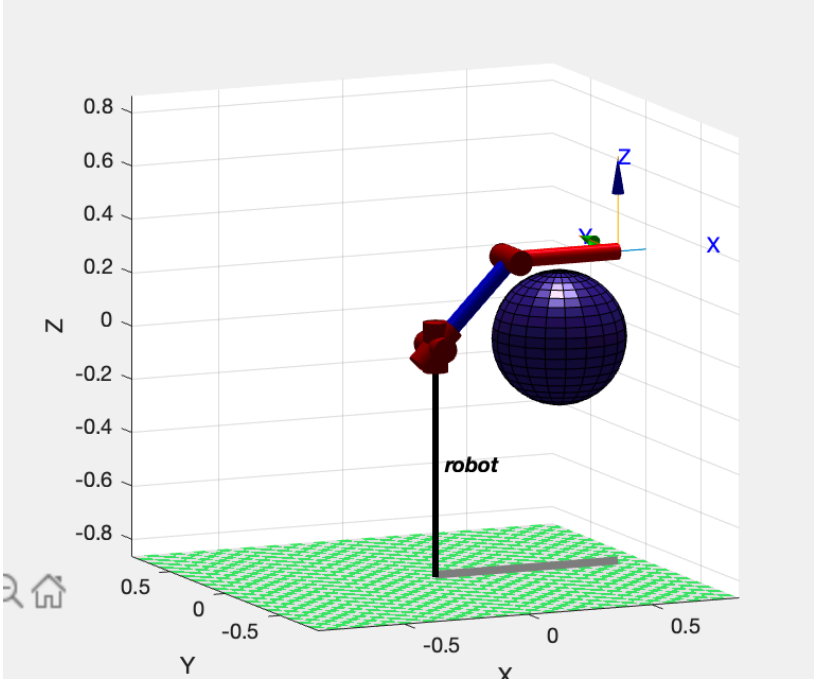
CS5335 HW3

Yiqiu Huang

C0



The function `in_collision` check collosion by create resolution point among link, if the distance from resolution point to sphere's centers is within sphere's raduis, then this is collision and it return `true`

The function `chack_edge` is creating ticks among different configuration, and will check collision when the robot is moving from one config to another config. By calling `in_collision`, it will return true if there is any collision.

Potential issue:

- There might be swept value collision between ticks.
- The algorithm will take long time when resolution is high

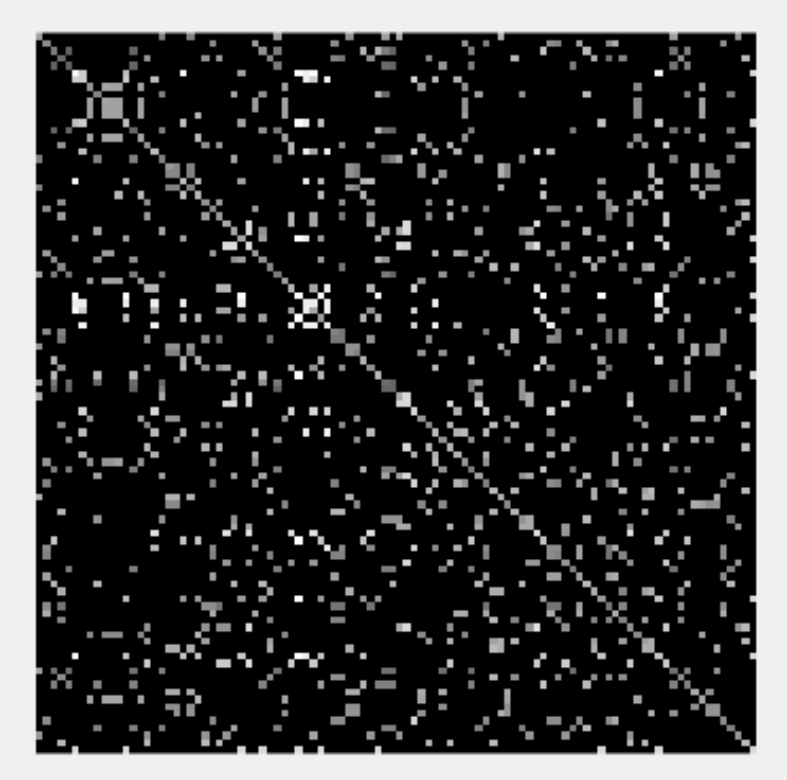
c1

Generate random point from config space

Number of samples: 100
Number within bounds: 100
Number in collision: 11

c2

Implement PRM algorithm to sample 100 points from free config space. The below figure is the adjacency matrix about the distance from sample point to another sample point.



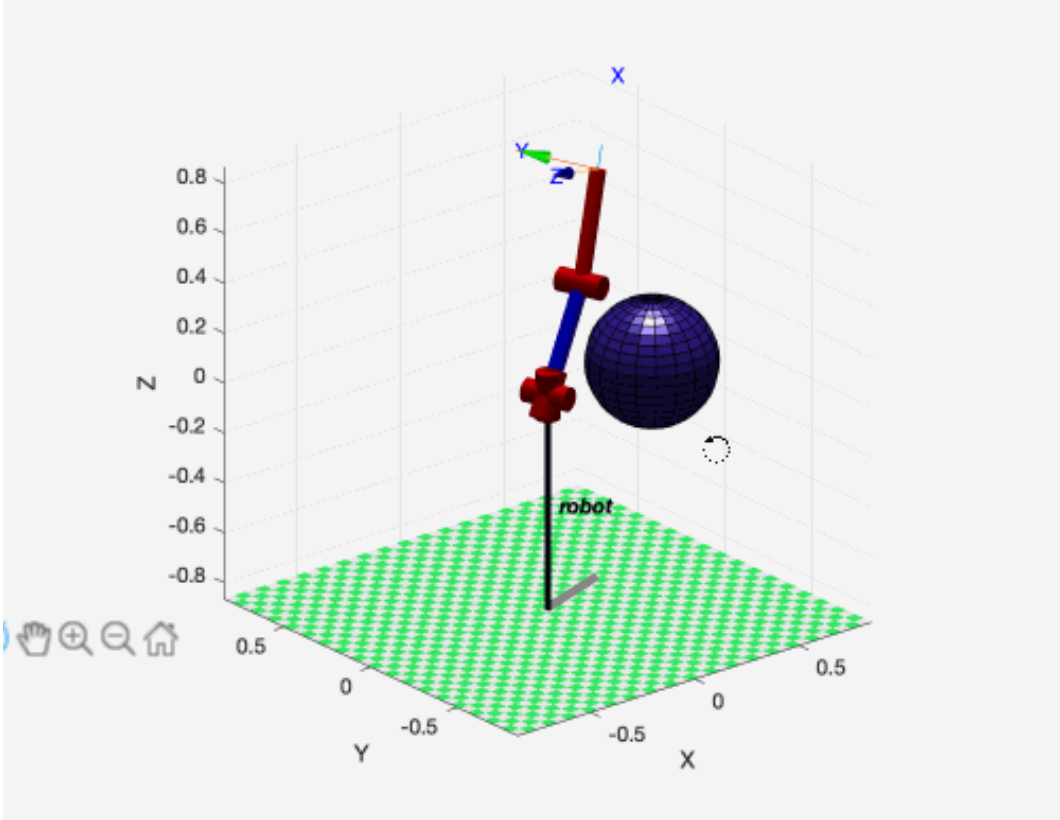
c3

Implement the PRM algorithm to find the path from approximate start config to approximate goal config with the adjacency matrix from c2.

Path found with 4 intermediate waypoints:

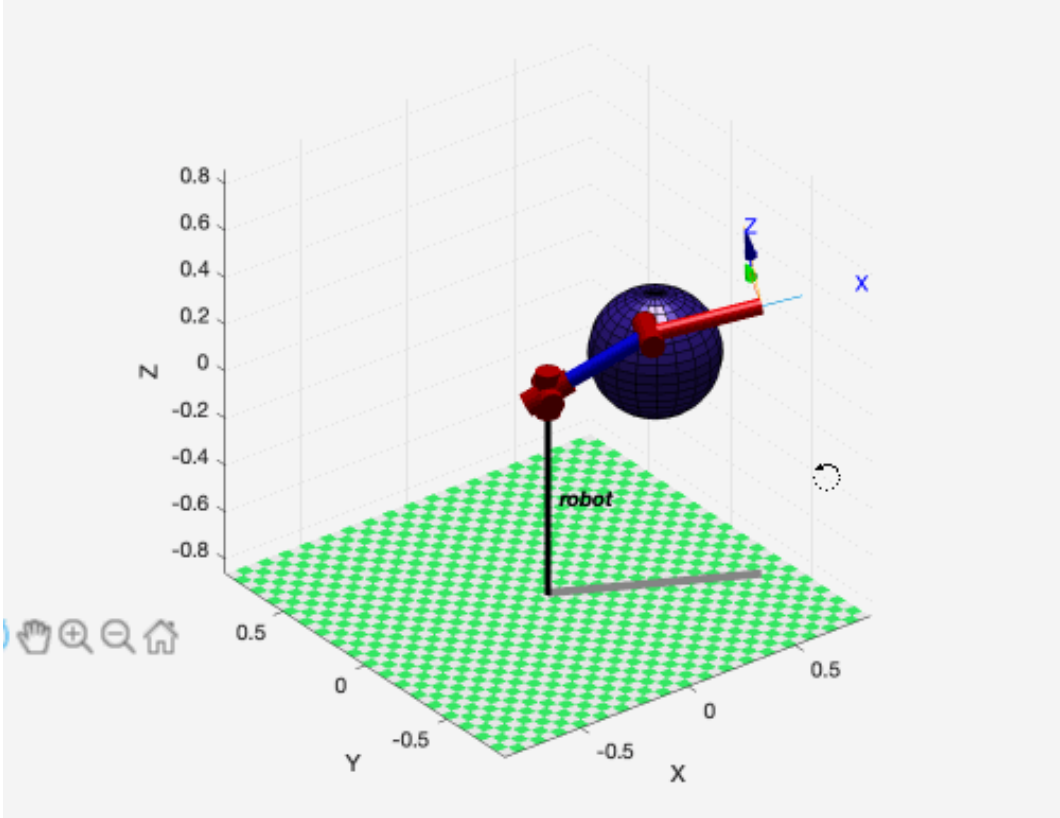
-0.0060	-0.3115	0	-0.4864
0.1855	-0.4584	0	-1.7404
0.7217	-0.9197	0	-2.2369
1.1395	-1.6201	0	-2.4837
0.9109	-2.1409	0	-2.8590
0.0970	-2.8541	0	-2.8122

Robot moving animation



c4

Path found with averagely around 30 intermediate waypoints



c5

Create another graph to simplfy the path. Those previously non-consecutive pair of point will attempt to be connected in the smoothing algorithm. The new graph denote the reachability of previous non-consecutive pair of point. After running graph finding algorithm again in the new graph, it's averagely simplified to around 3 point.