EDA HW3 Report

許育棠 113062556

How to compile and execute

How to Compile

In "HW3/src/", enter the following command:

$ make

How to Run

Usage:

$ ./hw3 <txt file> <out file><dead space ratio>

An executable file "hw3" will be generated in "HW3/bin/".

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

$ make clean

E.g., in "HW3/bin/", enter the following command:

$ ./hw3 ../testcase/public1.txt ../output/public1.out 0.1

Result of HW3_grading.sh

testcase	ratio	wirelength	runtime	status
public1	0.15	197140	59.51	success
public2	0.15	435338	311.34	success
public3	0.15	541904	533.52	success
public1	0.1	198320	72.21	success
public2	0.1	441758	354.56	success
public3	0.1	571577	532.50	success

Smallest dead space ratio that your program can handle while still finding valid solutions within a 10-minute time limit

Testcases	Public1	Public2	Public3
# of blocks	100	200	300

Smallest ratio	0.075	0.9	0.078

Algorithm Details (difference between paper)

1. Initial floorplan

Different from the initial floorplan taught in class (12V3V....NV), my approach will ensure that the initial floorplan's width will not exceed the maximum width. Details are explained in the next section. In this case, the initial floorplan will be closer to valid solutions, reducing the time needed to find a valid solution using simulated annealing.

2. Simulated annealing

Instead of only using one kind of simulated annealing to reduce both wirelength and area, my implementation contains two stages of simulated annealing which have different objectives. In the first stage, simulated annealing will be used to minimize floorplan area until a valid solution is found. In the second stage, the objective of simulated annealing is to minimize both wirelength and area.

3. Cost function

As mentioned, two stages of simulated annealing are used, each with a different cost function. In the first stage, the goal is to minimize the area until a valid solution is found. The cost is calculated by the amount by which the area exceeds the maximum allowable area. As a result, the cost of a valid solution will be 0 in the first stage. Once a valid solution is found, the second stage of simulated annealing incorporates wirelength into the cost function. The total cost is then calculated by adding the wirelength to the product of the first stage's cost and a multiplier of 20. This multiplication ensures that any solution exceeding the outline area contributes significantly to the overall cost, thus encouraging valid solutions while also considering the wirelength.

4. Perturb

I also implemented the three perturbation methods taught in class to modify the floorplan. However, instead of exchanging adjacent blocks, my approach randomly selects two blocks to exchange.

Additionally, the probability distribution for each perturbation method is not uniform. The probability of exchanging two blocks is set to 50%, while the probability of inverting an operator chain is 15%, and exchanging an operator and operand is set to 35%. These settings are based on heuristic.

5. **Parameter settings**

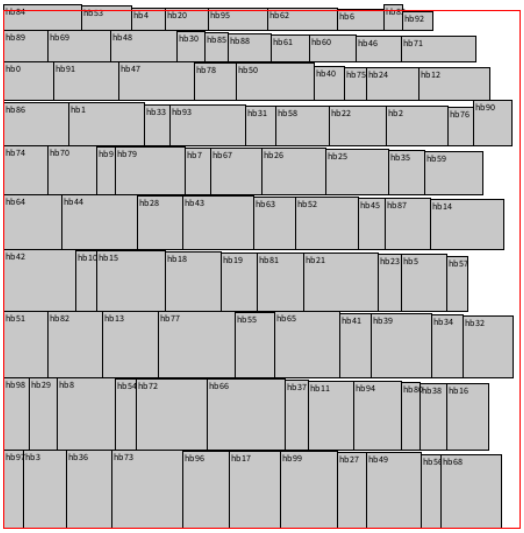
I initially attempted to set the temperature using the method taught in class. However, this approach did not outperform simply setting the temperature to 1000. Given that the method requires additional computations and produces worse results, I decided not to use it in my final implementation. Additionally, since simulated annealing is a stochastic algorithm, different random seeds can lead to different outcomes. To address this, I experimented with random seeds ranging from 1 to 10 and selected the seed that yielded the best result for each test case.

Initial Floorplan Method

My approach for generating the initial floorplan begins by sorting the blocks in descending order of height. If two blocks have the same height, the block with the smaller width is given priority. Once sorted, I place the blocks one by one, positioning each block to the right of the previous one within the same row. If the total width of the blocks exceeds the outline width, the next block is placed in a new row, and this process continues until all blocks are placed.

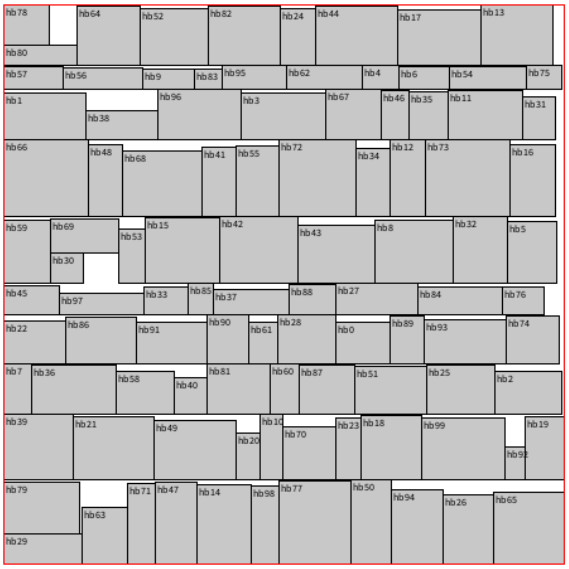
Tricks Used (use public1, 0.1 as example)

1. **Initial floorplan**



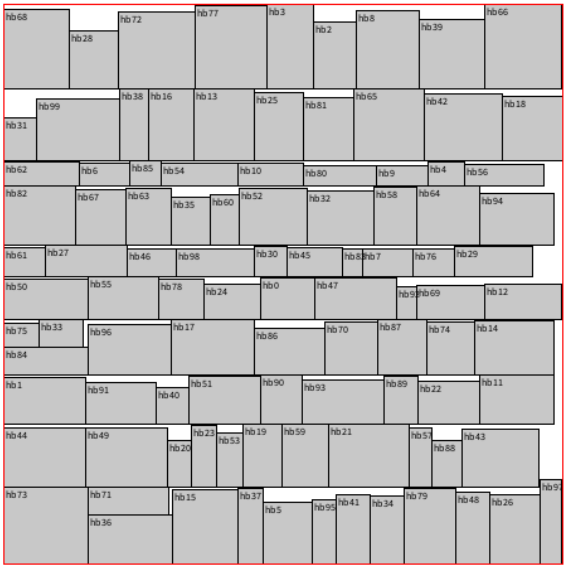
After initial floorplan, as you can see, it might not be a valid answer.

2. Simulated annealing for area



The first simulated annealing will continue to search for solution that is inside the outline.

3. Simulated annealing for wirelength+area



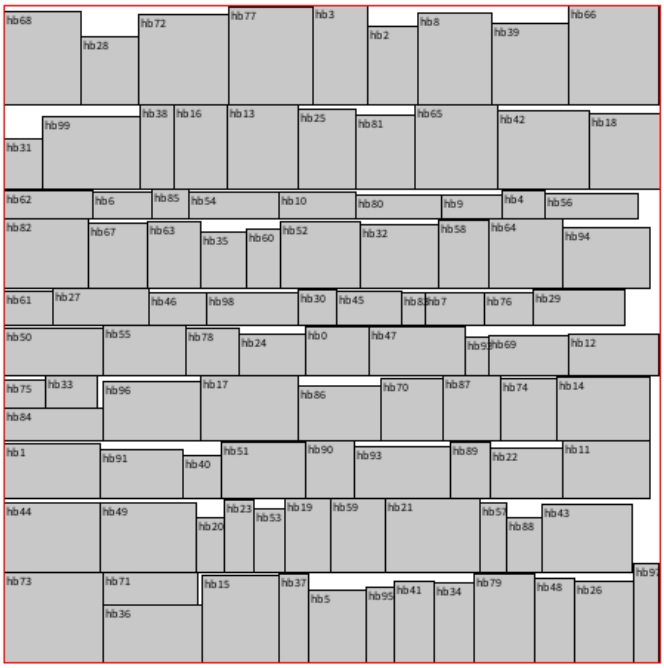
The second simulated annealing will search for solutions with lower wirelength but still be inside the outline.

What Have I Learned

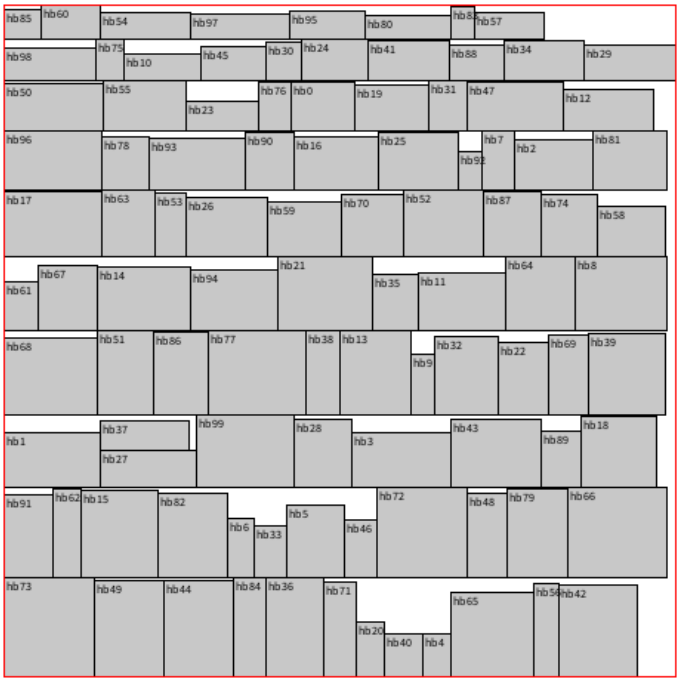
In this homework, I learned how to represent a slicing floorplan using a Normalized Polish Expression (NPE) and apply Stockmeyer's algorithm to compute the minimum area corresponding to a given NPE. Additionally, I gained hands-on experience with simulated annealing for optimization and realized that the choice of parameters and the design of the cost function significantly affect the final results. Overall, this assignment deepened my understanding of floorplan optimization and the impact of heuristic techniques in achieving efficient layouts.

Final Results for Each Testcase

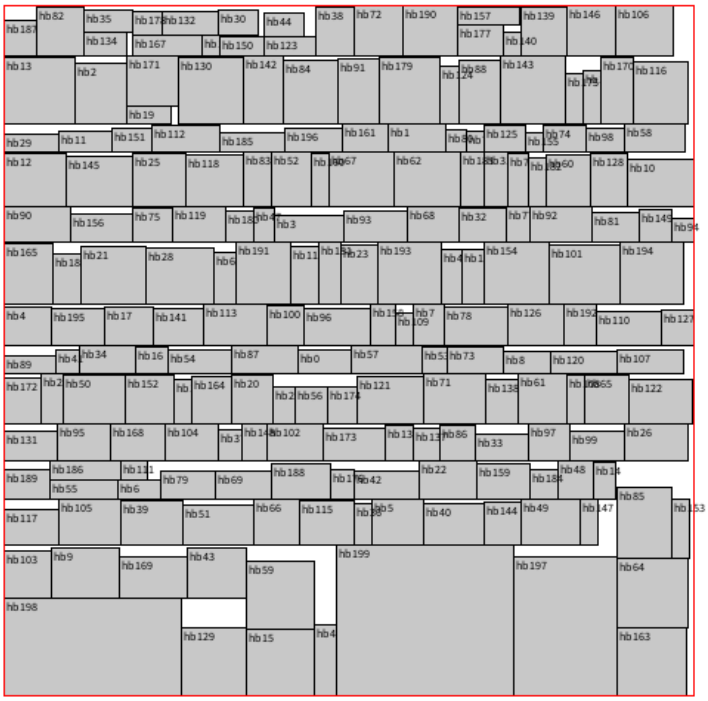
Public1 0.1



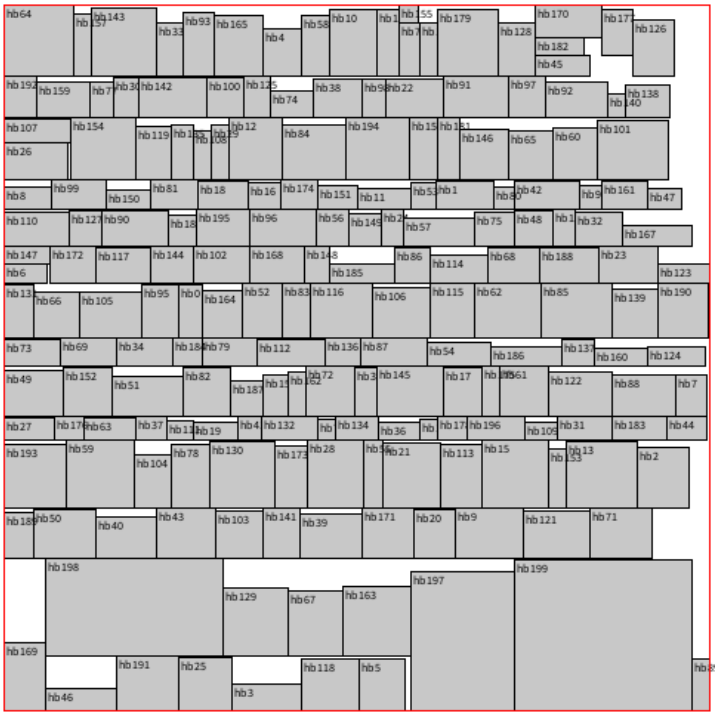
Public1 0.15



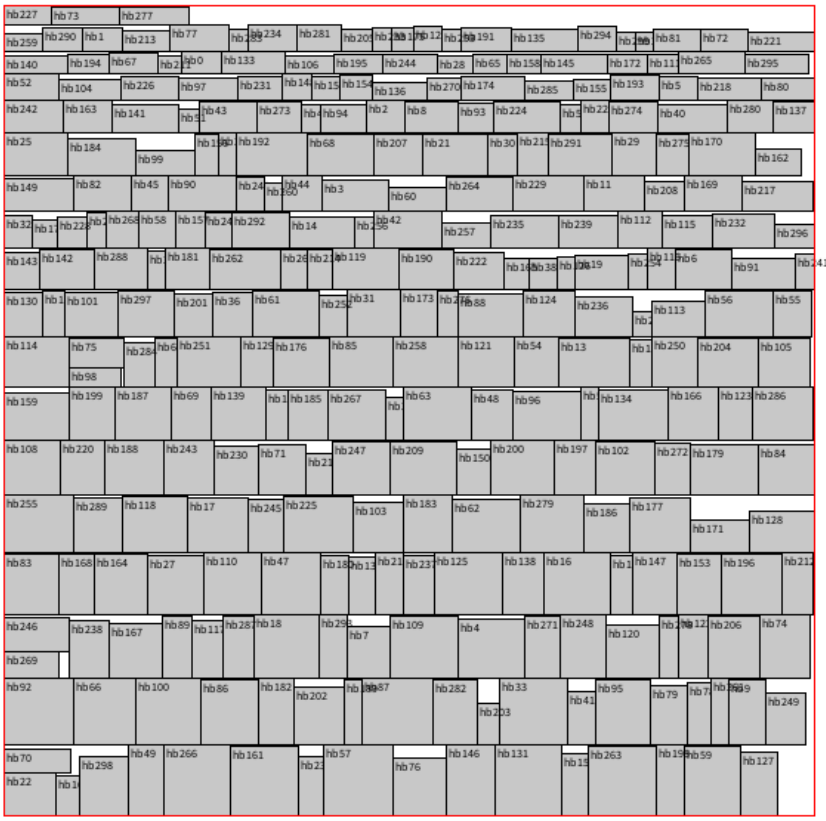
Public2 0.1



Public2 0.15



Public3 0.1



Public3 0.15