Assignment 8

Clay Whipp
ESE 448
Washington University in St. Louis

4/23/20

There were two parts to this assignment. In the first, I estimated four sets of coefficients, b and c, corresponding the four systems: throttle with states X and w, elevator with states θ and q, aileron with states ϕ and p, and rudder with states ψ and r. I used bang bang data to compute these estimates.

In the second part of the assignment, I found a better estimate of the transfer function for the motors using chirp data.

I used the same method to find all four, so I will use the b and c corresponding to throttle as the example to illustrate how I found all four sets of coefficients.

1 Solving for b and c

First I used the assumption that the state-space representation for Z and w was as follows,

$$\begin{bmatrix} \dot{z} \\ \dot{w} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & b \end{bmatrix} \begin{bmatrix} z \\ w \end{bmatrix} + \begin{bmatrix} 0 \\ c \end{bmatrix} T \tag{1}$$

Since b and c only appear in the second row/equation, I could discard the first row.

$$\dot{w} = \begin{bmatrix} b \end{bmatrix} w + \begin{bmatrix} c \end{bmatrix} T \tag{2}$$

I rewrote this as

$$\dot{w} = \begin{bmatrix} w & T \end{bmatrix} \begin{bmatrix} b \\ c \end{bmatrix}$$

I used the forwards euler discrete form to rewrite $\dot{w}$, $\dot{w} = \frac{w_{i+1} - w_i}{\Delta T}$. I then solve for b and c, which is shown below in Equation 1.

$$\begin{bmatrix} b \\ c \end{bmatrix} = \begin{bmatrix} w_i & T_i \end{bmatrix}^{-1} \left(\frac{w_{i+1} - w_i}{\Delta T}\right) \tag{3}$$

For ΔT I used $\Delta T = 0.005$. For w and T I used the bang bang data we were given. To get T_i, I took the motor speed data, multiplied by a diagonal matrix to make all the motor speeds positive (originally n_2 and n_4 were negative). I then used the motor mixing matrix to convert from motor speeds to TEAR.

From TEAR I got T and corrected it by subtracting $T_{trim} = 303$Hz, since our linear system was linearized about trim. I found T_{trim} initially to be around 298 from the T data. Once we found b and c, we tweaked T_{trim} by plotting the actual w data vs the w found using our b and c values. We tweaked the value until the actual w and our simulated w matched.

I corrected w, by using the detrend function in Matlab. This made it so the w data started out closer to 0 which is what we would expect at the start when the drone is in a hover.

I then determined the delay between inputting a throttle command and seeing an effect on the velocity, w. To do this I plotted w and T with sample points as the x-axis. From this plot I found the delay to be 9 sample points, so I subtracted this from the w to make it match up with T. The plot I used is shown below.

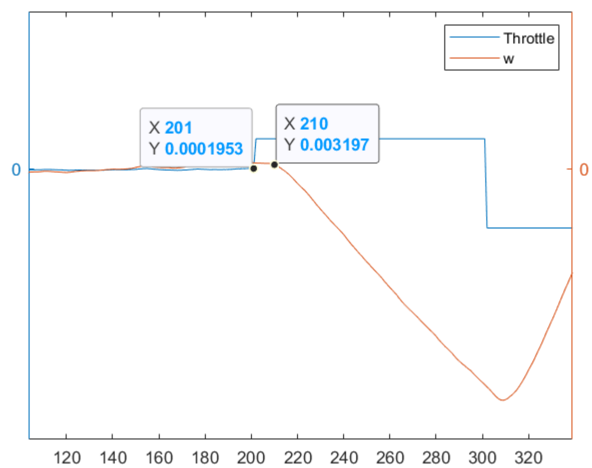

Figure 1: Plot used to find delay between T and w

I then restricted the data set to the points where the bang bang command was applied (200 to 400).

Now that I had properly cleaned up my data, I plugged it in to Equation 2. I solved for $b_T = 0.4972$ and $c_T = -0.0411$. Remember that these are the parameters associated with throttle which is why I denoted them "b_T" and "c_T". Using the same method, I solved for the other bs and cs. The only difference was I didn't correct E, A, or R like I did with T and T_{trim}. This is because there is no force acting in the direction of E, A, or R to cause it to be nonzero when the drone is stationary like there is with gravity and T. I found the other bc's to be: $b_E = -0.2834$, $c_E = 0.5097$, $b_A = 200.3453$, $c_A = 428.9286$, $b_R = 2.1345$,

$c_R = 2.8138$. The bc pairs for T, E, and R are reasonable for the ones for R seem way too high. I checked my code and it appears to be correct, so I am not sure what is causing this.

We wanted to check how accurate our b_T and c_T values were, so we used the Matlab function to generate values of w given a state space representation found using our b_T and c_T values. Since we set it to 0, $b_T = 0$. To find c I just did c = pinv(T)*w. I found c to be $c_T = -0.0013$. I plugged this b and c into the lsim and plotted this lsim w versus the actual w data. The plot is shown below.

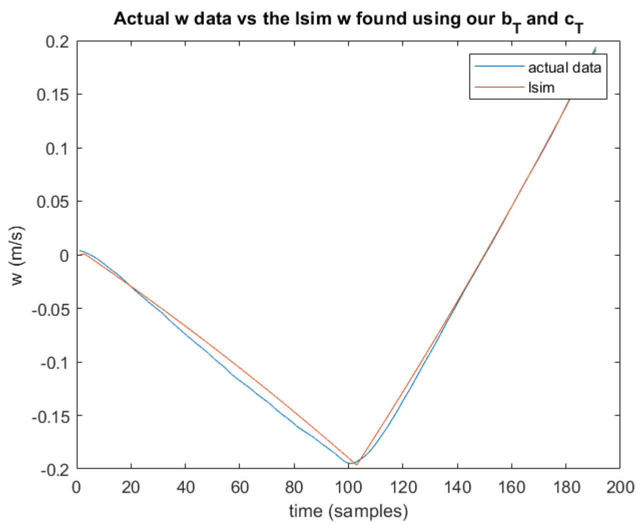

Figure 2: Lsim w and actual w over time samples

Professor Bhan suggested we could set $b_T = 0$ and just solve for c directly using a_z and T. We did this and then generated the lsim versus actual plot again. The plot is shown below.

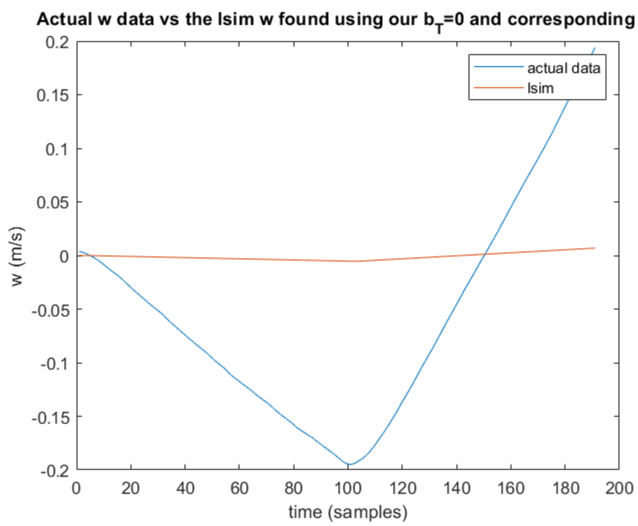

Figure 3: Lsim w and actual w over time samples with different b and c

This time the lsim w does not fit the actual w whatsoever, so I decided to go with the first b and c values I mentioned.

2 Finding the motor's transfer function

To find the motor's transfer function, I used the same equation, Equation 1, I used in Part 1 of the assignment. Here is that equation again in case you don't want to scroll up.

$$\dot{w} = b * w + c * T$$

This time I used the chirp data with an amplitude of 50. We were given chirp data with a variety of amplitudes, and I chose the 50 dataset because that's what Chris chose.

This time I let $= a_z$, so I could use a_z data from sensor data. I then solved for T. This yields the following equation.

$$T = \frac{a_z - b * w}{c}$$

My goal was to obtain the actual motor speed, n_a, and the commanded motor speed, n_c. Since T has the same magnitude as the motor speeds, I could use it as n_a. We were given the chirp signal which is the same as n_c.

I then cleaned up the data. Just like in Part 1 I detrended w. So it started at around 0. I did the same with a_z. I plugged in the data to the equation and found T_i which I used as n_c. Since I used throttle in this equation I used b_T

and c_T. I plotted n_c and n_a versus time to see how they compared. This plot is shown below.

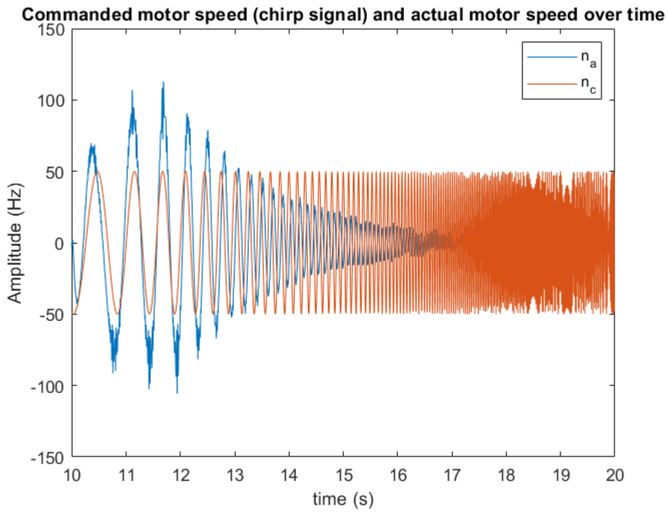

Figure 4: n_c and n_a over time

The gain peaks at around 2 at 12 seconds and decreases from there. I used the iddata and tfest Matlab functions to generate the transfer function. The transfer function is shown below. I adding in e^{45-e3} in the numerator to account for the delay, which we found to be 9 samples in the first part of the lab. Since the time step is 5ms, then 9 samples is 45ms or 45e-3s.

$$\frac{27.89e^{45e-3}}{s+23.13}$$

I made a bode plot using this transfer function. This bode plot is shown below.

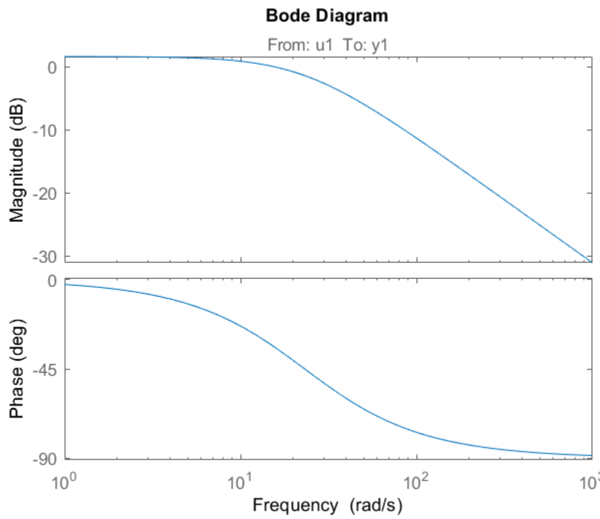

Figure 5: Bode plot of motor dynamics

We know the maximum gain is around 2. The 3db or the bandwidth is at 24Hz, which is higher than the bandwidth of 10Hz given for the actuator transfer fucntion in HW7.

3 Conclusion

This was a cool assignment. We determined the coefficients of the linear system and the actuator transfer function ourselves using bang bang and chirp data from the drone instead of just taking what Dr. Bhan gave us for granted. This filled in one more piece of the puzzle in understanding and controlling the drone.