I. Path planning method

The path planning method which we choose for the drone mainly depends on the environment and type of obstacles present in the environment. So, before choosing the path planning method, it is important to get familiarize with the obstacle space of the environment. By examining the obstacle space, we can choose which path planning method to use.

A. *Types of Obstacles*:

There are two types of obstacles. They are

1) **Static obstacles**: These obstacles does not move with respect to time. So, the obstacle space does not change with the passage of time. In an environment consisting of static obstacles, the generation of collision free optimal path happens only once.
2) **Dynamic obstacles**: These obstacles move with respect to time. So, the obstacle space constantly keep changing with the passage of time. In an environment consisting of dynamic obstacles, the collision free optimal path changes continuously as dynamic obstacles can come in the way of the path. Thus, the optimal path changes.

As we now have knowledge regarding the types of obstacles, we can develop an approach for path planning. The following approach is followed for path planning:

1) **Graph generation**: The first step of any path planning method is to generate a probabilistic graph of the environment. In order to generate a map, we use RRT graph generation algorithm. The RRT algorithm also takes into account the differential constraints of the drone.

   RRT (Rapidly Exploring Random Trees): In RRT, the map is explored with the help of trees beginning from the start configuration. Each node of the tree is randomly generated and connected to the end of the tree if the node generated is not in the obstacle space. If the node generated is in the obstacle space, the nearest edge point which is outside of the obstacle is selected.

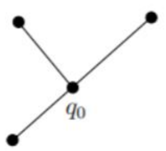

Figure 1. RRT

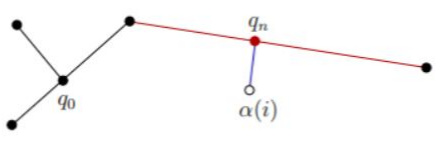

Figure 2. Connecting the tree to nearest edge point

2) **Cost Function: Potential Field Map** After generating the probabilistic graph, we need to find the collision free path. Potential field map is used to direct the search of collision free path by assigning priorities to each point in RRT based on its closeness to the obstacle.

$$p(P) = p_q(P) + p_O(P) \tag{1}$$

   where P is a random point in RRT, p_q is the contribution to the potential field map of the planning query and p_O is the contribution to the potential field map of the obstacles of the environment. In simpler terms, what we are doing is generating a cost for all the point generated by RRT algorithm. The cost generated for each point is the sum of contribution given by the obstacles and path.
   If $p_O > p_q$, this indicates that the point is closer to the obstacles and is given less priority.
   If $p_q > p_O$, this indicates that the point is closer to the path towards the goal point and is given more priority.
   The idea of potential field map is used to guide the graph search algorithm (described in next section) and therefore, our planner never falls in a local minima when calculating the collision-free path, computing always the most optimum path.

3) **Graph search algorithm:** The task of finding the optimal path is done by A^* algorithm. A^* algorithm finds the minimum cost path that connects the initial (current), P_O, and desired state, P_f , in the probabilistic graph. The A^* algorithm also takes into consideration the cost of the each point generated from the potential field map. This helps the algorithm to ignore the points which are closer to the obstacles. By taking into consideration the potential field map cost, A^* algorithm becomes more robust in finding the optimal path.

II. Software Used

1) Python version 3
2) Python version 2.7
3) Matplotlib
4) ROS
5) Gazebo
6) Rviz

III. Team members

1) Eashwar Sathyamurthy
2) Akwasi A Obeng