Actuator Constrained Trajectory Generation and Control for Variable-Pitch Quadrotors

Mark Cutler* and Jonathan P. How†

Control and trajectory generation algorithms for a quadrotor helicopter with variable-pitch propellers are presented. 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. Experimental results on a variable-pitch quadrotor demonstrate the control design and example trajectories.

I. Introduction

The past several years have seen significant growth in the area of multi-rotor helicopter flight and control. In particular, fixed-pitch quadrotor helicopters are widely used as experimental and hobby platforms, primarily due to their mechanical simplicity, robustness, and relative safety in the presence of humans. Considerable recent work exists on the modeling,[1–4] design,[5] control,[6–9] and trajectory generation[10–13] for quadrotors with fixed-pitch propellers. While relatively aggressive flight has been demonstrated with traditional fixed-pitch quadrotors, such as large deviations from hover attitude and nominal velocities[11,12] and aerobatic maneuvers,[7,9] the use of fixed-pitch propellers place fundamental limitations on the capabilities of the quadrotor to perform certain aggressive and aerobatic maneuvers characteristic of traditional pod-and-boom style helicopters.[14,15] In particular, the control bandwidth achieved using fixed-pitch propellers is limited by the rotational inertia of the motor/propeller combination. Additionally, generation of reverse thrust is not practical with fixed-pitch propellers. These limitations are, to a large extent, overcome with the addition of variable-pitch propellers to a quadrotor. While the variable-pitch propellers increase the mechanical complexity of the vehicle, it remains significantly less complex than a conventional helicopter since a swash-plate is not needed.

This paper builds on previous work by the authors, which detailed the design of a variable-pitch quadrotor.[16,17] While the control and trajectory generation 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,[11,12] the control law presented does not assume near hover flight regimes, allowing for large attitude deviations from hover.

Recent work demonstrates optimal trajectory generation for quadrotors using time-parametrized polynomials to represent the trajectory, guaranteeing smooth reference inputs to the quadrotor.[11]

*Research Assistant in the Aerospace Controls Lab at MIT cutlerm@mit.edu

†Richard C. Maclaurin Professor of Aeronautics and Astronautics, MIT. Associate Fellow AIAA. jhow@mit.edu

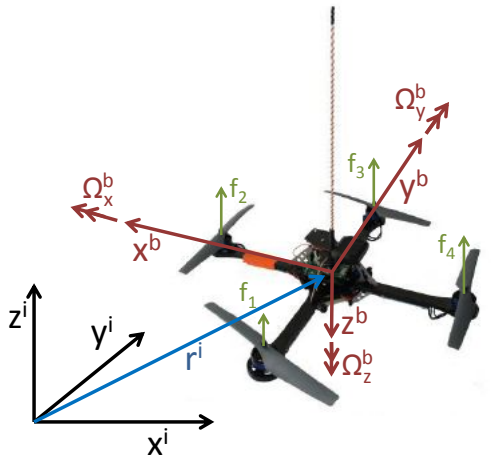


Figure 1. Quadrotor model and reference frames. superscript i denotes the inertial frame and superscript b denotes the body frame.

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 using switching control laws.[7,9,10]

The structure of the paper is as follows: first, a dynamic model of the quadrotor is developed in Section II, and a feedback control solution is proposed to control the quadrotor along a specified 3-D trajectory in $\mathbb{R}^3$ in Section III. Then, a closed-form solution for generating smooth trajectories through any number of time-parametrized waypoints is proposed in Section IV. An optimization method is proposed for constructing smooth minimum-time trajectories through waypoints while satisfying motor saturation constraints in Section IV.A. In Section IV.B a method for embedding attitude constraints along the path is presented. Finally, Section V shows the algorithms implemented on a variable-pitch quadrotor in both simulation and hardware.

II. Dynamic Model

Consider the quadrotor vehicle depicted in Figure 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

$$\begin{bmatrix} 0 \\ \mathbf{v}^i \end{bmatrix} = \mathbf{q}^* \otimes \begin{bmatrix} 0 \\ \mathbf{v}^b \end{bmatrix} \otimes \mathbf{q}, \tag{1}$$

where $\mathbf{q}^*$ is the quaternion conjugate of $\mathbf{q}$ and $\otimes$ is the quaternion multiplication operator.[18] The quaternion $[0, \mathbf{v}^T]^T$ is a pure imaginary quaternion (a quaternion with zero scalar part). The

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

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

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

$$\begin{bmatrix} 0 \\ \ddot{\mathbf{r}}^i \end{bmatrix} = \frac{1}{m}\mathbf{q}^* \otimes \begin{bmatrix} 0 \\ \mathbf{F}^b \end{bmatrix} \otimes \mathbf{q} - \begin{bmatrix} 0 \\ \mathbf{g}^i \end{bmatrix} \tag{2}$$

$$\dot{\boldsymbol{\Omega}}^b = \mathbf{J}^{-1}\left[\mathbf{M}^b - \boldsymbol{\Omega}^b \times \mathbf{J}\boldsymbol{\Omega}^b\right] \tag{3}$$

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[13]

$$\begin{bmatrix} f_{total} \\ \mathbf{M}^b \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ d & 0 & -d & 0 \\ 0 & d & 0 & -d \\ -c & c & -c & c \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{bmatrix} \tag{4}$$

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 a maximum and minimum value as

$$f_{\min} \leq f_i \leq f_{\max}, \quad i = 1, \ldots, 4 \tag{5}$$

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 $f_{\min} > 0$[7,12] 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 $f_{\min} = -f_{\max}$.

III. Closed-loop Control

Quadrotors are under-actuated and differentially flat.[11] 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

$$\ddot{\mathbf{r}}_{fb}^i = -\mathbf{k}_p\mathbf{e}_p - \mathbf{k}_i\mathbf{e}_i - \mathbf{k}_d\mathbf{e}_d + \mathbf{g}^i \tag{6}$$

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

$$\begin{aligned} \mathbf{e}_p &= \mathbf{r}^i - \mathbf{r}_d^i \\ \mathbf{e}_i &= \int_0^t \mathbf{e}_p(\tau)d\tau \\ \mathbf{e}_d &= \dot{\mathbf{r}}^i - \dot{\mathbf{r}}_d^i. \end{aligned}$$

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

$$\mathbf{F}^i = m\left(\ddot{\mathbf{r}}_d^i + \ddot{\mathbf{r}}_{fb}^i\right). \tag{7}$$

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 Eq. 2 yields

$$m\left(\begin{bmatrix}0\\ \ddot{\mathbf{r}}^i\end{bmatrix} + \begin{bmatrix}0\\ \mathbf{g}^i\end{bmatrix}\right) = \mathbf{q}^* \otimes \begin{bmatrix}0\\ \mathbf{F}^b\end{bmatrix} \otimes \mathbf{q}. \tag{8}$$

Substituting Eq. 7 for the left hand side of Eq. 8 and normalizing both sides gives

$$\begin{bmatrix}0\\ \bar{\mathbf{F}}^i\end{bmatrix} = \tilde{\mathbf{q}}_d^* \otimes \begin{bmatrix}0\\ \bar{\mathbf{F}}^b\end{bmatrix} \otimes \tilde{\mathbf{q}}_d \tag{9}$$

where the unit vectors are defined as

$$\bar{\mathbf{F}}^i = \frac{\mathbf{F}^i}{\|\mathbf{F}\|^i} \tag{10}$$

$$\bar{\mathbf{F}}^b = \frac{\mathbf{F}^b}{\|\mathbf{F}\|^b} = \begin{bmatrix}0 & 0 & \pm 1\end{bmatrix}^T \tag{11}$$

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[19]

$$\tilde{\mathbf{q}}_d = \frac{1}{\sqrt{2(1+\bar{\mathbf{F}}^{i^T}\bar{\mathbf{F}}^b)}}\begin{bmatrix}1+\bar{\mathbf{F}}^{i^T}\bar{\mathbf{F}}^b\\ \bar{\mathbf{F}}^i \times \bar{\mathbf{F}}^b\end{bmatrix}. \tag{12}$$

The sign of the z-component of $\bar{\mathbf{F}}^b$ in Eq. 11 is selected so that $\bar{\mathbf{F}}^{i^T}\bar{\mathbf{F}}^b \geq 0$, ensuring that the direction of the body-frame thrust vector is aligned with the direction of the inertial-frame acceleration vector.

Eq. 12 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.[20] 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

$$\mathbf{q}_d = \tilde{\mathbf{q}}_d \otimes \begin{bmatrix}\cos(\psi_d/2) & 0 & 0 & \sin(\psi_d/2)\end{bmatrix}^T. \tag{13}$$

The total quadrotor thrust f_{total} is computed by rearranging Eq. 8 and solving for the z component of the force vector

$$f_{total} = \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix} \left(\mathbf{q} \otimes \begin{bmatrix} 0 \\ \mathbf{F}^i \end{bmatrix} \otimes \mathbf{q}^* \right) \tag{14}$$

Note that using the actual vehicle attitude, $\mathbf{q}$, instead of the desired attitude, $\mathbf{q}_d$, in Eq. 14 projects the desired total force onto the actual body-frame z-axis, adjusting the commanded thrust based on current errors in the vehicle attitude.

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[21] this derivative is

$$\frac{d}{dt} \begin{bmatrix} 0 \\ \bar{\mathbf{F}}^i \end{bmatrix} = \frac{d}{dt} \left(\mathbf{q}_d \otimes \begin{bmatrix} 0 \\ \bar{\mathbf{F}}^i \end{bmatrix} \otimes \mathbf{q}_d^* \right) + \begin{bmatrix} 0 \\ \mathbf{\Omega}_d^b \times \bar{\mathbf{F}}^i \end{bmatrix} \tag{15}$$

$$\dot{\bar{\mathbf{F}}}^i = \mathbf{\Omega}_d^b \times \bar{\mathbf{F}}^i \tag{16}$$

The first term on the right hand side of Eq. 15 is zero since $\bar{\mathbf{F}}$ is constant in the body frame. Rearranging Eq. 16 gives the desired body-frame angular rate vector projected onto the body-frame x-y plane.

$$\mathbf{\Omega}_{d_{xy}}^b = \bar{\mathbf{F}}^i \times \dot{\bar{\mathbf{F}}}^i \tag{17}$$

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

$$\mathbf{\Omega}_{d_z}^b = \dot{\psi}_d \tag{18}$$

The time derivative of $\bar{\mathbf{F}}^i$ is explicitly calculated using the quotient rule on Eq. 10 as

$$\dot{\bar{\mathbf{F}}}^i = \frac{\dot{\mathbf{F}}^i}{\|\mathbf{F}^i\|} - \frac{\mathbf{F}^i (\mathbf{F}^{i^T} \dot{\mathbf{F}}^i)}{\|\mathbf{F}^i\|^3} \tag{19}$$

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

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 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. In practice, this attitude ambiguity is accounted for by ensuring 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,[18] 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

$$\underbrace{\mathbf{q}_d}_{\text{inertial frame}} = \underbrace{\mathbf{q}}_{\text{inertial frame}} \otimes \underbrace{\mathbf{q}_e}_{\text{body frame}} . \tag{20}$$

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 Eq. 20 using the conjugate properties of the quaternion yields the error quaternion, expressed in

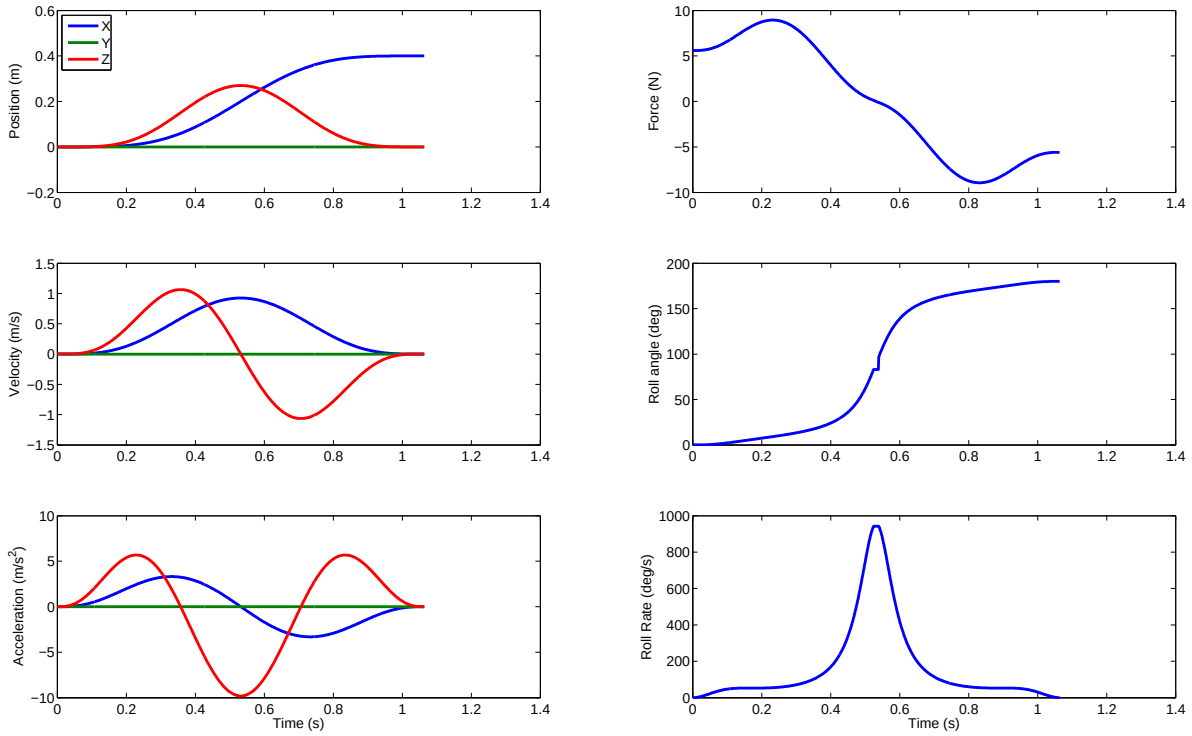


Figure 2. Trajectory generated by imposing a position free free-fall acceleration condition between two hover waypoints along the x-axis. The small corner in the commanded attitude trajectory comes from not computing new commanded attitudes when the total force command is close to zero. The vehicle goes inverted at the apex of the trajectory by explicitly changing $\sigma(t)$ from 1 to -1.

the body frame, as a simple quaternion multiplication between the actual attitude and the desired attitude.

$$\underbrace{\mathbf{q}_e}_{\text{body frame}} = \underbrace{\mathbf{q}^*}_{\text{inertial frame}} \otimes \underbrace{\mathbf{q}_d}_{\text{inertial frame}} . \tag{21}$$

Eq. 20 and 21 are similar to equations in previous work;[22] however, in this paper, the order of the quaternion multiplication differs so as to agree with standard notation and the rotation operation introduced in Eq. 1.[18]

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,[23–25] the attitude control is accomplished using proportional-derivative control on the attitude error and attitude rate error as

$$\mathbf{M}^b = -\operatorname{sgn}(q_e^0)\mathbf{K}_p\vec{q_e} - \mathbf{K}_d(\mathbf{\Omega}^b - \mathbf{\Omega}_d^b), \tag{22}$$

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 Eq. 4.

IV. Trajectory Generation

Given the control structure capable of tracking position and yaw reference commands developed in Section III, consider the problem of navigating through n waypoints in 3-space in an obstacle-free environment. Similar to previous work,[11,13] 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

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

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 IV.B. Third, polynomials for each of the four flat outputs, $x(t)$, $y(t)$, $z(t)$, and $\psi(t)$ 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) to the quadrotor 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 III, the inputs to the quadrotor are computed as a function of the first three derivatives of the position command. To ensure those inputs are smooth, the initial and final first four derivatives of position are constrained as

$$x(0) = x_0 \qquad x(t_f) = x_f \tag{23}$$

$$x^{(i)}(0) = 0 \qquad x^{(i)}(t_f) = 0 \qquad i = 1, \ldots, 4 \tag{24}$$

where the superscript in parentheses represents the i^{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.

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 $x(t_j^-)^{(1)} = x(t_j^+)^{(1)} = v_j$. When the velocity is not specified, the constraint is $x(t_j^-)^{(1)} - x(t_j^+)^{(1)} = 0$. 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.

IV.A. 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 Eq. 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

$$\mathbf{t} = \underset{\mathbf{t}}{\operatorname{argmin}} \quad t_{n-1} \tag{25}$$

$$\text{subject to} \quad f_{\min} \leq f_i \leq f_{\max} \quad i = 1, \dots, 4 \tag{26}$$

$$t_j > 0 \qquad\qquad j = 1, 2, \dots, n-1 \tag{27}$$

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. 2-3) are then inverted using the computed path as the reference command, returning the required forces and moments to fly that path. The individual motor thrust values are found by inverting the relationship in Eq. 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. 6 and 22 (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.

IV.B. Attitude Constraints

Specific attitude constraints can be incorporated into the desired path formulation by constraining the acceleration of the vehicle based on Eq. 8. 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

$$\begin{bmatrix} 0 \\ \ddot{\mathbf{r}}^i_{att} \end{bmatrix} = \frac{\|\mathbf{F}^b\|}{m} \mathbf{q}^*_{des} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \mathbf{q}_{des} - \begin{bmatrix} 0 \\ \mathbf{g}^i \end{bmatrix} \tag{28}$$

where $\|\mathbf{F}^i\|$ is chosen to scale the acceleration as desired. Eq. 28 allows the user to specify the attitude of the vehicle at polynomial nodes in the path. While the vehicle attitude between nodes

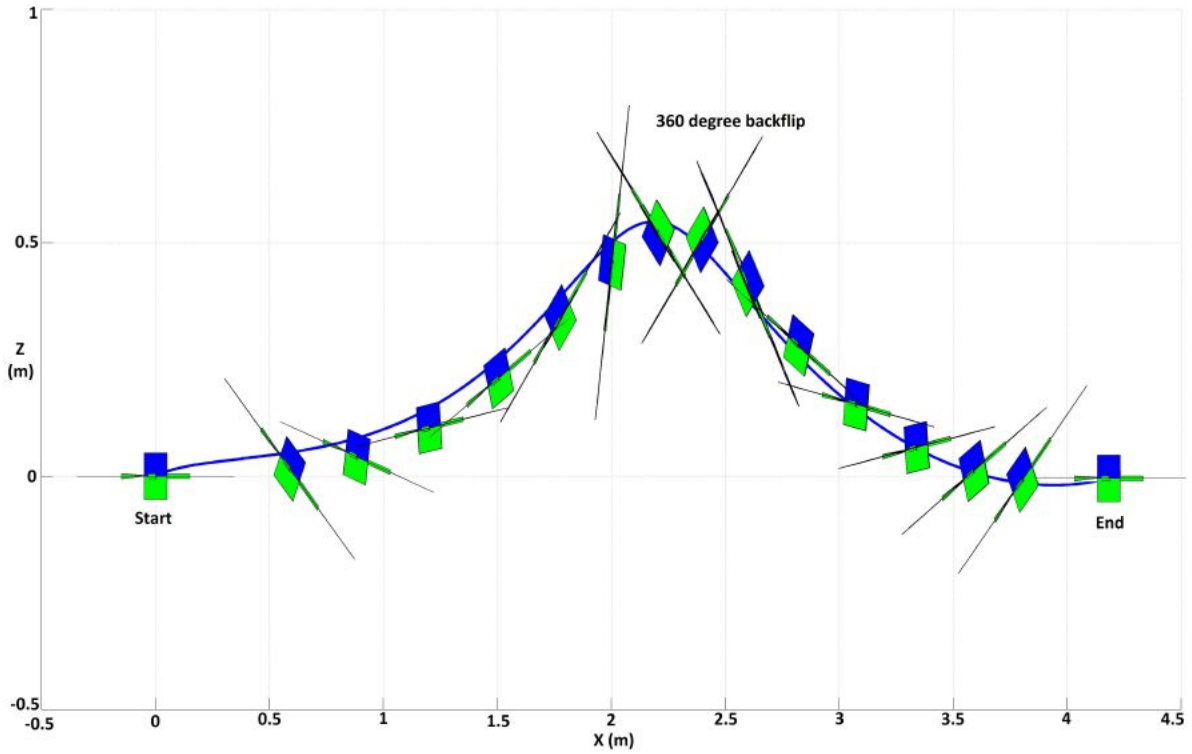
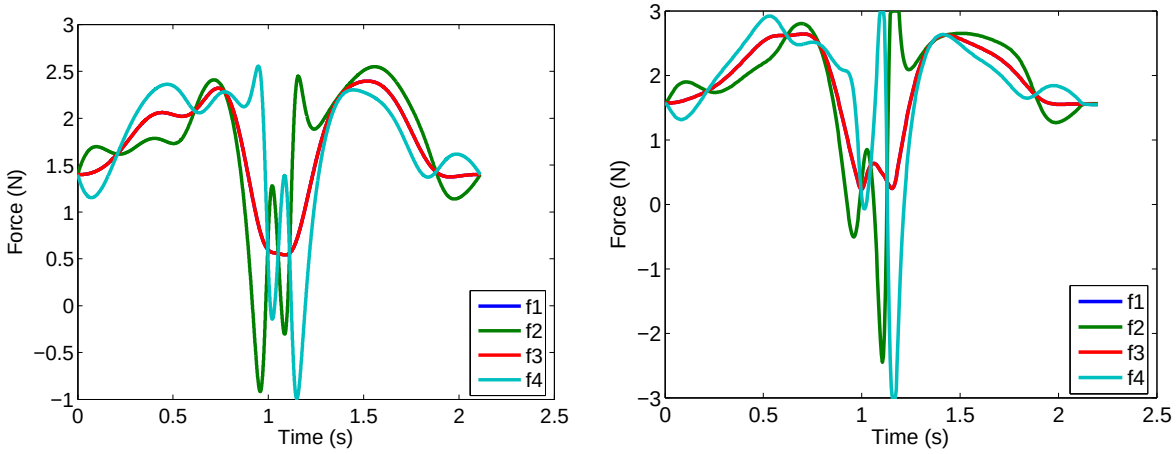


Figure 3. Simulation results of a 360 degree backflip. The flip is specified using a -90 degree roll constraint before the peak of the trajectory and a 90 degree roll constraint after the peak. The quadrotor starts and ends in hover.



(a) Nominal motor values

(b) Actual motor values

Figure 4. Example motor data from the backflip presented in Fig. 3. Fig. 4(a) shows the anticipated motor commands assuming open-loop, perfect tracking. These are the commands used by the optimizer in Section IV.A to find minimum-time trajectories. Fig. 4(b) shows the corresponding actual motor commands when following the trajectory in simulation.

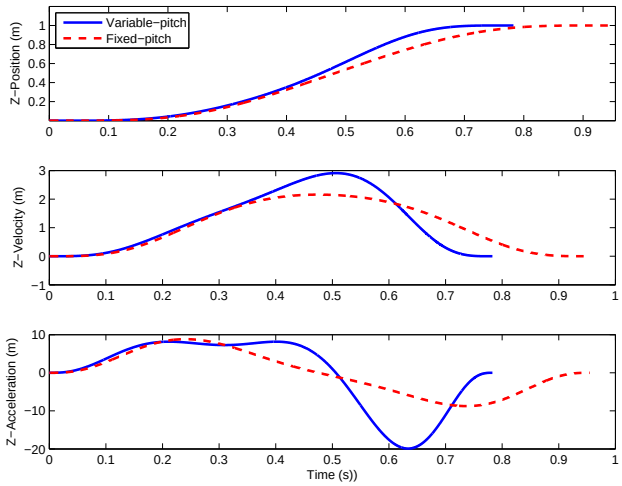


Figure 5. Two example vertical flight trajectories computed using the optimization routine in Section IV.A. Both trajectories have the same upper bound on motor thrust. The variable-pitch trajectory has a negative thrust lower bound, but the fixed-pitch trajectory has a lower bound of near zero. Note that the variable-pitch trajectory is shorter because it decelerates faster than gravity.

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.

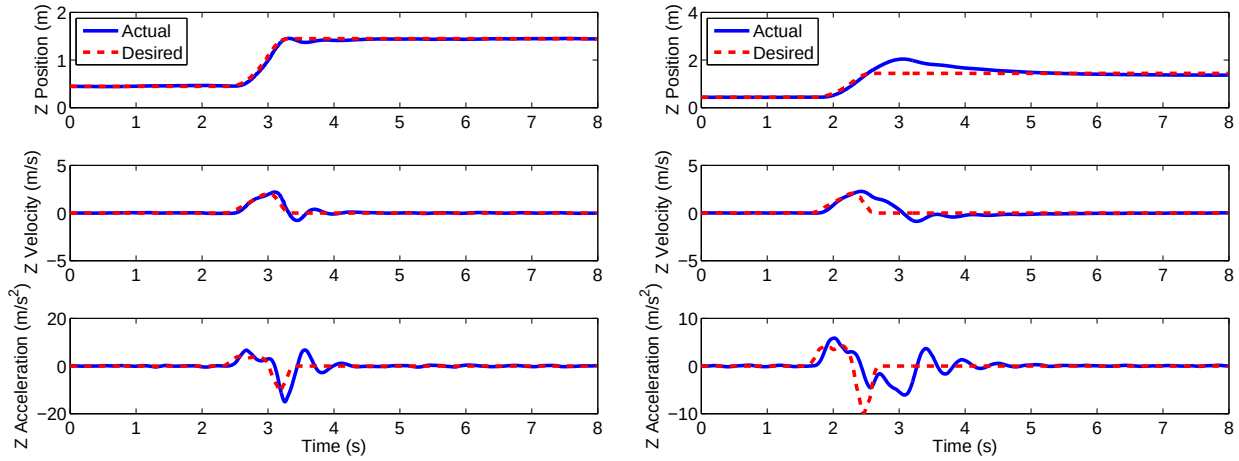
As mentioned in Section III, the attitude is not well defined from Eq. 12 when $\|\mathbf{F}^b\| = 0$ (the vehicle is in free-fall). However, interesting attitude maneuvers can be constructed by imposing an instantaneous free-fall constraint. In particular, Figure 2 shows the trajectory generated by 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 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.[9] Simulation results of the path are presented in Figures 3-4. 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 4 shows how the ideal motor commands compare to those actually generated in the simulation.

V. Experimental Results

The control and trajectory generation techniques are implemented on the Aerospace Controls Laboratory's variable-pitch quadrotor.[16,17] The quadrotor uses a Vicon[26] motion capture system for tracking the position and attitude of the vehicle, and on-board rate gyros for measuring the angular rate. A custom autopilot performs on-board attitude control at 1KHz while control reference commands are sent to the quadrotor from an off-board computer at 100Hz. See [27] for a more detailed design description of both the hardware and the software infrastructure.

In terms of the trajectory generation algorithm presented in Section IV, the variable-pitch quadrotor is advantageous because the addition of negative thrust more than doubles the effective thrust range for each of the 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 previous work,[16,17] variable-pitch propellers also increase the available controller bandwidth by effectively cancelling the motor actuator



(a) Variable-pitch flight data

(b) Fixed-pitch flight data

Figure 6. Flight data for the variable-pitch quadrotor flying the same trajectory in variable-pitch mode 6(a) and in fixed-pitch mode 6(b). The variable-pitch propellers allow for faster decelerations and better tracking of the position reference command.

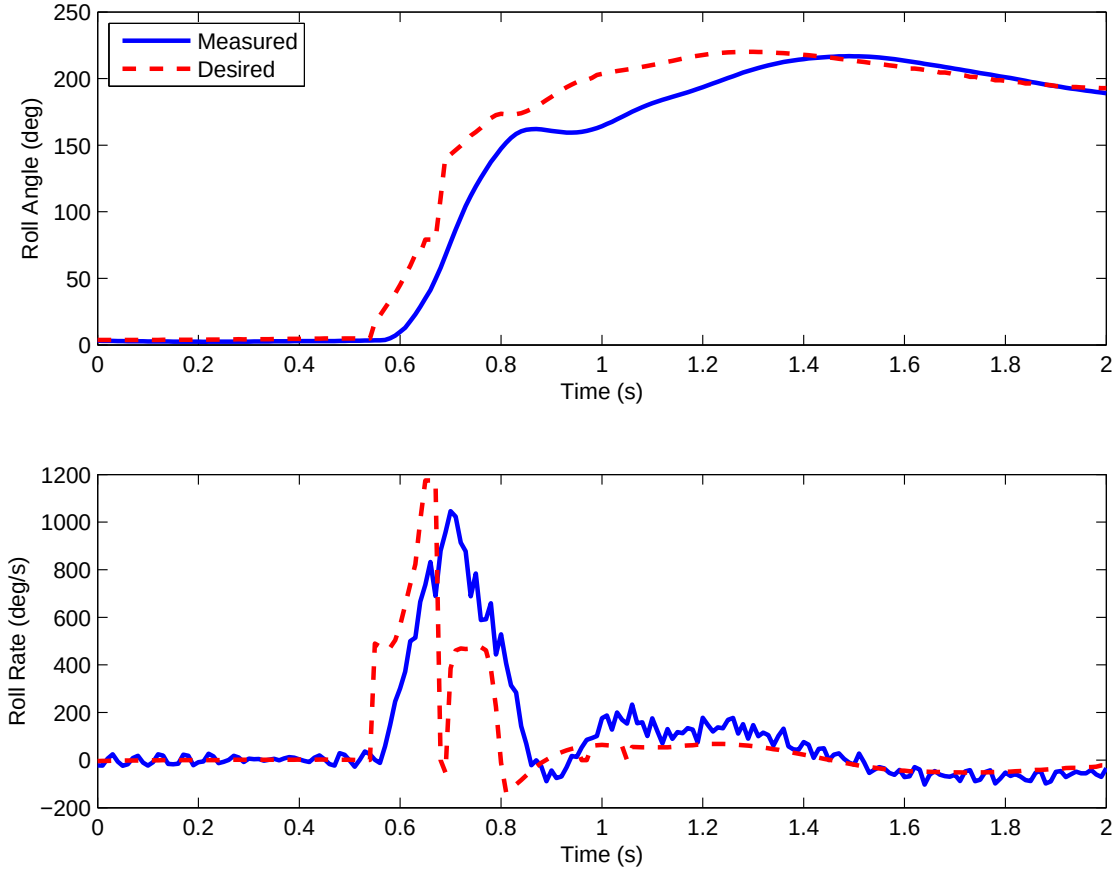
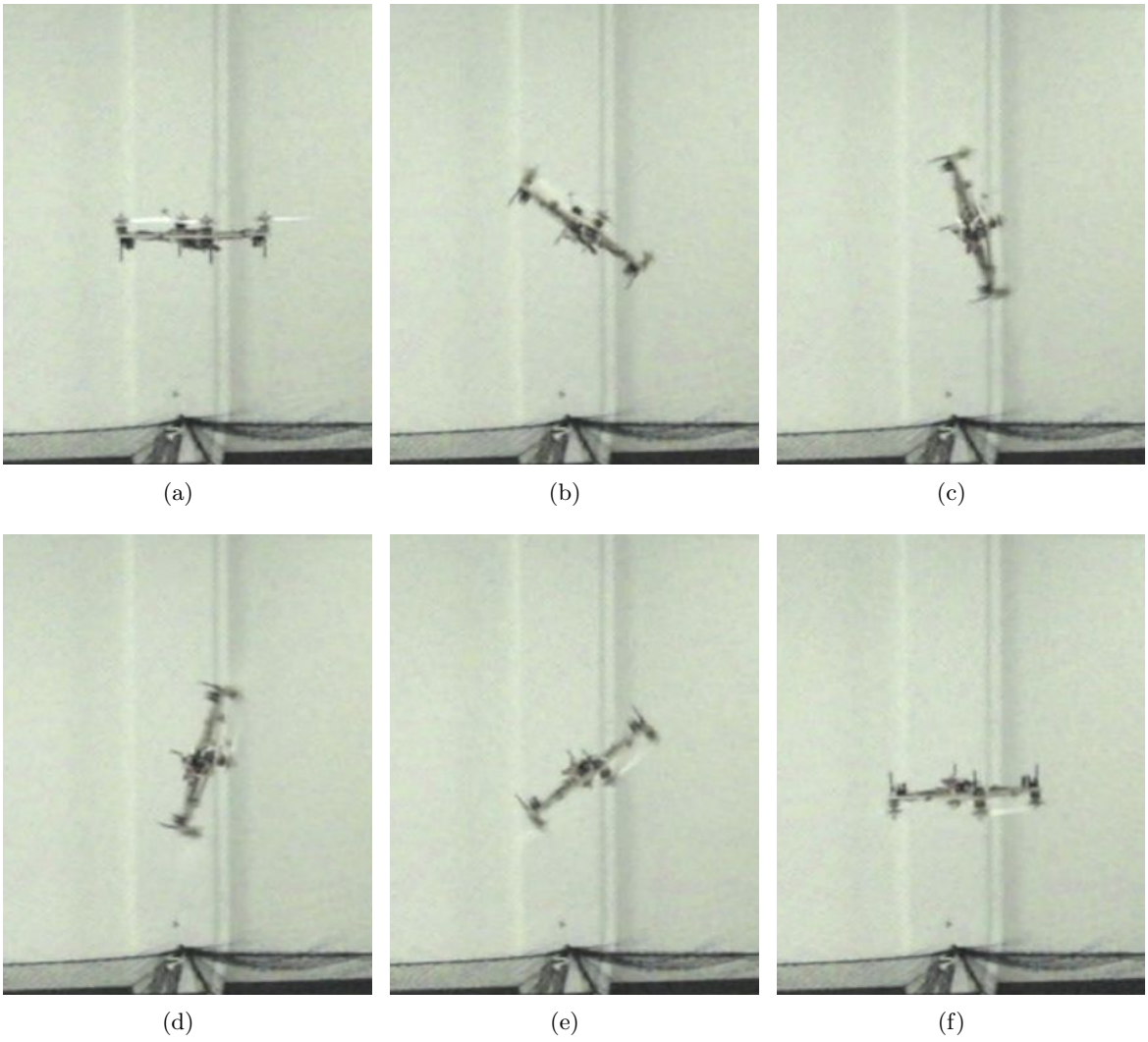


Figure 7. Commanded and measured roll and roll rate values from the quadrotor following a flipping maneuver. The measured values come from the on board rate gyros. The flip takes less than 0.4 seconds to complete. Snapshots of the quadrotor during the flip are shown in Fig. 8.



(a) (b) (c)

(d) (e) (f)

Figure 8. Variable-pitch quadrotor performing 180 degree flip by embedding a 90 degree roll constraint at the top of an arc in the X-Z plane.

dynamics. The variable-pitch propellers are thus able to change thrust substantially faster than corresponding fixed-pitch propellers.

Figure 5 shows an example trajectory found using the optimization routine presented in Section IV.A. 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 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 5 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.

Figures 6(a) and 6(b) show the tracking ability of the variable-pitch quadrotor. In both figures, the reference commands are the same; however, in variable-pitch mode the quadrotor tracks the reference position command with only 1% overshoot compared to 60% overshoot in fixed-pitch mode. The improved tracking performance in Figure 6(a) is due primarily to the large negative accelerations that are achieved only when the pitch of the propellers is allowed to vary.

Figure 7 shows the angular position and rate tracking abilities of the variable-pitch quadrotor.

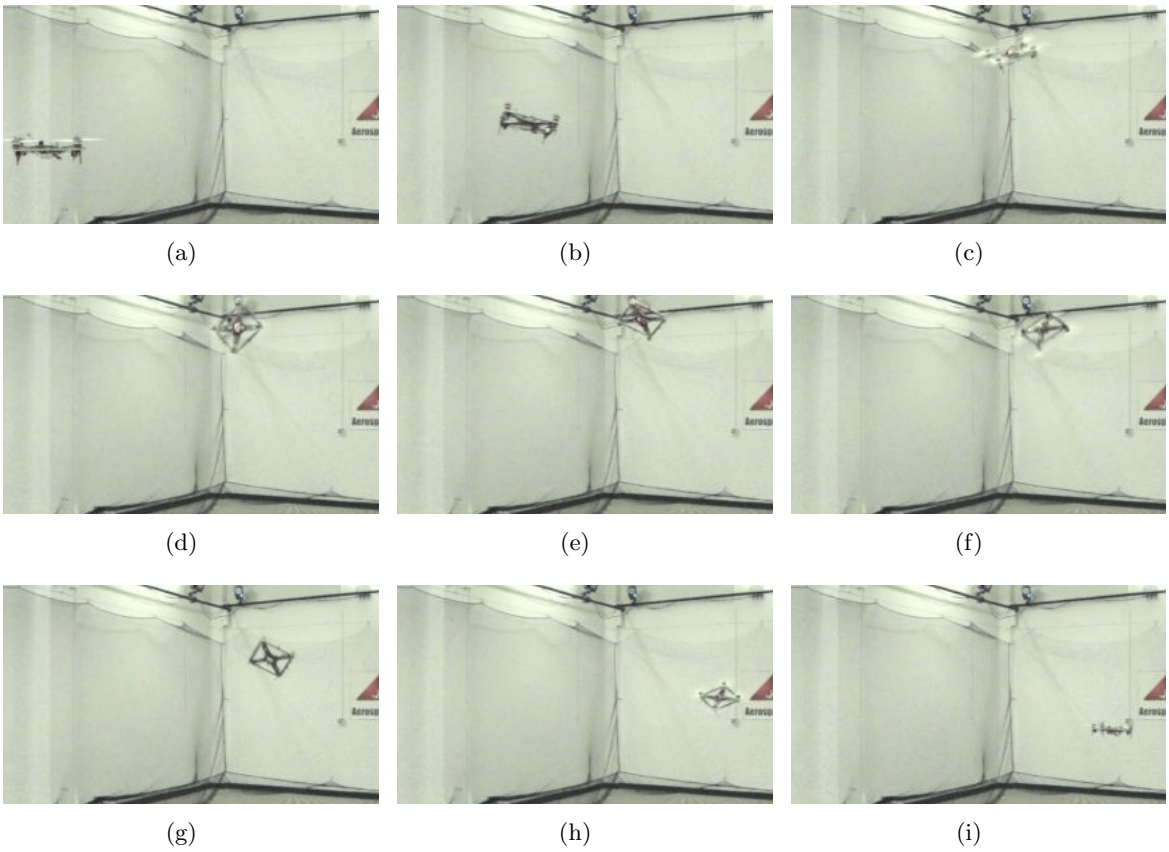


Figure 9. Variable-pitch quadrotor performing a 360 degree translating backflip. Simulations of this backflip are shown in Fig. 3. This maneuver was inspired by the Stanford STARMAC project.

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 8 shows snapshots of the variable-pitch quadrotor performing the flip.

Finally, snapshots of hardware results of the STARMAC-inspired backflip (simulation results shown in Figure 3) are shown in Figure 9. Videos of the flight experiments can be found at `http://www.youtube.com/user/AerospaceControlsLab`.

VI. Conclusion

This work presented 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 was 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 and in hardware on a custom variable-pitch quadrotor built in the Aerospace Controls Lab.

Acknowledgments

This paper is based upon work supported by the National Science Foundation Graduate Research Fellowship under Grant No. 0645960. The authors also acknowledge Boeing Research & Technology for support of the RAVEN [28] indoor flight facility in which the flight experiments were conducted.

References

[1]Amir, M. and Abbass, V., "Modeling of Quadrotor Helicopter Dynamics," *International Conference on Smart Manufacturing Application (ICSMA 2008)*, April 2008, pp. 100 –105.

[2]Erginer, B. and Altug, E., "Modeling and PD Control of a Quadrotor VTOL Vehicle," *IEEE Intelligent Vehicles Symposium*, June 2007, pp. 894 –899.

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

[4]Kim, J., Kang, M., and Park, S., "Accurate Modeling and Robust Hovering Control for a Quad–rotor VTOL Aircraft," *Journal of Intelligent and Robotic Systems*, Vol. 57, No. 1, 2010, pp. 9–26.

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

[6]Huang, H., Hoffmann, G., Waslander, S., and Tomlin, C., "Aerodynamics and control of autonomous quadrotor helicopters in aggressive maneuvering," *IEEE International Conference on Robotics and Automation (ICRA)*, May 2009, pp. 3277 –3282.

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

[8]Michael, N., Mellinger, D., Lindsey, Q., and Kumar, V., "The GRASP Multiple Micro-UAV Testbed," *IEEE Robotics & Automation Magazine*, Vol. 17, No. 3, 2010, pp. 56–65.

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

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

[11]Mellinger, D. and Kumar, V., "Minimum Snap Trajectory Generation and Control for Quadrotors," *IEEE International Conference on Robotics and Automation (ICRA)*, 2011.

[12]Hehn, M. and D'Andrea, R., "Quadrocopter Trajectory Generation and Control," *World Congress*, Vol. 18, 2011, pp. 1485–1491.

[13]Turpin, M., Michael, N., and Kumar, V., "Trajectory Design and Control for Aggressive Formation Flight with Quadrotors," *Proc. of the Intl. Sym. of Robot. Research. Flagstaff, AZ*, 2011.

[14]Abbeel, P., Coates, A., Quigley, M., and Ng, A. Y., "An application of reinforcement learning to aerobatic helicopter flight," *Advances in Neural Information Processing Systems (NIPS)*, MIT Press, 2007, p. 2007.

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

[16]Michini, B., Redding, J., Ure, N. K., Cutler, M., and How, J. P., "Design and Flight Testing of an Autonomous Variable-Pitch Quadrotor," *IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, May 2011, pp. 2978 – 2979.

[17]Cutler, M., Ure, N. K., Michini, B., and How, J. P., "Comparison of Fixed and Variable Pitch Actuators for Agile Quadrotors," *AIAA Guidance, Navigation, and Control Conference (GNC)*, Portland, OR, August 2011, (AIAA-2011-6406).

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

[19]Markley, F., "Fast quaternion attitude estimation from two vector measurements," *Journal of Guidance, Control, and Dynamics*, Vol. 25, No. 2, 2002, pp. 411–414.

[20]Chaturvedi, N., Sanyal, A., and McClamroch, N., "Rigid-body attitude control," *Control Systems, IEEE*, Vol. 31, No. 3, 2011, pp. 30–51.

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

[22]Michini, B., *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.

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

[24]How, J. P., Frazzoli, E., and Chowdhary, G., *Handbook of Unmanned Aerial Vehicles*, chap. Linear Flight Contol Techniques for Unmanned Aerial Vehicles, Springer, 2012.

[25]Chowdhary, G., Frazzoli, E., How, J. P., and Lui, H., *Handbook of Unmanned Aerial Vehicles*, chap. Nonlinear Flight Contol Techniques for Unmanned Aerial Vehicles, Springer, 2012.

[26]"Motion Capture Systems from Vicon," 2011, 14 Minns Business Park, West Way, Oxford OX2 0JB, UK `http://www.vicon.com/`.

[27]Cutler, M., *Design and Control of an Autonomous Variable-Pitch Quadrotor Helicopter*, Master's thesis, Massachusetts Institute of Technology, Department of Aeronautics and Astronautics, August 2012.

[28]How, J. P., Bethke, B., Frank, A., Dale, D., and Vian, J., "Real-Time Indoor Autonomous Vehicle Test Environment," *IEEE Control Systems Magazine*, Vol. 28, No. 2, April 2008, pp. 51–64.