EE CprE 491 – May 1718
MicroCART Senior Design Team
Week 8 Report

October 24 – 31
Faculty Advisors: Phillip Jones, Nicola Elia

Team Members:

Brendan Bartels — *Controls Software Key Concept Holder*
Kris Burney — *Ground Station Key Concept Holder*
Joe Bush — *Quadcopter Software Key Concept Holder*
Jake Drahos — *Team Webmaster*
Eric Middleton — *Hardware Maintainer*
Tara Mina — *Team Communications Leader*
Andy Snawerdt — *Control Systems Key Concept Holder*
David Wehr — *Team Leader*

Summary for Progress this Week

This week we finished taking our moment of inertia data from the lab in the Physics department, and we wrote MATLAB scripts to determine the moment of inertia of the quadcopter about each of its axes of rotation. Additionally, we tried to continue developing the Simulink model of the quadcopter system, and we began using our mathematical model that we developed during the process of identification of the quadcopter to try to predict the speed of the motors we characterized based on the duty cycle provided to the ESC. For the ground station sub-team, a basic CLI (command line interface) was demonstrated this week to the rest of the team, with some basic commands available to the user that will be sent to the quadcopter via Wifi, and the quadcopter will respond accordingly. These commands include setting the position of the quadcopter, like the yaw, pitch, and roll position, or getting data from the quadcopter, like its current position.

Past Week Accomplishments

- Got moment of inertia of quadcopter – Brendan, Tara, and Andy
    - Went to Physics department and returned to our data collection setup
    - Took data for the following tests:
        - ***Yaw calibration*** (setup for measuring yaw, but without the quadcopter, to get the moment of inertia of everything in the setup)
        - ***Yaw data*** (moment of inertia of the quadcopter about its yaw axis of rotation)
        - ***Pitch and roll calibration*** (setup for measuring pitch and roll, but without the quadcopter, to get the moment of inertia of everything in the setup)
        - ***Pitch data*** (moment of inertia of the quadcopter about its pitch axis of rotation)
        - ***Roll data*** (moment of inertia of the quadcopter about its roll axis of rotation)
    - For each test, took data for 15 runs:
        - 5 with a total mass of 205.1 grams
        - 5 with a total mass of 305.1 grams

        - 5 with a total mass of 105.1 grams
    - Wrote MATLAB scripts that:
        - Takes in absolute directory paths to where each of the collection of .csv data files are located for each of the tests run
        - Calculates the angular acceleration based on the angular position data and initial angular velocity data for each data run
        - Calculates the torque applied based on the mass used, which is an input that corresponds to each test run
        - Determines the moment of inertia (MOI) for each of the 15 runs per test, and averages this calculated value for all each test to determine the MOI of that test
        - Subtracts the moment of inertia of the calibration testing to get the moment of inertia of only the quadcopter, for each of the three axes of rotations
    - Found the moment of inertia about the pitch, roll, and yaw axes of rotation:
        - Pitch moment of inertia: 0.0218 kg-m^2
        - Roll moment of inertia: 0.0277 kg-m^2
        - Yaw moment of inertia: 0.0332 kg-m^2
    - Verified that these values had a reasonable order of magnitude
        - Compared these values with the values that Matt Rich got with the smaller quadcopter, which was smaller and had about ½ the moment of inertia
        - Verified that Matt Rich's values for the smaller quadcopter were a bit different between the pitch and roll measurements, like ours are
- Measured Zero-Load Current – Tara and Andy
    - Measured the current being drawn from an ESC when there is no load on it, which means that the propellor is removed
    - Control variable: the duty cycle of the PWM signal applied to the ESC
    - Measured values:
        - Current going to the ESC (A)
        - Motor speed (revolutions per minute)
    - Used least squares approximation to determine constant values that represent the zero-load current with respect to the motor speed according to the following expression:
        - $I_f = I_0 * sgn(\omega) + I_1 * \omega + I_2 * \omega^2$
    - Measured the residual error for the least squares approximation method compared with the previous approximation of I_f, which was a constant value of 0.511:
        - Least squares approximation ||e||^2 : 5.0823*10^-05 A^2
        - Constant I_f at 0.511 ||e||^2 : 0.6797 A^2
- Continued adding to Simulink model - Tara and Andy
    - Added functions that perform calculations in the "Actuation" block to determine:
        - Linear acceleration in the body frame of reference
        - Angular acceleration in the body frame of reference
    - Created a simple module that contains all quadcopter parameters and feeds them as inputs to other modules performing calculations in the Simulink model
- Made updates to website and website tools - Brendan

    - Created basic homepage and a page for the controls team
    - Created bash/stfp scripts to automate the deploy process for the website
- ESP8266 WiFi- David
    - ESP8266 gracefully handles aborted connections
        - Assumes only one connection at a time, so if a new request is received, it closes the old and uses only the new.
        - Buffers unsent data until a new connection has been made
    - Completed more extensive testing to verify the serial/TCP forwarding
- Ground station - Kris, Jake
    - Added a backend socket to the ground station
        - Multiple front-ends can now connect to the backend daemon
        - Tons of select(2)-related bugs surfaced
            - Fixed to best of knowledge
    - Began adding some environment variables
        - Disable quad
        - Select WiFi or Bluetooth
        - Change socket locations
        - Change IP address of quad (WiFi)
- WiFi Integration - David, Kris, Jake
    - Connected ESP chip to quadcopter and verified communication works
    - Updated ground station code to optionally use WiFi.
    - Successfully sent start packet, sent VRPN data, and received acknowledgement responses (responses for testing timing)
    - Did some timing and found delay to be around 40ms. Discovered that David's previous Python timing code was probably not getting the correct latency.
        - Updated the Python code, and seeing latencies around 6ms now.

Pending Issues

- Must find a way to improve our representation of motor speed from duty cycle – Tara and Andy
    - Using relevant expressions in Matt's thesis to model the motors, as described and represented above in the accomplishment called "Measured Zero-load Current"
    - Need to be able to have a very accurate representation of the speed of each of the four motors, based on the percent duty cycle we give the ESC, which is connected to the motors and drives them
    - Extremely important for when we want to control the movement of the quadcopter, because we do this by carefully changing the speed of rotation of each motor
    - Now, we have found all of the parameters relevant, but still not an accurate representation of motor speed
- Unnecessarily high WiFi latencies between ground station and quadcopter - Kris, Jake, and David
    - The latencies are higher than they need to be. There are possibly some low-level TCP things happening that is causing this

Individual Contributions

Team Member	Contribution	Weekly Hours	Total Hours
Brendan Bartels	MOI measurements, setup web dev env, software flowchart, PID documentation	10	56
Kris Burney	Ground Station Wifi support, ground station backend socket, bug squashing, latency testing	9	77
Joe Bush	Meeting, setting up tools	3.5	56
Jake Drahos	Ground station WiFi support, ground station backend socket	8	43
Eric Middleton	Ground station backend socket, Wifi support	7	76
Tara Mina	MOI measurements and calculations, Design document, zero-load current, Simulink model	13	78
Andy Snawerdt	MOI measurements, Zero-load current, Simulink model	10	79
David Wehr	WiFi code on ESP8266 chip, integration into ground station/troubleshooting	11	68

Comments and Extended Discussion

This week, we made a lot of progress in getting the mathematical model of the quadcopter system, and we are beginning to start the next phase of developing an optimal control system for the quadcopter. Now that we basically have all of the parameters of the quadcopter, we have started creating a Simulink diagram, and we will next need to start simulating position data with our current controller and running simulation tests on the Simulink model of our system. These tests need to be done with Simulink first, before we implement it in the actual software of the quadcopter, because we need to ensure that it is not unstable before we actually try using our designed controller when flying the quadcopter. If we upload a controller that makes the system unstable, we risk damaging the quadcopter.

Plans for Coming Week

- Finish Design Document 1 – All team members
    - Due this coming Sunday, November 6th
    - Template started
    - Different from the Project Plan in that we have to include test results and implementation and use these outcomes to explain and justify our design plan
- Meet with Matt to Discuss Possibilities to Better Represent Motor Speed - Tara and Andy
    - As described in the “Pending Issues” section above, we are having some issues trying to represent the motor speed with the different duty cycles of the PWM signals given to each motor’s ESC
    - Have already tried accounting for the angular velocity-dependent effects of the no-load current, as described above in the “Past Week Accomplishments” section
    - We are at a bit of a standstill with this issue at this point, so we need to discuss with Matt our next steps for trying to improve the model
- Continue with Simulink Model of the Quadcopter – Tara and Andy

    - Simulink model represents the quadcopter with the mathematical model derived in Matt Rich's thesis
    - Currently have most of the blocks in the "Actuation" high-level block, which represents the part of the system that will take in the current position of the quadcopter and will perform calculations to determine how to change the motor speed of each of the four motors in order to move the quadcopter to be closer to the position in space it is supposed to be at
    - Add the remaining functional blocks in the "Actuation" block of the control system
    - Use custom blocks, which uses MATLAB functions with input and output values that will represent a functional block, to implement some of the expressions in Matt's thesis and do calculations on the data
    - Custom blocks will make our Simulink block diagram model more simple and easy to follow, using MATLAB functions rather than trying to implement the mathematical expressions with summers, gain blocks, and other Simulink blocks
- Resolve TCP WiFi latency issues - David, Kris, Jake
- Update ESP8266 code to support UDP - David
- Add useful CLI commands for front end sockets - Kris, Jake
    - Toggle printing to back end
    - Menu command to list options
    - Quad Status (once as well as continuous)
    - etc..
- Quad software memory leaks - Joe

Summary of Weekly Advisor Meeting

This week, we updated Dr. Jones on some recent developments, including the current state of the CLI (command line interface) and its current functionalities, as well as the progress the controls team has made to arrange the data acquisition setup for getting the moment of inertia angle measurements. When discussing our next steps, Dr. Jones explained some issues that MicroCART has had in the past with tuning the PID controllers, and gave the ground station team some pointers that would help with maintaining the stability of the system: to minimize the latency of the communication between the camera system and the quadcopter. Furthermore, towards the end of the meeting, Dr. Jones made some recommendations for documentation to keep in mind for future teams, as well as for making sure our design work is simple and clear to be used by anyone who might be on future MicroCART teams easily.

- CLI (command line interface) updates
    - Dr. Jones wants screenshots showing the functionality of the current CLI
- Potential experimentation with PID controllers for team
    - Takes some experience to effectively tune PID controller
    - Tuning means determining good constants for the following parameters of each of the PID controller
        - P - proportional constant
        - I - integral constant
        - D - derivative constant
    - According to Dr. Jones, Joe Avey got really good at this in last year's team

    - However, even though Joe was very good at this, he had difficulty tuning the yaw PID controller
    - Dr. Jones's theory: the controller could not be tuned because the latency was too large, of the camera data of the position of the quadcopter to the quadcopter software
- WiFi communication updates
    - Found the current latency in the system (Bluetooth)
- Simulink model updates
    - Make sure we use custom blocks with MATLAB
    - This will help make our Simulink model more simple and clean
    - Do not want to implement the expressions in Matt's thesis with Simulink blocks, it is much easier to follow and debug if we write these as MATLAB functions
    - Need to put this Simulink model on GIT
- Software Data Flow diagram update
    - Brendan has made good progress on this
    - Almost completed, finished the "sensor_processing" section and most of the "controls_algorithm" section
    - Creating a systems-level representation of the data flow of the different modules in the quadcopter software that demonstrates how the data is being manipulated in calculations
    - Dr. Jones wants us to send this out for everyone as soon as we get this done
- Controls System Team
    - Need to add Matt Rich to our group Slack message
    - Matt wants to stay up-to-date with our progress
- Recommendation from Dr. Jones for documentation
    - Make sure we take a lot a lot of pictures and videos for each of the tests we run
    - Important for future teams to get a good sense of our team's accomplishments, how we did them, and their significance