Drone Simulation Report

Charles Yoo

I. Introduction and System Overview

For this project, I have used state estimation (in place of ground-truth states) to perform trajectory planning and control. To understand the implication of using state estimation, 2 models will be compared. The first model is a simple, base model will rely on a constant velocity controller. The second model is an improved version that uses minimum snap. The layout of the rest of this report is as follows: Section 2 addresses how a shift from ground-truth to estimation affects the performance of the planning and control. Section 3 shows the performance of the code during the flight simulations. Finally section 4 wraps up the paper by including the notable findings of this project and proposing areas for potential future work.

II. Ground-Truth vs. State Estimation

Using estimated states in place of ground-truth values introduces noise to the controller, which affect model performance. A constant velocity controller is not robust enough to deal with the noise and had poor performance regardless of the gains that were used. On the other hand, a controller that uses minimum snap is much better at dealing with the noise and has proven to be much more effective. Additionally, the orientation gains used for the base model were too aggressive for the improved model and had to be reduced by 70%. On the other hand, the position gains were tuned well and were not much of a problem for the improved model.

III. Flight Simulations

For this report, a modified version of the window map is used to compare the performance of the old and new code using estimated and ground-truth states.

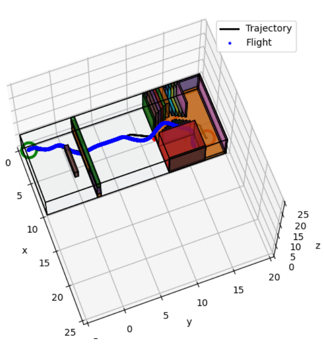

Fig. 1. Path using base code

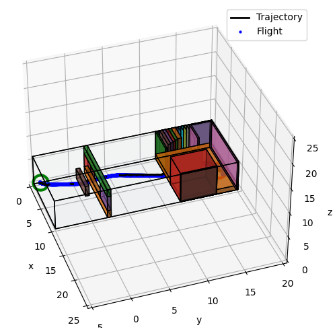

Fig. 2. Path using improved code

The figures above show the path of the quadcoptor when using estimated states. Figure 1 shows the base code that uses a constant velocity controller. Figure 2 shows the improved code that uses minimum snap. Over the long gap between the start and goal positions, state estimation error grows because of the sparsity of features in this region. The old code is unable to handle this noise properly and becomes offset from the desired trajectory in the goal region. As a result, the quadcopter collides with the walls and comes to rest at a position that is offset from the goal. On the other hand, the new code is robust enough where the noise has minimal effect on the flown path. As a result, the quadcopter is able to follow the desired trajectory and reaches the goal. Both codes were also run using ground-truth states. Since there is no noise, both codes successfully reached the goals. A summary of the results is shown in table 2.

	Ground-Truth		Estimated	
Criteria	Old	New	Old	New
No Collision	pass	pass	fail	pass
Reached Goal	pass	pass	fail	pass
Flight Time	12.5 s	10.7 s	150.0 s	9.5 s
Flight Distance	29.6 m	28.5 m	29.1 m	28.5 m
Planning Time	273.7 s	1.4 s	264.0 s	1.4 s

Table 2: Results on Altered Window Map

The results in the table show that the new code is an improvement over the old code regardless of whether the state is the ground-truth or estimated. When using minimum snap instead of constant velocity, the flight time and the distance travelled are shorter. Additionally, the streamlined A path planning that is used in this project significantly reduces the overall run time. This will likely be extremely useful for the extra credit portion where the quadcopter will need to repeatedly replan over local maps.

IV. Conclusion

The problem of state estimation and noise requires a robust controller so that the state estimation error does not become unmanageable. The constant velocity controller used previously no longer worked well with the introduction of noise. The limitations of the constant velocity controller became apparent for maps similar to the window map where there are regions between the start and end positions that have sparse features that can be picked up by the sensors. This caused the quadrotor to become offset from the desired trajectory and to fail. On the other hand, minimum snap trajectory generation did not suffer from noise issues. More testing needs to be done to further improve the performance of the controller so that the quadrotor is able to take more aggressive and faster paths without colliding with obstacles or losing control. Additionally, since the execution time has been improved, replanning the trajectory in local maps becomes feasible.