CS6135 VLSI Physical Design Automation

Homework 4: Placement Legalization

108062537 魏聖修

2.

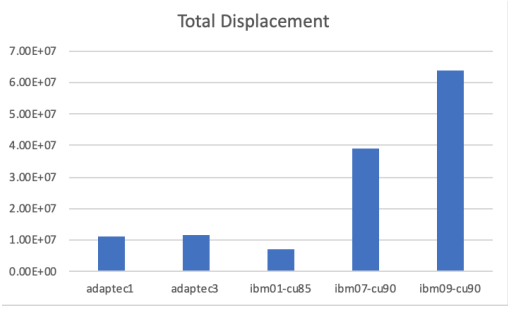
compile : make

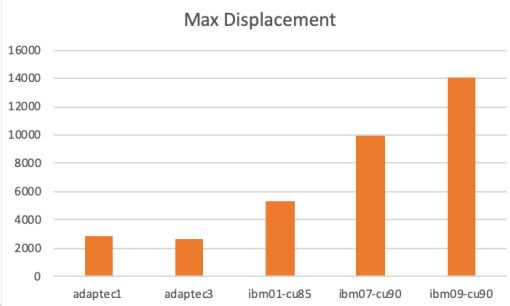
execute : ./hw4 <aux>

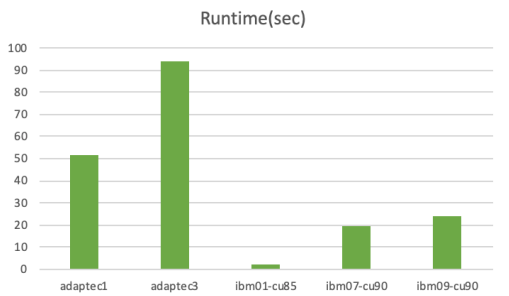
```
[[zscaxd5651@ic56 src]$ make
g++ -g -std=c++11 -O3 main.cpp -o  hw4
[zscaxd5651@ic56 src]$ time ./hw4 ../testcase/adaptec1/adaptec1.aux
```

3.

	Total Displacement	Max Displacement	Runtime(sec)
adaptec1	1.101E+07	2826.72	51.57
adaptec3	1.161E+07	2607.95	93.83
ibm01-cu85	0.719E+07	5337.06	2.18
ibm07-cu90	3.882E+07	9900.03	19.23
ibm09-cu90	6.389E+07	14050.1	23.96







4.

(1) Preprocess:

CutSubRows() :

首先去判斷是否存在 blockage，如果存在則依照 blockage 將 row 切成 subRow。否則直接將 Row 當成 subrow，之後演算法只會對 subrow 做操作。

Modify() :

為了滿足各個 site 的規格，要將 Global Placement 的座標先做一些前處理。主要是以目前座標(x, y)，都除以 sitewidth，再將結果四捨五入，最後在乘回 sitewidth，以對齊各個 site 座標。

還有由於某些 case 的 subrow 較多，為了避免運行時間太久，我會先選擇離目前 cell 最近的幾個 subrow 去做紀錄，目前設定 subrow 和 node 距離 50 以內。因為大部分的 node 最佳擺放位置都在高度距離較近的 subrow。若某個 node 在 50 範圍內真的塞不下，再去判斷，並改成對所有 subrow 進行擺放。此方法可以大幅減少計算，以此加速。結果比較在 part5。

(2) Algorithm:

SortNode() :

將 node 依照 x 座標由小到大 sort。

PlaceRow(i, j) :

將 node(i)試擺放在 subrow(j)上。大致相似於 paper 的實作，主要有

AddCell(): 如果某 node 和現在 subrow 上的任何一個 cluster 重疊，merge 這個 node 和這個 cluster 並更新座標。若無，則將此 node 視為一個新的 cluster。

Collapse(): AddCell()完成後，產生出來的那個新的 cluster 可能和其他的 cluster 產生重疊的情況，因此必須對他們全部做相應的調整。以下面的 AddCluster()分別做處理。

AddCluster(): 如果某 cluster 和現在 subrow 上的任何一個 cluster 重疊 merge 這兩個 cluster 並更新座標。若無，則將結束 PlaceRow()。

另外 paper 上決定 merge cluster 後更新的座標的方式和講義上有些許不同，兩者皆嘗試過後，發現講義上的效果較佳。

ComputeCost(i, j)：

cost 計算方式是計算 node(i)擺到 subrow(j)上後，node 的新座標和原本 global 座標的 Euclidean distance。

另外我也有做過 cost 不是只看一個 node，而是看整個 subrow 上的所有 node 的新座標和 global 座標的總和，以結果來看，效果差很多，基本上爛了一個量級 10^7 變成 10^8。

SetNodes(i, best_subrow)：

記住一個 cost 最小的 subrow 為 best_subrow，並真的把 node 放進這個 subrow，更新座標，也一併更新整個 subrow 上的 node 之座標，因為都可能相應改變。

Abacus():

基本上和 paper 的結構大致相同，先對想擺放的 subrow 去進行 PlaceRow()，並且 ComputeCost()，將 cost 最小 best_subrow 記住去 SetNodes()。較不一樣的只有上面 modify 時所說的加速方式。

此外，因為要做多次 trial & recover，為了避免麻煩，我的所有操作並沒有真正對 node 的座標去操作，而是在每個 subrow 上都記錄好一個資料結構 cluster 並對他們操作，然後每次 PlaceRow()之前，先記錄起來原始的 cluster 資訊，PlaceRow()之後再還原。

5.

Origin	Total Displacement	Max Displacement	Runtime(sec)
adaptec1	1.037E+07	2826.58	938.57
adaptec3	0.961E+07	2612.37	4494.19
ibm01-cu85	0.711E+07	5408.5	2.73
ibm07-cu90	3.826E+07	10428	22.21
ibm09-cu90	6.302E+07	13259	26.25
SpeedUp	Total Displacement	Max Displacement	Runtime(sec)
adaptec1	1.101E+07	2826.72	51.57
adaptec3	1.161E+07	2607.95	93.83
ibm01-cu85	0.719E+07	5337.06	2.18
ibm07-cu90	3.882E+07	9900	19.23
ibm09-cu90	6.389E+07	14050	23.96