

# **CS6135 VLSI Physical Design Automation**

## **Homework 2: Two-way Min-cut Partition**

Name: 賴琮翰

Student ID: 112062542

Date: 2023/11/3

## 1. How to compile and execute your program and give an execution example.

### Compile

In "HW2/src/", enter the following command:

```
$ make
```

An executable file "hw2" will be generated in "HW2/bin/".

If you want to remove it, please enter the following command:

```
$ make clean
```

### Execute

```
$ ./hw2 <txt file> <out file>
```

E.g., in "HW2/bin/", enter the following command:

```
$ ./hw2 ../testcase/public1.txt ../output/public1.out
```

### Execute example: 以 public1.txt 為例

```
[chlai22@ic21 src]$ make
g++ -o ../bin/hw2 -Ofast -std=c++17 main.cpp
[chlai22@ic21 src]$ cd ../
bin/      output/  src/      testcase/ verifier/
[chlai22@ic21 src]$ cd ../bin/
[chlai22@ic21 bin]$ ./hw2 ../testcase/public1.txt ../output/public1.out
pass init
pass
CutSize 247
[chlai22@ic21 bin]$
```

## 2. The final cut size and the runtime of each testcase.

	Cut size	Runtime
<b>Public1</b>	247	0.02
<b>Public2</b>	4670	6.19
<b>Public3</b>	29538	24.82
<b>Public4</b>	1645	0.25
<b>Public5</b>	887	0.74
<b>Public6</b>	94787	271.11

[1] Fiduccia; Mattheyses (1982). "A Linear-Time Heuristic for Improving Network Partitions". 19th Design Automation Conference.

```
[chlai22@ic21 HW2_grading]$ bash HW2_grading.sh
```

This script is used for PDA HW2 grading.

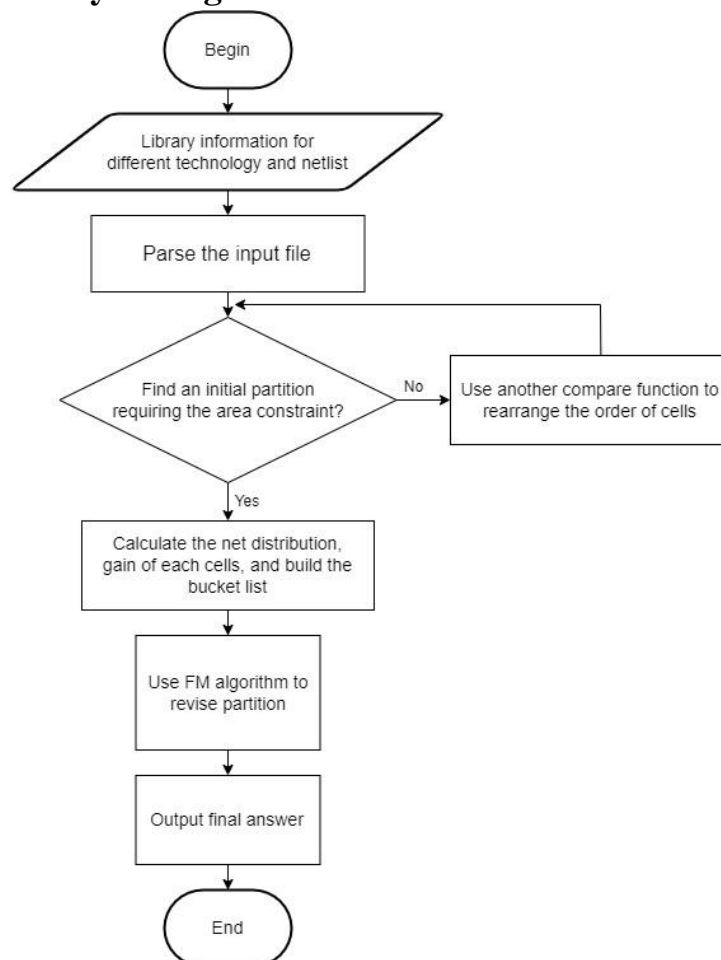
```
grading on 112062542:
rm: cannot remove '../bin/hw2': No such file or directory
make: *** [clean] Error 1
```

testcase	cutsizes	runtime	status
public1	247	0.02	success
public2	4670	6.19	success
public3	29538	24.82	success
public4	1645	0.25	success
public5	887	0.74	success
public6	94787	271.11	success

Successfully generate grades to HW2\_grade.csv

```
[chlai22@ic21 HW2_grading]$
```

### 3. The details of your algorithm.



[1] Fiduccia; Mattheyses (1982). "A Linear-Time Heuristic for Improving Network Partitions". 19th Design Automation Conference.

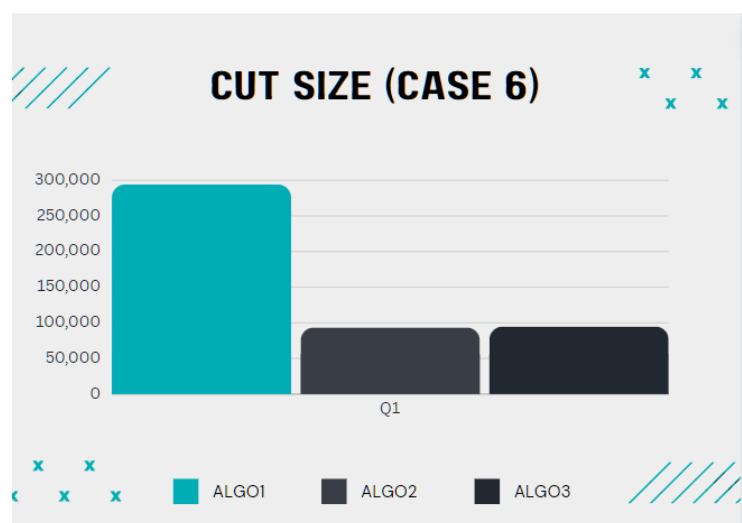
在讀取輸入檔案部分，我會先將不同technology的library cell的面積存在一個unordered map中，接著依據dieA與dieB分別使用的製程，對應到library cell的面積出netlist內的cell，代表cell連接net關係的net array存在cell中。為了操作方便，我另外替所有net都創立一個物件，裡面存net distribution和net連接的cell (cell array)。

接著，為了找到一個符合area constraint的initial partition，我所使用的策略是把一個存有每個cell的vector，cell按「特定順序」盡可能的先放入dieA，如果dieA已經放不下了，剩下的cell都嘗試放入dieB，最後再來檢查dieA和dieB是不是都符合area constraint限制。關於哪個cell要先擺放，此次作業中我寫了8種compare function，如果當前compare function沒辦法找出initial partition，則使用下一種compare function，直至找到一合法initial partition。

接著我使用課堂上教授的Fiduccia-Mattheyses (FM) Algorithm [1] 來優化initial partition。簡單來說，FM演算法可以在每個pass $O(P)$ 的時間內找出優化後的解，一次嘗試搬動一個當前具有最大gain值的cell，搬動後更新cell的gain值及net的distribution，最後真正搬動具有最大partial sum的那些cells。總體來說演算法大方向和原版FM演算法相同，其中一個不同之處在於area constraint的設立，原本的FM其area constraint是dieA面積佔dieA+dieB面積的比例，但本次作業的constraint較特別，改動後即可。另外，原本的FM bucket list是用vector + linked list去存的，我認為這樣做比較麻煩，於是我用map + unordered set實作bucket list。

#### 4. What tricks did you use to enhance your solution quality?

我認為有兩個地方對於成績提升的幫助巨大，一是找initial partition的時候，我使用多種compare function來找initial partition。二是用更動後的FM演算法來優化initial solution。以下的圖表說明FM演算法使用對於public case 6成績的影響，橫軸由左至右分別為：單純initial partition的結果、用FM演算法優化，partial sum相同時取較前者、用FM演算法優化，partial sum相同時取較後者。前者cut size = 93491，後者為94787，前者表現更為優異。不過這樣的差距可能還不能確定前者必定比後者好，有可能只是恰好在這個case表現好而已。



[1] Fiduccia; Mattheyses (1982). "A Linear-Time Heuristic for Improving Network Partitions". 19th Design Automation Conference.

## **5. What have you learned from this homework? What problem(s) have you encountered in this homework?**

本次作業中我學習到如何寫一個良好的partition演算法，用來解決2-way partition的問題，除此之外，我也學習到如何正確的寫好makeFile，這技能對於我未來的求職路非常有幫助。另外便是我也學習到C++如何把function當成參數傳進去另外一個function中，如此一來便可以達成以同一個function執行不同sorting方式input的結果，最後便是時間管理的能力，因為在寫這份作業的時候我剛好到美國領獎，行程安排加上時差讓我能夠寫程式的時間變得非常非常少，幸好我利用時間管理的技巧把工作切成多個細項逐步執行，才有辦法在deadline前完成這次作業。

問題1: 哪個cell要先放才能找出合法的initial partition? 一開始，我完全找不到一個general的compare function，無論怎麼寫compare function都會有1~2個public case無法找到initial partition，這樣一來後續的優化也完全無法進行，不過經過觀察，我發現到這些涵蓋的public case是可以互相cover的，也就是說只要用多個compare function，就能解決此問題。

問題2: 回溯不需要交換的cell時出現一堆bug。在FM演算法裡，每個pass會把全部cell都swap過，不過不是所有的cell最後都會搬動，那些沒被搬動的cell需要被還原，但可能是因為我程式沒寫好，一直出現錯誤，最後是直接把所有cell事先複製一份，有需要時直接還原即可。