LABORATORY II
Planning and Navigation

PRA0102 - Group4

Gun-hui Han

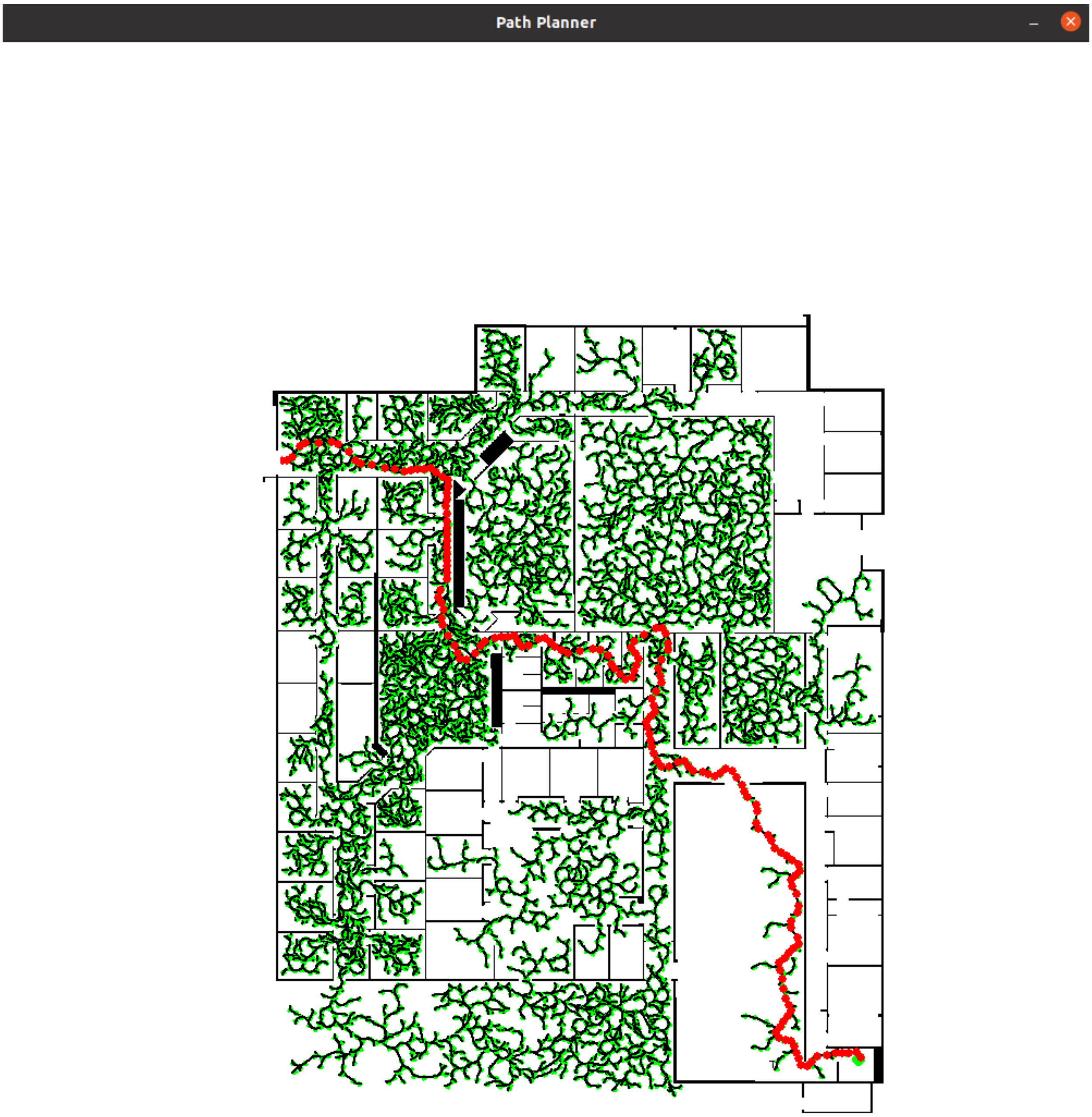
Seonghak Lee

Jisoo Park

ROB521 Mobile Robotics and Perception Spring 2024

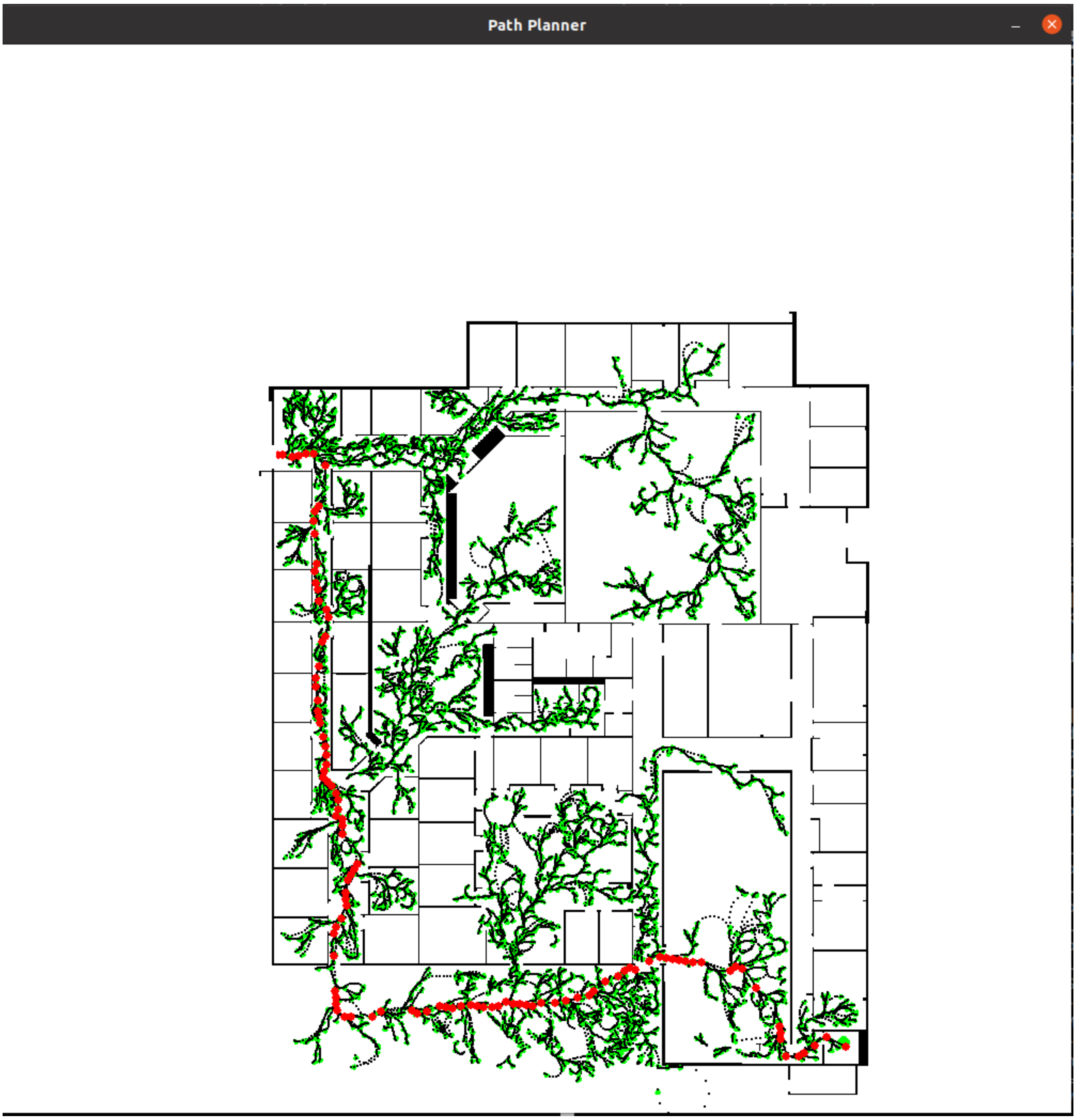
Part A

1. A picture with a successful output from RRT



The Rapidly-exploring Random Trees (RRT) algorithm starts with initializing a tree with a start node. It iterates over these steps: randomly samples a point in the space(sample_map_space), finds the nearest node in the tree to that point(closest_node), and generates a path towards the point while obeying the robot's motion limits(simulate_trajectory). If this path doesn't collide with obstacles(check_collision), it adds a new node at the end of the path. This process repeats until it finds a node close enough to the goal. Finally, it reconstructs the path from the goal to the start by tracing back through the parent nodes(recover_path).

2. A picture with a successful output from RRT*



RRT*, an enhancement of RRT, follows a similar initial process of sampling, finding nearest node, path steering, and collision checking. However, after adding a new node, it looks for nearby nodes and checks if this new node can offer a shorter path to them (rewiring), optimizing the overall path costs in the tree. The cost to come for each node is calculated using cost_to_come function. When rewiring is attempted, it generates paths between nodes(connect_to_node) and update the cost and parent of rewired nodes(update_children). It keeps iterating, adding, and possibly rewiring nodes to ensure most optimal path is found. The path is then recovered by tracing back from the goal node to the start.

3. A video of trajectory rollout

Please refer to attachment : partA_3.mp4

Part B

1. A video of trajectory rollout on the new Myhal map

Please refer to attachment : partB_1.mp4

2. Already marked by TA during lab session.