Floorplan and Power/Ground Network Co-Synthesis for Fast Design Convergence

Chen-Wei Liu[12] and Yao-Wen Chang[2]

[1]Synopsys Taiwan Limited
[2]Department of Electrical Engineering
National Taiwan University, Taipei Taiwan
April 11, 2006

Outline

- Introduction
- Proposed Design Flow
- Floorplan and P/G Network
  Co-Synthesis Algorithm
- Experimental Results

Outline

- **Introduction**
- Proposed Design Flow
- Floorplan and P/G Network Co-Synthesis Algorithm
- Experimental Results

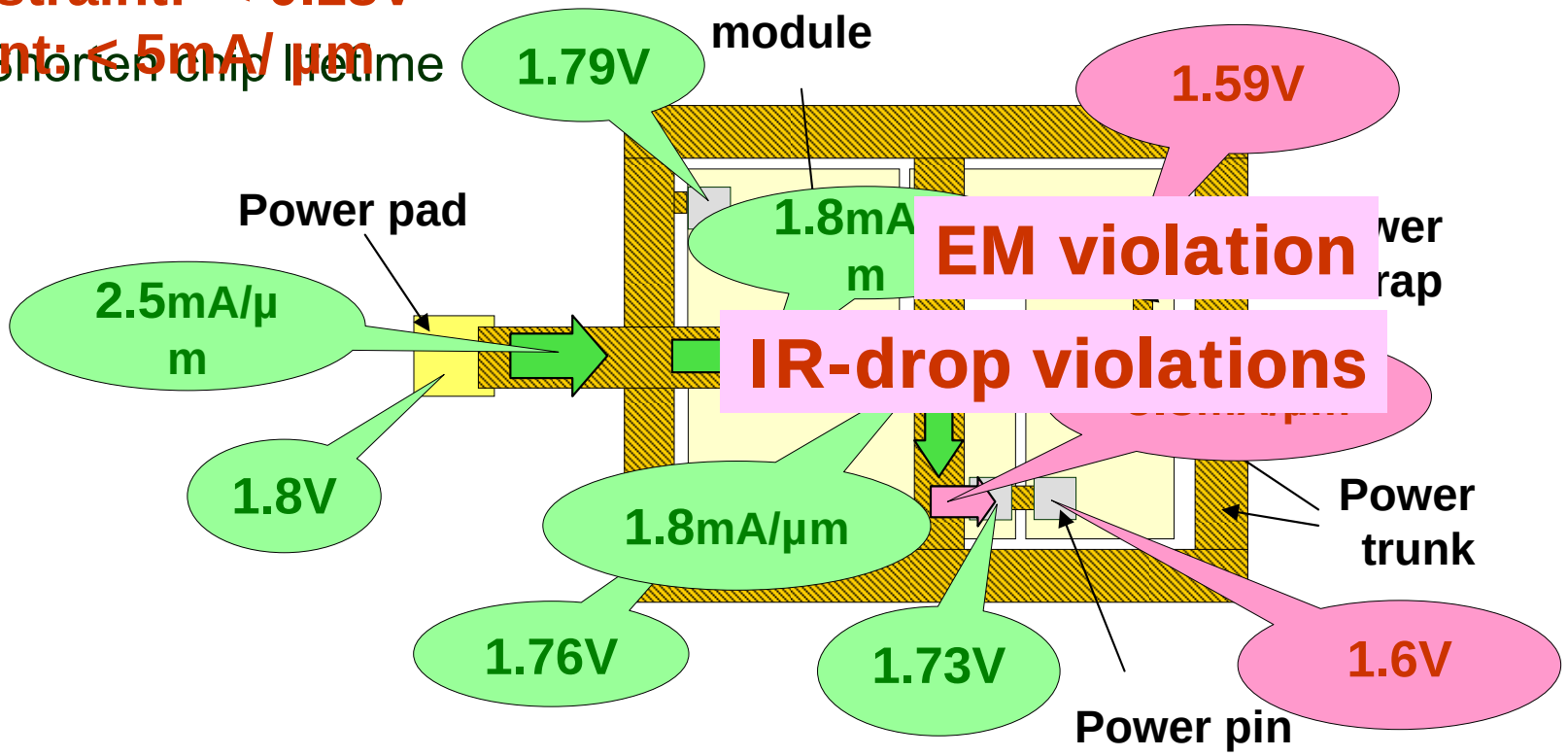
Introduction

- As technology advances, the metal width decreases while the global wirelength increases. Also, supply voltage is decreasing and the power density keeps increasing.
- The trends cause serious P/G network problems:
  - Voltage (IR) drop violation: serious IR-drop on a P/G network
    - Slow down clock rate and cause functional error
  - Electron-Migration (EM) violation: severe current density through a P/G wire
    - Shorten chip lifetime

Power Integrity Constraints

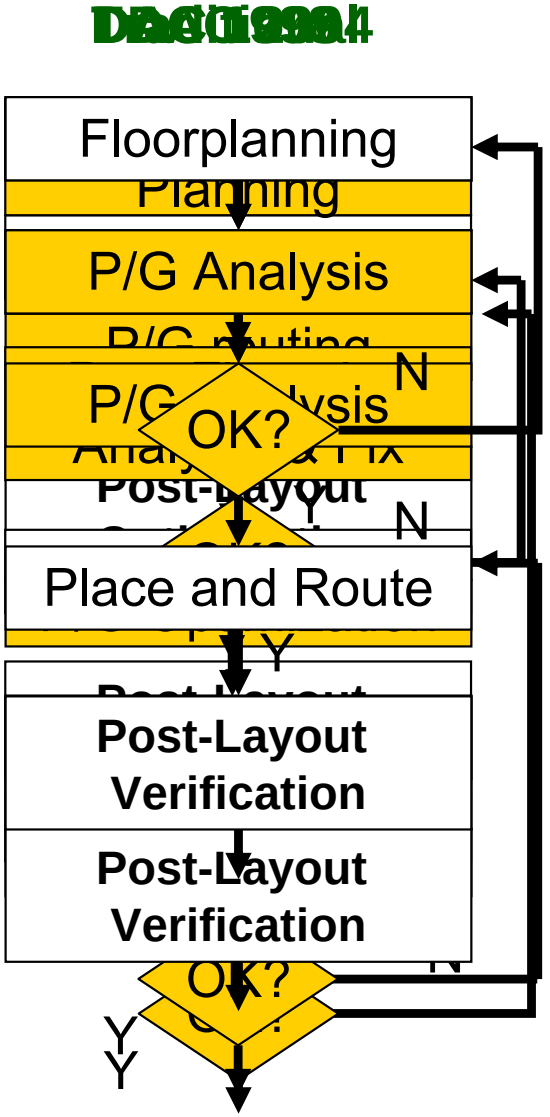
IR-drop constraint: < 0.18V

EM constraint: < 5mA/ µm



Previous Work

- Power integrity issues are dealt at post-layout stage in the traditional design flow
  - P/G network topology determination
    - Singh, et al., ISPD-04
  - P/G wire sizing
    - Wang and Shadoska, DAC-03,
    - Chowdhury et al., DAC-89
- As the design complexity increases, it is necessary to handle P/G network problems earlier
  - Dharchoudhury et al., DAC-98
    - Pre-floorplan, post-floorplan and post-layout P/G networks analysis and fix
  - Yim et al., DAC-99
    - Post-floorplan P/G networks planning
  - Wu and Chang, DAC-04
    - Iterative tree-structured P/G networks verification and fix with floorplan optimization



Our Contributions

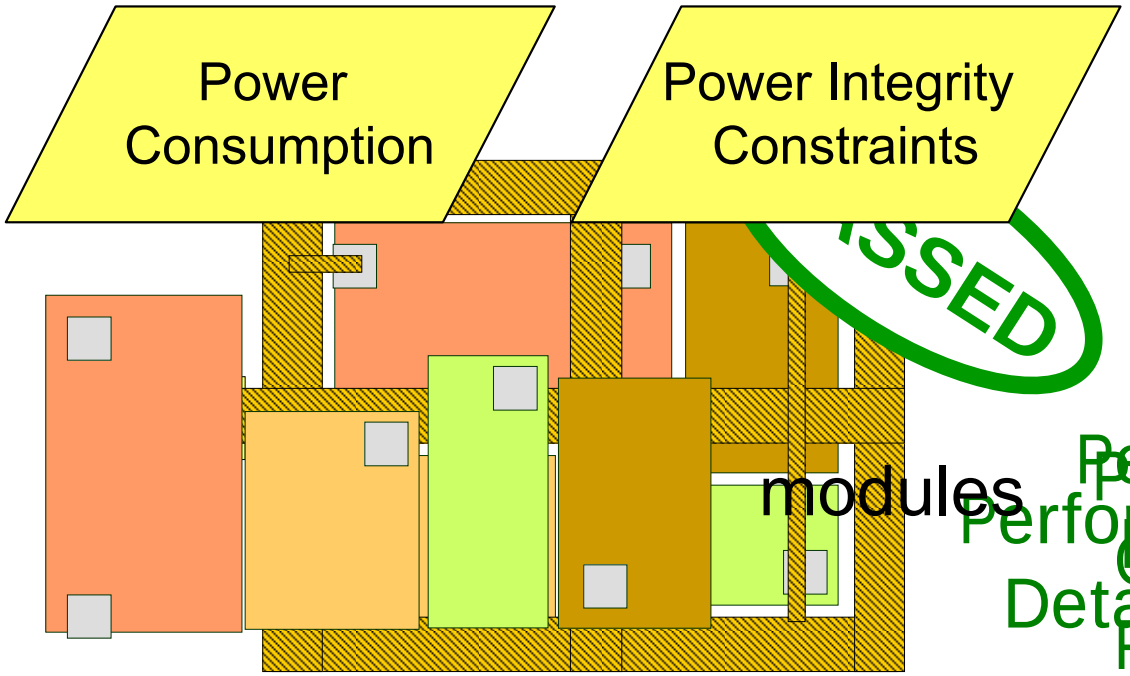
- Propose an automatic floorplan and P/G network co-synthesis method
- Develop a sophisticated model for fast P/G analysis
  - Make the co-synthesis design flow possible
- Develop P/G network aware method to reduce the floorplan solution space
  - Improve the runtime by an average of 68%
- Integrate into a commercial design flow to develop a power integrity driven design flow
  - 2.56X faster than the generic Astro design flow

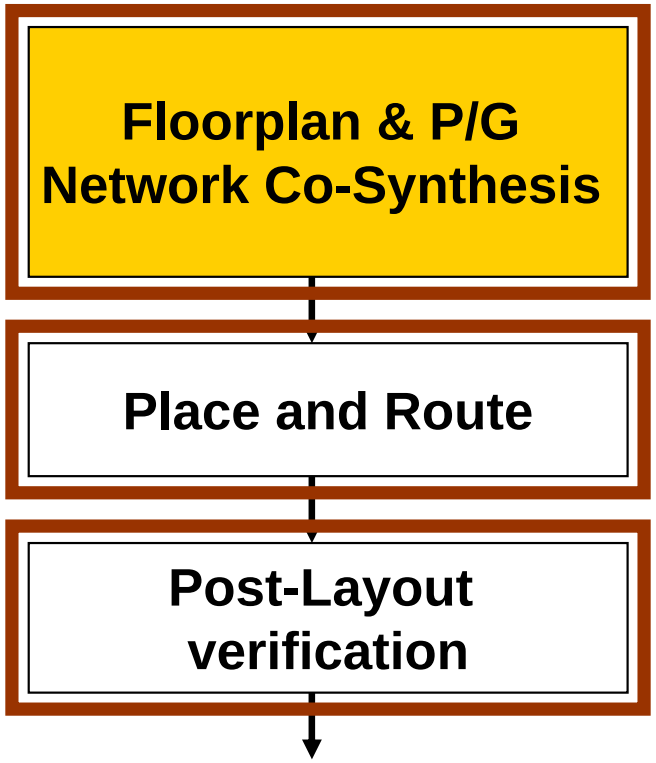
Outline

- Introduction
- **Proposed Design Flow**
- Floorplan and P/G Network
  Co-Synthesis Algorithm
- Experimental Results

Proposed Design Flow

- Floorplan & P/G network co-synthesis problem formulation
  - Given a set of modules, power consumption data, and power integrity constraints
  - Generate a floorplan and a power integrity feasible global P/G network
- Significantly improve the design convergence





Perform co-synthesis: Generate floorplan and P/G network plan

Perform place and route: Detailed P/G analysis

Perform post-layout verification:

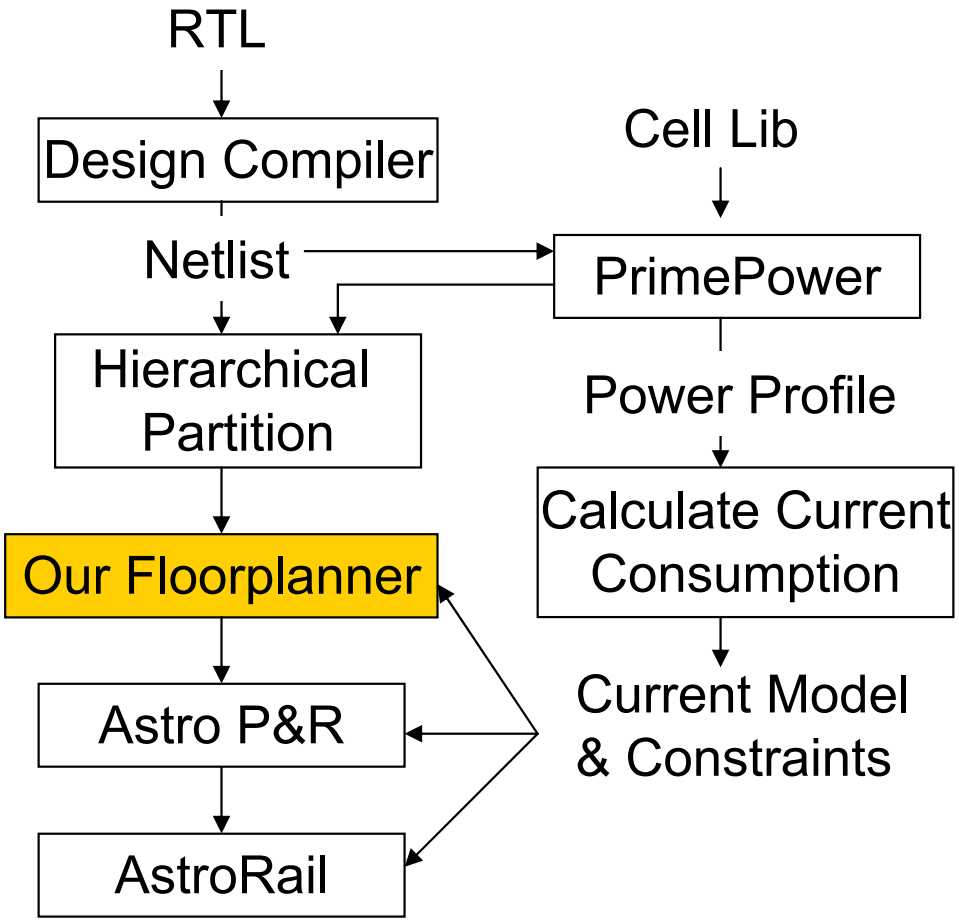
Implementation of the Design Flow

Data preparation

- Power profile
  - Power consumption data of the modules generated by PrimePower
- Hierarchical circuit partition
  - Organize the design into hard modules and soft modules according to the hierarchy

Post-layout verification

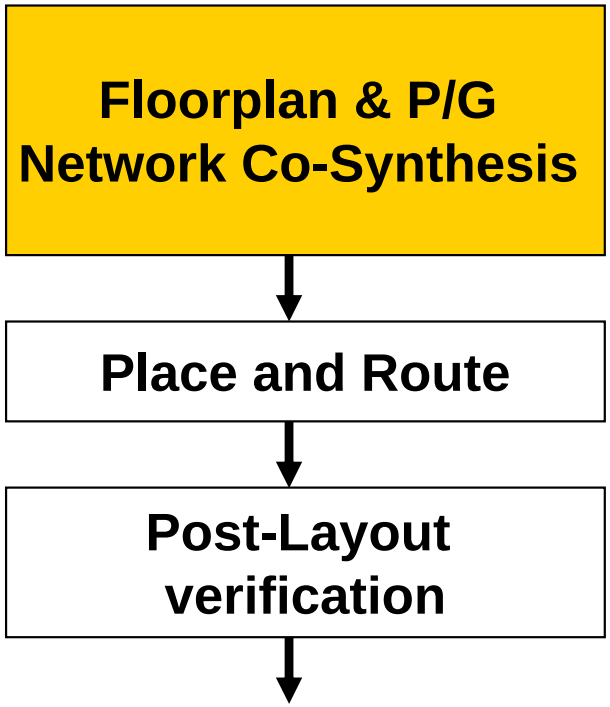
- AstroRail
  - Static cell-level P/G analysis



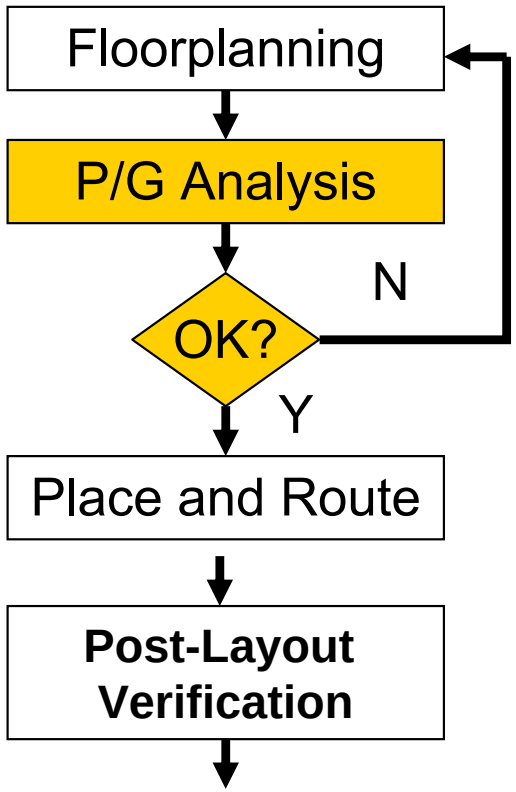
Flow Comparison

- Previous work: Only move the iterative fix to an earlier stage
- Our work: Further combine P/G planning into floorplanning
  - Thousands of floorplans are evaluated in a second
  - A very efficient, yet sufficiently accurate P/G network analysis method is needed

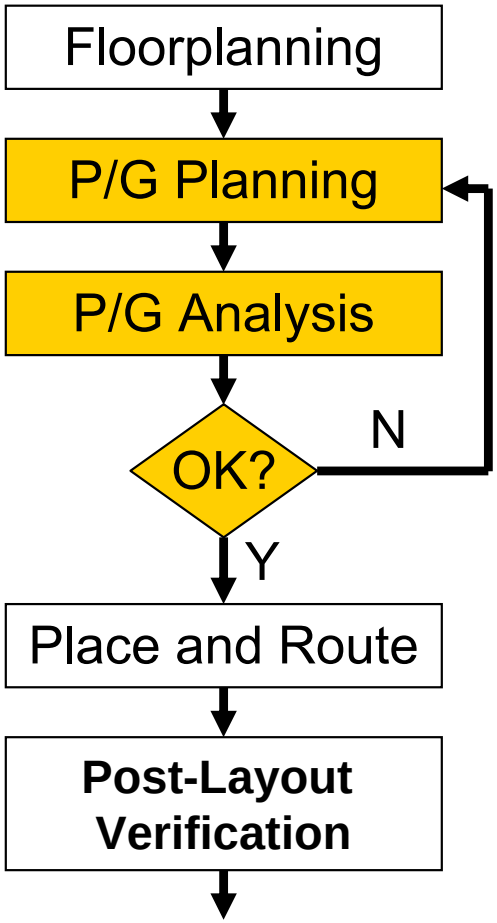
Ours



DAC-04



DAC-99



Outline

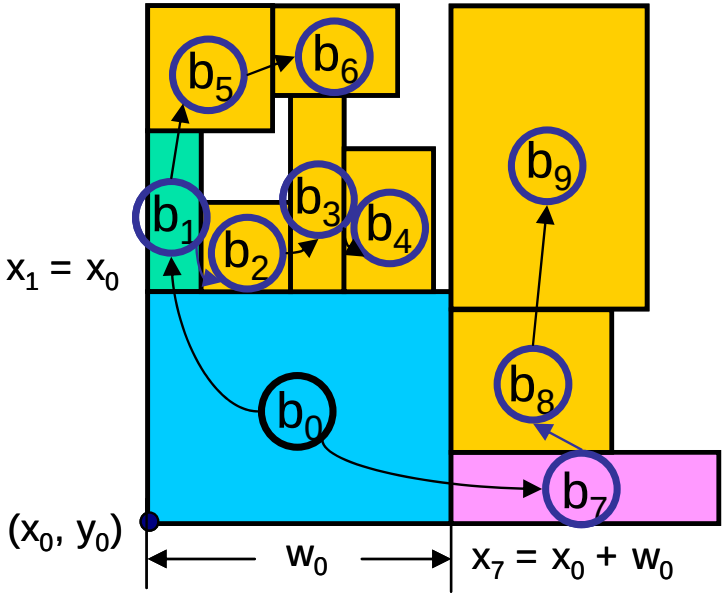
- Introduction
- Proposed Design Flow
- **Floorplan and P/G Network Co-Synthesis Algorithm**
- Experimental Results

Overview of the Co-Synthesis Algorithm

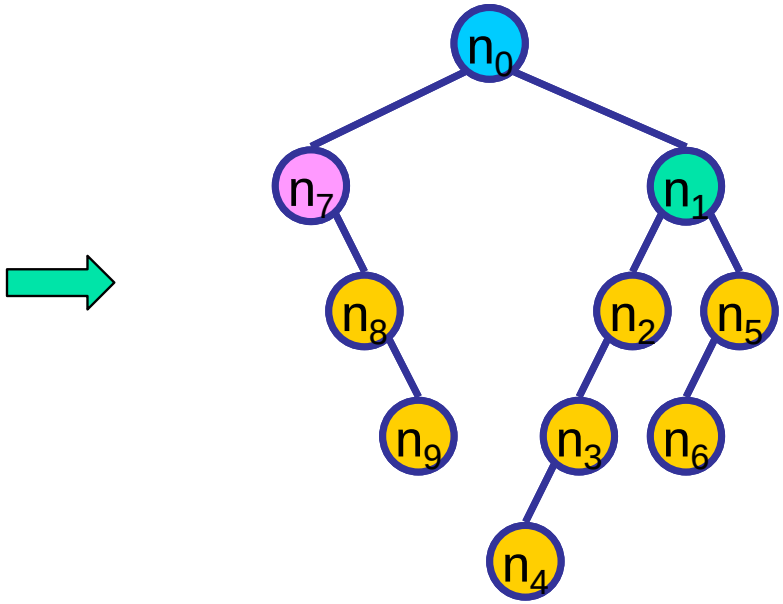
- B*-tree Floorplan Representation and Simulated Annealing (SA) Algorithm
- P/G network analysis
  - Global P/G Network Construction
  - P/G Network Modeling
  - P/G Network Evaluation
- Solution Space Reduction Technique (SSR)

B*-tree: Compacted Floorplan Representation

- Chang et al., "B*-tree: A new representation for non-slicing floorplans," DAC-2k.
  - Given a B*-tree, a legal floorplan can be obtained in amortized linear time
  - Root: The most left-bottom module
  - Left child: the lowest, adjacent block on the right ($x_j = x_i + w_i$)
  - Right child: the first block above, with the same x-coordinate ($x_j = x_i$)



A compacted floorplan



The corresponding B*-tree

Cost Function for Simulated Anealing

- Cost function:

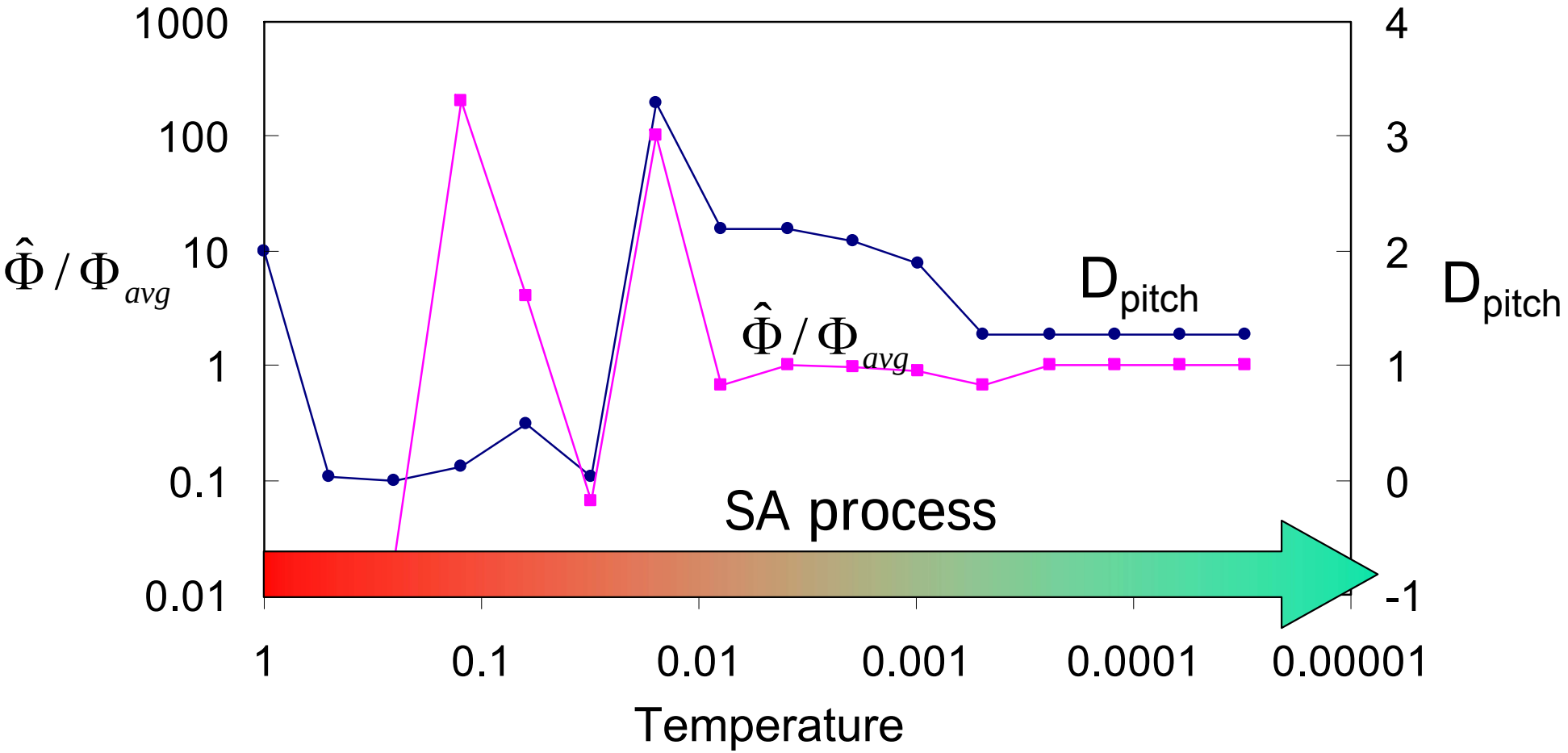
Wirelength Area P/G cost P/G Density

$$\Psi = \alpha \cdot W + \beta \cdot A + \gamma \cdot \Phi + \omega \cdot \frac{A}{D_{pitch}^2},$$

- W: wirelength
- A : area
- Φ: P/G network cost
- D_{pitch}: pitch of P/G network
  - Update by multiplying $\hat{\Phi} / \Phi_{avg}$
  - Φ_{avg}: Average P/G network cost at a temperature
  - $\hat{\Phi}$: $0 < \hat{\Phi} < 1$, a budget factor for adjusting the density of P/G networks
    Small $\hat{\Phi}$ for low P/G density and large one for high P/G density

Pitch Updating: An Example

- At the beginning of SA, D_{pitch} = 2 and $\hat{\Phi} = 0.02$
- During SA process, $D_{pitch} \Leftarrow \hat{\Phi} / \Phi_{avg} \times D_{pitch}$



$\hat{\Phi} / \Phi_{avg}$ converges to 1 while temperature cools down

P/G Network Cost

Φ: P/G network cost

EM cost **IR-drop cost**

$$\Phi = \theta \cdot \frac{|B_{em}|}{|B|} + (1-\theta) \cdot \frac{\sum_{\forall p_{vi} \in P_v} v_{p_{vi}}}{\sum_{\forall P_v \in P} V_{\lim, pi}}, \quad 0 < \theta < 1$$

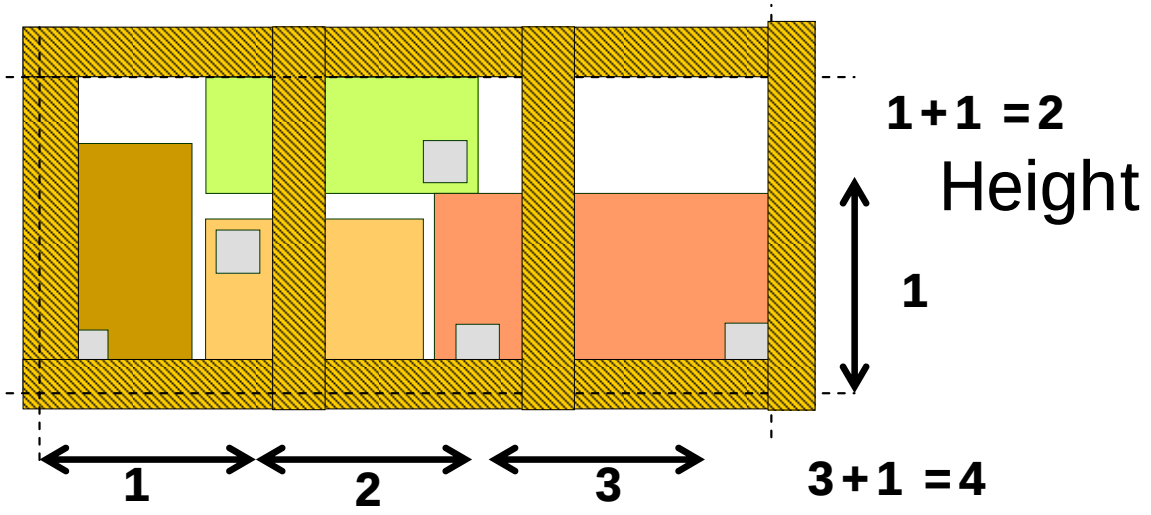
- B_{em}: set of branches violating electromigration constraints
- B : total branches of the P/G mesh
- v_{pvi}: amount of the violation at the pin p_{vi}
- P : set of all P/G pins
- P_v : set of violating P/G pins
- $V_{lim,pi}$: IR-drop constraint of the P/G pin p_i

P/G Network Construction

- For each floorplan, we construct a uniform global P/G network according to D_{pitch}
- The number of trunks is defined by

  round[width/D_{pitch}]+1 and round[height/D_{pitch}]+1

2X4 uniform P/G network is constructed

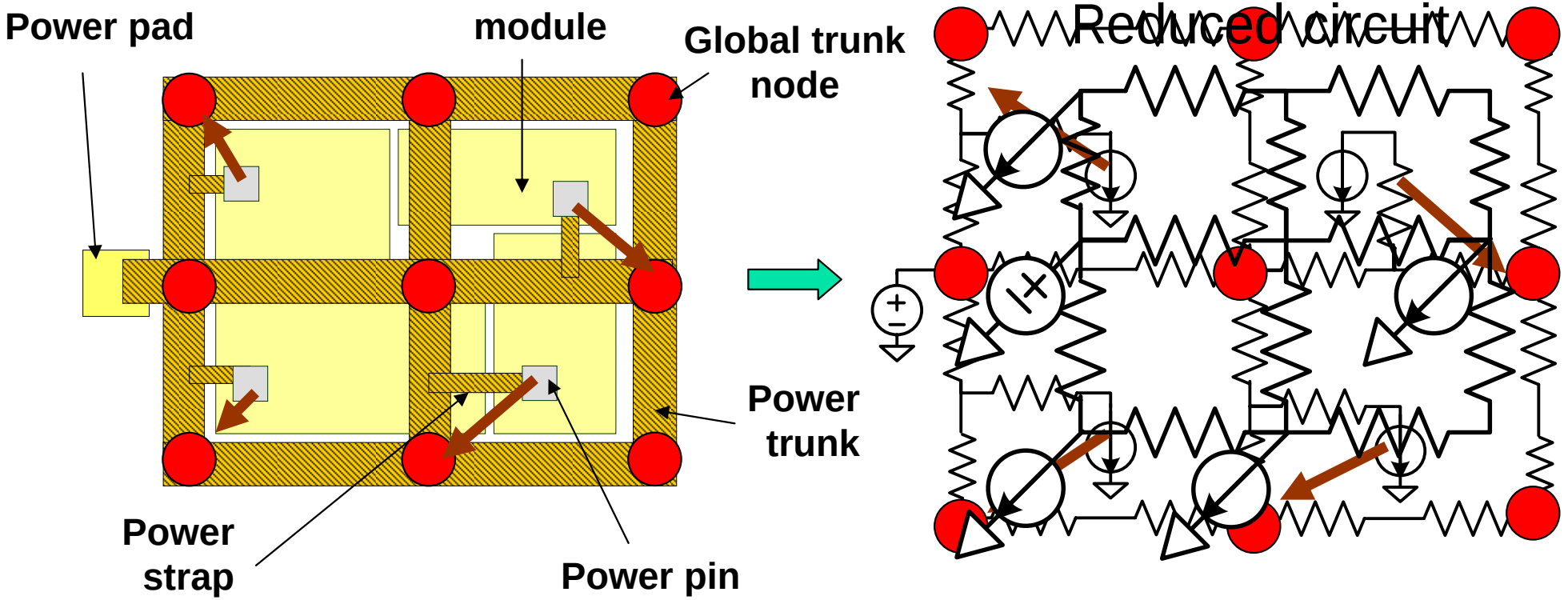


Calculate the P/G network dimention

P/G Network Modeling

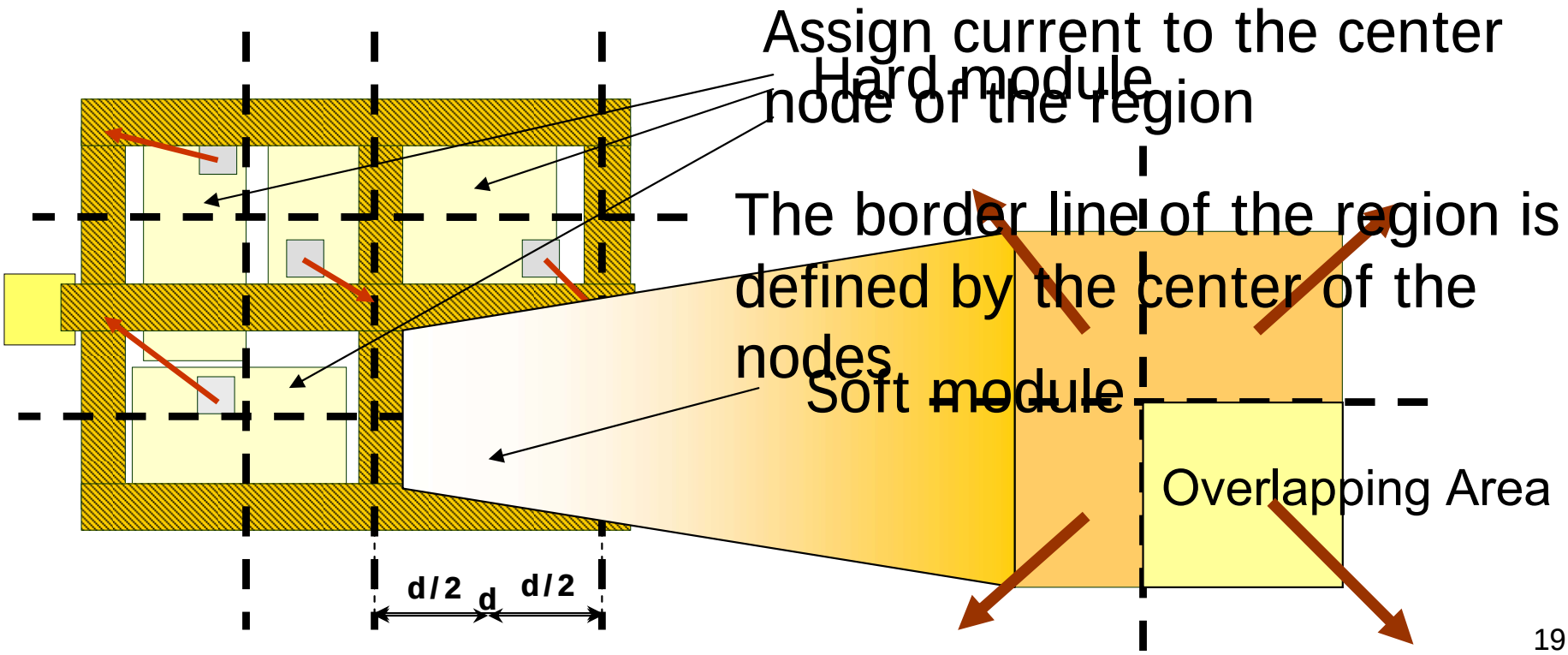
Apply static analysis for fast P/G network evaluation

- Use resistive P/G model
- Model a P/G pin as a current source
  - Current value: maximum current drawn from a P/G pin
- Reduce circuit size
  - Connect each current source to the nearest global trunk node



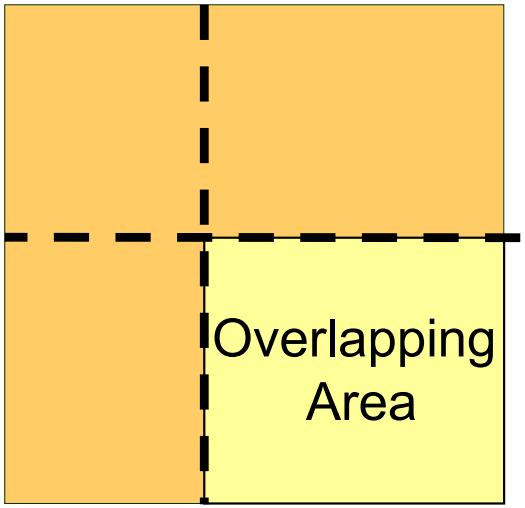
Macro Current Modeling

- Divide the floorplan into regions
- For Hard macros
  - Connect each P/G pins to the nearest node (center of the region)
- For Soft macros
  - Collect the largest current drawn by standard cells in the overlapping area of the region and the soft macro

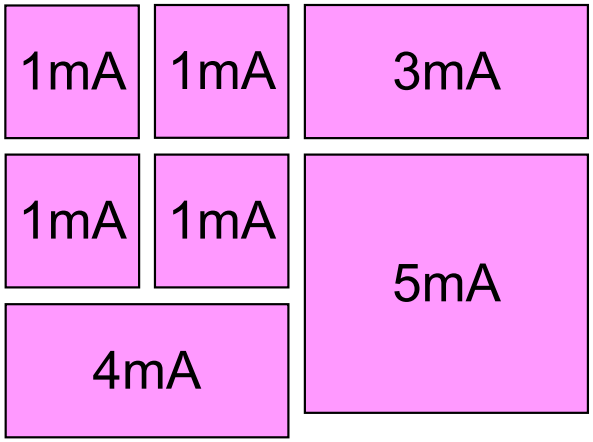


Soft Macro Modeling

- Derive the largest current drawn by standard cells of the overlapping area
  - Maximize the current of the overlapping area
  - Constraint: total stdcell area < the overlapping area
  - The problem is known as 0-1 Knapsack Problem (NP-complete)
- Approximate it by Fractional Knapsack Algorithm
  - Assume standard cells can be broken into arbitrary smaller pieces
  - Rank cells by current to area ratio
  - Apply a greedy algorithm (complexity O(n lg n))



Standard cells of the soft module



Evaluation of P/G Network

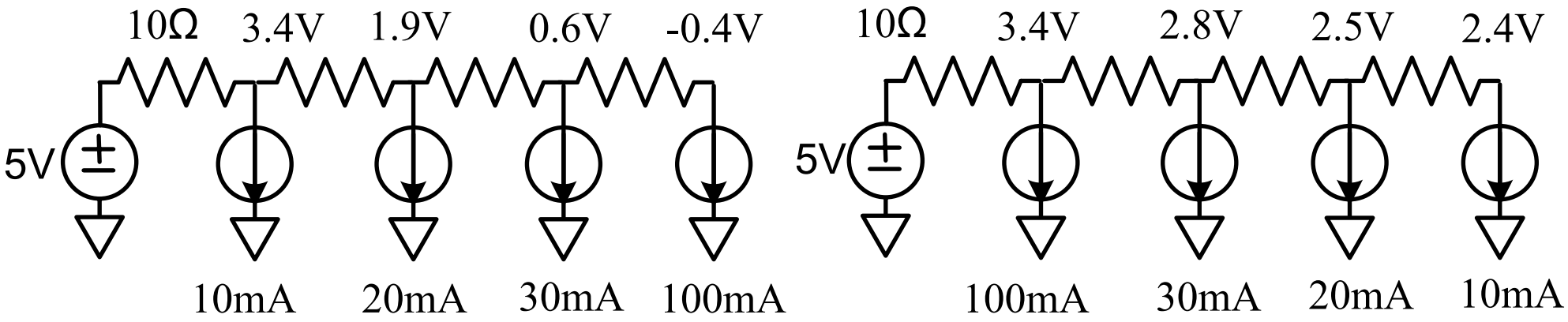
- The static analysis of a P/G network is formulated into the following modified nodal analysis (MNA) formula:

$$\mathbf{Gx = i}$$

  - **G**: conductance matrix (sparse positive definite)
  - **x** : vector of node voltages
  - **i** : vector of current and voltage sources
  - Dimensions of **G**, **i** and **x** are equal to the number of nodes in the P/G network
- Solve the linear equation
  - Apply Preconditioned Conjugated Gradient (PCG) method
  - The time complexity is linear

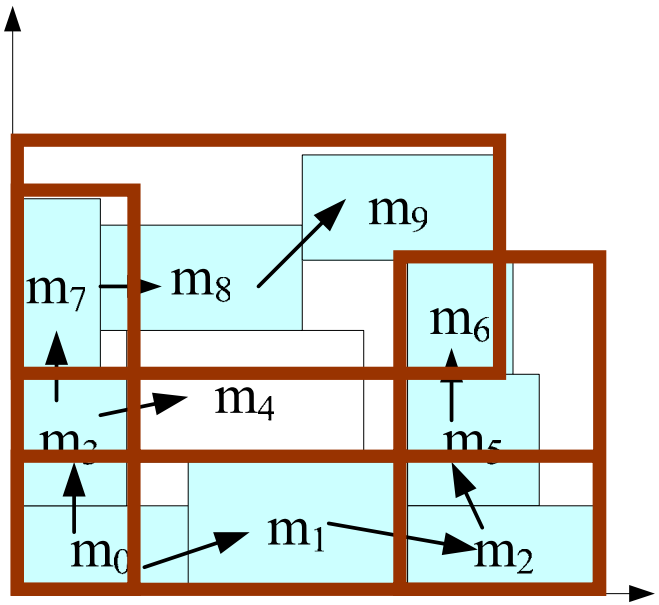
Solution Space Reduction

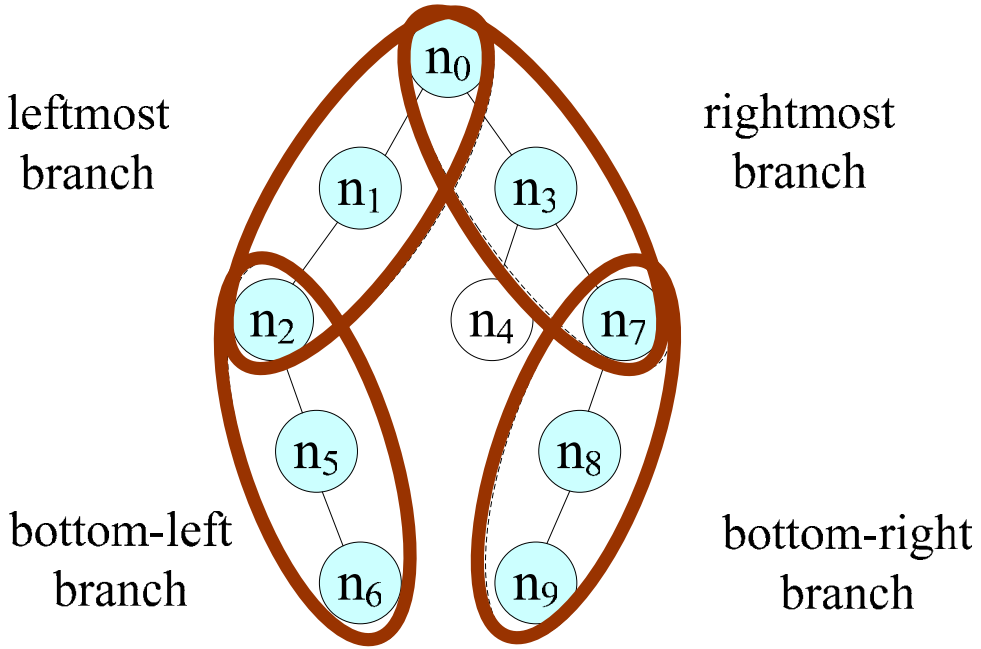
- The IR-drop of a P/G pin is proportional to the effective resistance between the P/G pin and the P/G pad
  - The closer the P/G pin is placed to the P/G pad, the smaller the IR-drop
- A technique to reduce the solution space
  - Place the modules consuming larger current (power-hungry modules) near the boundary of the floorplan
  - Place power pads near them



B*tree Boundary Properties

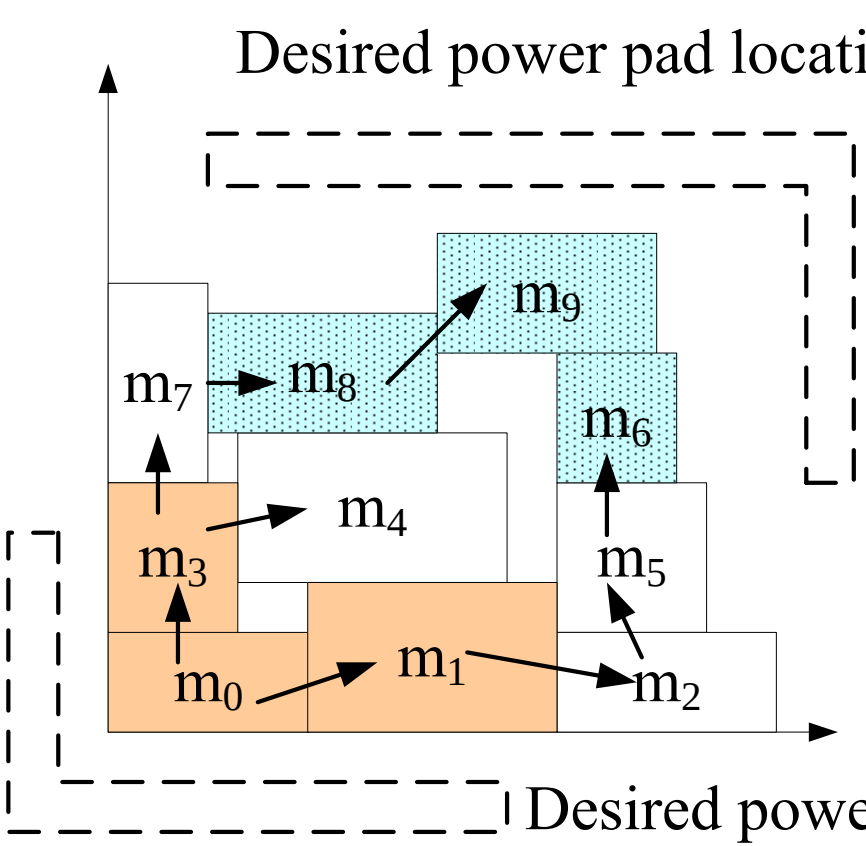
- Bottom-boundary condition
  - Bottom boundary modules are related to the leftmost branch
- Left-boundary condition
  - Left boundary modules are related to the rightmost branch
- Right-boundary condition
  - Right boundary module are related to the bottom-left branch
- Top-boundary condition
  - Top boundary modules are related to the bottom-right branch

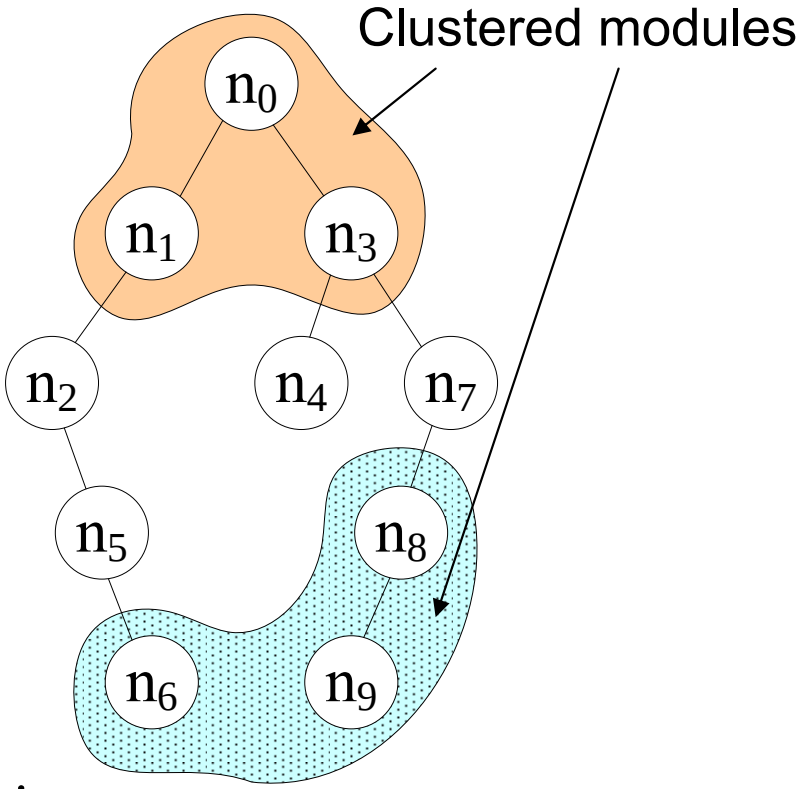




Power-Hungry Modules Handling

- Power-Hungry Modules
  - Are clustered and restricted to satisfy the boundary properties during B*-tree perturbation
  - P/G pads are placed near these modules





Outline

- Introduction
- Proposed Design Flow
- Floorplan and P/G Network
  Co-Synthesis Algorithm
- **Experimental Results**

Experimental Settings

- Implementation
  - Use GNU C++
- Platform
  - On Sun Blade 2000 with single 1GHz CPU and 8G memory
- OpenRISC
  - Open source 32bit RISC micro processor (OPENCORE)
  - UMC 0.18 technology
  - Compare to the Astro design flow with IR-drop driven placement (manually and iteratively fix P/G network faults)
- MCNC benchmark
  - TSMC 0.25 technology
  - Given large power consumption and small power budget (low P/G network density and only a pair of P/G pads)
    - In order to test the robustness of our floorplanner

Results on OpenRISC1200

Improvements in runtime and the max IR-drop

OpenRISC1200	*Astro Flow	*Astro w/ IR-drop Driven Placement	Our Flow	Ours vs. Astro w/IR-drop
Die Area (mm^2)	3.86	3.86	3.33	15.9%
Utilization (%)	62	62	72	13.9%
Wirelength (µm)	1655463	1539125	1540172	-0.1%
Avg. Delay (ns)	8.62	8.54	8.55	-0.1%
Max IR-drop (mv)	80.18	78.20	55.14	41.8
CPU Runtime (s)	505	346	135	2.56X
Iterations	4	3	1	-

Small overhead

Better IR-drop

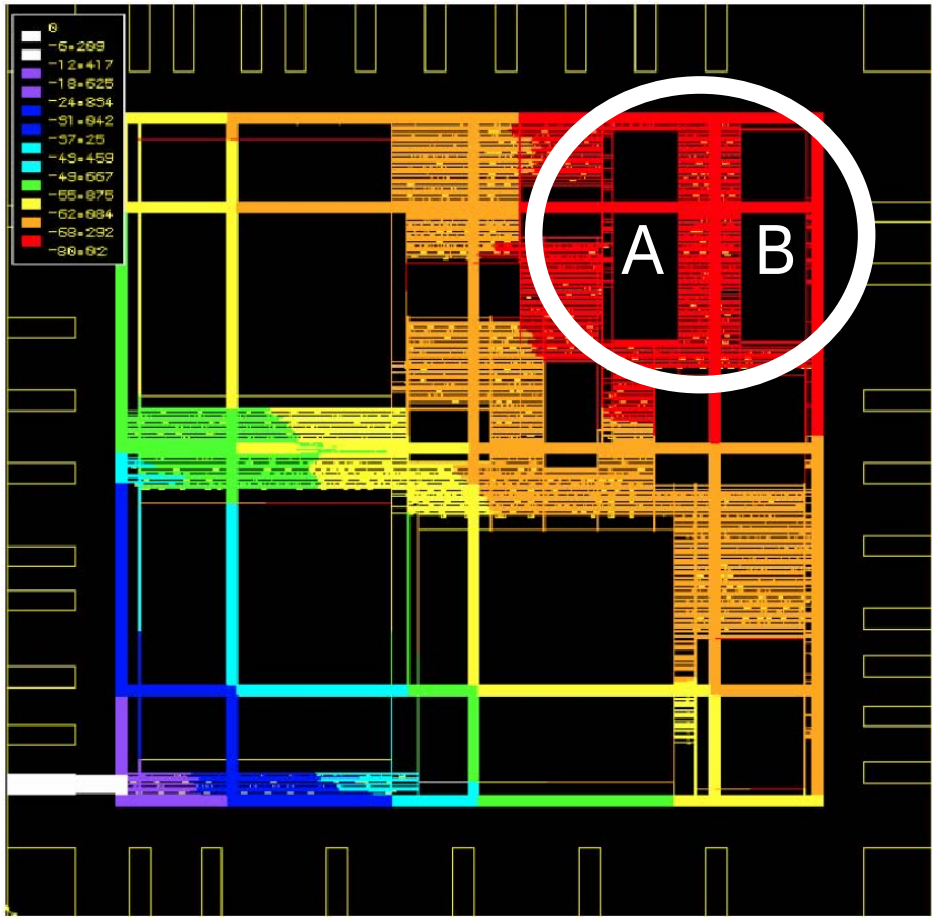
Speed improvement

*Need iterative and manual P/G network fix

Resulting Voltage Map

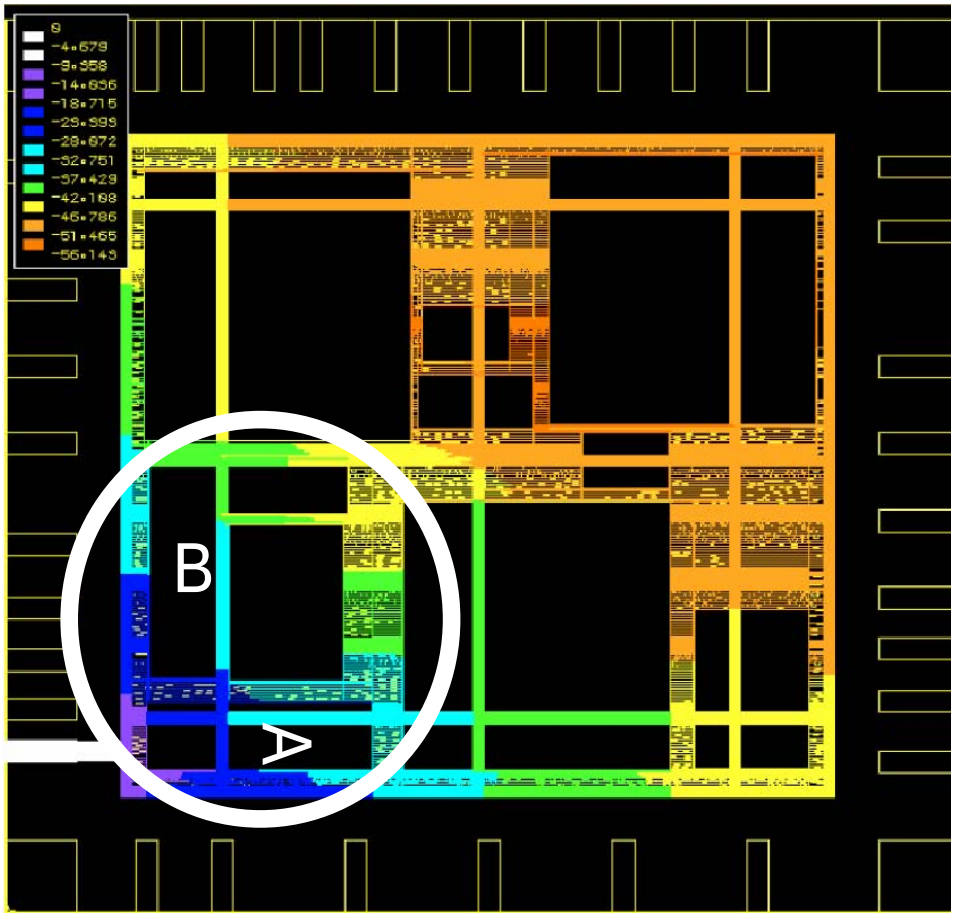
Astro design flow

Power-hungry blocks (register file A&B) are placed far from the power pad



Our design flow

Power-hungry blocks are placed beside the power pad



Results on MCNC Benchmark

Compared with plain B*-tree floorplanner, our floorplanner solves all the violations with small overhead

Circuit	Plain B*-tree Floorplanner				Ours Without SSR			
	WL	Area	#Vio	CPU	WL	Area	#Vio.	CPU
apte	435.5	48.21	6	1.1	440.4	49.8	0	165.2
xerox	387.6	20.42	39	3.3	401.5	21.3	0	122.3
Hp	155.5	9.56	38	3.2	187.1	11.2	0	58.2
ami33	58.4	1.31	99	8.8	69.0	1.4	0	43.4
ami49	864.6	39.86	195	42.2	832.8	39.8	0	1412
comp.	0.98	0.97	195	0.033	1	1	0	1

Results of Solution Space Reduction

The solution space reduction technique speeds up floorplanning 3x with similar quality

Circuit	Ours With SSR				Ours without SSR			
	WL	Area	#Vio	CPU	WL	Area	#Vio	CPU
apte	452.1	48.8	0	43.2	440.4	49.8	0	165.2
xerox	410.2	22.4	0	47.3	401.5	21.3	0	122.3
Hp	189.5	11.7	0	24.0	187.1	11.2	0	58.2
ami33	73.2	1.2	0	20.2	69.0	1.4	0	43.4
ami49	779.9	44.2	0	450.0	832.8	39.8	0	1412
comp.	0.99	1.04	0	0.32	1	1	0	1

Conclusions

- Have proposed a practical design flow compatible with commercial CAD design flow
- Have proposed an algorithm and a modeling technique to make floorplan and P/G network co-synthesis possible
- Have shown the solution space reduction method to speed up the co-synthesis algorithm
- Have shown the efficiency and effectiveness of the proposed power integrity driven design flow and applied on a real design

Thank You

Simulate Annealing Process

- Non-zero probability for up-hill climbing:

$$p = \min\left(1, e^{-\frac{\Delta\Psi}{T}}\right)$$

- Perturbations (neighboring solutions)
  - Op1: Rotate a block
  - Op2: Move a node/block to another place
  - Op3: Swap two nodes/blocks
  - Op4: Resize a soft block
- The cost function Ψ is based on the floorplan cost and P/G network cost
- T is decreased every n cycles, where n is proportional to the number of blocks