Yi-Hua Chung
Campus ID:
9086309664

DADS-PA2

2024-04-04

1 Compile and Run

1.1 How to Compile

To compile the code:

```
g++ main.cpp fixed_outline_floorplanning.cpp -Wall -O3 -std=c++17 -o fp
```

1.2 How to Execute

To execute the code:

```
# run && check
./fp [alpha] [input file_0_block path] [input file_1_nets path] [output file path]
python3 checker/checker.py $input_file output_$input_file
```

Example:

```
# alpha = 0.5
# run && check
./fp 0.5 input_pa2/1.block input_pa2/1.nets output_1
python3 checker/checker.py 1 output_1
```

2 Results

```
=================================
running test data input_pa2/1.block input_pa2/1.nets
#######################################################################
           input: 1
   num of blocks: 5
num of terminals: 3
     num of nets: 1
            area: 720000
 area difference: 0.0
            hpwl: 1000.0
      wirelength: 1000.0
 hpwl difference: 0.0
      total cost: 360500.0
            SAME
           LEGAL
        IN BOUND
#######################################################################
=================================
running test data input_pa2/2.block input_pa2/2.nets
#######################################################################
           input: 2
   num of blocks: 30
num of terminals: 3
     num of nets: 1
            area: 720000
 area difference: 0.0
            hpwl: 400.0
      wirelength: 400.0
 hpwl difference: 0.0
      total cost: 360200.0
```

```
            SAME
           LEGAL
        IN BOUND
######################################################################
=================================
running test data input_pa2/3.block input_pa2/3.nets
######################################################################
           input: 3
   num of blocks: 103
num of terminals: 3
     num of nets: 1
            area: 213869152
 area difference: 0.0
            hpwl: 700.0
      wirelength: 700.0
 hpwl difference: 0.0
      total cost: 106934926.0
            SAME
           LEGAL
        IN BOUND
######################################################################
=================================
running test data input_pa2/ami33.block input_pa2/ami33.nets
######################################################################
           input: ami33
   num of blocks: 33
num of terminals: 40
     num of nets: 121
            area: 1290366
 area difference: 0.0
            hpwl: 84052.5
      wirelength: 84052.0
 hpwl difference: 1.4871657594955534e-06
      total cost: 687209.25
            SAME
           LEGAL
        IN BOUND
######################################################################
=================================
running test data input_pa2/ami49.block input_pa2/ami49.nets
######################################################################
           input: ami49
   num of blocks: 49
num of terminals: 22
     num of nets: 396
            area: 40922448
 area difference: 0.0
            hpwl: 1554147.0
      wirelength: 1554147.0
 hpwl difference: 0.0
      total cost: 21238297.5
            SAME
           LEGAL
        IN BOUND
######################################################################
=================================
running test data input_pa2/apte.block input_pa2/apte.nets
######################################################################
           input: apte
   num of blocks: 9
num of terminals: 73
```

```
     num of nets: 96
            area: 48736688
 area difference: 0.0
            hpwl: 748012.0
      wirelength: 748012.0
 hpwl difference: 0.0
      total cost: 24742350.0
            SAME
           LEGAL
        IN BOUND
######################################################################
=================================
running test data input_pa2/hp.block input_pa2/hp.nets
######################################################################
           input: hp
   num of blocks: 11
num of terminals: 45
     num of nets: 70
            area: 9490320
 area difference: 0.0
            hpwl: 212695.0
      wirelength: 212695.0
 hpwl difference: 0.0
      total cost: 4851507.5
            SAME
           LEGAL
        IN BOUND
######################################################################
=================================
running test data input_pa2/xerox.block input_pa2/xerox.nets
######################################################################
           input: xerox
   num of blocks: 10
num of terminals: 2
     num of nets: 182
            area: 20779920
 area difference: 0.0
            hpwl: 532203.0
      wirelength: 532203.0
 hpwl difference: 0.0
      total cost: 10656061.5
            SAME
           LEGAL
        IN BOUND
######################################################################
```

Listing 1: Experiment Results

3 Encountered Challenges

1. I encountered challenges when trying to fit all the blocks within the specified constraints. Initially, I attempted to utilize the same setup for the SA algorithm as provided by TA. However, despite this, some of the test cases still exceeded the boundaries.

   Ultimately, I revised the PE initialization part by incorporating three heuristic strategies. Two strategies are employed for PE initialization. Initially, a randomized initialization is utilized. However, if we fail to find a suitable packing strategy that keeps all the blocks within bounds, we move on to the second initialization strategy. For the second strategy, we first sort all blocks based on width and height. Then, we select a block that minimizes the remaining space (after inserting this block) while considering whether to rotate the block. To illustrate, let's consider three blocks with the following settings:

   - Bound constraints: (width, height) = (200, 300)

- Block_0: 100 x 300
- Block_1: 100 x 100
- Block_2: 50 x 50

We start by packing Block_0. Without rotation, it leaves us with a remaining width of 100 (200 - 100) and a height of 0 (300 - 300). Since rotating Block_0 would exceed the width constraint, we insert it without rotation. This results in two remaining regions (one divided vertically, the other horizontally).

Next, we choose to insert Block_1. We insert it with and without rotation, selecting the option that leaves the remaining regions as small as possible. (...)

In the end, if all of the blocks can be inserted into the bound, we head on to the SA algorithm by using the packing results of strategy 2, or we use the initialization results from the random strategy as the input of the SA algorithm.

2. Additionally, I set an early break threshold to reduce the runtime.