Individual Lab Report 11

Progress Review 12

Pulkit Goyal

April 17, 2018

Team F - Falcon Eye
Pulkit Goyal
Pratibha Tripathi
Yuchi Wang
Rahul Ramkrishnan
Danendra singh

Individual Progress

I mainly worked towards finalizing the system as a whole. I added the GPS to the second EKF running in the system. I also tried phone's IMU and GPS in order to narrow down the problem. Danny, Pratibha and Rahul, contributed to different parts of this testing.

EKF1 that takes care of fusing the continuous data was working fine (as shown in Fig 1) but the problem is with fusing the GPS data which leads to discrete jumps in EKF2 values. I already checked the GPS values by plotting it separately but the error was within 3 meters.

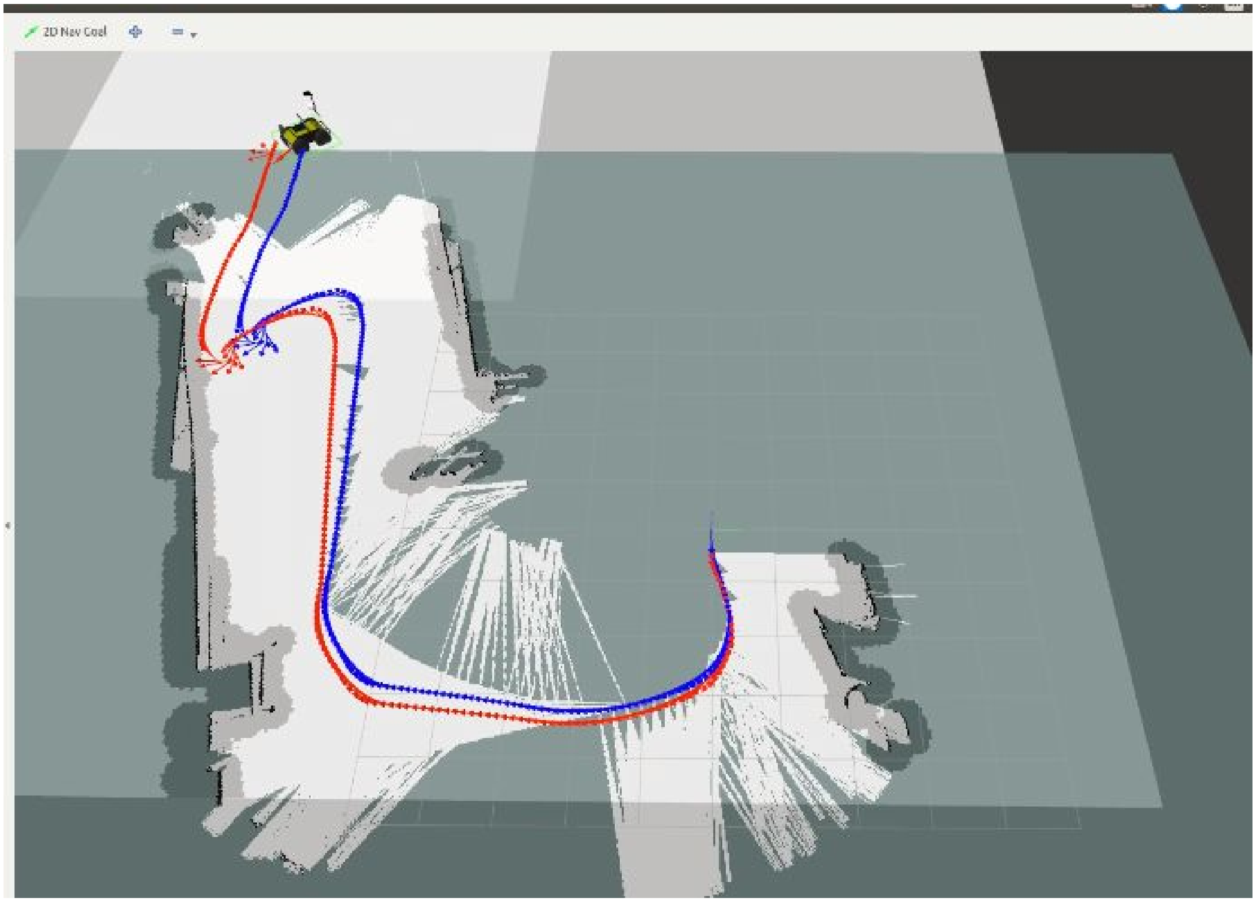

Fig 1: Blue is EKF output and Red Odometry data from wheel

There was definitely drift in the magnetometer values we were getting from IMU, affecting the Yaw, Pitch and Roll which is used by Navsat transform to convert the raw gps data to latitude and longitude values. I tried using the phone's IMU and GPS but the IMU values from phones were also drifiting a lot and GPS was not helping a lot.

The things which actually helped to solve the GPS fusion problem in EKF were:

1. Calibrating the IMU magnetometer properly. It was not working in the best of it's condition but reducing the drift in the yaw values we were getting out of IMu helped us solve the problem.

2. Visualizing the data into the odom frame instead of world frame helped me visualize the EKF values properly. The drift is there in odom frame too but it's not discrete jumps rather a smooth drift which is not much of a problem in our case. As shown in Fig.2, we took a long route using the autonomy on the Husky and the path taken by EKF is quite stable and drift-free.
3. Using this fused EKF I gave it GPS values to reach on to. It was pretty bad, that's when we realized that the GPS values we are getting are supposed to be averaged out before feeding it as a goal in navsat. This solved the problem. Robot is able to reach the given gps point's location with an error of around 3 meters.

Fig2. As can be seen, the gps values(black arrows) are quite continuous and approximately correct when the robot is moving

Right now the stack is written such that multiple values can be entered into a text file and the navigation code will read that file, reading all the given GPS locations and then executing path avoiding obstacles on them one by one as a final goal. To test the accuracy we entered the 4 different gps values, repeatedly 5 times in the file in order to test the repeatability of the test. Entered 4 GPS values created a rectangle and the robot retraced the rectangle avoiding the dynamic obstacles in order to reach the gps values one by one.

We only got two days of sunshine to test the GPS fusion with the EKF. We got many visitors(as shown in Fig 3) to see the robot's autonomous navigation.

Fig 3: Testing the robot in front of cute audience

Challenges

1. Weather was a very big challenge for all these developments as development was required to be done on GPS now. It was difficult to get GPS fix in such random weather. If I could go back in time and opt for a problem statement that didn't involve much of outdoor testing.
2. Course work definitely
3. It was quite difficult to narrow down the issue, working with so many hardware.
4. FInding a common time to work out on the Project.

Teamwork

1. Testing husky took a lot of effort and time, I, Rahul, Pratibha and Danendra were involved with mostly testing out the Husky and fuse the GPS values in EKF.
2. Yuchi worked on increasing the drone's height and increase the number of april tags detected. He also tried increasing the april tag detection speed.
3. Danendra and Rahul, intermittently helped Yuchi in the testing of the drone.

Future Plans

We specifically plan to work on the following tasks:

1. Entire team will work on total system integration, Rahul and Pratibha will help me on the Husky side whereas Danny will help Yuchi send out the GPS values to Husky. Integration should have the following of the GPS waypoints given by Bebop. We need to figure out the sequence of steps for waypoint following. We need to decide the

architecture for the multiple points given by drone. There are still some points for the integration:

a. Communication via common text file / subscriber-publisher method
b. Signal from Husky to Drone, when the gps point is reached asking for the next one
c. Drone sending the last GPS value of path to husky.

2. A lot of system integration testing is required. The days with good weather are few and GPS has important role to play in entire project.