RBE 550: Motion Planning
HW4 – RRT and RRT* Implementation

Dheeraj Bhogisetty
dbhogisetty@wpi.edu

RRT Algorithm:

Given a start and goal position in a map:

1. Create a tree and add the start position as root node.
2. Sample a random configuration from the map and if it is an obstacle discard the sample.
3. Find the nearest node in the tree to the sample node.
4. Create a new node in that lies on the line connecting nearest node and sample node and is at a distance of "step_size" from the nearest node.
5. If the new node is not an obstacle and if the connecting edge is collision free, add the node to the tree.
6. If the new node is in the neighbourhood of the goal position and the path is collision free then break the loop and connect to the goal node.
7. Otherwise repeat, steps 2 – 6 until max iterations.
8. Once goal node is found backtrack the path to the starting node.

RRT* Algorithm:

RRT* algorithm is very similar to the RRT algorithm except that the nodes in the neighbourhood of the sampled node are rewired for minimum cost (shortest path).

1. After step 4 in the above algorithm, find the neighbours of new node that are within a fixed distance from it.
2. From the neighbours, connect to the node which would result in minimum cost.
3. For each node in the neighbourhood, rewire the node (change parent to new node) if the resulting cost is less than its existing cost.
4. Continue from step 6 of above algorithm

Difference between RRT and RRT*:

- The main difference is in the way the trees are expanded in RRT and RRT*.
- In RRT* the nodes of the tree in the neighbourhood of the new sample node are rewired for minimum cost path. Whereas in RRT, new node samples are added without considering the cost of the path.
- Therefore, RRT* produces more optimal paths than RRT.
- But since RRT* has to rewire the nodes, it is computationally more expensive and slower when compared to RRT.

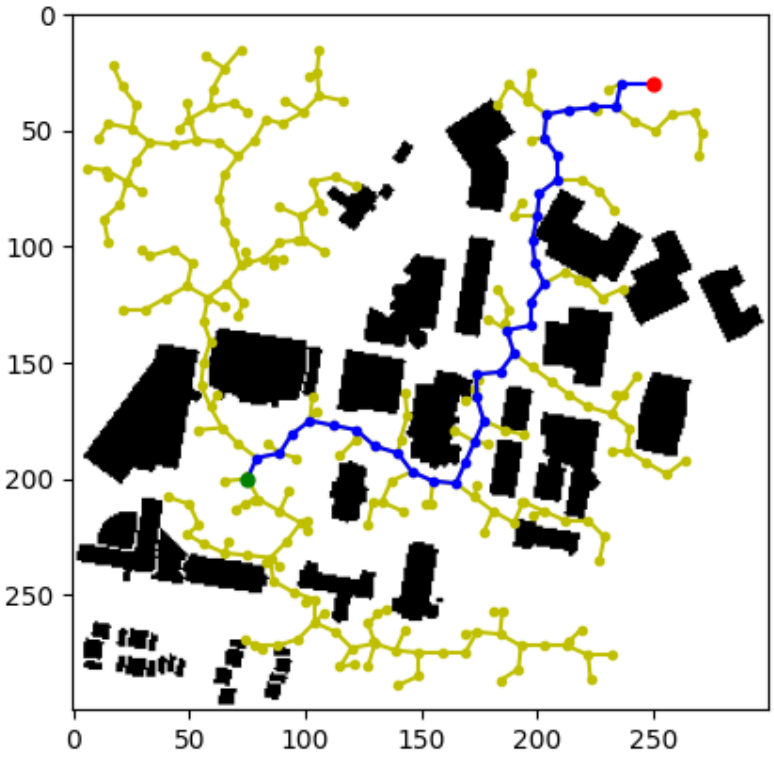
PRM vs RRT:

- PRM is a multi-query method whereas RRT is a single-query method
- In PRM roadmaps are calculated once and stored. So, it can calculate optimal paths for each query when given multiple queries.
- RRT builds the tree for each query and doesn't store the tree.
- RRT is particularly preferred for planning in higher dimensions as constructing a roadmap can be computationally expensive.

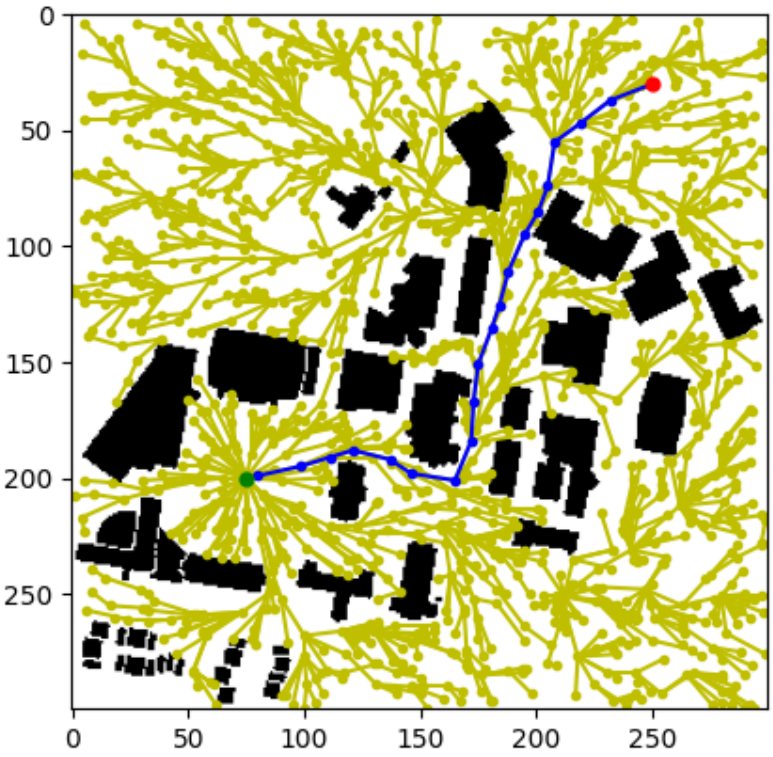
- Since PRM needs to rebuild roadmaps if the environment changes, it is not suitable for planning in dynamic environments. RRT is much faster and more efficient for planning in dynamic environments.
- PRM can generate optimal paths, whereas RRT doesn't guarantee an optimal path. However, RRT* can produce optimal paths.

Results:

RRT: (for 1000 samples)
It took 262 nodes to find the current path
The path length is 356.16



RRT*: (for 2000 samples)
It took 1439 nodes to find the current path
The path length is 305.76



From the results it can be seen that RRT* produces an optimal path but it is slower as it expands more nodes in order to find an optimal path. Whereas RRT is faster and it terminates as soon as it reaches the neighbourhood of the goal node.

References:

- RRT* explained - https://www.youtube.com/watch?v=_aqwJBx2NFk
- RRT and RRT* implementation – https://theclassytim.medium.com/robotic-path-planning-rrt-and-rrt-12319121378#:~:text=RRT*%20is%20an%20optimized%20version,to%20develop%20a%20shortest%20path.
- https://en.wikipedia.org/wiki/Rapidly-exploring_random_tree