Design and Control of an Autonomous Variable-Pitch Quadrotor Helicopter

by

Mark Johnson Cutler

B.S., Mechanical Engineering
Brigham Young University (2010)

Submitted to the Department of Aeronautics and Astronautics
in partial fulfillment of the requirements for the degree of

Master of Science in Aeronautics and Astronautics

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September 2012

Author .....
Department of Aeronautics and Astronautics
August 23, 2012

Certified by .....
Jonathan P. How
Richard C. Maclaurin Professor of Aeronautics and Astronautics
Thesis Supervisor

Accepted by .....
Eytan H. Modiano
Professor of Aeronautics and Astronautics
Chairman, Graduate Program Committee

Design and Control of an Autonomous Variable-Pitch Quadrotor Helicopter

by

Mark Johnson Cutler

Submitted to the Department of Aeronautics and Astronautics
on August 23, 2012, in partial fulfillment of the
requirements for the degree of
Master of Science in Aeronautics and Astronautics

Abstract

The aerospace community, particularly in academia, has seen a recent rise in the popularity of fixed-pitch quadrotor helicopters. The fixed-pitch quadrotor is popular largely because of its mechanical simplicity relative to other hovering aircraft. This simplicity, however, places fundamental limits on the achievable actuator bandwidth and the types of maneuvers possible to fly. This thesis explores the extent to which the addition of variable-pitch propellers to a quadrotor helicopter overcomes these limitations.

A detailed analysis of the potential benefits of variable-pitch propellers over fixed-pitch propellers for a quadrotor is presented. This analysis is supported with experimental testing to show that variable-pitch propellers, in addition to allowing for efficient generation of negative thrust, substantially increase the maximum rate of thrust change.

A nonlinear, quaternion-based control algorithm is presented for controlling the quadrotor. An accompanying trajectory generation method is detailed with an optimization routine for finding minimum-time paths through waypoints. The control law and trajectory generation algorithms are implemented in simulation and on a custom variable-pitch quadrotor. The quadrotor attitude control is performed on the vehicle using a custom autopilot. Position and attitude measurements are made with an off-board motion capture system. Several flight tests are shown with a particular emphasis on the benefits of a variable-pitch qaudrotor over a standard fixed-pitch quadrotor for performing aggressive and aerobatics maneuvers. To the best of the author's knowledge, this work marks the first documented, autonomous variable-pitch quadrotor built for agile and aggressive flight.

Thesis Supervisor: Jonathan P. How
Title: Richard C. Maclaurin Professor of Aeronautics and Astronautics

Acknowledgments

First and foremost, I'd like to thank my adviser, Professor Jonathan How. He is a great mentor and is constantly guiding, giving feedback, and helping me see the big picture. He also has an uncanny ability to immediately see solutions to new and difficult problems. I especially appreciate his love for flight and "cool hardware videos."

I would also like to thank all the members of the Aerospace Controls Lab. Most of what I have learned over the past two years has come from working and interacting with the talented students here. In particular, Buddy Michini is a master with hardware and embedded electronics and has spent countless hours tutoring and teaching me. Also, thanks to Tuna Toksoz, Kemal Ure, Josh Redding, Luke Johnson, Josiah VanderMey, Girish Chowdhary, Micheal Klinker, and Nick Kirkby for helping me with graduate school and with the various aspects of flight testing.

Finally, I want to thank my wife, Kristen, and our two, cute daughters, Hailey and Rachel. I am grateful for your distractions from work, especially when we get to play trains and eat "pom-pom" soup. You make life worth living and I am grateful for your love and support.

This thesis is based upon work supported by Aurora Flight Sciences and the National Science Foundation Graduate Research Fellowship under Grant No. 0645960. The author also acknowledges Boeing Research & Technology for support of the RAVEN [1, 2] indoor flight facility in which the flight experiments were conducted.

THIS PAGE INTENTIONALLY LEFT BLANK

Contents

List of Figures

List of Tables

THIS PAGE INTENTIONALLY LEFT BLANK

List of Notations

Chapter 1

Introduction

1.1 Motivation

Research on and development of small and micro Unmanned Aerial Vehicles (UAVs) has been increasing since the mid 1990s, primarily due to military interest and funding [3]. Among the many UAVs developed in recent years, a significant focus has been placed on rotorcraft because of their abilities to hover in place and to takeoff and land vertically. In particular, small, multi-rotor UAVs equipped with electric motors and fixed-pitch propellers have gained popularity as experimental and hobby platforms over the past 10-12 years. Fixed-pitch multi-rotor designs are mechanically simple since they are completely controllable without the complexity of the control linkages and swashplate that are inherent in traditional pod-and-boom style helicopters. Other than the motors themselves, there are no moving parts, making the vehicles mechanically robust. Also, multi-rotor designs permit the use of smaller propellers than those possible for similarly-size single-rotor helicopters, resulting in less kinetic energy being stored in the blades and thus decreasing the potential for damage should the blades come in contact with something or someone during flight.

The mechanical simplicity and robustness of fixed-pitch, multi-rotor vehicle designs, however, places fundamental limitations on the achievable flight performance of the vehicle. With fixed-pitch propellers and commonly utilized brushless motors and electronic speed controllers (ESCs), thrust can only be generated in one direc-

tion, thus preventing the multi-rotor vehicle from generating upward thrust (with respect to the vehicle body). Also, the attainable control bandwidth with fixed-pitch propellers is limited by the inertia of the motors and propellers. These limitations restrict the aggressive and aerobatic maneuvers a multi-rotor helicopter can perform, therefore limiting the future applicability of multi-rotor helicopters in agile intensive missions.

This thesis explores the addition of variable-pitch propellers to an autonomous quadrotor, a multi-rotor helicopter with four propellers. Variable-pitch propellers largely overcome the limitations resulting from fixed-pitch flight. Reversed thrust is efficiently achievable and control bandwidth is limited only by the speed of the variable-pitch actuation, not by the inertia of the motor-propeller combination. While variable-pitch propellers add some complexity to an otherwise simple and relatively robust quadrotor helicopter, the advantages of increased controller bandwidth and reverse thrust capabilities could justify the design when aggressive and agile flight is required.

This thesis also explores autonomous control of quadrotor helicopters. In particular, with a few exceptions, autonomously controlled helicopters have not been able to achieve the level of aerobatic and aggressive flight possible under the control of expert remote control (RC) pilots. Therefore, advanced control laws and trajectory generation methods are needed to take advantage of the increased hardware capabilities of the variable-pitch quadrotor and to bridge the performance gap between fully autonomous and remotely piloted helicopter flight.

1.2 Literature Review

Small, fixed-pitch quadrotors, developed primarily as toys, started gaining popularity as research platforms around 2002 [4]. Since then, small fixed-pitch quadrotors have, to a large extent, become the academic aerial research platform of choice. This is primarily due, as mentioned above, to their extreme mechanical simplicity, low cost, relative safety in the presence of humans, and ease of use indoors.

In traditional fixed-pitch quadrotors, stability and flight control are achieved by changing the voltage supplied to each of the four motors, inducing a change in the motor revolutions per minute (RPM) and, correspondingly, the thrust generated by each of the propellers. Several detailed descriptions of the modeling of quadrotors and their dynamics have been published recently [5–8]. Also, considerable work exists on various control schemes for controlling quadrotors [9–12]. Trajectory generation for quadrotors has also been considered, with several results showing time-optimal trajectory generation methods and corresponding tracking algorithms [13–15].

Controller bandwidth can be a significant problem for quadrotors, and becomes an issue for quadrotor stability as the size of the quadrotor increases [16]. Larger quadrotors require larger motors which, in turn, have larger inertias and cannot be controlled as quickly as smaller motors. Eventually, as the size increases enough, the quadrotor can no longer be stabilized through RPM control alone because the torque required to change the rotational velocity of the motor quickly exceeds the capacity of the motor. Thus, variable-pitch blades become necessary for larger quadrotors merely for stabilization purposes.

Interestingly, the first recorded construction of a quadrotor helicopter was in 1922 when Georges de Bothezat and Ivan Jerome built and flew the "Flying Octopus" [17]. De Bothezat's quadrotor was quite large (human-piloted) and utilized variable-pitch propellers for control. The quadrotor project, funded by the United State Army, was cancelled when the quadrotor proved difficult to fly and the military became more interested in autogiros. Another, more successful, full-scale quadrotor (Convertawings Model A) was built and flown by designer D. H. Kaplan in 1956 [18]. Cuts in defense spending, however, again cancelled this project shortly after initial test flights.

Several smaller, unmanned, variable-pitch quadrotors have been developed in more recent years. The HoverBot, developed by Johann Borenstein at the University of Michigan around 1992, utilized variable-pitch propellers and electric motors. The vehicle achieved promising results (no fully autonomous flights), but the project was discontinued after only three months due to funding cuts [19]. Several hobbyists have developed and are currently developing RC variable-pitch quadrotors and posting

their results on on-line forums and discussion groups (see e.g. [20]). The Vehicle Control Systems Lab at the National Cheng Kung University developed an autonomous variable-pitch quadrotor that demonstrated autonomous upright and inverted flight and flips [21], although no work appears to have been published on the project.

The past several years have also seen a significant increase in the area of autonomous aggressive, agile, and aerobatic flight, some of which has been demonstrated on fixed-pitch quadrotors. Some of the first autonomous helicopter aerobatics utilized "human-inspired" control logic where the controllers and control commands were generated after analyzing inputs from expert RC pilots flying aggressive and aerobatic maneuvers [22]. This work was later extended to a general framework where controllers were automatically generated for autonomous aerobatics given an expert RC pilot demonstration [23]. These outdoor, single-rotor, pod-and-boom style helicopters demonstrated extremely agile flight with maneuvers such as rolls, tic-tocs, hammerheads, and split-s's. However, these maneuvers required the input of expert pilots and were performed on relatively large outdoor helicopters.

Several research groups have successfully flown autonomous flips with fixed-pitch quadrotors, including triple flips with rotational rates exceeding 1600 degree/second [24]. Also, fixed-pitch quadrotors have been shown to fly through windows, perch on inverted surfaces [25], fly through moving hoops [15], juggle balls [26], hold inverted pendulums [27], and perform other aggressive, agile, and aerobatic tasks. These aerobatic and aggressive maneuvers, however, lag significantly behind the abilities of single-rotor helicopters, primarily because fixed-pitch multi-rotor helicopters lack the ability to generate negative thrust. This substantially hinders the flight envelope of the vehicle, making maneuvers such as 180 degree flips, tic-toc's, and split-s's impossible. Also, the fixed-pitch propellers make decelerating faster than gravity difficult. (While fast decelerations have been demonstrated with fixed-pitch quadrotors, they require flipping the quadrotor upside down to generate the required force. The flipping behavior is only beneficial if both the required decelerations and vertical displacements are large [14].)

The main purpose of this thesis is to explore the extent to which adding variable-

pitch propellers to a quadrotor overcomes the flight limitations of fixed-pitch quadrotors, specifically in the area of autonomous aggressive and aerobatic flight. To the best of the author's knowledge, this work marks the first documented, autonomous variable-pitch quadrotor built for agile and aggressive flight.

1.3 Contributions

The main contributions of this thesis are threefold:

1. A theoretical and empirical comparison of variable-pitch versus fixed-pitch propellers for quadrotors is performed.

2. A robust controller and trajectory generation algorithm for a variable-pitch quadrotor are developed.

3. Flight experiments are conducted on a custom variable-pitch quadrotor that verify the analytical findings.

Much of the material presented in this thesis comes from previously published work [28–30].

Chapter 2 gives an analysis of the dynamic differences in thrust output between a fixed-pitch and variable-pitch propeller. This analysis is validated with both simulation and experimental results. It is shown that variable-pitch actuation has significant advantages over conventional fixed-pitch configurations, including increased thrust rate of change, decreased control saturation, and the ability to quickly and efficiently reverse thrust. These advantages result in improved quadrotor tracking of linear and angular acceleration command inputs in both simulation and hardware testing.

Control and trajectory generation algorithms for a quadrotor helicopter are presented in Chapter 3. The control law is not based on near-hover assumptions, allowing for large attitude deviations from hover. The trajectory generation algorithm fits a time-parametrized polynomial through any number of waypoints in $\mathbb{R}^3$, with a closed-form solution if the corresponding waypoint arrival times are known a priori. When

time is not specified, an algorithm for finding minimum-time paths subject to hardware actuator saturation limitations is presented. Attitude-specific constraints are easily embedded in the polynomial path formulation, allowing for aerobatic maneuvers to be performed using a single controller and trajectory generation algorithm.

Chapter 4 details the design and construction of a custom variable-pitch quadrotor built at the Aerospace Controls Lab (ACL). The vehicle is controlled using a custom autopilot. An overview of the software infrastructure is also given.

Experimental results on the variable-pitch quadrotor demonstrating the control design and trajectory generation method are given in Chapter 5. These results validate the predicted performance increase of utilizing variable-pitch propellers over fixed-pitch propellers.

Finally, some concluding remarks and suggestions for future work are presented in Chapter 6.

Chapter 2

Actuator Comparison

As discussed in Chapter 1, quadrotor helicopters are used almost specifically because they are mechanically simple. Therefore, the addition of variable-pitch propellers to a simple quadrotor, in a certain sense, undermines the original purpose of the vehicle, and so the increased hardware complexity must be fully justified. In this chapter the potential benefits of adding variable-pitch actuation to a quadrotor are explored, both theoretically and experimentally. The variable-pitch propellers are shown to increase actuator bandwidth and allow more aggressive flight paths, at the obvious expense of somewhat more complicated hardware. The hardware trade-offs and complications are discussed in Chapter 4.

This chapter is organized as follows. Section 2.1 details the aerodynamic effects of adding a variable-pitch propeller to a quadrotor. These analytic results are verified by motor hardware testing in Section 2.2. Simulations of the variable-pitch quadrotor compared with a fixed-pitch quadrotor are presented in Section 2.3 followed by experimental results in Section 2.4. Finally, Section 2.5 summarizes the results and advantages of variable-pitch actuation on a quadrotor.

2.1 Thrust Actuation

With fixed-pitch propellers, given a constant motor rotational rate, the thrust produced by the propeller is constant (assuming the quadrotor is near hover). The only

way to change the thrust produced by the propeller is by changing the voltage to the motor, thereby inducing a change in the rotational rate of the propeller.

Adding variable-pitch propellers to the quadrotor platform results in an additional degree of freedom for varying the thrust produced by each motor-propeller combination. With variable-pitch propellers, thrust can be changed by either changing the blade pitch or by changing the rotational rate of the motors. These two actuators, to a large extent, overlap. For instance, with variable-pitch propellers a quadrotor can hover by spinning the propeller quickly and with a low blade pitch or by slowing the rotational rate of the motor and increasing the blade pitch. There are many combinations of motor speed and propeller pitch that yield identical thrust values. The number of possible combinations are only limited by the maximum propeller pitch (physical and aerodynamic limitations), the maximum available motor power, and the available discretization in the hardware motor and pitch commands.

For the purposes of this analysis, the output of the quadrotor control algorithm is assumed to be a desired thrust for each of the four motor/propeller combinations. (The details of the control algorithms will be presented in Chapter 3.) This section addresses the benefits of adding variable-pitch control to a quadrotor while addressing the allocation problem of which actuator to use, pitch or motor speed, when a given thrust value is desired.

2.1.1 Propeller Model

Analytical analysis of the variable-pitch actuation requires a detailed study of the propeller aerodynamics. The propellers used on the variable-pitch quadrotor are symmetric, tapered, nine inch diameter blades and are shown in Figure 2-1. To determine the lift and drag coefficients of the propellers to use in the analysis, an airfoil, operating Reynolds number, and Mach number must be selected. The maximum thickness to chord ratio is 0.899, and so a NACA 0009 airfoil is chosen to model the propeller airfoil. Assuming the blades are rotating at 8000 RPM, the Mach number near the end of the blades is about 0.25. This gives an operating Reynolds number of around 100,000.

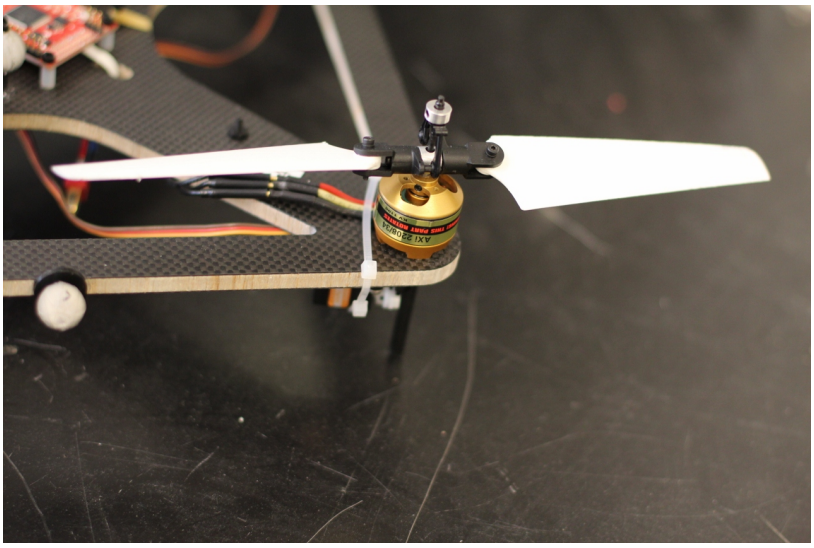


Figure 2-1: The variable-pitch mechanism and corresponding propellers used. The mechanism is commercial and is discussed in detail in Chapter 4.

Table 2.1: Propeller Aerodynamic Coefficients

C_{L0}	C_{La}	C_{Lmin}	C_{Lmax}	C_{D0}	C_{D2u}	C_{D2l}	$C_L C_{D0}$	RE_{ref}	RE_{exp}
0.0	2.87	0.0	0.7	0.01	0.0408	0.0408	0.0	100,000	-0.5

Using XFOIL [31], aerodynamic coefficients for the propeller blades are determined and, for reference, are displayed in Table 2.1.

2.1.2 Motor Model

The variable-pitch quadrotor, like most small, multi-rotor helicopters, utilizes brushless motors as brushless motors have higher torque and power for the same motor weight when compared to brushed motors. Therefore, the motors are not actually receiving a specific voltage that is proportional to motor speed as they are running. Instead, electronic speed controllers carefully create three out of phase pulse width modulated power signals that are sent to the motor. The power signals alternate the polarity of the electromagnets on the motor stator which keep the rotor spinning. The effective power that the motor sees, however, can be modelled as a constant voltage supplied to the motor.

The motor is modeled by a circuit containing a resistor, inductor, and voltage

Table 2.2: Motor Coefficients

K_V (rad/s/volts)	R (Ohms)	i_0 (Amps)
115.2	0.26	0.35

generator in series [32]. Kirchhoff's law applied to the equivalent motor circuit yields

$$v = v_R + v_{L_m} + e \tag{2.1}$$

which can be rewritten as

$$v = Ri + L_m \frac{\partial i}{\partial t} + \frac{\omega}{K_V}, \tag{2.2}$$

where R is the motor internal resistance, L is the inductance, ω is the rotational rate of the motor, and K_V is the voltage constant of the motor, expressed in rad/s/volt. As noted above, Equation 2.2 is an approximation for a brushless DC motor. A more detailed analysis and full derivation of the equations for each of the phases of the brushless motor are shown in [33]; however, the general governing equation presented agrees with Equation 2.2.

The motor torque, T_M, is modeled as being proportional to the difference between the current, i, and the no-load current, i_0, through the torque constant, K_Q, expressed in Amp/Nm.

$$T_M = \frac{(i - i_0)}{K_Q} \tag{2.3}$$

The motor dynamics are modeled as a simple first order differential equation in Eq 2.4 where $\dot{\omega}$ is driven by the motor torque and the load torque, T_L. The inertia, I, includes the motor and the propeller, T_M comes from the voltage generator, and T_L results from the propeller drag.

$$I\dot{\omega} = T_M - T_L \tag{2.4}$$

Typically, the inductance of small, brushless hobby motors is negligible when compared to the physical response of the system and so can be ignored [32]. Substituting Equation 2.2 and 2.3 into Equation 2.4 and neglecting the inductance term yields the

following differential equation for the motor speed, ω,

$$I\dot{\omega} = \left[\left(v - \frac{\omega}{K_V}\right)\frac{1}{R} - i_0\right]\frac{1}{K_Q} - T_L. \tag{2.5}$$

Table 2.2 shows the motor coefficients for the AXI 2208/34 brushless motors used on the variable-pitch quadrotor.

2.1.3 Nonlinear Motor-Propeller Model

QPROP [34] is used to determine the steady state values of the load torque due to the drag, T_L, and the lift, L, generated by the propellers as functions of ω and the pitch angle, α. Figures 2-2-2-4 show the output of QPROP. In each plot, lines of constant thrust (Newtons) are denoted by red numbers and lines of constant motor speed (RPM) are denoted by blue numbers. The grey areas indicating operating regimes that should be avoided due to motor vibrations and propeller stall. In Figure 2-2, the approximate hover thrust required by each of the four motors is denoted by the green line, indicating, as mentioned earlier, the many combinations of pitch and voltage settings that could be used to hover. The plot also shows how thrust can be increased from hover by either increasing motor voltage, increasing pitch, or by increasing both. Figure 2-3 shows the same information as Figure 2-2, except here the horizontal axis shows both positive and negative propeller pitch. One of the key benefits of the variable-pitch propeller actuator is the ability to move anywhere on this thrust plot. With fixed-pitch propellers the vehicle is restricted to moving only vertically on this plot with a fixed propeller pitch.

In Figure 2-4 the vertical axis shows power consumed by the motor as a function of propeller pitch. Another benefit of variable-pitch propellers over fixed-pitch propellers is the ability to choose more efficient operating regimes based on vehicle loading. Although not explored in this paper, Figure 2-4 suggests that propeller pitch could be adjusted to more power efficient settings as the required thrust increases or decreases.

Assuming the quadrotor is near hover, the lift and drag generated by the propeller

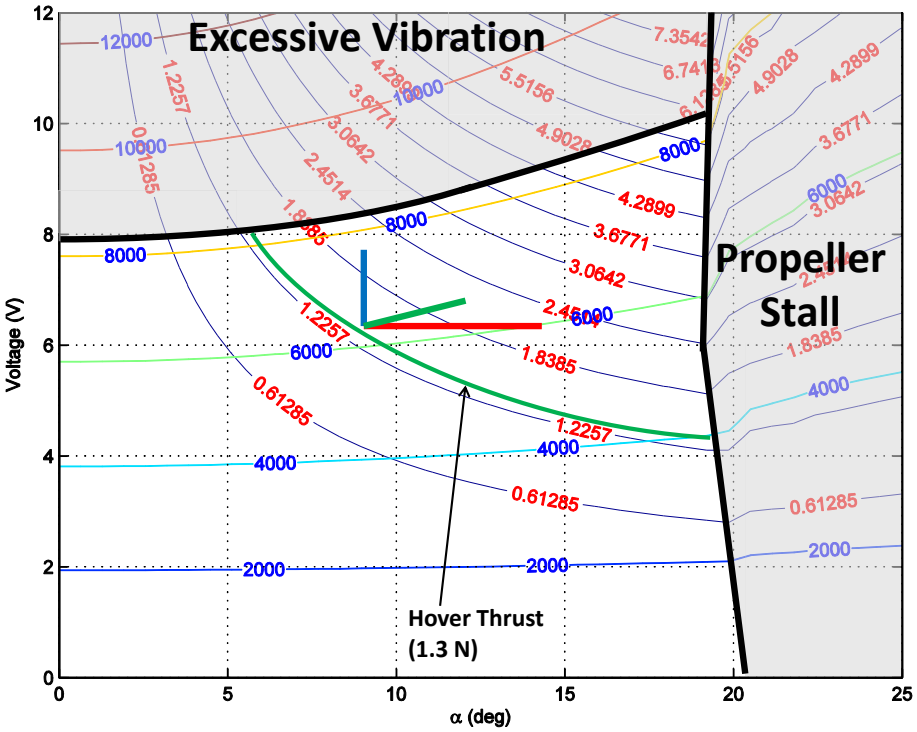


Figure 2-2: Thrust (N), displayed with red numbers, and motor speed (RPM), displayed in blue numbers, as a function of voltage and pitch. Only positive propeller pitch is displayed.

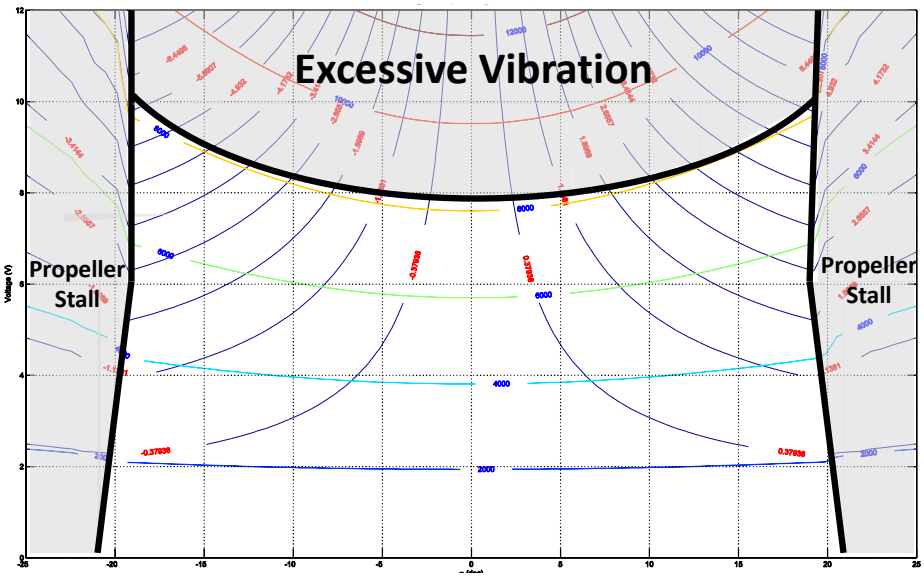


Figure 2-3: Same plot as Figure 2-2 but with both positive and negative pitch shown. One of the fundamental benefits of the variable-pitch actuators for quadrotors is the addition of negative thrust to the flight regime.

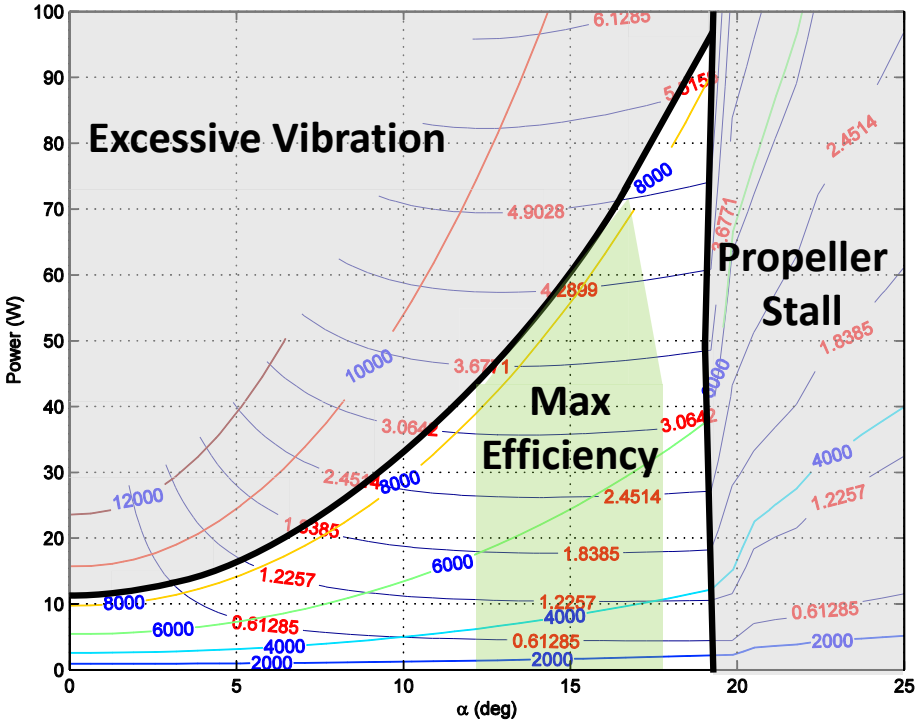


Figure 2-4: Thrust (N), displayed with red numbers, and motor speed (RPM), displayed in blue numbers, as a function of motor power and pitch. Only positive propeller pitch is displayed.

is modeled by [35]

$$L = \rho c R_p^3 \omega^2 C_{L\alpha} \frac{\alpha}{3} \tag{2.6}$$

$$T_L = \rho c R_p^4 \omega^2 \left(\frac{C_{D0} + C_{Di}\alpha^2}{4} - \frac{C_{L\alpha}\alpha\omega}{3R_p} \right). \tag{2.7}$$

These equations show that the lift and drag produced by the motor-propeller combination is affected by the motor speed, ω, and the propeller pitch angle, α. The remaining terms in the two equations are all constants. Combining the constants and calling them b_L, b_{D_1}, b_{D_2}, and b_{D_3}, Equation 2.6 and 2.7 are rewritten as

$$L = b_L \omega^2 \alpha \tag{2.8}$$

$$T_L = b_{D_1}\omega^2 + b_{D_2}\omega^2\alpha^2 + b_{D_3}\omega\alpha \tag{2.9}$$

Table 2.3: Estimated Lift and Drag Coefficients

b_L	b_{D1}	b_{D2}	b_{D3}
3.88e-07	9.96e-09	2.46e-10	4.33e-07

Using least-squares regression, the data generated by QPROP is used to estimate these constants. The numerical values of the constants are shown in Table 2.3 for reference. Substituting Equation 2.9 into Equation 2.5 results in the following nonlinear differential equation for $\dot{\omega}$.

$$I\dot{\omega} = \left[\left(v - \frac{\omega}{K_V}\right)\frac{1}{R} - i_0\right]\frac{1}{K_Q} - b_{D_1}\omega^2 - b_{D_2}\omega^2\alpha^2 - b_{D_3}\omega\alpha \tag{2.10}$$

2.1.4 Linearized Motor-Propeller Model

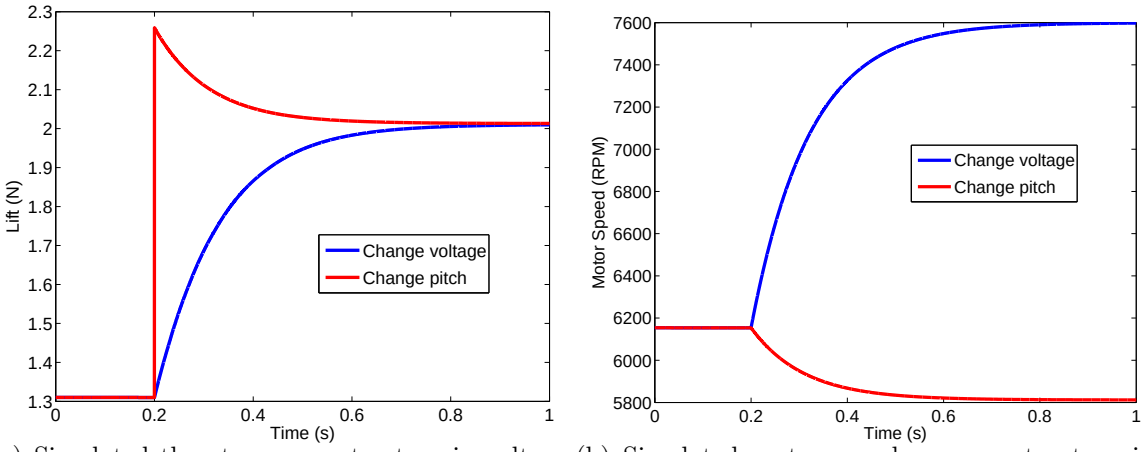
To develop further insights into the dynamics, the nonlinear equation for motor speed (2.10) can be linearized about the hover conditions ω_0 and α_0. The resulting state-space system is

$$\begin{aligned}\Delta\dot{\omega} = -\frac{1}{I}\left[\frac{1}{RK_VK_Q} + 2b_{D_1}\omega_0 + 2b_{D_2}\omega_0\alpha_0^2 + b_{D_3}\alpha_0\right]\Delta\omega \\ + \frac{1}{I}\begin{bmatrix}\frac{1}{RK_Q} & -2b_{D_2}\omega_0^2\alpha_0 - b_{D_3}\omega_0\end{bmatrix}\begin{bmatrix}\Delta v \\ \Delta\alpha\end{bmatrix}\end{aligned} \tag{2.11}$$

$$\Delta L = \begin{bmatrix}2b_L\omega_0\alpha_0\end{bmatrix}\Delta\omega + \begin{bmatrix}0 & b_L\omega_0^2\end{bmatrix}\begin{bmatrix}\Delta v \\ \Delta\alpha\end{bmatrix}. \tag{2.12}$$

The output of this system of equations is the change in lift (ΔL) generated by the propellers; the two inputs are the change in applied motor voltage (Δv) and change in propeller pitch ($\Delta\alpha$).

These linearized equations encapsulate one of the fundamental differences between variable-pitch and fixed-pitch actuators. With a fixed-pitch propeller, thrust is increased by increasing the motor voltage. This motor voltage is applied in Equation 2.11 and first increases the motor speed, which, in turn, increases the thrust



(a) Simulated thrust response to steps in voltage and pitch.

(b) Simulated motor speed response to steps in voltage and pitch.

Figure 2-5: Simulated thrust and motor speed responses to step increases in motor voltage and propeller pitch (the steps occur at 0.2 seconds). The thrust instantaneously increases with increasing pitch but is filtered by the motor dynamics with increasing motor voltage.

output. Thus, the thrust rate of change is fundamentally limited by the dynamics of the motor. However, with a variable-pitch actuator, there is a significant non-zero direct feed-through term in the lift output equation (2.12). Any change in the pitch of the blades directly affects lift, bypassing the motor dynamics. Of course, increasing the pitch of the propellers negatively impacts the speed of the motor and therefore negatively impacts the lift produced, but that decrease in lift is first filtered through the motor dynamics. As long as the mechanism for actuating pitch is fast when compared to the response of the motor, varying the pitch of the blades results in faster changes in thrust than varying the motor voltage.

Nonlinear simulation results from the data generated by QPROP for lift and motor speed responses to steps in voltage and pitch are shown in Figure 2-5. The direct feed-through from pitch to lift is apparent as the lift instantaneously increases when the step in pitch is commanded. The lift then decreases as the increased torque on the motor slows the motor down to the new steady state value.

Graphically, these two different actuators are shown in Figure 2-2. The blue line indicates a step in voltage from hover while the red line represents a corresponding step in pitch from the same initial conditions. The diagonal green line shows a

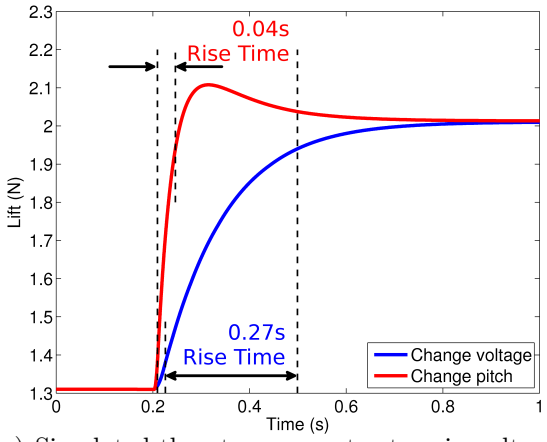
similar change in thrust by using both voltage and pitch inputs. The issue of control allocation and deciding which actuator to use is discussed in Section 2.1.6.

2.1.5 Hardware Effects

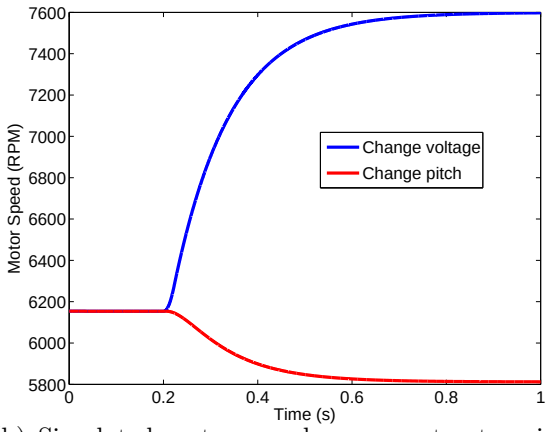
The linearized equations of motion in Equation 2.11 and 2.12 make a number of simplifying assumptions. In addition to the error introduced through linearization, the model assumes both voltage and pitch can be changed instantaneously and continuously. In reality, the voltage to the motor first passes through the dynamics of the electronic speed controller (ESC). Bench tests suggest that the dynamics of the ESC can be approximated as a rate limiter of 70 volts/sec. Also, on the actual quadrotor, the propeller pitch is actuated by a digital servo. The servo dynamics are approximately modeled by a rate limit and a small lag. No load specifications on the servos indicate a rate limit of about 900 degrees/sec. Bench test data indicates that the loaded response is roughly 70% of that value.

Actuator update rates can also affect the time responses of the two actuators. Bench testing of fixed-pitch quadrotors has demonstrated that increasing the update rate of the command signal to the ESC from 200 Hz to 500 Hz greatly improves the visible hover performance of the vehicle. The ESC's on the variable-pitch quadrotor are controlled over an I^2C bus and accept updates at up to 1000 Hz. The servos, however, accept PWM commands at a maximum of 333 Hz. Assuming the attitude loop is running at 1000 Hz or faster, the ESCs can accept commands three times as fast as the servos.

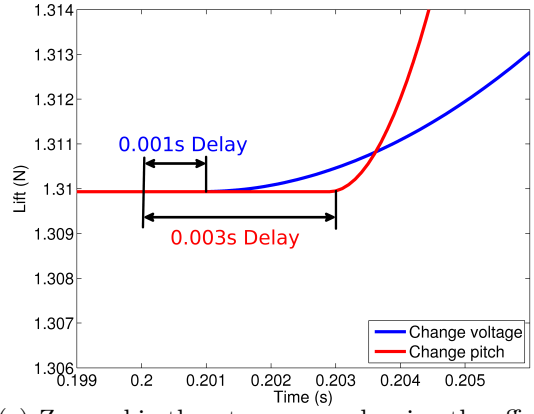
Figures 2-6(a) and 2-6(b) show the same step responses from Figure 2-5 with the effects of the rate limiters, the servo lag and the delay. The delay is modeled as a zero-order hold (ZOH). Digital control theory [36] indicates that a ZOH is well modeled as a delay of one half of the update rate of the actuator. To analyze a worst-case scenario, the ZOH from the update rate is modeled with a delay equal to the update rate of the actuator. Figure 2-6(c) shows the effects of this worst-case analysis of the update rates. The thrust initially increases faster when the voltage is changed, but the servo quickly overtakes it. The initially faster change in thrust with



(a) Simulated thrust response to steps in voltage and pitch.



(b) Simulated motor speed response to steps in voltage and pitch.



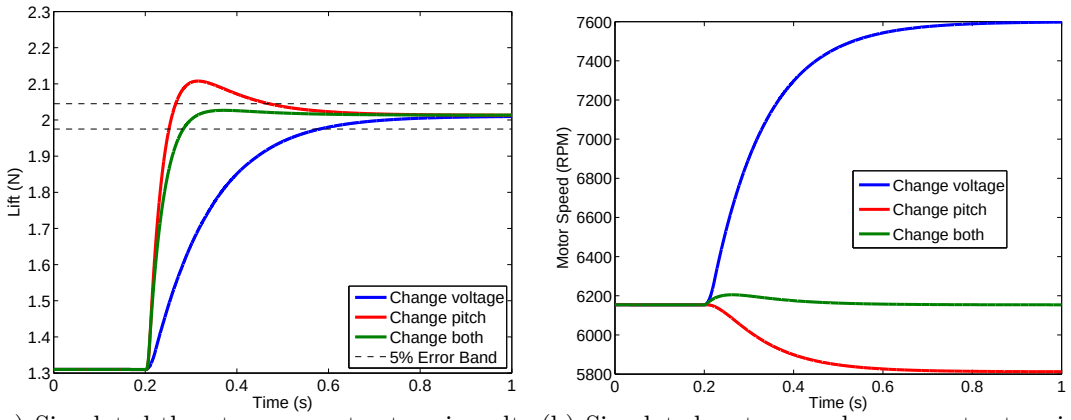
(c) Zoomed in thrust response showing the effects of the actuator digital update rates.

Figure 2-6: Simulated thrust and motor speed responses to step increases in motor voltage and propeller pitch (the steps occur at 0.2 seconds). Here rate limit, servo lag, and delay effects are modeled. Even with these effects modeled, varying propeller pitch still yields substantially faster changes in thrust than varying motor voltage.

voltage control may improve flight during maneuvers such as hovering where small, fast thrust changes are required to fly smoothly.

2.1.6 Control Allocation

Deciding which thrust actuation mechanism (varying voltage or varying pitch) or combination of the two to use depends on a number of factors, including what the control objectives are. For instance, if you are concerned about minimizing power consumption, Figure 2-4 suggests keeping the blade pitch between 12-18 degrees when hovering since the constant thrust lines have minimum power in this range. However,



(a) Simulated thrust response to steps in voltage and pitch.

(b) Simulated motor speed response to steps in voltage and pitch.

Figure 2-7: Simulated thrust and motor speed responses to step increases in motor voltage and propeller pitch (the steps occur at 0.2 seconds). Changing thrust by varying both motor speed and propeller pitch significantly reduces the effects of motor dynamics, yielding clean, fast changes in thrust.

if you are concerned about the ability to quickly increase the thrust of the motors, hovering with a lower pitch and higher RPM allows you to increase the pitch more before stalling the propellers. In other words, the motor-propeller combination contains more kinetic energy at hover that can be quickly converted into thrust by increasing the propeller pitch.

Agile and aggressive flight requires quick changes in thrust [9]. It follows that for agile flight the kinetic energy stored in the motor should be maximized. As indicated in Figure 2-2, bench tests show that vibration from the motors becomes excessive for the frame and autopilot around 8000 RPM, placing an upper limit on the kinetic energy that can be stored in the motor. Keeping the energy high under varying pitch values implies that the power to the motor must be adjusted to compensate for the varying propeller load.

One strategy for agile flight is to always keep the motor speed at its upper limit. Given a nominal hovering thrust, T_{L_0}, and maximum motor speed, ω_{max}, the required hover voltage, v_0, is found using Equation 2.5. Let ΔT_L be the required change in thrust to go from hover to a new thrust set point. The corresponding required change in motor voltage to ensure a constant steady state motor speed is found by solving

Equation 2.5 for Δv when $\dot{\omega} = \Delta\omega = 0$. Thus,

$$\Delta v = RK_Q \Delta T_L, \tag{2.13}$$

is the required change in voltage as the pitch is changed to keep the RPM constant. Adding power to the motor as the pitch is increased to keep the motor speed constant is effectively canceling the effects of the motor dynamics. Figures 2-7(a) and 2-7(b) demonstrate that this strategy increases the thrust quickly while avoiding a RPM and thrust decrease after the increase in pitch. As depicted in Figure 2-2 with the diagonal green line, this thrust change lies roughly parallel with the lines of constant motor speed, indicating that the steady-state motor speed remains constant. The mitigation of slow motor dynamics during thrust changes illustrates an advantage of using both voltage and pitch control in combination. In practice, this strategy is utilized on the variable-pitch quadrotor in order to maximize the potential for agile and aggressive flights.

2.2 Bench Motor Testing

The analytical results presented in the previous section are verified here using bench testing of one of the motors and propellers used on the variable-pitch quadrotor. A photograph of the testing setup is shown in Figure 2-8.

The motor and propeller are rigidly attached to a 5 pound load cell. The load cell measures vertical force and is used to measure the thrust generated by the propeller. It is powered by the amplifier and produces a ± 12 volt analog output that is proportional to the measured load. This analog output is sampled by the analog to digital converter at 10 kHz. The motor and servo pitch actuator are controlled by the autopilot and are commanded to give step increases in both the motor speed and pitch values. Other sensors on the test stand include a current and voltage sensor and an optical RPM sensor for measuring the speed of the propeller.

Figures 2-9 and 2-10 show the results of the bench testing. In Figure 2-9, steps in

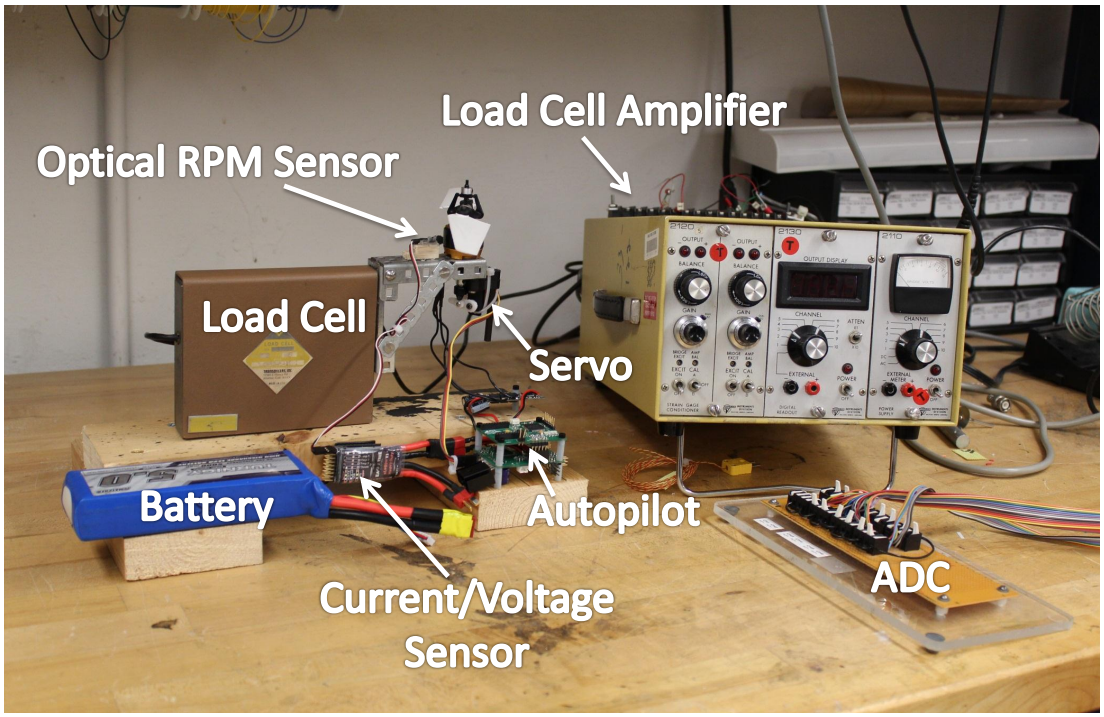
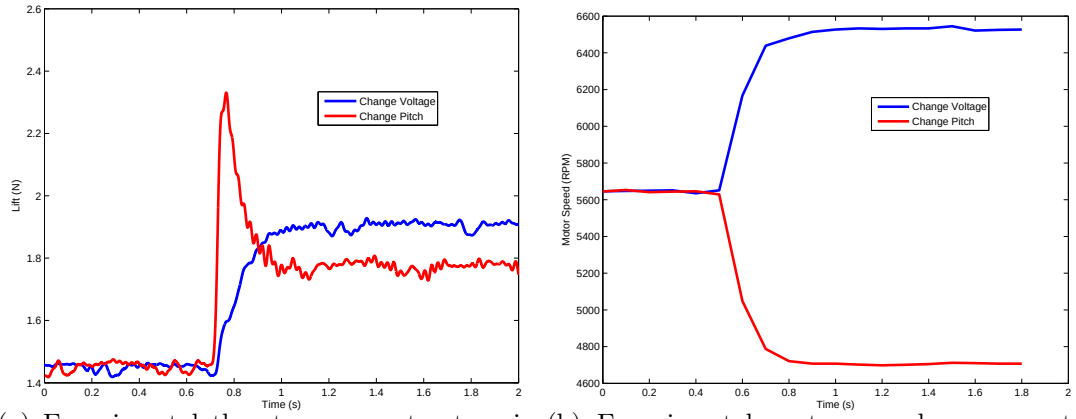


Figure 2-8: The test setup used for experimentally testing thrust changes using both fixed-pitch and variable-pitch propellers. The motor and propeller are attached to a 5 pound load cell that is powered by the amplifier. The autopilot controls the motor and servo. The analog load cell output is sampled and digitized by the analog to digital converter.

motor command and pitch command are given independently. While the decreases in thrust and motor speed after the initial command is given are greater than predicted by the simulations (Figure 2-6), the general shapes of the graphs agree with the simulated data. When increasing the motor command and keeping the pitch constant, the rate of change of the thrust is governed by the inertia of the motor and propeller, causing the thrust response to respond like a first order system. On the other hand, when pitch is increased with a constant motor command, the thrust changes nearly instantaneously. This is followed by a decrease in thrust as the increased propeller drag slows the motor down.

In Figure 2-10 the same data is displayed, but an example of changing both motor command and propeller pitch together is also overlaid. As predicted by the simulation, when the actuators are changed together, an effective cancellation of the motor dynamics happens since the steady-state motor speed remains constant. There



(a) Experimental thrust response to steps in voltage and pitch.

(b) Experimental motor speed response to steps in voltage and pitch.

Figure 2-9: Experimental thrust and motor speed responses to step increases in motor voltage and propeller pitch. The drop in motor speed after the pitch increase is larger than what was predicted in Figure 2-6(b); however, the shape of the graphs are consistent with the simulated data.

is a small dynamic response immediately after the pitch is actuated; however, the motor speed quickly returns to the previous same steady-state value.

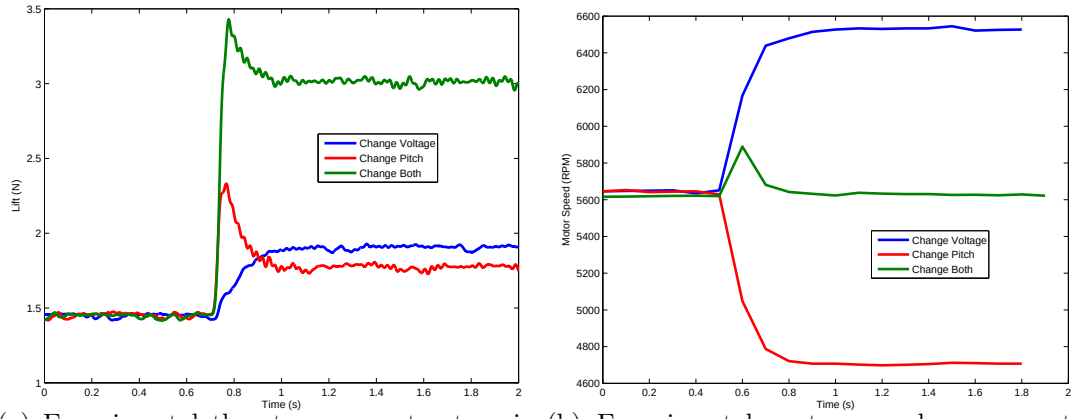
These experimental results support the theoretical analysis and show that varying the propeller pitch yields fast, almost instantaneous thrust changes.

2.3 Simulation

In Section 2.1, the dynamic relationships between thrust, pitch, and motor speed were developed. These relationships were verified experimentally using a load cell in Section 2.2. In this section, two nonlinear simulations of the variable-pitch quadrotor using the data from QPROP in Section 2.1 are used to show the extent to which variable-pitch propellers increase the flight performance of the vehicle as a whole.

2.3.1 Vertical Tracking

Only vertical flight is considered in the first simulation. The quadrotor is commanded to follow a sequence of constant vertical velocity commands. For several different velocity commands, the performance of the vehicle is considered using both fixed-pitch (motor voltage commands) and variable-pitch control. The quadrotor is simulated in



(a) Experimental thrust response to steps in voltage and pitch.

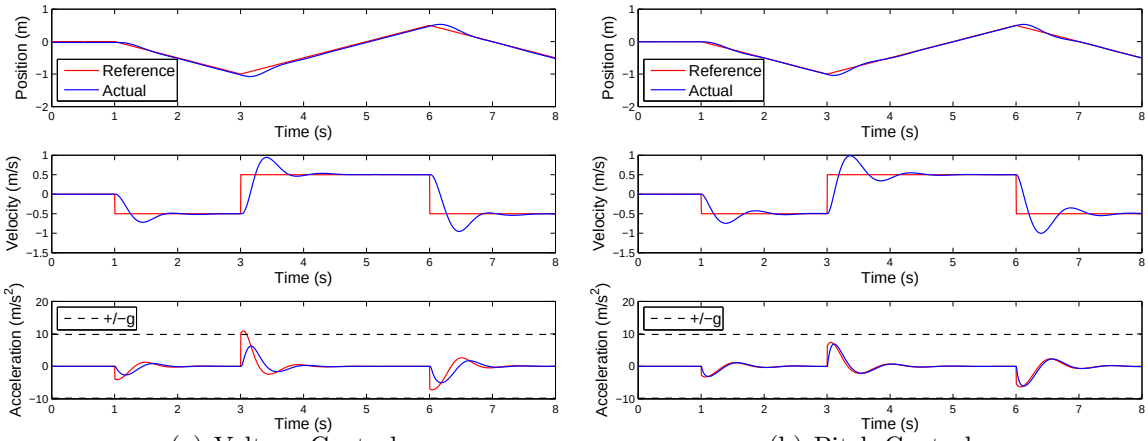
(b) Experimental motor speed response to steps in voltage and pitch.

Figure 2-10: Same results as Figure 2-9 but with data from varying both motor voltage and propeller pitch overlaid. When both actuators are used, the motor speed remains essentially constant, showing that the motor dynamics are largely cancelled.

closed-loop control using a proportional-derivative (PD) controller with position and velocity feedback. Since the two actuators generate different thrust values for the same input command, the PD gains for both actuators were selected to ensure the dominate linearized closed-loop poles are in the same location for both cases. Thus the two cases can be compared with each other despite being in closed-loop control.

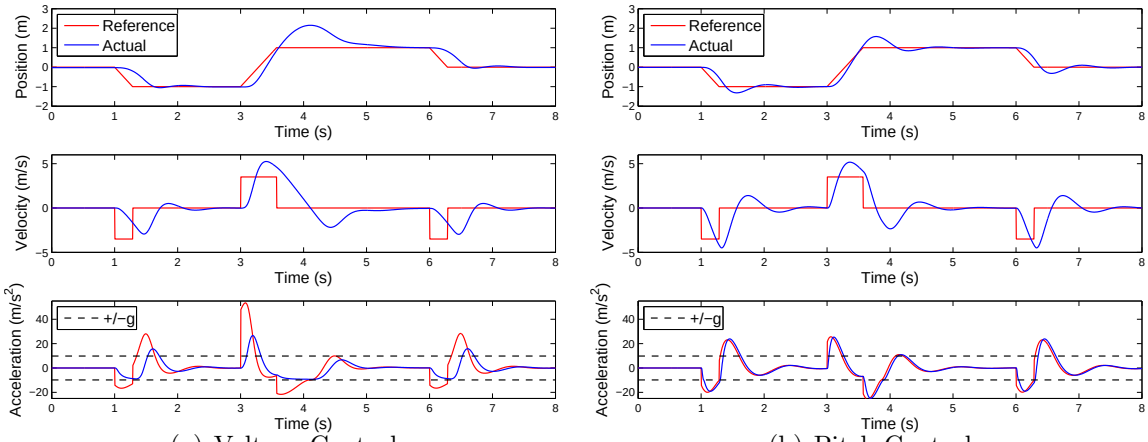
Figure 2-11 shows the time response of the simulated vertical tracking when the commanded vertical velocity is 0.5 m/s. Note that the closed-loop position and velocity responses are roughly the same for the two actuators since the dominant closed-loop poles are in the same location. Interestingly, though, the fixed-pitch controller commands a much higher thrust input (thrust and acceleration are directly proportional) to get the same response since the actuator is slower than the variable-pitch case. When under variable-pitch control, though, the thrust reference commands are lower because the actuator is able to keep the control error significantly smaller.

Figure 2-12 demonstrates that as the commanded vertical velocity is increased to 3.5 m/s, the two cases differ significantly both in reference command and in closed-loop time response. With larger velocity errors, the fixed-pitch controller's need to create large reference commands is even more pronounced than in Figure 2-11. Also, another of the fundamental differences between the two actuators is prominent in this plot. When the propeller pitch is fixed, the quadrotor's ability to decelerate is limited



(a) Voltage Control (b) Pitch Control

Figure 2-11: Vertical tracking at 0.5 m/s



(a) Voltage Control (b) Pitch Control

Figure 2-12: Vertical tracking at 3.5 m/s

by gravity since the motor cannot efficiently reverse its direction mid flight; however, when the pitch is controlled, large decelerations are realized due to the propellers' ability to quickly transition from positive to negative thrust. The acceleration plot in Figure 2-12(a) shows that even though the fixed-pitch controller requests accelerations below gravity, the actuator is saturated and cannot follow the reference command, resulting in a large overshoot in the position response.

The vertical tracking performance of the vehicle when in fixed-pitch and variable-pitch modes using controllers with the same dominant closed-loop poles is summarized in Figure 2-13. Each dot on the graph shows the sum of the absolute value of the position error at each time-step during the simulation. As expected, at low reference velocities the controllers have roughly the same error since the controllers perform

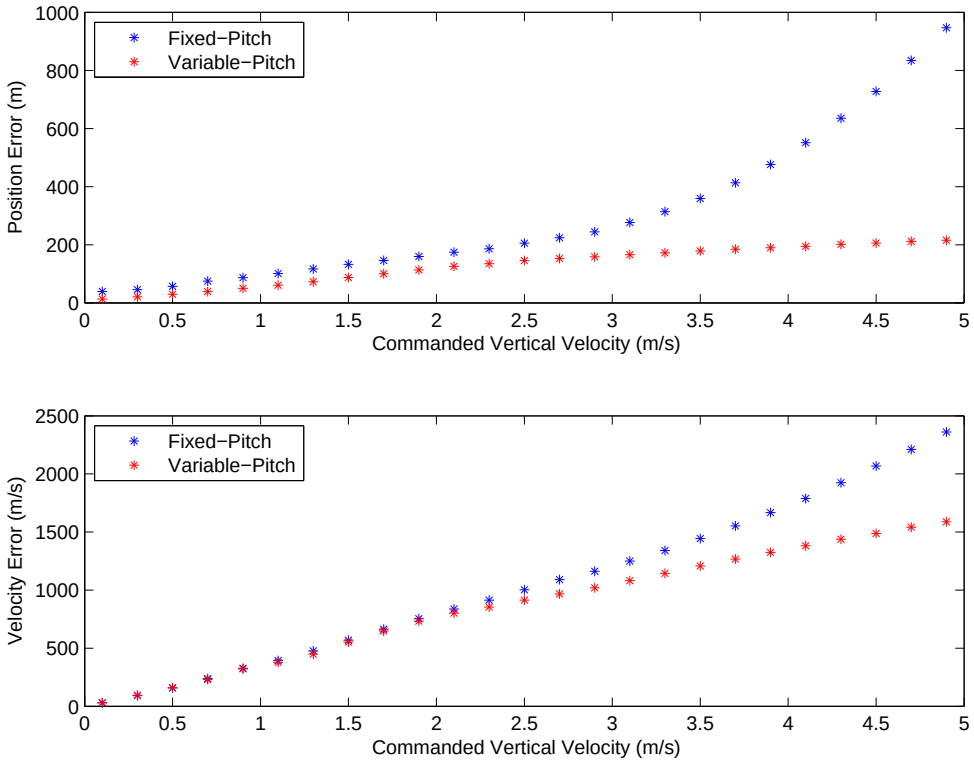


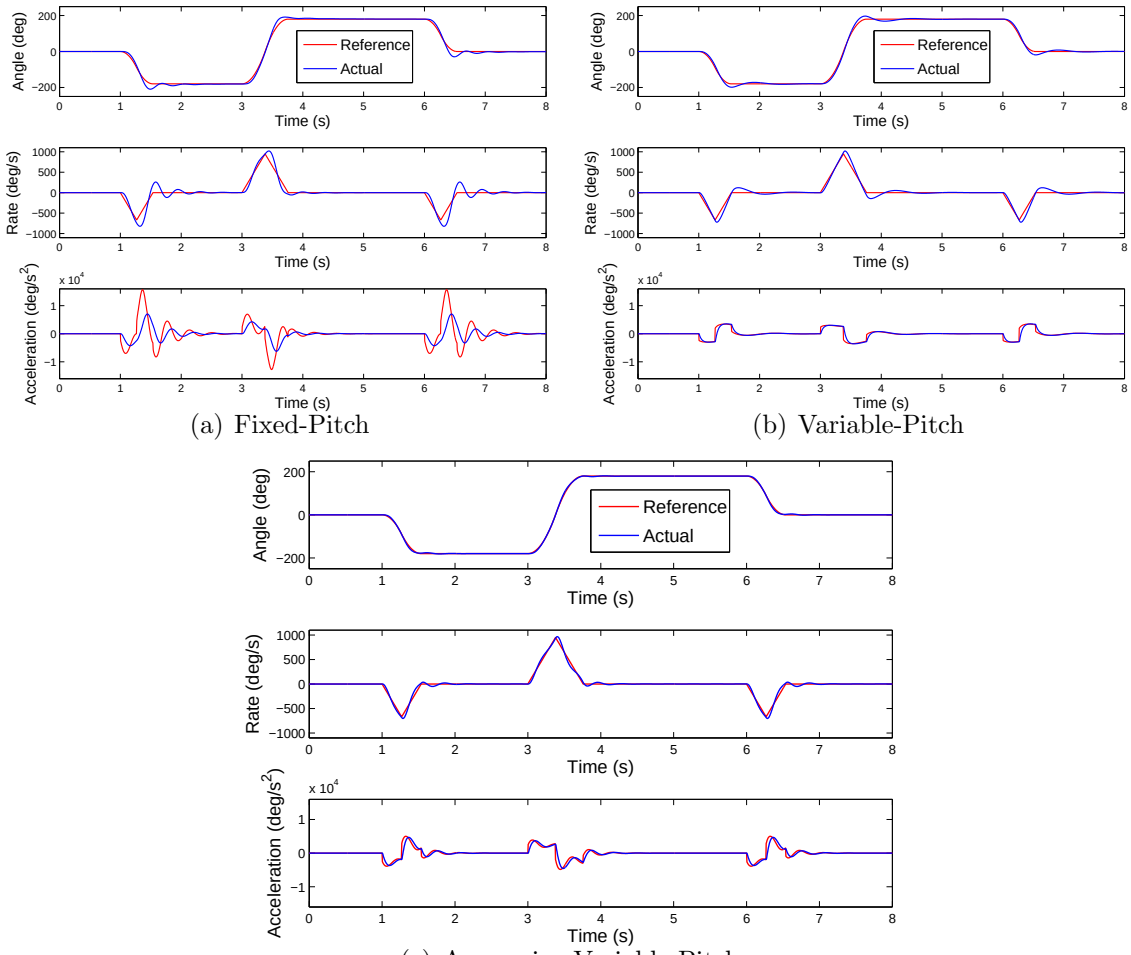
Figure 2-13: Vertical tracking error

similarly. However, the errors start drastically diverging at around 2 m/s commanded velocity, the speed at which the fixed-pitch controller's deceleration starts saturating.

Experimental validation of this simulation is provided in Section 5.2.

2.3.2 Rotational Tracking

The second simulation considers only the roll axis of the quadrotor. The vehicle is commanded to follow a sequence of constant angular acceleration commands. Constant acceleration is commanded instead of constant velocity as in the vertical simulation because the smoother command shaping greatly improves quadrotor response with the lightly damped roll axis. As in the vertical simulation, the controller used is a PD controller with roll and roll rate feedback. The controller gains are selected so that the two cases have the same dominant closed-loop pole locations. Figure 2-14 shows the roll, roll rate, and roll acceleration time responses for the two different actuators when the commanded roll acceleration is 2500 deg/s^2. Again, the fixed-pitch controller requests much larger reference thrust commands than the variable-pitch



(a) Fixed-Pitch

(b) Variable-Pitch

(c) Aggressive Variable-Pitch

Figure 2-14: Roll tracking at 2500 deg/s^2

controller in order to get similar roll time responses.

The accumulated roll and roll rate tracking errors as a function of commanded roll rate are displayed in Figure 2-15. As in the vertical flight simulation, the fixed-pitch and variable-pitch actuators perform similarly when the commanded angular acceleration is low. However, as the commanded acceleration, or desired aggressiveness, increases, the fixed-pitch propellers begin to saturate while the variable-pitch propellers do not. Theoretically, the variable-pitch propellers can produce at least twice the torque to the vehicle when rolling as the fixed-pitch propellers because of the ability to reverse the direction of thrust on one of the motors. Thus, the fixed-pitch propellers begin to saturate as one motor produces nearly zero thrust and the other motor produces maximum thrust. This saturation is clearly evident in the error plot

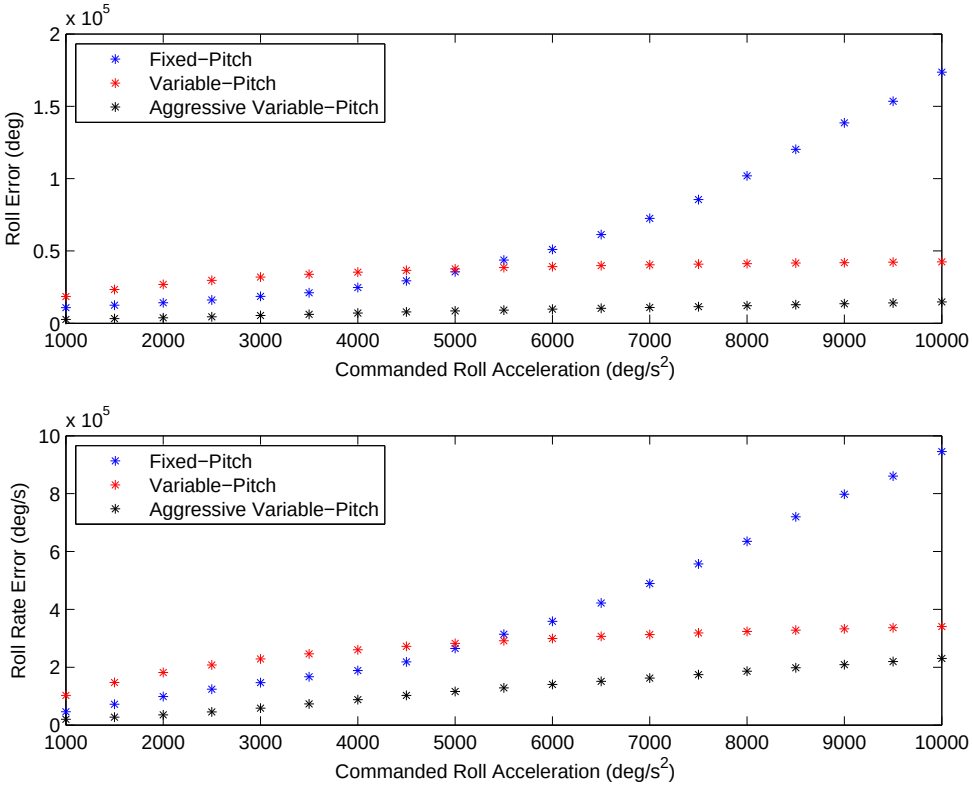
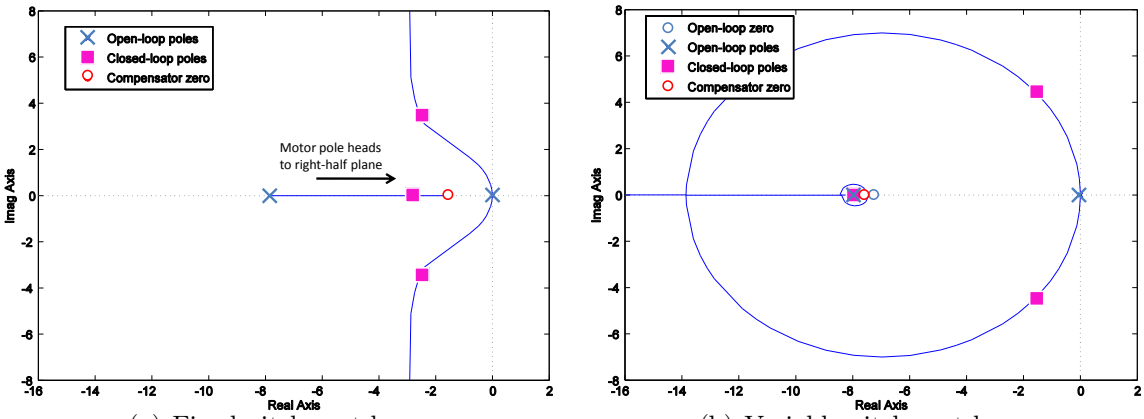


Figure 2-15: Roll tracking error

as the fixed-pitch error diverges from the variable-pitch error.

Another benefit of variable-pitch propellers when rotating about an axis lies in their ability to quickly track a reference command. This is more clearly shown in the linearized root locus plots of the closed-loop roll axis. For both actuators, the plant consists of three poles, two of which are at the origin for the roll axis. The variable-pitch actuator adds an additional zero near the motor pole coming from the direct feed through term in Equation 2.12. Figure 2-16 shows that although the two controllers have placed the closed-loop poles in roughly the same location, the gain on the variable-pitch controller in Figure 2-16(b) can still be increased significantly to improve the time response of the system. However, it is clear in Figure 2-16(a) that moving the compensator pole towards the origin to increase the speed of the closed-loop poles will only lead to the system becoming dominated by the motor pole. This pole will move into the right-half plane as the PD gains are increased. These plots highlight the fact that the variable-pitch controller largely mitigates the effect of the



(a) Fixed-pitch root locus

(b) Variable-pitch root locus

Figure 2-16: Root locus for fixed-pitch and variable-pitch propellers with a PD controller. In the fixed-pitch controller, the compensator zero is moved to the right to pull the poles at the origin to the left. In doing so, the motor pole heads to the right, eventually dominating the response and potentially causing the system to go unstable. The direct feed through term in the variable-pitch actuator, however, adds a zero near the motor pole allowing the poles at the origin to be pulled far into the left half plane.

motor dynamics on the system. The same phenomenon occurs in the vertical-control case.

Figure 2-14(c) shows the same reference commands but with a much more aggressive controller. The tracking performance is better with the more aggressive controller, and this performance is only attainable with variable-pitch control as the motor pole goes unstable in the fixed-pitch case. Figure 2-15 also shows the error improvement over the range of commanded roll rates using the nominal and more aggressive controllers. The variable-pitch controller can be tuned to deliver much more aggressive responses than the fixed-pitch controller can.

2.4 Experimental Actuator Testing

The simulation data from Section 2.3.2 is validated in this section with experimental testing of the quadrotor while it is attached to a test fixture that restricts motion to a single axis. The test fixture is shown in Figure 2-17. During the testing, the quadrotor is commanded to follow a series of roll and roll rate commands similar to those given

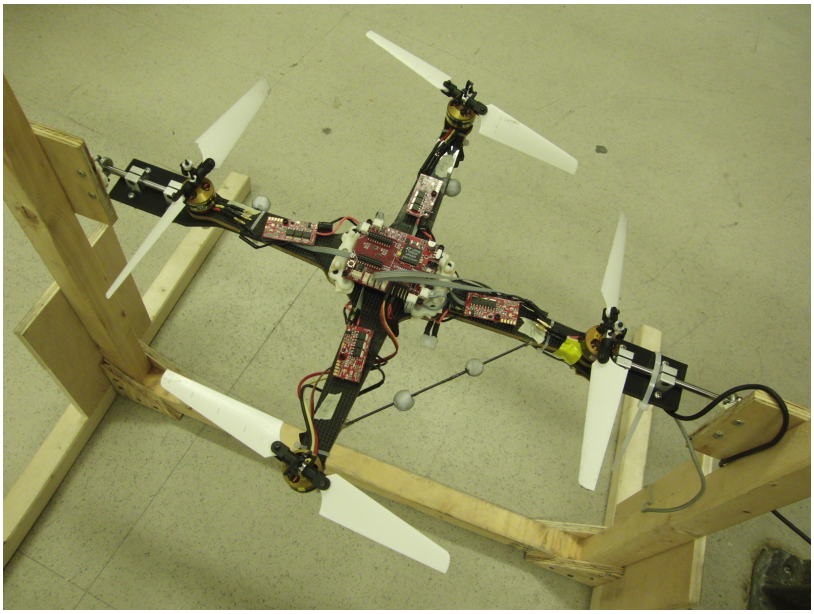
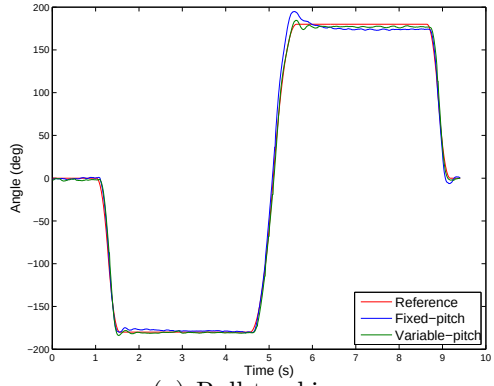


Figure 2-17: Quadrotor attached to roll test stand. Bearings mounted to the wooden frame allow for rotations about a single axis. The test stand is used for testing aggressive roll commands.

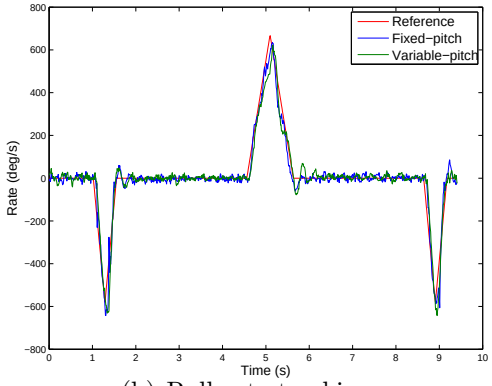
in the rotational simulation in Section 2.3.2. Several different trajectory profiles are tracked with varying commanded accelerations. The same control strategy (PD) used in simulation is used in the experiments.

Figure 2-18 shows experimental results of the quadrotor tracking an angular acceleration reference command of 2500 deg/s^2. As predicted from the simulation, the angle and angular rate tracking for both the fixed-pitch and variable-pitch cases are similar. The vehicle's inertia is small enough that, despite being a slower actuator, the fixed-pitch actuator can track the reference input well.

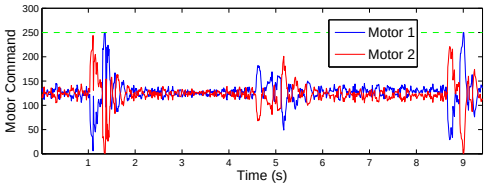
The difference between the two actuators is apparent, however, in the commanded values shown in Figures 2-18(c) and 2-18(d). These plots show the commands generated by the on-board controller and sent to the actuators. Both the commands to the motor via the ESC and the commands to the pitch via the servo are values between 0 and 250. For the motor, a command of 0 corresponds to off and 250 to maximum throttle. For the propeller, neutral pitch is set with a servo command of 125 while full positive pitch is 250 and full negative pitch is 0. The dashed green line in each plot shows the upper saturation limit of the actuators. Note that in the fixed-pitch

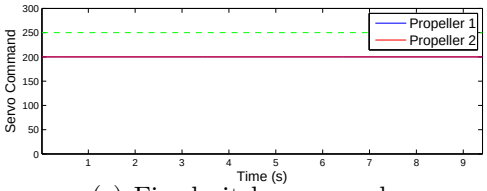


(a) Roll tracking

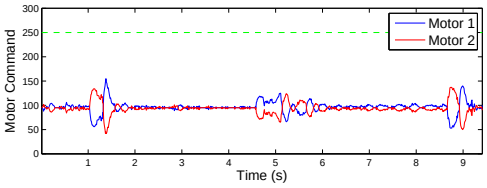


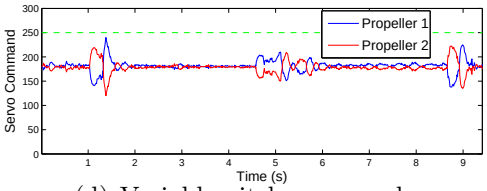
(b) Roll rate tracking





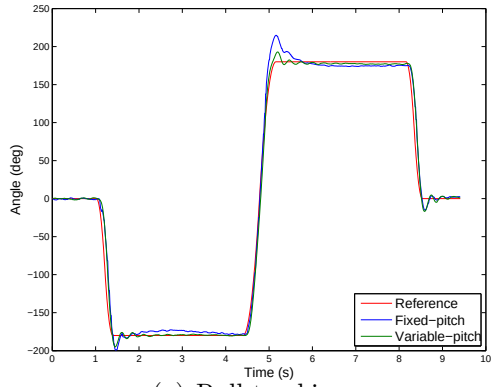
(c) Fixed-pitch commands



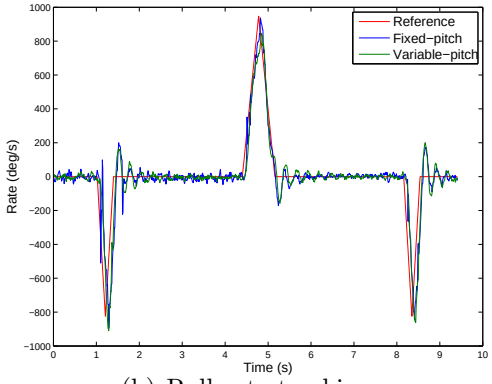


(d) Variable-pitch commands

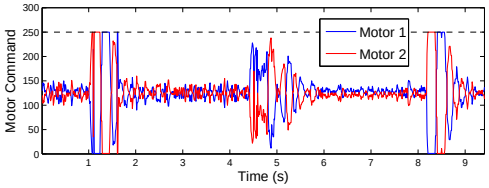
Figure 2-18: Experimental roll tracking at 2500 deg/s^2

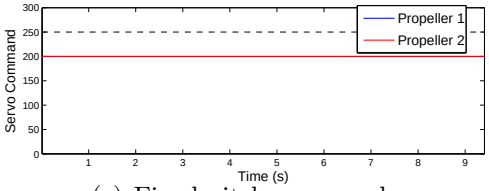


(a) Roll tracking

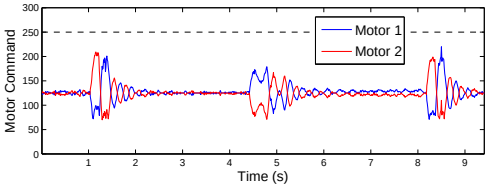


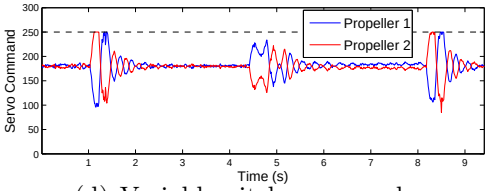
(b) Roll rate tracking





(c) Fixed-pitch commands





(d) Variable-pitch commands

Figure 2-19: Experimental roll tracking at 5000 deg/s^2

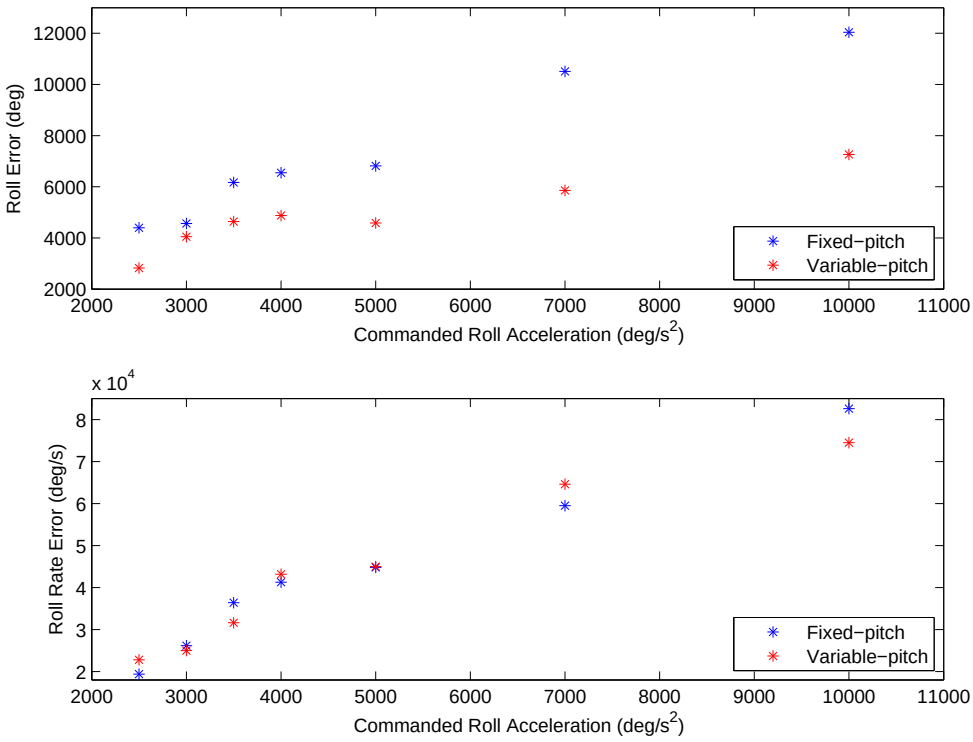


Figure 2-20: Experimental roll and roll-rate tracking error as a function of increasing commanded roll acceleration. The quadrotor is able to consistently track angular reference commands better with variable-pitch actuation than with fixed-pitch actuation. The differences between the actuators grow as the commanded aggressiveness increases and the fixed-pitch actuator begins to saturate.

case, the motor commands are nearly saturated during two of the flips while in the variable-pitch case, the servo and motor commands remain well within the saturation bounds.

The saturation issues are even more apparent as the commanded acceleration is increased. Figure 2-19 shows similar plots but where the commanded acceleration is 5000 deg/s^2. Figure 2-19(c) shows that the motor commands in the fixed-pitch case are saturated for the majority of the first and third flips. The variable-pitch case performs the same maneuvers with less error while only briefly saturating the pitch command. The plots of the commanded values emphasize the ability of the variable-pitch actuator to quickly track reference commands.

For each of the experimental data sets, a measure of the tracking error is computed by summing the absolute value of the difference between the reference command and

the actual value over the entire data set. These error values are shown in Figure 2-20. For a commanded acceleration of 10,000 deg/s^2 the quadrotor reaches a roll rate of 1400 deg/s, well into the agile flight regime. As predicted by simulation, the quadrotor performs better with the variable-pitch propellers than with fixed-pitch propellers.

2.5 Summary

This chapter details the analysis and experimental testing of variable-pitch actuators added to a quadrotor. Section 2.1 shows that variable-pitch actuators fundamentally improve the performance and capability of quadrotors. The ability to control thrust using propeller pitch in addition to motor voltage allows for a significantly faster rate of change in thrust as well as the ability to choose between more efficient or more agile operating regimes. The analysis is experimentally verified using a load cell to capture high rate thrust data in Section 2.2.

The simulation and experimental results in Sections 2.3 and 2.4 show that in nominal (non-agile) flight profiles there is little difference between fixed- and variable-pitch actuation (other than, of course, the ability to generate negative thrust in order to fly upside down and decelerate faster than gravity). However, in more aggressive maneuvers it is apparent that fixed-pitch actuation suffers from control saturation well before the variable-pitch configuration. The fixed-pitch saturation leads to significantly larger errors in attitude rate and position tracking for large commanded angular and linear accelerations, respectively. Thus, variable-pitch actuation increases the potential for agile maneuvering by expanding the control saturation limits.

Finally, the variable-pitch quadrotor has the fundamental advantage of being able to reverse thrust very quickly. Not only does this result in more accurate attitude and acceleration command tracking, but it also expands the envelope of attainable agile maneuvers. Aerobatic maneuvers such as 180 degree flips and tic-tocs that have been demonstrated by conventional helicopters are impossible for traditional quadrotors, but might be obtainable with variable-pitch actuation. In the next chapter, control laws and trajectory generation methods are developed to exploit the increased

capabilities of the variable-pitch quadrotor to fly aggressive and aerobatic maneuvers.

Chapter 3

Trajectory Generation and Control

This chapter describes the trajectory generation and control algorithms used on the variable-pitch quadrotor. While the algorithms presented here are implemented on a variable-pitch quadrotor, they are general and can be applied to quadrotors with fixed-pitch propellers as well. Similar to recent literature [15, 37], the attitude control law presented here does not assume near hover flight regimes, allowing the vehicle to track attitudes that deviate significantly from nominal conditions.

Recent work demonstrates optimal trajectory generation methods for quadrotors using time-parametrized polynomials to represent the trajectory, guaranteeing smooth reference inputs to the quadrotor [15]. The work presented here builds on the literature by presenting a method for tracking a series of waypoints given the physical limitations of the hardware actuators. Time-optimal solutions, subject to actuator saturation, for paths parametrized by polynomials are found. In addition, a method for embedding attitude specific constraints along the reference path is developed, allowing for aerobatic maneuvers such as flips to be performed with a single control law. Most previous aerobatic work with quadrotors was accomplished by the means of a switching control law [12, 24, 25].

The structure of the chapter is as follows: first, a dynamic model of the quadrotor is developed in Section 3.1 and a feedback control solution is proposed to control the quadrotor along a specified 3-D trajectory in $\mathbb{R}^3$ in Section 3.2. Then, a closed-form solution for generating smooth trajectories through any number of time-parametrized

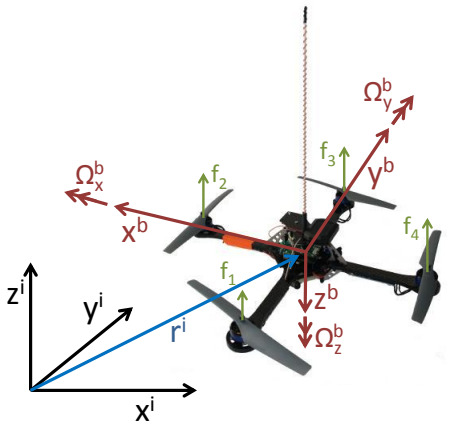


Figure 3-1: Quadrotor model and reference frames. superscript i denotes the inertial frame and superscript b denotes the body frame.

waypoints is proposed in Section 3.4. An optimization method is proposed for constructing smooth, minimum-time trajectories through waypoints while satisfying motor saturation constraints in Section 3.4.1. In Section 3.4.2 a method for embedding attitude constraints along the path is presented. Simulation results of these algorithms are shown in Section 3.5. Chapter 5 details the experimental implementations of the algorithms on a variable-pitch quadrotor.

3.1 Dynamic Model

Consider the quadrotor vehicle depicted in Figure 3-1 with mass, m, and mass moment of inertia, $\mathbf{J}$, where $\mathbf{J}$ is aligned with the body x, y, and z axes. Let the position of the center of mass of the quadrotor with respect to an inertial frame, i, be defined by $\mathbf{r}^i$. The attitude of the vehicle in the inertial frame is described by the quaternion $\mathbf{q}$ with the rotational velocities of the vehicle in the body frame, b, being $\mathbf{\Omega}^b$. The quaternion convention

$$\mathbf{q} = \begin{bmatrix} q^0 \\ \vec{q} \end{bmatrix}$$

is used where q^0 is the scalar portion and $\vec{q}$ is the vector portion of the quaternion. In particular, the quaternion rotation operation that rotates the vector $\mathbf{v}$ in $\mathbb{R}^3$ from

the body frame to the inertial frame is defined as

$$\begin{bmatrix} 0 \\ \mathbf{v}^i \end{bmatrix} = \mathbf{q}^* \otimes \begin{bmatrix} 0 \\ \mathbf{v}^b \end{bmatrix} \otimes \mathbf{q}, \tag{3.1}$$

where $\mathbf{q}^*$ is the quaternion conjugate of $\mathbf{q}$ and $\otimes$ is the quaternion multiplication operator [38]. The quaternion $[0, \mathbf{v}^T]^T$ is a pure imaginary quaternion (a quaternion with zero scalar part). Common quaternion properties are included in Appendix A for convenience.

The inertial-frame time derivative of $\mathbf{q}$ is related to the body rotational velocities by

$$\dot{\mathbf{q}} = \frac{1}{2}\mathbf{q} \otimes \begin{bmatrix} 0 \\ \mathbf{\Omega}^b \end{bmatrix}.$$

Using this quaternion formulation, the Newton-Euler equations of motion that describe the dynamic motion of the quadrotor are given by

$$\begin{bmatrix} 0 \\ \ddot{\mathbf{r}}^i \end{bmatrix} = \frac{1}{m}\mathbf{q}^* \otimes \begin{bmatrix} 0 \\ \mathbf{F}^b \end{bmatrix} \otimes \mathbf{q} - \begin{bmatrix} 0 \\ \mathbf{g}^i \end{bmatrix} \tag{3.2}$$

$$\dot{\mathbf{\Omega}}^b = \mathbf{J}^{-1}\left[\mathbf{M}^b - \mathbf{\Omega}^b \times \mathbf{J}\mathbf{\Omega}^b\right] \tag{3.3}$$

where $\mathbf{g}^i = [0, 0, g]^T$ is the inertial frame gravity vector, $\mathbf{F}^b = [0, 0, f_{total}]^T$ is the body frame thrust vector, and $\mathbf{M}^b$ is the body frame moment vector. Note that the placement of the motors on the quadrotor restricts the body frame thrust vector to always be aligned with the body frame z-axis.

Let the thrust generated by each of the four motors on the quadrotor be f_i. The total thrust f_{total} and quadrotor moments are related to the thrust of each of the four motors by [39]

$$\begin{bmatrix} f_{total} \\ \mathbf{M}^b \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ d & 0 & -d & 0 \\ 0 & d & 0 & -d \\ -c & c & -c & c \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{bmatrix} \tag{3.4}$$

where d is the distance from the center of mass of the vehicle to the motor mount and c is the drag coefficient that relates the yawing moment about the body z-axis to the thrust of the four motors. The thrust produced by each motor is bounded between maximum and minimum values as

$$f_{\min} \leq f_i \leq f_{\max}, \quad i = 1, \ldots, 4 \tag{3.5}$$

where $f_{\min}$ and $f_{\max}$ are determined by the physical characteristics of the motor, the available power, propeller, etc. With fixed-pitch propellers, the theoretical minimum thrust is $f_{\min} = 0$, but in practice one typically finds that $f_{\min} > 0$ [24, 37] since commonly used motor speed controllers cannot quickly start and stop the rotation of the motor. Turning one or more motors completely off mid-flight can lead to unstable behaviors for multi-rotor helicopters. For a variable-pitch system, one can design $f_{\min} = -f_{\max}$.

3.2 Closed-loop Control

Quadrotors are under-actuated and differentially flat [15]. The four motor thrust commands can therefore be determined by four flat outputs: an inertial-frame position reference command, $\mathbf{r}_d^i(t)$, in $\mathbb{R}^3$ and a desired yaw angle, $\psi_d(t)$. Given the flat outputs, the commanded thrust and moments are computed as follows. First, a *feedback acceleration vector* (the time dependence has been omitted for clarity), $\ddot{\mathbf{r}}_{fb}^i$, is computed as

$$\ddot{\mathbf{r}}_{fb}^i = -\mathbf{k}_p\mathbf{e} - \mathbf{k}_i \int_0^t \mathbf{e}(\tau)d\tau - \mathbf{k}_d\dot{\mathbf{e}} \tag{3.6}$$

where $\mathbf{k}_p, \mathbf{k}_i, \mathbf{k}_d$ are positive definite, diagonal, 3×3 gain matrices and the error terms are defined as

$$\mathbf{e} = \mathbf{r}^i - \mathbf{r}_d^i \tag{3.7}$$

$$\dot{\mathbf{e}} = \dot{\mathbf{r}}^i - \dot{\mathbf{r}}_d^i. \tag{3.8}$$

The feedback acceleration vector supplements the commanded (feedforward) acceleration by compensating for gravity and for errors in position and velocity.

Let the total commanded inertial-frame force required to keep the quadrotor on the desired trajectory be

$$\mathbf{F}^i = m\left(\ddot{\mathbf{r}}_d^i + \ddot{\mathbf{r}}_{fb}^i + \mathbf{g}^i\right). \tag{3.9}$$

Note that during hover, the commanded acceleration vector is zero and the force vector approaches $\begin{bmatrix}0 & 0 & mg\end{bmatrix}^T$ as the position and velocity errors approach zero, as expected.

The commanded inertial-frame force vector is used to compute the desired vehicle attitude and the total quadrotor thrust. Rearranging Equation 3.2 yields

$$m\left(\begin{bmatrix}0\\ \ddot{\mathbf{r}}^i\end{bmatrix} + \begin{bmatrix}0\\ \mathbf{g}^i\end{bmatrix}\right) = \mathbf{q}^* \otimes \begin{bmatrix}0\\ \mathbf{F}^b\end{bmatrix} \otimes \mathbf{q}. \tag{3.10}$$

Substituting Equation 3.9 for the left hand side of Equation 3.10 and normalizing both sides gives

$$\begin{bmatrix}0\\ \bar{\mathbf{F}}^i\end{bmatrix} = \tilde{\mathbf{q}}_d^* \otimes \begin{bmatrix}0\\ \bar{\mathbf{F}}^b\end{bmatrix} \otimes \tilde{\mathbf{q}}_d \tag{3.11}$$

where the unit vectors are defined as

$$\bar{\mathbf{F}}^i = \frac{\mathbf{F}^i}{\|\mathbf{F}^i\|} \tag{3.12}$$

$$\bar{\mathbf{F}}^b = \frac{\mathbf{F}^b}{\|\mathbf{F}^b\|} = \begin{bmatrix}0 & 0 & \pm 1\end{bmatrix}^T \tag{3.13}$$

and $\tilde{\mathbf{q}}_d$ is the desired quadrotor attitude (without accounting for the desired yaw angle) that aligns the body-frame thrust vector with the desired inertial-frame force vector. The minimum-angle quaternion rotation between the two unit vectors $\bar{\mathbf{F}}^i$ and $\bar{\mathbf{F}}^b$ in $\mathbb{R}^3$ is [40]

$$\tilde{\mathbf{q}}_d = \frac{1}{\sqrt{2(1 + \bar{\mathbf{F}}^{i^T}\bar{\mathbf{F}}^b)}}\begin{bmatrix}1 + \bar{\mathbf{F}}^{i^T}\bar{\mathbf{F}}^b\\ \bar{\mathbf{F}}^i \times \bar{\mathbf{F}}^b\end{bmatrix}. \tag{3.14}$$

The sign of the z-component of $\bar{\mathbf{F}}^b$ in Equation 3.13 is selected so that $\bar{\mathbf{F}}^{i^T}\bar{\mathbf{F}}^b \geq 0$, en-

suring that the direction of the body-frame thrust vector is aligned with the direction of the inertial-frame acceleration vector.

Equation 3.14 does not define a unique desired attitude for the vehicle. In particular, two ambiguities exist. First, quaternions double cover the special orthogonal group S0(3), meaning $\mathbf{q}$ and $-\mathbf{q}$ represent the same attitude [41]. In practice, this ambiguity is easily addressed by choosing the sign of $\tilde{\mathbf{q}}_d$ at the current time step to agree with the attitude commanded at the previous time step, such that $\tilde{\mathbf{q}}_d^T(t_k)\tilde{\mathbf{q}}_d(t_{k-1}) \geq 0$. Second, assuming the quadrotor is capable of producing negative thrust, an ambiguity exists between upright and inverted flight because the commanded global acceleration vector is the same in both cases. To fully disambiguate the desired attitude, an additional upright/inverted binary command variable, $\sigma_d(t) = \pm 1$, is needed, where 1 represents upright flight and -1 is inverted.

Finally, the desired vehicle attitude, $\mathbf{q}_d$, is computed by rotating $\tilde{\mathbf{q}}_d$ by the desired yaw angle, ψ_d, as

$$\mathbf{q}_d = \tilde{\mathbf{q}}_d \otimes \begin{bmatrix} \cos(\psi_d/2) & 0 & 0 & \sin(\psi_d/2) \end{bmatrix}^T. \tag{3.15}$$

The total quadrotor thrust, f_{total}, is computed as $f_{total} = \|\mathbf{F}^i\|$ since $\|\mathbf{F}^b\| = \|\mathbf{F}^i\|$ from Equation 3.10.

The desired quadrotor attitude rate is found by taking the time derivative of $\bar{\mathbf{F}}^i$ in the inertial frame. Utilizing the Transport Theorem [42], this derivative is

$$\frac{d}{dt}\begin{bmatrix} 0 \\ \bar{\mathbf{F}}^i \end{bmatrix} = \frac{d}{dt}\left(\mathbf{q}_d \otimes \begin{bmatrix} 0 \\ \bar{\mathbf{F}}^i \end{bmatrix} \otimes \mathbf{q}_d^*\right) + \begin{bmatrix} 0 \\ \boldsymbol{\Omega}_d^b \times \bar{\mathbf{F}}^i \end{bmatrix} \tag{3.16}$$

$$\dot{\bar{\mathbf{F}}}^i = \boldsymbol{\Omega}_d^b \times \bar{\mathbf{F}}^i \tag{3.17}$$

The first term on the right hand side of Equation 3.16 is zero since $\bar{\mathbf{F}}$ is constant in the body frame. Rearranging Equation 3.17 gives the desired body-frame angular rate vector projected onto the body-frame x-y plane.

$$\boldsymbol{\Omega}_{d_{xy}}^b = \bar{\mathbf{F}}^i \times \dot{\bar{\mathbf{F}}}^i \tag{3.18}$$

The third component of the angular velocity, the yaw rate, is directly computed from the input yaw command as

$$\mathbf{\Omega}^b_{d_z} = \dot{\psi}_d \tag{3.19}$$

The time derivative of $\bar{\mathbf{F}}^i$ is explicitly calculated using the quotient rule on Equation 3.12 as

$$\dot{\bar{\mathbf{F}}}^i = \frac{\dot{\mathbf{F}}^i}{\|\mathbf{F}^i\|} - \frac{\mathbf{F}^i(\mathbf{F}^{i^T}\dot{\mathbf{F}}^i)}{\|\mathbf{F}^i\|^3} \tag{3.20}$$

where $\dot{\mathbf{F}}^i = m\left(\dddot{\mathbf{r}}^i_d + \dddot{\mathbf{r}}^i_{fb}\right)$. In practice, $\dddot{\mathbf{r}}^i_{fb}$ is found by numerical differentiating $\ddot{\mathbf{r}}^i_{fb}$.

The calculations of desired attitude and attitude rate assume that $\|\mathbf{F}^i\| = \|\mathbf{F}^b\| \neq 0$, stemming from the fact that the attitude of the vehicle is irrelevant to the motion of the center of mass of the vehicle during free-fall because the motor net thrust is zero. However, the vehicle attitude is important as soon as the vehicle exits free-fall and so should be controlled the entire time. In practice, this attitude ambiguity is accounted for by ensuring that the reference trajectory does not command free-fall for a finite amount of time (the path only crosses or touches the singularity). In the controller, new desired attitude and attitude rates are computed only when $\|\mathbf{F}^i\|$ is above a small threshold, maintaining the previously commanded attitude and attitude rates while $\|\mathbf{F}^i\|$ is close to zero.

Utilizing the sequential rotation properties of quaternions [38], the desired vehicle attitude can be represented as a rotation from the inertial frame to the actual frame of the vehicle followed by a rotation from the vehicle frame to the desired vehicle orientation, as in

$$\underbrace{\mathbf{q}_d}_{\text{inertial frame}} = \underbrace{\mathbf{q}}_{\text{inertial frame}} \otimes \underbrace{\mathbf{q}_e}_{\text{body frame}} . \tag{3.21}$$

The quaternion $\mathbf{q}_e$ represents the error quaternion, or the attitude error of the vehicle expressed in the body frame. Note that in the special case of the actual and desired attitudes being equal ($\mathbf{q} = \pm\mathbf{q}_d$), the error quaternion is the identity quaternion ($\mathbf{q}_e = \begin{bmatrix}\pm 1 & 0 & 0 & 0\end{bmatrix}^T$). Rearranging Equation 3.21 using the conjugate properties of the quaternion yields the error quaternion, expressed in the body frame, as a simple

quaternion multiplication between the actual attitude and the desired attitude.

$$\underbrace{\mathbf{q}_e}_{\text{body frame}} = \underbrace{\mathbf{q}^*}_{\text{inertial frame}} \otimes \underbrace{\mathbf{q}_d}_{\text{inertial frame}} . \tag{3.22}$$

Equation 3.21 and 3.22 are similar to equations in previous work [43]; however, in this thesis, the order of the quaternion multiplication differs so as to agree with standard notation and the rotation operation introduced in Equation 3.1 [38].

With the error quaternion expressed in the body frame, the elements of the quaternion directly map to the required body-frame moments. Similar to other quaternion-based attitude control laws proposed [44–46], the attitude control is accomplished using proportional-derivative control on the attitude error and attitude rate error as

$$\mathbf{M}^b = -\operatorname{sgn}(q_e^0)\mathbf{K}_p\vec{q_e} - \mathbf{K}_d(\boldsymbol{\Omega}^b - \boldsymbol{\Omega}_d^b), \tag{3.23}$$

where q_e^0 and $\vec{q_e}$ are the scalar and vector portions of the error quaternion, respectively. The gain matrices, $\mathbf{K}_p$ and $\mathbf{K}_d$, are diagonal and positive definite. Given f_{total} and $\mathbf{M}^b$, the corresponding motor thrust commands are found by inverting the relationship in Equation 3.4.

3.3 Controller Stability Analysis

This section details an analysis of the stability of the acceleration control scheme developed in the previous section. The control law is shown to be asymptotically stable using Lyapunov's direct method. The analysis follows closely the framework of approximate-model-inversion based control (see e.g. [47, 48]) Let

$$\ddot{\mathbf{r}}^i = f(\mathbf{r}^i(t), \dot{\mathbf{r}}^i(t), \delta(t)) \tag{3.24}$$

describe the dynamics of the quadrotor where $\delta(t)$ are the inputs to the attitude controller. Since the true vehicle dynamics are not known, an approximate dynamic model $\hat{f}(\mathbf{r}^i(t), \dot{\mathbf{r}}^i(t), \delta(t))$ is assumed. This approximate dynamic model is invertible

with respect to $\delta(t)$ so that

$$\delta(t) = \hat{f}^{-1}(\mathbf{r}^i(t), \dot{\mathbf{r}}^i(t), \nu(t)) \tag{3.25}$$

where $\nu(t)$ is the control acceleration. Equations 3.6-3.20 detail the construction of the control acceleration and the inversion model $\hat{f}^{-1}$ that takes $\nu(t)$ and generates $\mathbf{q}_d$, $\mathbf{\Omega}_d^b$, and f_{total}, the inputs to the attitude controller.

Let the control acceleration be denoted as

$$\nu = -\mathbf{k}_p\mathbf{e} - \mathbf{k}_d\dot{\mathbf{e}} + \ddot{\mathbf{r}}_d^i - \nu_i(t) \tag{3.26}$$

where $\nu_i(t)$ is the integral portion of the feedback control acceleration. Note the time dependence of the variables is removed for clarity.

Let the modelling error between the true and approximate dynamics be

$$\Delta = f(\mathbf{r}^i(t), \dot{\mathbf{r}}^i(t), \delta(t)) - \hat{f}(\mathbf{r}^i(t), \dot{\mathbf{r}}^i(t), \delta(t)). \tag{3.27}$$

For simplicity, this modelling error is assumed constant so that $\dot{\Delta}(t) = 0$. This is a reasonable assumption even if the modelling error is slowly time varying, such as the trim throttle required to hover a quadrotor. Substituting this error term into Equation 3.24, the vehicle tracking error dynamics can be written as

$$\begin{aligned}\ddot{\mathbf{e}} &= \ddot{\mathbf{r}}^i - \ddot{\mathbf{r}}_d^i = \hat{f}(\mathbf{r}^i, \dot{\mathbf{r}}^i, \delta) + \Delta - \ddot{\mathbf{r}}_d^i \\ &= -\mathbf{k}_p\mathbf{e} - \mathbf{k}_d\dot{\mathbf{e}} + \ddot{\mathbf{r}}_d^i - \nu_i + \Delta - \ddot{\mathbf{r}}_d^i,\end{aligned}$$

or, in state space form as

$$\dot{\tilde{\mathbf{e}}} = \begin{bmatrix}\dot{\mathbf{e}} \\ \ddot{\mathbf{e}}\end{bmatrix} = \begin{bmatrix}\mathbf{0} & \mathbf{I} \\ -\mathbf{k}_p & -\mathbf{k}_d\end{bmatrix}\begin{bmatrix}\mathbf{e} \\ \dot{\mathbf{e}}\end{bmatrix} + \begin{bmatrix}\mathbf{0} \\ \mathbf{I}\end{bmatrix}(\Delta - \nu_i(t)) \tag{3.28}$$

$$\dot{\tilde{\mathbf{e}}} = A\tilde{\mathbf{e}} + B\left(\Delta - \nu_i(t)\right), \tag{3.29}$$

where $\tilde{e}$ is the tracking error. The control gains are selected so that A is a Hurwitz matrix. Thus, for any positive definite matrix Q, there exists a unique positive definite solution P to the Lyapunov equation

$$0 = A^T P + PA + Q. \tag{3.30}$$

Define the derivative of the integral control gain, $\nu_i(t)$, as $\dot{\nu}_i(t) = B^T P\tilde{\mathbf{e}}$ and let $\tilde{\nu}_i = \Delta - \nu_i$ be the error between the integral control term and the modelling error. The goal of the controller is to drive $\tilde{\nu}_i$ to zero.

A Lyapunov function candidate is constructed as

$$V = \frac{1}{2}\left[\tilde{\mathbf{e}}^T P\tilde{\mathbf{e}} + \tilde{v}_i^T\tilde{v}_i\right]. \tag{3.31}$$

Note that $V \geq 0$, and $V = 0$ only when $\tilde{\mathbf{e}} = \tilde{v}_i = 0$, making V a valid Lyapunov function candidate. The time derivative of Equation 3.31 is

$$\begin{aligned}
\dot{V} &= \frac{1}{2}\tilde{\mathbf{e}}^T P\dot{\tilde{\mathbf{e}}} + \frac{1}{2}\dot{\tilde{\mathbf{e}}}^T P\tilde{\mathbf{e}} + \tilde{v}_i^T\dot{\tilde{v}}_i \\
&= \frac{1}{2}\tilde{\mathbf{e}}^T P(A\tilde{\mathbf{e}} + B(\Delta - \nu_i)) + \frac{1}{2}(A\tilde{\mathbf{e}} + B(\Delta - \nu_i))^T P\tilde{\mathbf{e}} - \tilde{\nu}^T B^T P\tilde{\mathbf{e}} \\
&= -\frac{1}{2}\tilde{\mathbf{e}}^T Q\tilde{\mathbf{e}} \leq 0
\end{aligned}$$

Therefore, $V \leq 0$. Also, $\dot{V} = 0$ only when $\tilde{\mathbf{e}} = 0$. Thus, the Barbashin-Krasovskii-LaSalle theorem (see e.g. [49]) shows that the control law is asymptotically stable in the sense that $\tilde{\mathbf{e}} \to 0$ as $t \to \infty$.

Stability proofs of attitude control laws very similar to the one presented in Equation 3.23 (for the regulatory case) can be found in [44–46]. Therefore, both the acceleration and the attitude control loops can be shown to be stable independently. Due to the results established in [50] there exist therefore a set of gains and trajectory generation parameters that render the closed loop system stable. This intuitive result is reflected by the flight-tests results in Chapter 5.

3.4 Trajectory Generation

Given the control structure capable of tracking position and yaw reference commands developed in Section 3.2, consider the problem of navigating through n waypoints in 3-space in an obstacle-free environment. Similar to previous work [15, 39], a trajectory consisting of piecewise smooth polynomials of order m over $n-1$ time intervals is proposed. Using this formulation, the trajectory of the quadrotor is defined by

$$\mathbf{r}_d^i(t) = \begin{cases} \sum_{i=0}^{m} \alpha_{i,1} t^i & 0 \leq t < t_1 \\ \sum_{i=0}^{m} \alpha_{i,2} t^i & t_1 \leq t < t_2 \\ \vdots & \vdots \\ \sum_{i=0}^{m} \alpha_{i,n-1} t^i & t_{n-2} \leq t \leq t_{n-1} \end{cases}$$

where $\alpha_{i,n}$ is the i^{th} polynomial coefficient over the n^{th} time interval. Formulating the desired reference path as a series of polynomials offers several advantages. First, given the correct number of endpoint constraints at the segment boundaries and the corresponding segment times, a closed-form solution for finding the polynomial coefficients exists. Second, constraints on the velocity, attitude, and attitude rate of the quadrotor at any of the intermediate waypoints are easily incorporated in the path as constraints at the segment boundaries. Adding attitude constraints is discussed in more detail in Section 3.4.2. Third, polynomials for each of the four flat outputs, $\mathbf{r}_{d_x}^i(t)$, $\mathbf{r}_{d_y}^i(t)$, $\mathbf{r}_{d_z}^i(t)$, and $\psi_d(t)$ can be solved for separately using the same segment times. Finally, provided the boundary conditions ensure the continuity of at least the first four derivatives of the reference path, the quadrotor reference input commands (functions of the first three derivatives of position) will be smooth.

As an example, consider the x-dimension of a two waypoint problem, where the vehicle starts and stops in hover. As described in Section 3.2, the inputs to the quadrotor are computed as a function of the first three derivatives of the position command. To ensure that those inputs are smooth, the initial and final first four

derivatives of position are constrained as

$$\mathbf{r}^i_{d_x}(0) = x_0 \qquad \mathbf{r}^i_{d_x}(t_f) = x_f \tag{3.32}$$

$$\mathbf{r}^{i^{(k)}}_{d_x}(0) = 0 \qquad \mathbf{r}^{i^{(k)}}_{d_x}(t_f) = 0 \qquad k = 1, \ldots, 4 \tag{3.33}$$

where the superscript in parentheses represents the k^{th} time derivative of x. The formulation results in 10 constraints, 5 initial and 5 terminal conditions. Therefore, assuming the final time, t_f, is known, a 9^{th} order polynomial offers a closed-form solution to the problem.

Next, consider the same initial and final conditions, but now with $n-2$ intermediate waypoints that the trajectory must pass through. Assuming a desired arrival time associated with each waypoint is known, the problem maintains a closed-form solution as long as there are $10n - 10$ constraints. Constraining the position and first four derivatives of position at each waypoint provides the required number of constraints; however, this requires knowledge of the velocity, acceleration, jerk, and snap of the quadrotor at each waypoint. Alternatively, if only the position of the waypoint is important, the remaining $8(n-2)$ constraints are formed by ensuring continuity of the first 8 derivatives of position at the $n-2$ intermediate waypoints.

Example boundary conditions with waypoints $\mathbf{w} = [w_0, w_1, \ldots, w_{n-1}]$ are given by

$$\text{initial} = \begin{cases} \mathbf{r}^i_d(0) = w_0 & \\ \mathbf{r}^{i^{(k)}}_d(0) = 0 & k = 1, \ldots, 4 \end{cases}$$

$$\text{middle} = \begin{cases} \mathbf{r}^i_d(t_r^+) = w_r & r = 1, \ldots, n-2 \\ \mathbf{r}^i_d(t_r^-) = w_r & r = 1, \ldots, n-2 \\ \mathbf{r}^{i^{(k)}}_d(t_r^+) - \mathbf{r}^{i^{(k)}}_d(t_r^-) = 0 & k = 1, \ldots, 8 \end{cases}$$

$$\text{final} = \begin{cases} \mathbf{r}^i_d(t_{n-1}) = w_{n-1} & \\ \mathbf{r}^{i^{(k)}}_d(t_{n-1}) = 0 & k = 1, \ldots, 4. \end{cases}$$

As the constraints are formulated, the path starts and ends at hover and is required to pass through each of the waypoints.

Note that the formulation offers flexibility by allowing any of the first four derivatives of position to be user-specified at any of the intermediate waypoints. For instance, if the desired x component of velocity at waypoint j is v_j, the constraint becomes $\mathbf{r}^i_{d_x}(t_j^-)^{(1)} = \mathbf{r}^i_{d_x}(t_j^+)^{(1)} = v_j$. When the velocity is not specified, the constraint is $\mathbf{r}^i_{d_x}(t_j^-)^{(1)} - \mathbf{r}^i_{d_x}(t_j^+)^{(1)} = 0$. Constraining any of the derivatives of an intermediate waypoint to a known value is accomplished by removing one the higher-order continuity constraints at that waypoint. As long as the waypoint time and the initial and final conditions are specified, the solution for the desired trajectory and all its derivatives is closed-form and consists of a single matrix inversion. Care must be taken, however, when specifying several constraints at a single node of the polynomial. Position, its derivatives, and time are highly coupled and radical solutions to the polynomial formulation can be found when the constraints are not chosen properly. The following section proposes a method for ensuring the resulting paths are reasonable.

3.4.1 Actuator-Constrained Minimum-Time Trajectory Generation

While the preceding closed-form polynomial trajectory generation method ensures that all the reference commands to the quadrotor will be smooth, there is no guarantee that the commands will be within the feasible limits of the hardware actuators. For instance, any trajectory of non-zero length will become infeasible as the segment times approach zero because the corresponding velocity, acceleration, and attitude rate reference commands will approach infinity. This section presents an optimization method for finding the minimum segment times while not exceeding the physical constraints of the quadrotor.

The optimization returns the segment times that minimize the total path time subject to the motor saturation constraints in Equation 3.5. The optimization over

n waypoints with $\mathbf{t} = [\ t_1 \quad t_2 \quad \dots \quad t_{n-1}\ ]$ segment times is formulated as

$$\mathbf{t} = \underset{\mathbf{t}}{\operatorname{argmin}} \quad t_{n-1} \tag{3.34}$$

$$\text{subject to} \quad f_{\min} \leq f_i \leq f_{\max} \quad i = 1, \dots, 4 \tag{3.35}$$

$$t_j > 0 \qquad\qquad j = 1, 2, \dots, n-1 \tag{3.36}$$

The trajectory starts at the first waypoint with $t_0 = 0$. The decision variables $\mathbf{t}$ are the times at which the quadrotor passes through the $n-1$ remaining waypoints. Minimizing the last decision variable minimizes the total time of the trajectory since each segment time is constrained to be positive. A path is defined as feasible when none of the motor commands exceed the allowable motor thrust values. The calculation of these motor constraints is detailed below.

During each iteration of the optimizer, the reference path is calculated by solving the closed-form polynomial formulation for the coefficients $\alpha_{i,n}$ as specified above using the current value of $\mathbf{t}$. The equations of motion of the quadrotor (Eqs. 3.2-3.3) are then inverted using the computed path as the reference command, returning the required forces and moments to fly that path. (Note that this inversion requires knowing the angular acceleration of the vehicle. This calculation is detailed in Appendix B.) The individual motor thrust values are found by inverting the relationship in Equation 3.4. The calculated motor thrust values are only an approximation of the true thrust values commanded during flight due to errors in estimated model parameters (mass and inertia) and errors from ignoring the feedback control in Eqs. 3.6 and 3.23 (inverting the equations of motion using the reference path as the input assumes the quadrotor never deviates from the reference path). While the resulting segment times found from the optimization cannot guarantee that the commanded motor thrusts will never exceed the prescribed bounds, in practice $f_{\max}$ and $f_{\min}$ can be treated as tuning gains; decreasing the allowable thrust window for each motor decreases the overall aggressiveness of the resulting paths.

3.4.2 Attitude Constraints

Specific attitude constraints can be incorporated into the desired path formulation by constraining the acceleration of the vehicle based on Equation 3.10. Given a desired inertial-frame attitude $\mathbf{q}_{des}$ the corresponding required inertial-frame acceleration $\ddot{\mathbf{r}}^i_{att}$ is computed, up to an overall scale factor of the thrust magnitude, by solving

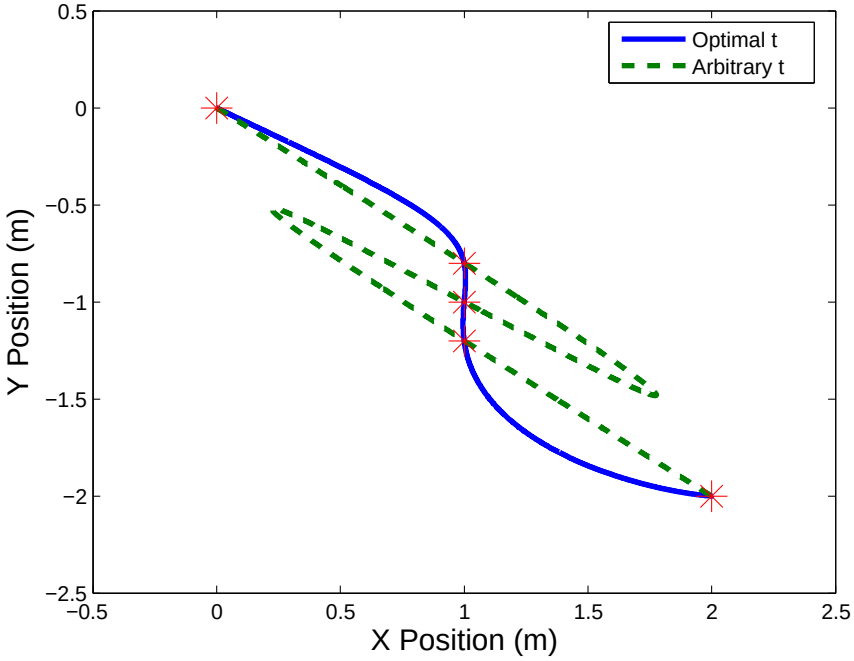
$$\begin{bmatrix} 0 \\ \ddot{\mathbf{r}}^i_{att} \end{bmatrix} = \frac{\|\mathbf{F}^b\|}{m} \mathbf{q}^*_{des} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \mathbf{q}_{des} - \begin{bmatrix} 0 \\ \mathbf{g}^i \end{bmatrix} \tag{3.37}$$

where $\|\mathbf{F}^i\|$ is chosen to scale the acceleration as desired. Equation 3.37 allows the user to specify the attitude of the vehicle at polynomial nodes in the path. While the vehicle attitude between nodes is not directly specifiable with the current algorithm, guaranteeing the vehicle attitude at a certain point in space can be beneficial for maneuvers such as flying through windows or performing aerobatics.

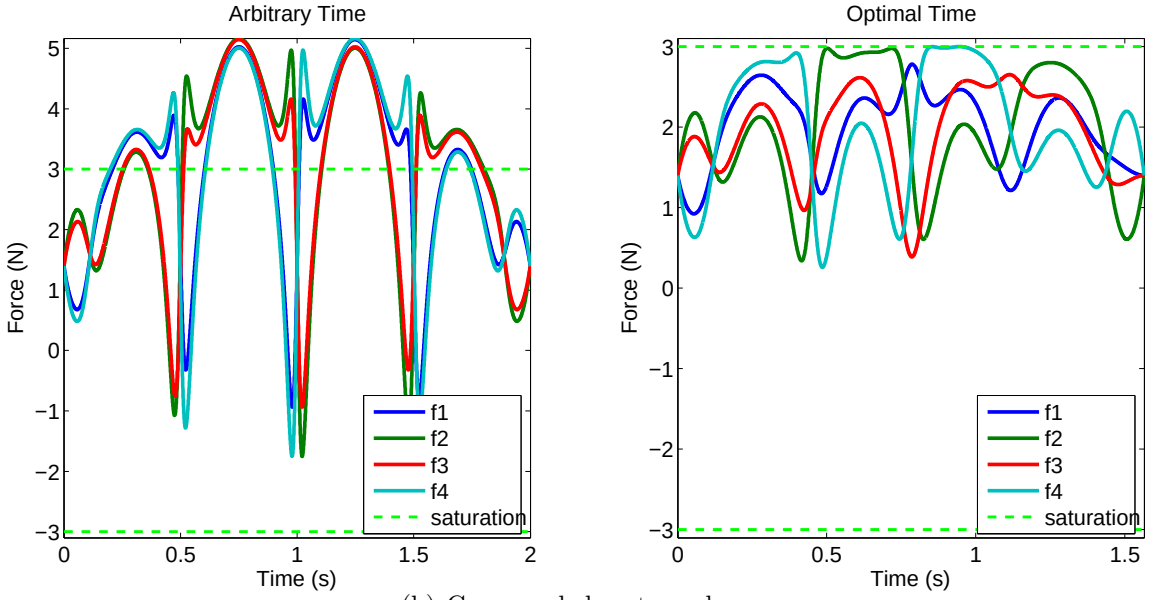
3.5 Trajectory Generation Simulations

An example path found with the optimization scheme from Section 3.4.1 is shown in Figure 3-2. The path is constrained to start and stop in hover and to pass through three intermediate waypoints, indicated in the plot by red stars. Initially, the time between each waypoint is arbitrarily set to be 0.5 seconds. This yields a strangely oscillatory path with motor commands that saturate significantly. After running the optimization routine, the motor commands are within the saturation bounds and the resulting path appears more reasonable. The aggressiveness of the path is easily tunable by lowering the saturation bounds of the actuators. Decreasing the saturation bounds will increase the flight time but decrease the actuator load.

In Figure 3-3 the trajectory starts at the origin at hover and ends at hover one meter upwards. Bench testing of the motors and propellers used on the variable-pitch



(a) Example path



(b) Commanded motor values

Figure 3-2: An example path showing the minimum time optimization. Both paths satisfy the constraints of starting and ending at hover and passing through the five waypoints; however, the optimal time path keeps the motor commands from saturating and completes the path in less time than the one with arbitrary waypoint arrival times.

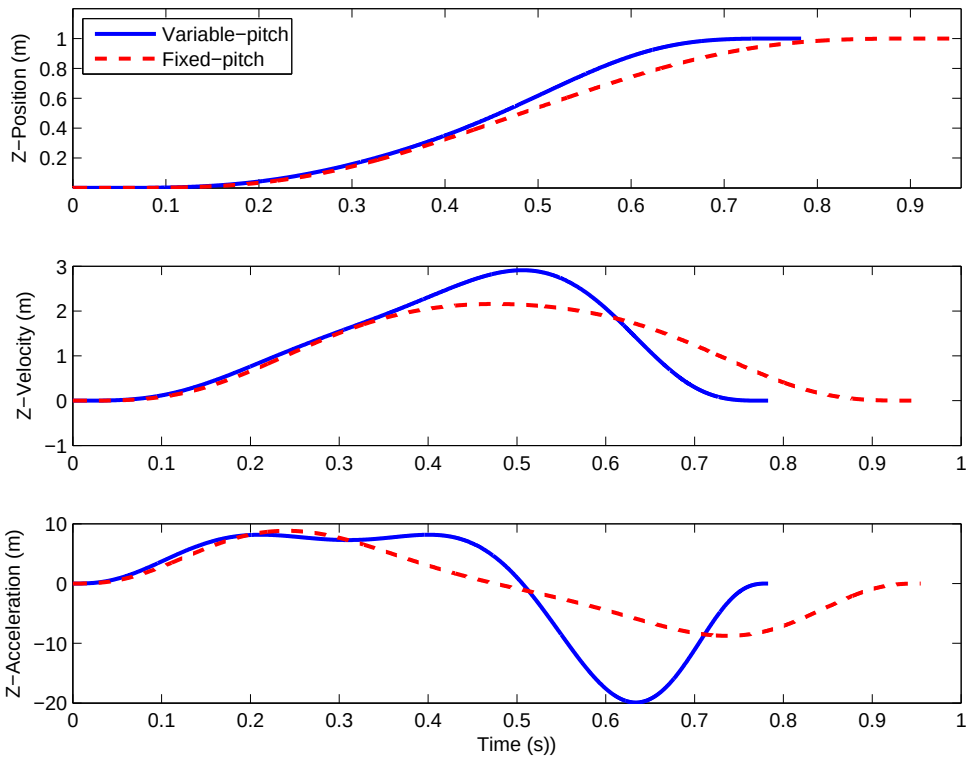
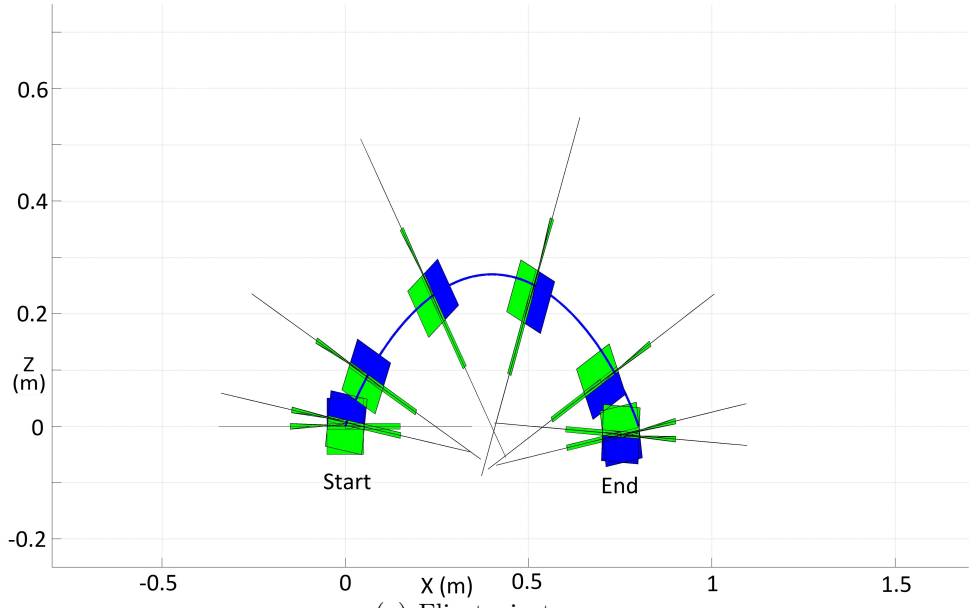


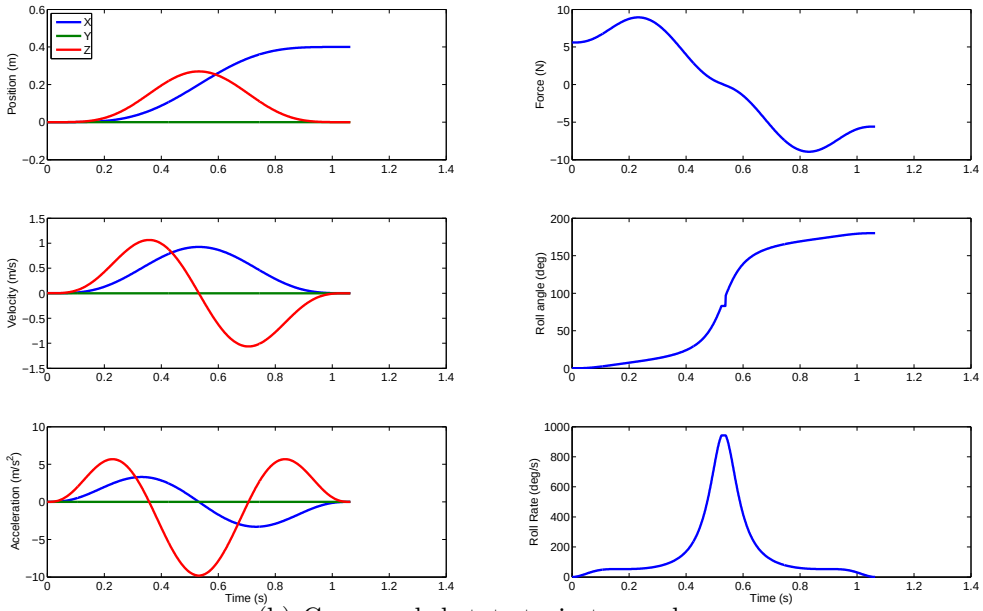
Figure 3-3: Two example vertical flight trajectories computed using the optimization routine in Section 3.4.1. Both trajectories have the same upper bound on motor thrust. The variable-pitch trajectory has a negative thrust lower bound, but the fixed-pitch trajectory has a lower bound of near zero. Note that the variable-pitch trajectory is shorter because it decelerates faster than gravity.

quadrotor show maximum and minimum possible thrust values of about 3 N and -3 N per motor, respectively. When the pitch is locked to a positive value (simulating a fixed-pitch propeller), the minimum thrust value increases to about 0.15 N. Figure 3-3 shows how the increased negative range of the variable-pitch propellers allows the quadrotor to decelerate faster than gravity, decreasing the overall feasible trajectory time.

As mentioned in Section 3.2, the attitude is not well defined from Equation 3.14 when $\|\mathbf{F}^b\| = 0$ (the vehicle is in free-fall). However, interesting attitude maneuvers can be constructed by imposing an instantaneous free-fall constraint. In particular, Figure 3-4 shows the trajectory generated by imposing an acceleration constraint of $-\mathbf{g}^i$ between two hover conditions at different locations along the x-axis. The quadrotor goes inverted after the instantaneous free-fall because $\sigma(t)$ is changed from



(a) Flip trajectory



(b) Commanded state trajectory values

Figure 3-4: Trajectory generated by imposing a position free free-fall acceleration condition between two hover waypoints along the x-axis. The small corner in the commanded attitude trajectory comes from not computing new commanded attitudes when the total force command is close to zero. The vehicle goes inverted at the apex of the trajectory by explicitly changing $\sigma(t)$ from 1 to -1.

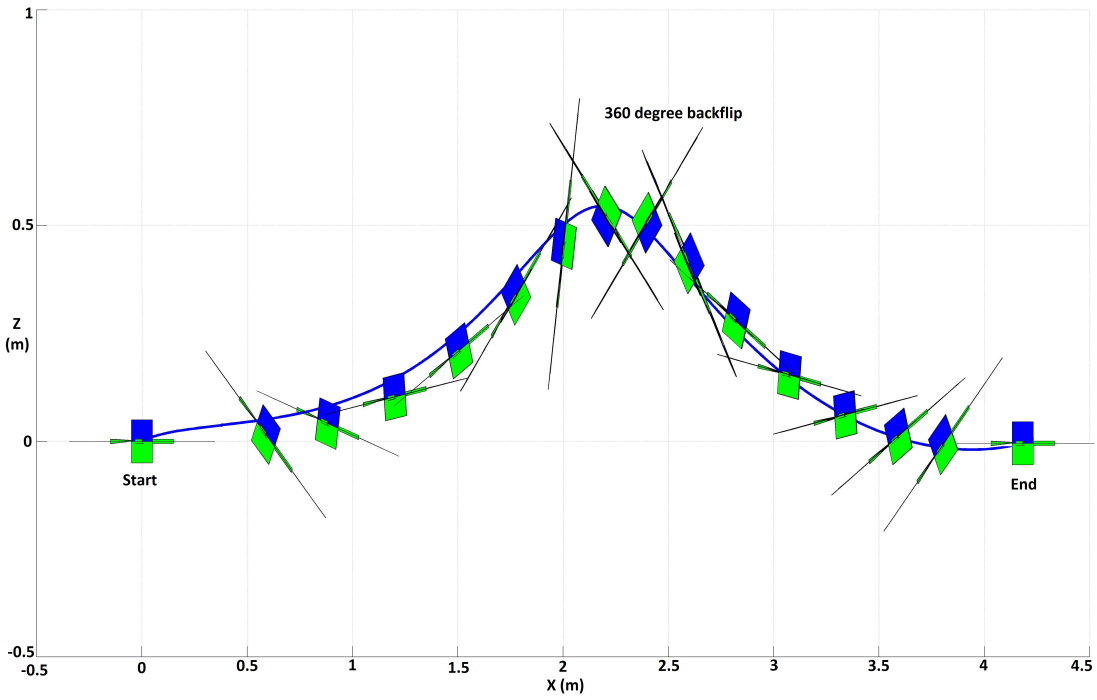
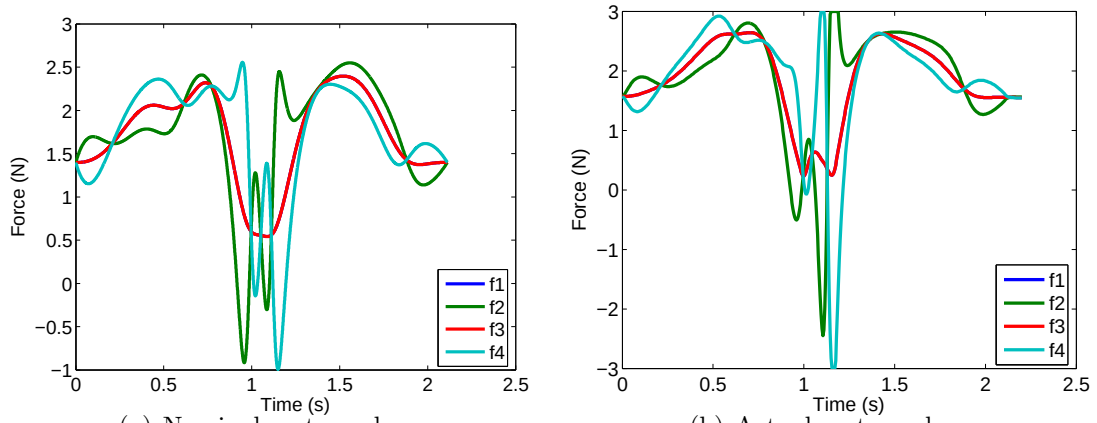


Figure 3-5: Simulation results of a 360 degree backflip. The flip is specified using a -90 degree roll constraint before the peak of the trajectory and a 90 degree roll constraint after the peak. The quadrotor starts and ends in hover.



(a) Nominal motor values (b) Actual motor values

Figure 3-6: Example motor data from the backflip presented in Figure 3-5. Figure 3-6(a) shows the anticipated motor commands assuming open-loop, perfect tracking. These are the commands used by the optimizer in Section 3.4.1 to find minimum-time trajectories. Figure 3-6(b) shows the corresponding actual motor commands when following the trajectory in simulation.

1 to -1 at that point.

Attitude constraints embedded in the path formulation are utilized to command a path similar to the backflip demonstrated on the Stanford STARMAC quadrotor [12]. Simulation results of the path are presented in Figures 3-5 and 3-6. The flipping motion is prescribed by embedding a -90 degree roll constraint just before the apex of the path and a 90 degree roll constraint just after the apex. Figure 3-6 shows how the ideal motor commands compare to those actually generated in the simulation.

3.6 Summary

This chapter presents a control law capable of tracking reference position trajectories that are smooth through the third derivative. The controller is also capable of controlling attitudes that vary significantly from hover. An algorithm is presented that generates time-optimal trajectories in $\mathbb{R}^3$ through an arbitrary number of waypoints subject to actuator saturation constraints. In addition, attitude-specific constraints are easily embedded in the commanded reference path, allowing for aerobatic maneuvers. The control and trajectory generation algorithms were implemented in simulation. These results are validated in on a custom variable-pitch quadrotor in Chapter 5.

Chapter 4

Hardware and Software Implementation

Obtaining successful flight results with the variable-pitch quadrotor requires both accurate control algorithms and robust physical hardware. This chapter addresses the technical challenges inherent with the physical construction of the variable-pitch quadrotor and the software required for agile flight. Also discussed is the simulation environment used to test trajectories and design controllers for the quadrotor.

4.1 Hardware Development

The variable-pitch quadrotor was designed and built entirely at MIT's ACL using a combination of custom and off-the-shelf components. A closeup of the most current version of the vehicle is shown in Figure 4-1.

The frame is cut from a sheet of carbon fiber sandwich material with a 1/4 inch balsa wood core. The frame is custom designed for the motors, speed controllers, servos, and electronics used on the vehicle. In particular, it is designed to minimize vibrations induced by the propellers, motors, and servos. Throughout the design process most of the hardware decisions made have been to reduce the amount of vibration experienced by the vehicle during flight. In addition to weakening mechanical connections and shortening the life of motor and servo bearings, strong vibrations

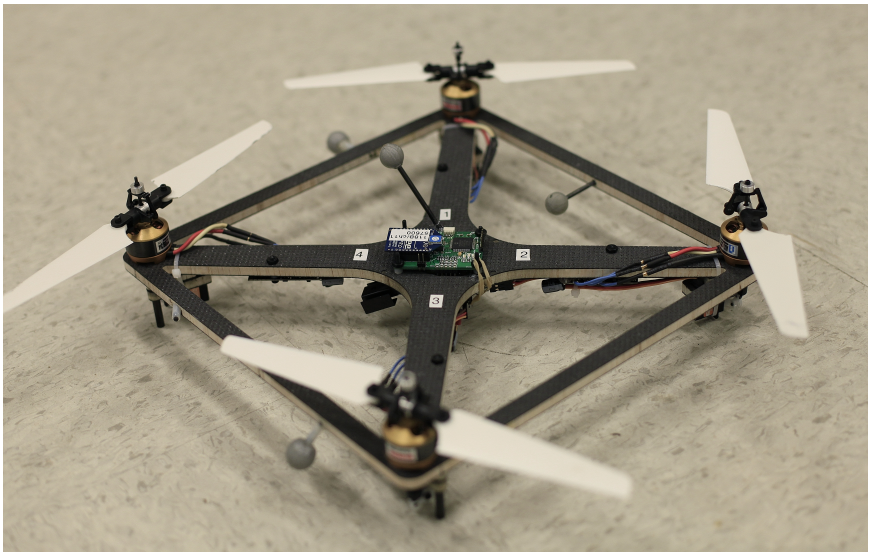


Figure 4-1: Variable-pitch quadrotor designed at the Aerospace Controls Lab. The servos that actuate the variable-pitch propellers are visible under each of the motors. The quadrotor measures 0.35 m across.

can distort the readings of on-board flight sensors such as rate gyros and accelerometers. The issue of vibrations in multi-copters is a well-understood and frequently discussed topic among hobby enthusiasts. Many of the hardware design choices on the variable-pitch quadrotor are inspired by hobbyists.

The original version of the variable-pitch quadrotor consisted of a single motor mounted at the center of the vehicle frame. Four timing belts transferred the motor power to the four rotors, with each belt running down one of the four quadrotor arms. The design was similar to the variable-pitch quadrotor developed in the Unmanned Vehicle Control Systems Lab at the National Cheng Kung University (NCKU) in Taiwan [21]. While a single motor design offers potential benefits, our experience is that the mechanical complexity of transferring rotary motion from a central motor to four propellers introduces significant vibrations to the vehicle. The team from NCKU saw these same problems stemming from vibrations and, as a result, were limited in the achievable gain magnitude of their PID control laws, ultimately resulting in relatively "loose" autonomous control.

The next iteration of the variable-pitch quadrotor featured four individual motors directly driving four propellers. As with the first verions, the propeller pitch was

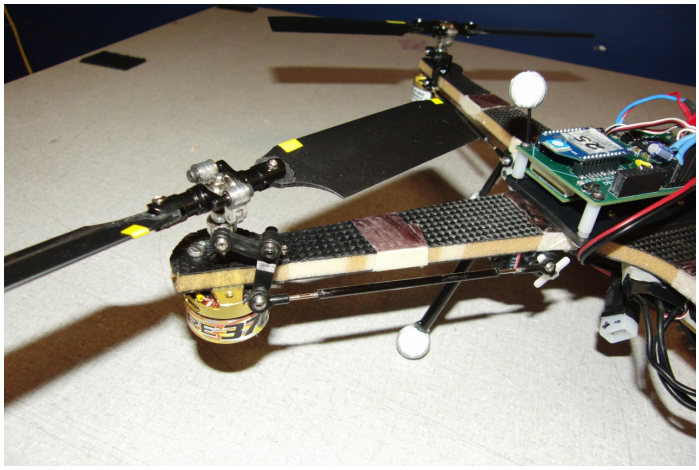
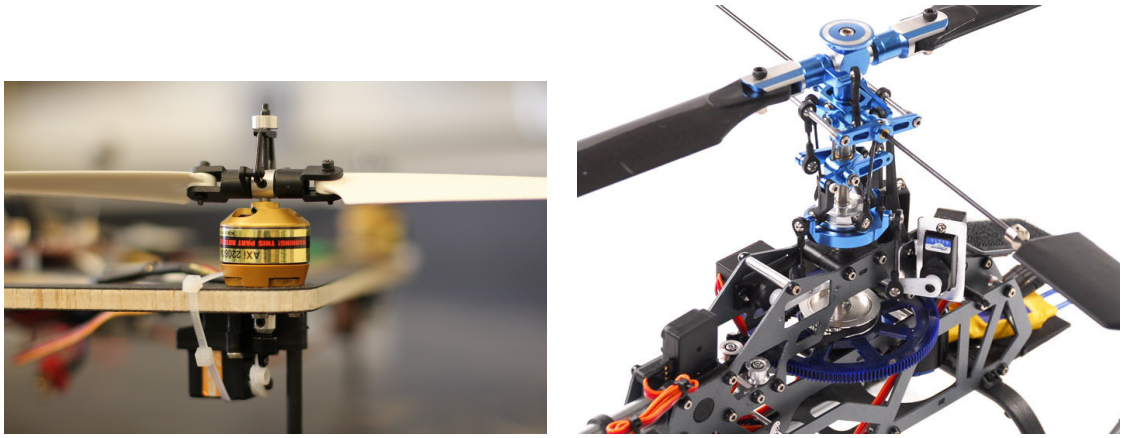


Figure 4-2: The second version of the variable-pitch quadrotor. The propeller pitch is actuated using a sliding control horn designed for RC model helicopter tail rotors.

actuated using a right-angle control horn slider connected via a pushrod to a servo mounted near the center of gravity of the quadrotor. A close-up of the variable-pitch mechanism on this version of the quadrotor is shown in Figure 4-2. Moving from one large, centrally-mounted motor to four smaller motors greatly reduced the vibrations during flight, allowing the quadrotor to hover and follow waypoints in both upright and inverted orientations. However, vibrations continued to be an issue and the vehicle never achieved the level of tight, robust control within the flight regime of similarly-sized fixed-pitch quadrotors [2, 15, 25].

In particular, vibrations came from inevitable slop in the pitch control mechanism and from propellers being improperly balanced. Of these two sources, the later was by far the most prevalent. Despite rigorously balancing each of the pairs of rotors, the quadrotor still routinely experienced vibrations strong enough to bend the stainless steel motor shafts. While never completely verified, the vibrations are mostly attributed to the significant distance between the motor and the propeller, creating a relatively large moment arm over which small disturbances from the propeller are magnified to create large torques about the quadrotor arms. As seen in Figure 4-2, the long propeller shaft is required due to the nature of the pitch control mechanisms.

Finally, robust, stable flight was achieved by replacing the RC helicopter tail rotor pitch control mechanisms with ones designed for RC airplane propellers. The new



(a) Variable-pitch actuator (b) RC helicopter swashplate

Figure 4-3: On the left is one of the pitch actuation mechanisms on the current version of the variable-pitch quadrotor. The servo, mounted below the motor, actuates the propeller pitch via a carbon fiber pushrod routed through a hollow shaft in the motor. The figure on the right shows a typical swashplate on a RC helicopter. The swashplate allows the helicopter to rotate the thrust vector with respect to the body frame, allowing the vehicle to move forwards, backwards, right and left. The swashplate, however, is mechanically much more complicated than the variable-pitch actuators used in this project.

pitch actuators use a carbon fiber pushrod routed through the center of a hollow shaft motor. The servo is mounted beneath the motor and the propeller is attached directly next to the motor can, minimizing the adverse vibrational effects of improperly balanced propellers. The pitch control mechanism is shown in Figure 4-3(a). There are several similar commercially available variable-pitch actuators built for RC airplanes. After testing several of them, the EVPU from MS Composit [51] was determined to have the least pitch slop and most robust design. Note that swashplates, needed on single-rotor helicopters to achieve agile flight (an example swashplate on a RC helicopter is shown in Figure 4-3(b)), are much more mechanically complex than the variable-pitch mechanism used on the quadrotor in this project.

Additionally, vibrations were further reduced by adding more support to the carbon fiber frame in the form of cross members between the arms. The cross members significantly stiffen the corners of the quadrotor where the motors are attached and hence dampen motor vibrations in the entire frame.

Other than the vibrational issues discussed above, the addition of variable-pitch

Table 4.1: Overall weight of the various components of the variable-pitch quadrotor. The servos and variable-pitch actuators make up about 15% of the overall weight of the quadrotor.

Component	Type	Qty	Unit Weight (gram)	Total Weight (gram)
Frame	Custom	1	90	90
Motor	Axi 2208/34 EVP	4	45	180
ESC	Mikrokopter	4	8.5	34
Servo	MKS DS480	4	10	40
Propellers	MS Composit EVPU	4	8	32
Variable-pitch actuator	MS Composit EVPU	4	8	32
Autopilot	Custom	1	14	14
Power distribution board	Custom	1	20	20
Wiring, connectors, etc.	N/A	N/A	20	20
Vehicle Weight			462	

propellers to the quadrotor has few adverse affects when compared to the mechanical simplicity of traditional fixed-pitch quadrotors. As shown in Table 4.1, the servos and variable-pitch components only make up about 15% of the total weight of the vehicle.

One interesting observation of the quadrotor construction and flight testing was the fact that the quadrotor has slightly better hover performance (slightly smoother attitude control) when flying in fixed-pitch mode as opposed to variable-pitch mode. One potential reason for this difference in flight quality was mentioned in Section 2.1.5. The update rate of the motor controllers is significantly faster than the update rate of the servos controlling the propeller pitch. Other possible reasons might include slop in the pitch actuator and actuator command discretization.

While the pitch control mechanism used on the variable-pitch quadrotor is one of the best that are commercially available, it is designed for RC airplanes. Therefore, the mechanism is not intended to handle precise, fast changes in pitch. There is a small amount of slop in the pitch of the blades that results from the mechanical linkage between the servo and the propellers. Since the quadrotor is inherently unstable, hovering flight requires small, fast thrust corrections to maintain stability which might be more difficult to accomplish by adjusting propeller pitch as opposed to motor speed.

Also, both the motor speed controller and the propeller servo accept commanded

values between 0 and 250. Since the motor only spins in one direction but the propeller must cover both the positive and negative pitch ranges, the motor effectively has twice as much control resolution as compared with the servo. This additional resolution may help the quadrotor hover better when the commands are coming from the motor as opposed to the servo.

The variable-pitch quadrotor represents a relatively small increase in both hardware complexity and added weight when compared to a fixed-pitch quadrotor; however, the addition of variable-pitch propellers greatly extends the flight envelope, allowing for more aggressive and agile flight than possible with fixed-pitch quadrotors. Flight results and details are shown in Chapter 5.

4.2 Software

The variable-pitch quadrotor is controlled via three main levels of software using successive loop closure, similar to what is described in Chapter 6 of [52]. The two main loops closed are the attitude control loop (commonly referred to as the inner loop) and the acceleration loop (commonly referred to as the outer loop), with reference inputs to the acceleration loop coming from a trajectory generator. An overview of the software and data flow is shown in Figure 4-4. The main control algorithms implemented in each of these three software levels is described in Chapter 3. This section describes some of the software implementation details.

4.2.1 Attitude Control

The attitude control loop is the lowest level of control on the variable-pitch quadrotor. The attitude control accepts as inputs a desired quaternion representing the desired attitude of the vehicle, desired angular rates about the three principle body axes, and a total thrust value. Motor and pitch setting commands are then set to match the actual attitude and attitude rates to the commanded values using the algorithms described in Section 3.2. The attitude control loop is run entirely on-board the embedded processor on the quadrotor.

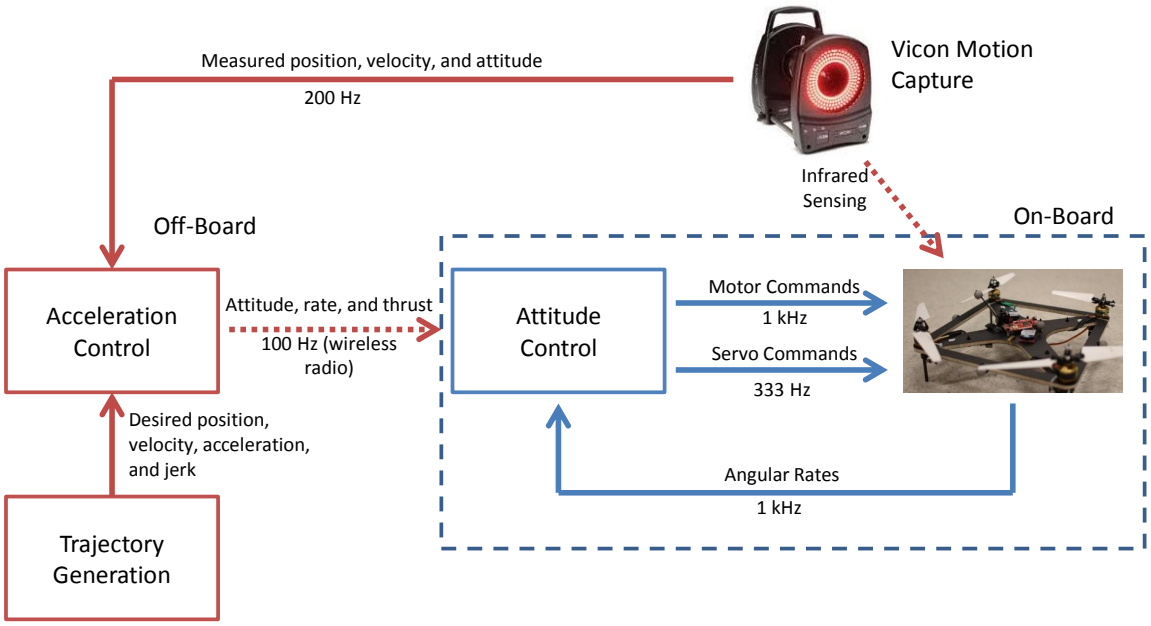
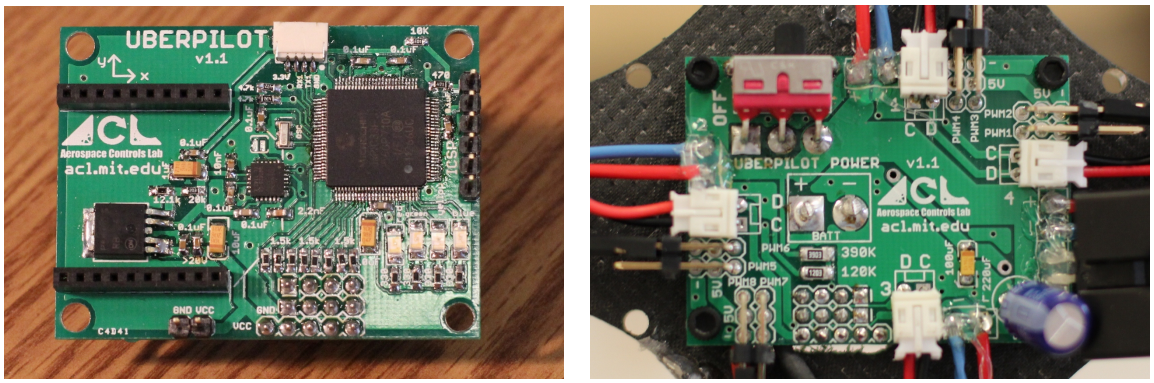


Figure 4-4: An overview of the software and data flow for the variable-pitch quadrotor. The on-board code is run on the UberPilot while the other loops run on a PC. All the communication except for via the wireless radio is handled by the Robot Operating System (ROS).

As mentioned in Section 4.1, mechanical vibrations in multi-rotor helicopters quickly degrade flight performance since on-board sensors (notably the rate gyros) are typically very sensitive to vibration. In addition to the hardware improvements made to limit the source and extent of the vibration, significant improvement in the flight performance was attained through the use of vibration-resistant rate gyros.

Earlier versions of the variable-pitch quadrotor performed attitude control using a 12-gram autopilot developed by Unmanned Innovation, Inc. [53]. Eventually, more control over the on-board attitude loop and communication code was desired and a custom autopilot, nick-named the "UberPilot," was developed to fly the quadrotor using custom software developed at MIT's ACL. The UberPilot circuit design is loosely based on the UAV Development Board version 4 from Sparkfun [54] and features a 10 gram (14 grams with wireless radio), 1.5 x 2.0 inch autopilot board. The autopilot board plugs into a 20 gram, similarly-sized power distribution board that distributes power and signal lines from the battery and the autopilot to the four motors and servos. The autopilot and power distribution boards are shown in Figure 4-5.



(a) UberPilot control board (b) UberPilot power distribution board

Figure 4-5: Custom electronics used to perform attitude estimation and control on the variable-pitch quadrotor. The control board (left) mounts on top of the quadrotor and houses a 16-bit microcontroller, 3-axis rate gyro, and wireless radio. The power distribution board (right) mounts beneath the control board and distributes power from the battery and signal lines to the electronic speed controllers and servos.

The UberPilot utilizes a 16-bit dsPIC33F microcontroller from Microchip Technologies Inc. [55] running at 50 MHz and a single chip, 3-axis rate gyro (ITG-3200) from InvenSense [56]. The ITG-3200 is designed to be vibration resistant and can measure rotations up to 2000 degrees per second. Communication from the UberPilot to the ground station is performed using a 2.4 GHz xBee wireless radio transmitting serial data at 57.6 kilobits per second.

Estimates of the vehicle attitude are computed on the UberPilot by integrating the rate gyro measurements. Drift in the gyro measurements and the corresponding degradation of the attitude estimates is accounted for by correcting the on-board attitude estimates with attitude measurements from an external Vicon motion capture system [57]. These external attitude estimates are sent via the wireless radio to the on-board processor and are added to the internal attitude estimates using a simple complimentary filter as

$$\mathbf{q}_o[\mathrm{i}+1] = \mathbf{q}_o[\mathrm{i}] + k\left(\mathbf{q}_e[\mathrm{i}+1] - \mathbf{q}_o[\mathrm{i}]\right) \tag{4.1}$$

where $\mathbf{q}_o$ is the on-board quaternion estimate, $\mathbf{q}_e$ is the external quaternion measurement, i is the discrete time index, and k is a gain such that the time constant of the

first order system is on the order of 5-10 seconds.

The Uberpilot samples the rate gyros at 1 kHz and computes an attitude solution at the same rate. The motor and servo commands are sent to their respective hardware actuators at 1 kHz, although the servo only responds to commands at 333 Hz. The majority of the control and estimation code is implemented using fixed-point math since the dsPIC33F does not have a floating point hardware unit. In future upgrades, the autopilot will probably use a faster processor that includes floating point hardware support. This will allow more of the computation to be done on-board and to for the autopilot to utilize more sensors, ultimately moving away from Vicon aided flight.

As mentioned above, the main interface between the autopilot and the off-board computer is via an xBee radio. Experiments have found the performance of these radios quite poor when they are tasked with high speed bi-directional communication. Therefore, rather than transmit back pertinent sensor data during vehicle flight to be logged on the off-board computer, the autopilot uses a second serial port to directly log data to a mini SD card. Flight data is logged via an OpenLog data logger from Sparkfun [54] at 115.2 kilobits per second. The data can then be easily imported into a computer and analyzed.

4.2.2 Acceleration Control

The primary roll of the acceleration control loop, run on an off-board PC, is to generate the desired attitude, angular rate, motor, and pitch commands needed to stabilize and track reference position, velocity, and acceleration commands. The controller uses reference values from the trajectory generator and measured values from the motion capture system to compute a desired linear acceleration and jerk in each of the three Cartesian coordinates. These desired accelerations and jerks are then mapped into desired attitudes, angular rates, and a desired total force as described in Section 3.2. The desired values are sent to the quadrotor over the wireless radio at 100 Hz (the rate is limited by the speed of the radio).

The acceleration control loop is designed around the assumption that the attitude

dynamics are "fast" relative to the position, velocity, and acceleration dynamics. Essentially, the outer control loop assumes that the desired attitude and angular rates sent to the quadrotor are instantaneously achieved. In practice, this assumption is well validated as the attitude and angular rates track the reference inputs well. Example attitude and attitude rate data is presented in Section 5.4.

The desired total force as calculated in this loop is first computed in Newtons, based on the mass of the vehicle and the desired accelerations. The force in Newtons is then mapped to the non-dimensional motor and pitch settings (inputs to the electronic speed controllers and servos) by fitting a function to experimental thrust data.

The software implementation of the acceleration control loop is greatly simplified through strong utilization of the Robot Operating System (ROS) [58]. ROS is a free and open source software package for robotics development and implementation. It handles all of the needed message passing between the various software components, with the exception of the wireless communication with the quadrotor which is handled with custom code. This is a major departure from the infrastructure discussed in [1, 2] where most, if not all, of the communication and code infrastructure was handled with custom software. As indicated in Figure 4-4, the information from the motion capture system is broadcast as ROS messages over an internal network. This enables any other PC on the network to subscribe to the messages and obtain real-time state information. ROS brings further benefits with the ability to plot, log, and replay data.

4.2.3 Trajectory Generation

The trajectory generation code creates the reference position, velocity, acceleration, and jerk commands that are sent to the acceleration control loop. The code is written in MATLAB and utilizes a ROS-MATLAB bridge to send commands in real-time to the other control loops. Since the trajectories being generated are time-parametrized polynomials, the trajectory generator sends desired states based solely on the initial position of the quadrotor and the current time. In practice, this strategy works quite well with robust and repeatable flight performance attainable. However, further work

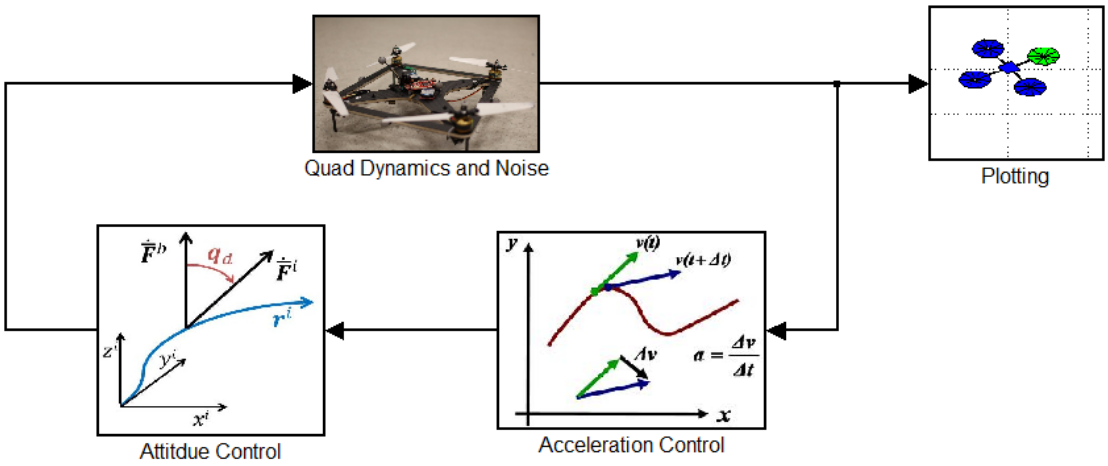


Figure 4-6: The top level of the simulation environment developed using MATLAB's Simulink. The simulation captures the main dynamics of the vehicle while ignoring aerodynamic effects. While never synced to real flight data, in practice the performance of the simulation and the actual vehicle is quite similar.

in the area will attempt to parametrize the trajectory polynomials by something other than time, such as percentage of the trajectory traveled, in order to close the loop around the trajectory generator and account for the current state of the vehicle when giving reference commands.

4.3 Simulation Environment

Critical to the success of the variable-pitch quadrotor is a 13-state simulator used to test controller algorithms and simulate trajectories before trying them on the actual hardware. The simulation is implemented in MATLAB's Simulink environment. The top level of the simulator block diagram is shown in Figure 4-6. The dynamic equations of motion (Equation 4.2-4.5) embedded in the simulator are standard first-order differential equations for a rigid body, here referenced from Chapter 11 of [59]. These equations are solved in the simulation environment using MATLAB's ode45 solver with a variable time step size. The quadrotor is visualized using a 3D plotter.

$$\dot{\mathbf{r}}^i = \mathbf{q}^* \otimes \begin{bmatrix} 0 \\ \dot{\mathbf{r}}^b \end{bmatrix} \otimes \mathbf{q} \tag{4.2}$$

$$\ddot{\mathbf{r}}^b = g \begin{bmatrix} 2(q_x q_z - q_y q^0) \\ 2(q_y q_z - q_x q^0) \\ q^{0^2} + q_z^2 - q_x^2 - q_y^2 \end{bmatrix} + \dot{\mathbf{r}}^b \times \mathbf{\Omega}^b + \begin{bmatrix} 0 \\ 0 \\ -\frac{1}{m} f_{total} \end{bmatrix} \tag{4.3}$$

$$\dot{\mathbf{q}} = \frac{1}{2} \mathbf{q} \otimes \begin{bmatrix} 0 \\ \mathbf{\Omega}^b \end{bmatrix} \tag{4.4}$$

$$\dot{\mathbf{\Omega}}^b = \mathbf{J}^{-1} \left[\mathbf{M}^b - \mathbf{\Omega}^b \times \mathbf{J}\mathbf{\Omega}^b \right] \tag{4.5}$$

The simulation captures the main dynamics of the vehicle while ignoring effects such as aerodynamic drag. Thus, the simulation environment is obviously not a perfect representation of the physical quadrotor; however, repeated testing of the hardware has shown the simulation to perform similarly to the quadrotor. In general, if a maneuver or controller is feasible in the simulation, it will work on the physical hardware. Also, if a maneuver or controller is infeasible in the simulation, it will definitely not work on the physical hardware.

4.4 Summary

The variable-pitch quadrotor, both hardware and control software, is almost entirely custom designed. The hardware uses typical RC airplane components and is built to minimize vibrations induced by the motors and variable-pitch actuators. The quadrotor is controlled with a custom autopilot which allows complete control over the lowest level of quadrotor control. The off-board control software is based on ROS and is written to be modular and easy to debug problems. Finally, a simulation environment built in MATLAB allows for quick prototyping of new maneuvers and control laws.

Chapter 5

Experimental Results

5.1 Introduction

Results of the control and trajectory generation techniques developed in Chapter 3 as implemented on the variable-pitch quadrotor are presented in this chapter. All the flights are performed in the RAVEN flight testing facility at MIT [1, 2].

In terms of the trajectory generation algorithm presented in Section 3.4, the variable-pitch quadrotor is advantageous because the addition of negative thrust more than doubles the effective thrust range for each of the four motors when compared to an equivalently powered fixed-pitch quadrotor. The reverse thrust capabilities of the variable-pitch quadrotor enable both inverted flight and vertical decelerations higher than gravity. As discussed in Chapter 2, variable-pitch propellers also increase the available controller bandwidth by effectively cancelling the motor dynamics. The variable-pitch propellers are thus able to change thrust substantially faster than corresponding fixed-pitch propellers.

Flight results in this chapter demonstrate the ability of the quadrotor to track paths upright and inverted, to quickly decelerate, and to perform aerobatic maneuvers using the position-based trajectory generation method.

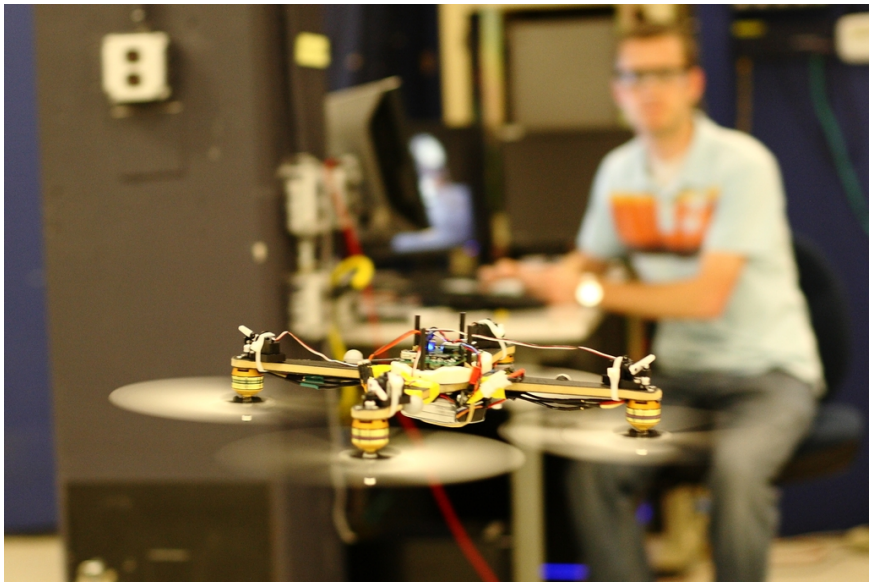


Figure 5-1: The quadrotor during inverted flight. The symmetry introduced by the variable-pitch propellers allows the quadrotor to track reference commands equally well upright or inverted.

5.2 Inverted and Upright Tracking

The variable-pitch quadrotor utilizes symmetric propellers. Combined with the inherent symmetry of the vehicle design, these propellers permit the quadrotor to fly equally well upright or inverted. The first set of flight results, shown in Figure 5-2, demonstrate the ability of the vehicle track the same path in either the upright or the inverted configuration.

5.3 Negative Thrust Decelerations

One of the primary advantages of the variable-pitch propellers for a quadrotor is the ability to generate negative thrust. While this allows the vehicle to fly upside down, it also brings the capability to decelerated quickly by momentarily reversing the propeller pitch to create upwards thrust. This capability is highlighted in Figures 5-3(a) and 5-3(b). In both figures, the quadrotor reference commands are the same; however, in variable-pitch mode the quadrotor is able to track the reference position command with only 1% overshoot compared to 60% overshoot when flying in fixed-

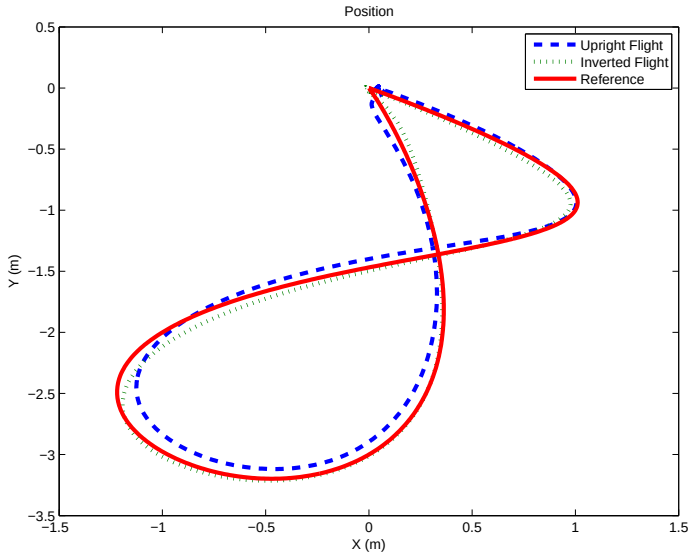
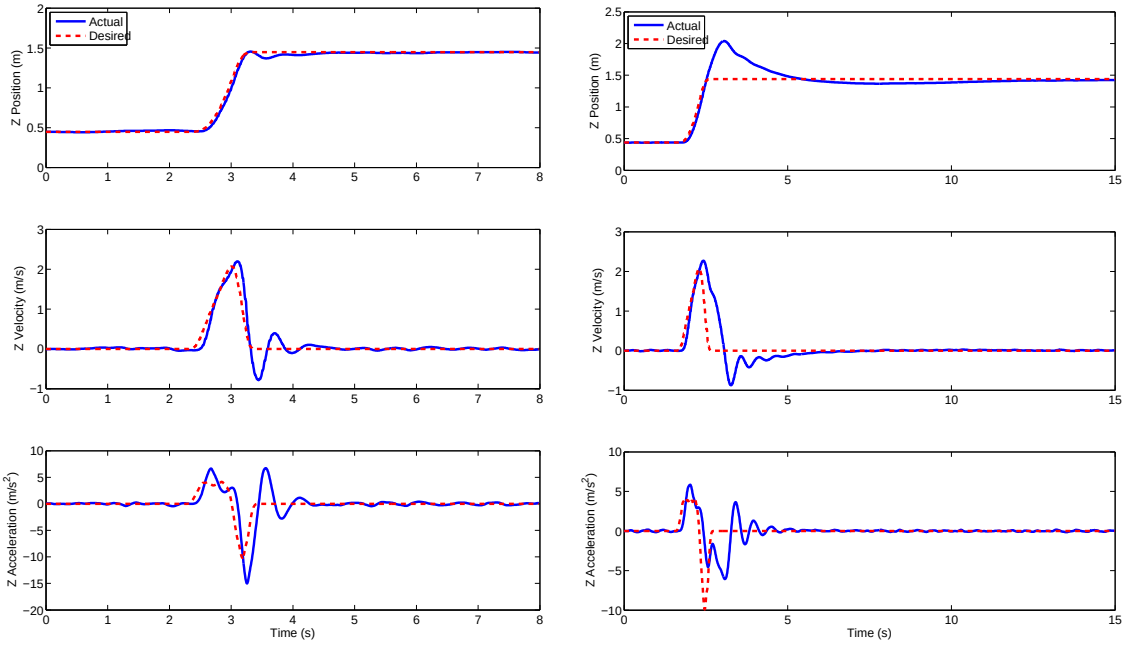


Figure 5-2: Path tracking qualities of the quadrotor. The vehicle is commanded to follow the same path both upright and inverted. Symmetry in the vehicle and propellers allow for similar flight characteristics upright or inverted.



(a) Variable-pitch flight data

(b) Fixed-pitch flight data

Figure 5-3: Flight data for the variable-pitch quadrotor flying the same trajectory in variable-pitch mode (5-3(a)) and in fixed-pitch mode (5-3(b)). The variable-pitch propellers allow for faster decelerations and better tracking of the position reference command.

pitch mode. The improved tracking performance in Figure 5-3(a) is due primarily to the large negative accelerations that are achieved only when the pitch of the propellers is allowed to vary.

5.4 Flips

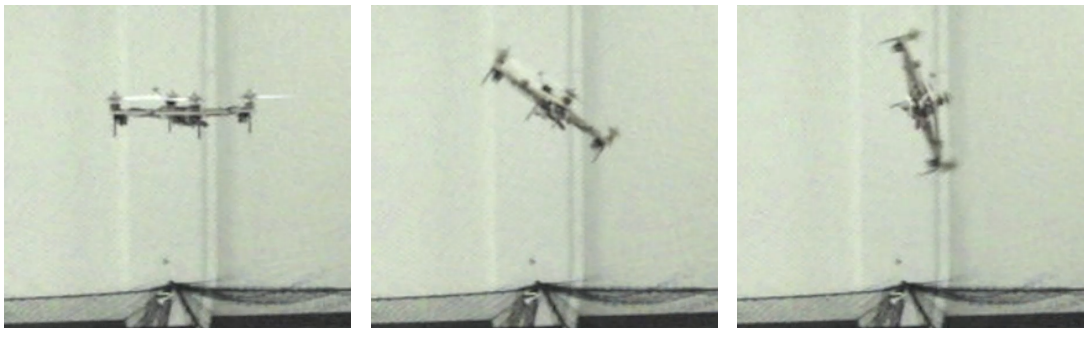
As described in Section 3.5, interested aerobatic maneuvers can be performed using the trajectory generation method by embedding attitude constraints along the path. The first example, snapshots of which are shown in Figure 5-4, shows the quadrotor performing a 180 degree flip. The quadrotor is commanded to follow a parabolic trajectory in the x-z plane, starting and stopping at hover, with a $-\mathbf{g}$ acceleration constraint imposed in the middle. At the apex of the parabola, the quadrotor is commanded to fly inverted, resulting in a 180 degree flipping maneuver.

Figure 5-5 shows the angular position and rate tracking abilities of the variable-pitch quadrotor during the 180 degree flip. The entire maneuver takes less than 0.4 seconds and the quadrotor rotates at over 1000 degrees per second.

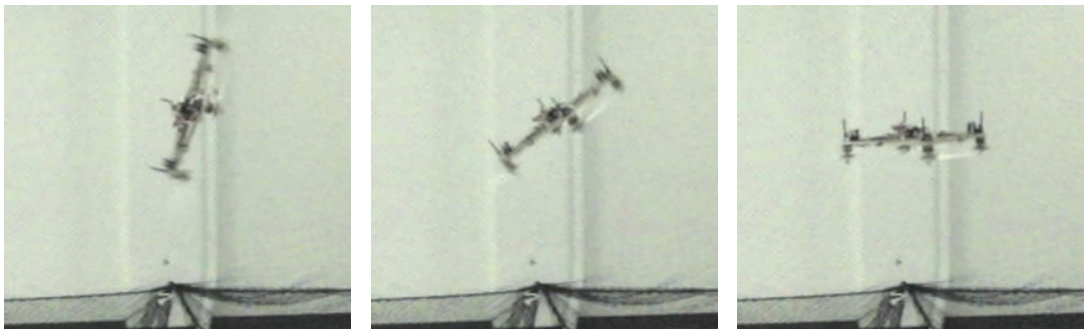
Next, a 180 degree flip is embedded into a translating path to demonstrate the ability of the vehicle and algorithms to perform moving aerobatics. This maneuver is shown in Figure 5-6. The vehicle starts and stops at hover and travels nearly 4 m/s forward and 2 m/s upwards just before the flip.

Finally, snapshots of hardware results of the STARMAC-inspired backflip (simulation results shown in Figure 3-5) are shown in Figure 5-7. The backflip is similar to the translating 180 degree flip in Figure 5-6 except the quadrotor performs a full 360 degree flip. This maneuver proved quite difficult for the quadrotor to execute because it requires a relatively high deceleration after the flip. As the quadrotor exists the flip the vehicle is moving nearly 3 m/s forward and over 2 m/s downwards. The vehicle requires nearly full thrust to slow to a stop.

Videos of these flight experiments can be found at `http://www.youtube.com/user/AerospaceControlsLab`.



(a) (b) (c)



(d) (e) (f)

Figure 5-4: Variable-pitch quadrotor performing a 180 degree flip by embedding a 90 degree roll constraint at the top of an arc in the X-Z plane.

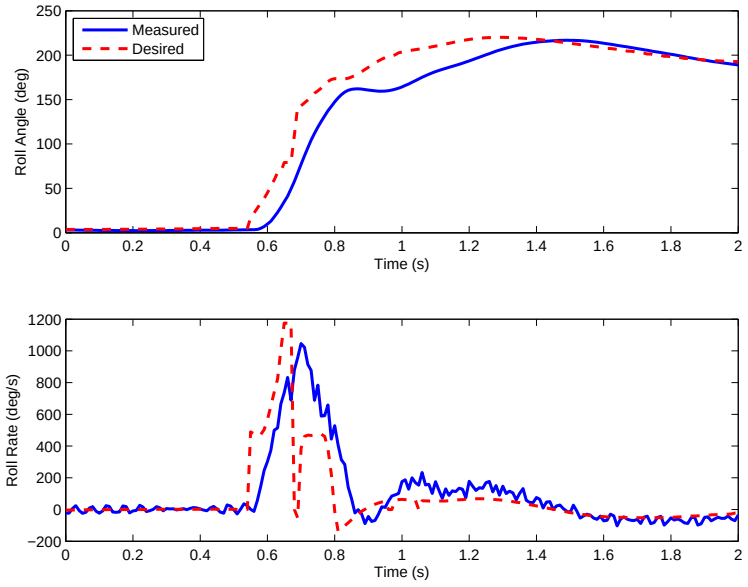


Figure 5-5: Commanded and measured roll and roll rate values from the quadrotor following a flipping maneuver. The measured values come from the on-board rate gyros. The flip takes less than 0.4 seconds to complete. Snapshots of the quadrotor during the flip are shown in Figure 5-4.

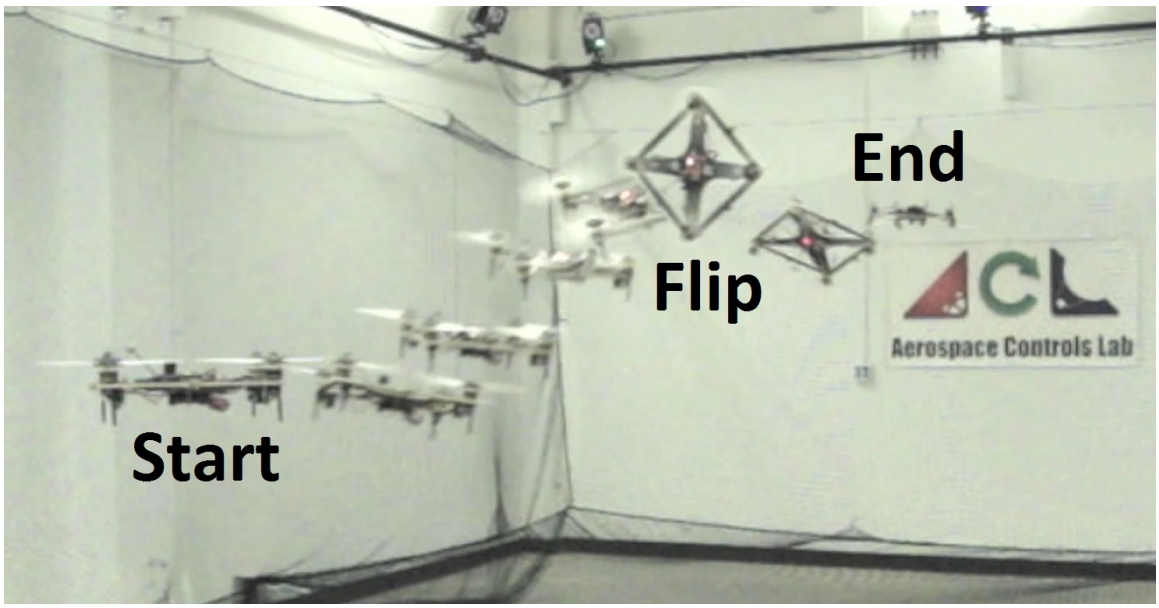
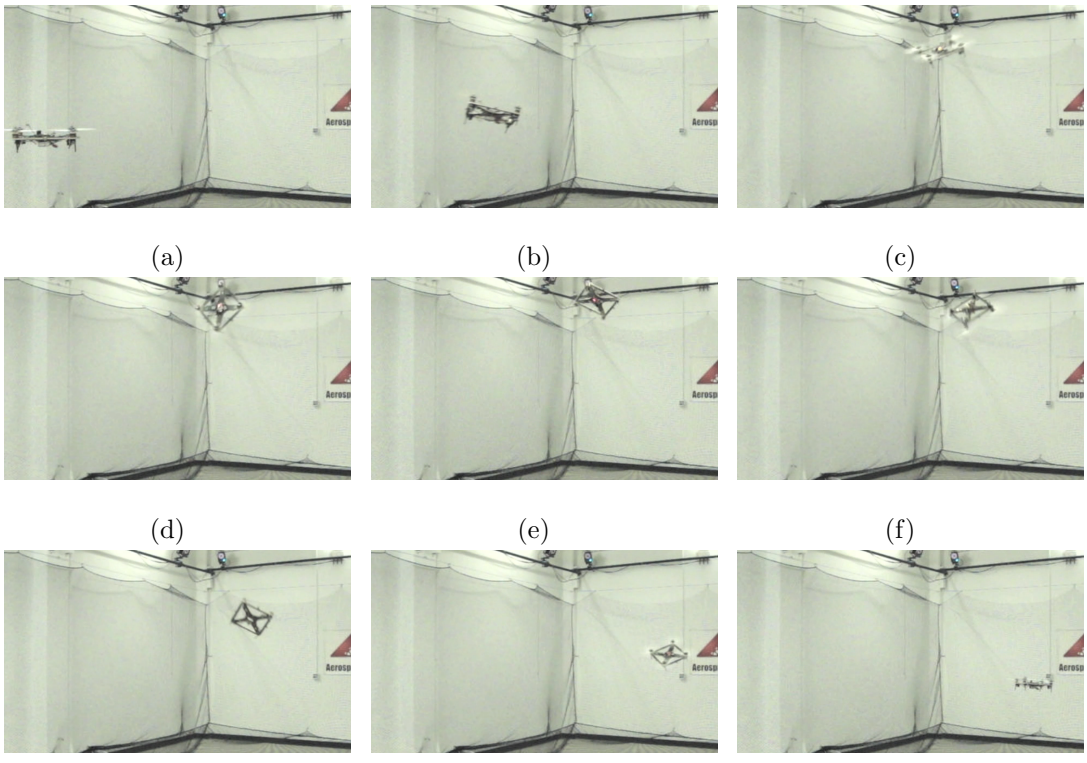


Figure 5-6: The quadrotor performing a translating 180 degree flip. The vehicle starts and ends at hover and performs a half back flip in the middle of the path. The vehicle travels forward at nearly 4 m/s during the maneuver.



(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

Figure 5-7: Variable-pitch quadrotor performing a 360 degree translating backflip. Simulations of this backflip are shown in Figure 3-5. This maneuver was inspired by the Stanford STARMAC project.

Chapter 6

Conclusion

6.1 Summary

This thesis details the design, development, and control of a variable-pitch quadrotor helicopter. The purpose of the variable-pitch quadrotor is to overcome fundamental limitations of fixed-pitch quadrotors without introducing the mechanical complexity of a pod-and-boom style helicopter. The variable-pitch quadrotor accomplishes this goal by increasing the actuator bandwidth and increasing the available thrust. In particular, the variable-pitch quadrotor allows for inverted flight and decelerations greater than gravity. These capabilities greatly increase the possible aggressive and aerobatic maneuvers that the quadrotor can perform as compared with a fixed-pitch quadrotor.

The main contributions of this thesis are (1) a detailed analysis of the benefits of variable-pitch propellers over fixed-pitch propellers for a quadrotor helicopter; (2) the design and implementation of closed-loop attitude and acceleration controllers with a trajectory generation algorithm for utilizing the increased actuator capabilities of the variable-pitch quadrotor; and (3) the construction and flight testing of a variable-pitch quadrotor, validating the actuator analysis and control design.

Chapter 1 gives background information on the recent multi-copter movement in the aerospace community, particularly among academic researchers and hobbyists. Some of the recent work on aggressive and aerobatic autonomous flight is highlighted,

with a particular emphasis on work performed by quadrotor helicopters flown indoors.

In Chapter 2, a detailed analysis and comparison of fixed-pitch and variable-pitch actuators for a quadrotor is performed. This analysis is verified using both bench motor testing and full hardware testing. Variable-pitch propellers are shown to yield a substantial increase over fixed-pitch propellers in the available thrust rate of change. Also, variable-pitch propellers can be utilized to generate negative thrust, increasing the potential for aerobatics and allowing for inverted flight.

A nonlinear, quaternion-based control scheme is outlined in Chapter 3. The controller does not rely on near-hover assumptions and avoids singularities by using quaternions. Also, the quaternion framework makes the controller computationally easy to implement on an embedded computer. A trajectory generation algorithm is also presented with an optimization routine for finding minimum-time trajectories relative to the actuator saturation bounds. Example simulation trajectories are shown.

An overview of the hardware design process and software infrastructures is presented in Chapter 4. The variable-pitch quadrotor built for the project is custom designed and uses a combination of custom and commercial parts. Low-level attitude control is performed using a custom autopilot. The autopilot allows for complete control of the control algorithms and access to on-board state data. Off-board control is performed with the assistance of a Vicon motion capture system that provides accurate position and attitude measurements of the vehicle.

Lastly, Chapter 5 presents some flight results of the variable-pitch quadrotor flying the control and trajectory generation algorithms from Chapter 3. The quadrotor flies trajectories both upright and inverted and is able to autonomously transition from upright to inverted flight.

6.2 Future Work

Significant future work remains in order to fully understand and exploit the capabilities of the variable-pitch quadrotor, particularly in the area of control design and

trajectory generation. As with many robotic platforms, the hardware is still more capable than the software controlling it. When controlled by expert pilots, conventional RC helicopters perform amazing aerobatics and aggressive flight. Given the correct control commands, the variable-pitch quadrotor should be capable of performing many of the same aerobatic maneuvers. The flight results in Chapter 5 show initial progress in this area; however, many more flight results could be obtained. Some main areas of future work needed in order to obtain these flight results include (1) developing a control law that accounts for aerodynamic effects, (2) relaxing the requirement that commanded paths do not command free-fall, and (3) designing more precise variable-pitch actuators.

The control methodology outlined in Section 3.2 essentially models the quadrotor as a point mass that can generate thrust in any direction. The acceleration control loop generates a commanded thrust magnitude and direction and assumes that the attitude control loop quickly matches the commanded values. These assumptions appear to be well validated when the vehicle velocity is low (less than about 4 m/s); however, when the velocity increases, the vehicle often starts deviating from the commanded path. Some of this deviation can be attributed to a lack of aerodynamic information in the control model, since aerodynamics effects become more significant as the vehicle velocity increases. Similar deviations at high velocities due to inaccurate modeling of aerodynamic effects was noted in [37].

The closed-loop control requires that the vehicle have some non-zero commanded thrust for the algorithm to compute the required attitude and attitude rates. Therefore, if the desired path commands free-fall for a finite amount of time, the current control algorithm cannot compute the necessary attitude and attitude rates during the free-fall. It may be necessary to switch to an attitude-specific control law during vehicle free-fall to avoid this singularity.

Finally, the variable-pitch quadrotor hardware could benefit from better designed variable-pitch actuators. As mentioned in Chapter 4, the variable-pitch actuators used in this thesis were designed for RC airplanes and exhibit some slop in the commanded pitch setting. Variable-pitch actuators capable of more precise pitch settings could

both decrease the vibration in the vehicle and increase the flight performance by improving the position and attitude tracking capabilities.

Appendix A

Quaternions

A detailed description of quaternions and their use as rotation operators can be found in [38]. For reference, a few of the common properties of quaternions are presented here.

When used to represent rotations, a quaternion consists of a scalar number defining the angle of rotation and a scaled unit vector defining the axis of rotation. There is no universal standard for the ordering of the elements of a quaternion. In this thesis the quaternion convention

$$\mathbf{q} = \begin{bmatrix} q^0 \\ \vec{q} \end{bmatrix} = \begin{bmatrix} q^0 \\ \vec{q}_x \\ \vec{q}_y \\ \vec{q}_z \end{bmatrix}$$

is used, where q^0 is the scalar portion and $\vec{q}$ is the vector portion of the quaternion.

Quaternion multiplication is denoted in this thesis by the symbol $\otimes$ and is defined in terms of cross and dot products as

$$\mathbf{q} \otimes \mathbf{p} = \begin{bmatrix} q^0 \\ \vec{q} \end{bmatrix} \otimes \begin{bmatrix} p^0 \\ \vec{p} \end{bmatrix} = \begin{bmatrix} q^0 p^0 - \vec{q} \cdot \vec{p} \\ q^0 \vec{p} + p^0 \vec{q} + \vec{q} \times \vec{p} \end{bmatrix}$$

and in terms of scalar multiplications as

$$\mathbf{q} \otimes \mathbf{p} = \begin{bmatrix} q^0 p^0 - \vec{q}_x \vec{p}_x - \vec{q}_y \vec{p}_y - \vec{q}_z \vec{p}_z \\ q^0 p_x + \vec{q}_x p^0 + \vec{q}_y \vec{p}_z - \vec{q}_z \vec{p}_y \\ q^0 p_y - \vec{q}_x \vec{p}_z + \vec{q}_y p^0 + \vec{q}_z \vec{p}_x \\ q^0 p_z + \vec{q}_x \vec{p}_y - \vec{q}_y \vec{p}_x + \vec{q}_z p^0 \end{bmatrix}$$

Note that quaternion multiplication is not commutative.

The quaternion conjugate, similar to the inverse of a rotation matrix, is defined as

$$\mathbf{q}^* = \begin{bmatrix} q^0 \\ -\vec{q} \end{bmatrix}$$

and, when multiplied with itself, yields the identity quaternion.

$$\mathbf{q} \otimes \mathbf{q}^* = \mathbf{q}^* \otimes \mathbf{q} = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}^T$$

Quaternions can be used to represent the attitude of a body by representing the attitude as a rotation from a fixed frame to the body's frame. When the fixed frame is the inertial frame, the quaternion describes the inertial attitude of the body. The angle of rotation, θ, described by the quaternion and the unit rotation vector, $\vec{u}$, are related to the elements of the quaternion as

$$\theta = 2\cos^{-1}(q^0)$$
$$\vec{u} = \frac{\vec{q}}{\sin\frac{\theta}{2}}.$$

Thus, given that $\|\vec{u}\| = 1$, it is clear that $\mathbf{q}$ is a unit quaternion since

$$\|\mathbf{q}\| = \sqrt{\mathbf{q}^* \otimes \mathbf{q}} = 1$$

Utilizing the conjugate properties of the quaternion, quaternions can be used to rotate vectors between reference frames. The quaternion rotation operation that

rotates the vector $\mathbf{v}$ in $\mathbb{R}^3$ from frame a to frame b is defined as

$$\begin{bmatrix} 0 \\ \mathbf{v}^b \end{bmatrix} = \mathbf{q}^* \otimes \begin{bmatrix} 0 \\ \mathbf{v}^a \end{bmatrix} \otimes \mathbf{q},$$

THIS PAGE INTENTIONALLY LEFT BLANK

Appendix B

Compute Angular Acceleration

Similar to the calculation of angular velocities in Section 3.2, the angular accelerations of the quadrotor can be related to the linear snap of the quadrotor, provided the linear acceleration is non-zero at all points along the path. Although the angular accelerations are not used for the real-time control scheme, they are needed for the path optimization explained in Section 3.4.1. Using the Transport Theorem to differentiate Equation 3.17 yields

$$\ddot{\bar{\mathbf{F}}}^i = \boldsymbol{\Omega}_d^b \times (\boldsymbol{\Omega}_d^b \times \dot{\bar{\mathbf{F}}}^i) + \frac{d}{dt}\boldsymbol{\Omega}_d^b \times \dot{\bar{\mathbf{F}}}^i. \tag{B.1}$$

Equivalently, the time derivative of Equation 3.12 is calculated explicitly as

$$\ddot{\bar{\mathbf{F}}}^i = \frac{d}{dt}\left(\frac{\dot{\mathbf{F}}^i}{\|\mathbf{F}^i\|} - \frac{\mathbf{F}^i(\mathbf{F}^{i^T}\dot{\mathbf{F}}^i)}{\|\mathbf{F}^i\|^3}\right) \tag{B.2}$$

$$= \frac{\ddot{\mathbf{F}}^i}{\|\mathbf{F}^i\|} - \frac{2\dot{\mathbf{F}}^i(\mathbf{F}^{i^T}\dot{\mathbf{F}}^i) + \mathbf{F}^i(\dot{\mathbf{F}}^{i^T}\dot{\mathbf{F}}^i) + \mathbf{F}^i(\mathbf{F}^{i^T}\ddot{\mathbf{F}}^i)}{\|\mathbf{F}^i\|^3} + \frac{3\mathbf{F}^i(\mathbf{F}^{i^T}\dot{\mathbf{F}}^i)}{\|\mathbf{F}^i\|^5} \tag{B.3}$$

Rearranging Equation B.1 gives the desired body-frame angular acceleration vector projected onto the body-frame x-y plane. The yaw angular acceleration is calculated by differentiating the yaw command twice.

$$\dot{\boldsymbol{\Omega}}_{d_{XY}}^b = \bar{\mathbf{F}}^i \times \left[\ddot{\bar{\mathbf{F}}}^i - \boldsymbol{\Omega}_d^b \times (\boldsymbol{\Omega}_d^b \times \dot{\bar{\mathbf{F}}}^i)\right] \tag{B.4}$$

$$\boldsymbol{\Omega}_{d_Z}^b = \ddot{\psi}_d \tag{B.5}$$

THIS PAGE INTENTIONALLY LEFT BLANK

Bibliography

[1] M. Valenti, B. Bethke, G. Fiore, J. P. How, and E. Feron. Indoor Multi-Vehicle Flight Testbed for Fault Detection, Isolation, and Recovery. In *AIAA Guidance, Navigation, and Control Conference (GNC)*, Keystone, CO, August 2006 (AIAA-2006-6200).

[2] J. P. How, B. Bethke, A. Frank, D. Dale, and J. Vian. Real-time indoor autonomous vehicle test environment. *IEEE Control Systems Magazine*, 28(2):51–64, April 2008.

[3] J.M. McMichael and M.S. Francis. Micro air vehicles-toward a new dimension in flight. *DARPA Document*, 1997.

[4] E. Altug, J.P. Ostrowski, and R. Mahony. Control of a quadrotor helicopter using visual feedback. In *IEEE International Conference on Robotics and Automation (ICRA)*, volume 1, pages 72–77, 2002.

[5] M.Y. Amir and V. Abbass. Modeling of quadrotor helicopter dynamics. In *International Conference on Smart Manufacturing Application (ICSMA 2008)*, pages 100 –105, April 2008.

[6] B. Erginer and E. Altug. Modeling and pd control of a quadrotor vtol vehicle. In *IEEE Intelligent Vehicles Symposium*, pages 894 –899, June 2007.

[7] M. Alpen, K. Frick, and J. Horn. Nonlinear modeling and position control of an industrial quadrotor with on-board attitude control. In *IEEE International Conference on Control and Automation*, pages 2329 –2334, dec. 2009.

[8] J. Kim, M.S. Kang, and S. Park. Accurate modeling and robust hovering control for a quad–rotor VTOL aircraft. *Journal of Intelligent and Robotic Systems*, 57(1):9–26, 2010.

[9] Haomiao Huang, G.M. Hoffmann, S.L. Waslander, and C.J. Tomlin. Aerodynamics and control of autonomous quadrotor helicopters in aggressive maneuvering. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 3277 –3282, May 2009.

[10] D. Gurdan, J. Stumpf, M. Achtelik, K. M. Doth, G. Hirzinger, and D. Rus. Energy-efficient Autonomous Four-rotor Flying Robot Controlled at 1 kHz. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 361–366, 2007.

[11] N. Michael, D. Mellinger, Q. Lindsey, and V. Kumar. The grasp multiple micro-UAV testbed. *IEEE Robotics & Automation Magazine*, 17(3):56–65, 2010.

[12] J.H. Gillula, H. Huang, M.P. Vitus, and C.J. Tomlin. Design of guaranteed safe maneuvers using reachable sets: Autonomous quadrotor aerobatics in theory and practice. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 1649–1654, 2010.

[13] G.M. Hoffmann, S. Waslander, and C.J. Tomlin. Quadrotor helicopter trajectory tracking control. In *AIAA Guidance, Navigation and Control Conference and Exhibit, Honolulu, Hawaii*, 2008.

[14] R. Ritz, M. Hehn, S. Lupashin, and R. D'Andrea. Quadrocopter performance benchmarking using optimal control. In *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*, pages 5179–5186. IEEE, 2011.

[15] D. Mellinger and V. Kumar. Minimum snap trajectory generation and control for quadrotors. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2011.

[16] P. Pounds and R. Mahony. Design principles of large quadrotors for practical applications. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 3265–3270, 2009.

[17] J.G. Leishman. *Principles of helicopter aerodynamics*. Cambridge Univ Pr, 2006.

[18] C. Gablehouse. *Helicopters and autogiros: a chronicle of rotating-wing aircraft*. Lippincott, 1967.

[19] J. Borenstein. The hoverbot, an electrically powered flying robot. 1992. (`http://www.cs.cmu.edu/~motionplanning/papers/sbp_papers/integrated1/borenstein_hovercraft.pdf`).

[20] G. d'Ambrosio and R. Navoni. Hg3 willy [video]. July 2011. (`http://youtu.be/M4uXmekZk-4`).

[21] Hsieh Chen. Variable-pitch quadrotor [video]. July 2011. (`http://youtu.be/fkSx3fSz0tE`).

[22] V. Gavrilets, E. Frazzoli, B. Mettler, M. Piedmonte, and E. Feron. Aggressive Maneuvering of Small Helicopters: A Human Centered Approach. *International Journal of Robotics Research*, 20:705–807, October 2001.

[23] Pieter Abbeel, Adam Coates, Morgan Quigley, and Andrew Y. Ng. An application of reinforcement learning to aerobatic helicopter flight. In *Advances in Neural Information Processing Systems (NIPS)*, page 2007. MIT Press, 2007.

[24] S. Lupashin, A. Schollig, M. Sherback, and R. D'Andrea. A simple learning strategy for high-speed quadrocopter multi-flips. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 1642–1648. IEEE, 2010.

[25] D. Mellinger, N. Michael, and V. Kumar. Trajectory generation and control for precise aggressive maneuvers with quadrotors. In *Int. Symposium on Experimental Robotics*, 2010.

[26] M. Muller, S. Lupashin, and R. D'Andrea. Quadrocopter ball juggling. In *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*, pages 5113–5120. IEEE, 2011.

[27] M. Hehn and R. D'Andrea. A flying inverted pendulum. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 763–770. IEEE, 2011.

[28] B. Michini, J. Redding, N. K. Ure, M. Cutler, and J. P. How. Design and flight testing of an autonomous variable-pitch quadrotor. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 2978 – 2979. IEEE, May 2011.

[29] M. Cutler, N. Kemal Ure, B. Michini, and J. P. How. Comparison of fixed and variable pitch actuators for agile quadrotors. In *AIAA Guidance, Navigation, and Control Conference (GNC)*, Portland, OR, August 2011. (AIAA-2011-6406).

[30] Mark Cutler and Jonathan P. How. Actuator constrained trajectory generation and control for variable-pitch quadrotors. In *AIAA Guidance, Navigation, and Control Conference (GNC)*, Minneapolis, Minnesota, August 2012.

[31] M. Drela. Xfoil- an analysis and design system for low reynolds number airfoils. *Low Reynolds Number Aerodynamics*, pages 1–12, 1989. (`http://web.mit.edu/drela/Public/web/xfoil/`).

[32] T. Bresciani. *Modelling, identification and control of a quadrotor helicopter*. Department of Automatic Control, Lund University, 2008.

[33] N. Hemati and M.C. Leu. A complete model characterization of brushless dc motors. *Industry Applications, IEEE Transactions on*, 28(1):172–180, 1992.

[34] M. Drela. Qprop users guide, 2009. (`http://web.mit.edu/drela/Public/web/qprop/`).

[35] P.J. Bristeau, P. Martin, E. Salaun, and N. Petit. The role of propeller aerodynamics in the model of a quadrotor UAV. In *European Control Conference*, 2009.

[36] G. Franklin, JD Powell, and S. Emami-Naeini. *Feedback Control of Dynamic Systems*. Prentice Hall, 2006.

[37] M. Hehn and R. D'Andrea. Quadrocopter trajectory generation and control. In *World Congress*, volume 18, pages 1485–1491, 2011.

[38] J. B. Kuipers. *Quaternions and Rotation Sequences: A Primer with Applications to Orbits, Aerospace, and Virtual Reality*. Princeton University Press, Princeton, NJ, 2002.

[39] M. Turpin, N. Michael, and V. Kumar. Trajectory design and control for aggressive formation flight with quadrotors. *Proc. of the Intl. Sym. of Robot. Research. Flagstaff, AZ*, 2011.

[40] F.L. Markley. Fast quaternion attitude estimation from two vector measurements. *Journal of Guidance, Control, and Dynamics*, 25(2):411–414, 2002.

[41] N.A. Chaturvedi, A.K. Sanyal, and N.H. McClamroch. Rigid-body attitude control. *Control Systems, IEEE*, 31(3):30–51, 2011.

[42] H. Baruh. *Analytical dynamics*. WCB/McGraw-Hill, 1999.

[43] Bernard Michini. Modeling and adaptive control of indoor unmanned aerial vehicles. Master's thesis, Massachusetts Institute of Technology, Department of Aeronautics and Astronautics, Cambridge MA, September 2009.

[44] B. Wie and P. M. Barba. Quaternion feedback for spacecraft large angle maneuvers. *AIAA Journal on Guidance, Control, and Dynamics*, 8:360–365, 1985.

[45] Jonathan P. How, Emilio Frazzoli, and Girish Chowdhary. *Handbook of Unmanned Aerial Vehicles*, chapter Linear Flight Contol Techniques for Unmanned Aerial Vehicles. Springer, 2012 (to appear).

[46] Girish Chowdhary, Emilio Frazzoli, Jonathan P. How, and Hugh Lui. *Handbook of Unmanned Aerial Vehicles*, chapter Nonlinear Flight Contol Techniques for Unmanned Aerial Vehicles. Springer, 2012 (to appear).

[47] Girish Chowdhary, Eric N. Johnson, Rajeev Chandramohan, Scott M. Kimbrell, and Anthony Calise. Autonomous guidance and control of airplanes under actuator failures and severe structural damage. *Journal of Guidance Control and Dynamics*, 2012. in-press.

[48] Girish Chowdhary, Tongbin Wu, Mark Cutler, Nazim Kemal Üre, and Jonathan How. Experimental results of concurrent learning adaptive controller. In *AIAA Guidance, Navigation, and Control Conference (GNC)*, Minneapolis, MN, August 2012. AIAA. Invited.

[49] Wassim M. Haddad and VijaySekhar Chellaboina. *Nonlinear Dynamical Systems and Control: A Lyapunov-Based Approach*. Princeton University Press, Princeton, 2008.

[50] K. Kannan Suresh. *Adaptive Control of Systems in Cascade with Saturation*. PhD thesis, Ph. D. Thesis, Georgia Institute of Technology Atlanta Ga, 2005.

[51] MsComposit. *9725 Owensmouth Ave. Chatsworth, CA 91311* `http://www.mscomposit.com`.

[52] R. Beard and T. McLain. *Small Unmanned Aircraft: Theory and Practice*. Princeton Univ. Press, Princeton, NJ, 2012.

[53] Unmanned Innovation Inc. *7 Sunset Way, Suite 100, Henderson, NV 89014* `info@unmannedinnovation.com`.

[54] Sparkfun Electronics. *6175 Longbow Drive Suite 200 Boulder, CO 80301* `http://www.sparkfun.com`.

[55] Microchip Technology Inc. *2355 West Chandler Blvd. Chandler, AZ 85224* `http://www.microchip.com`.

[56] Invensense. *1197 Borregas Ave. Sunnyvale, CA 94089* `http://www.invensense.com`.

[57] Motion capture systems from Vicon. Technical report, 2011. Online `http://www.vicon.com/`.

[58] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A.Y. Ng. Ros: an open-source robot operating system. In *ICRA Workshop on Open Source Software*, volume 3, 2009.

[59] Warren F. Phillips. *Mechanics of Flight*. Hoboken, N.J. John Wiley & Sons, 2010.