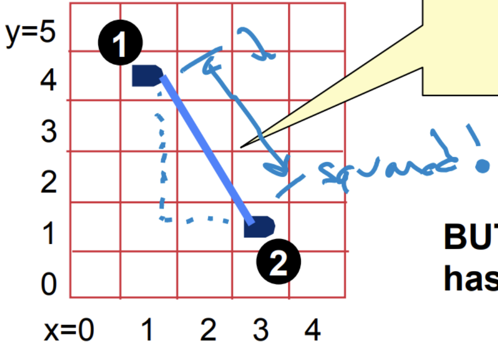
Quadratic Placement

- **Goal: Write an *equation* whose *minimum* is the placement(!)**
  - If you have a million gates, need a million **(xi,yi)** values as result
  - Formulate an appropriate **cost function** for all the gate-level **(xi, yi)**:
    $F(x_1, x_2, \ldots x_{1M}, y_1, y_2, \ldots y_{1M})$
  - …then solve **analytically** for $X^*=(x_1, x_2, \ldots x_{1M})$, $Y^*=(y_1, y_2, \ldots y_{1M})$ to **minimize F( )**
  - The resulting set of values of **X*, Y*** give you the placement of all 1M gates

- **For 2-point net: squared length of "distance" line between points**
  - Quadratic length = $(x1-x2)^2 + (y1-y2)^2$

A "2-point net"

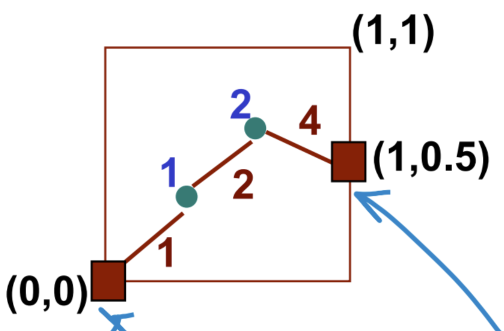


Quadratic wirelength
$=(3-1)^2 + (4-1)^2$
$=13$

BUT… what happens if your net has *more* than 2 points in it?

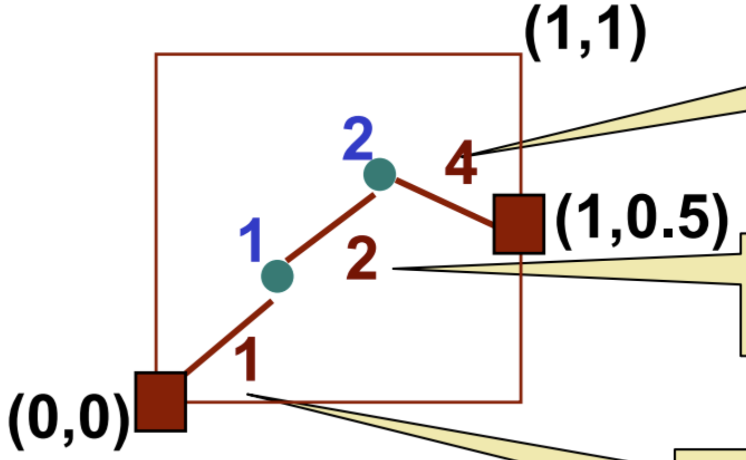
Quadratic Placement

- **Goal: Write an *equation* whose *minimum* is the placement(!)**
  - If you have a million gates, need a million **(xi,yi)** values as result
  - Formulate an appropriate **cost function** for all the gate-level **(xi, yi)**:
    $F(x_1, x_2, \ldots x_{1M}, y_1, y_2, \ldots y_{1M})$
  - …then solve **analytically** for $X^*=(x_1, x_2, \ldots x_{1M})$, $Y^*=(y_1, y_2, \ldots y_{1M})$ to **minimize F( )**
  - The resulting set of values of **X*, Y*** give you the placement of all 1M gates



- **Chip surface is a rectangle**
  - **X** from 0 to 1; **Y** from 0 to 1
  - This it totally arbitrary, btw
- **2 gate "points", index 1 and 2**
- **3 nets, each with a weight**
  - Each net is 2 points to keep manual example small and easy
  - Weights are **1, 2, 4** in diagram
- **2 pads**
  - **Pad** = **fixed** pin (red square) on the edge of the chip. These do not move.

Quadratic Placement



$$4(x2-1)^2 + 4(y2-0.5)^2$$

$$2(x2-x1)^2 + 2(y2-y1)^2$$

$$1(x1-0)^2 + 1(y1-0)^2$$

No terms with an **xi** multiplied by a **yj**

Can **separate** **x** and **y** terms in this sum

Add these together: this is the **Quadratic Wirelength**

So… how do we optimize this…?

Quadratic Placement

- **Basic calculus!** **Differentiate, set derivative to 0, then solve!**
  - But this is multiple variables? So, we do **partial derivatives**, set each to 0, solve

Q(X): $4(x2-1)^2 + 2(x2-x1)^2 + 1(x1-0)^2$

$$\partial Q/\partial x1 = 0 + 4(x2-x1)(-1) + 2(x1) = 6\,x1 - 4\,x2 = 0$$

$$\partial Q/\partial x2 = 8(x2-1) + 4(x2-x1) + 0 = -4\,x1 + 12\,x2 - 8 = 0$$

Q(Y): $4(y2-0.5)^2 + 2(y2-y1)^2 + 1(y1-0)^2$

$$\partial Q/\partial y1 = 0 + 4(y2-y1)(-1) + 2(y1) = 6\,y1 - 4\,y2 = 0$$

$$\partial Q/\partial y2 = 8(y2-0.5) + 4(y2-y1) + 0 = -4\,y1 + 12\,y2 - 4 = 0$$

Quadratic Placement

- **Hey – these are linear equations! We know how to solve these!**

Q(X): $4(x2-1)^2 + 2(x2-x1)^2 + 1(x1-0)^2$

Minimize

$$\begin{bmatrix} 6 & -4 \\ -4 & 12 \end{bmatrix} \begin{bmatrix} x1 \\ x2 \end{bmatrix} = \begin{bmatrix} 0 \\ 8 \end{bmatrix}$$

$x1 = 0.571 \quad x2 = 0.857$

Q(Y): $4(y2-0.5)^2 + 2(y2-y1)^2 + 1(y1-0)^2$

Minimize

$$\begin{bmatrix} 6 & -4 \\ -4 & 12 \end{bmatrix} \begin{bmatrix} y1 \\ y2 \end{bmatrix} = \begin{bmatrix} 0 \\ 4 \end{bmatrix}$$

$y1 = 0.286 \quad y2 = 0.429$

- Two **matrix** equations: $Ax=b_x$ and $Ay=b_y$. If you have **N** gates, matrix is **NxN**
- **Same** matrix for **X,Y, different** **b** vectors. **X, Y, b** vectors also have **N** elements

Quadratic Placement

- **For $Ax = b_x$ vector…**
  - If gate i connects to a pad at (xi, yi) with a wire with weight wi
  - Then set $b_x[i] = wi \cdot xi$

$$\begin{bmatrix} A \end{bmatrix}\begin{bmatrix} x \end{bmatrix} = \begin{bmatrix} b_x \end{bmatrix}$$

i^{th} element of b_x vector

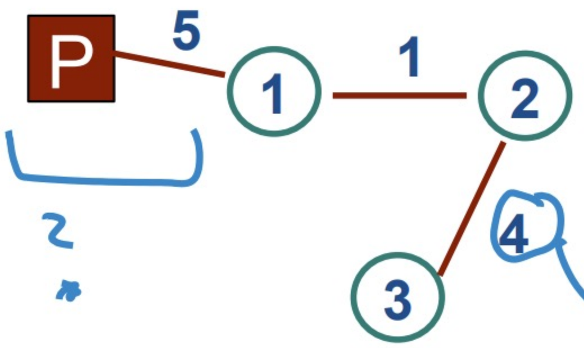
- **For $Ay = b_y$ vector…**
  - If gate i connects to a pad at (xi, yi) with a wire with weight wi
  - Then set $b_y[i] = wi \cdot yi$

$$\begin{bmatrix} A \end{bmatrix}\begin{bmatrix} y \end{bmatrix} = \begin{bmatrix} b_y \end{bmatrix}$$

i^{th} element of b_y vector

Quadratic Placement: What is Matrix **A**?

- **Surprisingly simple recipe to build the required A matrix**
  - First, build the **NxN** connectivity matrix, called **C**
  - If gate **i** has a 2-point wire to gate **j** with weight **w**, the **c[i,j] = c[j,i] = w**, else = **0**
  - New (bigger) example, with 3 gates, 3 wires (with weights) and 1 pad (**P**)

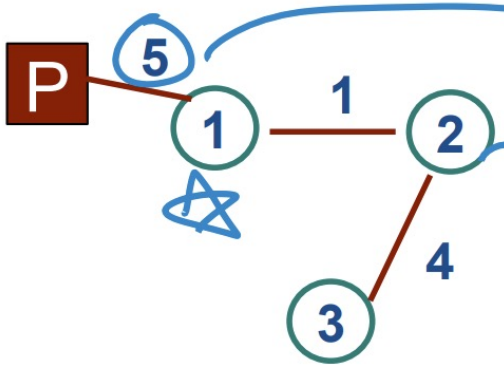


$$C = \begin{array}{cc} & \begin{matrix}1 & 2 & 3\end{matrix} \\ \begin{matrix}1\\2\\3\end{matrix} & \begin{bmatrix}0 & 1 & 0\\1 & 0 & 4\\0 & 4 & 0\end{bmatrix}\end{array}$$

Note: **C** matrix ignores the pads

Quadratic Placement: What is Matrix **A**?

- **Use the connectivity C matrix to build A matrix**
  - Elements **a[i,j] not** on the matrix diagonal are just **a[i,j] = -c[i,j]**
  - Elements **on the diagonal** are **a[i,j]= $\sum_{j=1,n}$ c[i,j] + (weight of any pad wire)**
    - …ie, add up the i^{th} row of C **and then** add in weight on a (possible) wire to pad
  - Same example with 3 gates, and 1 fixed pad.



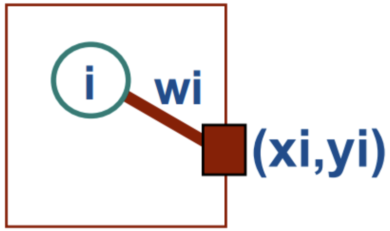
$$C = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 4 \\ 0 & 4 & 0 \end{pmatrix} \qquad A = \begin{pmatrix} 6 & -1 & 0 \\ -1 & 5 & -4 \\ 0 & -4 & 4 \end{pmatrix}$$

a[i,j]

Note: **A** matrix accounts for pads

diagonal

How to Build **b** Vectors?



- **For Ax = b_x vector…**
  - If gate **i** connects to a pad at **(xi, yi)** with a wire with weight **wi**
  - Then set **b_x[i] = wi • xi**

$$\begin{bmatrix} A \end{bmatrix} \begin{bmatrix} x \end{bmatrix} = \begin{bmatrix} b_x \end{bmatrix}$$

i^{th} element of b_x vector

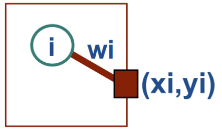
- **For Ay = b_y vector…**
  - If gate **i** connects to a pad at **(xi, yi)** with a wire with weight **wi**
  - Then set **b_y[i] = wi • yi**

$$\begin{bmatrix} A \end{bmatrix} \begin{bmatrix} y \end{bmatrix} = \begin{bmatrix} b_y \end{bmatrix}$$

i^{th} element of b_y vector

Quadratic Placement

How to Build b Vectors?



- **For Ax = b_x vector…**
  - If gate **i** connects to a pad at **(xi, yi)** with a wire with weight **wi**
  - Then set **b_x[i] = wi • xi**

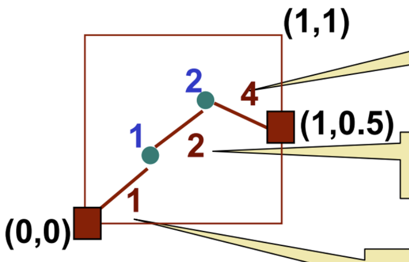
$$\begin{bmatrix} A \end{bmatrix}\begin{bmatrix} x \end{bmatrix} = \begin{bmatrix} b_x \end{bmatrix}$$

i^{th} element of b_x vector

- **For Ay = b_y vector…**
  - If gate **i** connects to a pad at **(xi, yi)** with a wire with weight **wi**
  - Then set **b_y[i] = wi • yi**

$$\begin{bmatrix} A \end{bmatrix}\begin{bmatrix} y \end{bmatrix} = \begin{bmatrix} b_y \end{bmatrix}$$

i^{th} element of b_y vector



$4(x2-1)^2 + 4(y2-0.5)^2$

$2(x2-x1)^2 + 2(y2-y1)^2$

$1(x1-0)^2 + 1(y1-0)^2$

No terms with an **xi** multiplied by a **yj**

Can **separate x** and **y** terms in this sum

Add these together: this is the **Quadratic Wirelength**

So… how do we optimize this…?

Q(X): $4(x2-1)^2 + 2(x2-x1)^2 + 1(x1-0)^2$

Minimize

$$\begin{bmatrix} 6 & -4 \\ -4 & 12 \end{bmatrix}\begin{bmatrix} x1 \\ x2 \end{bmatrix} = \begin{bmatrix} 0 \\ 8 \end{bmatrix}$$

x1 = 0.571 x2 =0.857

Q(Y): $4(y2-0.5)^2 + 2(y2-y1)^2 + 1(y1-0)^2$

Minimize

$$\begin{bmatrix} 6 & -4 \\ -4 & 12 \end{bmatrix}\begin{bmatrix} y1 \\ y2 \end{bmatrix} = \begin{bmatrix} 0 \\ 4 \end{bmatrix}$$

y1 = 0.286 y2 =0.429

- Two **matrix** equations: **Ax=b_x** and **Ay=b_y**. If you have **N** gates, matrix is **NxN**
- **Same** matrix for **X,Y, different** b vectors. **X, Y, b** vectors also have **N** elements