Shortest Paths

Karen Li - https://github.com/kkl224/MobileRobotics

***Abstract*— The goal of this lab practice is to implement and test algorithms to move in an environment with obstacles using the planning technique of finding the shortest path and trajectory planning.**

I. Methodology

A. Map the Environment

I used the mats in the lab as grids and mapped a 6x8 mats space for this experiment. I treat each non-obstacle map as a node and black out the obstacles.

B. Define a start & end point for BFS

Once I determine the starting and ending point, I used **Network X** to find the shortest path.

C. Convert the path result to positional information

I then converted the shortest path to positional information in the lab and set them as way-points for the robot to follow.

II. Conclusion

A. Can you make a robot continuously follow a loop?

Yes. There are 2 possible ways to accomplish this goal. The first approach is, instead of making the robot following a path, we should make it following a cycle. We can use the function *simple_cycles* from **Network X** to compute a simple cycle and traverse each node one after another infinitely. The second approach is simply going back and forth with the shortest path found; That is, when you've reached either end, you start heading towards the other end.

B. How would you solve the problem if you don't know where the obstacles are?

If I do not know where the obstacles are, I will need the help from external sensors, like an infrared or ultrasonic sensor, or a camera to detect the obstacles. There are also 2 approaches to solve this problem. First is to send the robot doing a obstacles mapping in the environment so that it knows where the obstacles are. Then we can do what we did for this lab to compute the shortest path so that the robot knows where to go. The second approach is to do a random walk mapping. As it detects the obstacles in the environment, it starts to keep a record of them and update these information on its map. Eventually, it will map out the environment and be able to head towards the destination.

III. Link to the recorded video

https://drive.google.com/drive/u/0/folders/1B_OpH78jWSxmbH-nFRTLQw57f6jRC3R1

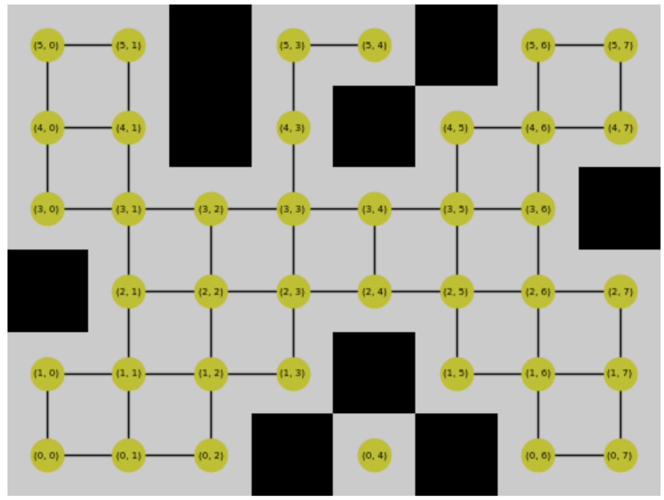

Fig. 1. *6 x 8 mats lab space with each node representing available way-point to traverse and black boxes representing obstacles.*

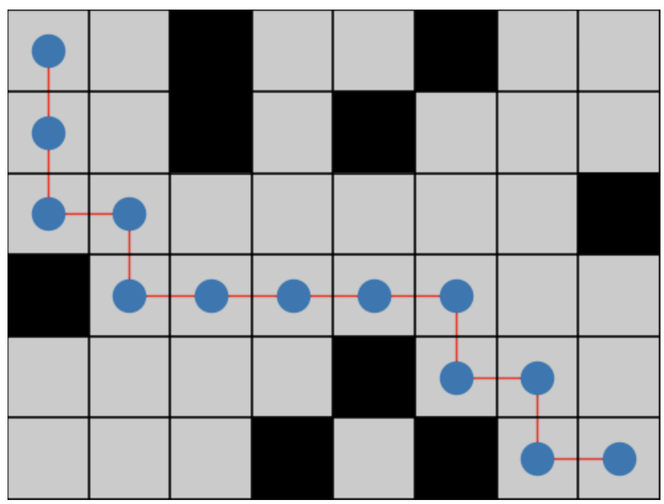

Fig. 2. *Shortest path found using BFS. Starting at (5, 0). Ending at (0, 7)*

```
Node 1 :  [-1.0, -2.5]
Node 2 :  [-1.0, -1.7999999999999998]
Node 3 :  [-1.0, -1.0999999999999996]
Node 4 :  [-0.30000000000000004, -1.0999999999999996]
Node 5 :  [-0.30000000000000004, -0.3999999999999999]
Node 6 :  [0.3999999999999999, -0.3999999999999999]
Node 7 :  [1.0999999999999996, -0.3999999999999999]
Node 8 :  [1.7999999999999998, -0.3999999999999999]
Node 9 :  [2.5, -0.3999999999999999]
Node 10 :  [2.5, 0.30000000000000004]
Node 11 :  [3.1999999999999993, 0.30000000000000004]
Node 12 :  [3.1999999999999993, 1.0]
Node 13 :  [3.8999999999999995, 1.0]
```

Fig. 3. *x and y coordinates of the shortest path in lab*