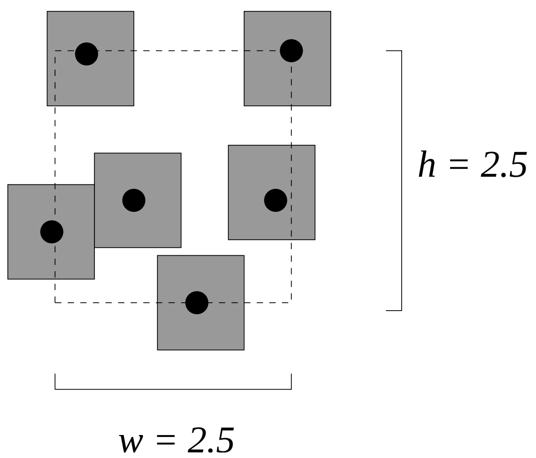
Optimization in VLSI design, E2004
- Exercises for week three.

Jens Egeblad

September 23th, 2004

1 Exercise 1 - LP formulation of bounding-box net-length (45 min)

The bounding-box net-length model estimates the required wire-length of a net by half the perimeter of the bounding box of the pins of the net. Example:



On the figure the shaded rectangles are modules and the black circles are pins. All the pins are part of a net and the net's bounding-box is shown. The width of the bounding-box is $w = 2.5$ and the height is $h = 2.5$. Therefore in this case the bounding-box net-length is $w + h = 5$.

Assume that all pins are placed in the center of modules and that we are given a problem with two nets n_1, n_2 and three modules $m_1, \ldots, m_3$. Let (x_i, y_i) for $i = 1, \ldots, 3$ describe the *center* of module i. We require that module 1 has it's center at $(1, 1)$. n_1 is connected to modules m_1 and m_2 and n_2 is connected to modules m_2 and m_3.

Now the following is an LP formulation of the associated *unconstrained* placement problem with bounding-box net-length.

$$\begin{array}{l} \text{Minimize} \;\; n_{1r} - n_{1l} + n_{1t} - n_{1b} + n_{2r} - n_{2l} + n_{2t} - n_{2b} \\ \text{s.t.} \end{array}$$

$$\begin{array}{rcl} x_1 & \geq & n_{1l} \\ x_1 & \leq & n_{1r} \\ x_2 & \geq & n_{1l} \\ x_2 & \leq & n_{1r} \\ y_1 & \geq & n_{1b} \\ y_1 & \leq & n_{1t} \\ y_2 & \geq & n_{1b} \\ y_2 & \leq & n_{1t} \\ x_2 & \geq & n_{2b} \\ x_2 & \leq & n_{2t} \\ x_3 & \geq & n_{2b} \\ x_3 & \leq & n_{2t} \\ y_2 & \geq & n_{2b} \\ y_2 & \leq & n_{2t} \\ y_3 & \geq & n_{2b} \\ y_3 & \leq & n_{2t} \\ x_1 & = & 1 \\ y_1 & = & 1 \end{array} \quad , \qquad (1)$$

where $x_2, y_2, x_3, y_3, n_{1l}, n_{1r}, n_{1b}, n_{1t}, n_{2l}, n_{2r}, n_{2b}, n_{2t}$ are free variables and $x_1 = 1$ and $y_1 = 1$. n_{il} and n_{ir} are the left and right x-coordinates of the the bounding-box for net i and n_{ib} and n_{it} are the bottom and top y-coordinates of bounding-box for net i.

Q1: You can determine the optimal values of (1) by inspection or other means. Remember modules are allowed to overlap. What is the optimal position of modules m_2 and m_3?

1.1 Splitting the problem in two.

(1) can actually be split in two linear programs which can be optimized separately.

Q2: Formulate one of these two programs.

Help: The objective function can be rewritten as:

$$\text{Minimize} \;\; (n_{1r} - n_{1l} + n_{2r} - n_{2l}) + (n_{1t} - n_{1b} + n_{2t} - n_{2b}).$$

where the first part only depends on the x-coordinates and the second part only depends on the y-coordinates of the modules.

1.2 Non-overlapping constraints.

Now assume that module m_i has width $w_i > 0$ and height $h_i > 0$. Unfortunately the solution to (1) contains overlap.

Q3: If you could add any number of constraints and *integer* variables to (1), how would you enforce that module m_1 and m_2 do not overlap in the solution?

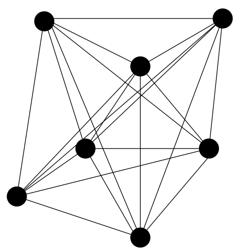
Help: To formulate expression like: "$x_1 + a \leq x_2$ or $x_2 + b \leq x_1$" you can add the constraints:

$$\begin{aligned} x_1 + a &\leq x_2 + t \cdot M \\ x_2 + b &\leq x_1 - t \cdot M + M \\ t \in \{0, 1\} & \end{aligned}$$

where M is a *very* large number. Also remember that there are *two* dimensions!

Q4: Given m modules how many integer variables and constrains do you need to ensure that there is no overlap?

Exercise 2 - The clique net-length (45 min)



Given a net n which connects modules $M(n)$ the quadratic clique net-length $CL(n)$ of n estimates the wire-length required to route the net n. It does this by adding the shortest distance between all pins together. If we assume that a pin is placed at the center of it's module and that the coordinates of the center of each incident module $m \in M(n)$ are (x_m, y_m), then we may express the clique net-length as:

$$CL(n) = \frac{1}{2} \sum_{m_1 \in M(n)} \sum_{m_2 \in M(n)} (x_{m_1} - x_{m_2})^2 + (y_{m_1} - y_{m_2})^2.$$

As an alternative to the clique net-length the quadratic star-netlength:

$$ST(n) = \sum_{m \in M(n)} (x_m - x_s)^2 + (y_m - y_s)^2,$$

where $x_s = \frac{1}{|M(n)|} \sum_{m \in M(n)} x_m$ and $y_s = \frac{1}{|M(n)|} \sum_{m \in M(n)} y_m$ is often used in the litterature.

Q5: Can you come up with a reason why one would prefer the quadratic *star* net-length to the quadratic *clique* net-length?

1.3 The clique net-length is separable

$ST(n)$ can be rewritten as a sum of two functions: one which depends on the x-coordinates of the modules and one which depends on the y-coordinates:

$$ST(n) = ST_x(n) + ST_y(n),$$

where $ST_x(n) = \sum_{m \in M(n)} (x_m - x_s)^2$ and $ST_y(n) = \sum_{m \in M(n)} (y_m - y_s)^2$.

Q6: Can you do something similar with CL(n)?

1.4 Optimal position of *one* module

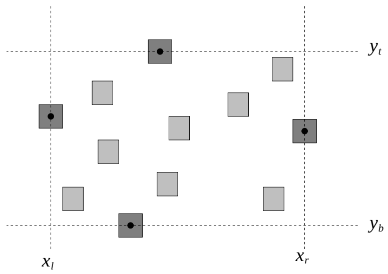
Assume that we have modules m_1, m_2 and m_3 with coordinates. (x_1, y_1), (x_2, y_2) and (x_3, y_3). Futher m_1 must be placed at $(0,0)$ and m_3 must be placed at $(1,1)$. The modules are connected by two nets n_1 and n_2. n_1 connects modules m_1 and m_2. n_2 connects modules m_2 and m_3. The weights of both nets are 1. We now wish to determine the optimal position of the module m_2. That is we wish to determine (x_2, y_2) such that

$$CL(N) = CL(n_1) + CL(n_2)$$

is minimal.
Q7: What is the optimal value of (x_2, y_2) given that m_1 and m_3 are fixed?

Help: You may be able to use the result from Q6 combined with some simple calculus!

1.5 Property of optimal solutions



The clique net-length has a nice property. Assume that a VLSI placement problem has a number of fixed modules (the four darkest on the figure) and that the x-coordinate of the left-most pin of the fixed modules is x_l, the right-most is x_r the y-coordinate of the top-most is y_t and the y-coordinate of the lowest pin is y_l, then the optimial solution to the VLSI placement problem with quadratic clique net-length will have all module centers within these boundaries.

Q8: Argue that an optimal solution to the placement problem with quadratic clique net-length cannot have a module center outside these boundaries. Why not?

Exercise 3 - Fast Fourier Transform (15 min)

In the article "Generic Global Placement and Floorplanning" by Eisenmann and Johannes the real-value force function:

$$\mathbf{f}(x,y) = \frac{k}{2\pi}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} D(x',y')\frac{\mathbf{r}-\mathbf{r}'}{|\mathbf{r}-\mathbf{r}'|^2}dx'dy',$$

is evaluated on a discrete bin-structure. The force at the center of each bin (i,j) is evaluated by:

$$\mathbf{f}_{ij} = \frac{k}{2\pi}\sum_{i'=1}^{m}\sum_{j'=1}^{m} D_{i'j'}\frac{\mathbf{r}_{ij}-\mathbf{r}_{i'j'}}{|\mathbf{r}_{ij}-\mathbf{r}_{i'j'}|^2}w \cdot h,$$

where $D_{i'j'}$ is the density of bin (i,j) and r_{ij} is the center coordinate of bin (i,j).

The Fast Fourier Transform can be used to evaluate

$$f(i,j) = \sum_{i'=1}^{n}\sum_{j'=1}^{n} g(i,j) \cdot h(i',j')$$

for *all* $i,j = 1,\ldots,n$ in $O(n^2 \log n)$ time.

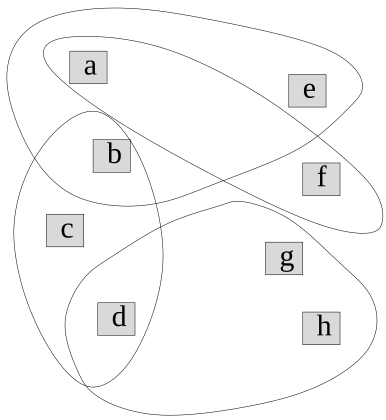
Q9: How can you use the Fast Fourier Transform to evaluate the force function $\mathbf{f}(i,j)$ for each bin (i,j)? (write up the functions g and h)

Exercise 4 - Graph Partitioning (15 min)

Given a hypergraph $H(V,E)$ a bisection of the graph is a division of the vertices from V into two disjoint subsets V_1 and V_2 such that $|V_1| = |V_2|$. Then quality of the partition can be evaluated solely based on the number of edges which have vertices in both V_1 and V_2.

$$cut(V_1,V_2) = |\{e \in E | e \cap V_1 \neq \emptyset \text{ and } e \cap V_2 \neq \emptyset\}|.$$

(Remember that a hyperedge can contain more than one vertex.)



Q10: Find an optimal bisection of the graph from the figure. How do you know it is optimal?

Exercise 5 - Design a global placement heuristic (30 min)

Q11: Briefly describe how you would design your own *global* placement heuristic?

(It may be helpful to consider some of the topics covered by exercises 1-4)

Q12: How large problems can you handle?