Independent Elevator Simulation Project

Wanxiang Xie

April 2020

Contents

1 Problem description

1.1 Research about feasible parameters

The elevator system uses a medium-sized commercial office building as a usage scenario to simulate running on a floor of 30F, each floor is 4 meters high, so the total height is 120 meters. This building is equipped with 3 elevators for the office building, each with a maximum capacity of 10 people According to the standard, the maximum speed of the elevator is no less than the total height of the building / 60 = 120/60 = 2m/s So, we use an elevator with V_max = 2m/s to simulate.

Assuming that when the elevator is within the weight limit, as shown in the following figure, it starts and stops at a constant acceleration.

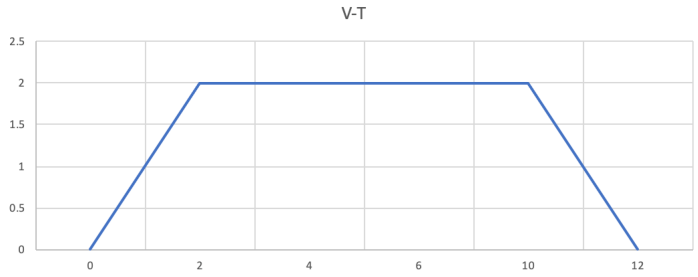


Figure 1: v - t graph

According to the GB/T10058-2009 3.3.3 standard, when the rated speed of the elevator is within 1.0m / s¡V ¡=2.0m / s, the acceleration and deceleration speed is not less than 0.5m / s, therefore we use the acceleration a = 1 m/s

Refer to the British "CIBSE GUIDE" standard, here are several important criteria for measuring users experience:

AI (INT): average running interval time (the same group of elevators, how long will an elevator arrive on average)

AWT: average waiting time (the time from when the user enters the elevator hall to when the elevator arrives). According to CIBSE Standard, AWT is generally in the range of 30-50s.

HC5: Five Minute Handling Capacity (the ratio of the number of people that the elevator can carry to the total service number in 5 minutes during the peak period) is acceptable within 10

AJT: average travel time (the time after the user enters the elevator until the elevator reaches the destination and opens the door)

ATTD: the time from when the user enters the elevator hall until it reaches the destination floor completely

Note: All the above concepts refer to the peak period (morning or evening)

Usually the most important indicators to measure a building are the reference HC5 and INT, sometime need to measure AWT at the same time. So My programming are mianly measure these three parameters.

1.2 Parameter

building's total height:120m
numbers of floor: 30F
each floor height: 4m
numbers of elevator: 3
elevator max speech: 2m/s
elevator launch and stop acceleration: 1m/s
each elevator capacity: 10 people

1.3 Result assumption

First of all, let us make a rough calculation of the above parameters according to my design:

AWT: Assuming that passengers are evenly distributed on each floor, with the middle floor 15F as the standard, the AWT of a single elevator = (4 + 116/2 +10) / 2 = 36s

AI(INT): It's hard to assume it, because I can't predict passengers choice before I implement.

HC5: According to the total number of 500 people in the office building, during the early peak period, it is assumed that the opening and closing time of the elevator is 10s each time, and the round-trip time of an elevator is (4 +116/2 + 10) * 2 = (4s + 58s + 10s) * 2 = 72s * 2 = 144s, then the three elevators run at full load for 5 minutes at the same time, and the number of people that can be loaded is 60s * 5 / 144s * 30 = 62.5 people Thus, the lift capacity of this elevator system in 5 minutes HC5 = 67.1 / 500 * 100

Note: More accurate and detailed indicator calculations will be given based on the data later and I will compare my assumption and result in the

end.

2 Design

There are 3 questions:

1.How many threads will be used?

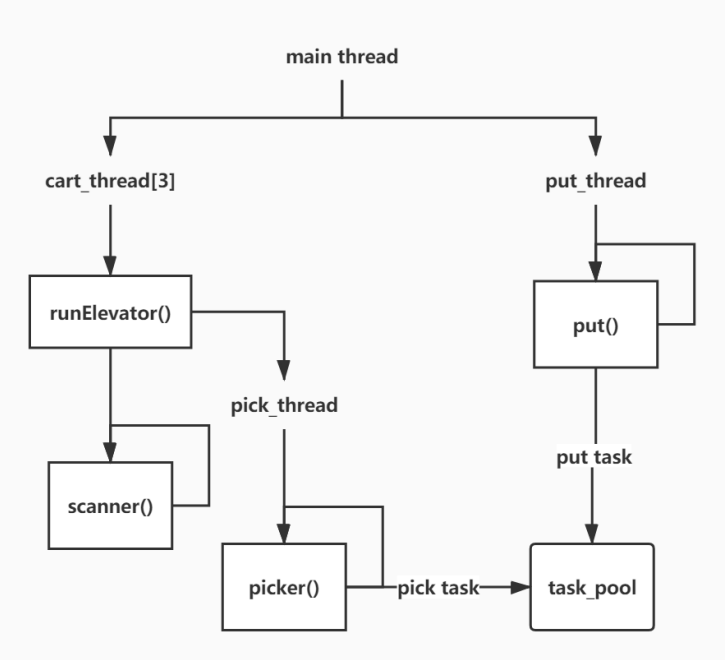
2.How to put task and pick task(the task just like people press button outside the elevator)?

3.Which algorithm to use to do task?

4.How to set target floor(target floor just like people enter in cart and press button)?

In the main thread, I create 3 threads cart_thread[3], each thread control a single cart to run and pick tasks.And I also create a put_thread to put task into task_pool.

In each cart_thread, beside running to move up and down(using scanner()), I create a pick_thread to pick task from task_pool.



2.1 Function

runElevator():two main functions, one is to create pick_thread; Other is let cart do scanner() in while loop.

scanner():the cart will scan from current floor up or down, the first find clicked button will set as move direction and do goUp_wayDown() then do goDown_wayDown() or vice versa.

put(): this function will create random tasks and put to task_pool.(use hash_set)

picker(): this function has two pick ways.

1.when cart isn't running, cart will directly use Proximity algorithm to do task.

2.when cart is running, cart will try to find tasks in the same direction and use Waydown algorithm to do task.

Obviously, We can find answers for Q1,Q2,Q3 above. But how about Q4?

In my design, I have a setFloor() function to set Current Floor whenever cart arrive a task floor(Like passenger has been taken). And then this function will immediately set new target.

You may ask, how to set new target?

Well, I have two modes in my program. If you choose Normal mode, only the same direction's button will be set. If you choose HC5 Test mode, the cart will always set up direction because in this mode, we need to simulate Early peak time, people just press button at basement(0F) or ground(1F) to go up, and then press higher target floor in cart, right?

3 Visualization

3.1 Choose Mode

When program is running, you should choose one mode to run.

1.Nomral: program will never end, it will show you elevators running processing and buttons panel status.

2.HC5 Test: program will stop at 500 seconds, the timing is started when the scheduler assigns tasks(not running time).

Figure 2: Choose Mode View

3.2 Normal Mode

There are three section:building simulation area, cart button panel simulation area and parameter area.

1.Building simulation: to simulate 3 carts go up and down, when it arrived clicked floor, it will open and close door(wait 10s).

2.Cart button panel simulation: to simulate each cart inner button panel, when button select, "A" will be shown.

3.Parameter area: to show various parameter, like Run time, Responsed Task Count,Arrived Target Count,AWT(average waiting time),AI_INT(average running interval time).

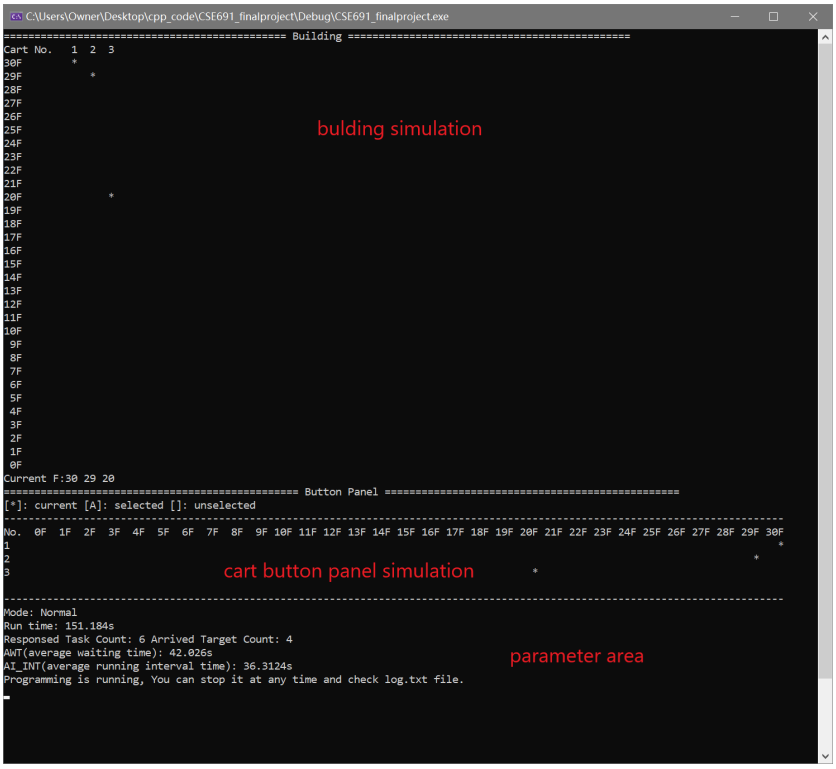


Figure 3: Normal mode

3.3 HC5 Mode

There are three section:building simulation area, cart button panel simulation area and parameter area. What's different with Normal mode is Parameter area.

3.Parameter area: Only HC5 Test mode will show HC5(five minute handling capacity) parameter.

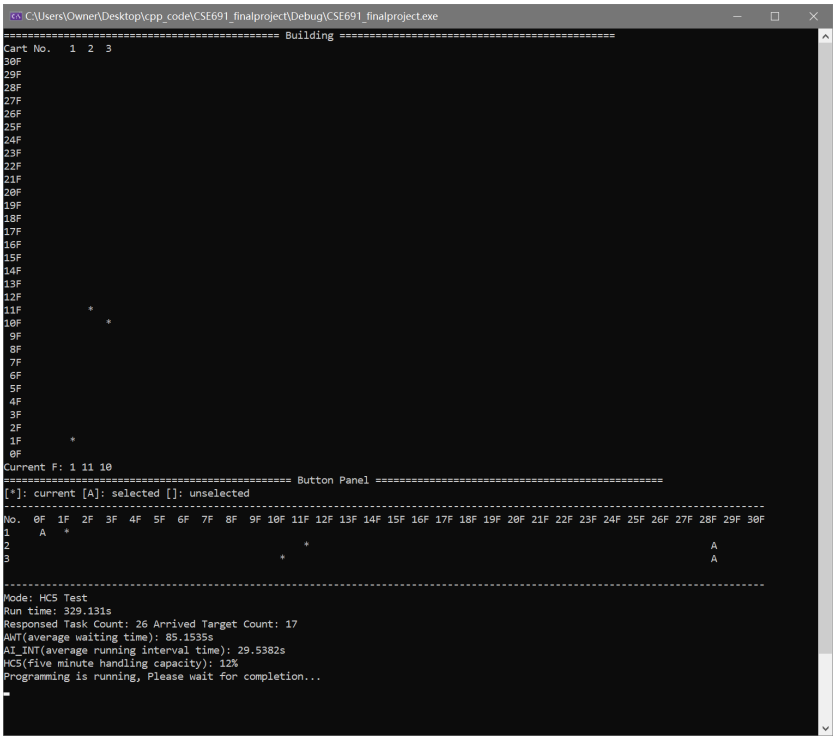


Figure 4: HC5 Test mode - running

When HC5 is calculated finished, Console will show prompt below. You can see final HC5 value.

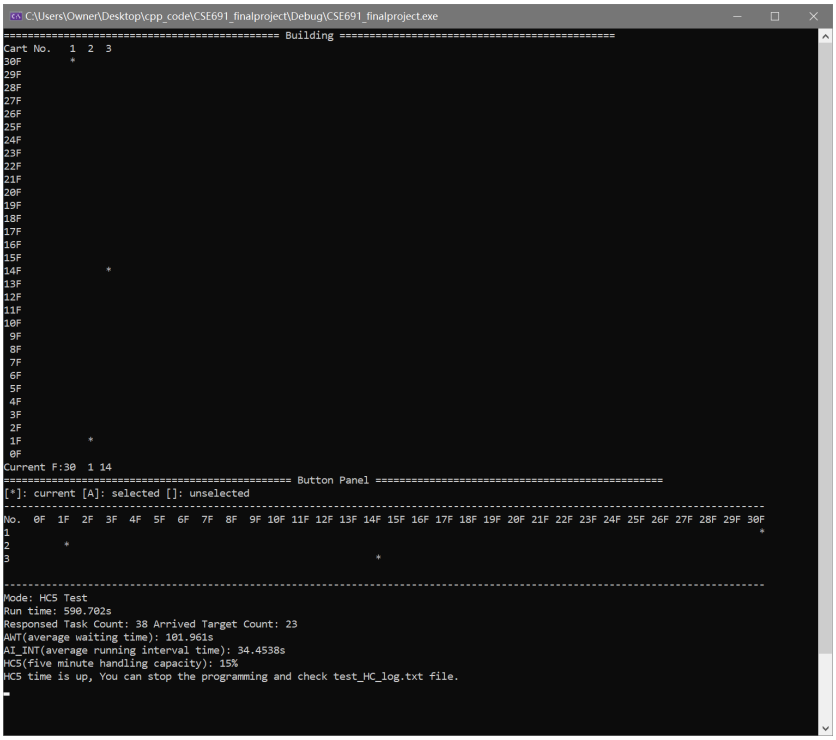


Figure 5: HC5 Test mode - finished

4 Output File

8.There are three output file.

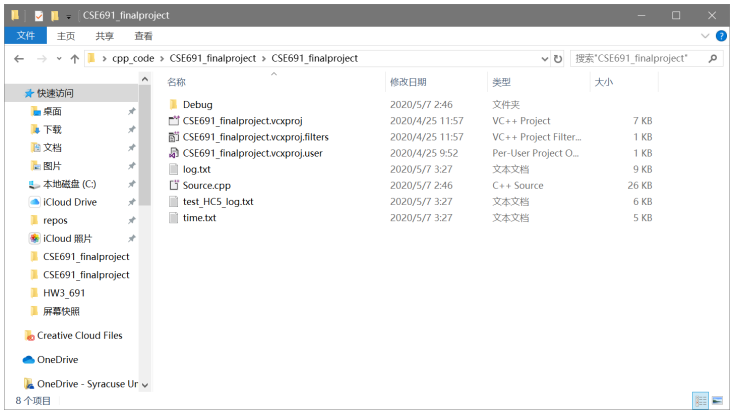


Figure 6: HC5 Test mode - finished

log.txt is mainly to record every important operation.(most important)

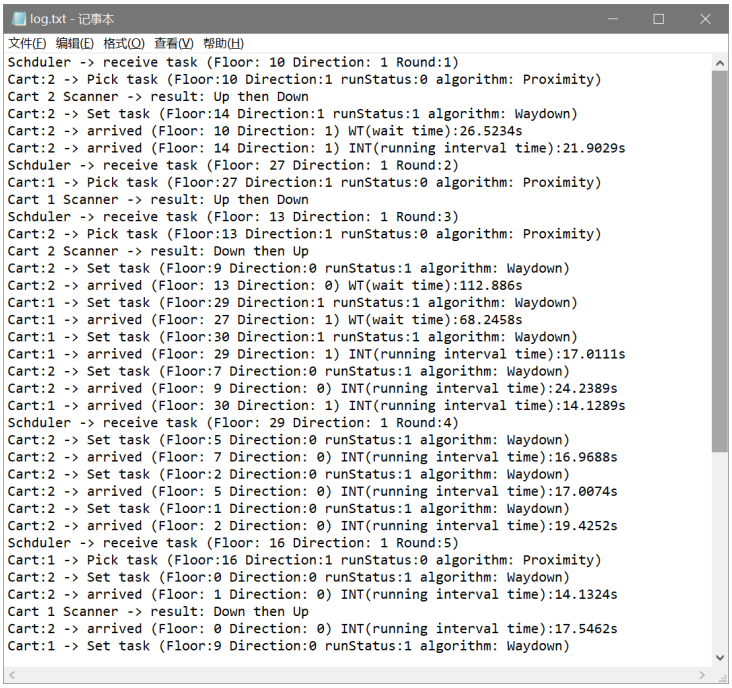


Figure 7: HC5 Test mode - finished

test_HC5_log.txt only record opreatorion about HC5.

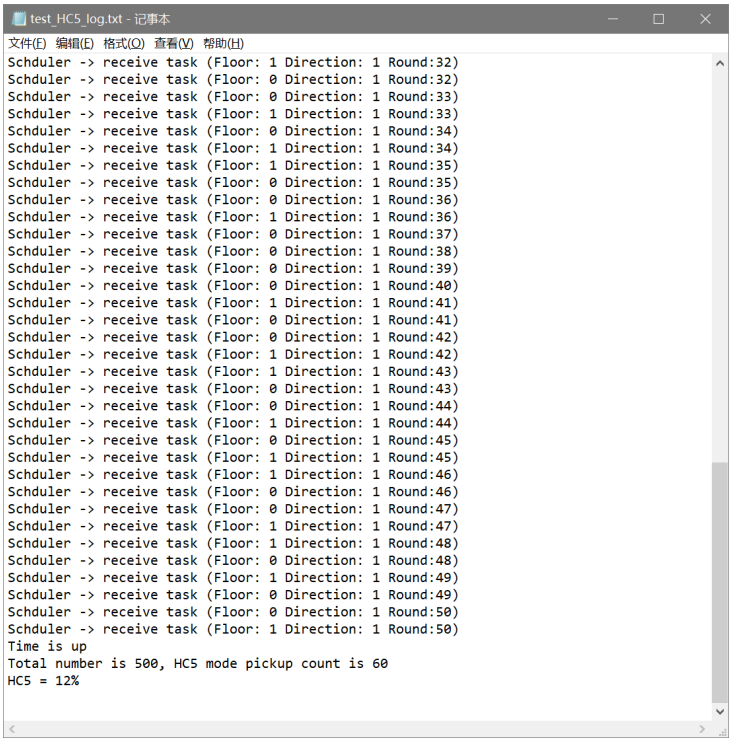


Figure 8: HC5 Test mode - finished

time.txt is to record elevator operator on each timepoint.

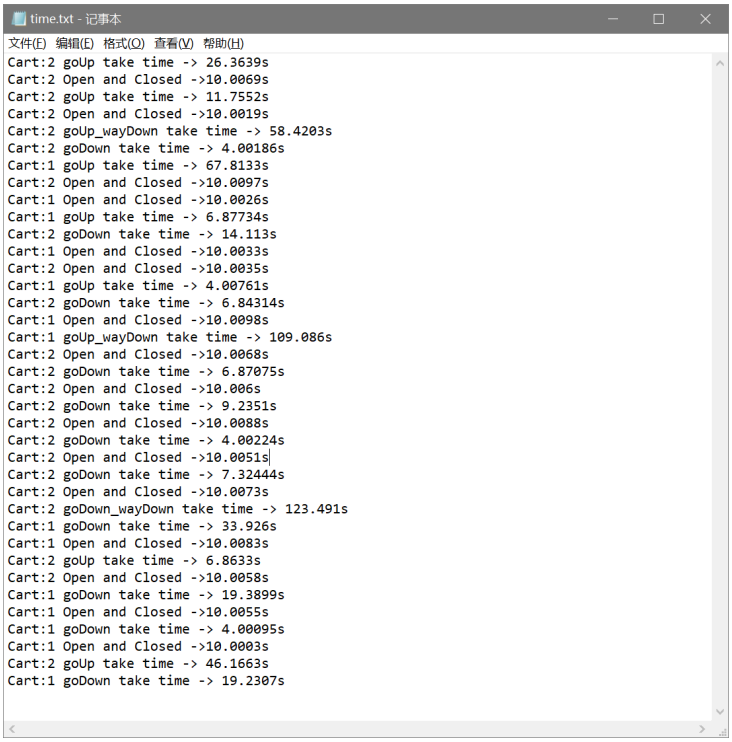


Figure 9: HC5 Test mode - finished

You should delete all of them befor you restart the program.

5 Conclusion

5.1 Result

Remember my previous assumption?
AWT:
Assuming: 36s
Result: 28-50s in first 20 tasks,50-86 in further tasks in HC5 mode. 34-130S in Normal mode
AI(INT):
Assuming: nil
Result: 27-31s in Normal mode; 35-48s in HC5 mode
HC5:

Assuming: 12.5%
Result: 12-18%

5.2 Analysis

As we can see in result, the AI(INT) and HC5 is acceptable, But AWT is pretty much higher than we assume.
Why AWT is so higher,especially as time progresses?
I think it is because when the task is continuously added, the next task needs to wait for the previous task to be completed before it gets the desired and there is a 10 second opening and closing overhead each time.