AE483: Lab #3
Flight Control

T. Bretl

October 15, 2017

1 Goal

Your primary goal in the next three weeks will be to design, implement, and test (in simulation and in experiment) a controller that makes a quadrotor hover, both at a fixed position and at a position that changes as a function of time.

2 First Week

Your objective this week is to design, implement, and test a controller in simulation.

2.1 Control to Hover at Constant Position with Zero Yaw

2.1.1 Simulate

Download the MATLAB script `lab3_simulate.m`. This script provides a template for control design and implementation in simulation. The structure of this template should look familiar to you — it uses exactly the same coding patterns that have been suggested in class and on the homework. Please use this template to do the following things:

- Design a controller to keep the quadrotor at hover with zero yaw, at some desired position.
- Simulate the application of this controller if the quadrotor starts at some other position.

Note that you are responsible for ensuring that the input u can be realized by spin rates $\sigma_1, \ldots, \sigma_4$ that are within bounds.

2.1.2 Visualize

Download the MATLAB script `lab3_visualize.m` (as well as the data files `quadmodel.mat` and `mocapmodel.mat`). This script provides a template to visualize the results of simulation. In particular, the outputs from `lab3_simulate.m` are the inputs to `lab3_visualize.m`. Again, you should recognize the structure of this template — it is similar to what you saw in Lab #1. Please use this template to do the following thing:

- Create a movie in `.mp4` format showing the results of simulation (i.e., of applying your controller if the quadrotor starts at some position other than the desired position).

Being able to visualize the results of simulation may also help you to debug your simulation code.

2.1.3 Test

Please answer these questions:

- Does the quadrotor always achieve hover with zero yaw at the desired position? If not, can you characterize the initial conditions for which this does and does not happen?
- Are you satisfied with the performance of your controller in simulation? Can you quantify how "good" your controller is?
- If you are using discrete-time LQR for control design, what is the impact of changing the diagonal entries in the weighting matrices Q and R? In particular, how do these changes affect your answers to the first two questions (if at all)?

You are encouraged to discuss your answers both with colleagues in other groups and with your TA.

2.2 Control to Hover at Changing Position with Zero Yaw

You have likely noticed that the model produced by linearization about hover at zero yaw is always the same, regardless of the equilibrium position. What this means is that your control design should (in theory) work just as well for any desired position, not just the one you chose in Section 2.1. In fact, you should find that the only change required to make your controller track a changing position is to change the equilibrium state as a function of time. The `lab_simulate.m` code suggests doing this by computing the current desired position `odesi` as a function of the current time `ti`, and then by replacing the first three elements of the equilibrium state `xe` with the desired position `odesi`. Please do the following things:

- Choose a function of time for the desired position (e.g., a circle, a figure eight, etc.). This function of time is often called a *trajectory*. You may want to start with a simple trajectory, then get creative. The sky is the limit.
- Create a movie in `.mp4` format showing the results of simulation when your controller is applied to track the trajectory you chose.
- Reconsider the questions in Section 2.1.3. Are your answers still the same?

In both this and the previous section, you are encouraged to do everything you can to try to "break" your controller (e.g., starting from weird initial conditions, choosing an aggressive trajectory to follow, etc.). The more it breaks in simulation, the less it breaks in experiment (hopefully).

2.3 In-Lab Deliverables

Confirm that you have done the following things in lab on the first day:

1. Show the TA your movie of control at a constant position.
2. Show the TA your movie of control at a changing position.
3. Discuss with the TA your answers to the questions posed in Section 2.1.3, both for control at a constant position and for control at a changing position.

Please come back to the lab during office hours to do anything you did not finish. You will implement and test exactly this same controller in experiment with the real quadrotor next week.

3 Second Week

Your objective this week is to implement and test your controller in experiment, for hover at a fixed position with zero yaw.

3.1 Implement

Download and import `AscTec_SDK_v3.0_Lab3.zip` to eclipse. Open `lab.c` and modify `lab3()` to implement your controller, which will likely have the standard form

$$u = u_e - K(x - x_e).$$

Comments in `lab.c` provide some guidance in translating your MATLAB implementation to a C implementation. You should also modify the parameter values at the top of `lab.c` (both physical parameters like m and controller parameters like K) so that these values are consistent with what you used in simulation. You need not worry this week about implementing the method that converts commanded inputs (i.e., choices of u) to spin rates. The method that is already implemented in C code should be equivalent to what you implemented in simulation.

3.2 Test

3.2.1 Setup

Grab a hummingbird vehicle, its matching (color-coded) RC transmitter, and the associated (numbered) XBee radio. Flash the vehicle using the procedure from Lab 1. Connect the XBee radio using a USB-to-mini-USB cable. Download the ground station code from the course website and open the solution file with Visual Studio 2010. Change the IP address and location where data files are saved in `main.c`, using the instructions in Lab 1. Find desired position in `planner.c`.

3.2.2 Flight

Place the vehicle on the ground, turn it on, and arm the motors. Run the ground station. Flip the RC transmitter switch up to begin the flight. Flip the switch back down to end the flight — be sure to have someone ready to catch the vehicle as the motors return to idle at the end of the flight! (Ask a TA to do this on your first flight.) Verify that data were collected and store these data for future use. Please be sure also to record video of your flight (e.g., with a smartphone).

3.2.3 Analysis

Plot both position and desired position as a function of time. Consider these questions:

- How closely does the position converge to the desired value?
- What change to your controller (in particular, to u_e and/or to K) would make the position converge more closely to the desired value?

You may wish to focus, in particular, on errors in altitude. You will likely find that the vehicle's altitude is not at all what it should be. (In fact, you may find that your quadrotor does not leave the ground at all.) Why is this? Try to improve your results by changing the parameters in your controller (u_e and/or K) and repeating your experiment.

3.3 In-Lab Deliverables

Confirm that you have done the following things in lab on the second day:

1. Show the TA your plot of position and desired position as a function of time.
2. Discuss with the TA your answers to the questions posed in Section 3.2.3.
3. Show the TA your plot of position and desired position as a function of time, after you have made at least one change to your controller that improves your results.

Please come back to the lab during office hours to do anything you did not finish.

4 Third Week

Your objective this week is to implement and test your controller in experiment, for hover at a *changing* position with zero yaw. Please proceed exactly as you did last week (Section 3), modifying `planner.c` in the ground station code so that the desired position is a function of time—the same function of time that you tested in simulation (Section 2). As usual, please verify that data were collected and that video was recorded (e.g., with a smartphone) after each flight. Consider these questions:

- How closely does the actual position match the desired position?
- How closely do the results in experiment match the results in simulation?
- What change to your controller would make the actual position in experiment *either* match the desired position more closely *or* match the results in simulation more closely?
- Are you satisfied with the performance of your controller in experiment? Can you quantify how "good" your controller is?

You are encouraged to modify your controller to improve your results in whatever way you can. Think carefully about why you consider one set of results better than another—really focus on the question of what you mean by "good" performance.

4.1 In-Lab Deliverables

Confirm that you have done the following things in lab on the third day:

1. Show the TA your plot of position and desired position as a function of time, both in simulation and in experiment, from at least one flight (in which the desired position changes as a function of time).
2. Discuss with the TA your answers to the questions posed above.

Please come back to the lab during office hours to do anything you did not finish.

5 Report

Write a report with four sections:

- goal
- method of approach
- results
- discussion

In particular, you must include a description of your approach to control design and implementation. You must include results from both simulation and experiment, both for a desired position that is constant and for a desired position that changes as a function of time. You must include a discussion of these four questions: (1) how closely does the actual position match the desired position in each case? (2) how closely do the results in experiment match the results in simulation, and how do you explain any differences? (3) how "good" is your controller? (4) what changes did you make to improve the performance of your controller, and how well did these changes work? Clear explanations of problems or crashes that occurred in lab are also always welcome.

You are encouraged to go beyond these requirements. Your report must be submitted as a PDF with single-space text no larger than 12-point font and with 1-inch margins. It must be a minimum of six pages and a maximum of eight pages. You are strongly encouraged, but not required, to use LaTeX to prepare your report. You must submit your report no later than 11:59PM on Friday, November 10. Submission details will be posted to piazza.