ECE 5460/6460: VLSI Design Automation

Homework 4

Due: 10/23/2014

1. (100 points) **Programming assignment:** Here you are required to program the **entire** simulated annealing algorithm for slicing floorplan. The algorithm is described in detail in the textbook (page 121), and is also discussed in class. To develop this algorithm, you will use the **cost()** routine developed in HW3.
    a. Use the **cost()** algorithm as the subroutine to evaluate a given topology, implement the simulated annealing algorithm to find a good floorplan for a given set of modules. The objective function is the area of the floorplan (Cost = Area). For this assignment, we will **not** consider the wire length for optimization. Also, all the modules will be hard modules with free orientations.

Input Formats:

- - - - - - - - - - - -

The input consists of two parts:

1. Parameters (these will be hard coded in your code).
2. Names, areas and aspect ratios of the modules – You need to parse these from your input file. Note that we will test the code on other input files with the same format as input_file.txt used for HW3.

More detailed description of parameters.

1. Parameters

They are of the following form:

```
2.0     (*  chipratio  *)
10      (*  nmoves     *)
0.85    (*  ratio      *)
-1      (*  t0         *)
0.005   (*  lambdatf   *)
```

```
3               (*      iseed           *)
6               (*      n               *)
0.99            (*       P              *)
0.001           (*    epsilon           *)
```

chipratio**:** Final chip ratio must be between 1/chipratio and chipratio; chipratio must be greater than 1.0

nmoves**:** Number of attempt moves per temperature iteration. Use 10 as the default value. Larger values of nmoves in general produce better results. The total execution time of the program grows linearly with respect to nmoves.

Ratio**:** Ratio for decreasing temperatures. Use 0.85 as the default value. Larger values of ratios in general produce better results and have much longer execution time.

t0**:** Initial temperature. If the input t0 is negative (e.g. -1), the program will compute an initial temperature (in this case you will need to add a negative sign to make the initial temperature positive); otherwise the program will use the input positive t0 as the initial temperature. Default value is -1.

lambdatf**:** The program will change ratio to 0.1 when temperature falls below lambdatf * t0. After ratio is set to 0.1, the program in general will terminate very quickly.

iseed**:** Seed for random number generator.

n**:** Number of modules (this will depend on your input file).

P: Initial probability of accepting uphill moves

N: Calculated using N = nmoves * n

Instructions on submission:

1. PLEASE USE INPUT FILE “input_file.txt” AS AN INPUT FOR YOUR PROGRAM (we will however test your program with other input files as well)
2. Output: Please output the **initial topology**, **initial cost**, and the **final optimized topology and the final cost in your program** (after annealing). Do not print out topology or cost in intermediate iterations.
3. Please submit a hardcopy of the printout of your program in class and/or blackboard, and submit the **.exe** file through blackboard.

This assignment will be graded on the following criteria

a) Correct execution [70 points] - we will test your programs for different test files.
b) Code commenting [10 points] - will be awarded for properly commenting the code.
c) Solution quality [20 points] – We will check the performance of the annealer from the quality of the final floorplan.