16-811 Math Fundamentals for Robotics
Assignment 6

Sajal Maheshwari
sajalm@andrew.cmu.edu

November 19, 2019

1 Collaborations

1.1 Question 4

I have used code/looked at algorithms from multiple sources for this assignment.
The URLs along with what they were used to implement are listed below :

1. To find orientation of thre ordered points : https://www.geeksforgeeks.org/orientation-3-ordered-points/
2. To check if two line segments intersect : https://www.geeksforgeeks.org/check-if-two-given-line-segments-intersect/
3. To check if a point lies inside a polygon : http://alienryderflex.com/polygon/

2 Code

All codes are written in Python.

2.1 Question 1

For question 1, the code is in the file q1.py. This runs the code to find the convex hull 'numTest' number of times. For each iteration, we generate 'numPoints' number of points, and visualize the convex hulls generated over each iteration. Some example results can be seen in Fig. 1

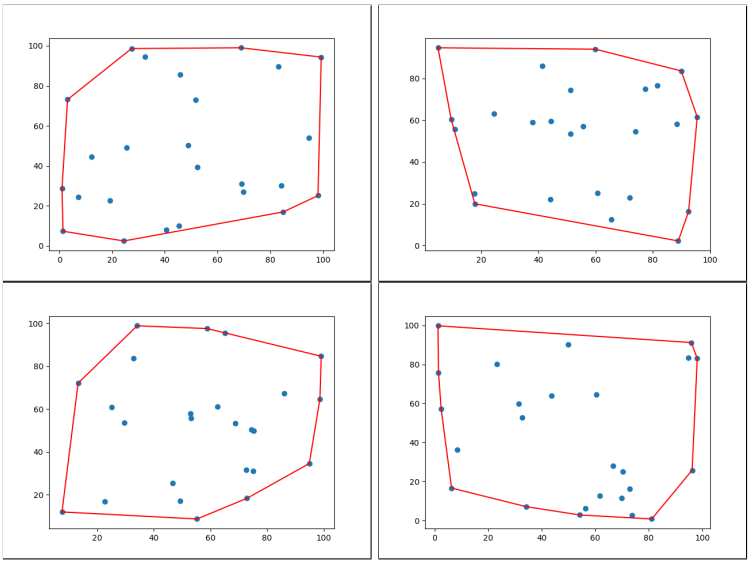

Figure 1: Figure shows convex hull of points in the graph.

2.2 Question 2

For question 2, the code is in the file q2.py. This code generates 'numPolygon' number of polygons. Each polygon is generated randomly using convex hull of some random points. The startPoint and endPoint are then chosen. The code outputs a graph showing the path from startPoint to endPoint. Some results can be seen in Fig. 2.

2.3 Question 3

For question 2, the code is in the file q2.py. This code generates 'numPolygon' number of polygons. Each polygon is generated randomly using convex hull of some random points. The startPoint and endPoint are then chosen. The code outputs a graph showing the path from startPoint to endPoint. Some results can be seen in Fig. 3.

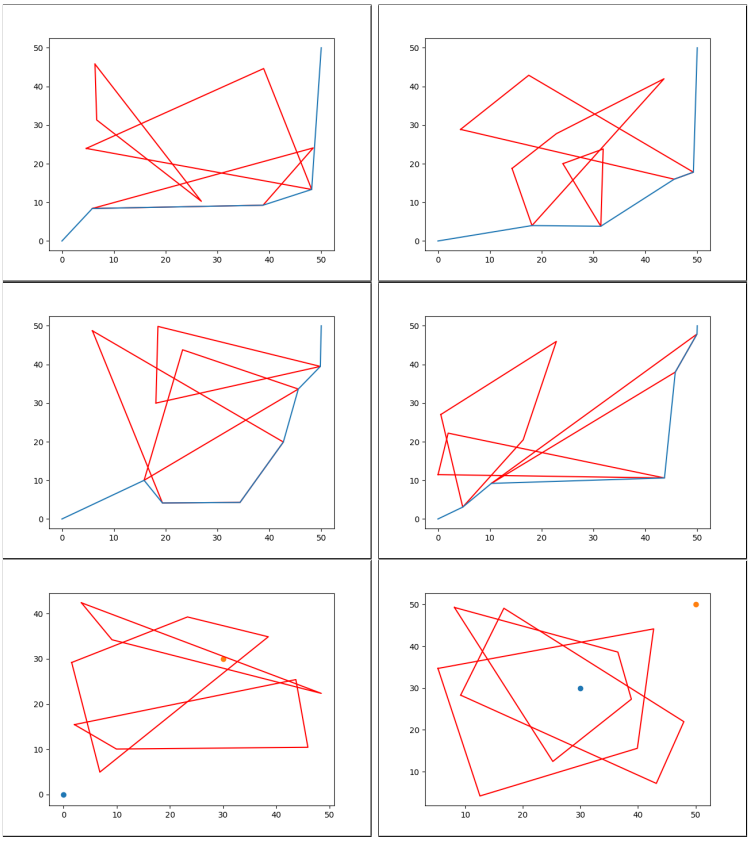

Figure 2: Figure shows path from start to end point in blue, while the obstacle boundaries are shown in red. The start and end points where no path is possible can also be seen, with start point in blue and end point in orange.

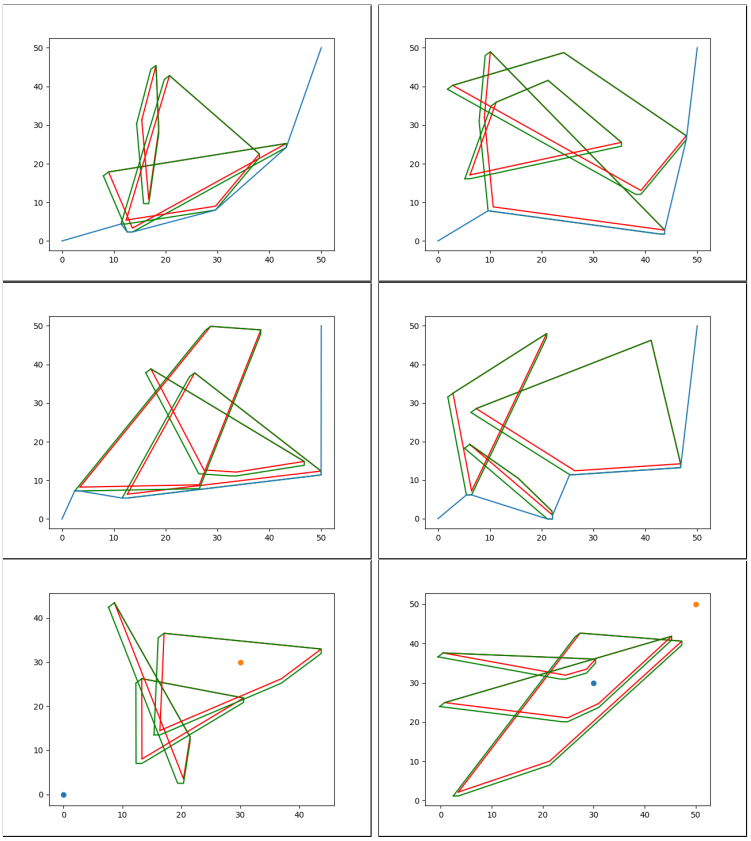

Figure 3: Figure shows path from start to end point in blue, while the obstacle boundaries are shown in red. The regions not allowed for the robot to enter due to its size are bounded in green. The start and end points where no path is possible can also be seen, with start point in blue and end point in orange.