

Fast Placement Optimization of Power Supply Pads



Yu Zhong and Martin D. F. Wong

Department of Electrical and Computer Engineering

University of Illinois at Urbana-Champaign

Outline

- Introduction
- Previous methods
- Incremental power grid analysis
- Power pad placement algorithm
- Experimental Results
- Conclusions

Introduction

- Power grid networks are designed to provide adequate voltages and currents.
- Supply voltages are assumed to be constant across the chip to ensure reliable performance.
- Voltage values on a power grid fluctuate due to
 - Increased resistances of metal lines
 - High current levels
 - Package pin inductances
- The IR drop on power grids are getting significant
 - Increasing complexity of VLSI circuits
 - Increasing power (current) consumption
 - Decreasing supply voltage
 - Reduced noise margin and increased gate delay

Introduction

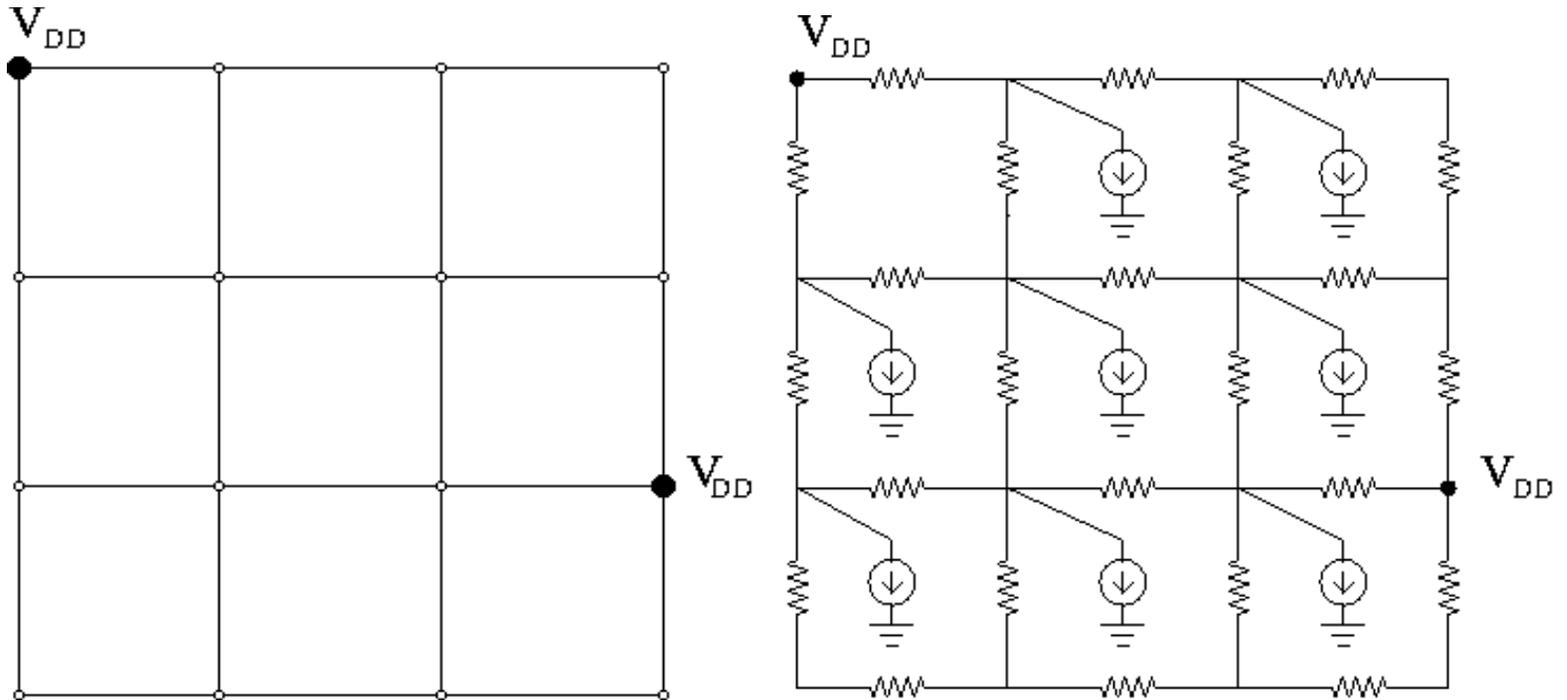
- Difficulties in the analysis of power grids:
 - Large scale simulation
 - Usually millions of elements in power grid
 - Even DC analysis is difficult
 - SPICE-level accuracy simulation is required
 - Runtime is slow
 - Memory inefficiency
 - e.g. 1 million node \rightarrow 1 trillion elements in matrix
 - Short of memory
- The design of power grids becomes even more difficult due to the bottleneck of simulation.

Introduction

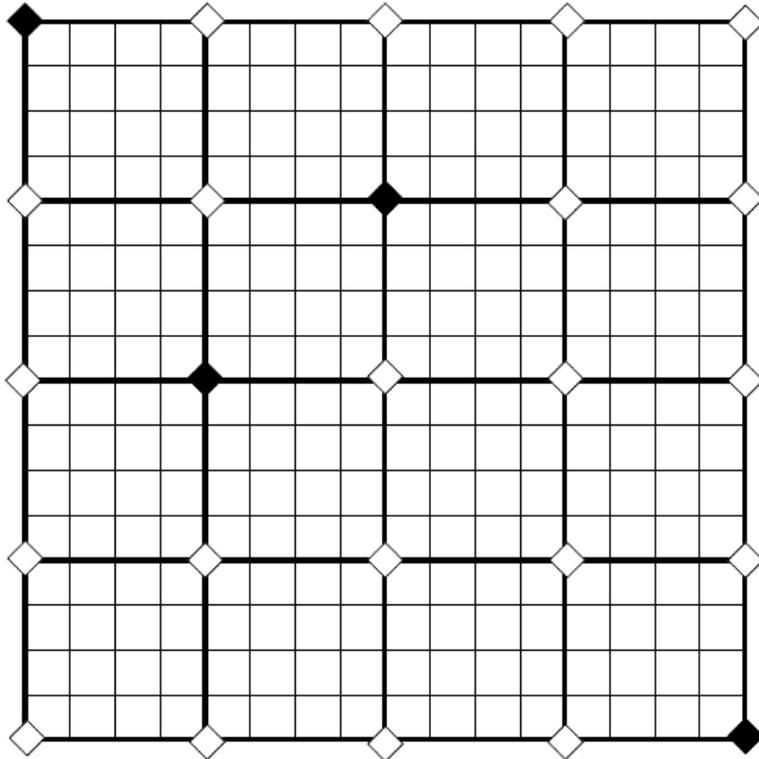
- Reduce IR drop in power grid design
 - Widen the power routes
 - Increasing the width decreases the resistance and the IR drop
 - Not always possible due to the constraints in routing area
 - Maximize the use of decoupling capacitance
 - De-caps can deliver additional current needed by power grids
 - De-caps are scattered throughout the power grid
 - Limited by the available silicon area
 - Optimize the placement of the power I/O pins
 - Power pad placement is a difficult problem, since the number of candidate pad locations can be extremely large.
 - Wire-bond package: all pads are located on peripheral power ring
 - Flip-chip package: C4 pads can be at various points within the chip.
 - Advantage of C4: ability to place power pads anywhere on the die to reduce the IR drop, as opposed to just the periphery

Model of Power Grid

- DC model for a power grid.



Problem Formulation



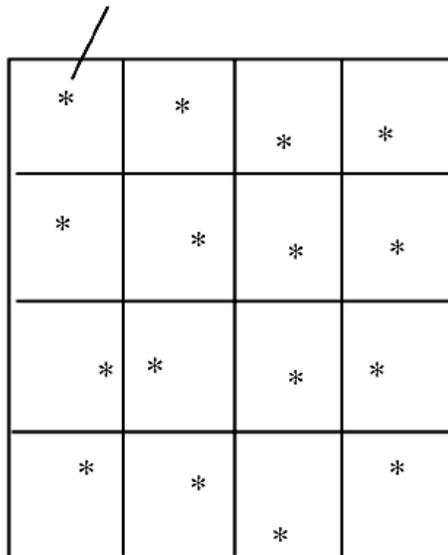
Given a fixed number of power pads and a set of pad candidate locations, find the optimal positions for the power pads, such that the worst IR drop is minimized.

- ◇ V_{dd} Candidate Location
- ◆ V_{dd} Pad

Previous Method

- M. Zhao et. al. “Optimal placement of power supply pads and pins” DAC 2004
- Generate a macro-model of the power network with the candidate pad locations and the observation nodes as ports of the model, construct a mixed integer linear programming (MILP) program, and solve the MILP program to get the pad locations.

The worst *IR*-drop node in the region

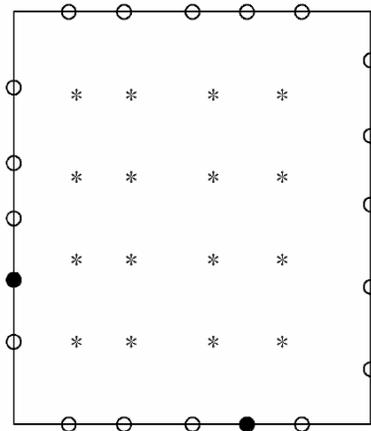


*	*	*	*
*	*	*	*
*	*	*	*
*	*	*	*

- The observation nodes (OBS) are selected to present the worst IR-drop in each region.
- Example: 4.9 M-nodes circuit ~ 132 OBS
- Disadvantage: the worst voltage locations can shift with change of pad locations, so the calculation of worst IR drop is not accurate.
- If increase # of OBS, slow down macro-modeling computation of power grid

Previous Method

- MILP is very expensive for large number of integer variables. # of variables in MILP of this problem is equal to # pad candidates (PC).
- Example: 30 power pads, consider only 78 pad candidates for a 4.9M-node circuit



- pad candidate
- * observation node
- the selected pad

For a circuit with 1 M nodes,

- wire-bond package: all possible pad locations on the periphery, ~ 1 K pad candidates
- flip-chip package: possible pad locations (C4 pads) all over the chip, ~ 1 M pad candidates

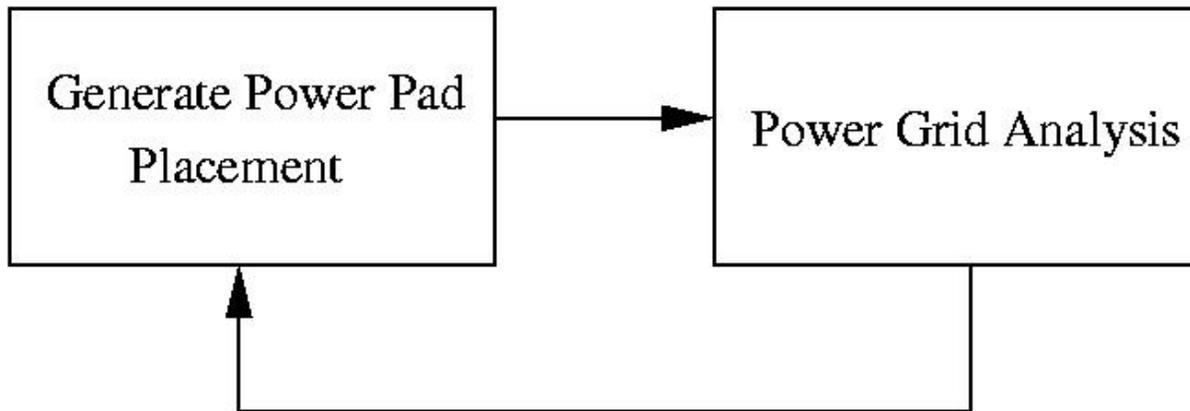
Therefore, MILP is not efficient and prohibitive for large # of pad candidates.

Our Method

- A fast localized node-based iterative method to compute IR drop
 - Key enabler for efficient optimization
 - Localized efficient computation and its time complexity is independent of the power grid size
- Develop a simulated annealing algorithm to minimize the IR drop
- Good scalability for large power grids with large numbers of pad candidate locations.
- Prohibitive to use any existing power grid analysis method with simulated annealing
 - The runtime of a 16-million-node power grid analysis: ~ 30 minutes
 - Total number of movements in simulated annealing: ~ 10,000
 - Total runtime:~200 days

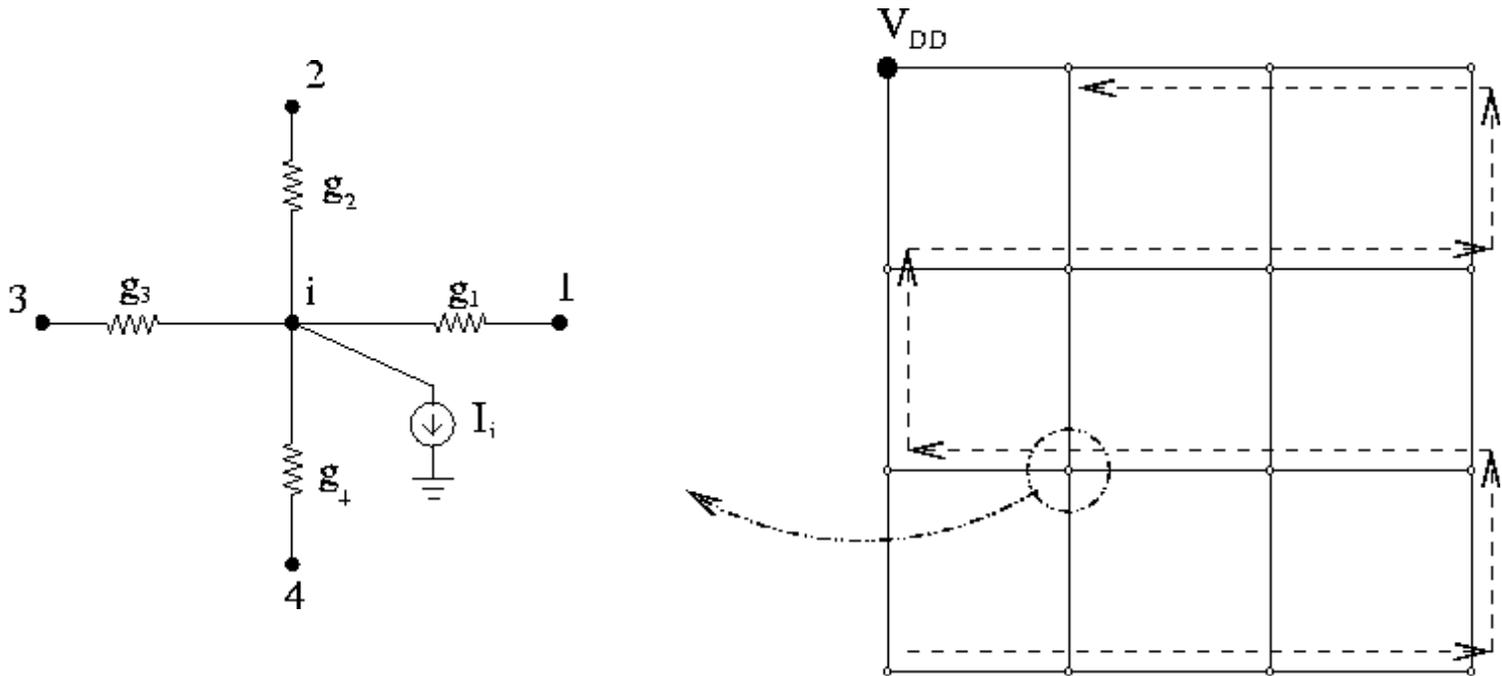
Our Method

- In each step of simulated annealing, we move one V_{DD} pad to generate a new power pad placement.
- We use a localized node-based iterative method to update the voltages on power grid efficiently.
- **Key:** Fast power grid analysis after each pad movement



Search method: simulated annealing

Node-based Method



Apply Kirchoff's current law at node i

$$V_i = \frac{(g_1 V_1 + g_2 V_2 + g_3 V_3 + g_4 V_4 - I_i)}{(g_1 + g_2 + g_3 + g_4)}$$

Node-based Method

$$\bar{V}_i^{(k+1)} = \left(\sum_{j<i} g_{ij} V_j^{(k+1)} + \sum_{j>i} g_{ij} V_j^{(k)} - I_i \right) / \sum_j g_{ij}$$

Basic node-based method:

$$V_i^{(k+1)} = \bar{V}_i^{(k+1)}$$

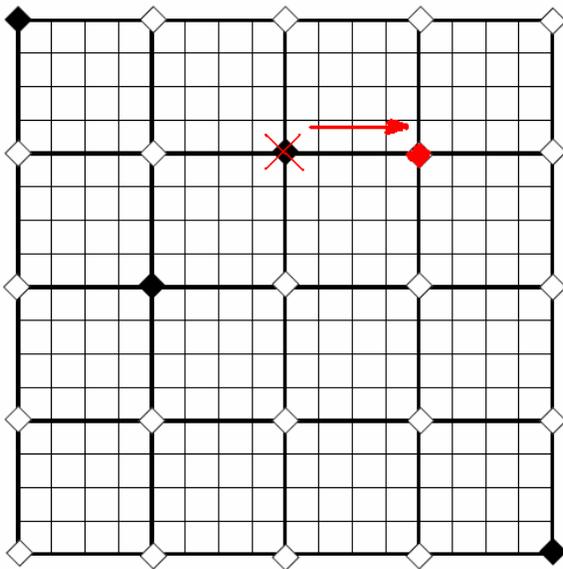
Improved node-based method:

$$V_i^{(k+1)} = \omega \bar{V}_i^{(k+1)} + (1 - \omega) V_i^{(k)}$$

We can improve the convergence rate of the generic node-based method by choosing an appropriate extrapolation factor ω .

How to Generate Next Pad Placement?

- Assume initial voltages on power grid with initial placement of power pads are computed.
- Each movement in simulated annealing generates the next pad placement.
- Basic operation: moving a randomly picked power pad to another available candidate location.

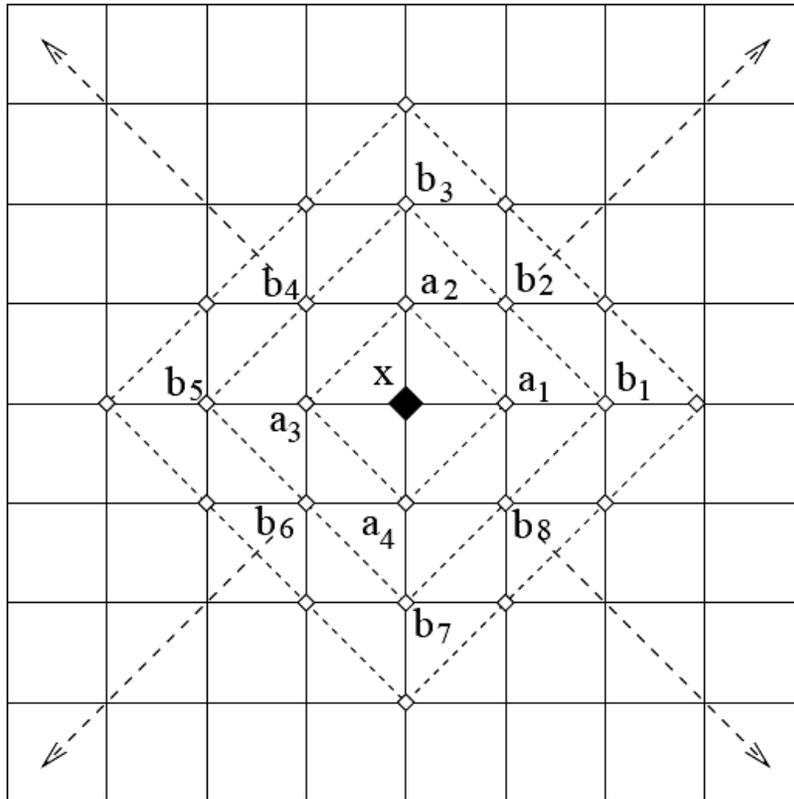


The movement of a power pad can be decomposed into:

- Delete a V_{DD} pad from its old location
- Add it to a new location

Fast Voltage Update

- How to compute the voltage change if we delete/add a V_{DD} pad from node x ?

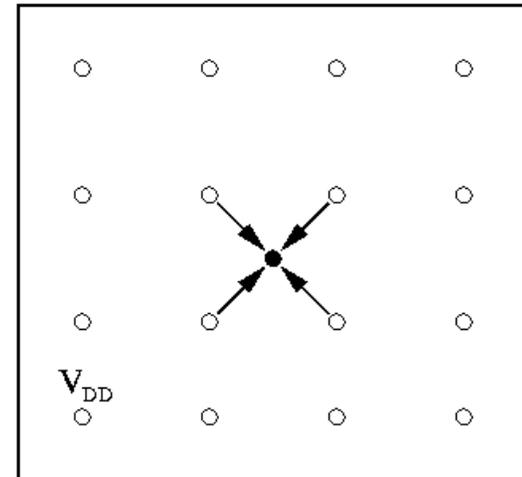


Apply breadth first traversal to visit the neighbors of node x .

Use improved node-based method to iteratively update those node voltages from the source node x in the order of wave propagation.

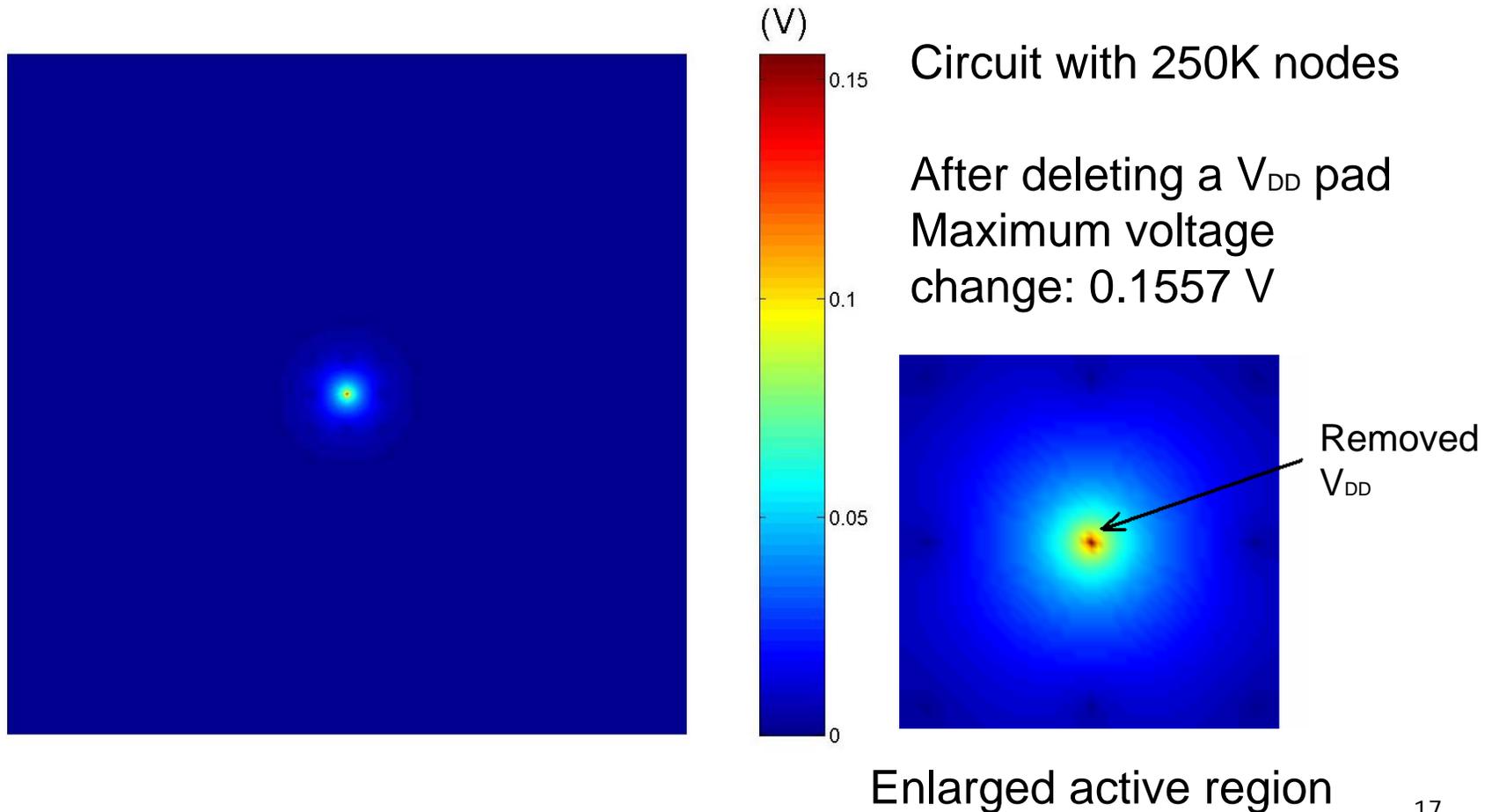
Fast Voltage Update

- In simulated annealing, impossible to use improved node-based method to iteratively update the voltages of all the nodes on power grid :
 - The runtime of a 16-million-node power grid analysis: ~ 30 minutes
 - Total number of movements in simulated annealing: ~ 10,000
 - Total runtime: ~200 days
- Effect of deleting/adding a V_{DD} pad is local.
- The voltage of a node on power grid is mostly determined by the neighboring V_{DD} pads. Those V_{DD} pads far away have insignificant influence.



Fast Voltage Update

- Example of local effect of deleting a V_{DD} pad.



Fast Voltage Update

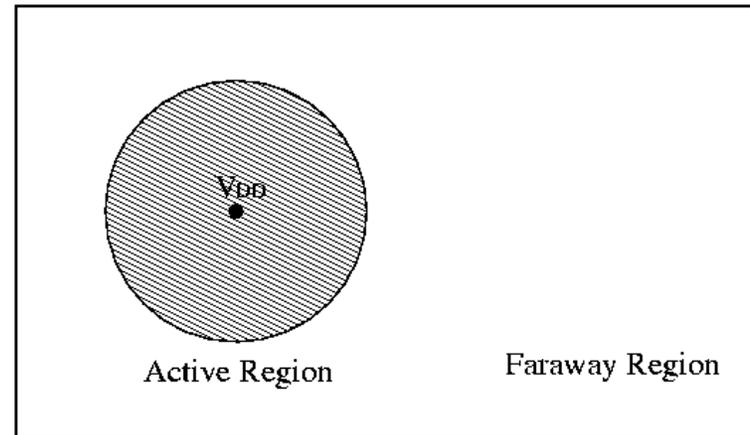
- Use localized node-based method to update voltages after deleting/adding a VDD pad.
 - Only iteratively compute the active region of the change of V_{DD} .
 - Ignore the faraway region, which is not influence by the change of V_{DD} .
 - The runtime of each movement is independent of the size of the power gird. So it is efficient and scalable for large problems.
- We do not limit the size of active region to a fixed number, since it may change under different conditions.

Instead, use error bound

$$\delta_{\text{active}} \ll 1$$

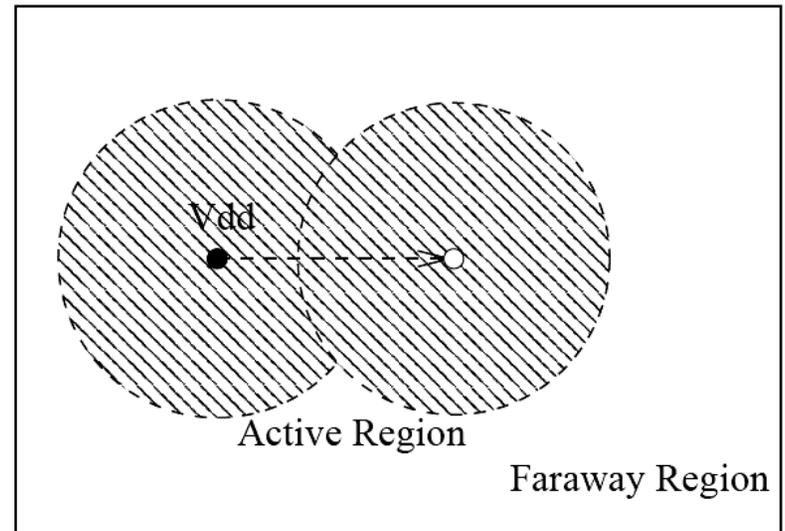
to check if the wave propagation vanish or not.

In one iteration, if the voltage change at some node is smaller than δ_{active} , we terminate this iteration and start a new iteration from the wave source VDD again.



Fast Voltage Update

- Localized voltage update after moving one V_{DD} from node x to node y :
 - There are two point sources x and y in the breadth first traversal of wave propagation.
 - If the old and new locations of V_{DD} are close to each other, the active region of deleting and adding a power pad may have some overlapping region. So the number of nodes which need to compute will become even smaller.



Fast Voltage Update

Voltage Recomputation after One Pad Movement

```
//  $x, y$ : The old and new locations of power pad
//  $V^{(0)}$ : The initial voltage values for all nodes
//  $\epsilon \ll 1$ : error bound used to control number of iterations
//  $\delta_{active} \ll 1$ : error bound used to determine active region
   $k = 0$  // iteration number
  repeat //begin iteration
     $k = k + 1$ 
     $nodeQ =$ empty first-in first-out queue
     $nodeQ = nodeQ \cup x \cup y$ 
    repeat // begin breadth first traversal
       $i = nodeQ.extractFirst()$ 
      Compute  $V_i^{(k)}$  using Equation (2)
      if  $|V_i^{(k)} - V_i^{(k-1)}| > \delta_{active}$ 
        for each neighbor node  $j \in N_i$  and  $j \notin nodeQ$ 
           $nodeQ = nodeQ.pushback(j)$ 
    until  $nodeQ$  is empty traversal
  until  $\max_i |V_i^{(k)} - V_i^{(k-1)}| < \epsilon$ 
```

Simulated Annealing

- Simulated annealing algorithm to optimize the power pad placements with the objective of minimizing voltage drops
 - The temperature schedule $T_m = r \times T_{m-1}, r = 0.85$
 - At each temperature, a number of power pad movements are attempted.
 - The goal of design is minimizing the worst voltage drop in the power grid and reducing the standard deviation of voltage drops so that the voltage values on the whole power grid will become more uniform.
 - Cost function: $C(\mathbf{V}) = \alpha V_{\text{worst}}^2 + \beta \sum V_i^2 / N$

where V_i is the voltage at node i , N is the number of nodes. Choosing V_i^2 instead of V_i will give larger penalties to large IR drops.

α and β are constants to make a tradeoff between the worst voltage drop and the deviation in the voltage drops.

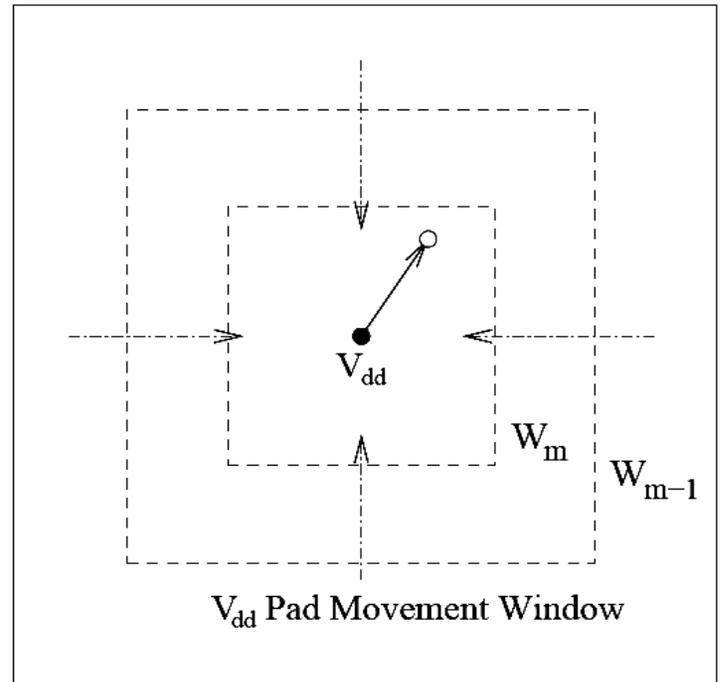
Simulated Annealing

- Place all power pads uniformly on the power grid.
- pick a V_{DD} pad randomly and move it to another empty candidate location.

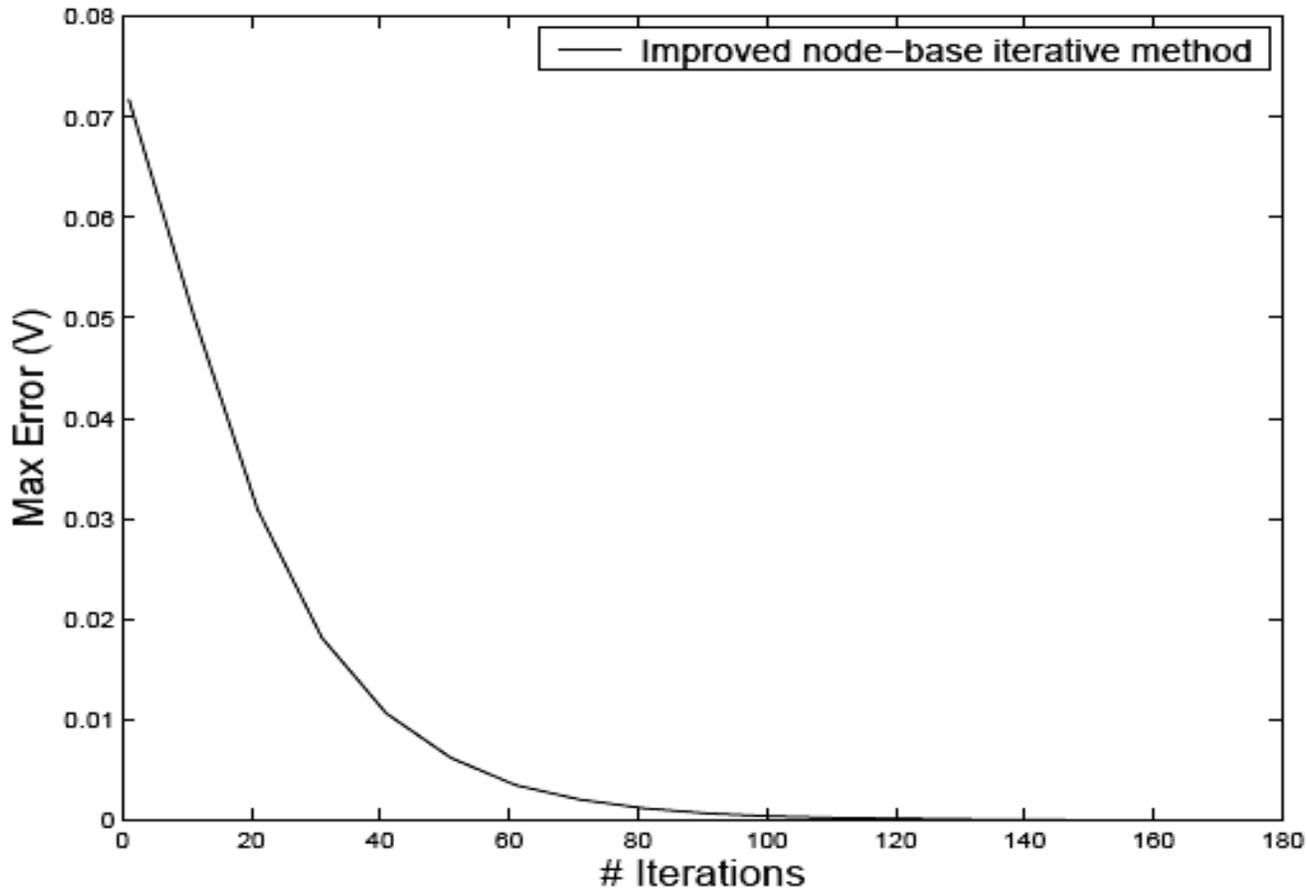
- $\begin{cases} \Delta C \leq 0 & \text{accept the move} \\ \Delta C > 0 & \text{accept the move with} \\ & \text{probability } e^{-\frac{\Delta C}{T_m}} \end{cases}$

- For a V_{DD} to be moved, we randomly choose its new location within a window

- As the temperature decreases, $T_m = r \times T_{m-1}$ shrink this window gradually with $W_m = r \times W_{m-1}$, where r is the temperature scaling factor and W_m is the window size.



Simulated Annealing



Circuit C1 with
251K nodes.

Tradeoff between runtime and maximum error in the Improved node-based method

Simulated Annealing

- Improve the efficiency of the simulated annealing algorithm
 - Tradeoff between runtime and maximum error in the improved node-based method
 - Introducing some tolerable error at high temperatures
 - Save runtime at high temperature
 - At high temperature, fast estimation of change of cost is more important than accurate voltage computation
 - As temperature decreases, we improve the accuracy of the node-based iterative method gradually.
 - At low temperatures, we make accurate computation without error.
 - Error tolerance $\varepsilon_m = r \times \varepsilon_{m-1}$, where r is the temperature scale factor

Experimental Results

Circuits	#nodes	#pads	#PCLs	Initial Design		After Pad Optimization		
				MaxV(V)	σ (V)	MaxV(V)	σ (V)	time (m:s)
P1	10K	25	441	0.161	0.086	0.068	0.009	0:09
P2	251K	121	10201	0.191	0.098	0.097	0.013	1:13
P3	1M	121	40401	0.254	0.114	0.148	0.018	5:06
P4	4M	441	161604	0.234	0.110	0.121	0.016	21:37
P5	16M	441	646416	0.398	0.134	0.196	0.024	72:50

#pads: number of V_{DD} pads

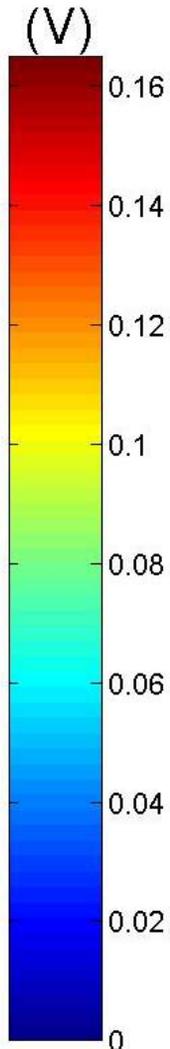
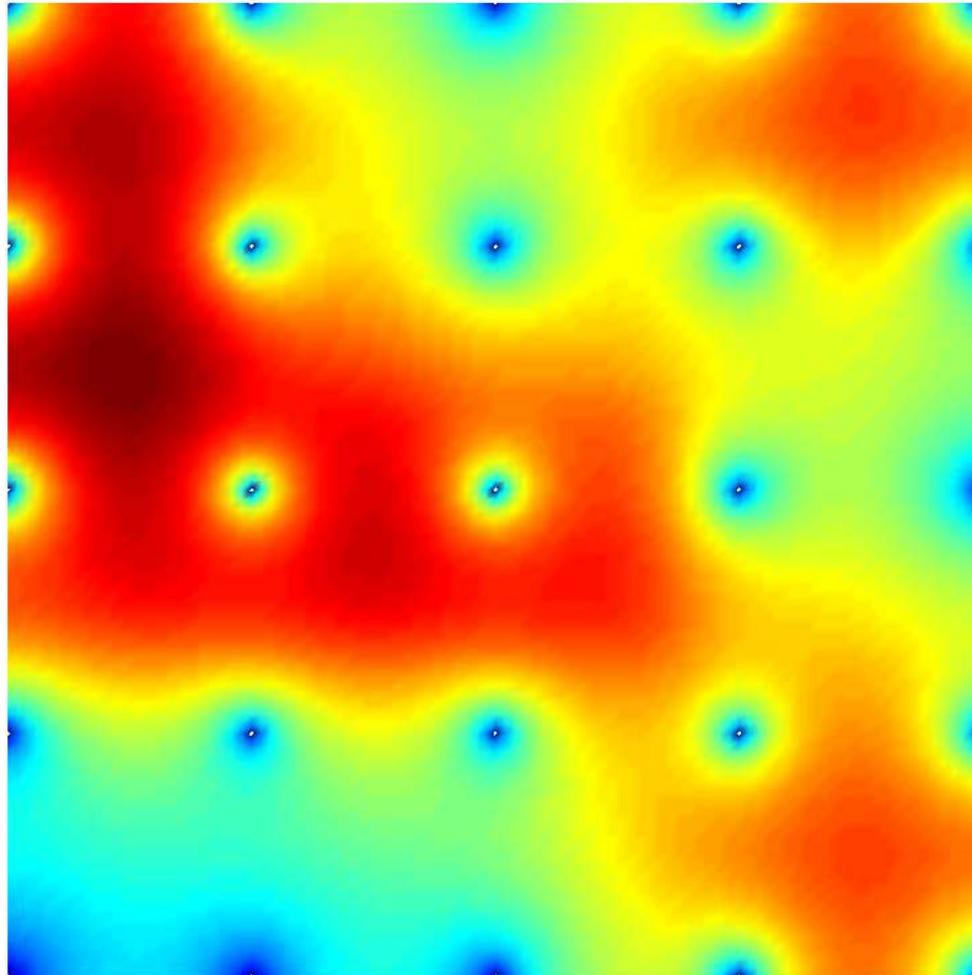
#PCLs: number of pad candidate locations

MaxV: worst voltage drop

σ : standard deviation of the voltage drops

time: runtime of the placement optimization

Experimental Results

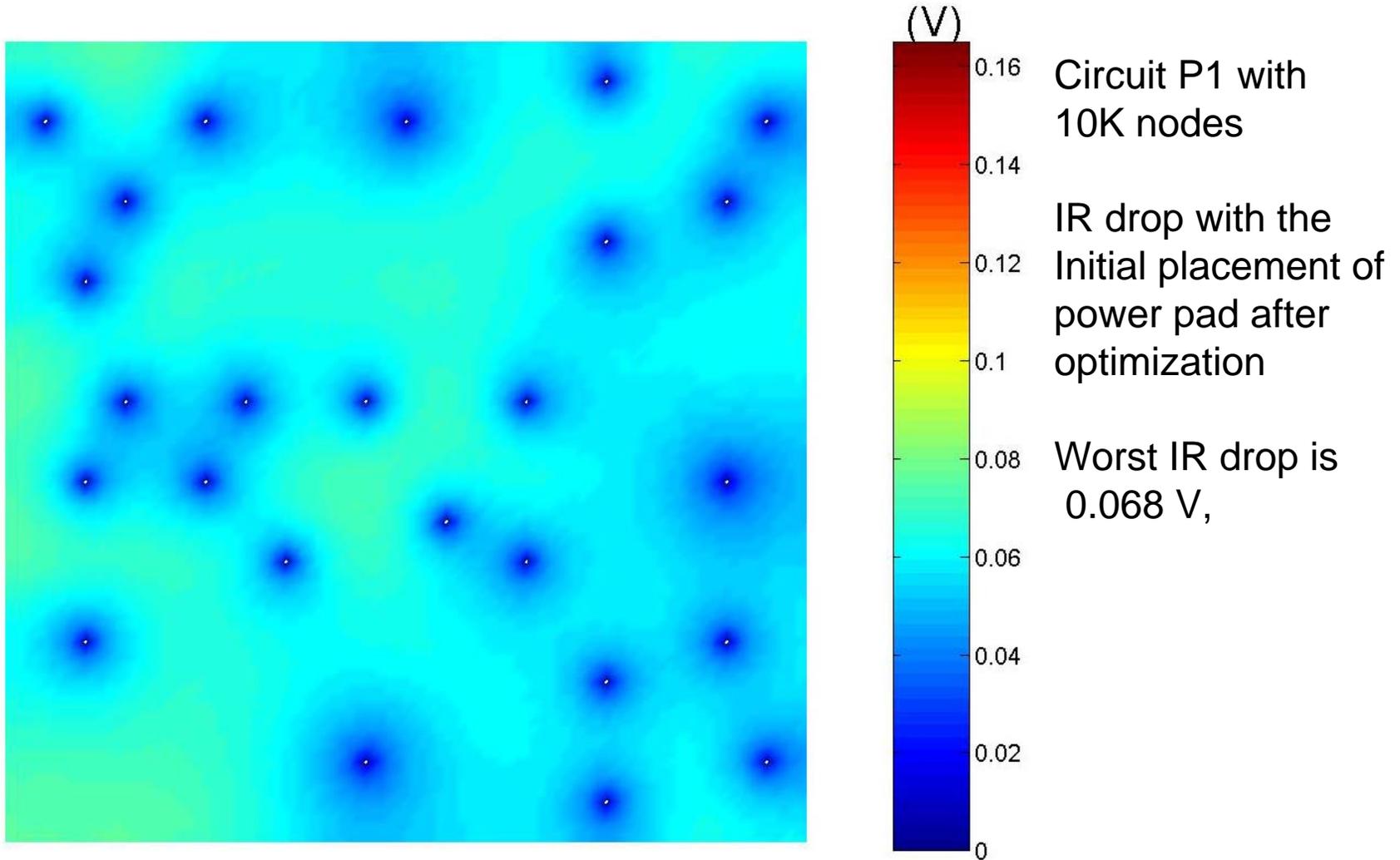


Circuit P1 with
10K nodes

IR drop with the
Initial placement of
power pad before
optimization

Worst IR drop is
0.161 V

Experimental Results



Conclusion

- Placement optimization of power supply pads is a big challenge.
- Localized node-based iterative method
 - Fast computation of voltages after each pad movement
 - Independent of the total size of the power grid
 - Scalable for large power grids
- Simulated annealing based placement algorithm
- Experimental results show
 - Improvement of the worst IR drop and uniformity of IR drop
 - Good runtime characteristics for large number of power pad location candidates in multi-million-size circuits.