Team F

Individual Lab Report 9

Progress Review 10

Yuchi Wang

Teammates
Danendra Singh
Pulkit Goyal
Rahul Ramakrishnan
Pratibha Tripathi

April 6, 2018

1 Individual Progress

For this PR, I worked on the Bebop 2 navigation outdoors. This consisted of implementing the heuristic and testing the robustness outside of Wean Hall.

1.1 Heuristic

The heuristic used for the drone is Euclidean distance to the target. This is calculated by using the Haversine formula for GPS coordinates. During testing, we have used the CFA lawn as the final destination. Internally, the algorithm goes through all the vertices in the graph, calculates the distance between the GPS coordinate of the vertex and the target location, and selects the tag closest to the destination. The drone proceeds to explore this tag.

1.2 Pruning paths

In addition, it is necessary to keep track of which vertices have already been visited. The way that the algorithm does this is by checking the location of the drone against each April Tag every iteration. If the location of the Tag is within a certain distance of the drone, then it marks that specific Tag as explored. This tag is then ignored when searching for a new target to explore. By doing so, we guarantee that we will eventually explore all the nodes in a graph.

1.3 Outdoor testing

Testing the drone was also a major component for this PR. Most of the testing was done on March 17 during the Spring Break. Two main videos of the tests are available and linked below. The first video demonstrates the exploration ability of the drone. At the beginning, the drone does not know the path to the final destination. It sees the April Tags on the ground and keeps exploring the Tag closest to the goal. By doing so, it is able to find paths even if they are curved and non-linear.

The second video demonstrates the pruning ability of the drone. Here we have two branches off the starting node. The first branch points directly to the goal and it is what the drone explores first. However, as it explores, it marks each visited node as explored and once it reaches the end, it realizes that all nodes in the current branch have visited and thus it goes onto to expand the second node.

Videos:

1. `https://drive.google.com/open?id=15UD7eO6R7nxOMaabjiQEawc-s6PKswul`
2. `https://drive.google.com/open?id=13qDHri44OTqtip5wjYDDxpw5Kbyb3cNR`

2 Challenges

The biggest challenge has been with testing the drone outdoors and managing everyone's time. Testing the drone outdoors requires good weather and minimal bystanders. Since this was over the spring break, the amount of people were minimal but the weather could have been improved. There was a lot of wind which made detection of April Tags harder. In addition, it was very cold and not pleasant to work outdoors.

Testing the drone should also be ideally done with more than 1 person. This is to watch for other people and to prevent the drone from losing control (and getting stuck in a tree). Unfortunately, due to everyone's plans during the break, I was unable to coordinate someone else to help me with testing.

Another challenge is that we didn't actually have enough April Tags to fully test the system. At the moment, we only have 10 tags which is why we had to split up the experiment into 2 parts. We are making more Tags so we can test the system in a more complete manner.

The last technical challenge has to do with GPS drift. The GPS signal drifts over time and the important part is that the noise from the GPS is not independent of time. That is, we cannot assume that over a period of 1 minute, the mean of the GPS coordinates that we read will be the true GPS coordinate. This is problematic for two reasons. First, it makes the low pass filter that we use for the GPS coordinates useless for these long-term drifts since they are not high frequency changes. Second, as discussed above, the drone uses the GPS coordinates of the Tags to prune nodes. However, because the live GPS signal drifts, it is possible for the drone to become desynchronized with the GPS tags that are stored in memory. This can potentially cause the drone to prune a node when it has not actually passed over it. A solution to this problem is to update all the GPS coordinates each time by cascading and propagating through the graph. However, our graph structure itself has a lot of noise since it is constructed via relative transforms. Unfortunately, we do not think there is a sufficient and complete solution at this time. We may have to organize the April Tags far enough such that this situation doesn't happen.

3 Teamwork

Rahul, Pulkit, Pratibha and Danendra worked on developing costmaps and integrating the Husky navigation package into the software stack. They have been testing the Husky outdoors to try and integrate the GPS data into the Kalman filter but they have encountered some issues. First their settings kept getting saved over from the previous run and then they have experienced a lot of drift when integrating the GPS data. Instead, now they are simply testing the Husky indoors with imu and odometry. Rahul and Pratibha specifically have been in charge of testing the Husky outdoors. They have spent a couple of days trying to get the GPS integration working.

I did not work with my teammates for this PR since they are all working on the Husky and I am the only member developing on the Bebop 2.

4 Future plans

Progress on the drone is mostly complete - all that is really left is system integration. Thus, most of our work will be focused on the Husky. Since we do not think that all 5 members can work on a single part of a single robot, I will be looking at some upgrades for our system. Specifically, I will procure more April Tags for the drone since the current ones that we have are insufficient in quantity and quality. This will necessitate testing to ensure that their reflective properties are usable. In addition, I will begin the system integration part for the Drone. This requires several components. First, it needs to record the GPS location of the Husky. Second, using this location, it needs to find the path that will guide the Husky to the goal. This is different from the similar functionality on the drone since the Husky is limited by the edges in the graph - it cannot simply fly to a vertex like the Bebop 2 can. Some template code has been developed for the Bebop 2 but it will need to be further tested outdoors. Lastly, all the code needs to be ported to the Husky PC to run on the same ROS network. This may introduce CPU bottlenecks - we will need to test.

Pulkit and Rahul will continue to work oon the obstacle detection and avoidance aspect of the Husky. Prior testing indicates that this functionality is not reliable - they will optimize the parameters to make it more robust. Danendra and Pratibha will be testing the Husky outdoors to integrate the GPS signal into the navigation package. We need to use the GPS signal so this part is high priority.