William Seto

Team B: Auto Pirates

Teammates: Bikramjot Hanzra, Shiyu Dong, Tae-Hyung Kim, Tushar Chugh

ILR10

March 31, 2015

Individual Progress

For this progress review, I focused on helping my teammates integrate the new functions and then analyzing the results of the most recent field test.

After my teammates worked to develop the functions for inflating the obstacle costs and modifying the costmap to bias the path planner to generate paths to the right side of the “middle line”, I helped to resolve minor bugs that came up. For example, after modifying the code for the cost inflation, it broke some previous functionality that we had for introducing “fake obstacles” in the river. So I had to do some additional refactoring before we restored the functionality.

Although things ran fairly well during the field test, there were a few strange occurrences. One thing which we saw again was that some messages were either dropped or not publishing at all for a short period of time. Previously, we thought it was due to the onboard computer not being able to handle the large volume of messages we were putting through the network. This is still a possibility but we noticed that it only happens with the Marker messages. Unfortunately, we didn’t record more of other topics to infer if it was a problem with the software simply running too slow or if there is some bug that is causing the nodes to freeze. We will figure out a more thorough way to observe this in the next test. For now, it’s not a major issue and we can simply restart the software.

Also, I analyzed the paths we were taking during the test. For the most part, the boat was able to follow them well. However, even with our current cost inflation, it’s still not good enough in terms of producing some predictable behavior. One precarious situation is that when we are approaching the obstacle, the path planner may plan to go on one side of it in one iteration, but then may plan a path on the other side in the next iteration. This is shown in Figure 1. The problem is that the boat will waver left and right and then end up on a collision course with the obstacle until the last possible second. We were somehow luckily able to steer clear regardless.

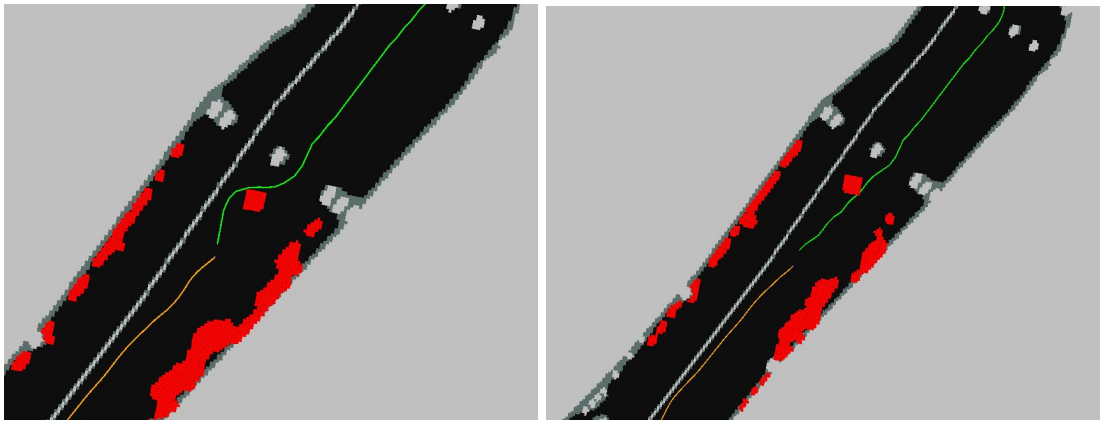

Figure 1: Path planner can’t make up its mind

Challenges

As it always is, integration ended up taking a while and we had limited time to prepare for the field test. Although the functions for obstacle inflation and right sided driving were fairly quick to implement, there were several minor issues as mentioned before that took a couple of hours to fix. The problem is that our functionality is growing and that we need to test multiple things after introducing some changes. In this case, we had to test the fake obstacle functionality and then the simulator functionality. The good thing is that testing those parts allowed us to find the bugs before the field test.

One challenge which we observed during the field test which we may not be able to do anything about is the presence of noise in the radar data. Although our current filtering is helping, there are certain bridges on the river that induce a large amount of noise. This sometimes leads to large sections being marked as obstacles, causing the boat to take a very strange path. The tradeoff is that we shouldn't filter more aggressively since the actual obstacles sometimes have a smaller signature than the noise itself, so we're kind of just working based off of hope here.

Teamwork

Shiyu Dong

Shiyu worked editing and creating the costmaps to bias the boat to drive on the right side of the river.

Bikramjot Hanzra

Bikram implemented the functionality to inflate the obstacle costs, which then Tushar and I worked to integrate into the path planner.

Tae-Hyung Kim

Tae-hyung worked on interfacing with the Novatel INS with a program, and then logging the raw data onto ROS bag files.

Tushar Chugh

Tushar worked on integrating the inflated obstacle costs as well implementing the behaviors for using the costmaps to follow the rules of the road.

Plans

Our plans for the next field test and before the next progress review are to make incremental improvements to what we have now. For one thing, we currently have not implemented a stopping condition for the path planner; it simply loops nonstop until we kill the program. We will address this when we develop a simple GUI which can specify destinations as well as a ‘START’ and ‘STOP’ button. Furthermore, we will be looking into improved obstacle inflation to address the issue mentioned above.