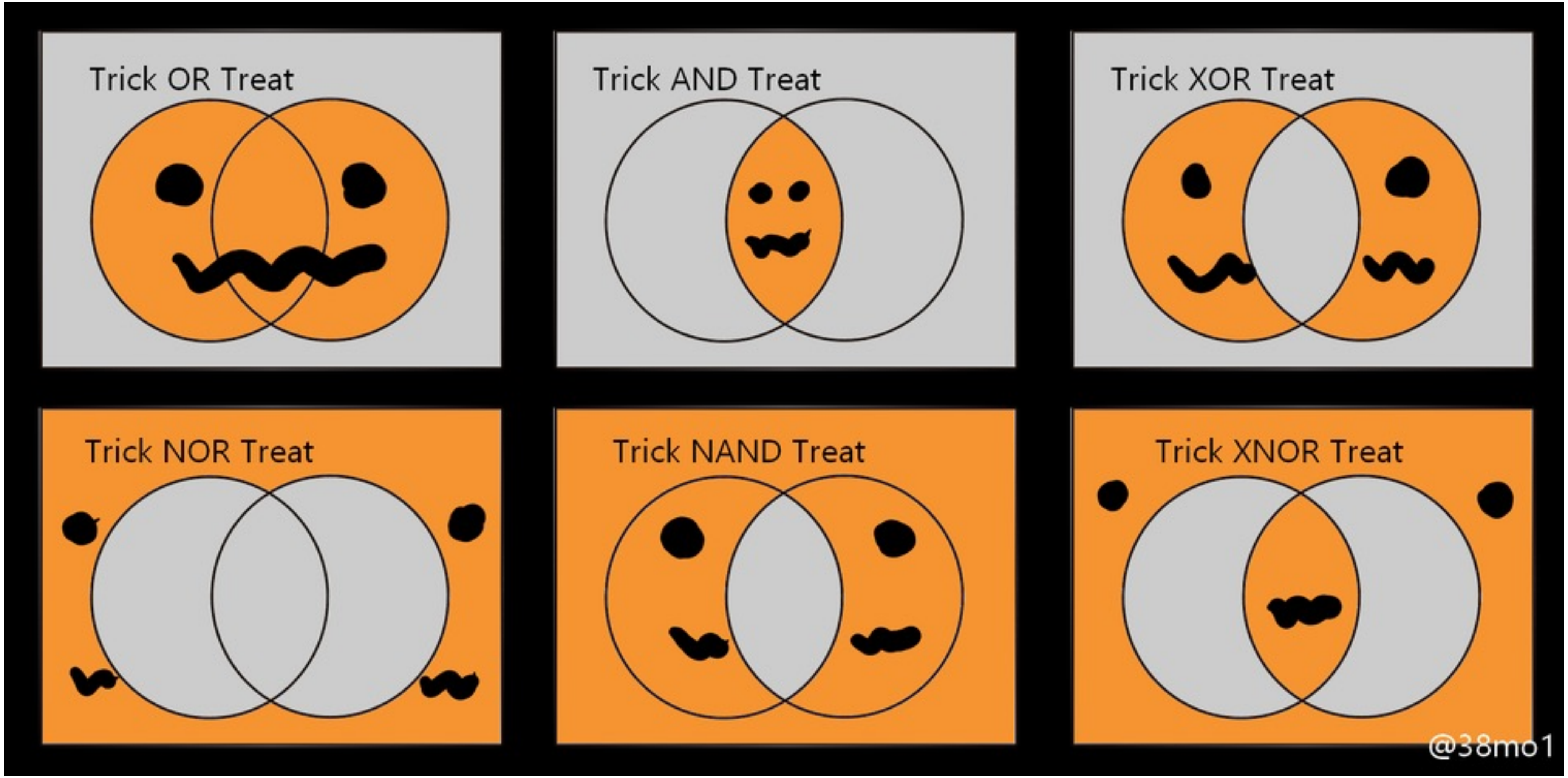
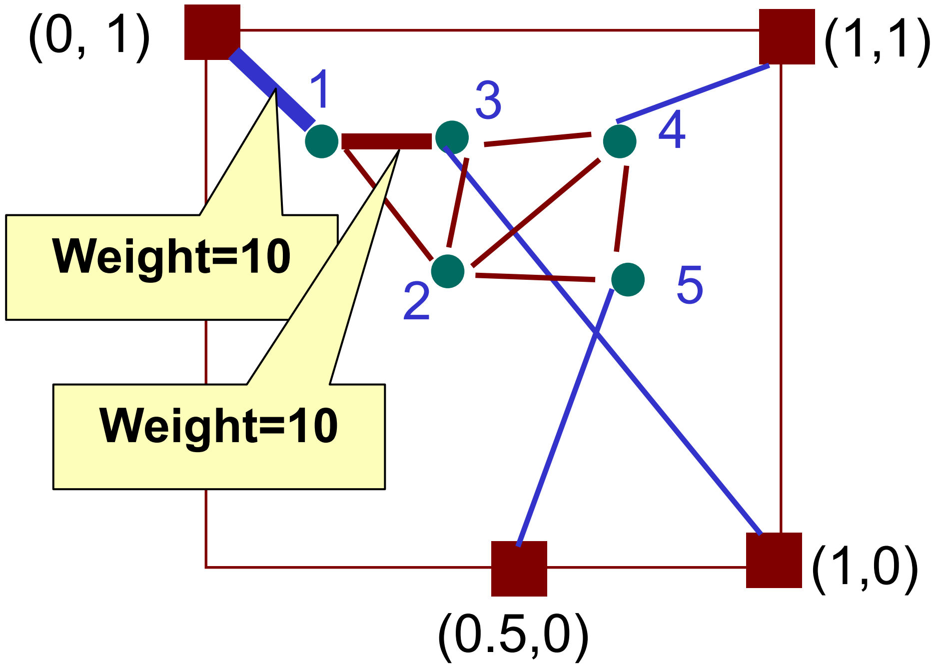
Lecture 15: Placement – V

Tsung-Wei (TW) Huang
Department of Electrical and Computer Engineering
University of Utah, Salt Lake City, UT

Happy Halloween!



Recap: Quadratic Placement Formulation



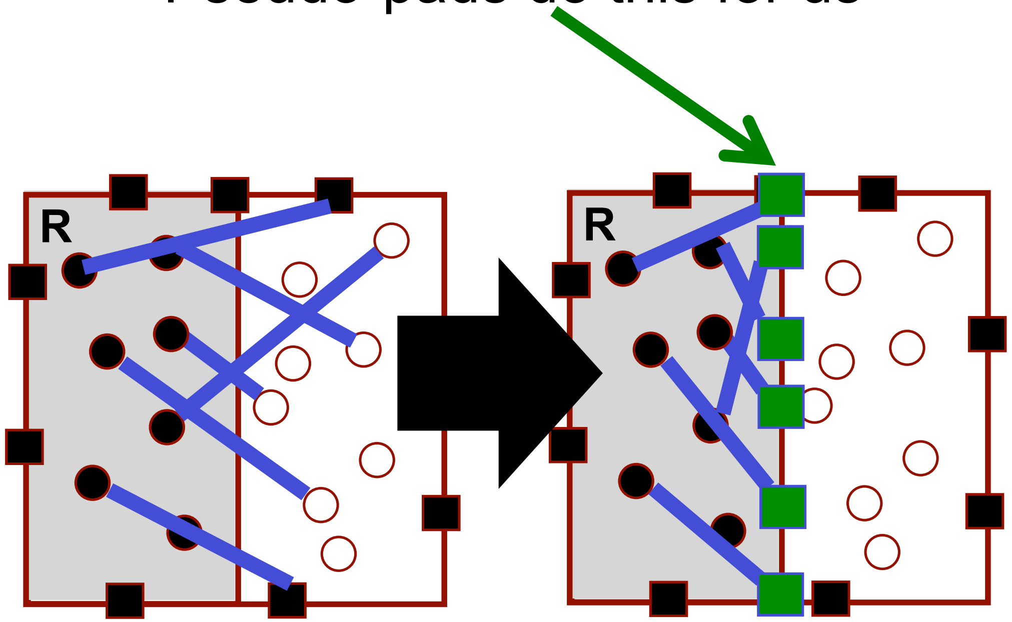
All wire weights = 1 *except* two highlighted:
gate1 to **pad** and **gate1** to **gate2**

$$\mathbf{C} = \begin{pmatrix} 0 & 1 & 10 & 0 & 0 \\ 1 & 0 & 1 & 1 & 1 \\ 10 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 \end{pmatrix} \qquad \mathbf{A} = \begin{pmatrix} 21 & -1 & -10 & 0 & 0 \\ -1 & 4 & -1 & -1 & -1 \\ -10 & -1 & 13 & -1 & 0 \\ 0 & -1 & -1 & 4 & -1 \\ 0 & -1 & 0 & -1 & 3 \end{pmatrix}$$

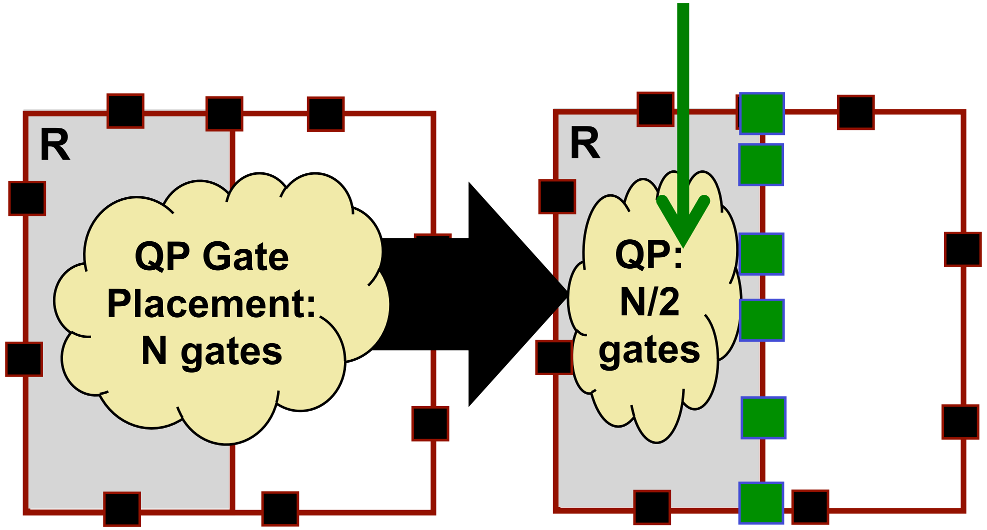
$$\mathbf{b_x} = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \\ 0.5 \end{pmatrix} \qquad \mathbf{b_y} = \begin{pmatrix} 10 \\ 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}$$

Recap: Partition-based Placement

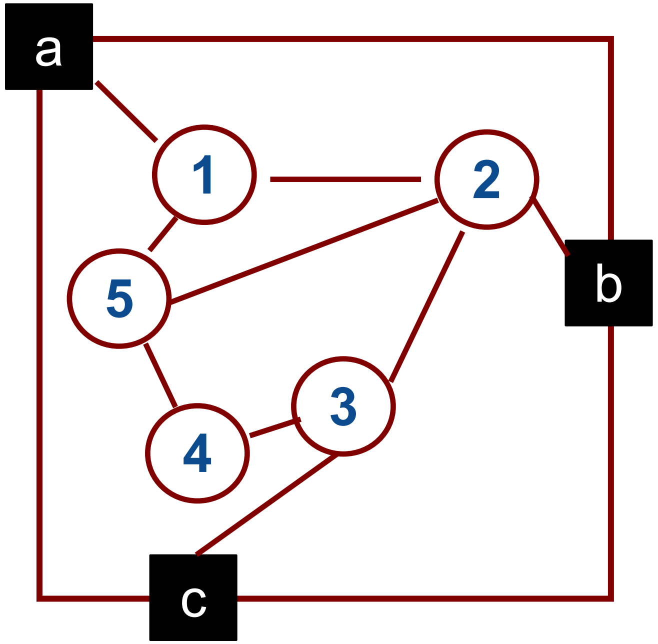
- **Cannot ignore gates outside region we are re-placing**
  - Want gates inside to *feel pull* from wires to *gates outside* region
  - Pseudo-pads do this for us



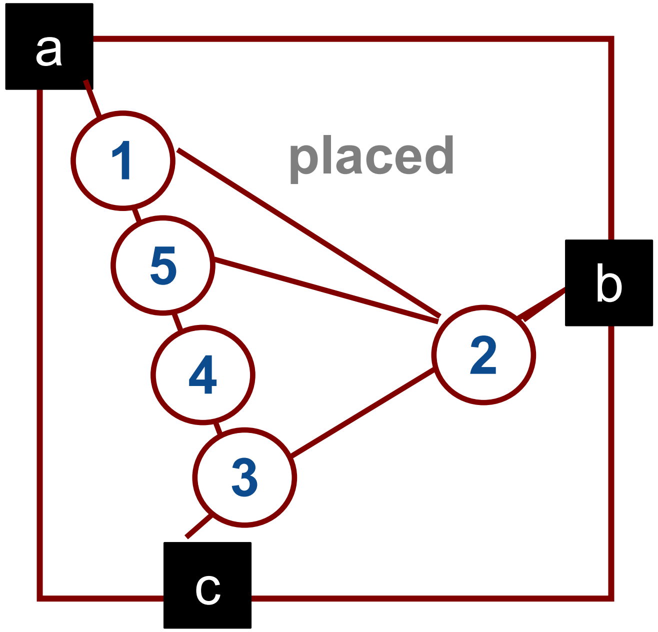
- **Pseudo-pads guarantee all gates re-locate inside region**
  - Think of wires as 'springs' that each pull gates *toward* other gates or pads
  - If pads (real & pseudo) are on edges of region – **QP keeps gates inside**



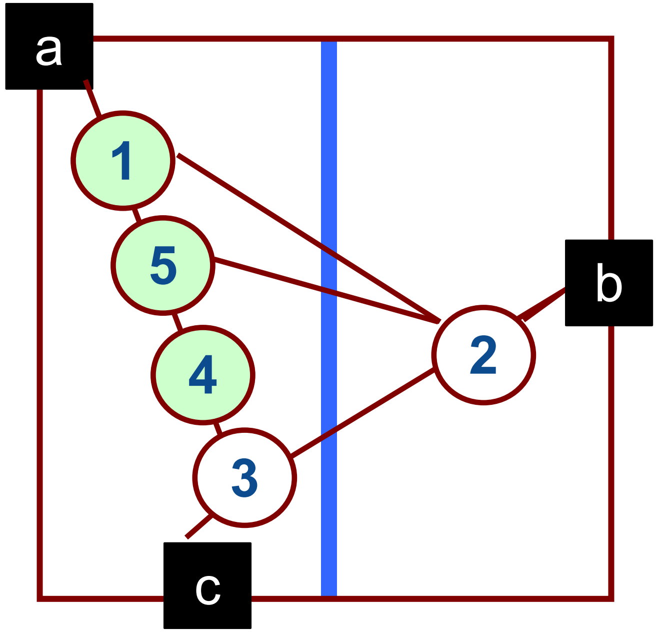
Recap: Example



1. Initial netlist
5 gates (1,2,3,4,5)
8 wires
3 pads (a,b,c)

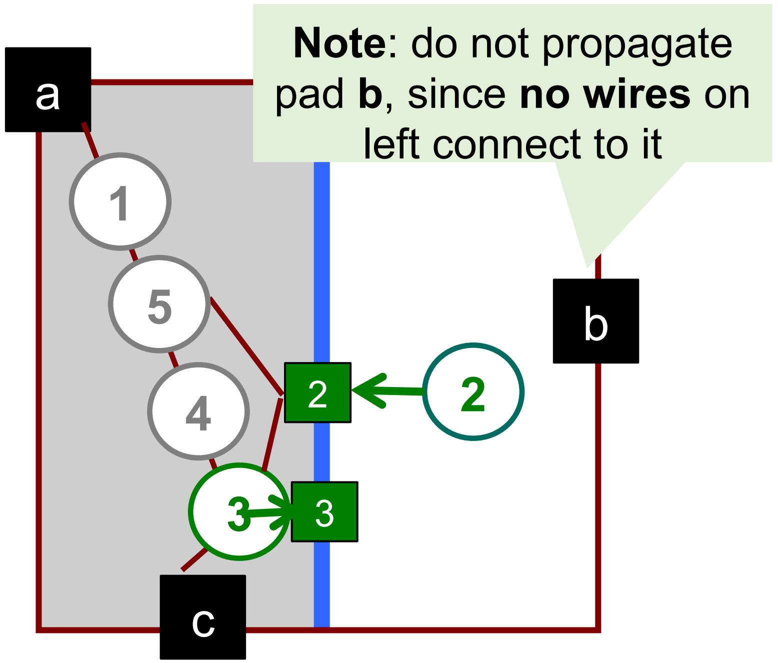


2. Initial QP

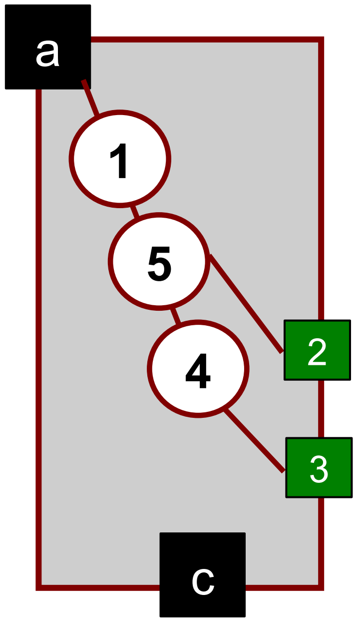


3. First partition
Sort on **X**:
Gate order 1 5 4 3 2
Pick: **1 5 4** on left

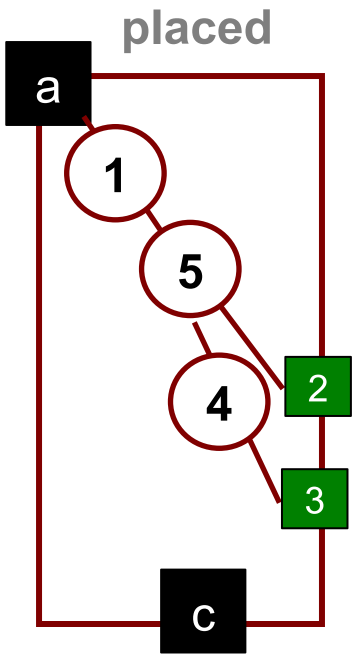
Recap: Example (cont’d)



4. Propagate gates/pads
Right-side gates: 2,3
Right-side pads: b
Push to cut, using **y** coordinates

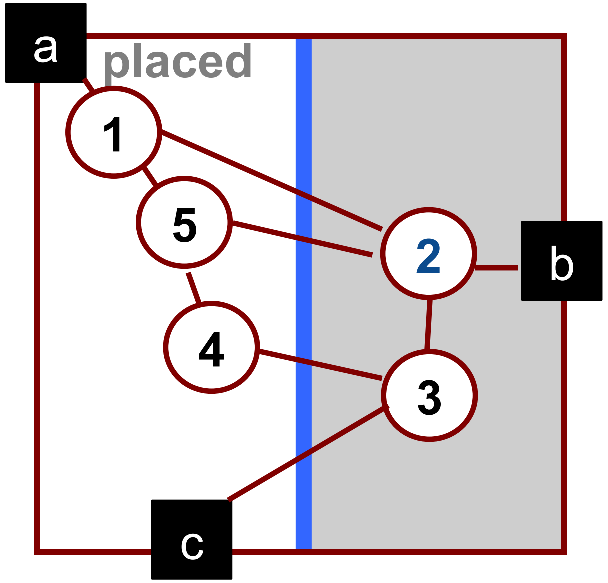


5. 2^{nd} QP input
This is set up for this new smaller placement

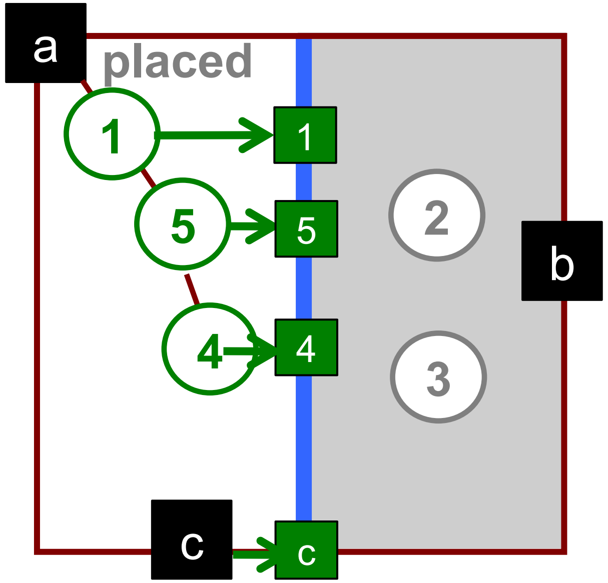


6. 2^{nd} QP solved
New placement

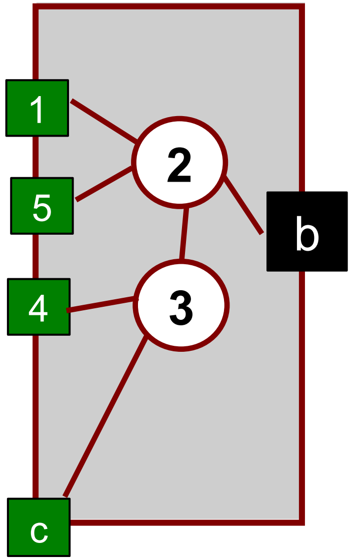
Recap: Example (cont’d)



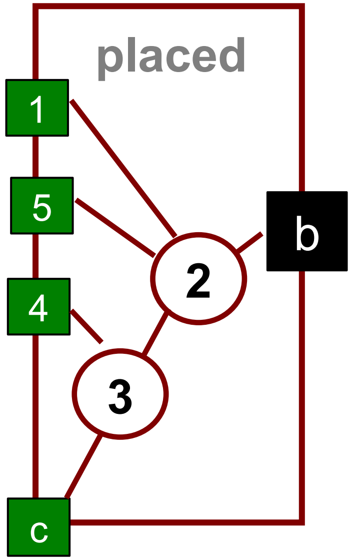
7. Left side placed.
Now, re-place right-side gates.



8. Propagate gates/pads
This is set up for next, new smaller placement

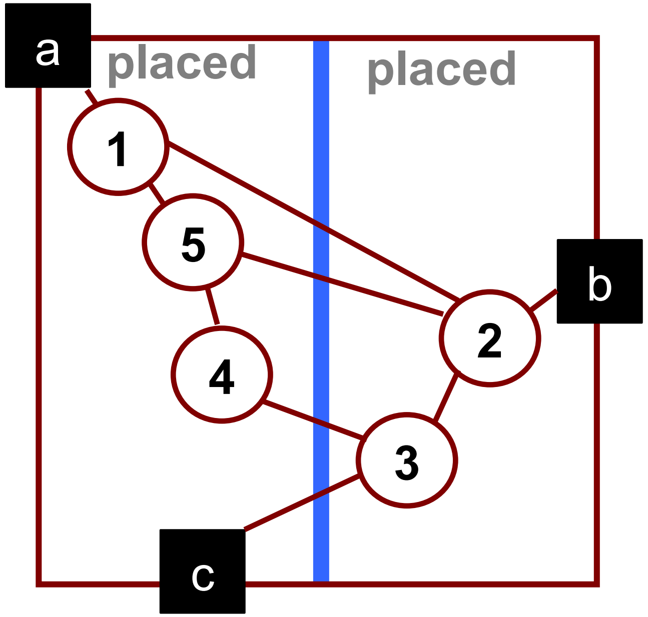


9. 3^{nd} QP input
This is set up for this new smaller placement

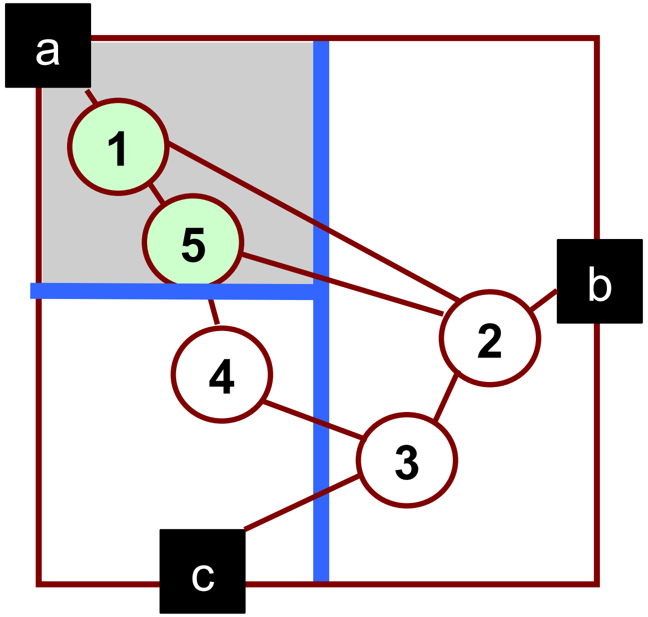


10. 3^{nd} QP solve

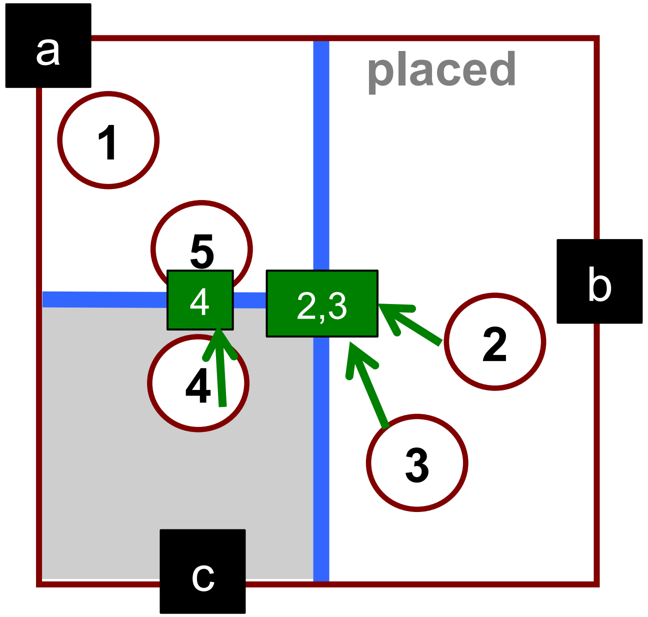
Recap: Example (cont'd)



Repeat: Horizontal partition on left

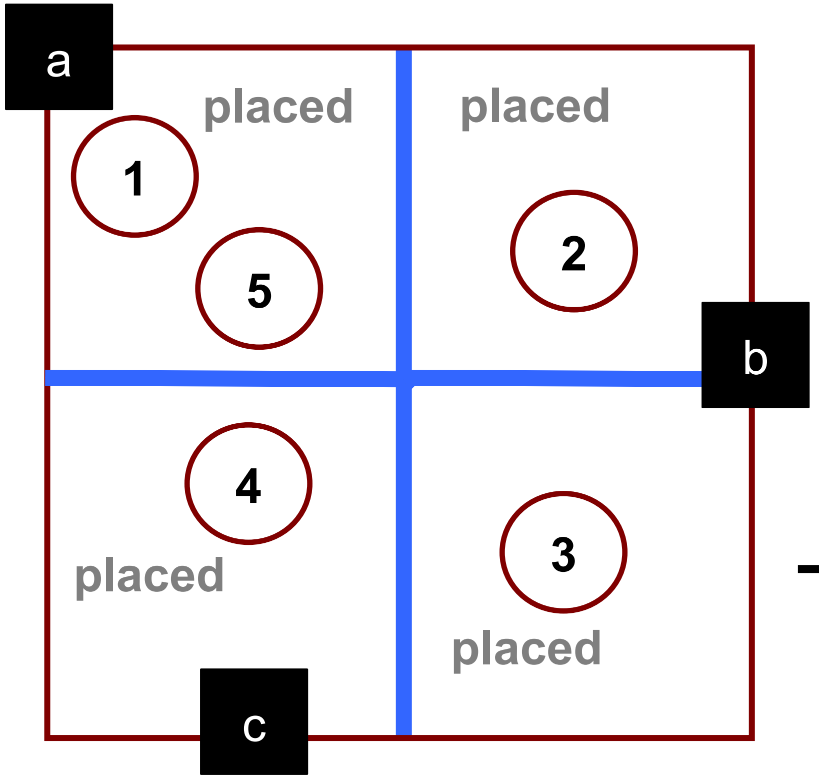


Focus on top.
Sort gates on **Y**
Assign gates 1,5 to region.



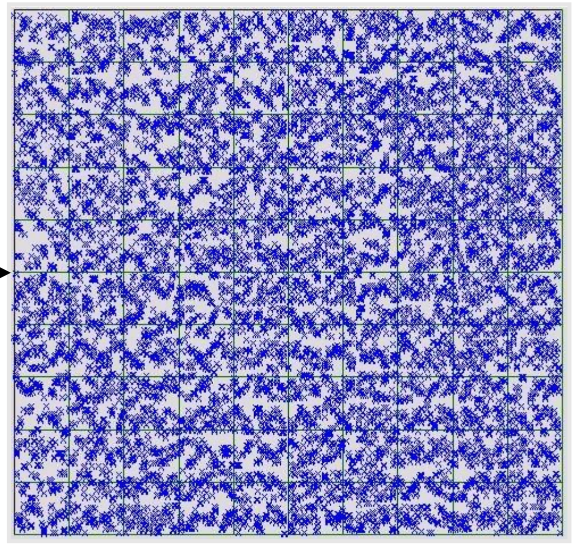
Propagate gates & pads

Recap: Keep Repeating this Recursion



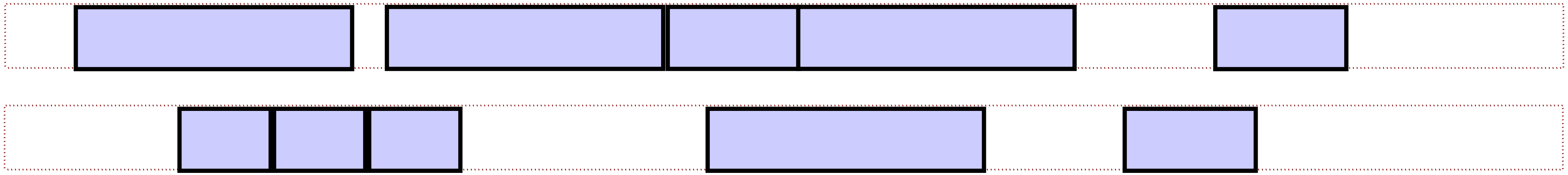
Continue…

- **Keep recursively partitioning…**
  - Usually, you continue until you have a “small” number of gates in each region
  - Small >> 1 typically. **10-100** for example
  - Get a good, “global” placement, but not a “final” placement



Recap: Legalization

- **Still need to force gates in precise rows for final result**
  - QP methods *cannot* force individual gates into standard cell rows, without overlaps



- **Solution step is called: Legalization**
  - Many different algorithms. One easy way to do this is by annealing!
  - Do local improvement based on swaps of nearby gates
  - To anneal, set **T=HOT** to be very small (cold), so don't disrupt QP result

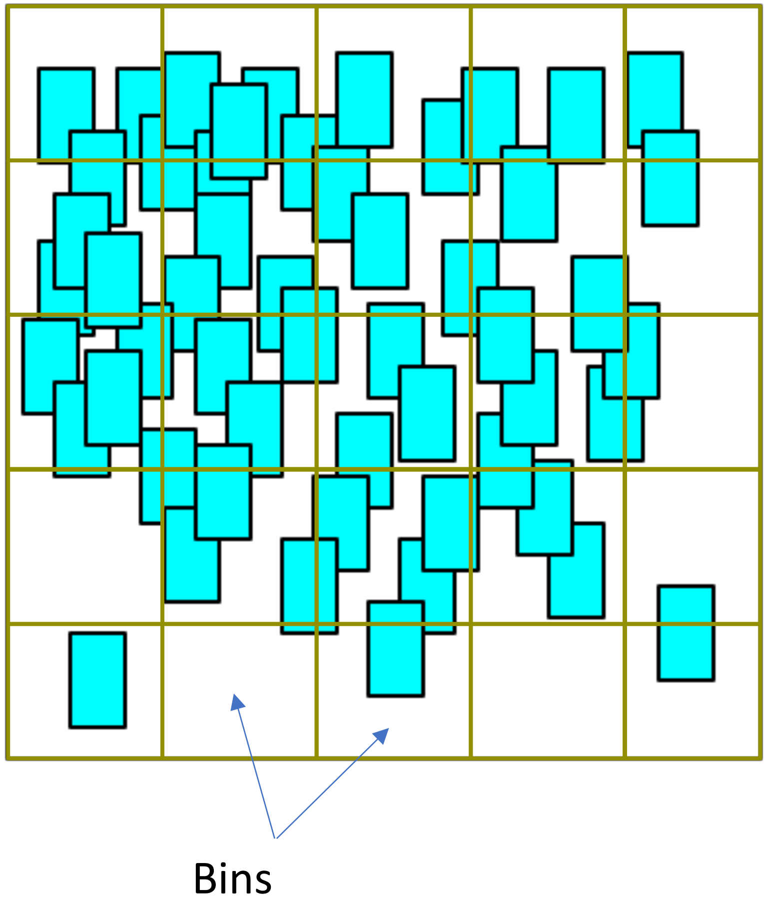
Nonlinear Placement

- **Mathematical formulation**
  - d_i denotes the density of bin i

$$\min_{\boldsymbol{x},\boldsymbol{y}} \; WL(\boldsymbol{x},\boldsymbol{y}),$$
$$s.t. \;\; d_b(\boldsymbol{x},\boldsymbol{y}) \leq t_d, \forall b \in Bins$$

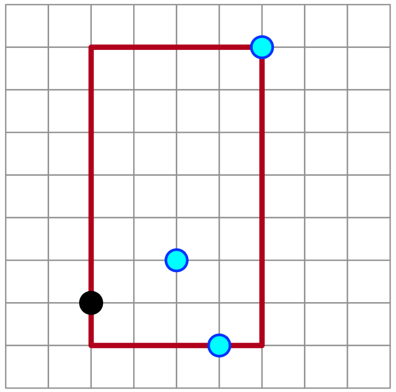
- **Nonlinear placement objective**
  - Lagrangian relaxation

$$\min_{\boldsymbol{x},\boldsymbol{y}} \; \underbrace{WL(\boldsymbol{x},\boldsymbol{y})}_{\text{Wirelength}} + \lambda \underbrace{D(\boldsymbol{x},\boldsymbol{y})}_{\text{Density}}$$



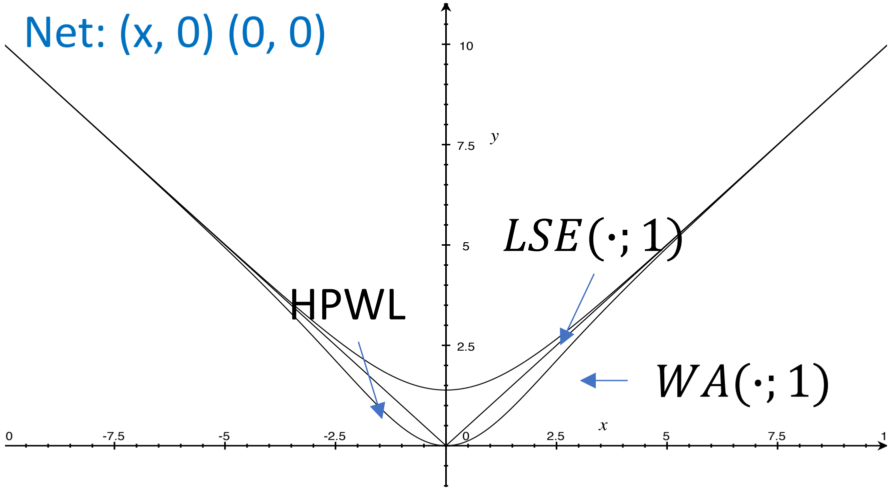
Wirelength Smoothing

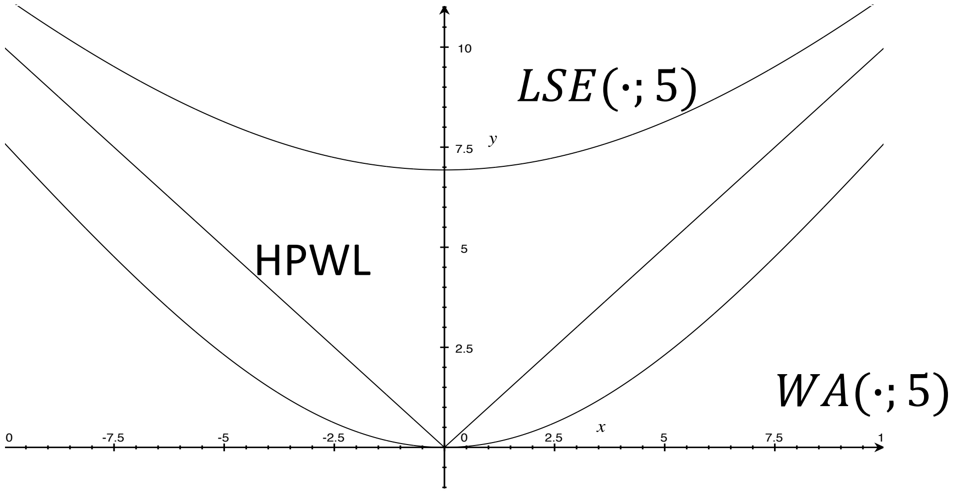
- $WL(\boldsymbol{x},\boldsymbol{y}) = \sum_{e\in E} WL_e(\boldsymbol{x},\boldsymbol{y})$
- $HPWL = max|x_i - x_j| + max\,|y_i - y_j|$
  - Equivalently $\left(\max_i x_i - \min_i x_i\right) + \left(\max_i y_i - \min_i y_i\right)$
- Log-sum-exp (LSE)
  - $LSE(\boldsymbol{x};\gamma) = \gamma \ln \sum_i e^{\frac{x_i}{\gamma}}$
  - $\max\{x_1,\cdots,x_n\} < LSE(\boldsymbol{x};\gamma) \le \max\{x_1,\cdots,x_n\} + \gamma\ln(n)$
  - $LSE(\boldsymbol{x};\gamma) \approx \max\{x_1,\cdots,x_n\}$
  - $-LSE(\boldsymbol{x};-\gamma) \approx \min\{x_1,\cdots,x_n\}$
  - $WL_e(\boldsymbol{x},\boldsymbol{y};\gamma) = \gamma(\underbrace{\ln\sum_{v_i\in e} e^{\frac{x_i}{\gamma}} + \ln\sum_{v_i\in e} e^{-\frac{x_i}{\gamma}}}_{x} + \underbrace{\ln\sum_{v_i\in e} e^{\frac{y_i}{\gamma}} + \ln\sum_{v_i\in e} e^{-\frac{y_i}{\gamma}}}_{y})$



Wirelength Smoothing (cont'd)

- **Weighted average (WA)**
  - $WA(\boldsymbol{x}, \boldsymbol{y}; \gamma) = \left(\frac{\sum_{v_i \in e} x_i e^{x_i/\gamma}}{\sum_{v_i \in e} e^{x_i/\gamma}} - \frac{\sum_{v_i \in e} x_i e^{-x_i/\gamma}}{\sum_{v_i \in e} e^{-x_i/\gamma}} \right) + \left(\frac{\sum_{v_i \in e} y_i e^{y_i/\gamma}}{\sum_{v_i \in e} e^{y_i/\gamma}} - \frac{\sum_{v_i \in e} y_i e^{-y_i/\gamma}}{\sum_{v_i \in e} e^{-y_i/\gamma}} \right)$
- **Larger $\boldsymbol{\gamma}$ → smoother, but less accurate**



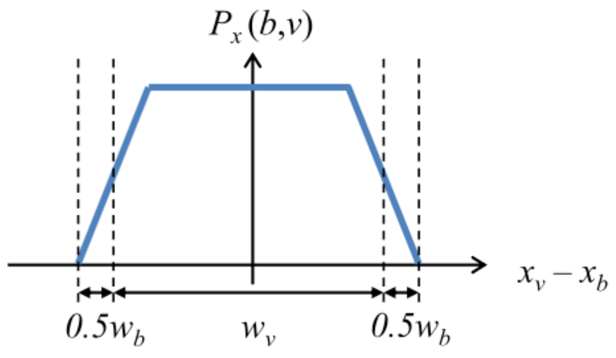


Nonlinear Placement

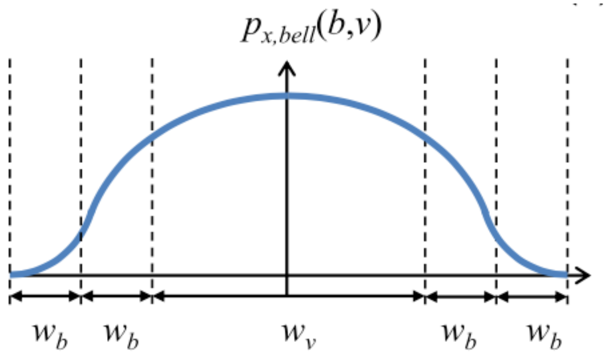
- Chen, Tung-Chieh, et al. "NTUplace: A ratio partitioning based placement algorithm for large-scale mixed-size designs." ISPD 2005
- Chen, Tung-Chieh, et al. "NTUplace3: An analytical placer for large-scale mixed-size designs with preplaced blocks and density constraints." IEEE TCAD 2008.
- Hsu, Meng-Kai, et al. "NTUplace4h: A novel routability-driven placement algorithm for hierarchical mixed-size circuit designs." IEEE TCAD 2014
- Huang, Chau-Chin, et al. "NTUplace4dr: a detailed-routing-driven placer for mixed-size circuit designs with technology and region constraints." IEEE TCAD 2017

Density Penalty

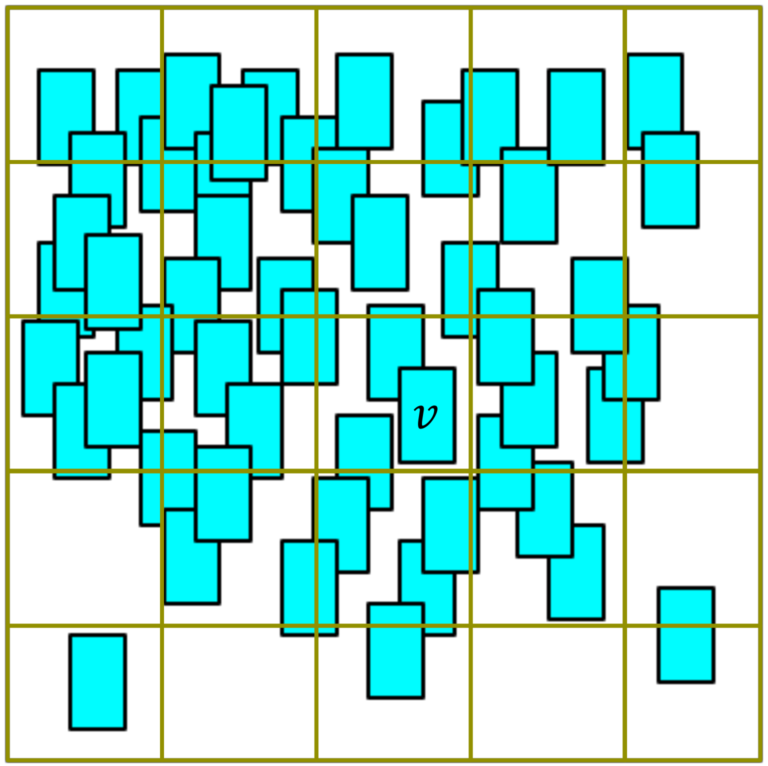
- **Potential function for standard cells**
  - $P_x(b,v)$ and $P_y(b,v)$ are the overlap functions between bin b and cell v
  - $D_b(\boldsymbol{x},\boldsymbol{y}) = \sum_{v \in V} P_x(b,v) P_y(b,v)$



Non-smooth
Non-convex



Bell-shape smoothing



Bell Function Examples

- Gaussian function, the probability density function of the normal distribution. This is the archetypal bell shaped function and is frequently encountered in nature as a consequence of the central limit theorem.

  $$f(x) = ae^{-(x-b)^2/(2c^2)}$$

- Fuzzy Logic generalized membership bell-shaped function[2][3]

  $$f(x) = \frac{1}{1 + \left|\frac{x-c}{a}\right|^{2b}}$$

- Hyperbolic secant. This is also the derivative of the Gudermannian function.

  $$f(x) = \operatorname{sech}(x) = \frac{2}{e^x + e^{-x}}$$

- Witch of Agnesi, the probability density function of the Cauchy distribution. This is also a scaled version of the derivative of the arctangent function.

  $$f(x) = \frac{8a^3}{x^2 + 4a^2}$$

- Bump function

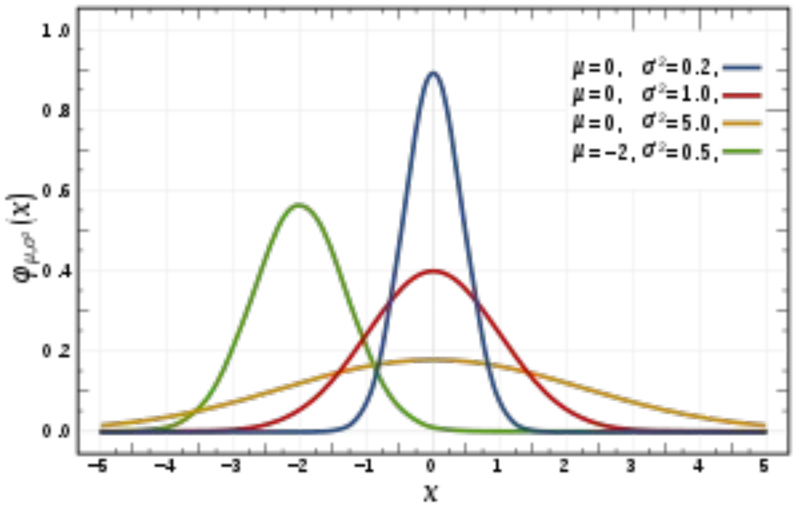
  $$\varphi_b(x) = \begin{cases} \exp \frac{b^2}{x^2 - b^2} & |x| < b, \\ 0 & |x| \geq b. \end{cases}$$

- Raised cosines type like the raised cosine distribution or the raised-cosine filter

  $$f(x; \mu, s) = \begin{cases} \frac{1}{2s}\left[1 + \cos\left(\frac{x-\mu}{s}\pi\right)\right] & \text{for } \mu - s \leq x \leq \mu + s, \\ 0 & \text{otherwise.} \end{cases}$$

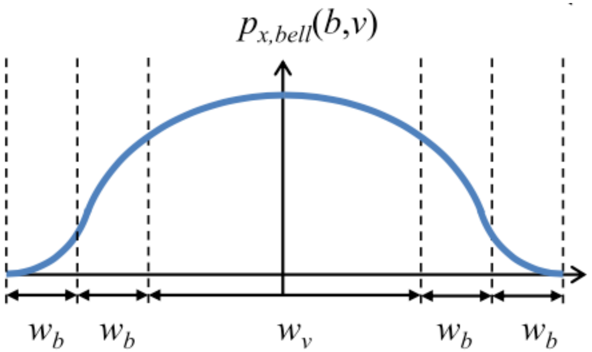
- Most of the window functions like the Kaiser window
- The derivative of the logistic function. This is a scaled version of the derivative of the hyperbolic tangent function.

  $$f(x) = \frac{e^x}{(1 + e^x)^2}$$



Density Penalty (cont'd)

- Potential function for standard cells
  - $P_x(b, v)$ and $P_y(b, v)$ are the overlap functions between bin b and cell v
  - $D_b(\boldsymbol{x}, \boldsymbol{y}) = \sum_{v \in V} P_x(b, v) P_y(b, v)$



Bell-shape smoothing

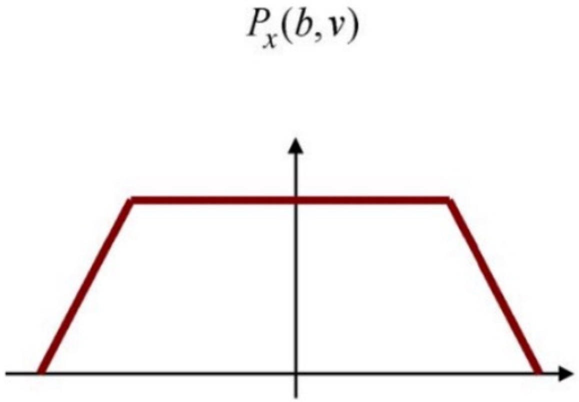
$$p_x(b,v) = \begin{cases} 1 - a d_x^2, & 0 \le d_x \le \frac{w_v}{2} + w_b \\ b\left(d_x - \frac{w_v}{2} - 2w_b\right)^2, & \frac{w_v}{2} + w_b \le d_x \le \frac{w_v}{2} + 2w_b \\ 0, & \frac{w_v}{2} + 2w_b \le d_x \end{cases}$$

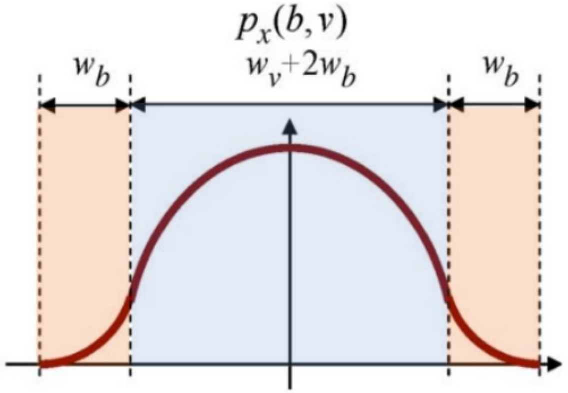
where $$a = \frac{4}{(w_v + 2w_b)(w_v + 4w_b)}$$

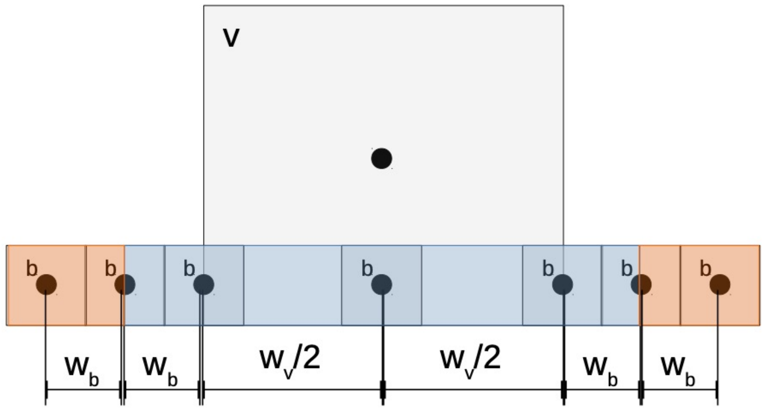
$$b = \frac{2}{w_b(w_v + 4w_b)}$$

Variables *dx* (*dy*) is the **absolute** center-to-center distance between cell *v* and bin *b* in the *x* (*y*) direction

Visualization







$$p_x(b,v)=\begin{cases} 1-ad_x^2, & 0\le d_x\le \frac{w_v}{2}+w_b \\ b\left(d_x-\frac{w_v}{2}-2w_b\right)^2, & \frac{w_v}{2}+w_b\le d_x\le \frac{w_v}{2}+2w_b \\ 0, & \frac{w_v}{2}+2w_b\le d_x \end{cases}$$

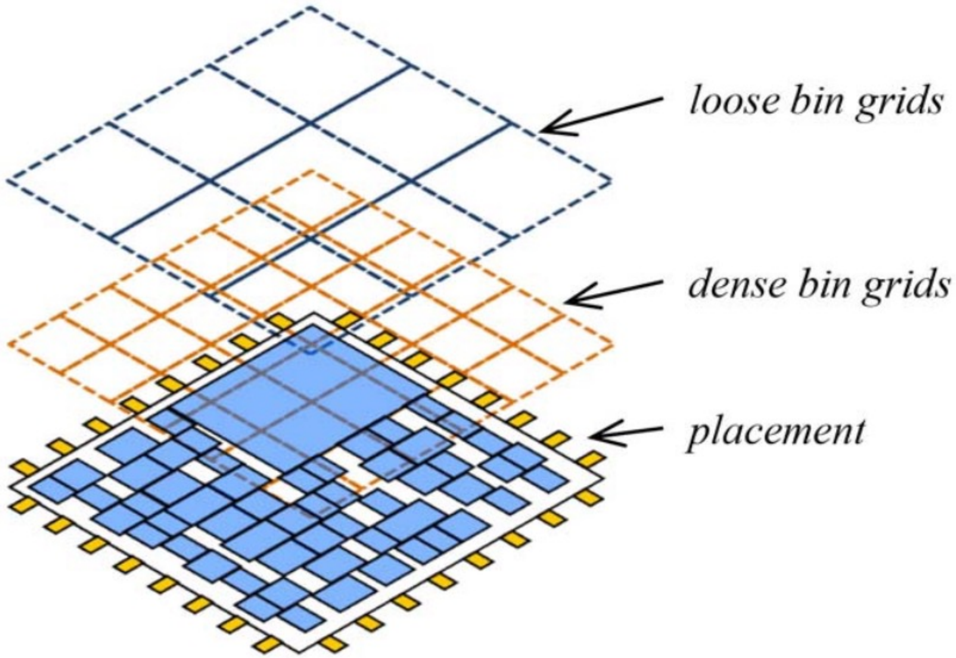
Density Penalty (cont'd)

- Potential function for standard cells
  - Smoothed potential function
  - $\widehat{D_b}(\boldsymbol{x}, \boldsymbol{y}) = \sum_{v \in V} \widehat{P_x}(b, v) \widehat{P_y}(b, v)$
- $\min_{\boldsymbol{x}, \boldsymbol{y}} \; WL(\boldsymbol{x}, \boldsymbol{y}) + \lambda D(\boldsymbol{x}, \boldsymbol{y})$

$$\Downarrow$$

$$\lambda \sum_b \left(\widehat{D_b}(\boldsymbol{x}, \boldsymbol{y}) - t_d \right)^2$$

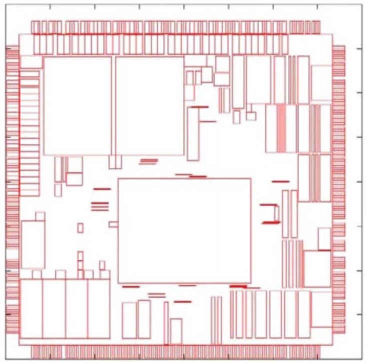
- Challenges
  - Gradient only has local view
  - Need multi-level bins



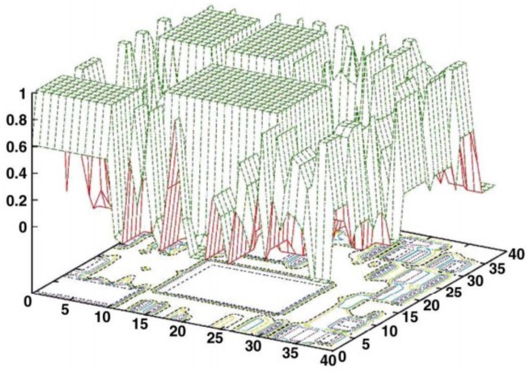
Multi-level bins

Density with Fixed Macro

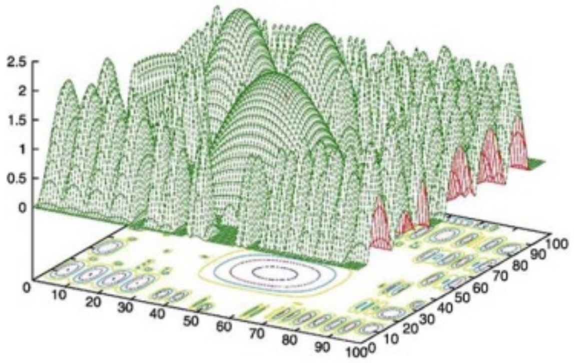
- Potential function for fixed macros
  - Bell-shape smoothing works well for standard cells
  - For fixed macros, $P'(x,y) = G(x,y) * P(x,y)$



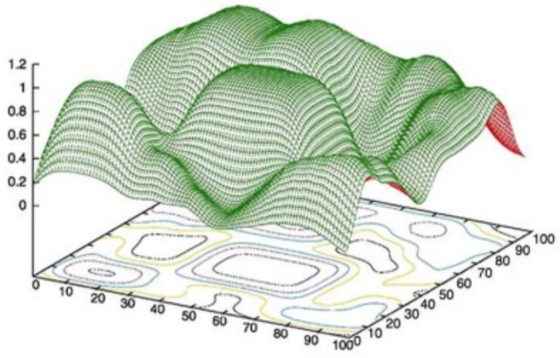
ISPD2005
adaptec2



Exact potential
$P(x,y)$



Bell-shape smoothing



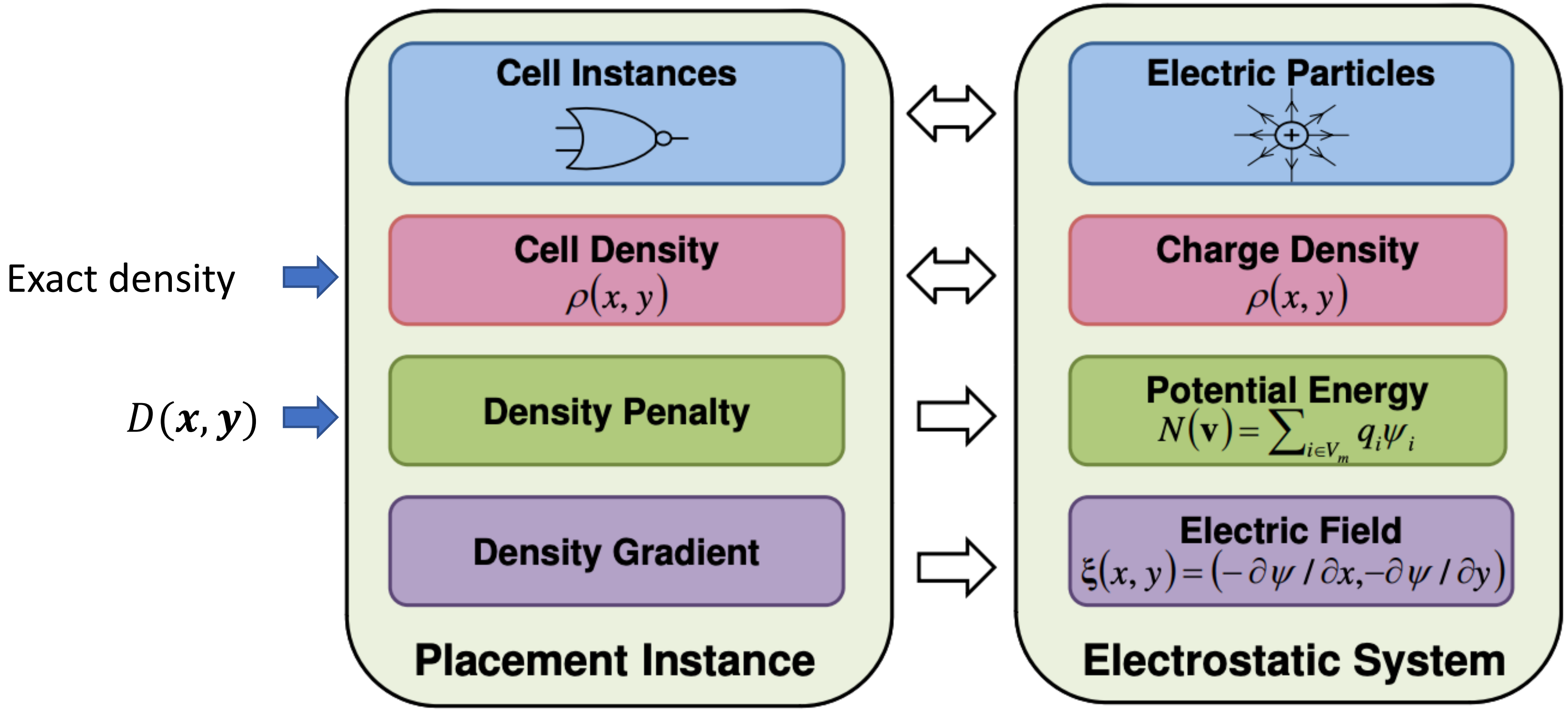
Gaussian smoothing
$P'(x,y)$

Nonlinear Placer – ePlace

- Lu, Jingwei, et al. "ePlace: Electrostatics-based placement using fast fourier transform and Nesterov's method." ACM TODAES 2015.
- Cheng, Chung-Kuan, et al. "RePlAce: Advancing solution quality and routability validation in global placement." IEEE TCAD 2018.
- Lin, Yibo, et al. "DREAMPlace: Deep learning toolkit-enabled gpu acceleration for modern vlsi placement." IEEE TCAD 2020. (DAC 2019 Best Paper Award)

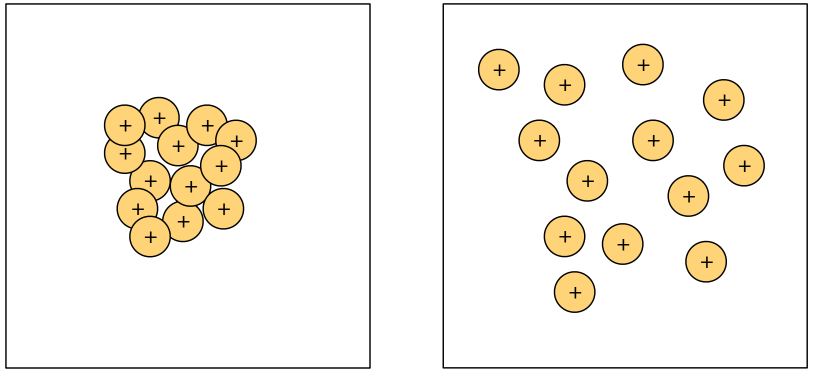
ePlace: Electric Potential

- $\min\limits_{\boldsymbol{x},\boldsymbol{y}} \; WL(\boldsymbol{x},\boldsymbol{y}) + \lambda\, D(\boldsymbol{x},\boldsymbol{y})$



Electrostatic System: Physics Effects

- **Isolated electrostatic system**
  - Balanced distribution ⇔ minimum potential energy
- **Minimizing the potential energy will spread out cells**



- **To consider $\boldsymbol{t_d < 1}$**
  - $s.t.\ \ d_b(\boldsymbol{x}, \boldsymbol{y}) \le t_d, \forall b \in Bins$
  - Insert fillers: dummy cells filling the area
  - $area_{fillers} + area_{cells} = area_{placeable} \times t_d$
  - Fillers have no connections

Poisson's Equation for Electrostatic System

$$\begin{cases} \nabla \cdot \nabla \psi(x,y) = -\rho(x,y), \\ \hat{\mathbf{n}} \cdot \nabla \psi(x,y) = \mathbf{0},\ (x,y) \in \partial R, \\ \underbrace{\iint_R \rho(x,y)}_{\text{Total charge}} = \underbrace{\iint_R \psi(x,y)}_{\text{Total energy}} = 0. \end{cases}$$

To remove DC component

$a_{0,0} = 0$

Zero-frequency component

$$\xrightarrow{\text{Solution}}$$

$$a_{u,v} = \frac{1}{m^2} \sum_{x=0}^{m-1} \sum_{y=0}^{m-1} \rho(x,y) \cos(w_u x) \cos(w_v y).$$

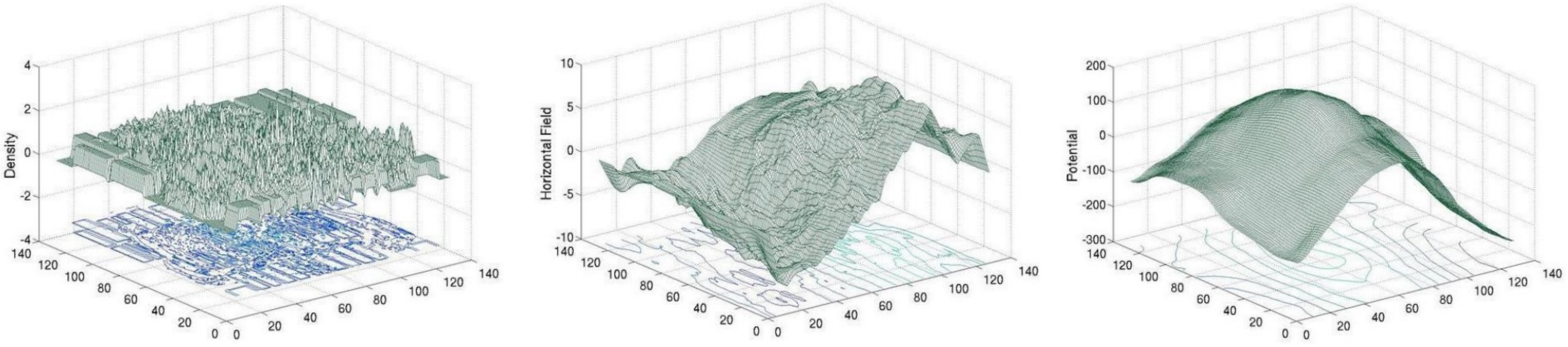
$$\rho_{DCT}(x,y) = \sum_{u=0}^{m-1} \sum_{v=0}^{m-1} a_{u,v} \cos(w_u x) \cos(w_v y),$$

$$\psi_{DCT}(x,y) = \sum_{u=0}^{m-1} \sum_{v=0}^{m-1} \frac{a_{u,v}}{w_u^2 + w_v^2} \cos(w_u x) \cos(w_v y),$$

$$\begin{cases} \xi_{X_{DSCT}} = \sum_u \sum_v \frac{a_{u,v} w_u}{w_u^2 + w_v^2} \sin(w_u x) \cos(w_v y), \\ \xi_{Y_{DCST}} = \sum_u \sum_v \frac{a_{u,v} w_v}{w_u^2 + w_v^2} \cos(w_u x) \sin(w_v y). \end{cases}$$

In forms of DCT and DST

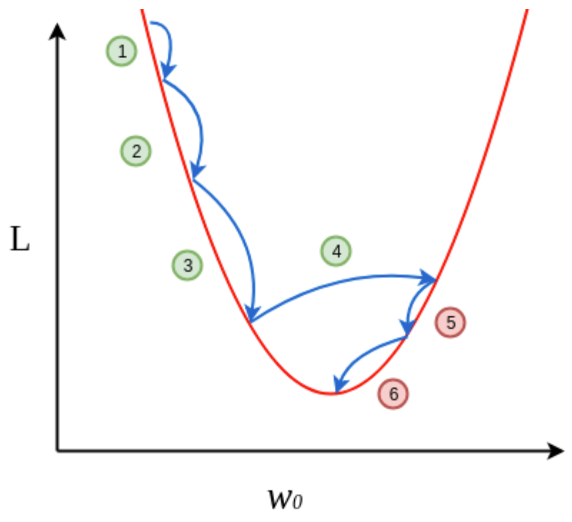
Electrical Potential

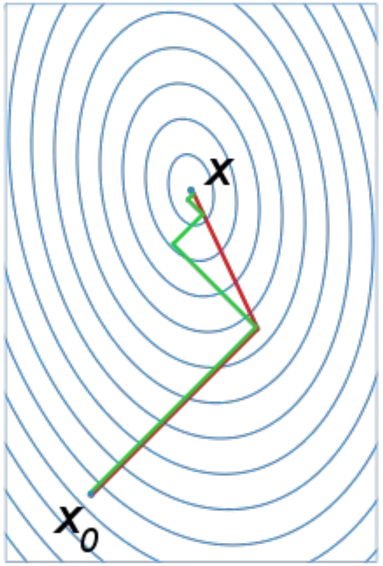


(a) Electric density. (b) Horizontal electric field (c) Electric potential.

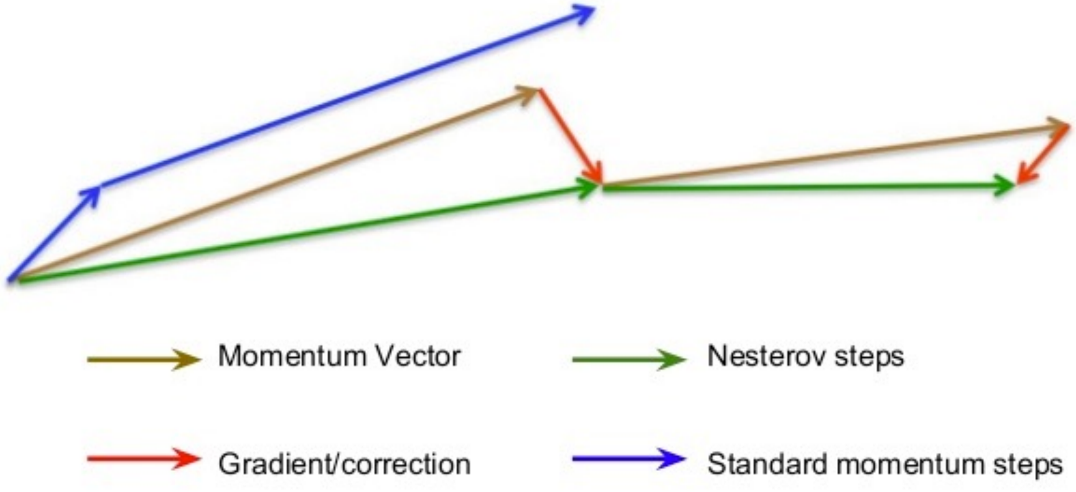
Gradient Solver

- [Conjugate gradient (CG) descent]
  - Between steepest descent & Newton method
  - Avoid computation of Hessian



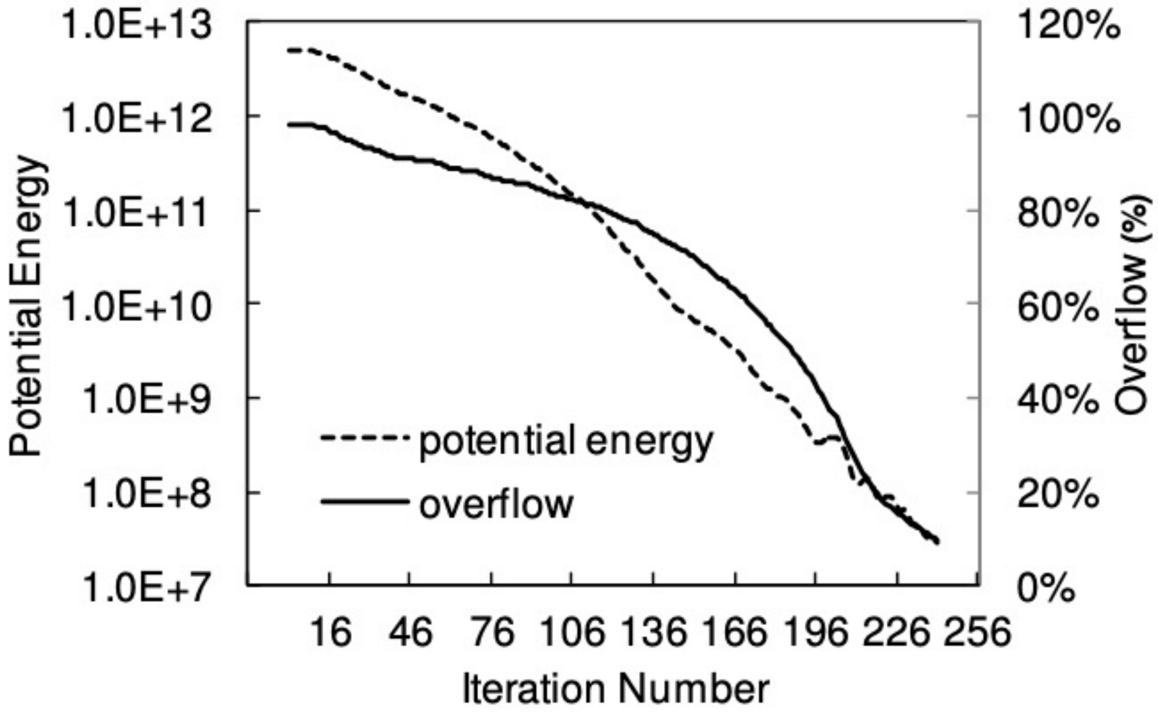


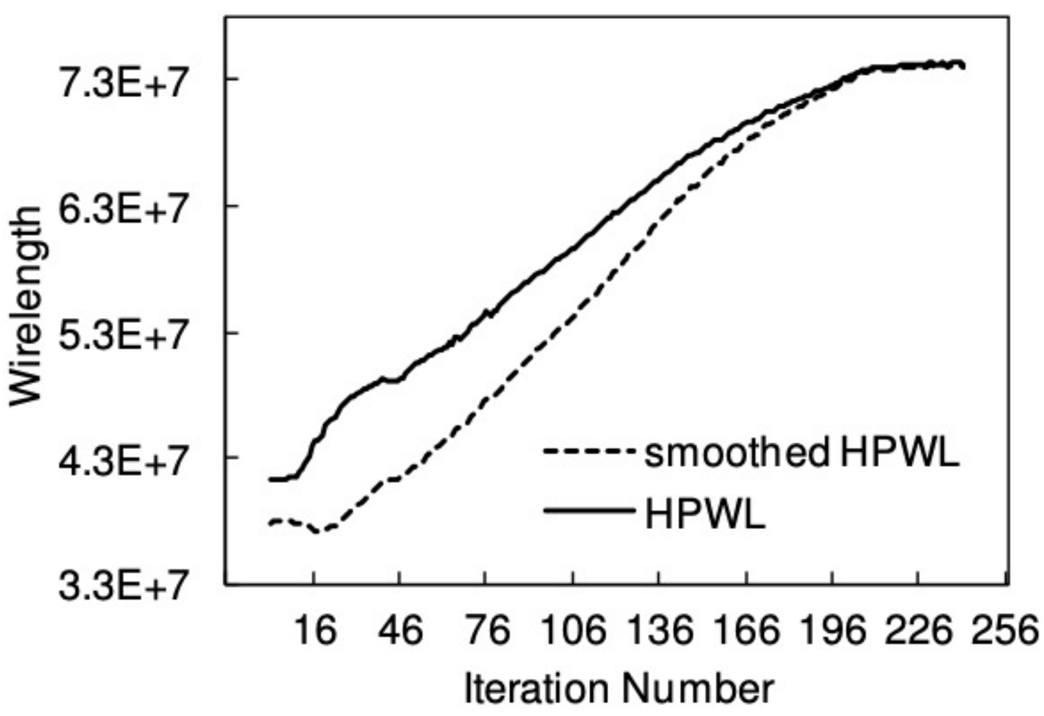
- [Nesterov's accelerated gradient descent]



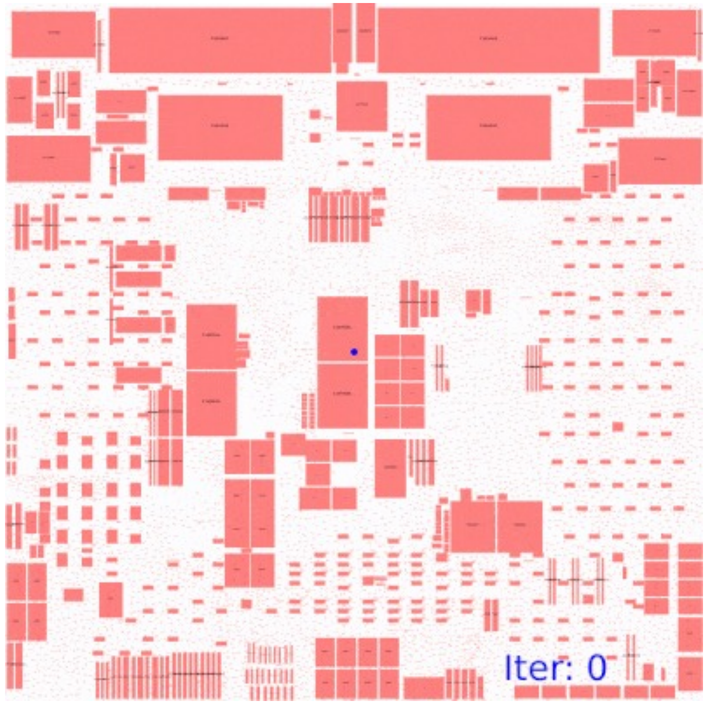
Source: Lecture by Geoffrey Hinton

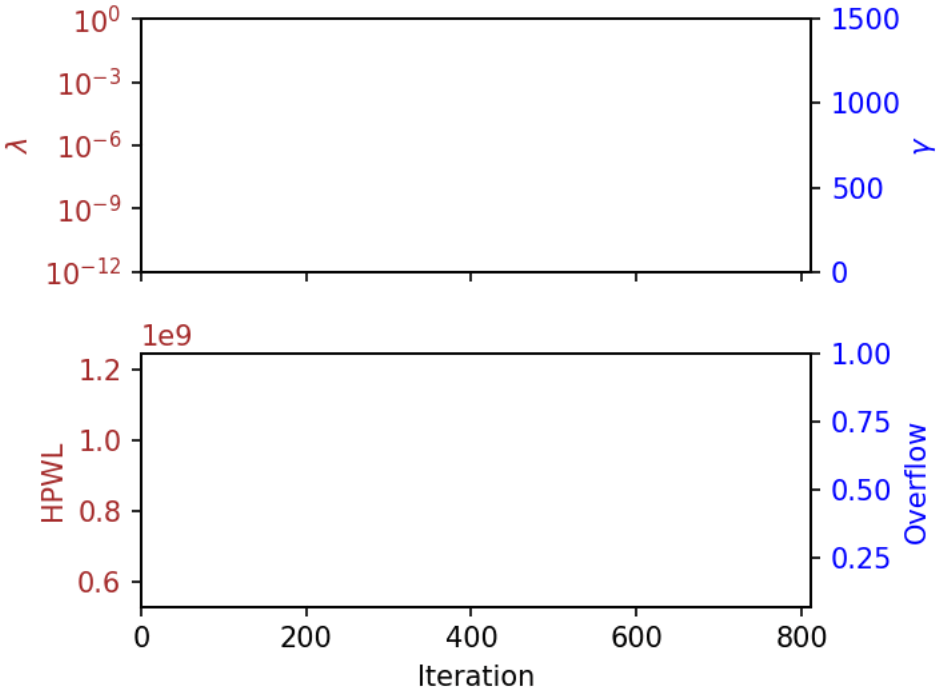
Gradient Descent Iterations



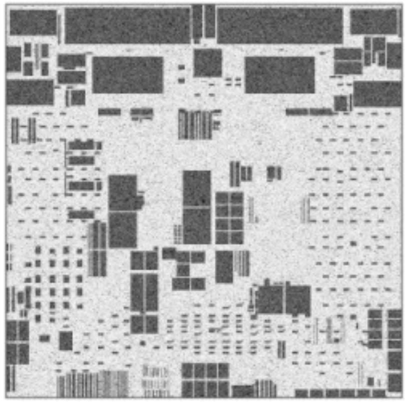


Industrial Example

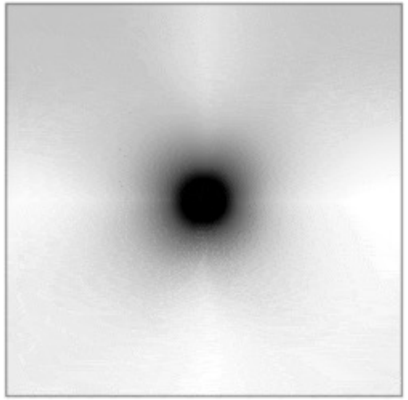




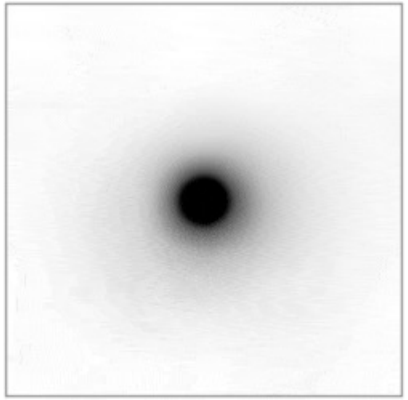
Placement Metrics



Density Map



Potential Map



Field Map

Summary

- **We have discussed nonlinear placement**
  - Wirelength smoothing: LSE and WA
  - Density potential (NTUplace)
  - Electric potential (ePlace)
- **Open-source tools**
  - DREAMPlace: https://github.com/limbo018/DREAMPlace
  - RePlAce: https://github.com/The-OpenROAD-Project/RePlAce

In-class Presentation: 11/16

- **Placement research presentation on 11/16 (in class)**
  - Yibo Lin, Shounak Dhar, Wuxi Li, Haoxing Ren, Brucek Khailany and David Z. Pan, "DREAMPlace: Deep Learning Toolkit-Enabled GPU Acceleration for Modern VLSI Placement", *ACM/IEEE Design Automation Conference (DAC)*, Las Vegas, NV, Jun 2-6, 2019
  - Spindler, Peter and Schlichtmann, Ulf and Johannes, Frank M., "Abacus: fast legalization of standard cell circuits with minimal movement," *ACM Proceedings of the 2008 International Symposium on Physical Design (ISPD)*, pp. 47–53, 2008
- Upload your pptx to https://github.com/tsung-wei-huang/ece5960-physical-design/issues/12 before presentation