Physical Design PA1 Report

吳牧庭 R08943094

Overall flow

1. Find a balanced initial partition
2. Compute cell gain
3. Build bucket list
4. Partition iteration
5. Restore best move
6. Repeat 2. Until the result is satisfactory

Partition iteration flow

1. Get max legal cell
2. Move and lock cell
3. Update gain
4. Update the best move
5. Repeat 1. Until no more legal cell to move

Initial partition

Since there's no guarantee that an initial partition with less cut is bound for a better final partition, I just roughly split the cells in half by their ids. The rationale behind it is that cell ids are enumerated according to the sequence they are added on a net. Cells with closer ids is more likely to be on the same net, and tend to cluster to the same partition.

Bucket-list

A vector of doubly linked-lists connecting cells of the same gain. With 2 bucket-lists, we can decide which cell to move in constant time. The bucket-list is implemented by vector to achieve random access.

Simulated annealing

Partitioning stops when extra_iter == max_extra_iter, where max_extra_iter = 5. If max_acc_gain==0, then extra_iter += 1, else extra_iter = 0. This terminating condition leads to longer runtime yet lower cutsize.