ENPM 661 Final Exam

Planning for Autonomous Robots Faculty:
Reza Monfaredi

Mano srijan battula
UID: 118546490
mbattula@umd.edu

Department of Robotics
University of Maryland
United States

1. Give the best Big-O characterization for each of the following running time estimates (where n is the size of the input problem)

(a) $log(n) + 1000$

ans) $O(log(n))$

(b) $nlog(n) + 15n + 0.002n^2$

ans) $O(n^2)$

(c) $37n + nlog(n^2) + 50000log(n)$

ans) $O(nlog(n))$

(d) $1000n^2 + 16n + 2^n$

ans) $O(2^n)$

(e) $n + (n - 1) + (n - 2) + ... + 3 + 2 + 1$

ans) $O(n^2)$

(f) $2^{10} + 3^5$

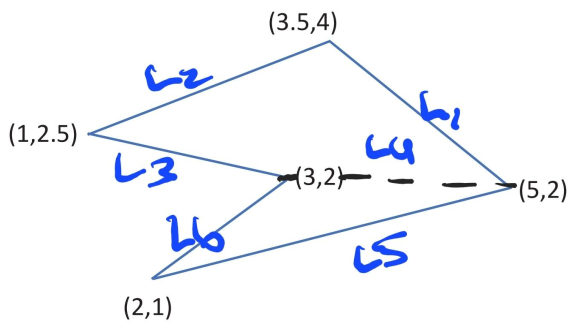
ans) $O(1)$

2) Compare RRT and PRM:

Rapidly Exploring Random Tree (RRT) and Probabilistic Roadmap (PRM) are sampling based motion planning algorithms used in robotics to find the best path for the robot to take from its starting point to its destination. The RRT technique starts by creating a list that contains the environment's obstacle information, the initial and target locations, the maximum edge expansion distance, and the RRT search tree. After that, the initial location is put to the search tree, and it is checked to see if it is the destination location. If this is the case, the path is produced by appending this objective point to the search tree. The algorithm is completed when the tree's output is returned.If that is not the desired location, a random point is chosen, the distance between that point and the nearest node is calculated, and the new node's location is determined. The closest point is found, added to the tree, and checked again to see if it is the objective location. This process is repeated until the new node is designated as the goal node. PRT, like RRT, starts with setting up the initial and objective locations, as well as environmental obstacle information, iterations, the maximum number of nearest nodes to each node, the maximum edge expansion distance, and a list of obstacles. The sample of these points is now checked based on the supplied number of points.If that doesn't work, a random spot is chosen and checked again. However, if the number of points has been sampled, a KD - Tree is formed using the sampled points, and the closest points in the tree are found. The number of dimensions of the environment is represented by K. The nearest points are next checked to see if they are blocked by anything. If not, they are all linked and a roadmap is created. The shortest path to the target from the initial place is found using a graph search method, and the decided path and roadmap are returned by the PRM algorithm, finishing the program.

3) A higher-dimensional abstract space is configuration space. It's a space that holds all of the robot's various configurations, such as position and orientation. A robot in c-space is referred to as a point robot.
The configuration space is essential because, once understood, many motion planning issues that appear to be different in terms of geometry and kinematics can be solved using the same planning algorithms.

4)

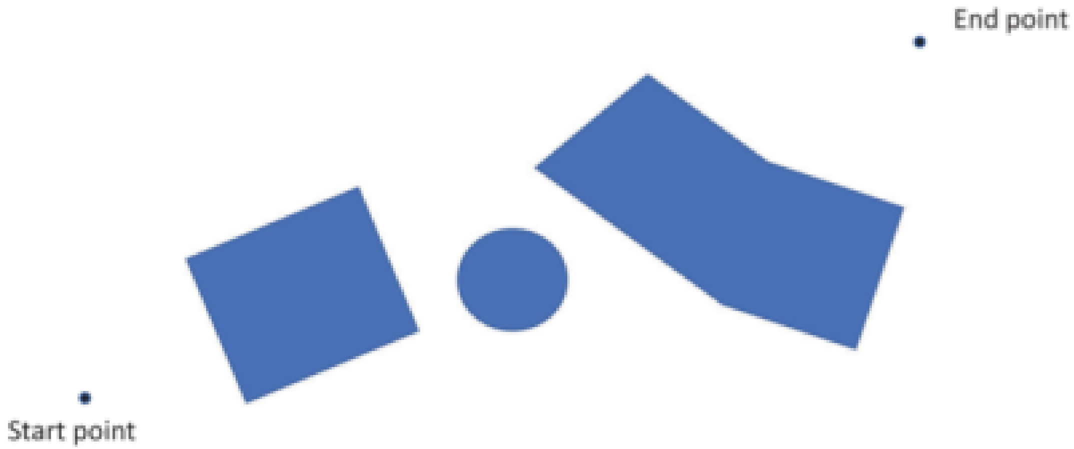


Lines and their corresponding half- planes:

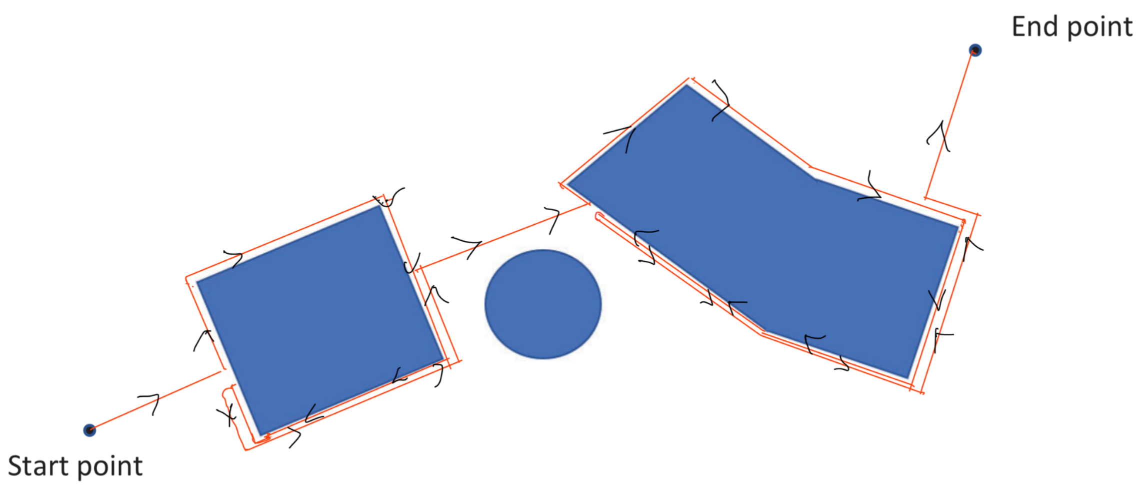
L1 = $f_1(x,y) = y + 1.33x – 8.66$, H1 = { $(x,y) \in W \mid f_1(x,y) <= 0$ }
L2 = $f_2(x,y) = y – 0.6x – 1.9$, H2 = { $(x,y) \in W \mid f_2(x,y) <= 0$ }
L3 = $f_3(x,y) = y + 0.25x – 2.75$, H3 = { $(x,y) \in W \mid f_3(x,y) >= 0$ }
L4 = $f_4(x,y) = y – 2$, H4 = { $(x,y) \in W \mid f_4(x,y) >= 0$ }
L5 = $f_5(x,y) = y – 0.33x – 0.35$, H5 = { $(x,y) \in W \mid f_5(x,y) >= 0$ }
L6 = $f_6(x,y) = y – x + 1$, H6 = { $(x,y) \in W \mid f_6(x,y) <= 0$ }

Obstacle representation in Union space :
O = (H1 ∩ H2 ∩ H3 ∩ H4) ∪ (H4 ∩ H5 ∩ H6)

5) Show the paths that would be generated using Bug1 and Bug2 algorithms for the Question 5: following map:

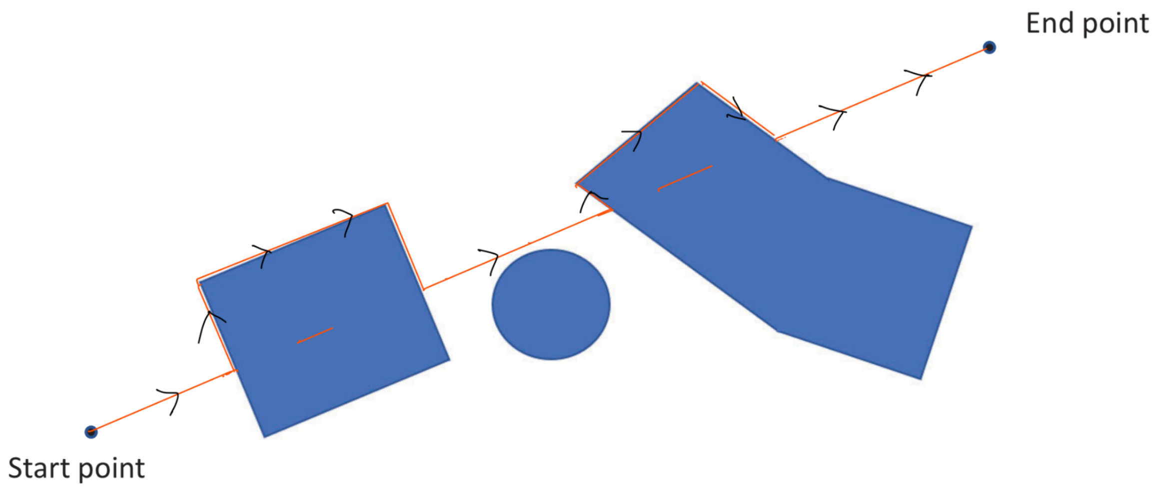


According to the bug 1 algorithm(exhaustive algorithm);
Steps:
When path to goal is blocked:
1) Go all the way around that obstacle.
2) Backtrack to closest point to the goal.
3) Continue toward goal.



According to the bug 2 algorithm(greedy algorithm):6

Steps:
1)Draw a line L from start to goal When path to goal is blocked.
2) Go around each obstacle until both at L and closer to goal than previously found obstacle L.
3) Continue toward goal.



6) Bezier control curvers for 4 control points:

$$P = (1 - u)^3P_1 + 3(1 - u)^2uP_2 + 3(1 - u)u^2P_3 + u^3P_4$$

Now Since we know the points are P_1, P_2, P_3, P_4

The equation for the points are:

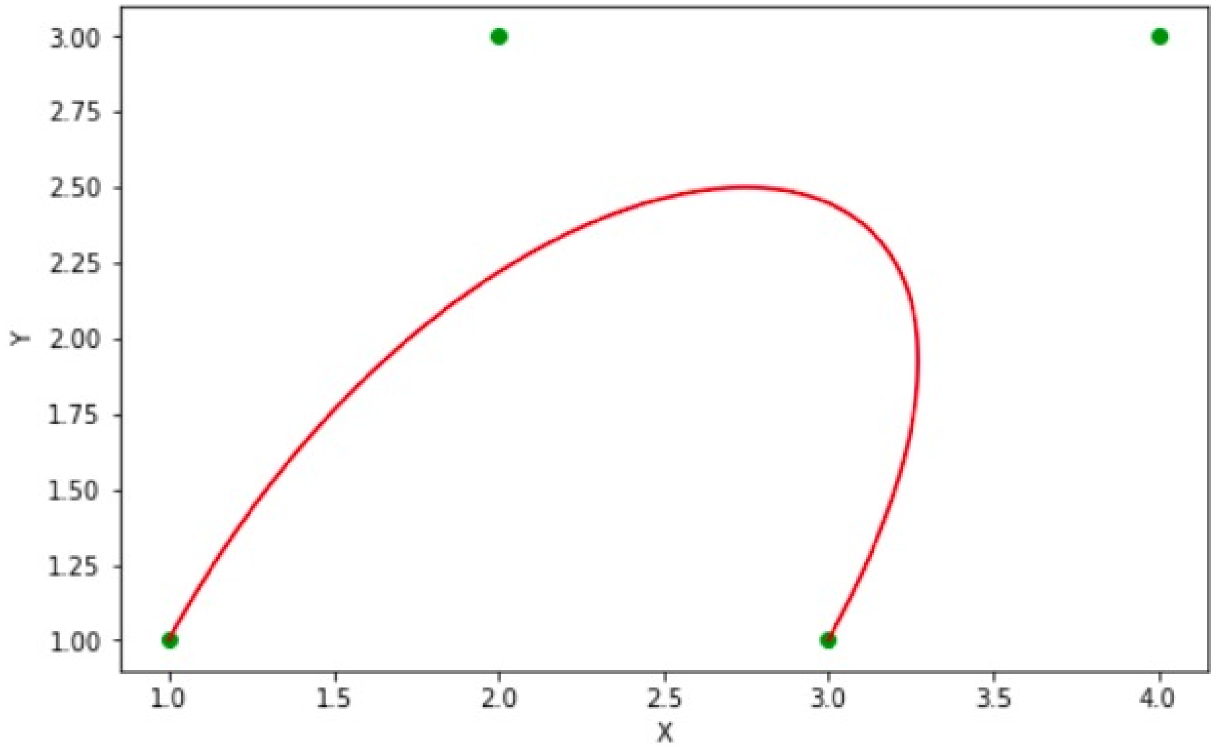
$$X(4) = (1 - u)^3X_1 + 3(1 - u)^2uX_2 + 3(1 - u)u^2X_3 + u^3X_4\ Y$$

$$(4) = (1 - u)^3Y_1 + 3(1 - u)^2uY_2 + 3(1 - u)u^2Y_3 + u^3Y_4$$

7) The points are :

Point	u	*x*	*y*
1	0.1	1.326	1.54
2	0.2	1.688	1.96
3	0.3	2.062	2.26
4	0.4	2.424	2.44
5	0.5	2.75	2.5
6	0.6	3.016	2.44
7	0.7	3.198	2.26
8	0.8	3.272	1.96
9	0.9	3.214	1.54

Bezier Curve



8)

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	

Now if the above is the final position it needs to be, we can use Manhattan Distance as a heuristic and it is calculated as follows:
Manhattan Distance for tile 1: 0
Manhattan Distance for tile 2: 1
Manhattan Distance for tile 3: 3
Manhattan Distance for tile 4: 5
Manhattan Distance for tile 5: 2
Manhattan Distance for tile 6: 1
Manhattan Distance for tile 7: 3
Manhattan Distance for tile 8: 0
Manhattan Distance for tile 9: 3
Manhattan Distance for tile 10: 1
Manhattan Distance for tile 11: 3
Manhattan Distance for tile 12: 4
Manhattan Distance for tile 13: 6

Manhattan Distance for tile 14: 2
Manhattan Distance for tile 15: 2
And the total heuristic value is :36

9)velocity tuning graph using vertical Cell decomposition method.