

ECE 1387 - CAD for Digital Circuit Synthesis and Layout

Exercise #3 – Fiduccia-Mattheyses (FM) Partitioning

Spring 2009

J. Anderson

Assignment Date: April 3, 2009

Due Date: April 17, 2009 (by email to instructor)

Late Penalty: -1 mark per day late, with total marks available = 10

Fiduccia-Mattheyses (FM) partitioning was a breakthrough in partitioning research as it allowed a good-quality bisection to be computed in $O(p)$ time, where p is the number of pins in the circuit being partitioned. In this exercise, you will experiment with the FM heuristic and gain some familiarity with the algorithm.

On the course website, you will find a `.zip` file containing a C++ implementation of FM. The code was written by researchers at UCLA and the University of Minnesota. I have modified it slightly to be able to compile on EECG Solaris and Linux machines. Download the code and compile it. The `makefile` should work on EECG and produce a binary called `dohpart`. Read the `README` file in the directory to find out how to run the binary. Refer to the original FM paper on the course website for detailed information on FM partitioning.

There are 4 benchmark circuits on the website called `ibm02.net`, `ibm01.net`, `ibm12.net` and `ibm08.net`. These benchmarks are (obviously) courtesy of IBM and a description of the benchmarks and their format can be found here:

<http://vlsicad.ucsd.edu/UCLAWeb/cheese/ispd98.html>

What to do and what to hand-in?

1. Run FM on the four benchmarks with 5 different random seeds with the balance condition set to 0.45. Record the crossing count (“cutset”) for each of the runs. You should have 5 data points for each benchmark. How much does the crossing count vary with random seed? Where, in the FM algorithm, do random numbers come into play?
2. Repeat #1 above with the balance condition set to 0.25. Comment on whether and by how much crossing count depends on the balance condition.
3. The FM heuristic executes a number of “passes”. What constitutes a pass of FM? Modify the `part(void)` method in the C++ code in the `hgraph.cc` source file so that a maximum of 5 passes of FM are executed. Repeat step #1 above (with the balance condition set to 0.45). Comment on whether, and by how much, the repeated passes of FM affects the final crossing count.
4. How does the FM heuristic do hill climbing? Start with a fresh copy of the FM code, and modify the `getprefix(int&)` method in the `hgraph.cc` file so that hill climbing is turned off. Comment on whether, and by how much, hill climbing affects the final crossing count.