EE 449 MS2 REPORT:
Plant Identification and Modeling

Justin Palm

Andrew Nelson

Andy Bradford

Introduction

It is the goal of this paper to address the quad rotor system dynamics. We will describe the plants input, outputs, and internal state. A state space representation will be derived and we shall show how this non-linear system is linearized around a stable hovering point. In order to accomplish this task, the parameters involved in the quad rotor system are defined and estimated. It should be noted that what our customer (Distributed Space Systems Laboratory) wants of this project is to determine how closely an existing model and simulation matches the actual quad rotor in flight. The plant in question thusly has already been modeled and we are using it to basis for this report. [1]

Plant Identification

In order to determine the dynamics of the quad rotor system, the plant must be identified. Figure 1 below shows a free-body diagram of the quad rotor.

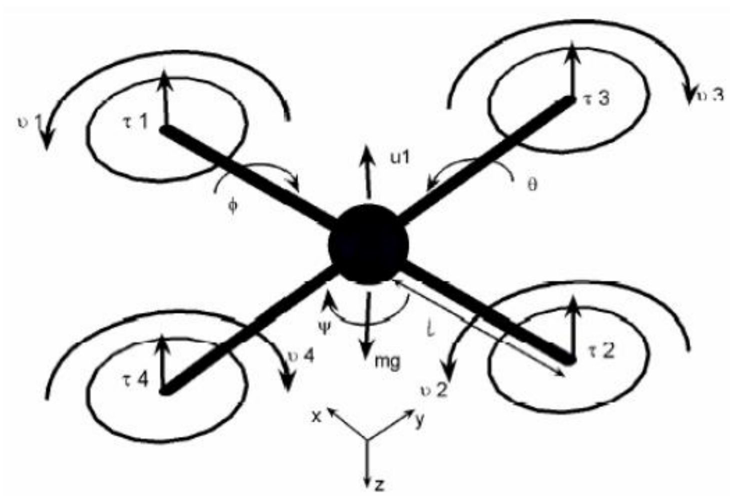
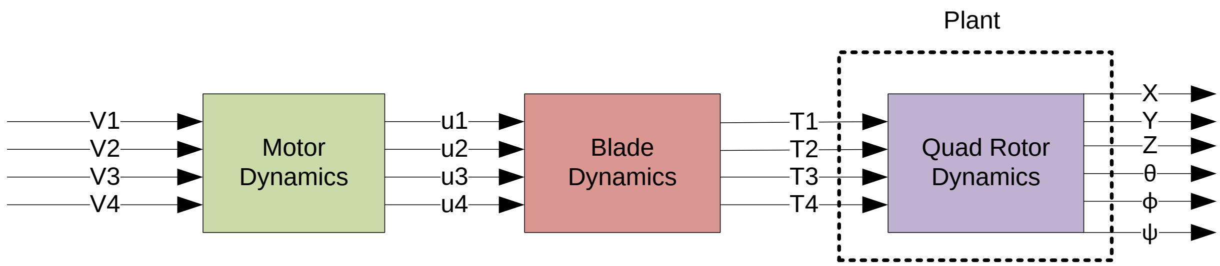


Figure 1: Free body diagram of the quad rotor

As one can see, the force of gravity must balanced by the net thrust generated by each of the four motors. As we will demonstrate later, the state of the system can be manipulated by adjusting the relative thrusts generated by the rotor blades.

Starting at a high level, we can begin to break down how the quad rotor behaves to various inputs by looking at the forces on the system. As a first pass, figure 2 shows how the plant is identified and defines the parameters listed.



Parameter	Meaning	Parameter	Meaning
V1	Voltage to motor 1	T3	Thrust of motor 3
V2	Voltage to motor 2	T4	Thrust of motor 4
V3	Voltage to motor 3	X	Position along X axis
V4	Voltage to motor 4	Y	Position along Y axis
u1	Motor 1 blade velocity	Z	Position along Z axis
u2	Motor 2 blade velocity	θ	Pitch
u3	Motor 3 blade velocity	ϕ	Roll
u4	Motor 4 blade velocity	ψ	Yaw
T1	Thrust of motor 1		
T2	Thrust of motor 2		

Figure 2: Plant identification and parameter list

Voltages control the motors which provide a rotational velocity. This velocity is then transformed into thrust through the airfoil dynamics. This thrust is applied at a distance from the center of gravity thus changing the state of the system. This methodic approach helps us identify the plant and leads us to determine what its inputs and outputs should be.

This type of system is commonly referred to as MIMO (Multiple-Inputs Multiple-Outputs). The inputs to the system then consist of four independent thrusts generated by the rotors and the outputs are positions in the X, Y, Z directions as well as an *attitude* component representing angle of attack. This makes sense since an object can be fully described by six degrees of freedom.

State Representation

In order to determine the state vector we must first define the reference frames in which we are working. Figure 3 below shows the usual inertial reference frame X,Y,Z and a body reference frame I,J,K. In this way, we can relate the attitude (yaw, pitch, and roll) to the inertial reference from. With respect to the body reference frame, there can be rotation around each of the axis (I,J,K) denoted (ϕ,θ,ψ). There is one final reference frame that must be added to this model and that is a single axis that points through the “nose” of the quad rotor and is taken to be the positive I axis.

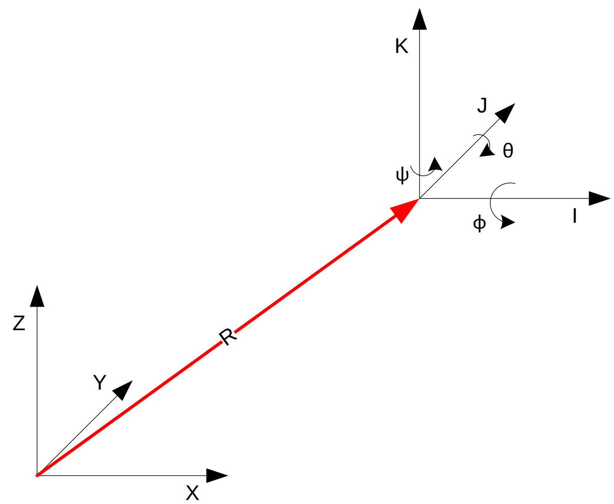


Figure 3: Reference frame translation diagram

The forces on the system will result in translational and rotational components in the body reference frame. These must be transformed back to the inertial frame to be useful to our design. The Velocity vector in the body frame is simply the spatial time derivates of position in that frame. Similarly, the rotational velocity is the spatial time derivate of angle. Euler angles relate the yaw, pitch and roll (or attitude) to the inertial frame. Lastly, the position of the vehicle with respect to the inertial frame is given as a vector in X, Y, Z.

Translational Velocity referred to body:

$$V = \begin{bmatrix} u \\ v \\ w \end{bmatrix}$$

Angular velocity referred to body:

$$\omega = \begin{bmatrix} p \\ q \\ r \end{bmatrix}$$

Euler Angles relate attitude to inertial frame:

$$\Phi = \begin{bmatrix} \varphi \\ \theta \\ \psi \end{bmatrix}$$

Position referred to inertial frame:

$$R = \begin{bmatrix} X \\ Y \\ \boldsymbol{Z} \end{bmatrix}$$

The vector equations above describe completely the state of the system. We can now define the state vector as:

$$x = \begin{bmatrix} V \\ \omega \\ \Phi \\ R \end{bmatrix}$$

The next step in defining the state space equations is to find the time derivative and try to linearize the model around a hovering position. This step has not yet been done. We will be working closely with our customer to determine this fundamental step as none of the members of our group have had much if any experience working with complex highly non-linear and unstable systems.

It was suggested (and shall be done) to take a step back and attempt to describe a state space model on a similar two dimensional *bi* rotor model. This will allow us to gain the experience and knowledge to undertake the much more complex three dimensional *quad* rotor model. We hope to gain as much insight as possible so that when we compare the actual quad rotor system to the model created by the D.S.S.L. [1].

Model Parameters

Due to the fact we are expanding and furthering the work accomplished by Heemstra [1], a system model and model parameters have already been determined. However, if we were to start from scratch, the method of developing the parameters in order to develop a system model should be understood and derived. The parameters for this project include: mass, thrust parameter, torque constant, Inertia matrix, rotor inertia, and four displacement vectors,

Mass is simply determined by placing the entire system on a scale. Next, the thrust parameter is to develop a linear relation between input voltage and angular velocity. From the angular velocity, we then can apply it to another relation that says that the square of the angular velocity times the thrust parameter gives the thrust term. This then gives the final relation for thrust versus input voltage and should have a nearly linear graph. Thrust would be validated by using a thrust stand most likely incorporating a force gauge or possibly a scale.

The next parameter is the torque constant, *k*. This important parameter is used in determining and incorporating the yaw control, because yaw is achieved through differential torque between all four motors. This constant is defined by the ratio of the power into the blade or ‘propeller’ divided by the angular velocity or rotation rate.

The moment of inertia is another important parameter needing to be visited. Because the moment of inertia could be developed a few different ways, we leaned towards the mathematical approach. It was determined that modeling the quad rotor as rods, cylinders and plates, the parallel axis theorem could be used to sum the moments of inertia due to each individual component. The rotor inertia could also be determined various ways. One being, hang it from a piece of string, measure the oscillations such as a pendulum (making sure you stay within a small angle approximation) and the other was to assume the blades as flat plates, weigh them and then determine it mathematically.

The table below shows the parameters that are used in the model being used by our customer.

Table 1: Parameters used in model used by DSSL [1]

Parameter	Symbol	Value	Units
Mass	m	0.48	kg
Thrust parameter	$\tilde{b}$	4.3248×10^{-5}	$kg \cdot m$
Torque constant	k	5.96927×10^{-8}	$N \cdot m \cdot s^2$
Inertia matrix	I	$\begin{bmatrix} 6.532 \times 10^{-3} & 0 & 0 \\ 0 & 6.6944 \times 10^{-3} & 0 \\ 0 & 0 & 1.2742 \times 10^{-2} \end{bmatrix}$	$kg \cdot m^2$
Rotor inertia	I_r	1.1998×10^{-4}	$kg \cdot m^2$
Vector to motor 1	$\mathrm{r}_{m_1/b}$	$\begin{bmatrix} 0.2319 & 0 & d_1 \end{bmatrix}^T$	m
Vector to motor 2	$\mathrm{r}_{m_2/b}$	$\begin{bmatrix} 0 & 0.2319 & d_1 \end{bmatrix}^T$	m
Vector to motor 3	$\mathrm{r}_{m_3/b}$	$\begin{bmatrix} -0.2319 & 0 & d_1 \end{bmatrix}^T$	m
Vector to motor 4	$\mathrm{r}_{m_4/b}$	$\begin{bmatrix} 0 & -0.2319 & d_1 \end{bmatrix}^T$	m

There are other aerodynamic factors that are not taken into account. One of these is ground effect which is the increased efficiency of a rotor or wing as it is within one and a half wingspan/ rotor diameters of the ground due to a ‘cushioning effect’. Translational lift is the increased lift efficiency as The controller is assumed to be robust enough to transcend these ‘disturbances’, so not much consideration is given to them.

Controllable & Observable

The concept of controllability denotes the ability to move a system around in its entire configuration space using only certain admissible manipulations [1]. If an input to a system can be found that takes every state variable from a desired initial state to a desired final state, the system is said to be controllable; otherwise the system is uncontrollable [2]. The term observability when applying to control systems means that by just observing the system outputs, it is possible to determine the behavior of the entire system. The rank is another term worth mentioning, which simply is the maximum number of linearly independent columns of a matrix. Now to determine if the quad rotor is controllable, the state matrix must have a particular property, that is, the rank of the controllability matrix must equal the number of rows in that matrix [5].

When we ran the simulation provided by our customer in Matlab, the Controllability matrix returns a rank of 12 which is also the number of independent equations (rows). We believe this implies that the system is controllable but there are some doubts. We attempted to contact our customer to resolve this issue, but at the time of this writing, they were unavailable. The rank of the observability Matrix is 12 which is equal to the number of states of the system.

Again, there are doubts about this as well. Once we are able to meet with the customer, these issues will be resolved.

Open Loop Testing

Most of the last two weeks was spent getting the quad rotor working in an open loop configuration. This serves several purposes. The first was to test Simulink's ability to interface over a serial link with the Xbee wireless controllers. The second was to evaluate the quad rotor's power and motor controller stages. Since hardware problems tend to cause large delays in a project when something goes wrong, we felt it was best to evaluate and fix any issues in these systems early on. The last reason was to get a rough idea of the vehicle's dynamics.

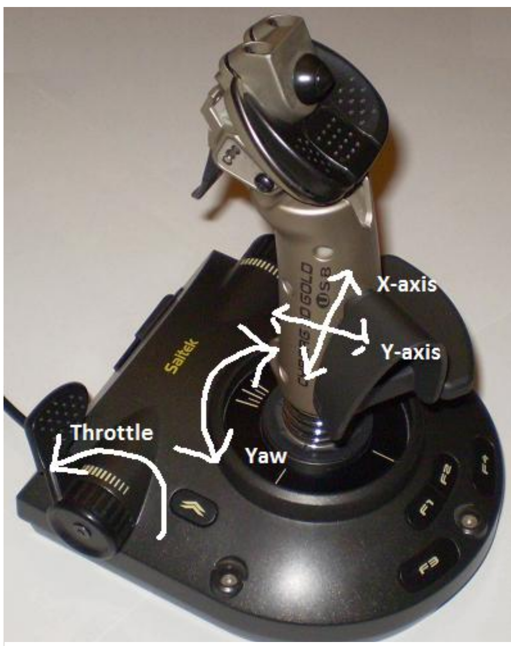


Figure 5: Joystick Control

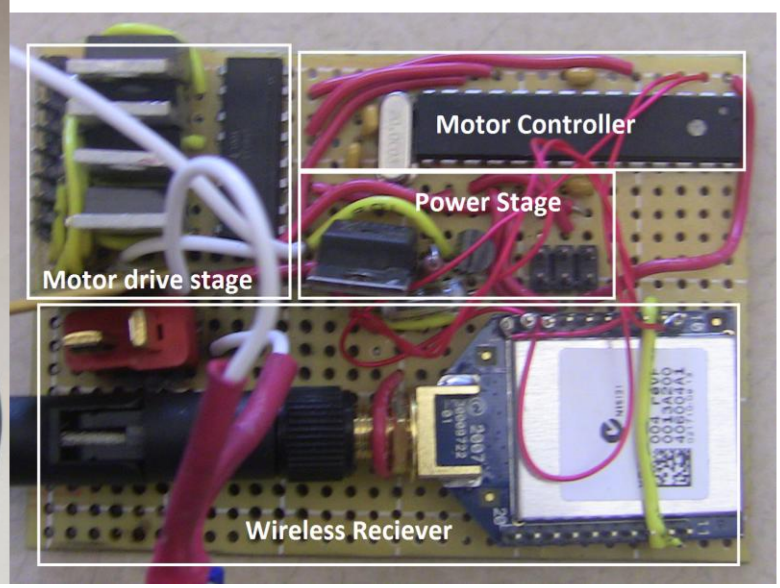


Figure 4: Quad Rotor Electronics

By using a joystick to control the motion of the quad rotor, the goal was to have something that we could start out as full open loop control and by adding feedback for various systems on the quad rotor, move to something that could give simple high level directions of motion. The control scheme we used for the joystick is shown in Figure 4.

In order to interface with the quad rotor, all the electronics had to be built from scratch. Due to time constraints being too short to try and manufacture a PCB and a breadboard being too delicate to put on a flying craft, we opted to solder the controller using protoboard. The end result is shown in Figure 5 with the various sections of the controller highlighted.

From open loop testing, unlike what the basic model had us believe, the system is very unstable and takes significant levels of control just to make it hover. With equal PWM signals sent to all four motors, there was a very strong yaw rotation even before the quad rotor began to leave the ground. This leads to believe that the motors may not be spinning at quite the same rate. Do to how unstable the quad rotor ended up being, we were not able to even achieve a hover. Another point we found is that the quad rotor seems to have a lot more thrust than the measurements in [1] show. The paper shows the quad rotor should be capable of only about 1.44lbs of thrust and with the system loaded the quad rotor currently weighs in at 1.1lbs. This conflicts with the quad rotor taking off at less than 20% duty cycle. Most likely the actual lift capacity of the quad rotor is several times larger than what the paper states.

From these early test flights we have been able to form a plan of action for building up the controller in pieces.

1. Interface the Vicom system with the Simulink model
2. Implement Yaw feedback control to make open loop flying easier
3. Implement feedback control on X and Y axis making hover only requiring throttle control
4. Implement feedback control on the throttle for stable closed loop hover.
5. Convert the joystick input to control general direction of flight while quad rotor maintains stability
6. Implement waypoint tracking where the quad rotor will fly to a given X,Y,Z coordinate and hover.

By breaking the implementation into several phases, we think we will be able to make the complex control design much simpler. With the rapid prototyping we should be able to achieve using Simulink and Real Time Workshop for the control loop, we should be able to get the quad rotor flying successfully in a relatively short amount of time.

Bibliography

[1] B. Heemstra. (2010) *Linear Quadratic Methods Applied to Quad rotor Control.* Unpublished Master's thesis. University of Washington.

[2] C. Balas. (2007) *Modeling and Linear Control of a Quadrotor*. Masters thesis. Cranfield University. https://dspace.lib.cranfield.ac.uk/bitstream/1826/2417/1/Modelling%20and%20Linear%20Control%20of%20a%20Quadrotor.pdf

[3] S. Bouabdallah, A. Noth, and R. Siegwart, “PID vs LQ control techniques applied to an indoor micro quadrotor”, 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2004.(IROS 2004). Proceedings, vol. 3, pp. 1-6.
P. Castillo, A. Dzul, and R. Lozano, “Real-Time Stabilization and Tracking of a Four-Rotor Mini Rotorcraft”, IEEE Transactions on Control Systems Technology, Vol 12, No 4, July, 2004.

[4] McKerrow, P. (2004), "Modelling the Draganflyer four rotor helicopter", *2004 IEEE International Conference on Robotics and Automation, April 2004, New*

Orleans, pp. 3596.

[5] Observability. (2010, March 29). In *Wikipedia, The Free Encyclopedia*. Retrieved 00:26, April 24, 2010, from http://en.wikipedia.org/w/index.php?title=Observability&oldid=352709914