**Robot Autonomy HomeWork3**

Chien Chih Ho

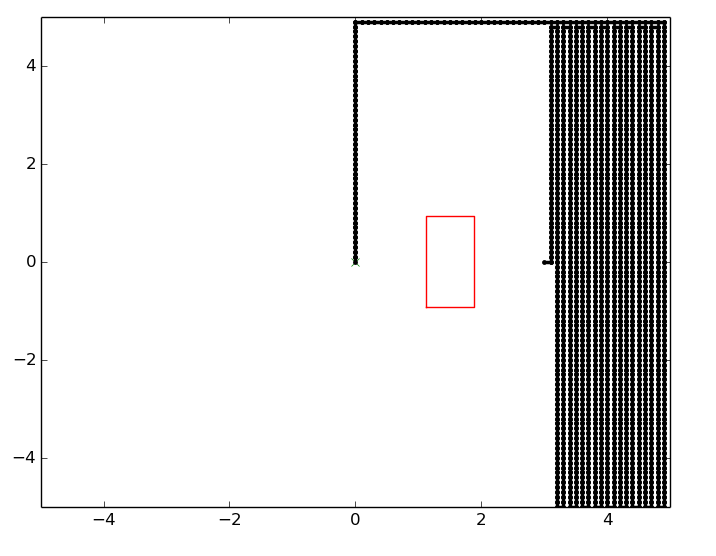
I-Chen Jwo

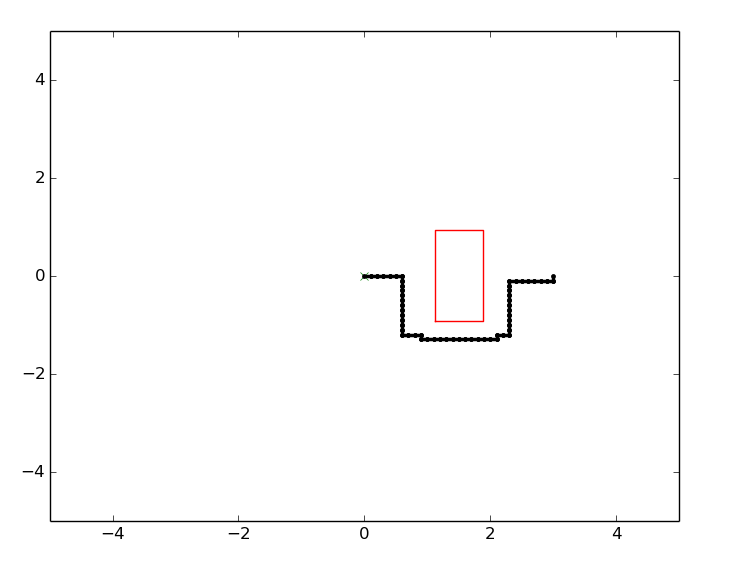
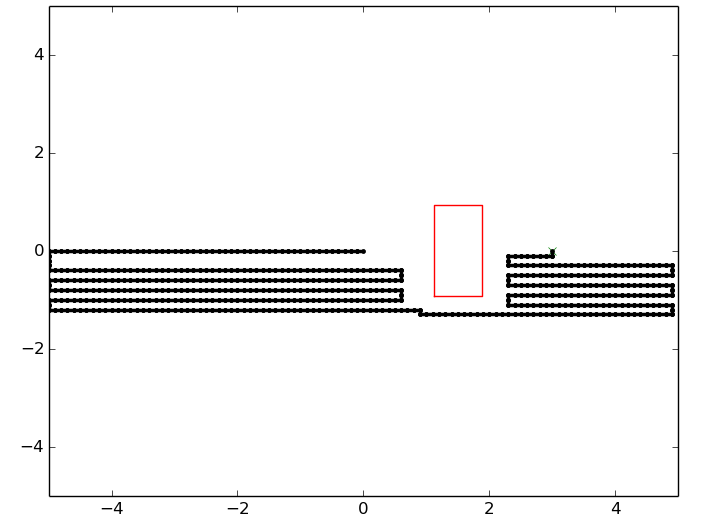
Luka Eerens

Question 1:

**2DOF**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Planner | Resolution | Path Length | Plan Time | # Of Nodes |
| BFS | 0.05 | 2.750000000000002 | 256.664994955 | 110 |
| BFS | 0.1 | 2.800000000000002 | 61.2189359665 | 56 |
| BFS | 0.25 | 2.75 | 47.5707991123 | 22 |
|  |  |  |  |  |
| DFS | 0.05 | 196.50000000001037 | 95.2153410912 | 7860 |
| DFS | 0.1 | 96.49999999999883 | 79.9091560841 | 1930 |
| DFS | 0.25 | 36.5 | 5.18593502045 | 292 |
|  |  |  |  |  |
| A\* | 0.05 | 51.95 | 254.562458038 | 2079 |
| A\* | 0.1 | 26.800000000000146 | 61.6172719002 | 537 |
| A\* | 0.25 | 18.0 | 4.69428014755 | 145 |

**BFS DFS**



**A\***

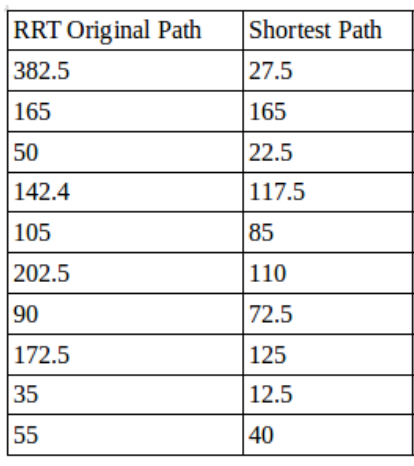
Question 2:

We are using two different A\* algorithms for 2D and 7D. We tried to make a general A\* but we have not been able to finish it in time.

We have tried to get this section done. We were short on time, and did manage to get some things working which included the visualisation of the robot arm moving. However it seems that due to pressure and timing, that we while trying to integrate the code, that we made changes that we were not able to backpedal on and ultimately led to the code having bugs.

Question 3:

Below is the RRT algorithm paths that we have obtained as well as the shortest paths we have obtained from them.



Comparing the median value to the A\* algorithms gives:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| RRT Original Path Length | RRT Shortened Path Length | A\* 0.05 Resolution Path Length | A\* 0.1 Resolution Path Length | A\* 0.25 Resolution Path Length |
| **105** | **85** | **51.95** | **26.80** | **18.00** |

The advantage of using an A star algorithm instead of using RRT with path shortening boils down mainly to the cost function of the A\* algorithm. The cost function provides a scalar feedback signal that allows the graph to grow in the direction of a target destination. RRTs propagate without any directionality and this may lead them to branch out away from the destination as the graph grows and this may take the computer longer to reach the destination, as more nodes will be created, and a greater distance to travel will be needed. The shortening algorithm cuts the fat of it, but it still had to build a big messy network of nodes before reaching the end goal. A-star is thus more efficient, as it has a forcing function to grow in the right direction.