University of Toronto, Faculty of Applied Science and Engineering
Department of Electrical and Computer Engineering

ECE 1387 - CAD for Digital Circuit Synthesis and Layout
Assignment #2 – Analytical Placement and Spreading

October 2015 J. Anderson

Assignment Date: October 9, 2015
Due Date: October 23, 2015, at the beginning of class.
Late Penalty: **-2 marks per day late, with total marks available = 20**

You are to write an implementation of a basic analytical placer (AP), with overlap removal (spreading). As described in class, you will formulate the placement problem mathematically as a system of linear equations to be solved. You will use an existing package (UMFPACK) to solve the linear system (see announcement on Piazza page).

Your program should display its progress and results using the same graphics package as used in Assignment #1 (available on course webpage). Your graphics should show the placement results and the connectivity between blocks (rat's nest of wires). Blocks (cells) should appear as points in your placement and be labeled with block numbers (see below).

The netlist file input format has two sections. The two sections are separated from one another by a –1 appearing by itself on a line. The first section specifies the blocks to be placed and the connectivity between them. Each line has the following form:

blocknum $netnum_1$ $netnum_2$ $netnum_3$ ... $netnum_n$ -1

Where blocknum is a positive integer giving the number of the cell, and the $netnum_i$ are the numbers of the nets that are attached to that block. Every block that has the same $netnum_i$ on its description line is attached. Note that each block may have a different number of nets attached to it. Each line is terminated by a –1.

Example input file:

```
1 2 3 4 -1
2 5 4 -1
3 5 6 2 -1
4 6 3 -1
 -1
1 50 0
4 0 50
-1
```

In this example, block 1 is connected to nets 2, 3 and 4. Note that each net may be connected to more than two blocks (that is, there are multi-fanout nets). Also note that net numbers are not related to block numbers.

As discussed in class, the AP formulation requires there to be a set of pre-placed (fixed) objects, normally I/Os. The second section of the netlist file specifies the placement of fixed objects. It has the following form:

blocknum x_position y_position

In the above example, block 1 is pre-placed at the position with x = 50, y = 0. The list of fixed objects is terminated by a –1 by itself on a line.

You should run your placer on the Assignment #2 test circuits provided on the course web page.

What to do and what to hand in?

Your placer must run on the ECF network. Instructions for electronic submission of your placer (including source code) will be posted on the course's Piazza page close to the assignment deadline.

Your report should include a short description of the flow of your program, the main routines and what they do, assuming that I have basic knowledge of analytical placement.

1. **Do for all circuits:** Formulate and solve the analytical placement problem assuming the *clique* net model[1]. Do not do any overlap removal in this step. Your program should display the placement and rat's nest (wires between cells) using the graphics package. Hand in a plot of the placement results. Your program should also compute the half-perimeter bounding box (BB) wirelength (WL) of the placement. Hand in a table showing the half-perimeter BB WL for each placed test circuit.

2. **Do for test circuit 2 ONLY:** I/O placement is a key aspect of placement. Here you will examine the impact of I/O placement on the placement results. Assuming the same set of fixed locations, have your program randomly interchange some of the fixed blocks with one another and repeat Step #1 above for each different I/O placement. Discuss the effect of the different I/O placements on the WL of the placement. Find the average, minimum, and maximum WLs across a number of random I/O placements (try enough so the average doesn't change much). Propose ideas on how one might determine a "good" I/O placement.

3. **Do for all circuits:** Implement a simple form of overlap removal. Given that the placement area spans from (0,0) to (100,100), use the results of Step 1 to divide the non-fixed blocks into four groups of roughly equal size. Each group will consist of the blocks closest to each quadrant of the placement area. That is, one group will represent the ~n/4 blocks closest to the lower-left corner of the placement; a second group will represent the ~n/4 blocks closest to the top-left corner of the placement, and so on. Having divided the blocks, reformulate the placement problem as in Step 1, but with a modification: In your reformulation, introduce 4 *new* (artificial) fixed blocks, placed at the **four centres** of the die quadrants (i.e., one at (25,25), one at (25,75), etc.). Introduce (artificial) two-pin nets from each new fixed block to each block in the group corresponding to the fixed block's quadrant. Solve the formulation, compute its BB wirelength, and display and plot the results. Describe how you partitioned the blocks. Hand in a table showing the WLs. Do not include the WL of the artificial connections in your WL numbers. What happens to WL when the new fixed blocks are introduced? Experiment with changing (increasing/decreasing) the weights of the artificial two-pin nets; comment on the WL results when different weights are used.

4. **Do for test circuit 2 ONLY:** Repeat step #3 recursively to perform further spreading by continuing to modify the mathematical formulation. For example, take the group of blocks you assigned to the lower-left quadrant in Step #3, and divide this group into four sub-groups, each corresponding to a sub-quadrant of the lower-left quadrant. Introduce fixed blocks at the centers of the four sub-quadrants (i.e., one at (12.5,12.5), one at (37.5,12.5), etc.). Introduce (artificial) two-pin nets from each new fixed block to each block in the corresponding sub-group. Do the same for the other quadrants of the die. Solve the new placement formulation.

   Assuming that the placement die area is partitioned into a set of 100 regions (see figure below), *R*, each with dimensions of 10x10 units, continue spreading recursively (dividing quadrants into sub-quadrants, inserting fixed blocks and artificial nets, and re-solving) until the following stopping criterion is met:

---

[1] In the clique model, a net with p pins is represented as a complete graph (clique) with p*(p-1)/2 edges. Each edge in the complete graph has weight of 2/p. For example, a net with 2 pins has 1 edge with edge weight = 1. A net with 4 pins has 6 edges with edge weight = 2/4.

$$Overlap = \frac{\sum_{r \in R} \max(0, N_r - 2)}{N} \leq 0.15$$

where N is the total number of moveable objects (non-I/O objects), and N_r is the total number of moveable objects placed in region r. N_r should be computed assuming that objects are points having zero area. The intuition behind the above relation is that each 10x10 region can accommodate two placed objects. You should aim to meet the above criterion with minimal damage to BB WL. Provide plots showing the progression of the spreading process. Provide a plot of the spread placement that meets the stopping criteria. Report the BB WL when the stopping criterion is met. In the report, briefly describe your implementation and highlight any clever techniques you used. **In class, I will report on the results achieved by everybody and mention the best WL achieved.**

OPTIONAL: For this part, instead of applying step #3 recursively as instructed above, you may use ANY spreading strategy you wish, including SimPL, as described in class, along with the bound-to-bound net model (also optional). If you decide to go this route, briefly describe your solution and optimizations. Note that it is not my expectation (or a requirement for full marks) that everyone opt for this route and implement SimPL or some other more advanced algorithm. This option is simply provided for those individuals who are really interested in AP.