

2023 International Symposium on Physical Design



Voltage-Drop Optimization Through Insertion of Extra Stripes to a Power Delivery Network

Jai-Ming Lin , Yu-Tien Chen, Yang-Tai Kung, and Hao-Jia Lin

Speaker: Hao-Jia Lin

March 2023

Department of Electrical Engineering,
National Cheng Kung University, Tainan, Taiwan

2/25/2023



Outline

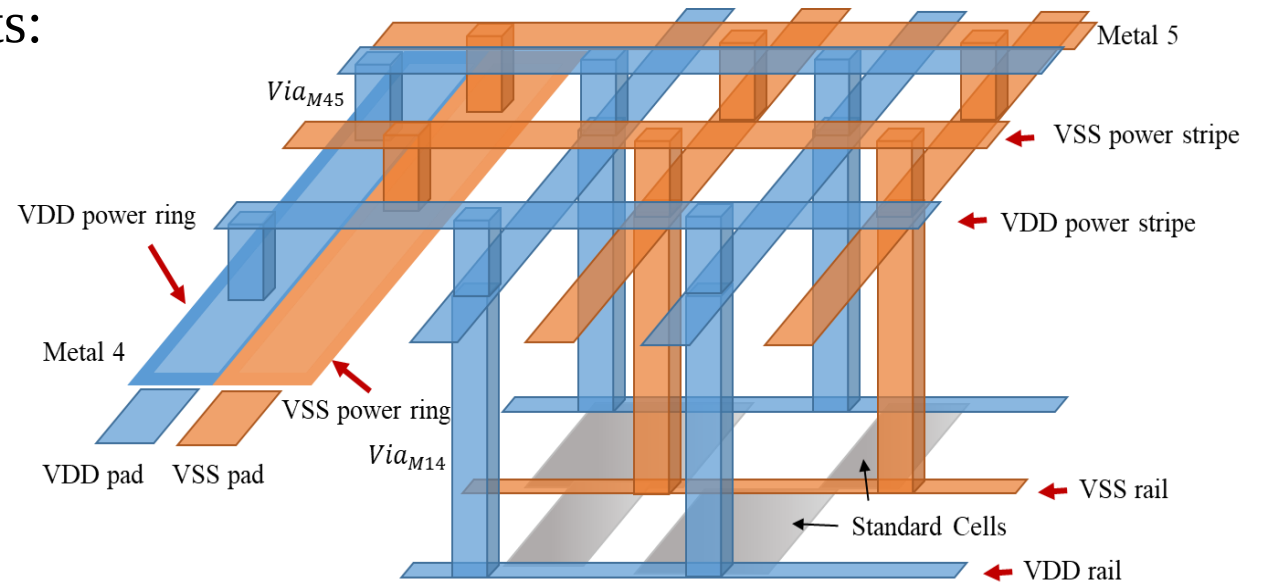
- ◆ Introduction
- ◆ Problem Formulation
- ◆ Our PDN Optimization Methodology
- ◆ Experimental Results
- ◆ Conclusion

Outline

- ◆ Introduction
- ◆ Problem Formulation
- ◆ Our PDN Optimization Methodology
- ◆ Experimental Results
- ◆ Conclusion

Introduction

- ◆ Powerplanning becomes a more important step in the physical design because an improper power network will induce severe IR-drop violations which not only impact circuit performance but also may induce function failure.
- ◆ A Power delivery network (PDN for short) provides supply voltage to macros and standard cells, which is composed of the following elements:
 - ◆ Power pads
 - ◆ Power rings
 - ◆ Power stripes
 - ◆ Horizontal power stripes (**HPSs** for short)
 - ◆ Vertical power stripes (**VPSs** for short)
 - ◆ Vias
 - ◆ Power rails
- ◆ PDN optimization after the post-placement stage becomes indispensable for a modern VLSI design.



Our Contributions

- ◆ Propose a PDN optimization approach by insertion of additional power stripes, which can repair voltage violations without deteriorating routability.
 - ◆ Construct IR-drop high related regions (HRRs for short) to indicate regions which require more currents.
 - ◆ Propose a minimum-cost flow problem (MCFP for short) formulation to find the topology of an additional power delivery path (PDP for short) and determine the width of edges in the path.
 - ◆ Consider obstacles by construction of an obstacle-aware spanning graph.
 - ◆ Minimize usage of routing resource while meeting current demands in the voltage violation regions.
 - ◆ Avoid deteriorating routing congestion by adding power stripes to the locations which have severe voltage violations and less routing congestion by dynamic programming (DP for short).
 - ◆ Fix the problem by inserting power stripes into less routing congestion locations in HRRs after global insertion step.
- ◆ Experimental results show that our methodology can use much less routing area to repair violations than other sizing methods [1], [2], and induce less routing overflow.

[1] S.S.-Y. Liu, C.-J. Lee, C.-C. Huang, H.-M. Chen, C.-T. Lin and C.-H. Lee, “Effective Power Network Prototyping Via Statistical-Based Clustering and Sequential Linear Programming,” in Proc. DATE, Mar. 2013.

[2] J.-M. Lin, Y.-T. Kung, Z.-Y. Huang, I.-R. Chen, “A Fast Power Network Optimization Algorithm for Improving Dynamic IR-drop,” in Proc. ISPD, Mar. 2021.

Outline

- ◆ Introduction
- ◆ Problem Formulation
- ◆ Our PDN Optimization Methodology
- ◆ Experimental Results
- ◆ Conclusion

Problem Formulation

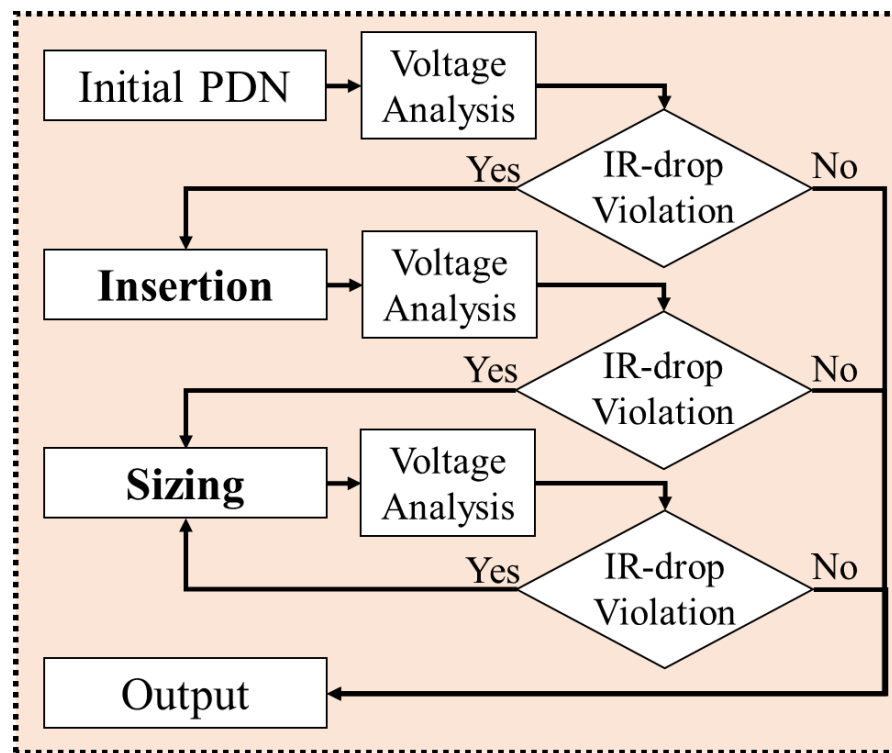
- ◆ Assume the shapes and locations of a power ring, HPSs, and power rails are determined.
- ◆ Input:
 - ◆ Locations and shapes of standard cells and macros from DEF and LEF files
 - ◆ Power consumption of standard cells and macros from a power profile
 - ◆ An initial PDN of a chip from TCL file
 - ◆ DRC rule from a technology file
- ◆ Output:
 - ◆ The locations, lengths, and widths of VPSs
- ◆ Constraints:
 - ◆ The IR-drop constraint
 - ◆ $\bar{v}$ denotes the maximum tolerable voltage drop value, and $\bar{v} = \theta \times v_s$ where v_s is the supply voltage and θ is the allowable voltage drop ratio.
 - ◆ The minimum width constraint
 - ◆ The maximum width constraint

Outline

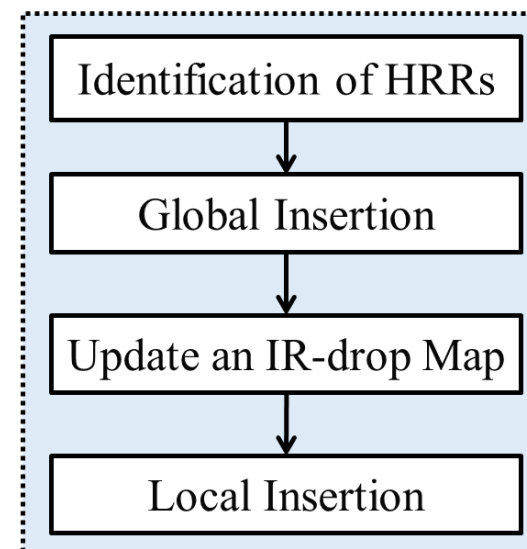
- ◆ Introduction
- ◆ Problem Formulation
- ◆ Our PDN Optimization Methodology
 - ◆ Overview of Our Methodology
 - ◆ Identification of HRRs
 - ◆ Global Insertion
- ◆ Experimental Results
- ◆ Conclusion

Overview of Our Methodology

- ◆ A two-stage optimization methodology:
 - ◆ **Insertion** stage: Insertion of additional VPSs.
 - ◆ **Sizing** stage: Sizing of VPSs [2].



(a)



(b)

Outline

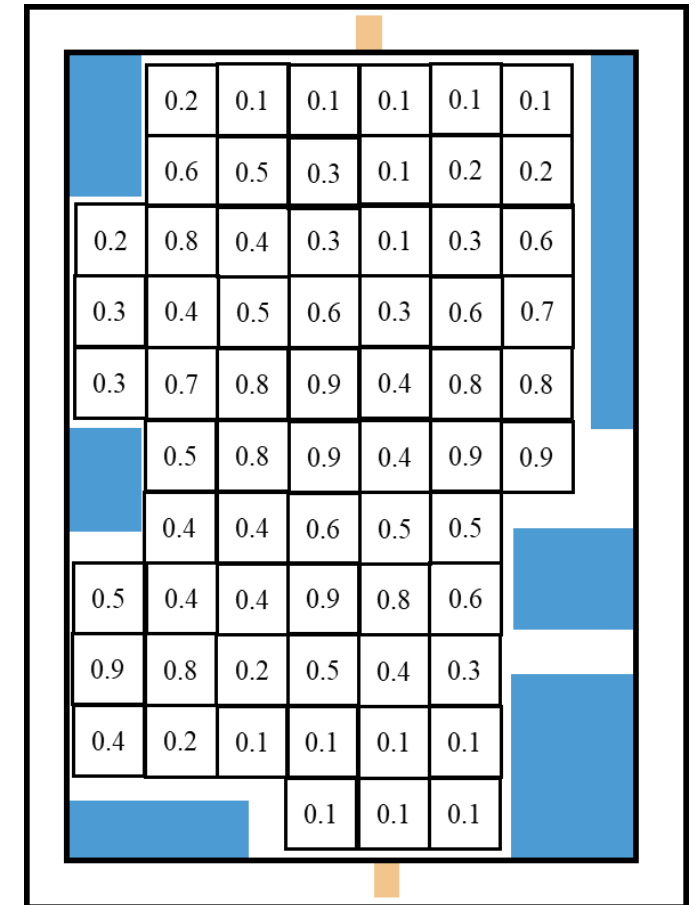
- ◆ Introduction
- ◆ Problem Formulation
- ◆ Our PDN Optimization Methodology
 - ◆ Overview of Our Methodology
 - ◆ Identification of HRRs
 - ◆ Global Insertion
- ◆ Experimental Results
- ◆ Conclusion

Construction of a Stripe-Inserting Tendency Map

- ◆ Divide the region with power stripes into uniform grids g_i 's.
- ◆ Estimate the **thirsty for power resource** in each grid g_i by the function $\varphi(g_i)$ as follows:

$$\varphi(g_i) = \alpha * \frac{\log(P_g(i))}{\log(P_g^{max})} + \beta * \frac{D_p(i)}{D_p^{max}} + \gamma * \frac{D_s(i)}{D_s^{max}}$$

- ◆ $P_g(i)$ denotes the **total power consumptions** of cells and macros in g_i .
- ◆ P_g^{max} denotes the maximum value of $P_g(i)$'s for all g_i 's.
- ◆ $D_p(i)$ denotes the Manhattan **distance** from g_i to its **nearest power pad**.
- ◆ D_p^{max} denotes the maximum value of $D_p(i)$'s for all g_i 's.
- ◆ $D_s(i)$ denotes the Manhattan **distance** from g_i to its **nearest VPS**.
- ◆ D_s^{max} denotes the maximum value of $D_s(i)$'s for all g_i 's.
- ◆ $\alpha, \beta, \text{ and } \gamma$ denote user specified parameters.



▲ An SITM

Identification of HRRs

- ◆ Construct a graph $M(V, E_M)$ according to the grids g_i 's:
 - ◆ Initialize a vertex v_i for each g_i .
 - ◆ Initialize an edge (v_i, v_j) for every contiguous grids g_i and g_j .
- ◆ Apply **Best-Choice algorithm** [3] to cluster vertices in $M(V, E_M)$ and select HRRs from the resulting clusters.
 - ◆ Each vertex v_i is considered as a cluster in the beginning, and the cost $\varphi(v_i)$ of v_i equals to $\varphi(g_i)$.
 - ◆ Repeatedly **combine** adjacent vertex v_i and v_j with **the smallest score value** and replace them by a new vertex v_k in $M(V, E_M)$.
 - ◆ The cost $\varphi(v_k)$ of v_k is estimated by the following equation:
$$\varphi(v_k) = \frac{\sum_{g_l \in v_k} \varphi(g_l)}{s(k)}$$
 - ◆ The region corresponding to v_k is considered as an HRR h_j if its **area** is **large** enough and $\varphi(v_k) > \chi$.
 - ◆ χ is user specified value.

Identification of HRRs (cont'd)

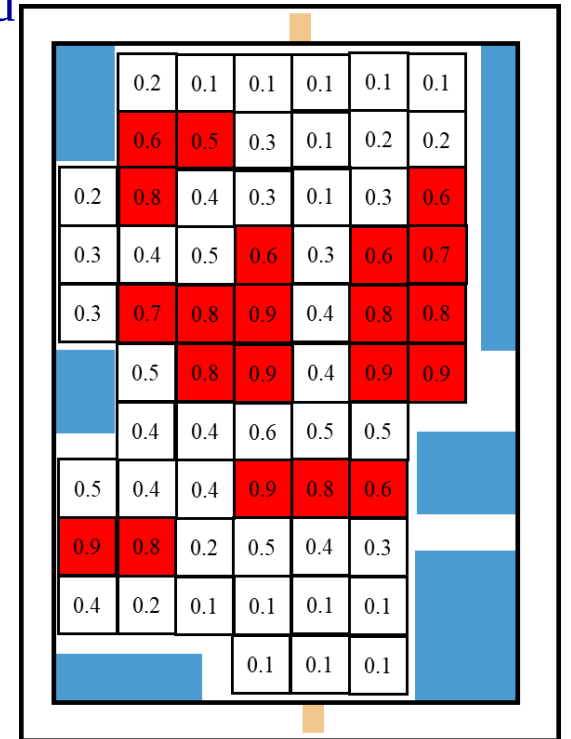
- ◆ The **score value** to merge v_i and v_j is computed by the following equation:

$$\omega(i, j) = \sigma(|\varphi(v_i) - \varphi(v_j)|) * \sigma\left(\frac{D(i, j)}{W_{chip} + L_{chip}}\right) * (s(i) + s(j))$$

- ◆ $D(i, j)$ denotes the Manhattan distance between geometry center of the associated regions of v_i and v_j .
- ◆ W_{chip} and L_{chip} denote the width and length of a chip, respectively.
- ◆ $s(i)$ denotes the number of grids in v_i .
- ◆ $\sigma(x)$ denotes **the sigmoid function**.

$$\sigma(x) = \frac{1}{1 + e^{-slope(x-m)}}$$

- ◆ *slope* and *m* denote user specified parameters.



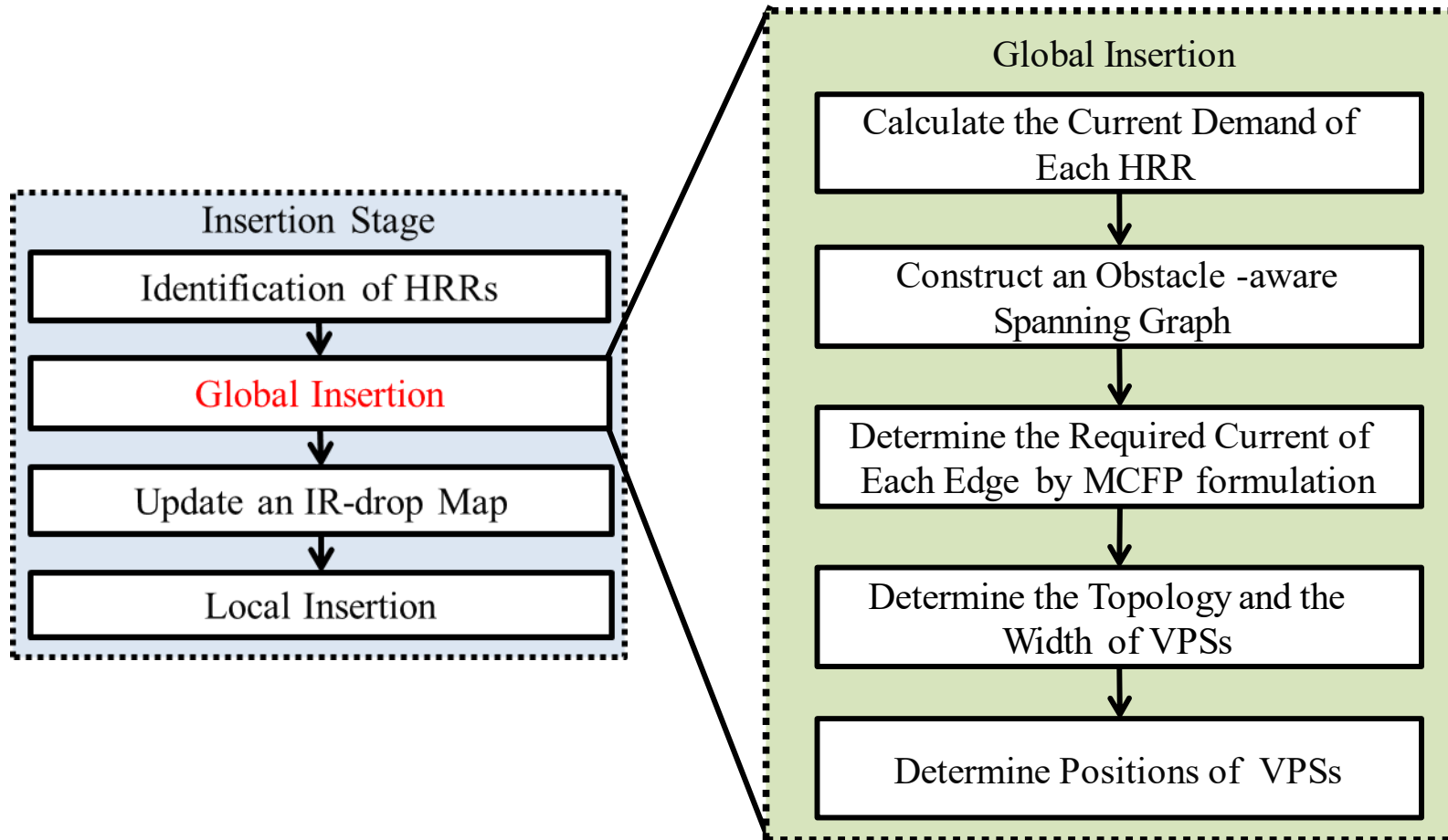
▲ HRRs.

Outline

- ◆ Introduction
- ◆ Problem Formulation
- ◆ Our PDN Optimization Methodology
 - ◆ Overview of Our Methodology
 - ◆ Identification of HRRs
 - ◆ Global Insertion
- ◆ Experimental Results
- ◆ Conclusion

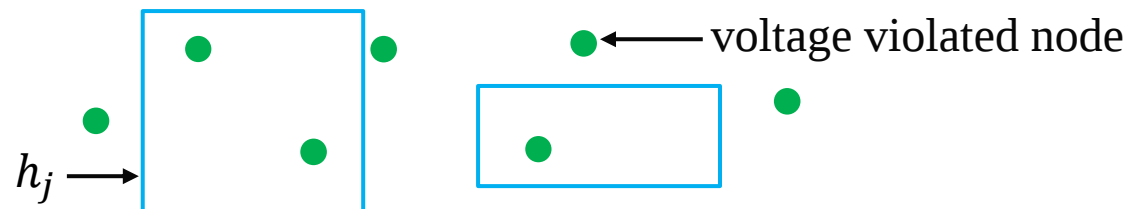
Global Insertion

- ◆ The global insertion step in insertion stage:



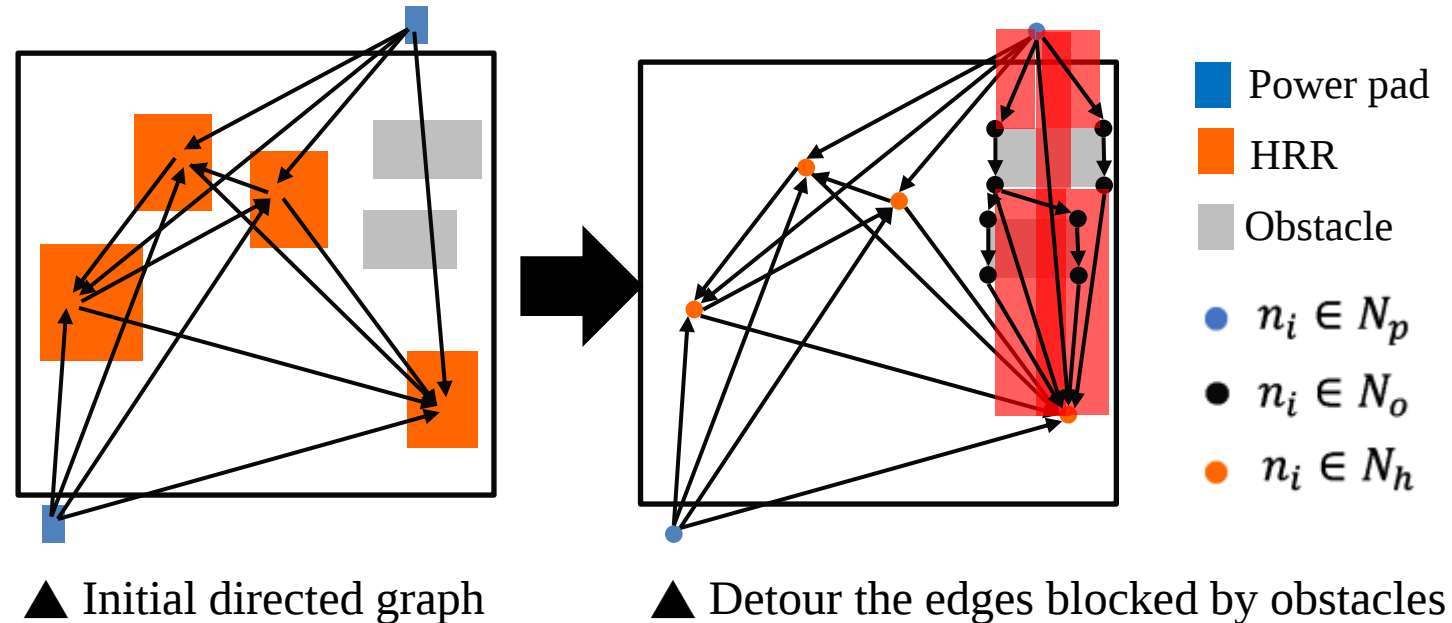
Calculation of the Current Demand for Each HRR

- ◆ Construct a graph $K(U, E_K)$ according to a PDN.
 - ◆ Each node $u_i \in U$ denotes a cross-point between a VPS and an HPS.
 - ◆ Each edge $(u_i, u_j) \in E_K$ represents the adjacent nodes u_i and u_j .
- ◆ Current demand d_i of each node u_i in a power network K is estimated as follows:
$$d_i = \frac{\Delta(i)}{\Omega(i)}$$
 - ◆ $\Delta(i)$ represents the voltage violation value of u_i .
 - ◆ $\Omega(i)$ denotes the resistance of the power delivery path from u_i to its nearest power pad.
- ◆ Assign each voltage violated node to the closest HRR h_j according to the Manhattan distance between u_i and the weighted center of h_j .
- ◆ The current demand I_j of each h_j is obtained by accumulation of d_i 's assigned to h_j .



Construction of an Obstacle-aware Spanning Graph (cont'd)

- ◆ Construct a directed graph $\vec{G}(N, \vec{E})$ which contains possible paths to find the power delivery path from power sources to all HRRs.
 - ◆ N is composed of a set of nodes n_i 's.
 - ◆ N_p denotes a set of nodes for power pads.
 - ◆ N_h denotes a set of nodes for HRRs.
 - ◆ N_o denotes the nodes for corners of obstacles (i.e., without power consumption).
 - ◆ $\vec{E}$ is composed of a set of edges $\vec{e}_{i,j}$'s.



Determination of the Required Current of Each Edge

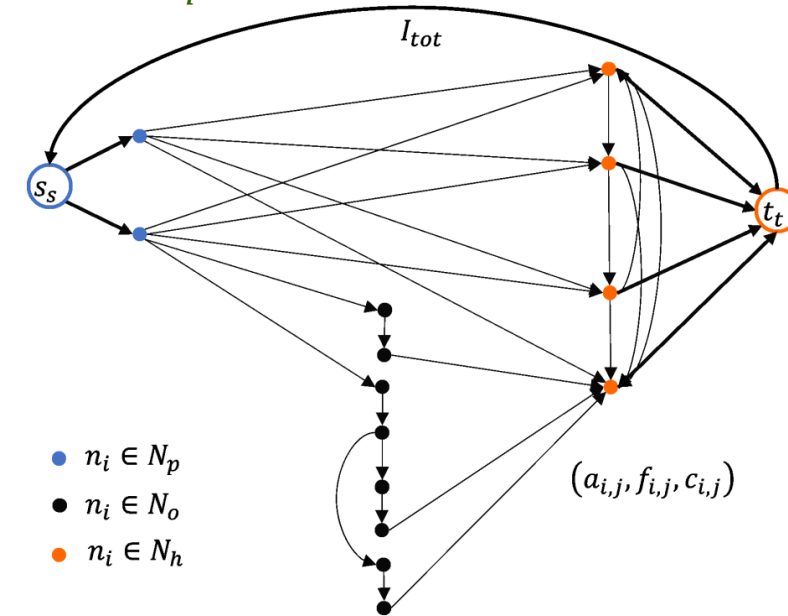
- ◆ Transform $\vec{G}(N, \vec{E})$ into a flow network to determine the required current $f_{i,j}$ of each edge $\vec{e}_{i,j}$ in $\vec{G}(N, \vec{E})$ which can meet **current demand in each HRR**.
 - ◆ Add a pseudo super sink t_t and add an edge for each node $n_i \in N_h$ and t_t .
 - ◆ Add a pseudo super source s_s and connect s_s to each node $n_i \in N_p$ by an edge.
 - ◆ Connect t_t to s_s by an edge with a required current value I_{tot} .
 - ◆ I_{tot} equals to the total demand of all nodes in N_h and the total supply of all nodes in N_p .

- ◆ Each edge $\vec{e}_{i,j}$ in the network is associated with a triple $(a_{i,j}, f_{i,j}, c_{i,j})$.

- ◆ $c_{i,j}$ is the current capacity of $\vec{e}_{i,j}$ which is the upper bound of $f_{i,j}$.
- ◆ $a_{i,j}$ is the cost of $\vec{e}_{i,j}$ which is computed by the following equation:

$$a_{i,j} = \varsigma \times \frac{\Delta y_{i,j}^2}{L_{chip}^2} + (1 - \varsigma) \times \frac{\Delta x_{i,j}}{W_{chip}}$$

- ◆ ς is a user specified parameter whose value is between 0 and 1.



Determination of the Required Current of Each Edge (cont'd)

◆ Two lemmas for the cost of each edge:

◆ Lemma 1. The area $A_{i,j}$ of inserted VPSs to an edge $\vec{e}_{i,j}$ is proportional to $|f_{i,j}| \Delta y_{i,j}^2$.

◆ $\Delta y_{i,j}$ denotes the height of $B_{i,j}$, which is the smallest bounding box enclosing $\vec{e}_{i,j}$.

◆ Lemma 2. The IR-drop $\Delta v_{i,j}$ in the horizontal direction to an edge $\vec{e}_{i,j}$ is proportional to $|f_{i,j}| \Delta x_{i,j}$.

◆ $\Delta x_{i,j}$ denotes the width of $B_{i,j}$.

◆ According to the lemmas, total **routing resource** and **voltage violations in the vertical direction** are minimized after the MCFP is solved because it will optimize the following objective function:

$$\min. \sum_{\vec{e}_{i,j} \in \vec{E}} |f_{i,j}| * a_{i,j}$$

◆ $f_{i,j}$ is the current flowing through each $\vec{e}_{i,j}$

Determination of the Topology and the Width of VPSs

- ◆ The topology of power delivery paths (PDA) are determined according to those edges $\vec{e}_{i,j}$'s with nonzero $f_{i,j}$
 - ◆ Each edge $\vec{e}_{i,j}$ represents a PDP and is denoted by $D_{i,j}$.
 - ◆ Insert pieces of VPSs along $D_{i,j}$.

- ◆ The wire width in $D_{i,j}$ is $w_{i,j} = \left\lceil \frac{W_{i,j}^V}{\frac{W_{i,j}^V}{W_{max}}} \right\rceil$.

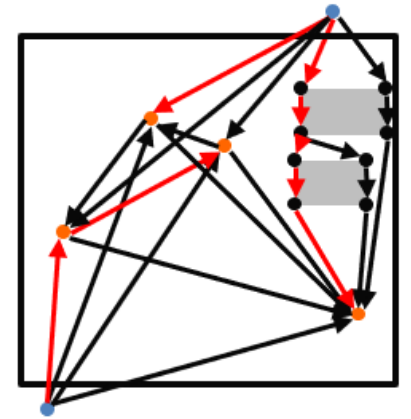
- ◆ W_{max} denotes the maximum width of a net in the layer.
- ◆ According to the equivalent circuit model:

- ◆ $v_a = v_s - \bar{v} \Rightarrow \bar{v} = v_s - v_a$
 - v_s and v_a are the voltages of source and $Node_a$, respectively.

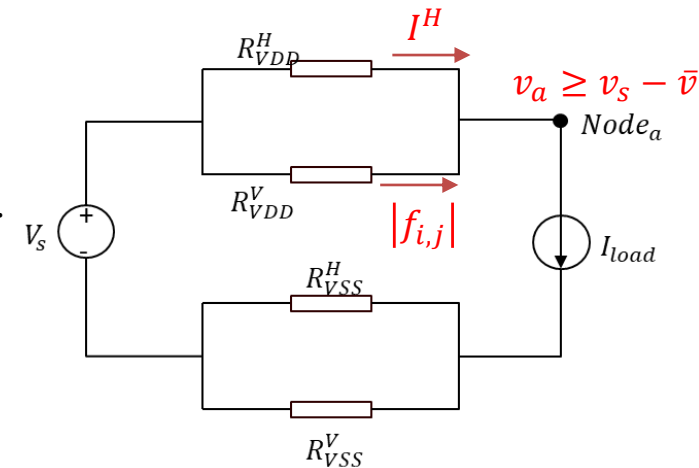
- ◆ $\bar{v} = |f_{i,j}| \times R_{VDD}^V \Rightarrow \bar{v} = |f_{i,j}| \times \rho^V \times \frac{\Delta y_{i,j}}{W_{i,j}^V}$
 - R_{VDD}^V and ρ^V are the resistance and the electrical resistivity of the layer of the VPSs, respectively.

- ◆ The required total width $W_{i,j}^V$ of $D_{i,j}$ is computed as follows:

$$W_{i,j}^V = \frac{|f_{i,j}| \times \rho^V \times \Delta y_{i,j}}{\bar{v}}$$



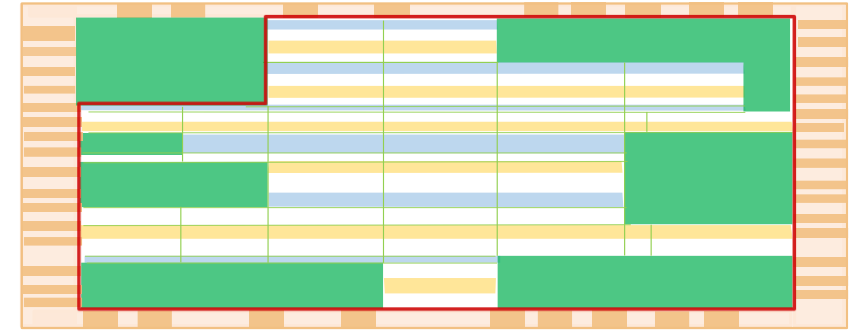
▲ The final topology for power delivery



▲ The equivalent circuit model

Determination of Positions of VPSs

- ◆ The positions to insert VPSs along $D_{i,j}$ is determined by the dynamic programming (DP) algorithm [4].
 - ◆ Let $B_{i,j}$ denote the bounding box enclosing $D_{i,j}$.
 - ◆ $B_{i,j}$ is divided into several bins b_l 's.
 - ◆ The width of b_l is identical to the pitch of a power routing track.
 - ◆ The height of b_l is equal to the height of r_k where it locates.

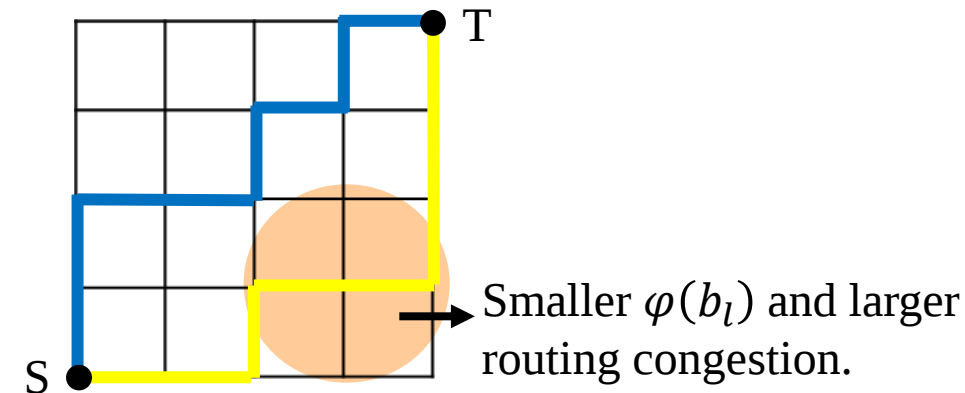


▲ Rows constructed in a placement region

- ◆ To insert VPSs at regions with larger thirst of additional power resource and less routing congestion, we find a monotonic routing path in $B_{i,j}$ with least cost $\psi(b_l)$ according to the cost function as follows:

$$\psi(b_l) = \zeta * \frac{\lambda(b_l)}{\lambda_b^{max}} + \mu * \left(1 - \frac{\varphi(b_l)}{\varphi_b^{max}}\right)$$

- ◆ $\lambda(b_l)$ denotes the total routing demand in b_l .
- ◆ λ_b^{max} denotes the maximum value of $\lambda(b_l)$'s for all b_l 's in $B_{i,j}$.
- ◆ $\varphi(b_l)$ denotes the thirsty of additional power resource of bin b_l .
- ◆ φ_b^{max} denotes the maximum value of $\varphi(b_l)$'s for all b_l 's in $B_{i,j}$.
- ◆ ζ and μ are user specified parameters.



Outline

- ◆ Introduction
- ◆ Problem Formulation
- ◆ Our PDN Optimization Methodology
- ◆ **Experimental Results**
- ◆ Conclusion

Experimental Results

◆ Environments:

Programming Language		C++
Linux Workstation	CPU	Intel® Xeon® E5500 2.27GHz
	Memory	90GB
	System	Cent OS 5.1

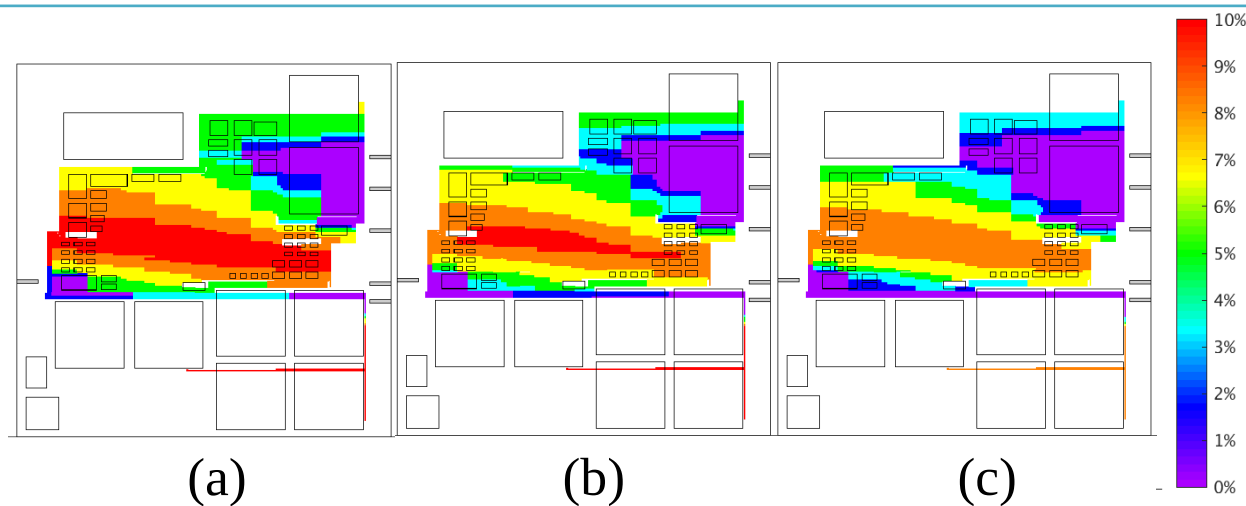
◆ Benchmarks: industrial designs

◆ The maximum IR-drop constraint is set to 10% of supply voltage.

Cir.	# of Cells	# of Macros	Supply Voltage (V)	max. IR-drop Constraint (mV)
Cir1	165090	31	1.08	108
Cir2	974362	118	1.08	108
Cir3	232872	70	1.08	108
Cir4	616593	53	1.32	132
Cir5	672952	222	1.10	110
Cir6	787141	51	1.20	120

Effect of Each Stage in Our Optimizing Methodology

Circuit	Original Status			Global Insertion Step			Local Insertion Step			Final Status		
	Area ($10^6 \mu m^2$)	Total O.V.	Max. IR- drop (mV)	Area ($10^6 \mu m^2$)	Total O.V.	Max. IR- drop (mV)	Area ($10^6 \mu m^2$)	Total O.V.	Max. IR- drop (mV)	Area ($10^6 \mu m^2$)	Total O.V.	Max. IR- drop (mV)
Cir1	4.844	51180	118.5	4.851	51444	113.4	4.853	51597	109.9	4.863	52706	106.4
Cir2	33.496	437566	129.6	33.559	439725	120.1	33.598	446258	112.1	33.651	448067	106.1
Cir3	5.371	42721	127.6	5.397	43555	118.5	5.426	45153	111.6	5.441	45862	105.5
Cir4	6.223	66740	154.2	6.232	67440	144.3	6.237	68342	135.2	6.247	70573	130.5
Cir5	1.107	98441	121.3	1.111	98441	117.2	1.112	98441	112.3	1.115	98441	108.4
Cir6	5.171	39318	131.4	5.172	39318	127.7	5.178	39318	122.7	5.184	39322	118.3
Nor.	1.000	1.000	1.000	1.002	1.005	0.947	1.003	1.018	0.899	1.005	1.026	0.864



Comparison of Our Methodology with Other Approaches

- ◆ Although our runtime is a little slower than the two approaches, our methodology can repair the voltage violations effectively by using a little routing resource.
 - ◆ The “Total O.V.” of the window-based sizing method and SLP method are larger than ours by 4.4% and 5%, respectively.
 - ◆ The “Increased Area” of the window-based sizing method and SLP method are much larger than our approach by 11 and 15 times, respectively.

Circuit	Window-Based Sizing Method [2]			SLP Method [1]			Our Method		
	Increased Area ($10^3\mu m^2$)	Total O.V.	Time (s)	Increased Area ($10^3\mu m^2$)	Total O.V.	Time (s)	Increased Area ($10^3\mu m^2$)	Total O.V.	Time (s)
Cir1	299	55220	16.25	371	56416	16.97	19	52706	17.60
Cir2	1749	467612	198.25	2465	469638	205.55	155	448067	224.97
Cir3	376	46784	29.11	457	47146	31.07	70	45862	33.01
Cir4	460	70840	58.94	511	71819	60.94	24	70573	61.26
Cir5	67	105397	53.42	96	109008	56.63	8	98441	58.74
Cir6	364	42347	28.18	420	42497	29.89	13	39322	31.14
Nor.	11.471	1.044	0.900	14.948	1.055	0.940	1.000	1.000	1.000

[1] S.S.-Y. Liu, C.-J. Lee, C.-C. Huang, H.-M. Chen, C.-T. Lin and C.-H. Lee, “Effective Power Network Prototyping Via Statistical-Based Clustering and Sequential Linear Programming,” in Proc. DATE, Mar. 2013.

[2] J.-M. Lin, Y.-T. Kung, Z.-Y. Huang, I.-R. Chen, “A Fast Power Network Optimization Algorithm for Improving Dynamic IR-drop,” in Proc. ISPD, Mar. 2021.

Outline

- ◆ Introduction
- ◆ Problem Formulation
- ◆ Our PDN Optimization Methodology
- ◆ Experimental Results
- ◆ Conclusion

Conclusion

- ◆ Propose a routability-aware PDN optimization methodology.
 - ◆ We have found proper power delivery paths with LP formulation to meet current demands in the voltage violation regions while considering obstacles.
 - ◆ We have placed power stripes in the locations which have severe voltage violations and less routing congestion according to the DP algorithm.
- ◆ The experimental results have shown that our methodology can repair voltage violations by inducing a few routing overflows and using a little routing area.

End

Thank You For Your Attention

