Project 2 report

**Using FIFO to transmit speed and path:**

The FIFO transmit the path in this way:

The server gives the path of grid sequence to the sc\_fifo and it give these grid sequence of four robots to four robots module. Then the four “robot module” use sc\_fifo to send it to the processing module.

The processing module controls the **robot location update** based on the path information. I created an individual simulation of the whole process. The code is just like below (Figure 1/2/3):

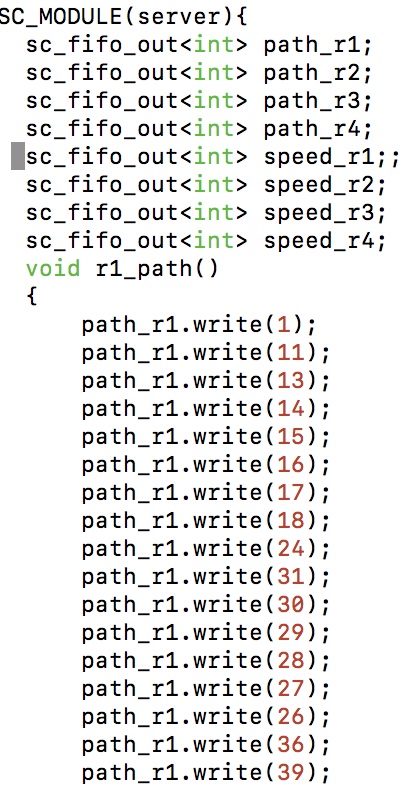


Figure 1: FIFO out of the path from sever

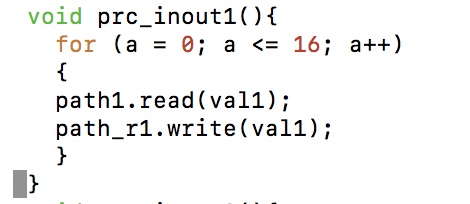


Figure 2: The path information goes through the robot module

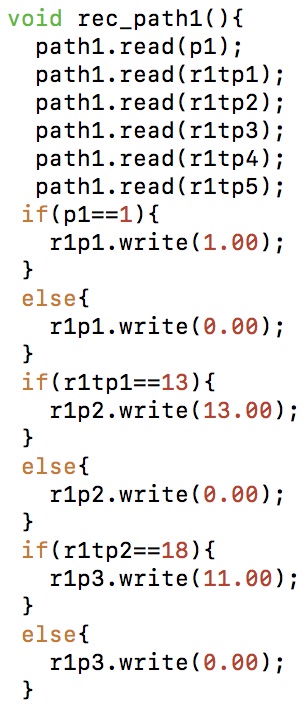


Figure 3: The processing module gets the path information and change it to location information

And this is just the test of the path information transmission, in the real simulation, I just transmit the turning point of the path to the processing module to simplify the simulation.

Also, the speed transmits using FIFO in the same way.

**Timer:**

I generate a timer to let the four robots start at different time. The code of timer is like this (Figure4):

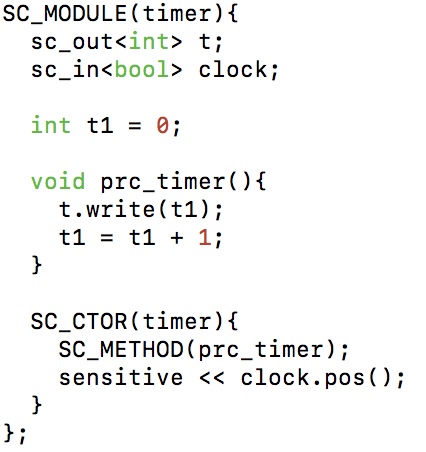
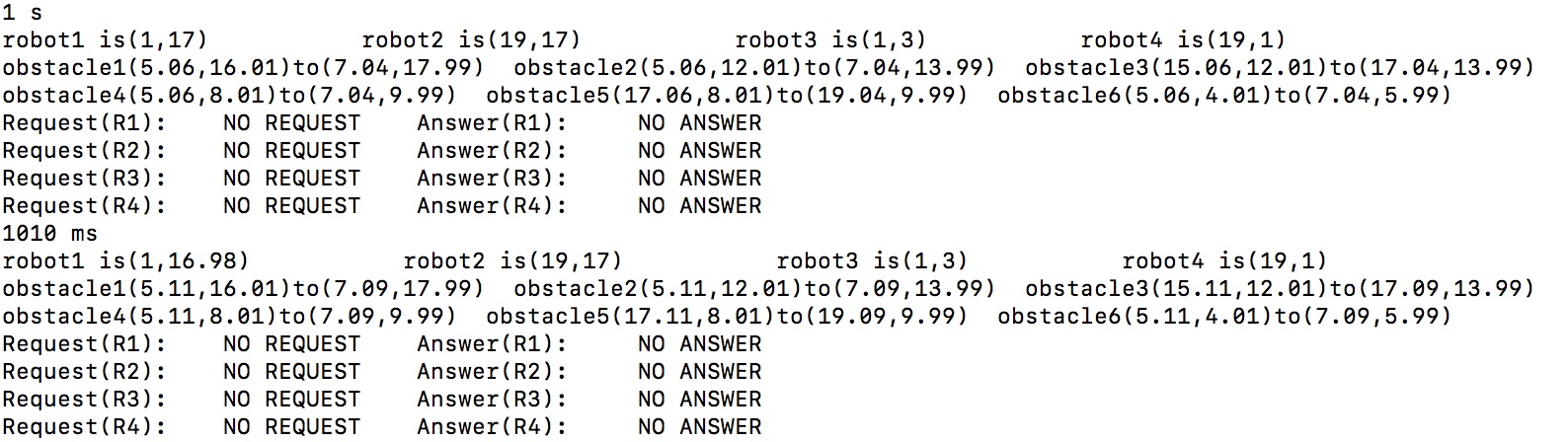


Figure 4: Timer module

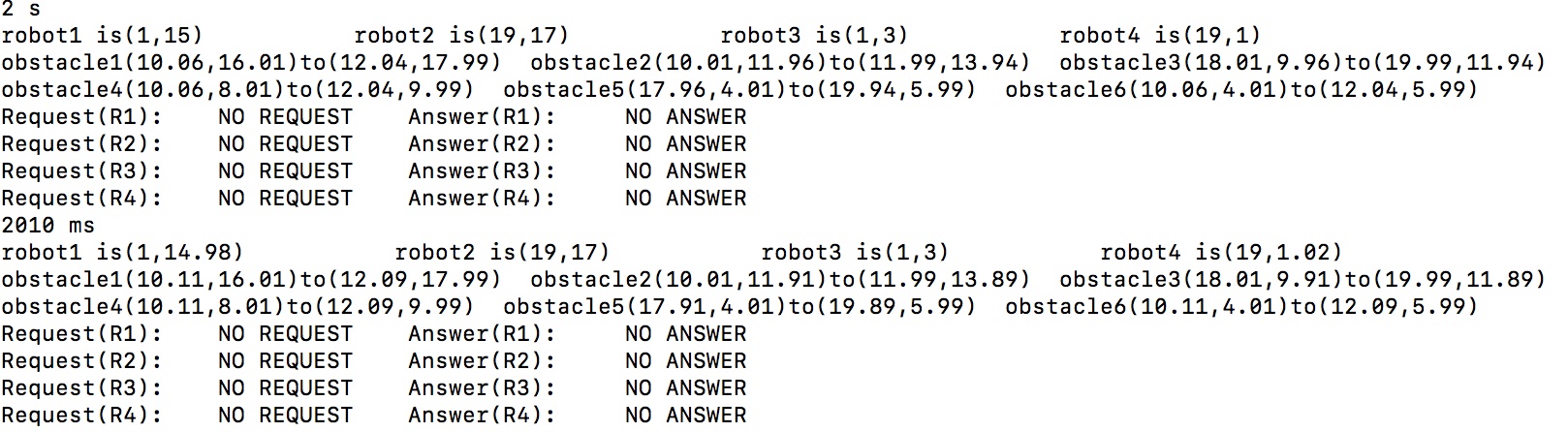
Every clock cycles the t will increase one, and the clock cycle is 10ms, so starts at 1s, 2s, 5s and 7s is just set t larger than 999,1999,4999 and 6999 in update module of processing module.

And we can see them start at their own time slot in the simulation.

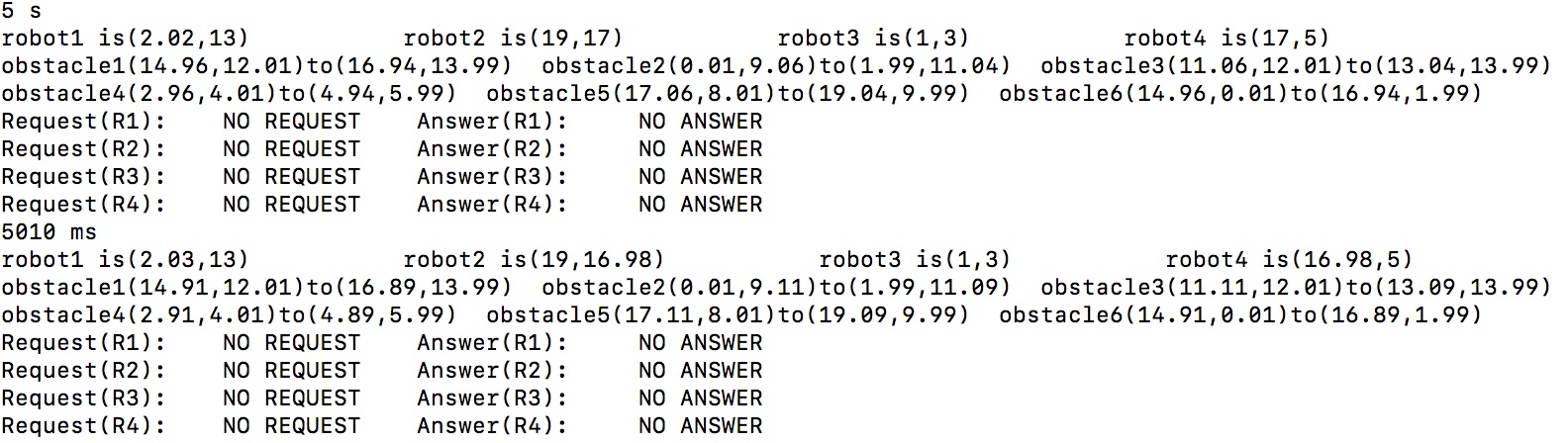
R1 starts at 1s:



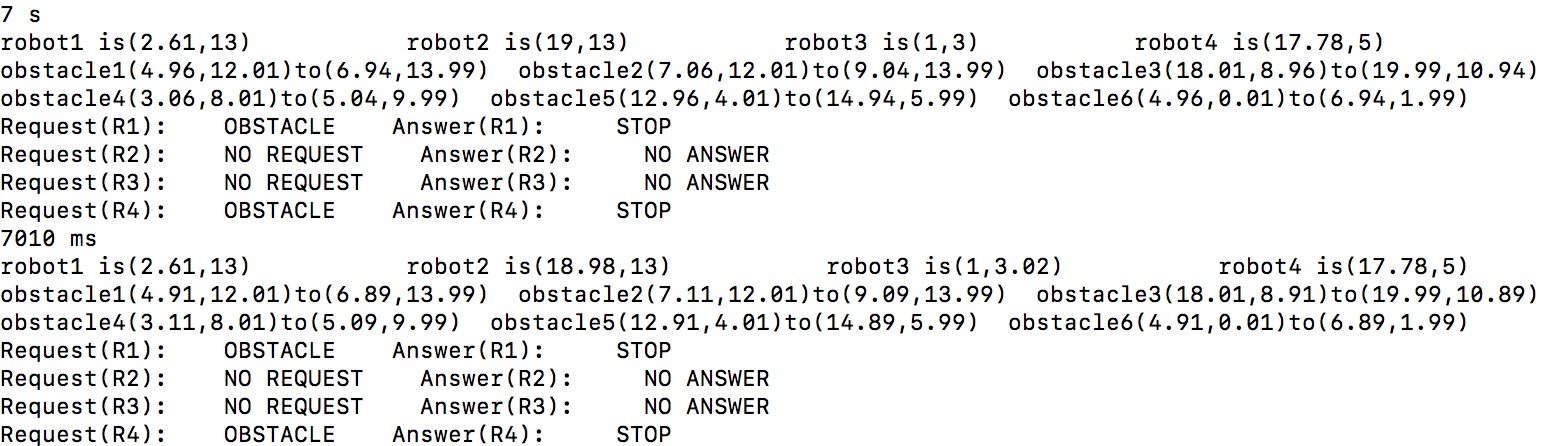
R4 starts at 2s:



R2 starts at 5s:



R3 starts at 7s:



**Speed Control:**

Analysis before simulation:

The node map I generated is listing below: (Figure5):

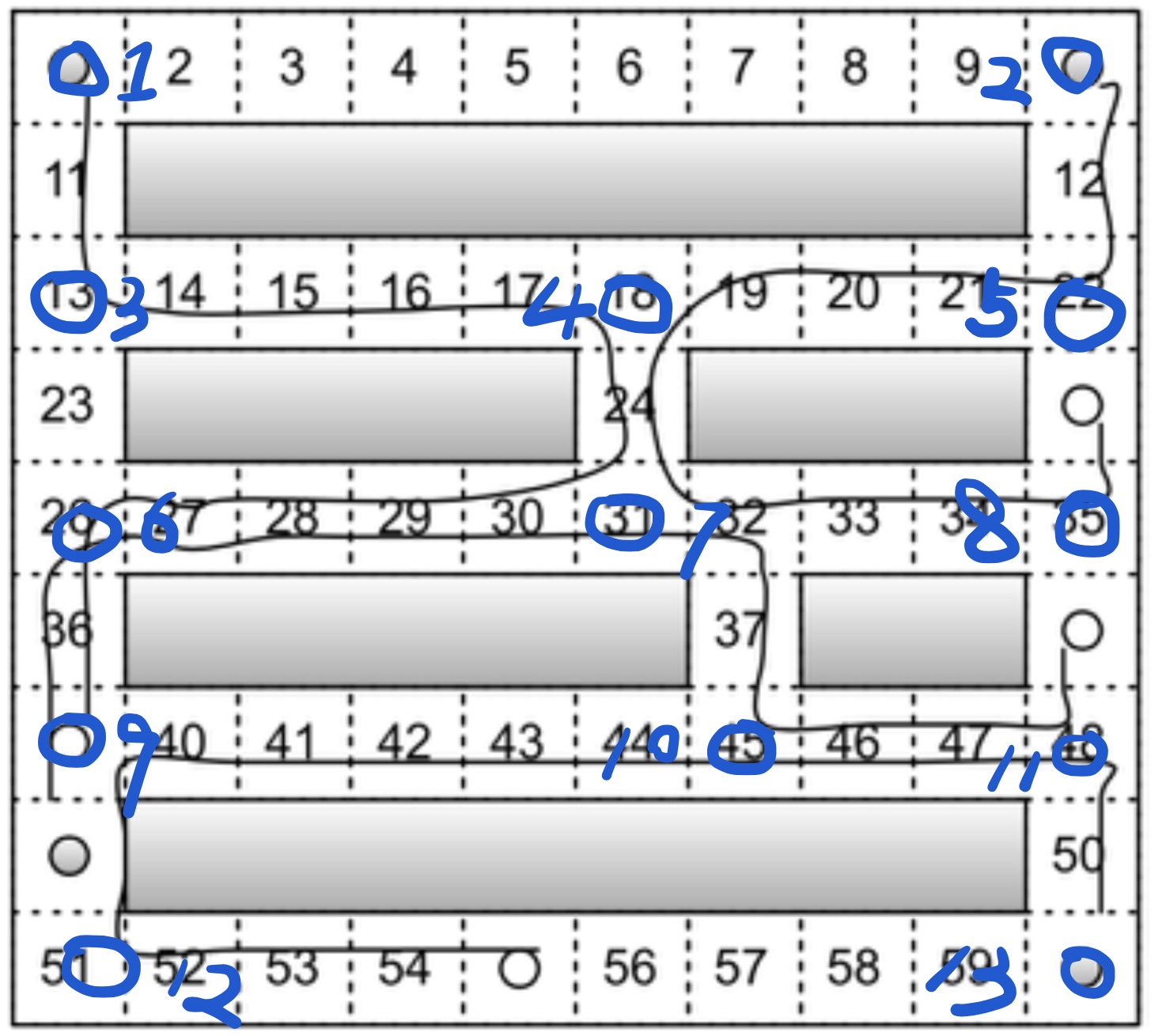


Figure 5: Node map

Grid 31 is node7, if Robot1 arrive at node 7 first, the R3 will always have a deadlock. So, we need R3 to arrive at node7 first. R3 need always keep the highest speed from node 9 to node 6 to node 7, which is 2m/s. The time it arrives at node 7 is 8sec and it starts at 7sec, the whole time is 15sec.

Grid 18 is node 4, R1 and R2 will arrive at node 4, because the R2 will not have deadlock with R3 at node 7, so I would like R2 to arrive at node4 first. So, time form grid 22(node5) to node 4 is 4 sec. R2 starts at 5sec, and from node2 to node 5 it spends 2sec to navigate, so the estimated time form R2 to node 4 is 11 sec. R1 spends 2sec from grid 1(node1) to (grid 13) node3, and it starts at 1sec, so the speed of it must less than

.

As a result, we set the speed of R1 to be 1m/s from node 3 to node 4.

Grid 31(node 7) is a special node, there are three robots will arrive it. We know that R3 need arrive it earlier than R1, now I would like to judge the arriving sequence of R2 and R3. R2 arrives at node 4 at 11sec if it always keeps 2m/s speed, R3 arrives at node 7 at 15sec if it always keeps 2m/s. From node 4 to node7, the distance is 4m. R2 can go through them using 2sec if keep the highest speed. So, the whole time it can arrive at node 7 is 13 sec. So R2 will arrive at node 7 first and then R3 and then R1.

Now, I would like to discuss the speed of R1 from node 4 to node 7. The time it spends from node 1 to node 4 is

It needs to arrive at node 7 later than 15s, so the speed of it from node 4 to node 7 must less than

.

As a result, we set the speed of R1 to be 1m/s from node 4 to node 7.

Another 2 nodes it may have deadlocks are Grid 45(node 9) and Grid 39(node 10). But actually, according to the. Computation, they won’t have deadlock. If R4 using the highest speed 2m/s as its speed, it will arrive at node 10 speeding 5sec, it starts at 2sec, so the whole time is 7sec. However, at that time R3 just start moving, so they won’t meet each other. Also, if R1 first arrives at node 10, there will be a deadlock appear. But, R4 arrives at node 10 using 13sec with highest speed, and at that time R1 still don’t arrive at node 7. So, there won’t be a deadlock.

As a result, the order robots approach the nodes and the speed of them are listing below (Figure 6):

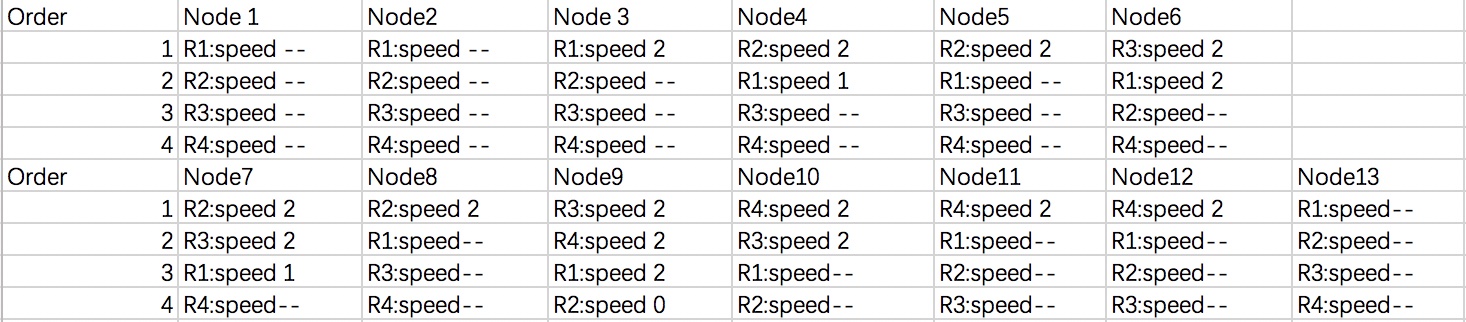


Figure 6: The order of robots approaching

And, the speed variation plots are below (Figure 7):

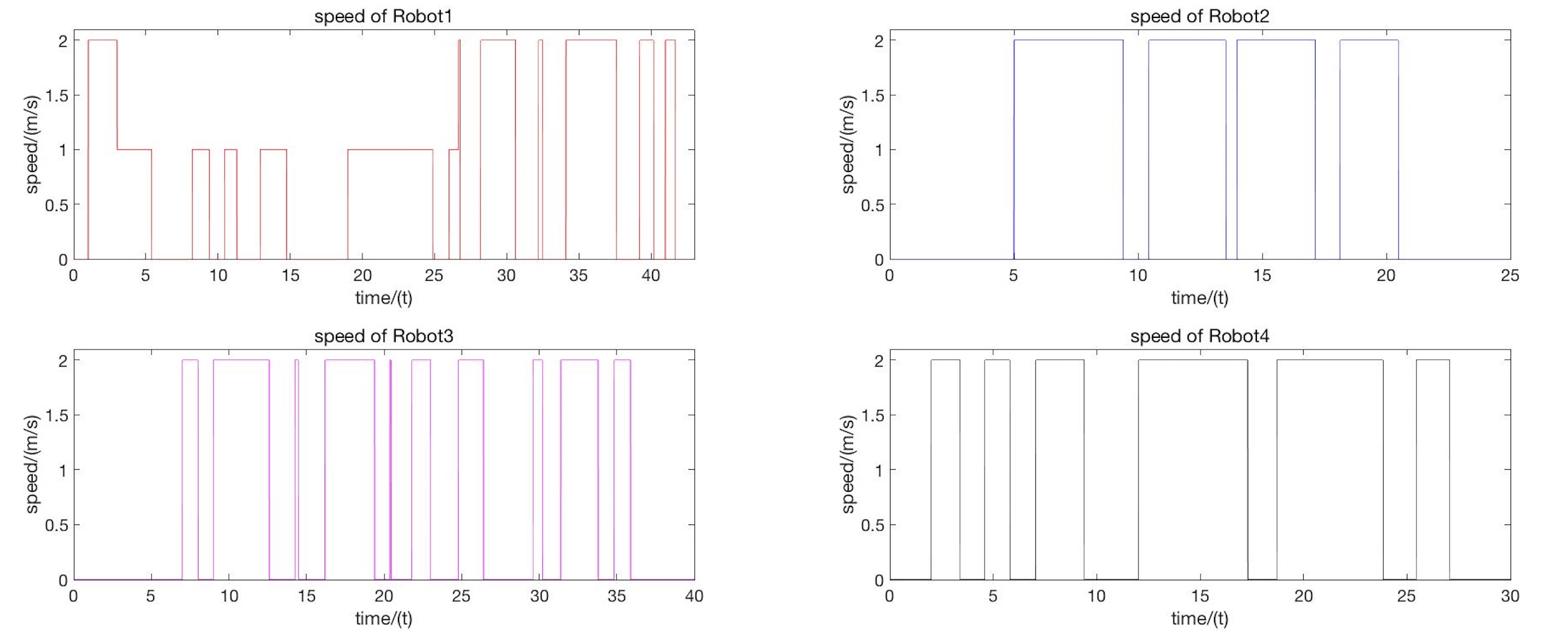


Figure 7: The speed variation plots

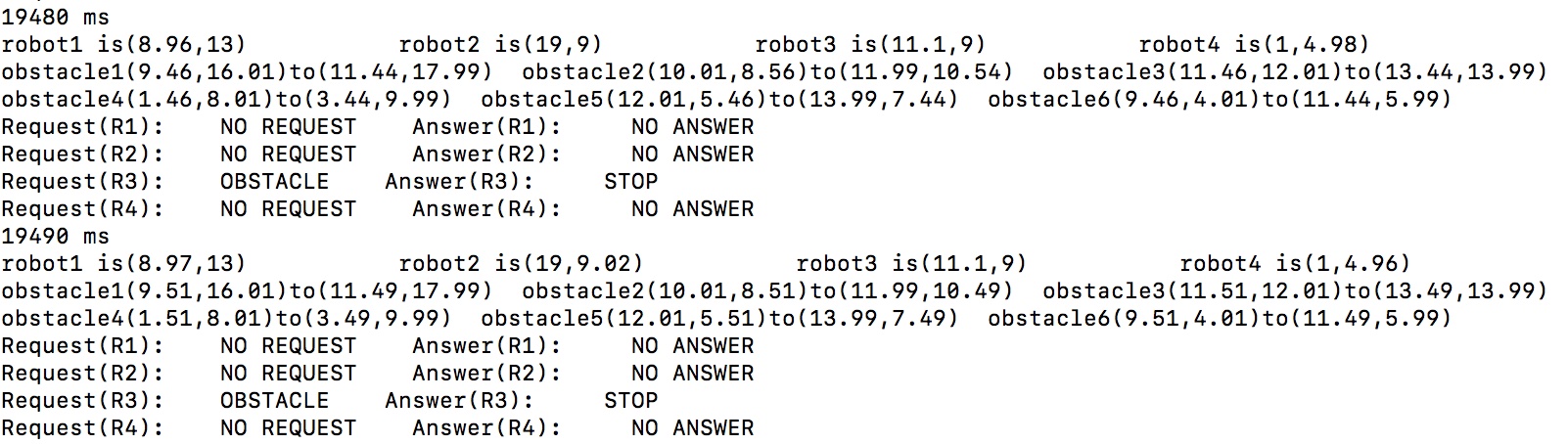
When the speed is 0, the robot may meet obstacle or just didn’t starts or other robots who have the priority to go through the node stop because of the obstacles.

**Stop because others stop situation:**

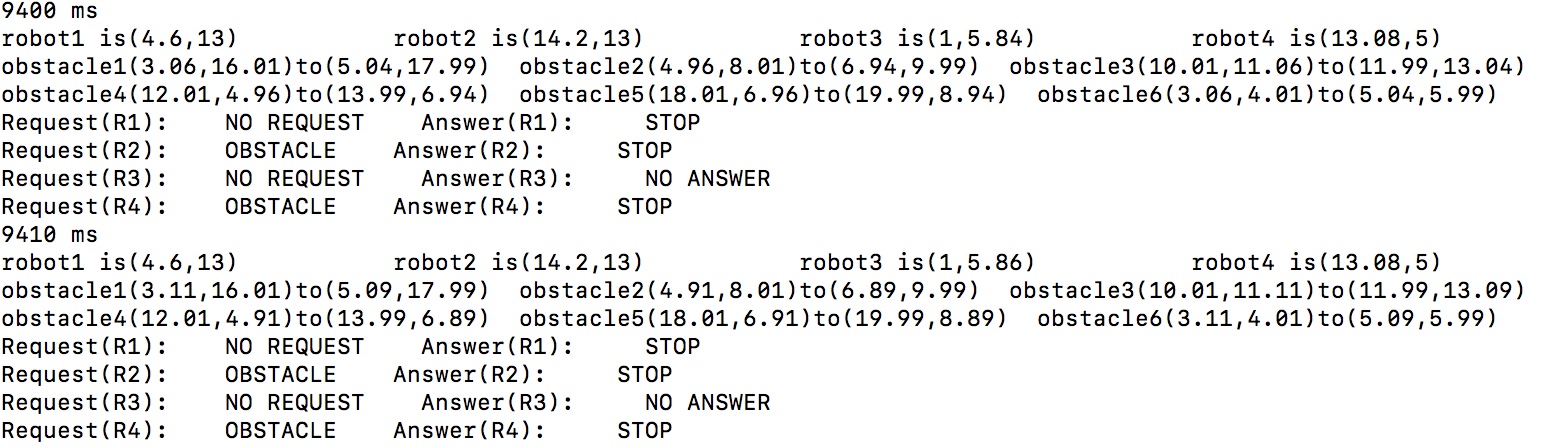
When two robots are now approaching a same node and one robot have priority, the robot who have priority meet the obstacles will also stop another robot, which because if it still navigate, the deadlock maybe happened.

And there are several situations that R3/R2/R1 have:

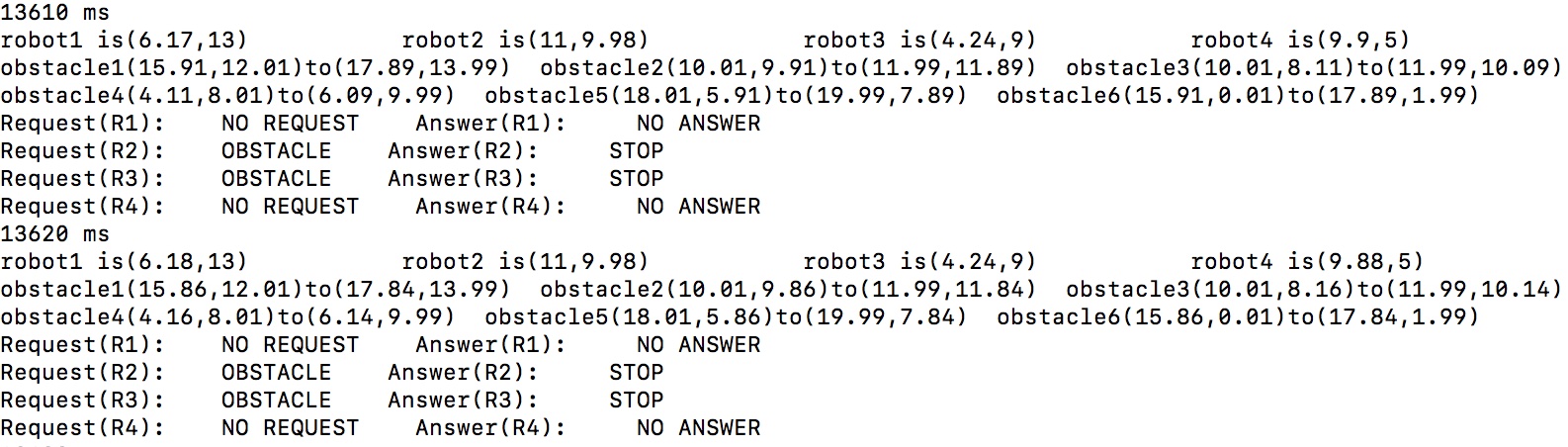
R3 stop, R1 not stop because R1 still not approaching node 7, it’s approaching node 4:



R2 stop, R1 stop too because they are both approaching node 4:

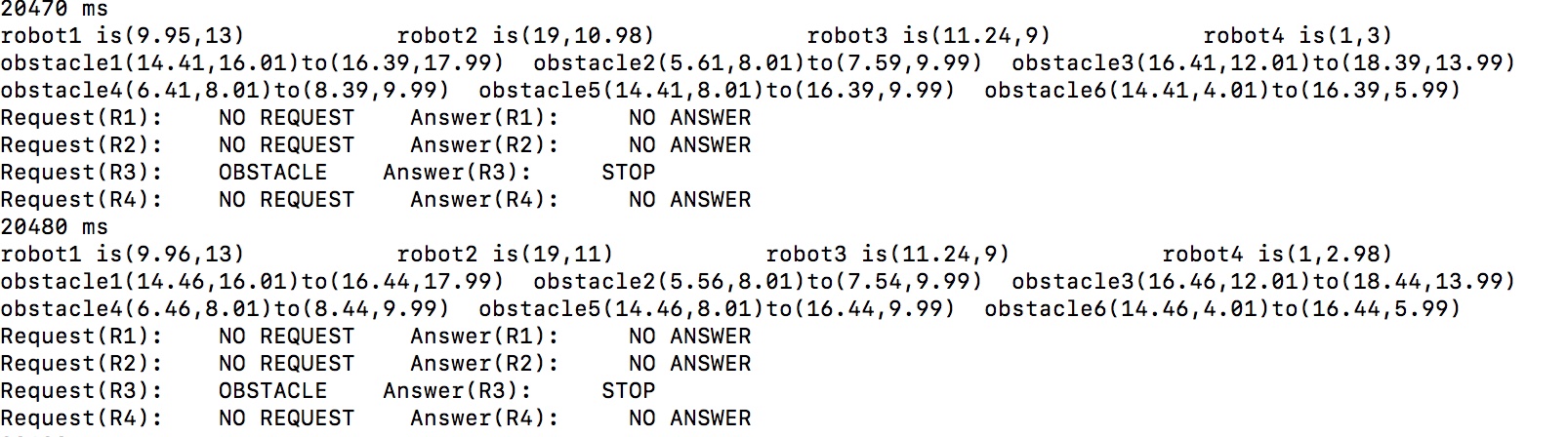


R2 stop, R1 not stop because they are not approaching the same node:

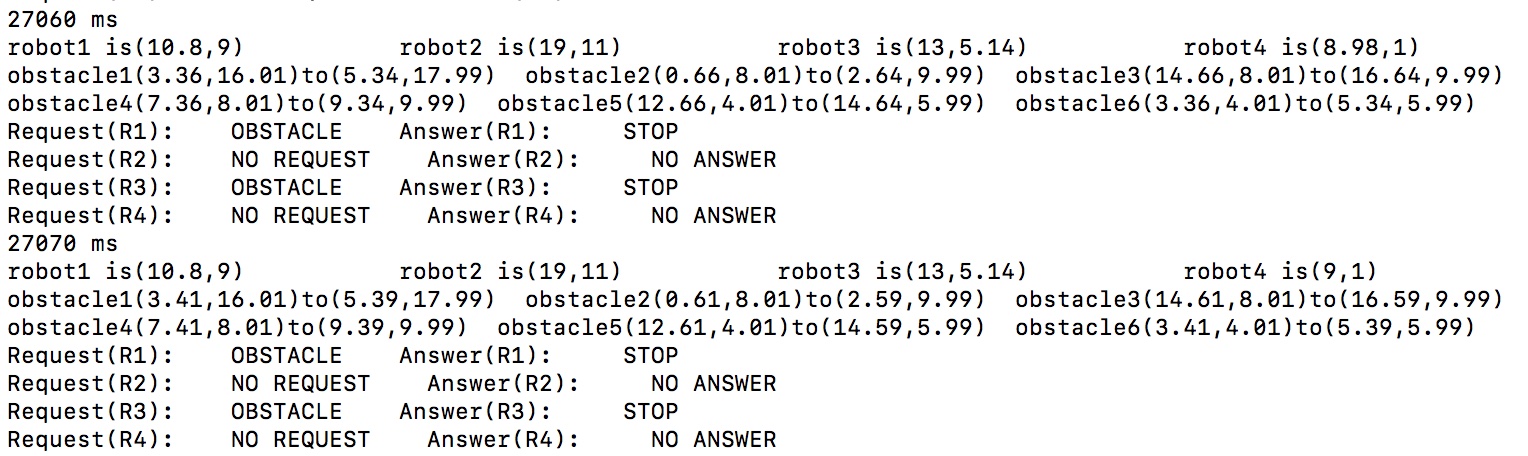


**End time:**

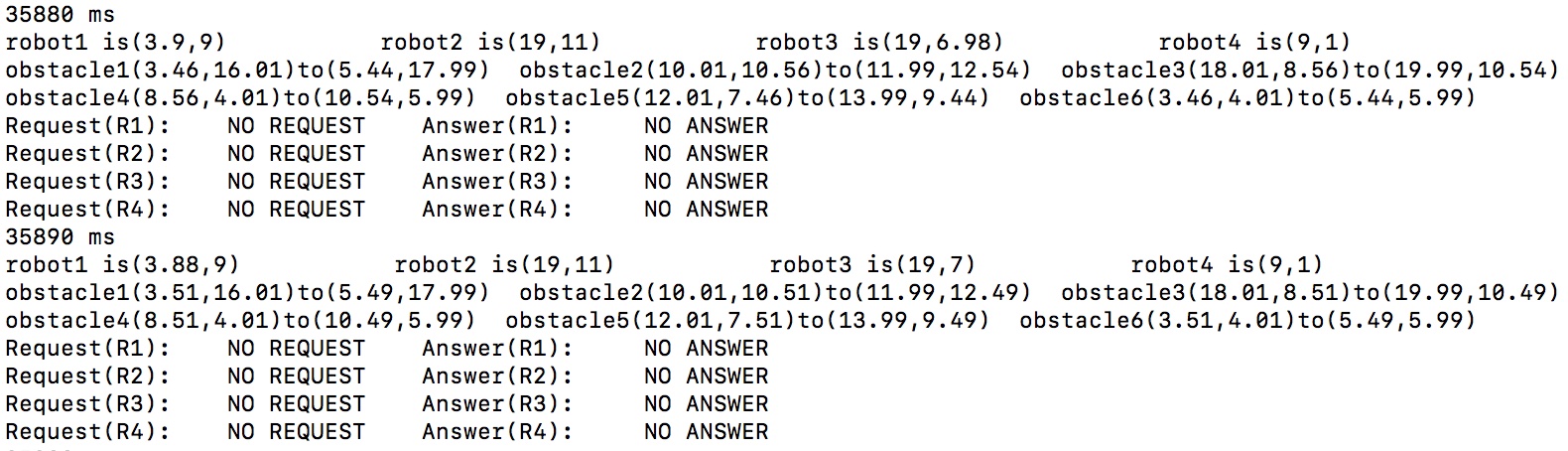
R2 arrives at the destination at 20.48s:



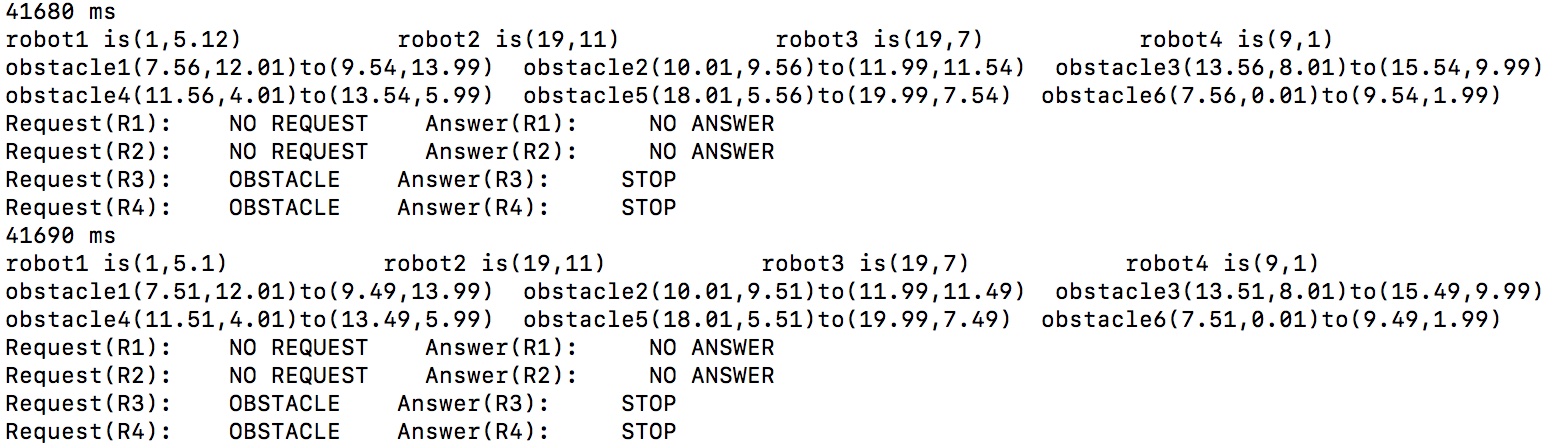
R4 arrives at the destination at 27.07s:



R3 arrives at the destination at 35.89s:



R1 arrives at the destination at 41.69s:



So the sequence the four robots arrive at the destination is R2、R4、R3、R1.

**Summary:**

With the speed control, the four robots navigate to their own destinations accurately without

deadlock. I am satisfied of my job I did in project 2 and look forward to doing well in project3.