CS6135 VLSI Physical Design Automation

Homework 5: Placement Legalization

Due: 23:59, June 6, 2025

1. Introduction

In this homework, you are asked to adapt and implement an existing algorithm, published in the ISPD-08 paper entitled "Abacus: fast legalization of standard cell circuits with minimal movement" by Spindler, Schlichtmann and Johannes, to legalize a given global placement result with minimal total displacement (measured by Euclidean distance).

2. Problem Description

(1) Input:

- Max displacement constraint for each cell.
- A set of standard cells (and blockages), where each standard cell (or blockage) has a rectangular shape specified by its width, height, and coordinates. The design is composed of single-row height movable cells, and multiple-row height fixed blockages.
- Chip specification, such as the coordinates of each row, the row height, the site width, and the number of sites in a row.

(2) Output:

- The coordinates of each cell after legalization and the total displacement t_d as well as the max displacement m_d. The coordinates of each cell are specified by its lower-left corner. Please take the ceiling of the total displacement after summing all the cell displacement and take the ceiling of the max displacement.

(3) Objective:

Cells are not allowed to be rotated, but they are allowed to be moved. The total displacement of the legalization result should be as small as possible subject to the following constraints.

1. Aligning constraint: Each cell is not allowed to cross multiple rows, and must align its left boundary with the edge of the site.
2. Non-overlapping constraint: No cell overlaps with other cells or blockages.
3. Max displacement constraint: The displacement of each cell should be less than or equal to the max displacement threshold.

3. Input File

(1) The *.txt* file:

The .txt file specifies the information of max displacement constraint, cells, blockages, and rows in the placement region. Here is an example:

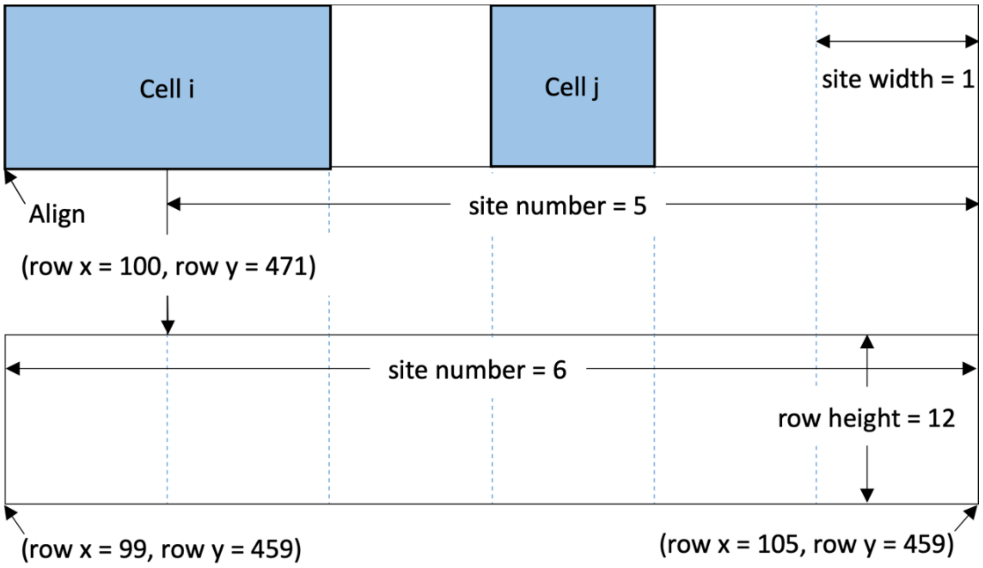
```
MaxDisplacementConstraint 12
// MaxDisplacementConstraint threshold of max displacement

NumCells 3
// NumCells the number of cells
Cell c0 1 12 10.0 10.0
// Cell cell name cell width cell height cell x cell y
        ⋮

NumBlockages 1
// NumBlockages the number of blockages
Blockage b0 4 24 11 10
// Blockage block name block width block height block x block y

NumRows 2
// NumRows the number of rows
Row r0 1 12 10 10 10
// Row row name site width row height row x row y site number
        ⋮
```

- Only the coordinate of the cell is floating type.


Cell i
Cell j
site width = 1
Align
site number = 5
(row x = 100, row y = 471)
site number = 6
row height = 12
(row x = 99, row y = 459)
(row x = 105, row y = 459)

4. Output File

(1) The *.out* file:

The .out file specifies the total displacement, the max displacement, and the legalization result containing the coordinates of each cell. Here is an example:

```
TotalDisplacement 12
// TotalDisplacement number of total displacement
MaxDisplacement 7
// MaxDisplacement maximum displacement from all the cells

c0 10 10
// cell name cell x cell y
    ⋮
```

5. Language/Platform

(1) Language: C/C++

(2) Platform: Unix/Linux

6. Report

Your report must contain the following contents, and you can add more as you wish.

(1) Your name and student ID

(2) How to compile and execute your program and give an execution example.

(3) The total displacement, the max displacement and the runtime of each testcase. Paste the screenshot of the result of running the **HW5_grading.sh**.

(4) The details of your implementation. If there is anything different between your implementation and the algorithm in the ISPD-08 paper, please reveal the difference(s) and explain the reasons.

(5) How did you handle the row if it is divided by the blockage?

(6) What methods did you use to handle the max displacement constraint?

(7) What tricks did you do to speed up your program or to enhance your solution quality?

(8) What have you learned from this homework? What problem(s) have you encountered in this homework?

7. Required Items

Please compress `HW5/` (using tar) into one with the name `CS6135_HW5_${StudentID}.tar.gz` before uploading it to eeclass.

(1) `src/` contains all your source code, `Makefile` and `README`.

- ➢ `README` must contain how to compile and execute your program. An example is like the one shown in HW2.

(2) `output/` contains all your outputs of testcases for the TA to verify.

(3) `bin/` contains your executable file.

(4) `CS6135_HW5_${StudentID}_report.pdf` contains your report.

You can use the following command to compress your directory on a workstation:

```
$ tar -zcvf CS6135_HW5_{StudentID}.tar.gz <directory>
```

For example:

```
$ tar -zcvf CS6135_HW5_113000000.tar.gz HW5/
```

8. Grading

- ✓ 80%: Total displacement and max displacement of each testcase. For public case, you need to satisfy the max displacement constraint, while you only need to generate a valid result on hidden testcases.
- ✓ 20%: The completeness of your report.

Notes:

- ● Make sure the following commands can be executed.
  - ■ Go into directory "`src/`", enter "`make`" to compile your program and generate the executable file, called "`hw5`", which will be in directory "`bin/`".
  - ■ Go into directory "`src/`", enter "`make clean`" to delete your executable file.
- ● Please use the following command format to run your program.

  ```
  $ ./hw5 *.txt *.out
  ```

  E.g.:

  ```
  $ ./hw5 ../testcase/public1.txt ../output/public1.out
  ```
- ● Use arguments to read the file path. Do not write the file path in your code.
- ● Your program must be terminated within 1 minutes for each testcase.
- ● Please use ic21, ic22 to test your program.
- ● We will test your program by a shell script with GCC 9.3.0 on the servers mentioned above. Please make sure your program can be executed by **HW5_grading.sh**. If we cannot compile or execute your program by the script, you will get 0 points on your programming score.
- ● Note that any form of plagiarism is strictly prohibited, including the code found on GitHub and the code from any student who took this course before. If you have any problem, please contact TA.