ELEC 5660: Introduction to Aerial Robotics
Project 1: Phase 2

Assigned: Mar. 5, 2019 Due: Mar. 12, 2019

1 Project Work

In phase 1, a controller is implemented and the quadrotor can track a pre-defined trajectory. Phase 2 will focus on trajectory generator. A carefully designed path planner can enable the quadrotor to operate aggressively and precisely. Your task includes:

1.1 Trajectory Generator

A natural way to command the quadrotor is to set waypoints that it needs to pass by. What the path planner needs to do is to generate a trajectory that

1. connects all waypoints (including start and goal points).
2. meets smoothness criterion.

Two sets of waypoints are provided in `test_trajectory.m`. You need to design two more sets of waypoints (with at least 6 waypoints). You can choose either 5^{th} order polynomial trajectory or minimum snap trajectory [1]. The later one is our recommendation and will win you bonus points.

2 Structure of Simulator

The simulation code is almost the same but a `trajectory_generator.m`. See comments for details.

3 Tutorial

You can use the naive trajectory generation method in your lecture(only smoothness and connection of waypoints is required), or you can try the optimization-based method(We encourage you generate the trajectory using this method). If you prefer the latter one, you have two ways to implement it.

1. You can use the method in the slides of Lecture 4 to map the original constrained quadratic program (QP) to an unconstrained QP, and then obtain the closed form solution of the unconstrained QP directly.
2. You can use "quadprog" function in Matlab to solve the constrained QP. This function is originated in Matlab, and you can type in "help quadprog" in the command window for more detail.

4 Submission

When you finish the assignment you may submit your code and documents on **canvas** before **Mar. 12, 2019 23:59:59**. The project name for this assignment is titled "`proj1phase2`".

Your submission should contain:

1. A **maximum 2-page** document including:
   (a) Figures plotted by simulator.
   (b) Statistics about your controller. (For example, RMS error between current state and desired state for position, velocity).
   (c) Analysis of your result. (For example, parameter studies).
   (d) Any other things we should be aware of.
2. Files `controller.m`, `trajectory_generator.m`, as well as any other Matlab files you need to run your code.

You will be graded on successful completion of the code and how quickly and accurately your quadrotor follows the generated path. This time we will also test one other set of waypoints which will not be released.

References

[1] D. Mellinger and V. Kumar, "Minimum snap trajectory generation and control for quadrotors," in *Proc. of the IEEE Int. Conf. on Robotics and Automation*, Shanghai, China, May 2011.