Weekly Report for Yu Hu's work in week10

March 20, 2005

1. **Add a new heuristic in routing.** When I test grid.10 and grid.20 in routing, I found that running time is much longer than I expected. So I add a new heuristic, **tree grows towards source**, to narrow the searching space further. The experimental results show that this idea can prune about 3/4 options. When I test grid.10, the running time decreases from 2346s to 1952s. However, I can not conquer grid.20 by even adding this heuristic.

2. **Clean up my code (buffer insertion and routing), finish the experiments, and write the code document.** I've tested all examples appeared in King Ho's DAC'05 paper for buffer insertion (s1.cmp C s9.cmp) and tree construction (grid.2 C grid.6) with both single-Vdd and dual-Vdd. The results are ok. Also I've tested some extremely large cases (r1.cmp - r5.cmp for buffer insertion, and grid.10 for routing). I can not get a result for the largest case for routing, grid.20, at present.

   I write a code document for my implementation and modification in BIC for fast buffer insertion and buffered tree construction. A simple user manual and the comments for some important classes are given in this document.

3. **Read several papers about Flip-Flop insertion.** I've read through the FF insertion paper (C47. L. Simonson, K. Tam, N. Akkiraju, M. Mohan and L. He, "Leveraging Delay Slack in Flip-flop and Buffer Insertion for Power Reduction", International Symposium on Quality Electronic Design, pages: 69-74, March 2004.). I think that FF insertion is much like the buffer insertion.

   The main contribution of FF insertion in my ICCAD'05 submission will be **dual-Vdd FF insertion and FF-tree construction**. The problem formulation can be similar as the LPDP (Low Power Dynamic Programming) problem in paper C47, which is minimize power while satisfying RAT and latency (FF number) at sinks.

   There two comparisons should be made for FF insertion,

   (a) Compare to Maximum Slack FF-tree construction to show the capability of power minimization.

   (b) Compare the single-Vdd and dual-Vdd of my lower power FF-tree construction to show the effectiveness of dual-Vdd FF insertion.

   To generate test cases, it's needed to give a latency (the number of FF should be inserted in upstream) in each sink. Also, I need a FF library, which includes input capacitance, output resistance, power per switch, and setup time. If we need to consider dual-Vdd FF insertion, a dual-Vdd version of FF library is needed.

4. **Begin to write the outline of ICCAD submission.** As Prof. He's advice, the tentative title is: prediction and sampling based algorithms for power optimal buffer insertion and buffered tree construction. The organization is as follows,

   Sect 1 Introduction

   Sect 2 single-vdd buffer insertion o modeling and problem formulation
   detailed review of exisiting algorithm (with pesudo-code for the base algorithm)
   new algorithms
   experiment results

   Sect 3 single-vdd buffered tree construction
   problem formulation and algorithms
   experiment results

Sect 4 dual-vdd buffering
modeling and problem formulations
extension to algorithms
experiments

A draft of my paper will be completed within the next week.