# Lecture 6: Analytical Placement

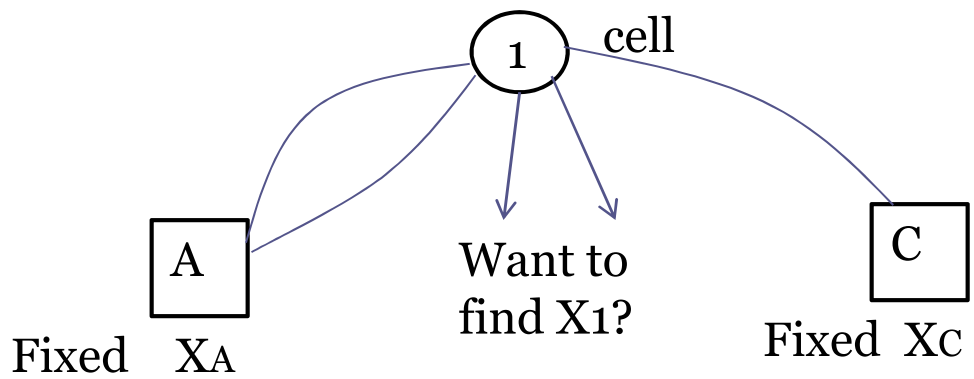
ECE 1387 – CAD for Digital Circuit Synthesis and Layout
2010

Scribe: S. Yu   Instructor: J. Anderson

# Analytical Placement

- Analytical Placement (AP) is used in commercial ASIC placers and Xilinx placer
- Main idea: “compute” placement for all cells in one shot
  - Simulated Annealing: better QoR
  - Analytical: faster run-time
- Most popular Analytical Placement: “quadratic placement”
  - Minimize quadratic WL (wirelength)

# Analytical Placement




- Linear WL: $\min \phi = |X_1 - X_A| + |X_1 - X_A| + |X_1 - X_C|$
- Quadratic WL: $\min \phi = (X_1 - X_A)^2 + (X_1 - X_A)^2 + (X_1 - X_C)^2$

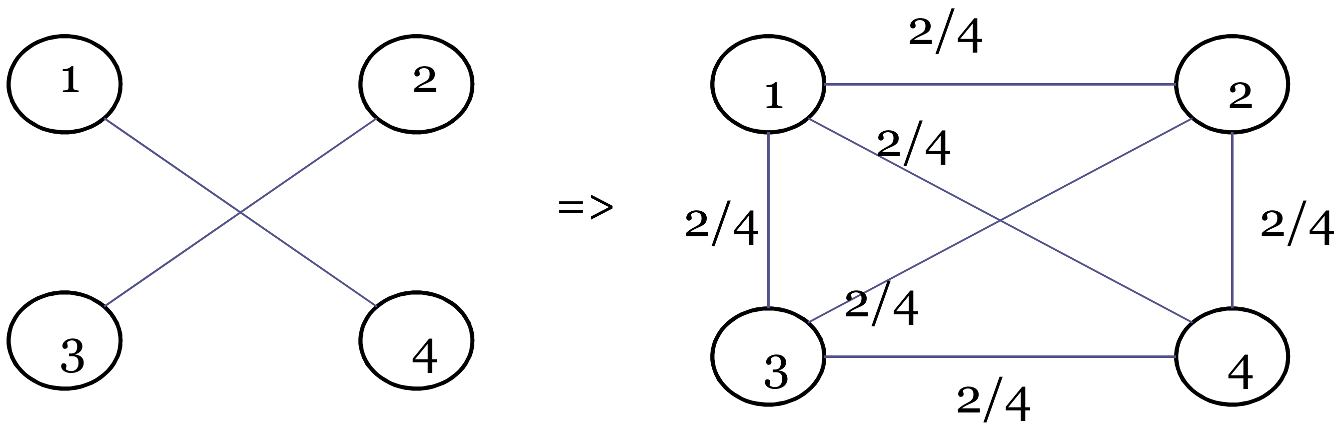
# Linear Vs. Quadratic Wirelength

- Linear WL:
  - Cell 1 is placed right beside A
  - Better proxy for actual WL
- Quadratic WL:
  - Cell 1 will be placed between A and C
  - May be better proxy for delay
  - Long wires get shorter
  - Short wires get longer

# Analytical Placement: Input

- Netlist of cells and connections
- Set of weights representing connectivity
- Between cells:
  - $W_{i,j} = 0$, if cell i not connected to cell j
  - $W_{i,j} > 0$, if cell i connected to cell j
- A "net model"
  - How to represent nets and compute weights
- For A2, use "clique" model

# Clique Net Model




- Weight on each edge = 2/p
- P = # of pins (4 in this example)
- Total weight of net= [p*(p-1)/2]*2/p = p-1
- p-1 is the # of edges in a tree with p nodes

# Star Net Model

- Star model is also popular.
- In the star mode, each net is replaced by a dummy node.  Each cell on the net has an edge to the dummy node.
- Star model results in more variables in AP formulation; clique model results in more non-zero weights.

- One can use the clique model for low-fanout nets; use star model for high-fanout nets.

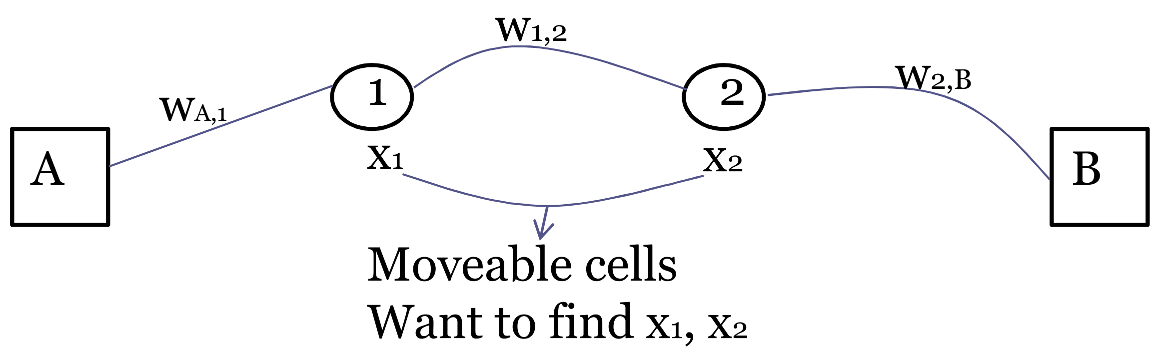
# AP Objective Function

$$\Phi = \sum_{i=1}^{m} \sum_{j=i+1}^{m} w_{i,j}(x_i - x_j)^2 + w_{i,j}(y_i - y_j)^2$$

- $x_i$ = x-position of cell i
- $y_i$ = y-position of cell i

- Can separate into $\Phi_x$, $\Phi_y$ and solve for x, y-dims separately to find min $\Phi_x$ and min $\Phi_y$
- How to minimize the above? Take the partial derivative w.r.t. each $x_i$ and set eqns to 0.
- Note: If there is > 1 connection between 2 cells, add weights of the connection to produce edge weight in AP

# AP Formulation Example




- $\Phi_x = w_{A,1}(x_A - x_1)^2 + w_{1,2}(x_1 - x_2)^2 + w_{2,B}(x_2 - x_B)^2$ - two unknowns $x_1$, $x_2$
- $\delta\Phi_x / \delta x_1 = w_{A,1}(x_A - x_1)(-2) + w_{1,2}(x_1 - x_2)(2) = 0$
  - $\delta\,\Phi_x / \delta\, x_1 = -w_{a,1} \bullet x_A + w_{A,1} \bullet x_1 + w_{1,2} \bullet x_1 - w_{1,2} \bullet x_2 = 0$
- $\delta\,\Phi x / \delta\, x_2 = w_{1,2}(x_1 - x_2)(-2) + w_{2,B}(x_2 - x_B)2 = 0$
  - $\delta\,\Phi x / \delta\, x1 = -w_{1,2} \bullet x_1 + w_{1,2} \bullet x_2 + w_{2,B} \bullet x_2 - w_{2,B} \bullet x_B = 0$

connectivity

$$\begin{pmatrix} w_{A,1}+w_{1,2} & -w_{1,2} \\ -w_{1,2} & w_{1,2}+w_{2,B} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} w_{A,1} \bullet x_A \\ w_{2,B} \bullet x_B \end{pmatrix}$$

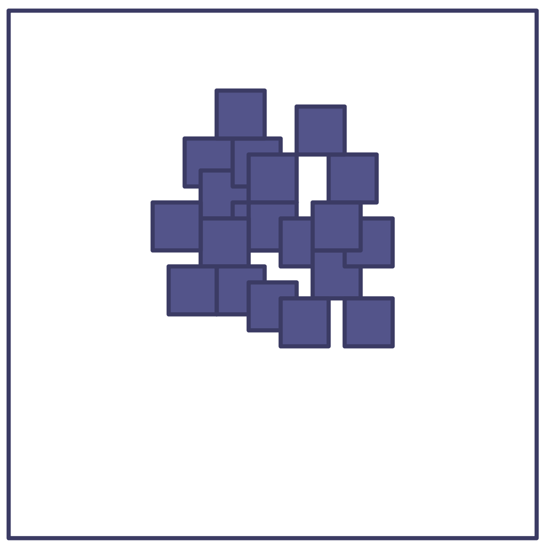
Anchoring vector

# AP Formulation Example

- Linear system, 2 equation and 2 unknowns
- Can write as: $Q\vec{x} = \vec{b}$
- Q is symmetric sparse matrix
- Only one solution to system => easy to solve with standard solver, e.g. conjugate gradient linear system solver.

# AP Formulation

- Note, however, that so far we have not included any constraints that prevent cells from overlapping with one another.
- Solving such a formulation leads to:

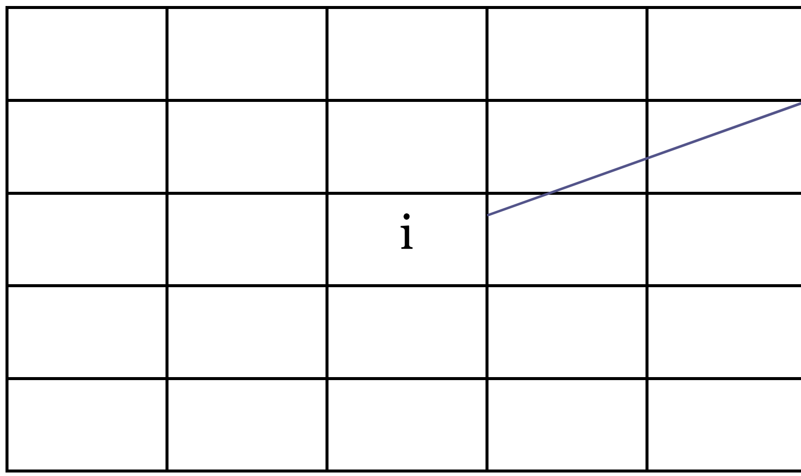


Highly-overlapped placement.

Must alter the formulation to remove the overlaps.

# Removing Overlap: Fast Place (IEEE TCAD 2005)

Superimpose
"bin structure"




Consider bin i,
define utilization:

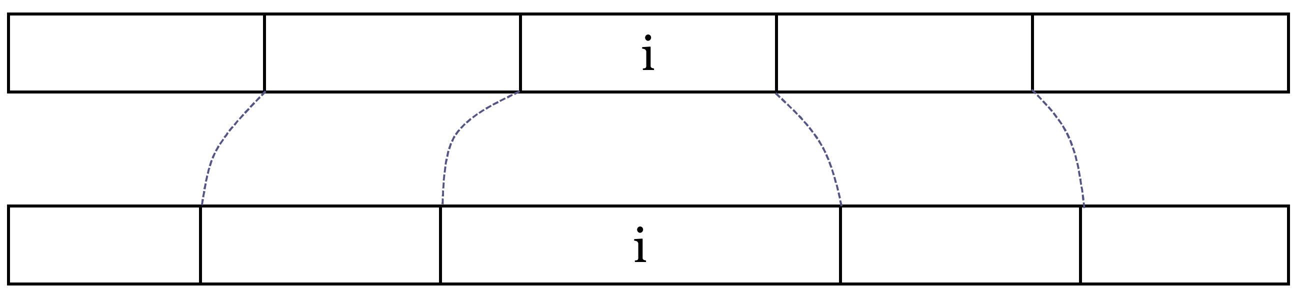
$$U_i = \frac{\sum_{j \in bin_i} area_j}{Area_i}$$

Capacity of bin i

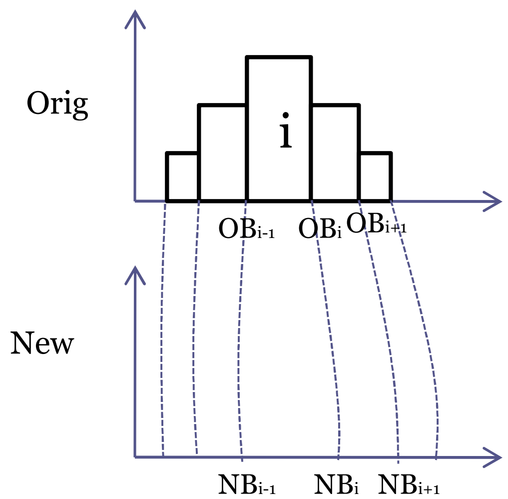
- Shift cells away from high utilization bins towards neighbouring low-utilized bins
  - First shift in x-dim
  - Then shift in y-dim

# Fast Place: Example

- Consider a row of bins in bin structure
- Define a new bin structure corresponding to the utilization
- Map cells linearly b/w 2 bin structures



# Fast Place: Example

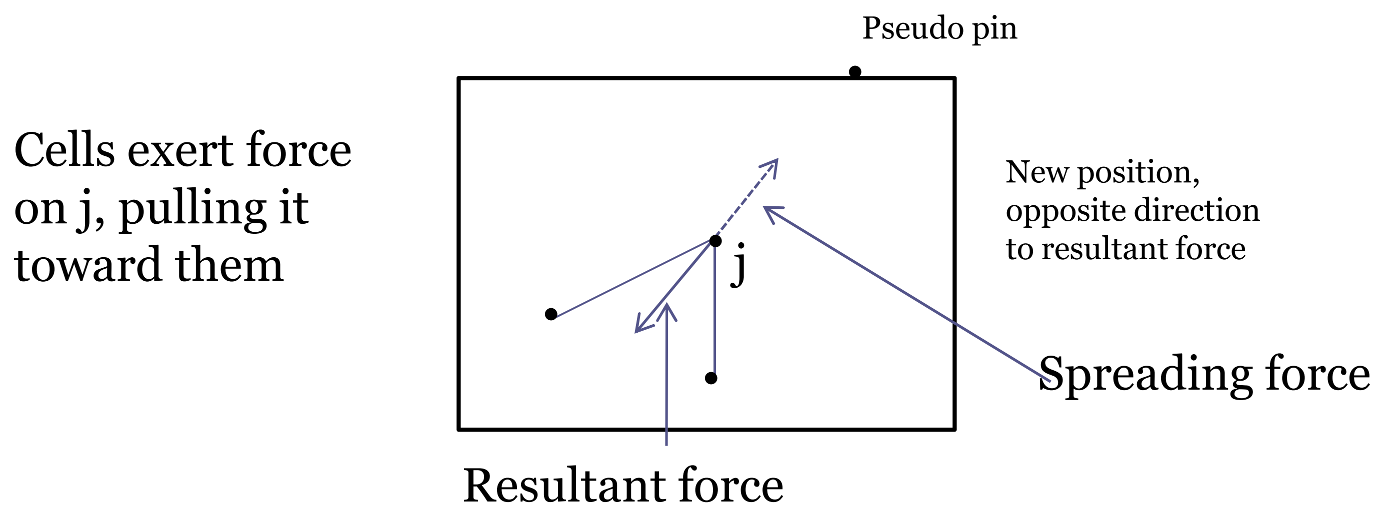



Need to find NB's: New bin boundary

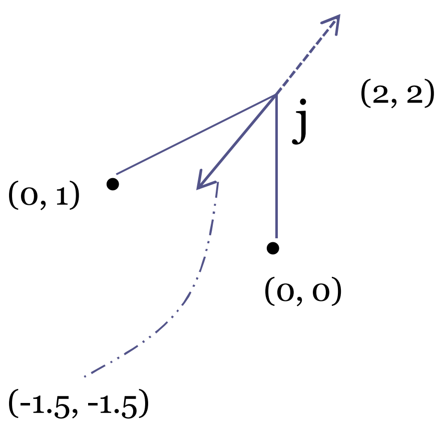
- How to find $NB_i$?
  - $NB_i$ shift East if $U_i$ is high
  - $NB_i$ shift West if $U_{i+1}$ is high
- One way is:
  - $NB_i = (OB_{i+1} \bullet U_i + OB_{i-1} U_{i+1})/(U_i + U_{i+1})$
  - Problem if $U_{i+1} = 0$
  - Then $Nb_i = OB_i + 1$ and $NB_{i+1} = Ob_i$
  - Bin bounds cross over!
- So fast place does:
  - $NB_i = (OB_{i+1} \bullet (U_i + \delta) + OB_{i-1}(U_{i+1} + \delta))/(U_i + U_{i+1} + 2\delta)$
  - $\delta = 1.5$ (typical)

# Fast Place: Example

- So now we have $X_j$’ and $Y_j$’ for cell j (shifted position)
- However, if we re-solve AP, cells flip back to original position
- Need to modify to math to reflect the shifts
  - Introduce a connection to pseudo pin into the AP math



# Resultant Force: Example




- Use Hooke's law to compute resultant force
- F = - K • x (K is spring constant)
- X-component of resultant force
  - (0 – 2) • 0.5 + (1-2) • 0.5 = -1.5
- Y-component of resultant force
  - (0 – 2) • 0.5 + (1-2) • 0.5 = -1.5
- The resultant force:
  x = -1.5 and y = -1.5
- Spreading is in the opposite direction
  - Use spreading force to find pseudo pin location