for**
 - 11: **repeat**
 - 12: Assign cell x_i to nearest power stripe c_r
 - 13: Place c_r in mean position of cells $x_i \in C_r$
 - 14: **until** Position of all stripes $c_r \in C$ is stable
-

needs to place in position where the summation of distance between cells to its nearest power branch is minimized. This can be easily regarded as a cluster problem where a single power stripe represents one cluster and a cell represents an element.

The problem of determining location of power stripes is defined as follows. Given a set of cells $X = \{x_1 \dots x_n\}$ and k , determine the position of k power stripes $C = \{c_1 \dots c_k\}$ where μ_r denotes mean point of c_r such that within cluster sum of square ϕ defined in Eq. (1) is minimized.

$$\min \phi = \sum_{r=1}^k \sum_{x_i \in C_r} \|x_i - \mu_r\|^2 \quad (1)$$

In k-means clustering, the quality of clusters is susceptible to selection of initial centers. It is desirable to choose initial centers that are evenly separated from each other. Thus, method proposed in [16] is applied to carefully select centers at initial phase.

Alg. 1 describes the procedure of k-means clustering. In Alg. 1, $D(x_i)$ denotes the shortest distance from a cell x_i to the nearest power stripe c_r . At initial phase, one cell is randomly selected as the position for the first power stripe. Then the position for rest of the $k - 1$ stripes are selected based on probability proportional to $D(x_i)^2$. Each time a stripe is placed, $D(x_i)$ will be updated for all cells x_i . After location of initial k power stripes are selected, the rest of the algorithm follows the concept of k-means clustering [15]. Each iteration of k-means clustering involves two procedures. First, cells are assigned to nearest power stripe. Then, power stripe is moved to the mean location of the cells. Iteration terminates when position of power stripe is unchanged compared to previous round.

B. Determining Number of Stripes Using Gap Statistics

In this subsection, a deterministic method based on Gap Statistic [14] is proposed to decide an appropriate number of power stripes.

Algorithm 2 Determine Number of Stripes

- 1: $k = 0$
 - 2: **repeat**
 - 3: $k = k + 1$
 - 4: Call Alg. 1 to determine position for k stripes
 - 5: $D_r = \sum_{i, i' \in C_r} (x_i - x_{i'})^2$
 - 6: $W_k = \sum_{r=1}^k \frac{1}{2n_r} D_r$
 - 7: $Gap(k) = E^*(\log(W_k)) - \log(W_k)$
 - 8: **until** $Gap(k) \geq Gap(k - 1) - s_k$
 - 9: **return** $k - 1$
-

In cluster analysis, error measure W_k is a measure of dispersion for k number of clusters. As k increases, W_k decreases monotonically. However, at certain k , W_k decreases rapidly and flattens onward. The location of such rapid decrease in W_k is believed to provide an optimal number of k .

In Eq. (2), the measure in magnitude of decrease in W_k denoted by $Gap(k)$ is defined as the difference between $\log(W_k)$ and its expected value. In Eq. (3), the expected value of $\log(W_k)$ is obtained by taking B Monte Carlo samples of $\log(W_k)$. Smallest value of k can be obtained when $Gap(k)$ is less than $Gap(k - 1)$.

Alg. 2 describes the procedure to decide k . D_r in line 5 measures the dispersion for a cluster C_r . W_k in line 6 measures the dispersion for all clusters which is the summation of all D_r divided by $2n_r$. n_r denotes the element size in C_r . Stripe number k is determined by measuring $Gap(k)$ in each iteration and terminates when $Gap(k) \leq Gap(k - 1)$.

$$Gap(k) = E^*(\log(W_k)) - \log(W_k) \quad (2)$$

$$E^*(\log(W_k)) = 1/B \sum_b^B \log(W_{kb}^*) \quad (3)$$

After number and position of power stripes are determined, the width of every stripe is sized to minimum allowed width. Nodal analysis on power network is initiated to check if IR drop constraint is violated. Process terminates if there is no IR drop observed for any nodes. However, if IR drop is observed for certain nodes that violates given IR drop constraint, optimization technique in Section IV is initiated to optimize the constructed power network.

IV. POWER NETWORK OPTIMIZATION USING SEQUENTIAL LINEAR PROGRAMMING

In this section, we present an optimization methodology that incrementally adjusts stripe width until voltage of every node is above a given threshold. Our optimization is based on sequential linear programming (SLP). A sensitivity matrix ($G_s = \partial V / \partial R$) is reconstructed in each iteration. The organization of this section is organized as follows. First, the problem of power network optimization is defined. Then, the concept of sensitivity matrix is introduced. Final part of this section describes how original problem is modified such that SLP can be applied.

A. Formulating Power Network Optimization Problem

The initial power network consists of horizontal and vertical stripes. Each stripe consists a set of nodes representing a set of points on that stripe. Two adjacent nodes on a stripe are connected by an edge. When a stripe increase its width, all edge resistance on that stripe decreases which results in increase on all nodal voltage on that stripe.

Hence, determining the width of power stripes such that minimum voltage on the power network is above the IR-drop constraint is a

TABLE I. VARIABLES USED IN SEQUENTIAL LINEAR PROGRAMMING

Variable	Description
V_{min}	minimum acceptable voltage
R_k	resistance for stripe k
L_k	length for stripe k
W_k	width for stripe k
G_s	sensitivity matrix ($(\partial V / \partial R)$)
∂R_k	change in resistance for stripe k
∂V_i	change in voltage for node i
e_{jk}	edge j on stripe k
r_{jk}	resistance for edge j on stripe k
l_{jk}	length for edge j on stripe k
w_{jk}	width for edge j on stripe k

power network optimization problem. The problem of power network optimization is defined as follows.

The Power Network Optimization Problem. *Given a nodal graph that represents the initial power network and a set of nodes with voltage less than V_{min} , use minimal power network area by adjusting stripe width on power network such that voltage on every node is above V_{min} .*

Table I lists the notations used in this section. The mathematical expression of power network optimization problem is defined in Eq. (4). Here, we define the set of horizontal stripes as $K_x = \{K_{x1}, K_{x2}, \dots, K_{xn}\}$ and the set of vertical stripes as $K_y = \{K_{y1}, K_{y2}, \dots, K_{ym}\}$. The area for a given stripe is width of stripe (W_k) multiplied by the length of stripe (L_k). Stripe width is bounded within a max. W_{max} and min. W_{min} constraint.

$$\begin{aligned}
 \min. \quad & \sum_{k \in \{K_x, K_y\}} L_k \cdot W_k \\
 \text{s. t.} \quad & V \geq V_{min} \\
 & W_{min} \leq W_k = \frac{\rho L_k}{R_k} \leq W_{max}
 \end{aligned} \tag{4}$$

According to Kirchoff Current Law (KCL) and Kirchoff Voltage Law (KVL), the voltage drop in a power mesh network from any voltage source V_x to any node n_i is equivalent for all paths $p_{xi} \in P$. In other words, the voltage drop for a node n_i is contributed by every edge resistance in power network.

B. Construction of Sensitivity Matrix

Since the voltage value for a given node is affected by every edge resistance. A sensitivity matrix G_s is constructed to measure how voltage value for a node is affected by every edge resistances in power network. Eq. (5) is a single row on matrix G_s which represents how voltage of a single node is affected every other edge resistance.

$$g_i = \{\partial v_i / \partial r_1, \partial v_i / \partial r_2, \dots, \partial v_i / \partial r_m\}. \tag{5}$$

Using Eq. (5), the change to voltage on node n_i in response to slightest change in resistances can be described using Eq. (6).

$$v'_i = \partial v_i + v_i = g_i \partial r + v_i \tag{6}$$

Solving Eq. (6) is computationally un-affordable. Thus, we limit the scope to only consider edges that have the most effect on a particular node n_i . We assume the edges that are most influential to a given node n_i lie on shortest path from node n_i to nearest voltage source.

Minimize power network area by optimizing individual edge resistance based on Eq. (6) is impractical. Since when a power stripe increases its width, every edges on that stripe are adjusted with same proportion simultaneously. Thus, scope of optimization should focus on stripe level rather than edge level.

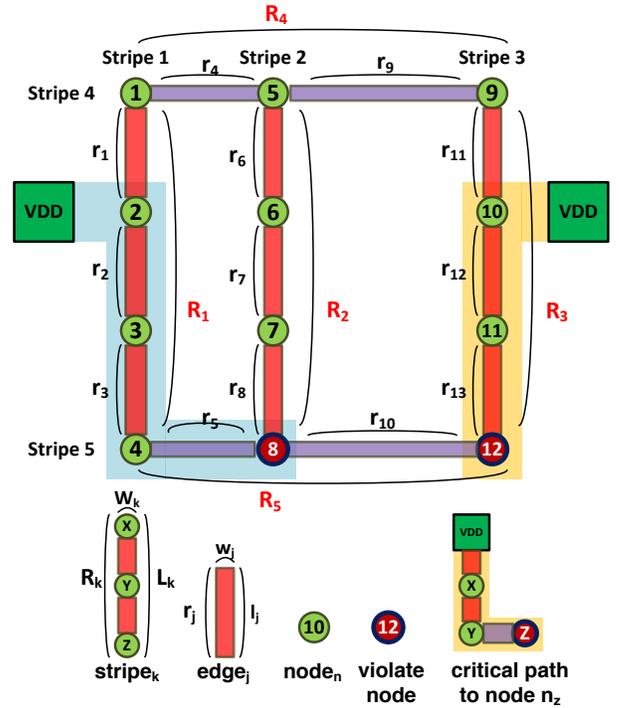


Fig. 4. A two dimensional graph representation of power mesh network. Node n_8 and n_{12} are nodes with voltage below given threshold V_{min} . Paths highlighted in light blue and light orange are the critical paths violating nodes to nearest voltage pad. $v_8 = i_2 r_2 + i_3 r_3 + i_5 r_5 = i_2 \frac{l_2}{L_1} R_1 + i_3 \frac{l_3}{L_1} R_1 + i_5 \frac{l_5}{L_5} R_5$. $v_{12} = i_{12} r_{12} + i_{13} r_{13} = i_{12} \frac{l_{12}}{L_3} R_3 + i_{13} \frac{l_{13}}{L_3} R_3$. Change in nodal voltage can be described in terms of change in stripe resistance. $\partial v_8 = \frac{i_2 l_2 + i_3 l_3}{L_1} \partial R_1 + \frac{i_5 l_5}{L_5} \partial R_5$, $\partial v_{12} = \frac{i_{12} l_{12} + i_{13} l_{13}}{L_3} \partial R_3$.

In this regard, we model the change in stripe resistance as combination of change in resistance on all edges on that stripe. Here r_{jk} denotes edge resistance r_j on stripe k . Resistance value r_{jk} can be represented by stripe resistance R_k using Eq. (7).

$$\partial r_{jk} = \partial (l_{jk} / L_k) R_k \quad j \in k \tag{7}$$

Fig. 4 is a simple example on adjusting wire width to minimize IR drop. Node n_8 is node with voltage below given threshold V_{min} . $v_8 = i_2 r_2 + i_3 r_3 + i_5 r_5 = i_2 \frac{l_2}{L_1} R_1 + i_3 \frac{l_3}{L_1} R_1 + i_5 \frac{l_5}{L_5} R_5$. Eq. (8) is the sensitivity matrix for g_8 and Eq. (9) is the difference for power stripe.

$$g_8 = \left\{ \frac{i_2 l_2 + i_3 l_3}{L_1}, 0, 0, 0, \frac{i_5 l_5}{L_5} \right\} \tag{8}$$

$$\partial R = \{\partial R_1, \partial R_2, \partial R_3, \partial R_4, \partial R_5\} \tag{9}$$

C. Optimization Using Sequential Linear Programming

After one round of nodal analysis, nodes with voltage value less than V_{min} can be identified. Here we define $Z = \{v_1, v_2, \dots, v_z\}$ as the set of nodal voltages that violates voltage constraint. The corresponding sensitivity matrix to Z is $G_s = \{g_1, g_2, \dots, g_z\}$. Note that each element in sensitivity matrix g_i is essentially current value on edge e_j . Any change in resistance value will change every value in G_s . Thus, sensitivity matrix needs to be reconstructed every time after an adjustment to stripe width is made. After sensitivity matrix G_s is constructed, relation between change in stripe resistance to change

in nodal voltage can be obtained in Eq. (10). Optimization can then focus on determining appropriate ∂R value for each stripe to increase nodal voltage for violating nodes.

$$Z' = G_s \partial R + Z \quad (10)$$

To determine the ∂R value to minimize the difference between voltage for violating nodes and V_{min} . We formulate the optimization problem in Eq. (11).

$$\min. \sum_{\forall i \in \{Z\}} V_{min} - v_i \quad (11)$$

Optimizing Eq. (11) will greedily use as much power stripe area as possible to optimize given objective since power stripe area is not concerned in primary objective. Thus, increase in power network area should also be considered in primary objective.

However, since ∂R is the only variable in Eq. (10), change in width of power stripe W_k must translate in terms of change in stripe resistance, ∂R_k . Given that length of power stripe L_k is fixed, Eq. (12) derives the relation between ∂W_k to ∂R_k .

$$\begin{aligned} \partial W_k &= W'_k - W_k = \frac{\rho L_k}{R_k - \partial R_k} - \frac{\rho L_k}{R_k} \\ &= \rho L_k \frac{\partial R_k}{(R_k - \partial R_k) R_k} \simeq -\frac{\rho L_k}{R_k^2} \partial R_k \end{aligned} \quad (12)$$

Power stripe area can be considered into primary objective using Eq. (12). Original optimization problem defined in Eq. (4) can then be formulated as Eq. (14). In each iteration of solving Eq. (12), change in stripe's resistance ∂R is determined and width power stripes are adjusted according to Eq. (13). β is a penalty function that penalizes increase in power stripe area. In this paper, the value of β is set to 0.5 and decreases by factor of 1.02 for all benchmarks in each iteration to accelerate convergence by alleviating penalty on increase in stripe area.

$$\partial W_k = -\frac{\rho L_k}{R_k^2} \partial R_k \quad (13)$$

In short summary, in each iteration, SLP will determine which power stripe needs to be adjusted to gain maximum efficiency minimizing IR drop. To limit increase in power stripe area, increase in power stripe area is translated to increase in stripe resistance and a penalty function β is introduced to penalize greedy increase in power stripe area. Since sensitivity matrix G_s is nonlinear, ∂R is limited to 1% of R in each iteration and G_s is reconstructed after each iteration.

$$\begin{aligned} \min. \quad & \sum_{i \in n} (V_{min} - v_i) - \beta \sum_{k \in \{K_x, K_y\}} \frac{\rho L_k^2}{R_k^2} \partial R_k, \quad \forall v_i \leq V_{min} \\ \text{s. t.} \quad & V' = V + G_s \partial R \\ & \frac{\rho L_k}{W_{min}} \geq R_k + \partial R_k \geq \frac{\rho L_k}{W_{Max}}, \quad \forall k \in \{K_x, K_y\} \\ & \partial R_k \leq 0.01 R_k, \quad \forall k \in \{K_x, K_y\} \end{aligned} \quad (14)$$

V. EXPERIMENTAL RESULT

We experiment our framework on two real design cases. The first design is a micro processing unit with 29996 cells in TSMC 90nm technology. Second design is a LDPC decoder with 296122 cells in UMC 90nm technology¹ The output of our framework is

¹We have obtained RTL code for both designs and then synthesize both designs in commercial environment. The micro processing unit design is obtained from ITRI, Taiwan. The LDPC design is obtained from corresponding authors in [13].

a set of commands that draws the topology of power network on EDI [12] platform. The power analysis engine in EDI is used for final verification on design rule violations including IR/EM violations. Both designs after our power network synthesis are DRC/LVS clean.

Design information of the given designs including sheet resistance, cell's power dissipation, via resistance, dimension and position of cell are obtained using OpenAccess [11]. The entire algorithm is written in C++ with standard template library. Our algorithm is compiled under g++ 4.1.2 using Cent OS 5.0 on Intel XEON 5530 machine running at 2.4 Ghz. IBM ILOG CPLEX Optimizer v12.2 [17] is used to solve linear programming in Section IV. Design rule on antenna effect is neglected and activity factor is set to 0.2 for all cells².

Experimental results are presented in two subsections. First subsection performs our framework on a micro processing unit design and second subsection performs our framework on a Low Density Parity Check (LDPC) decoder design.

A. Micro Processing Unit Design (TSMC 90nm)

In first design, three different approaches are conducted to synthesize the power network. First approach synthesizes power network using clustering based method described in Section III and optimizes power network using SLP described in Section IV. Second approach is commonly seen in most manual approach which synthesizes power network by placing power stripes evenly apart and adjusts width of power stripes uniformly. Third approach synthesizes power network using same method as second approach which is a uniform mesh but optimizes it using SLP. In this paper, adjustment of power stripes limits to main power branches and width of pre-planned power stripes are unchanged. Voltage sources are placed at top left and bottom right corner of the design.

Table II lists the performance of the three approaches using first design. Minimum voltage V_{min} is set to 0.860V and optimization terminates when min. voltage of power network obtained from our nodal analysis exceeds V_{min} . Regarding to Table II, only very slight deviation between min. voltage using our nodal analysis and power analysis in EDI which demonstrates accuracy of our power network model and analysis. Our optimization method (Cluster/SLP) achieves best area efficiency while satisfying given V_{min} constraint.

Table III lists the distribution from EDI of nodal voltage for all three approaches. Although Uniform/SLP approach have better voltage distribution than cluster/SLP approach, it has 366 (132+234) nodes below the 0.865625V threshold. In contrast, the Cluster/SLP approach has only 3 nodes below the 0.865625V threshold using less power network area.

B. LDPC Decoder (UMC 90nm)

Table IV compares our optimization framework with manual approach. The manual approach is the power network synthesized in the work done in [13]. The topology of power network done in [13] is a uniform mesh based power network which relied on power network synthesizer provided by the commercial design platform. Designer specifies the number and width of stripes in horizontal and vertical direction, and then synthesizer generates corresponding mesh network. Generally, designers tend to over design the power network to avoid IR and EM violations. However, over design the power network leaves tremendous burden at routing stage. On this particular design [13], routing took nearly 24 hours to complete. Reduce number of power stripes and minimize width of each individual power stripe reduces routing block area.

Regarding to Table IV, the power network generated using our optimization framework can achieve 9% less power network area while

²Our power analysis engine is verified on the entire set of TAU benchmarks [7] and achieves accuracy within 0.0005%.

TABLE II. STATISTICS OF POWER NETWORK OPTIMIZATION ON MICRO PROCESSING UNIT DESIGN (TSMC 90NM) USING CLUSTERING BASED MESH NETWORK W/ SLP SIZING, UNIFORM POWER MESH NETWORK W/ UNIFORM SIZING AND UNIFORM POWER MESH NETWORK W/ SLP SIZING

	Cluster/SLP	Uniform/Uniform	Uniform/SLP
Synthesis Method	Clustering	Uniform Mesh	Uniform Mesh
Optimization Method	SLP Sizing	Uniform Sizing	SLP Sizing
# of Hori. Stripes	4	4	4
# of Vert. Stripes	3	4	3
Avg. Vol. (V) - EDI	0.871866	0.871477	0.872905
Min. Vol. (V) - EDI	0.865624	0.864340	0.860875
Min. Vol. (V) - Our	0.866926	0.868062	0.860250
Hori. Pow. Area(μm^2)	9909.36	16350.44	9909.36
Vert. Pow. Area(μm^2)	7192.57	8631.08	8173.38
Ttl. Pow. Area (μm^2)	17101.93	24981.53	18082.74
Norm. Ttl. Area(μm^2)	0.685	1.000	0.724

TABLE III. DISTRIBUTION OF NODAL VOLTAGE ON MICRO PROCESSING UNIT DESIGN (TSMC 90NM) USING CLUSTERING BASED MESH NETWORK W/ SLP SIZING, UNIFORM POWER MESH NETWORK W/ UNIFORM SIZING AND UNIFORM POWER MESH NETWORK W/ SLP SIZING

Voltage Range	Cluster / SLP		Uniform / Uniform		Uniform / SLP	
	Node #	%	Node #	%	Node #	%
0.900000-0.881250	3639	1.08	2237	0.65	5006	1.48
0.881250-0.878125	4665	1.38	4173	1.21	5631	1.66
0.878125-0.875000	23807	7.06	19524	5.64	41224	12.16
0.875000-0.871875	90171	26.73	74100	21.42	139979	41.29
0.871875-0.868750	211378	62.67	240199	69.43	146070	43.09
0.868750-0.865625	3642	1.08	5555	1.61	717	0.21
0.865625-0.862500	3	0.00	180	0.05	132	0.04
0.862500-0.860875	0	0.00	0	0.00	234	0.07

TABLE IV. ANALYSIS OF POWER NETWORK ON A LDPC DECODER(UMC 90NM) USING OUR PROPOSED OPTIMIZATION WITH MANUAL APPROACH

	Ours	Manual [13]
Synthesis Method	Clustering	Manual Mesh
Optimization Method	SLP Sizing	Manual Sizing
# of Hori. Stripes	9	9
# of Vert. Stripes	22	33
Avg. Vol. (V) - EDI	0.850177	0.848048
Min. Vol. (V) - EDI	0.790982	0.785062
Hori. Pow. Area(μm^2)	36855.00	49140.00
Vert. Pow. Area(μm^2)	77006.00	76450.00
Ttl. Pow. Area (μm^2)	113861.00	125590.00
Norm. Ttl. Area(μm^2)	0.907	1.000

maintaining a higher minimum nodal voltage across entire chip. This demonstrates our proposed approach can effectively minimize power network area while not violating the IR and EM design rules.

VI. CONCLUSIONS

In this paper, we propose a complete and realistic power network synthesis and optimization flow. Using placed design as input, our methodology analyzes given design, synthesizes an appropriate power network and then iteratively refines the power network until it satisfies the given constraint. Optimization of power network can only be achieved through accurate modelling and nodal analysis. Our methodology is evaluated to the best of our extent on both academic and industrial designs. Evaluation demonstrates effectiveness of our methodology.

REFERENCES

- [1] H. Chen, C.-K. Cheng, A. B. Kahng, M. Mori, and Q. Wang, "Optimal Planning for Mesh-Based Power Distribution," in *Proceedings of the Asia and South Pacific Design Automation Conference*, ASP-DAC, pp. 444–449, 2004.
- [2] X.-D. Tan, R. Shi, and J.-C. Lee, "Reliability-Constrained Area Optimization of VLSI Power/Ground Networks via Sequence of Linear Programmings," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 22, pp. 1678–1684, Dec. 2003.
- [3] X. Wu, X. Hon, Y. Ca, C. Cheng, J. Gu, and W. Dai, "Area Minimization of Power Distribution Network Using Efficient Nonlinear Programming Techniques," in *Proceedings of the International Conference on Computer Aided Design*, pp. 153–157, 2001.
- [4] H. Su, K. Gala, and S. S. Sapatnekar, "Fast Analysis and Optimization of Power/Ground Networks," in *Proceedings of the International Conference on Computer-Aided Design, ICCAD*, pp. 477–482, 2000.
- [5] S. Boyd, L. Vandenberghe, A. E. Gamal, and S. Yun., "Design of Robust Global Power and Ground Networks," in *Proceedings of the International Symposium on Physical Design, ISPD*, pp. 60–65, Apr. 2001.
- [6] T. Wang and C. C. Chen, "Optimization of the Power/Ground Network Wire-Sizing and Spacing Based on Sequential Network Simplex Algorithm," in *Proceedings of the International Symposium Quality Electron Design, ISQED*, pp. 157–162, 2002.
- [7] Z. Li, R. Balasubramanian, F. Liu, and S. Nassif, "2011 TAU Power Grid Simulation Contest: Benchmark Suite and Results," in *Proceedings of the International Conference on Computer Aided Design*, pp. 478–481, 2001.
- [8] J. Yang, Z. Li, Y. Cai, and Q. Zhou, "PowerRush: A linear Simulator for Power Grid," in *Proceedings of the International Conference on Computer Aided Design*, pp. 483–487, 2012.
- [9] Z. Zeng, T. Xu, Z. Feng, and P. Li, "Fast Static Analysis of Power Grids: Algorithms and Implementations," in *Proceedings of the International Conference on Computer Aided Design*, pp. 488–493, 2012.
- [10] C.-H. Chou, N.-Y. Tsai, Hao-Yu, C.-R. Lee, Y. Shi, and S.-C. Chang, "On the Preconditioner of Conjugate Gradient Method – A Power Grid Simulation Perspective," in *Proceedings of the International Conference on Computer Aided Design*, pp. 494–497, 2012.
- [11] M. Guiney and E. Leavitt, "An Introduction to OpenAccess: An Open Source Data Model and API for IC Design," in *Proceedings of the Asia and South Pacific Design Automation Conference*, pp. 434–436, 2006.
- [12] "EDI 10.1." <http://www.cadence.com/products/di/pages/default.aspx>.
- [13] C.-L. Chen, K.-S. Lin, H.-C. Chang, W.-C. Fang, and C.-Y. Lee, "A 11.5-Gbps LDPC Decoder Based on CP-PEG Code Construction," in *Proceedings of ESSCIRC*, pp. 412–415, Sept. 2009.
- [14] R. Tibshirani, G. Walther, and T. Hastie, "Estimating the Number of Clusters in a Data Set via The Gap Statistic," *Journal of The Royal Statistical Society B-statistical Methodology*, vol. 63, pp. 411–423, 2001.
- [15] S. Lloyd, "Least Squares Quantization in PCM," *IEEE Transactions on Information Theory*, vol. 28, no. 2, pp. 129–137, 1982.
- [16] D. Arthur and S. Vassilvitskii, "k-means++: The Advantages of Careful Seeding," in *Symposium on Discrete Algorithms*, pp. 1027–1035, 2007.
- [17] "IBM ILOG CPLEX Optimizer." <http://www-01.ibm.com/software/integration/optimization/cplex-optimizer/>.