A Fully Quaternion based Nonlinear Attitude and Position Controller

Hardik Parwana[1]
Kyoto University, Japan

Jay Patrikar[2] and Mangal Kothari[3]
Indian Institute of Technology Kanpur, India

In this paper, a novel fully quaternion based second order stable error dynamics controller is developed to address the attitude tracking problem in general and the position tracking problem of fixed pitch and variable pitch quadrotors in particular. Since the singularity associated with Euler angle representation is avoided using a unit quaternion approach, our algorithm finds its advantage in controlling highly maneuverable systems like variable pitch quadrotor which are capable of generating negative thrust and transition to inverted flight. Our controller is also well suited for rigid systems like satellites which require tracking of large-angle amplitude trajectories. Numerical simulations show good tracking performance using the proposed control methods for satellites and quadrotors. Experimental results on commercially available fixed pitch quadrotor using open-source architecture validate the novel attitude controller.

[1] Research Student, Dept. of Mechanical Engineering and Science, Kyoto University, Japan, parwana.hardik.22r@st.kyoto-u.ac.jp
[2] Undergraduate Student, Dept. of Aerospace Engineering, Indian Institute of Technology Kanpur, India, jaypat@iitk.ac.in.
[3] Assistant Professor, Dept. of Aerospace Engineering, Indian Institute of Technology Kanpur, India, mangal@iitk.ac.in, AIAA member

Nomenclature

$\mathcal{B}$	= Body (local) frame
$\mathcal{E}$	= Earth (global) frame
$\vec{w}$	= body angular velocity, rad/s
$\boldsymbol{J}$	= Moment of inertia tensor, kgm^2
$\vec{M}$	= moment, Nm
$\vec{r}$	= center of mass position in World frame, m
$\vec{F}$	= force produced by rotors, N
i	= (superscript) Inertial frame
b	= (superscript) Body frame
i	= (subscript) rotor number
ρ	= density of air, kg/m^3
A	= rotor disk area, m^2
Ω	= rotor blade rotation angular speed, rad/s
V_{tip}	= rotor tip speed, m/s
C_{T_i}	= non-dimensional thrust coefficient of i^{th} rotor
C_{Q_i}	= non-dimensional torque coefficient of i^{th} rotor
C_{l_α}	= lift curve slope, rad^{-1}
θ_{0_i}	= blade collective pitch angle of i^{th} rotor, rad
$C_{d_{0_i}}$	= zero lift drag coefficient of airfoil of i^{th} rotor
R	= rotor blade radius, m
c	= chord length of rotor blade
N_b	= number of blades in rotor
λ_i	= inflow factor of i^{th} rotor
d	= moment arm of Quadrotor rotors, m
T	= thrust of quadrotor, N
$[l\ m\ n]^T$	= X, Y, and Z component of moment in $\mathcal{B}$ frame, Nm
q	= quaternion
$\phi\ \theta\ \psi$	= Roll, Pitch and Yaw angles, rad
ζ	= $diag(\zeta_1, \zeta_2, \zeta_3)$ Gain matrix
ω_n	= $diag(\omega_{n_1}, \omega_{n_2}, \omega_{n_3})$ Gain matrix
d	= desired quantity
e	= error quantity
m	= mass of body, kg
g	= acceleration due to gravity, 9.81 m/s^2

I. Introduction

Designing smooth control inputs for rigid body attitude and translation dynamics has long been a problem in focus. Various approaches proposed in the past to fulfilling this objective can primarily be divided into two domains: one focusing on developing suitable control inputs, with PID being the most applied one, and the other focused on choosing better representation of state variables. Much of current research can be found combining these new approaches in both or either of these domains and studying its efficacy. The synthesis of control inputs involve linear control methods such as PID and its variants and nonlinear control methods such as Feedback Linearization, Nonlinear Dynamic inversion, and Sliding Mode and other adpative control theories. The translation dynamics representation has been done mostly in Euclidean space for Earth bound systems and in Polar or spherical coordinates for outer space systems. As for the orientation representation, the attitude of a rigid body belongs to the configuration space known as Special Orthogonal group SO(3) and is represented in most general terms as 3×3 rotation matrix [1]. The space can, however, be parameterized in terms of fewer parameters. The most used representation is Euler angle formulation. However, it suffers from gimbal lock which results in a singularity in the formulation. Recently quaternions and rotation matrix have become the preferred tool to overcome the singularity of Euler angles. The applications of these approaches have been validated on various systems such as satellites, aircrafts and more popularly on UAVs which, owing to their smaller size, also provide the advantage of a cheap experimental platform for conceptual verification.

Among these small UAVs, quadrotors have recently gained attention for their usefulness in multi-agent missions, mapping and localization, acrobatic performances, and as autonomous payload delivery systems. They are mechanically simple however, their inherent instability and nonlinear dynamics make their control challenging. Various control strategies have been designed which enable them to perform aggressive maneuvers such as multiple flips [2], flight through narrow openings [3], etc. Most of these approaches use hierarchical structures and divide the problem into two loops of position and attitude control.

Emil and George [4] proposed a nonlinear P^2 controller with the commanded moment proportional to the vector part of the quaternion error and actual angular velocity. Tayebi and McGilvray

[5] designed a similar PD^2 feedback structure for the hover equilibrium point stabilization based on quaternion, quaternion velocity and angular velocity.The stability analysis is given using a Lyapunov function designed in a manner similar to the backstepping approach. A quaternion based nonlinear robust feedback controller was proposed by Xian et al [6] in which they stabilize the second order error dynamics in attitude and altitude while using filters to estimate the angular velocities. Their controller however requires inversion of a Jacobian matrix which becomes singular if first component of error quaternion is zero. A similar controller was also designed by Bangura and Mahony [7] with moment as a function of quaternion and angular velocity error.

In this paper, we impose second order stable dynamics for the error quaternion in designing the attitude controller. Since we control second derivative instead of first, the control inputs are expected to be smoother than first order error dynamics which directly controls the first derivative or the direction of response. To the best of our knowledge most of the previous quaternion feedback controllers including those mentioned above are dependent only on rate of vector part of error quaternion vector for the feedback. However, it is our understanding that the scalar part can also also guide how fast the error dynamics converges. We also observed experimentally that during the gain tuning process the scalar part contribution can affect smoothness of the controller. Also, the derivative of quaternion has a nonlinear relationship with the angular velocity and hence our feedback, which is function of quaternion derivative rather than the angular velocity directly, is also unlike most of the previously proposed controllers which have angular velocity feedback.

Despite being successfully manipulated, the mechanical simplicity of fixed-pitch quadrotors impose some fundamental constraints on its performance. These do not scale well to large size as the stabilization of larger quadrotors become difficult through RPM control alone given the magnitudes of the torque involved and limited control bandwidth making them unsuitable for larger payloads. These limitations can be overcome using variable pitch quadrotors [8, 9].

Not much attention has been directed towards variable pitch quadorotors in recent history however, they have been a subject of research in the past. Georges de Bothezat and Ivan Jerome in 1922 built and flew the "Flying Octopus" [10], a vehicle with rotors located at each end of a truss structure of intersecting beams, placed in the shape of a cross and controlled through changing pitch angle of blades. Johann Borenstein [11] at University of Michigan developed "Hoverbot", which is the first documented effort at designing and flying a small scale quadrotor with variable pitch control, though it never achieved flight beyond tethered hovering. Recently, many hobbysists have demonstrated flights with small scale models. A detailed analysis of flight performance was however undertaken by Cutler et al. [8], and some important facts were established: (i) the ability to generate negative thrust could be used to perform inverted flight; and (ii) faster rate of change of thrust was observed with changing pitch than with changing RPM of motors. More recently, pointing to the fact that the linear mapping between moments in body frame and the actuator control does not work with variable pitch quadrotor due to nonlinear actuator dynamics involved with changing pitch angle of blades, Namrata et al [9] designed a control allocation methodology, which is also employed in this paper.

Our attitude controller holds in general for any rigid body and can be applied to many significant areas. For example, with the growth in number and operational requirements of rigid body spacecraft, systems often have a need for highly accurate pointing maneuvers in numerous tasks of practical importance like weather monitoring, communications, survey of resources etc in which there arises a need to maintain a satellite in a fixed orientation with respect to the Earth. B. T. Costic et al [12] have addressed the attitude tracking problem for a rigid spacecraft using unit quaternion based adaptive approach that obtains asymptotic attitude tracking without the need for angular velocity measurements. In this paper, we present simulation results by implementing the proposed second order error dynamics quaternion based attitude controller for achieving the desired satellite orientation.

The rest of the paper is organized as follows: A brief introduction to quaternion formulations is presented in Section II. Section III presents the dynamic models of various rigid bodies for whom the controller is being designed. Section IV describes the attitude controller, which in general holds for any rigid body, and Section V describes the complete trajectory tracking algorithm for the quadrotor. Simulation results are then presented in Section VI and experimental results for the attitude and position controller are presented in Section VII. Finally, we state some important conclusions in Section VIII.

II. Quaternion Preliminaries

Quaternions [13, 14] are hyper complex number of rank 4 which consists of a scalar part q_0 and vector part $\vec{q}$

$$\mathbf{q} = \begin{bmatrix} q_0 \\ \vec{q} \end{bmatrix} = [q_0 \ q_1 \ q_2 \ q_3]^T \tag{1}$$

When using them to represent the attitude of a body, they have the property of unity norm, i.e., $|\mathbf{q}| = 1$. The quaternion $\mathbf{q}$ in this paper will be reserved for representing attitude except when stating identities which may hold for more general cases. We will be following the Hamilton convention of quaternion, according to which the transformation between body and world frame is given by

$$\mathbf{x}^i = \mathbf{q} \circ \mathbf{x}^b \circ \mathbf{q}^* \tag{2}$$

where $\mathbf{x} = \left[0 \ \vec{x}^T\right]^T$ is the quaternion corresponding to any vector $\vec{x}$. Throughout the paper the bold counterpart of any vector will represent its corresponding quaternion. The quaternion product operator $\circ$ is defined as

$$\mathbf{q} \circ \mathbf{p} = \begin{bmatrix} p_0 q_0 - \vec{p} \cdot \vec{q} \\ q_0 \vec{p} + p_0 \vec{q} + \vec{q} \times \vec{p} \end{bmatrix} \tag{3}$$

The rotation matrix $\mathbf{R}_{\mathcal{B}}^{\mathcal{E}} = (R(\mathbf{q}))$, transforming a vector from body from local body frame $\mathcal{B}$ to global Earth frame $\mathcal{E}$ is then given by

$$R(\mathbf{q}) = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1 q_2 - q_0 q_3) & 2(q_0 q_2 + q_1 q_3) \\ 2(q_1 q_2 + q_0 q_3) & (q_0^2 - q_1^2 + q_2^2 - q_3^2) & 2(q_2 q_3 - q_0 q_1) \\ 2(q_1 q_3 - q_0 q_2) & 2(q_0 q_1 + q_2 q_3) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix} \tag{4}$$

The rate of change of quaternion $\dot{q}$ can be related to the angular velocity $\vec{w}$. For the purpose of representing the attitude dynamics, let $\mathbf{w}$ be the quaternion for $\vec{\omega}$, then the time derivative of quaternion, called its rate, is given by

$$\dot{\mathbf{q}} = \frac{1}{2} \mathbf{q} \circ \mathbf{w} \tag{5}$$

Note that in the above equation the angular velocity is expressed in the local body frame of reference.

III. Dynamic Modelling for Rigid Systems

In this section, we state the dynamical model for a satellite. The dynamics of a fixed-pitch quadrotor is also discussed which is finally extended to a variable pitch quadrotor.

A. Satellite Attitude Dynamics

The actuators present in a spacecraft can generate torque in the body frame $\mathcal{B}$ about all the three axis by the application of a pair of forces opposite to each other but equal in magnitude and acting in a direction normal to the line joining the actuators [12]. Given the attitude quaternion $\mathbf{q}$, the attitude dynamics of satellite is governed by the equations:

$$\mathbf{J}\dot{\vec{\omega}} = -\vec{\omega} \times \mathbf{J}\omega + \vec{M} \tag{6}$$

$$\dot{\mathbf{q}} = \frac{1}{2}\mathbf{q} \circ \mathbf{w} \tag{7}$$

where $\vec{M}$ is generated by the actuators present on the spacecraft. In our work, we will not be dealing with position control of spacecrafts and hence the related dynamics are not mentioned here.

B. Fixed-Pitch and Variable-Pitch Quadrotor Dynamics

The Newton-Euler equations of motion for a rigid body can be used to describe the motion of Quadrotor in a three dimensional space. The axis systems chosen are as shown in Fig 1 with the body and inertial world frame Z axis directed downwards. $\vec{F}^b$ and $\vec{F}^i$ represent the components of $\vec{F}$ in $\mathcal{B}$ and $\mathcal{E}$ frame respectively. Note that $\vec{F}^b$ acts along the body z axis, i.e., $\vec{F}^b = [0\ 0\ f]^T$ where f is the total thrust produced by the rotors. The gravitational force is taken to act along the positive z direction of inertial frame through the center of mass of the symmetrical quadrotor. The dynamics are then given by [15]:

$$\ddot{\vec{r}} = \frac{1}{m}\vec{F}^i - \vec{g} \tag{8}$$

$$\mathbf{J}\dot{\vec{\omega}} = \vec{M}^b - \vec{\omega} \times \mathbf{J}\vec{\omega} \tag{9}$$

where $\vec{\omega}$ is the angular velocity in the body frame and $\mathbf{J}$ is the moment of inertia tensor in the body axis system. Given a rotation matrix, $\mathbf{R}_{\mathcal{B}}^{\mathcal{E}}$, which operates from Body frame to inertial Earth

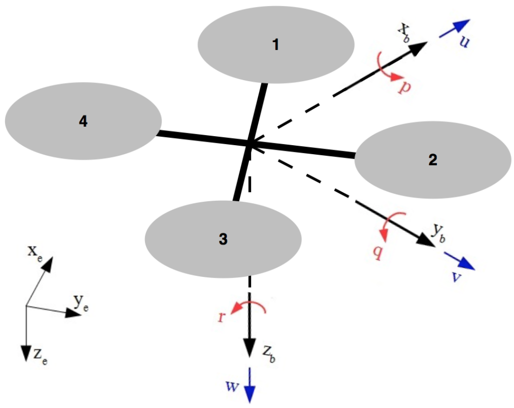


Fig. 1: Inertial and body frame of reference

frame, it can be established that $\vec{F}^i = R_{\mathcal{B}}^{\mathcal{E}} \vec{F}^b$.

Equations of Motion

The translation and attitude kinematics and dynamics from Eqs.(8) and (9) can be recast using the quaternion method of attitude representation as

$$\ddot{\mathbf{r}} = \frac{1}{m}\mathbf{q} \circ \mathbf{F}^b \circ \mathbf{q}^* - \mathbf{g} \tag{10}$$

$$\mathbf{J}\dot{\vec{\omega}} = \vec{M}^b - \vec{\omega} \times \mathbf{J}\vec{\omega} \tag{11}$$

$$\dot{\mathbf{q}} = \frac{1}{2}\mathbf{q} \circ \mathbf{w} \tag{12}$$

Rotor Dynamics

The control input and actuation is fundamentally of different type in a variable pitch quadrotor than the fixed-pitch quadrotor. We choose to provide a fixed RPM and vary the pitch of the propellers in order to change the thrust. Some previous works [8] have already elaborated on the advantages of using this mode of actuation and thus the in-depth details are excluded here.

The thrust and torque of each rotor can be obtained from Blade Element Momentum Theory (BEMT) as a function of the thrust coefficient. Given ρ, A, Ω, and hence the rotor tip speed $V_{tip} = \Omega R$, the non-dimensional coefficients are defined in the usual way as $C_T = T/(\rho A V_{tip}^2/2)$ and $C_Q = Q/(\rho A R V_{tip}^2/2)$. Following the same approach as in work done by Namrata et al. [9], the non-dimensional coefficients for each rotor are defined in terms of rotor parameters as

$$C_{T_i} = \frac{1}{2}\sigma C_{l\alpha}\left(\frac{\theta_{0_i}}{3} - \frac{\lambda_i}{2}\right) \tag{13}$$

$$C_{Q_i} = \frac{1}{2}\sigma\left(\frac{\lambda_i C_{l\alpha}\theta_{0_i}}{3} - \frac{\lambda_i^2 C_{l\alpha}}{2} + \frac{C_{d0_i}}{4}\right) \tag{14}$$

σ is defined in terms of N_b, c, and R as $\sigma = \frac{N_b c}{\pi R}$. The only unknown in the above equations is the inflow factor, λ_i, which can be approximated from the momentum theory for hovering flight condition as

$$\lambda_i = \sqrt{\frac{C_{T_i}}{2}} \tag{15}$$

The above substitution gives us the collective pitch input to the rotor and the torque coefficient for each rotor as function of C_T

$$\theta_{0_i} = \frac{6C_{T_i}}{\sigma C_{l\alpha}} + \frac{3}{2}\sqrt{\frac{C_{T_i}}{2}} \tag{16}$$

$$C_{Q_i} = \frac{1}{2}\sigma\left(\frac{\sqrt{2}C_{T_i}^{3/2}}{\sigma} + \frac{C_{d0_i}}{4}\right) \tag{17}$$

The thrust and the torque for the i^{th} rotor are given by

$$T_i = KC_{T_i} \tag{18}$$

$$Q_i = KRC_{T_i} \tag{19}$$

where $K = \rho\pi R^2 V_{tip}^2$ which is a constant if the RPM is maintained constant. Contrary to most of the previous literature, we let C_Ts be negative if the thrust produced by the rotor is in the downward direction in body frame.

$$\begin{aligned} T &= (T_1 + T_2 + T_3 + T_4) \\ &= K(C_{T_1} + C_{T_2} + C_{T_3} + C_{T_4}) \end{aligned} \tag{20}$$

The rolling and pitching moments can be obtained from contributions of each rotor. Due to the relative sense of rotations, rotors 1 and 3 generate torque along the positive body z axis while rotors 2 and 4 generate torque about negative body z axis. The final expression for total moments acting on the quadrotor are as shown in equations below.

$$l = d \times K(C_{T_1} - C_{T_2} - C_{T_3} + C_{T_4}) \tag{21}$$

$$m = Kd(C_{T_1} + C_{T_2} - C_{T_3} - C_{T_4}) \tag{22}$$

$$n = \frac{KR}{\sqrt{2}}(|C_{T_1}|^{3/2} - |C_{T_2}|^{3/2} + |C_{T_3}|^{3/2} - |C_{T_4}|^{3/2}) \tag{23}$$

IV. Attitude Control

In this section, we formulate a controller to drive the vehicle configuration towards the attitude setpoint computed by the position controller or passed on by the user. We impose second order error stabilizing dynamics for achieving the desired attitude. The quaternion error is defined as:

$$\mathbf{q}_e = \mathbf{q}_d^* \circ \mathbf{q} \tag{24}$$

where $\mathbf{q}^*$ denotes the conjugate or inverse of the unit quaternion. It can be interpreted as the rotation corresponding to transformation from local to desired frame of reference. The equilibrium point for the error quaternion is $\mathbf{q}_e = [1\ 0\ 0\ 0]^T$ which corresponds to perfect overlap between desired and actual orientations. The second order dynamics imposed on each component of $\mathbf{q}_e$ separately can be written together in vector form as:

$$\ddot{\mathbf{q}}_e + 2\zeta\omega_n\dot{\mathbf{q}}_e + \omega_n^2\left(q_e - \begin{bmatrix}1\\0\\0\\0\end{bmatrix}\right) = 0 \tag{25}$$

where ζ and ω_n are diagonal matrices given by:

$$\zeta = \begin{bmatrix}\zeta_1 & 0 & 0 & \\ 0 & \zeta_2 & 0 & 0\\ 0 & 0 & \zeta_3 & 0\\ 0 & 0 & 0 & \zeta_4\end{bmatrix},\ \omega = \begin{bmatrix}\omega_1 & 0 & 0 & \\ 0 & \omega_2 & 0 & 0\\ 0 & 0 & \omega_3 & 0\\ 0 & 0 & 0 & \omega_4\end{bmatrix}$$

Quaternion Error Dynamics

From $\mathbf{q} = \mathbf{q}_d \circ \mathbf{q}_e$, we get

$$\dot{\mathbf{q}}_e = \mathbf{q}_d^* \circ (\dot{\mathbf{q}} - \dot{\mathbf{q}}_d \circ \mathbf{q}_e) \tag{26}$$

Using Eq.(5) for both $\mathbf{q}$ and $\mathbf{q}_d$, we get

$$\begin{aligned}\dot{\mathbf{q}}_e &= \mathbf{q}_d^* \circ (\frac{1}{2}\mathbf{q} \circ \mathbf{w} - \frac{1}{2}\mathbf{q}_d \circ \mathbf{w}_d^D \circ \mathbf{q}_e) \\ &= \frac{1}{2}\mathbf{q}_e \circ (\mathbf{w} - \mathbf{q}_e^* \circ \mathbf{w}_d^D \circ \mathbf{q}_e) \end{aligned} \tag{27}$$

where $\mathbf{w}_d$ is the desired angular velocity quaternion and D stands for desired quaternion frame. Noting that B denotes body frame and that by definition $\mathbf{q}_e$ operates on a vector in body (actual) quaternion frame and maps it to desired attitude frame, the above can be simplified in terms of angular velocity error $\tilde{w}$ as

$$\begin{aligned}\mathbf{w}_d^D &= q_e \circ \mathbf{w}_d^B \circ \bar{q}_e \\ \Longrightarrow \dot{\mathbf{q}}_e &= \frac{1}{2}\mathbf{q}_e \circ (\mathbf{w} - \mathbf{w}_d^B) = \frac{1}{2}\mathbf{q}_e \circ \tilde{\mathbf{w}} \end{aligned} \tag{28}$$

where $\tilde{\mathbf{w}}$ denotes error in angular velocity quaternion. The second derivative of quaternion would naturally involve angular acceleration terms. Writing Eq.(28) as $\tilde{\mathbf{w}} = 2\mathbf{q}_e^* \circ \dot{\mathbf{q}}_e$, we obtain

$$\begin{aligned}\dot{\tilde{\mathbf{w}}} &= 2\dot{\mathbf{q}}_e^* \circ \dot{\mathbf{q}}_e + 2\mathbf{q}_e^* \circ \ddot{\mathbf{q}}_e \\ &= 2(||\mathbf{q}_e||^2, 0) + 2\mathbf{q}_e^* \circ \ddot{\mathbf{q}}_e \end{aligned} \tag{29}$$

The vector part of the error quaternion, $\dot{\vec{\tilde{\omega}}}$ in $\dot{\tilde{\mathbf{w}}} = \left[0\ \vec{\tilde{\omega}}^T\right]^T$ is therefore

$$\begin{aligned}\dot{\vec{\tilde{\omega}}} &= 2\begin{bmatrix} -q_1 & q_0 & q_3 & -q_2 \\ -q_2 & -q_3 & q_0 & q_1 \\ -q_3 & q_2 & -q_1 & q_0 \end{bmatrix}_e \ddot{\mathbf{q}}_e \\ &= 2G\ddot{\mathbf{q}}_e \end{aligned} \tag{30}$$

where the matrix G is defined as above for convenience.

Controller

We can get the desired local angular acceleration from the error dynamics using Eqs.(30) and (25) as follows

$$\begin{aligned}\dot{\vec{\omega}}_D &= \dot{\vec{\tilde{\omega}}} + \dot{\vec{\omega}}_d^B \\ &= 2G(-2\zeta_i\omega_{n_i}\dot{\mathbf{q}}_e - \omega_{n_i}^2\left(\mathbf{q}_e - [1\ 0\ 0\ 0]^T\right)) + \dot{\vec{\omega}}_d^B \end{aligned} \tag{31}$$

where $\vec{\omega}_D$ is the desired angular velocity rate which is passed to the control allocation as the angular acceleration to be achieved. Note that in a satellite and fixed-pitch quadrotor, this would be the last step and the required moments and thrust can be obtained from the analysis till now which then, in case of quaternion, can be converted to required rotor speeds directly using the transformation matrix. However in case of variable pitch quadrotors, since we cannot directly find C_{T_i} due to the nonlinear relationship as seen in Eq.(23), we maintain an additional loop to achieve the angular velocity rate derived above. This will be elaborated more when we discuss variable pitch quadrotor.Finally note that in Eq.(31), the derivative of the desired angular velocity in body frame $\dot{\vec{w}}_d^B$ can be obtained using $\vec{\omega}_d^B = R(\mathbf{q}_e)^T \vec{\omega}_d^D$ and hence

$$\dot{\vec{\omega}}_d^B = \dot{R}^T \vec{\omega}_d^D + R^T \dot{\vec{\omega}}_d^D \tag{32}$$

A. Stability Analysis

The control input chosen satisfies the stable second order error dynamics [16] given by Eq.(25) and is thus known to be exponentially stable. We provide Lyapunov stability analysis for proving global stability of the control input. Consider a Lyapunov candidate function

$$\begin{aligned} V &= \alpha \vec{q}_e^T \vec{q}_e + \alpha (q_{e_0} - 1)^2 + \frac{1}{2} \vec{\tilde{\omega}} J \vec{\tilde{\omega}}^T \\ &= 2\alpha(1 - q_{e_0}) + \frac{1}{2} \vec{\tilde{\omega}} J \vec{\tilde{\omega}}^T \end{aligned} \tag{33}$$

where $\alpha > 0$ is a constant.

Theorem 1. *Assume that the desired attitude trajectory $\boldsymbol{q}_d$ is known, continuous and has bounded first and second order derivatives. Also, assume that the gain matrices ζ and ω_n are just multiples of identity matrices. Let $\alpha > 0$ be a scalar such that $\alpha = 2J\omega_n^2$. Then the second order error dynamics imposed on $\boldsymbol{q}_e$ in Eq.(25) and the resulting control law synthesized in Eq.(31) render the attitude dynamics almost globally asymptotically stable around the identity equilibrium error quaternion $[1\ 0\ 0\ 0]^T$ and corresponding zero angular velocity error $\vec{\tilde{\omega}} = [0\ 0\ 0]^T$.*

We now provide a proof of the theorem. The derivative of Lyapunov function can be obtained as

$$\dot{V} = -2\alpha \dot{q}_{e_0} + \frac{1}{2} \dot{\vec{\tilde{\omega}}} J \vec{\tilde{\omega}}^T + \frac{1}{2} \vec{\tilde{\omega}}^T J \dot{\vec{\tilde{\omega}}}^T \tag{34}$$

Since J is a positive definite matrix, the last two terms, both being scalar and transpose of each other, are equal. Therefore,

$$\begin{aligned}
\dot{V} &= -2\alpha \dot{q}_{e_0} + \vec{\omega}^T J \dot{\vec{\omega}}^T \\
&= \alpha \vec{q}_e^T \vec{\omega} + 2\vec{\omega} J G \left(-2\zeta\omega_n \dot{\mathbf{q}}_e - \omega_n^2 (\mathbf{q}_e - [1\ 0\ 0\ 0]^T) \right) \\
&= \alpha \vec{q}_e^T \vec{\omega} - 2\vec{\omega}^T J G \zeta \omega_n G^T \vec{\omega} - 2\vec{\omega} J G \omega_n^2 (\mathbf{q}_e - [1\ 0\ 0\ 0]^T)
\end{aligned} \tag{35}$$

where we have used the formula $\mathbf{q}_e \circ \tilde{\mathbf{w}} = G^T \vec{\omega}$. Now assuming ζ and ω_n to be diagonal matrices with equal diagonal entries respectively, we can replace their product by a scalar and denote it by Λ. Also, it can be easily verified that $GG^T = I$ and therefore the above equations reduce to:

$$\dot{V} = \alpha \vec{q}_e^T \vec{\omega} - 2\vec{\omega}^T J \Lambda \vec{\omega} - 2\vec{\omega} J G \omega_n^2 (\mathbf{q}_e - [1\ 0\ 0\ 0]^T) \tag{36}$$

Expanding the above equation using G matrix from Eq.(30), we can get $G(\mathbf{q}_e - [1\ 0\ 0\ 0]^T) = \vec{q}_e$ (Note that $G(\mathbf{q})\mathbf{q} = \vec{0}_{3\times 1}$). Finally using the assumption that $\alpha = 2J\omega_n^2$, we get

$$\begin{aligned}
\dot{V} &= \alpha \vec{q}_e^T \vec{\omega} - 2\vec{\omega}^T J \Lambda \vec{\omega} - 2\vec{\omega} J \omega_n^2 \vec{q}_e \\
&= -\vec{\omega} \Gamma_1 \vec{\omega}
\end{aligned} \tag{37}$$

where $\Gamma_1 = 2J\Lambda$. Since the above expression is negative semi definite, we can conclude that $\dot{V}$ is negative semi-definite. By differentiating Eq.(37) and using boundedness of input signals through control law, it can easily be shown that $\ddot{V}$ is bounded and thus $\dot{V}$ is uniformly continuous. Therefore by Barbalat's lemma [17], $\dot{V}$ goes to zero as $t \to \infty$ and $\vec{\omega}$ is globally asymptotically stable.

It can also be shown [7] using the boundedness of $\dot{\vec{\omega}}_d$, $\ddot{\vec{\omega}}_d$ and using the global asymptotically stable property of $\vec{\omega}$ as derived above, that $||\dot{\vec{\omega}}|| \to 0$ as $t \to \infty$. Now consider the error dynamics given by Eqs.(30). It can be further evaluated as:

$$\dot{\vec{\omega}} = 2G \left(-\zeta\omega_n \mathbf{q}_e \circ \tilde{\mathbf{w}} - \omega_n^2 \left(\mathbf{q}_e - [1\ 0\ 0\ 0]^T \right) \right) \tag{38}$$

$$= -2\zeta\omega_n \vec{\omega} - \omega_n^2 \vec{q}_e \tag{39}$$

where the steps are repeated as in derivation of $\dot{V}$. The above equation can be rearranged as

$$||\omega_n^2 \vec{q}_e|| \;=\; ||-\dot{\tilde{\vec{\omega}}} - 2\Lambda\tilde{\vec{\omega}}|| \leq ||\dot{\tilde{\vec{\omega}}}|| + ||2\Lambda\tilde{\vec{\omega}}|| \tag{40}$$

Since every term on right hand side goes to zero, it follows that $||\vec{q}_e|| \to 0$. This implies that $q_{e_o} \to \pm 1$. The asymptotic stability for $q_{e_0} = 1$ can be achieved by choosing a proper value of α so that the basin of attraction for given initial error quaternion is large and does not include the set $(\mathbf{q}_e(t_0)_{q_{e_0}=-1}, \tilde{\vec{\omega}}(t_0))$. For further analysis, consider an equivalent form of the above Lyapunov function as $V = 2(1 - q_{e_0}) + \frac{1}{2\alpha}\tilde{\vec{\omega}}^T J \tilde{\vec{\omega}}$. For $q_{e_0} \neq -1$, we can always choose α such that $V(\mathbf{q}_e(t_0), \tilde{\vec{\omega}}(t_0)) < 4$. Therefore it follows that

$$V(\mathbf{q}_e(t), \tilde{\vec{\omega}}(t)) < V(\mathbf{q}_e(t_0), \tilde{\vec{\omega}}(t_0)) < V([-1\ 0\ 0\ 0]^T, \vec{0}) \leq V([-1\ 0\ 0\ 0]^T, \tilde{\vec{\omega}}(t_0))$$

and hence $\mathbf{q}_e(t)$ does not go to $[-1\ 0\ 0\ 0]$. The system is therefore almost global asymptotically stable for this choice of α. Note that for the sake of simplicity in mathematical proof, the gain matrices ζ and ω_n have been assumed to be multiples of Identity matrix. However, the verification of results in simulation and in experiment has been done with dissimilar diagonal entries also. This has resulted in better performance for all practical purposes.

V. Trajectory Tracking

This section develops a Nonlinear Dynamic Inversion(NDI) [18, 19] based controller for the quadrotor. Unlike a fixed-pitch quadrotor, the controller of a variable pitch quadrotor has a wider range of actuation to choose from, since negative thrust can also be generated. This increases the range of feasible trajectories with higher accelerations than was possible before. However, the rotor dynamics complexity also increases and it does not allow us to directly allocate the control input from the moment requirements due to nonlinearities in the yawing moment term (Eq.(23)) and the fact that C_T does not remain constant for variable pitch quadrotors. We therefore propose a controller with one outer and one middle layer loop for position and attitude control, like most other present controllers have, and an additional third inner loop to dynamically allocate control to determine blade pitch angles of individual rotors. Note that the first two loops for fixed-pitch and variable pitch quadrotor largely remain the same. The following discussion thus holds valid for both of them and any point of difference will be mentioned at the appropriate place.

A. Outer Loop

Since the quadrotors are differentially flat [3], the four flat outputs, which are three spatial positions and the yaw angle, together completely determine the required four thrust command for rotors. Let $\vec{r}_d$ be the desired quadrotor position in the inertial frame and $\vec{e} = \vec{r} - \vec{r}_d$ be the error in position. In accordance with the NDI methodology, we choose second order error stabilizing dynamics for translation as

$$\ddot{\vec{e}} + 2\zeta_p \omega_{n_p} \dot{\vec{e}} + \omega_{n_p}^2 \vec{e} = 0 \tag{41}$$

where ζ and ω_n are design matrices. Using the fact that the only forces that act on quadrotor are net thrust force and gravity, the above equation can be used to compute the desired attitude and net thrust T.

$$\begin{aligned} \ddot{\vec{r}} &= \ddot{\vec{r}}_d + 2\zeta\omega_n(\dot{\vec{r}}_d - \dot{\vec{r}}) + \omega_n^2(\vec{r}_d - \vec{r}) \quad (42) \\ T_d &= m|\ddot{\vec{r}} - \vec{g}| \quad (43) \end{aligned}$$

where T_d is the desired thrust, $\vec{g} = [0\ 0\ g]$ and we have used the fact that only force acting on quadrotor besides the gravitational force is the rotor thrust. The desired quaternion now is the one that relates the desired thrust in body and inertial frame. Following on similar lines to work done by Cutler and How [8], we get from Eq.(12)

$$\mathbf{F}^i = m(\ddot{\mathbf{r}} - \mathbf{g}) = \mathbf{q}_{d'} \circ \mathbf{F}^b \circ \mathbf{q}_{d'}^* \tag{44}$$

where $q_{d'}$ is the desired quaternion that orients the thrust vector with the desired inertial frame force. It is equal to the minimum angle rotation between the two unit vectors $\hat{F}^i$ and $\hat{F}^b$ given by [20]:

$$q_{d'} = \frac{1}{\sqrt{2(1 + \hat{F}^b \cdot \hat{F}^i)}} \begin{bmatrix} 1 + \hat{F}^b \cdot \hat{F}^i \\ \hat{F}^b \times \hat{F}^i \end{bmatrix} \tag{45}$$

where $||\vec{F}^b|| = ||\vec{F}^i|| = T$. Note that $q_{d'}$ takes into account only the roll and pitch angles since any arbitrary yaw motion would still align the vectors in body and inertial frame. Also, $\hat{F}^b = [0\ 0\ \pm 1]$

depending on the direction of thrust vector. In the case of fixed-pitch quadrotor, we have only one direction and hence $\hat{F}^b = [0\ 0\ \ -1]$. However in case of variable pitch quadrotor we choose the direction of the thrust vector so that given the current orientation, the thrust has largest component along the desired force $\vec{F}^i$. Also, note that due to dual covering of attitude space, that is, since $\mathbf{q}$ and $-\mathbf{q}$ represent the same quaternion, there can be a discontinuity in the control law. Therefore we perform a check so that current time step quaternion is closest to the previous time step quaternion, that is, $\mathbf{q}_{d_{t-1}}^T \mathbf{q}_{d_t} \geq 0$. Finally since the desired yaw angle is given as input from trajectory generation algorithm, the desired quaternion can be obtained by complementing $q_{d'}$ with ψ_d in following way [8]:

$$q_d = q_{d'} \circ [\cos(\psi_d/2)\ 0\ 0\ \sin(\psi_d/2)] \tag{46}$$

B. Middle Loop

The output of the outer loop is passed to the attitude controller described in Section IV. The required moment can now be obtained from Eq.(12). Note that the desired angular velocity rate required in the attitude controller can be obtained numerically from desired attitude history. We follow the same procedure as done by Cutler [8] and hence omit it here.

C. Inner Loop

In this control allocation loop, we obtain the desired thrust coefficients. Note that unlike the fixed-pitch quadrotor, the $C_T s$ do not remain constant and hence the matrix which converts net thrust and moments to rotor speeds, and which has been used by most of the present literature is not valid.

Due to nonlinearities involved in the expression for yawing moment, C_{T_i} cannot be obtained explicitly. In order to perform this operation efficiently, we propose an additional loop that computes desired rate of change of C_{T_i}, which we call the virtual control input $U = \left[\dot{C}_{T_1}\ \dot{C}_{T_2}\ \dot{C}_{T_3}\ \dot{C}_{T_4}\right]$ that drives $[T\ l\ m\ n]$ to $[T_d\ l_d\ m_d\ n_d]$. We can formulate this using Eqs.(20-23) however we would need rate of change of thrust and moments. To this end, we again choose a second order stable error dynamics in terms of rate of change of angular acceleration for control input synthesis. The assumption, with same reasoning as the much followed approach of position and attitude loop, is that the inner loop dynamics are fast enough so that the middle loop stability is ensured.

$$\ddot{\vec{\omega}} = \ddot{\vec{\omega}}_d^B + 2\zeta_{CA}\omega_{n_{CA}}(\dot{\vec{\omega}}_d^B - \dot{\omega}) + \omega_{n_{CA}}\omega_{n_{CA}}^T(\vec{\omega}_d^B - \vec{\omega}) \tag{47}$$

where $\vec{w}_d^B$ and $\dot{\vec{w}}_d^B$ are the outputs of the middle and outer loop. For simplicity, we take $\ddot{\vec{w}}_d^B$ to be zero. The moment rate is now obtained by taking time derivative of Eq.(12)

$$\dot{\vec{M}} = \mathbf{J}\ddot{\vec{\omega}} + \dot{\vec{\omega}} \times \mathbf{J}\vec{\omega} + \vec{\omega} \times \mathbf{J}\dot{\vec{\omega}} \tag{48}$$

For thrust derivative we choose first order error dynamics with a gain $k_p > 0$ as

$$\dot{T} = k_p(T_d - T) \tag{49}$$

where T_d is obtained from position control loop. Using Eqs.(48) and (49), and taking derivative of Eqs.(21, 22, 23), we get

$$\begin{bmatrix} \dot{C}_{T_1} \\ \dot{C}_{T_2} \\ \dot{C}_{T_3} \\ \dot{C}_{T_4} \end{bmatrix} = \begin{bmatrix} K & K & K & K \\ Kd & -Kd & -Kd & Kd \\ Kd & Kd & -Kd & -Kd \\ \frac{3KR}{2}\sqrt{\frac{|C_{T_1}|}{2}} & -\frac{3KR}{2}\sqrt{\frac{|C_{T_2}|}{2}} & \frac{3KR}{2}\sqrt{\frac{|C_{T_3}|}{2}} & -\frac{3KR}{2}\sqrt{\frac{|C_{T_4}|}{2}} \end{bmatrix}^{-1} \begin{bmatrix} \dot{T} \\ \dot{\vec{M}} \end{bmatrix} \tag{50}$$

Once the virtual input U is obtained in the above way, it is integrated with the system dynamics to obtain the coefficients at any time given the initial coefficients.

VI. Simulation Results

In this section, we present numerical results to show the effectiveness of the proposed algorithm. The performance is demonstrated by three examples. First, we show results for attitude tracking by taking example of a sample satellite attitude history. Next, we show results for trajectory tracking of a fixed-pitch fixed pitch quadrotor. Finally to demonstrate the extended flight regime associated with a variable pitch quadrotor, we present an interesting example.

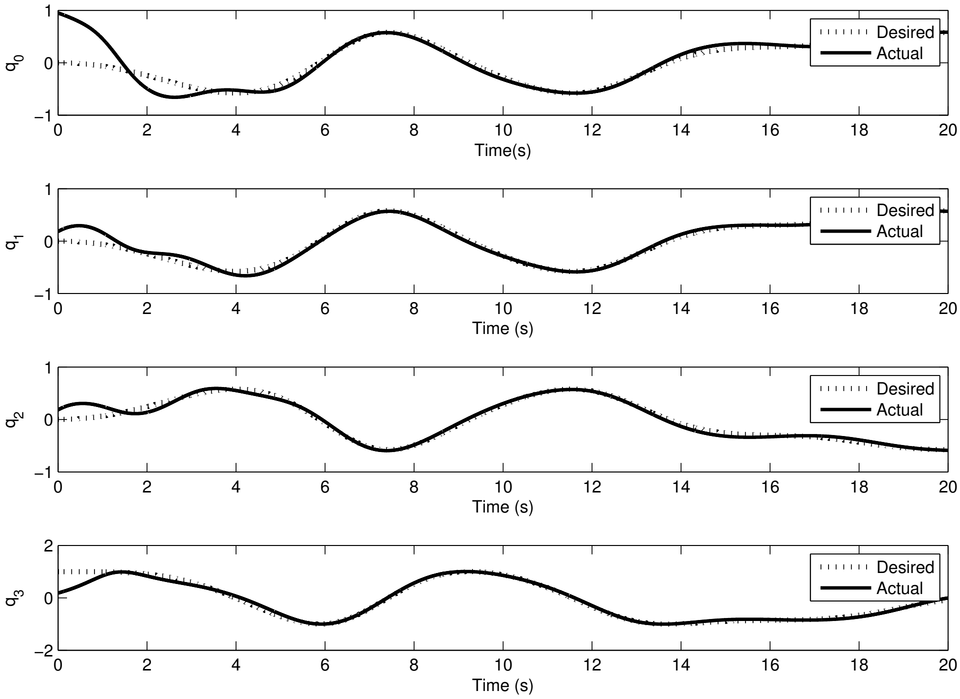


Fig. 2: Quaternion tracking using attitude control

A. Stabilization and Attitude tracking for satellite

First, the capability of the attitude controller is demonstrated by tracking a sample attitude history. The following angular velocity is taken from work by Costic et al. [12]

$$w_d(t) = \begin{pmatrix} 0.3\cos(t)(1-e^{-0.01t^2}) + [0.08\pi + 0.006\sin(t)]te^{-0.01t^2} \\ 0.3\cos(t)(1-e^{-0.01t^2}) + [0.08\pi + 0.006\sin(t)]te^{-0.01t^2} \\ 0.3\cos(t)(1-e^{-0.01t^2}) + [0.08\pi + 0.006\sin(t)]te^{-0.01t^2} \end{pmatrix}$$

The initial quaternion is chosen as $\mathbf{q}_{t=0} = [0.94846\ 0.1826\ 0.1826\ 0.1826]^T$ and the desired quaternion history is obtained from above angular velocity and taking $\mathbf{q}_{d_{t=0}}$ as $[1\ 0\ 0\ 0]^T$. The values of gain matrices is chosen such that $\zeta_i = 0.3$ and $\omega_n = diag(0.5, 2, 21.5)$. Fig.2 shows the desired and actual quaternion trajectory and Fig.3 shows the desired and actual angular velocity. It can be seen that both the desired attitude and angular velocity is achieved within 6 seconds.Hence, the proposed approach is able to track the attitude trajectory quite accurately.

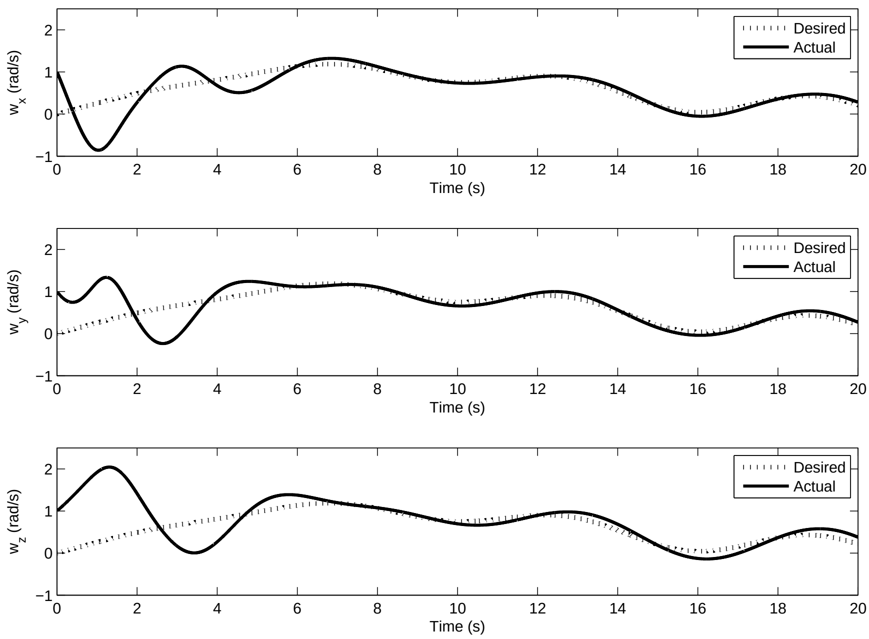


Fig. 3: Angular velocity tracking using attitude control

B. Fixed Pitch Quadrotor

We demonstrate trajectory tracking for a fixed-pitch quadrotor using spiral trajectory shown below:

$$X_d = \begin{pmatrix} \frac{\sin(\pi t/2)}{2} \\ \frac{\cos(\pi t/2)}{2} \\ t \end{pmatrix} \tag{51}$$

the gain matrices are chosen as $\zeta_i = 0.08$, $\omega_n = diag(2, 8, 8, 6)$ for the attitude loop and $\zeta_{p_i} = 0.1$ and $\omega_{p_n} = 1$. The initial state is $\vec{r}_{t=0} = (0\ 0\ 0)^T$, $\dot{\vec{r}} = (0\ 0\ 0)^T$, $\mathbf{q} = (1\ 0\ 0\ 0)^T$ and $\vec{\omega} = (0\ 0\ 0)^T$. The time step chosen for simulation is 0.001s and the position loop is run at 100 Hz while the attitude loop is run at 1000Hz. Fig.4 shows the desired and actual position of the quadrotor as a function of time. Fig.7a gives a 3 dimensional visualization of the same path. Fig.5 shows Euler angle variation just for better understanding as they are more intuitive and gives a better idea of orientation at any point of time. Fig.6 shows the quaternion variations, which are the actual mode or representing attitude in our work. Finally, Fig.7b shows the thrust variation with time. From above plots, it is clear that the proposed controller is able to track the desired trajectory effectively. The rate of

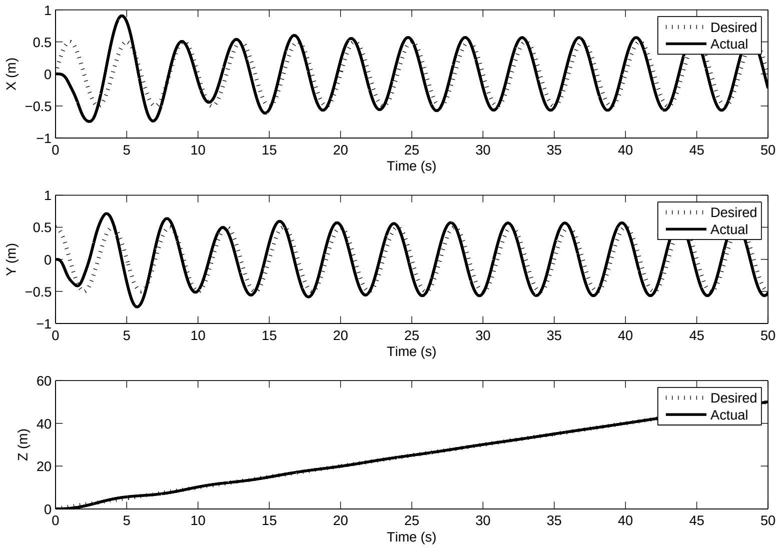


Fig. 4: Fixed Pitch Quadrotor position control simulation results

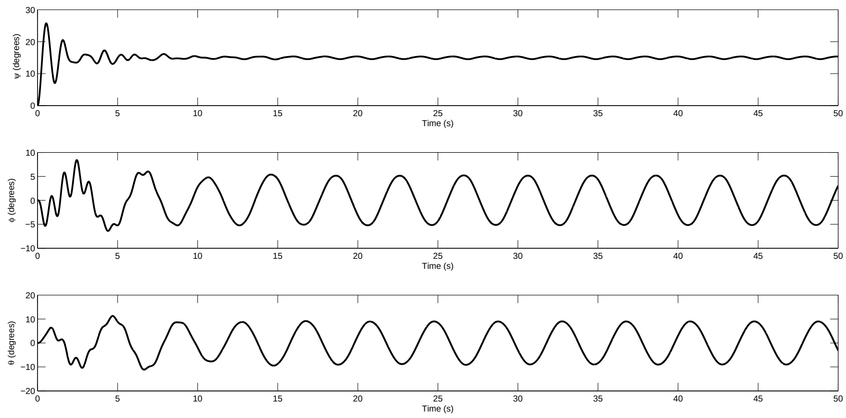


Fig. 5: Euler angles variation for fixed pitch quadrotor simulations

convergence is a matter of tuning and higher gains can be chosen for faster tracking.

C. Variable Pitch Quadrotor

The nonlinear control allocation methodology [9] for variable pitch actuation is adopted in this example to demonstrate the working of this new vehicle. Since the position and attitude controller remains the same as for a fixed-pitch quadrotor, we take this opportunity to present a special but simple maneuver not possible in fixed-pitch quadrotors. The output of the attitude controller is fed

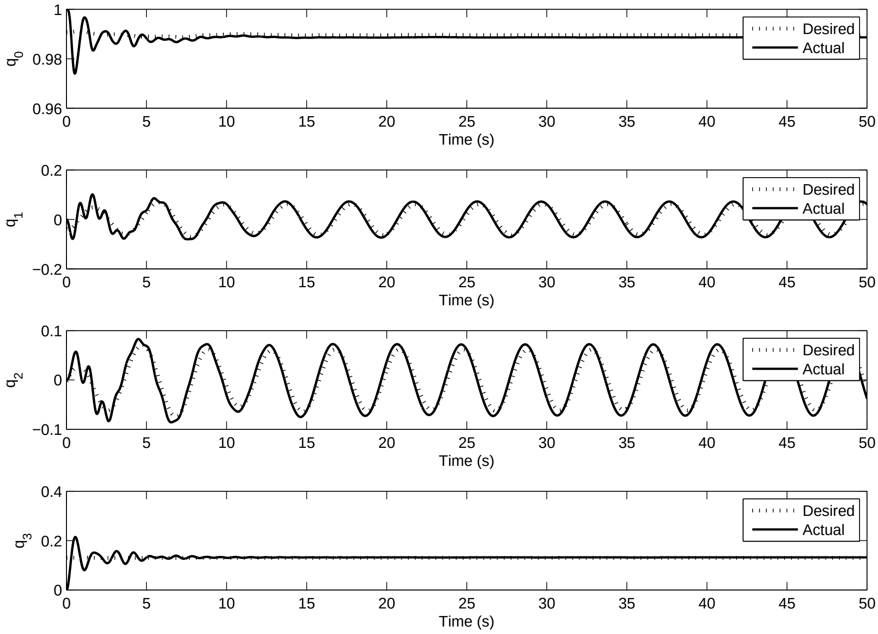
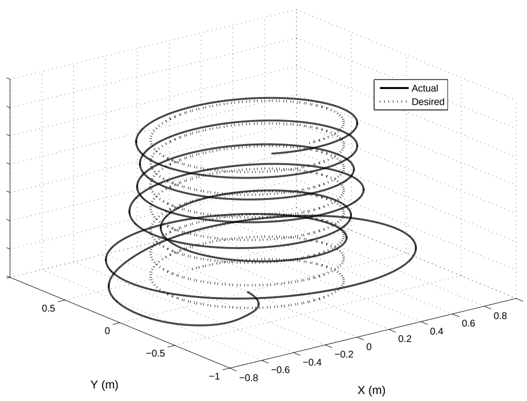
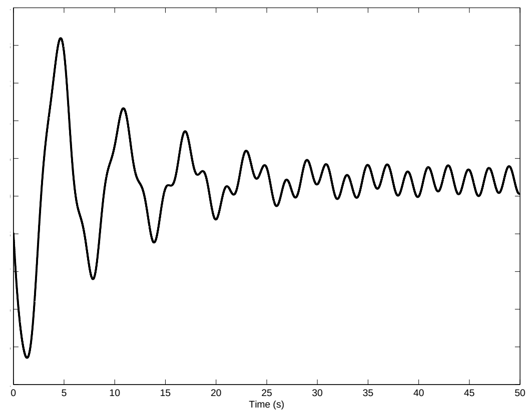


Fig. 6: Quaternion tracking performance for fixed pitch quadrotor simulation



(a) 3D Visualization of Quadrotor spiral path during trajectory tracking control



(b) Quadrotor thrust as function of time

Fig. 7: Fixed Pitch Quadrotor simulation

into the control allocation loop to achieve the desired thrust and moments for tracking.

The quadrotor is given a step input in the z direction. As already mentioned in Section III, the positive Z is chosen downwards. It is observed in the results that the high gains result in large

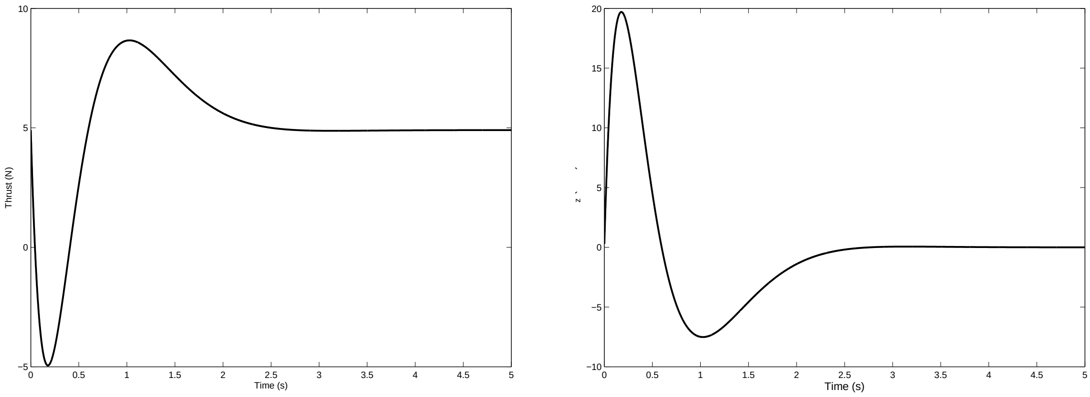
Parameter	Value	Parameter	Value	Parameter	Value	Parameter	Value
M	0.5 kg	V_t	50.89 m/s	ζ_{p_1}	0.8	ζ_1	0.3
I_{xx}	0.001 $kg\ m^4$	RPM	2700	ζ_{p_2}	0.8	ζ_2	0.3
I_{yy}	0.001 $kg\ m^4$	K	322.96 N	ζ_{p_3}	0.8	ζ_3	0.3
I_{zz}	0.001 $kg\ m^4$	$C_{l\alpha}$	5.23	ω_{p_1}	2.0	ω_{n_1}	1
L	0.2 m	N_b	2	ω_{p_2}	2.0	ω_{n_2}	4
ρ_{air}	1.225 kg/m^4	c	0.03 m	ω_{p_3}	2.0	ω_{n_3}	4
R	0.18	σ	0.1061			ω_{n_4}	3

Table 1: Parameters for Variable Pitch simulation

positive acceleration being demanded in the along the positive Z. This is achieved through creating negative thrust. The various parameters chosen for this simulation are given in Table.1.

Fig.9 shows the trajectory of the quadrotor. Fig.8a shows the thrust variation. Note that owing to the large initial error in position and due to the choice of gains, large acceleration is demanded initially in the downward direction which is achieved by generating negative thrust. After it reaches its designated point, the thrust comes back again to the hover value. Fig.8b shows the corresponding acceleration of the quadrotor. The maximum acceleration achieved is about 20 m/s^2 where 9.81 m/s^2 is the contribution from gravity and rest is from rotors. Thus we have achieved greater than g acceleration with quadrotor being in upright position. This demonstrates the increased control bandwidth associated with variable pitch quadrotors. Finally, Fig.10 shows the C_T variation for individual rotors and a flight regime not achievable by fixed-pitch quadrotors. Note that as mentioned previously, positive values correspond to thrust generation in the upward body direction while negative values correspond to the thrust along downward body direction.

A large acceleration may result in a high value of advanced ratio seen by the propellors which may lead to loss of thrust. But as the acceleration is only greater than g for a short duration (<0.5s), speeds in excess of 3 m/s are not achieved. This restricts the advanced ratio to lower values. Also thrust-loss associated with large deceleration maneuvers is not experimentally reported by Cutler et al. [8]. The effect is therefore neglected in the simulations.



(a) Quadrotor thrust variation with time

(b) Quadrotor achieved vertical acceleration

Fig. 8: Variable pitch quadrotor simulation

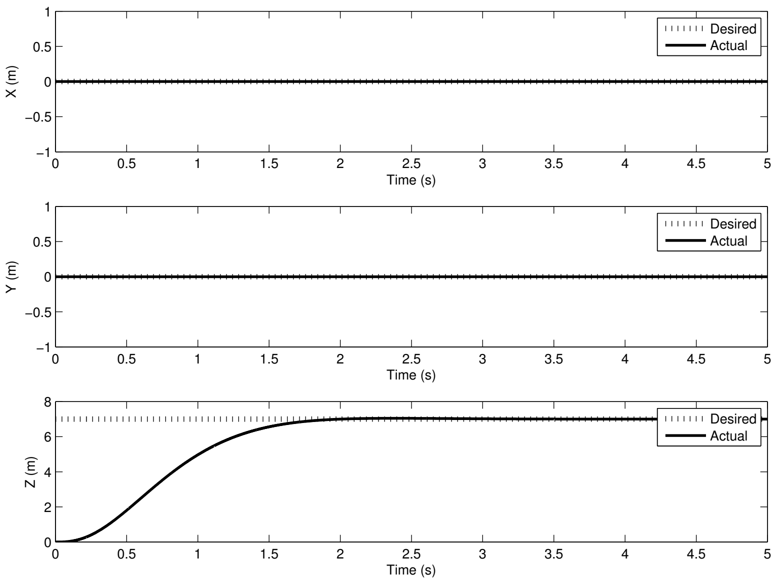


Fig. 9: Trajectory plot for variable pitch quadrotor simulations

VII. Experimental Validation

In this section, we discuss the experimental results using the proposed controller. The details of experimental setup followed by results of attitude and position tracking are presented.

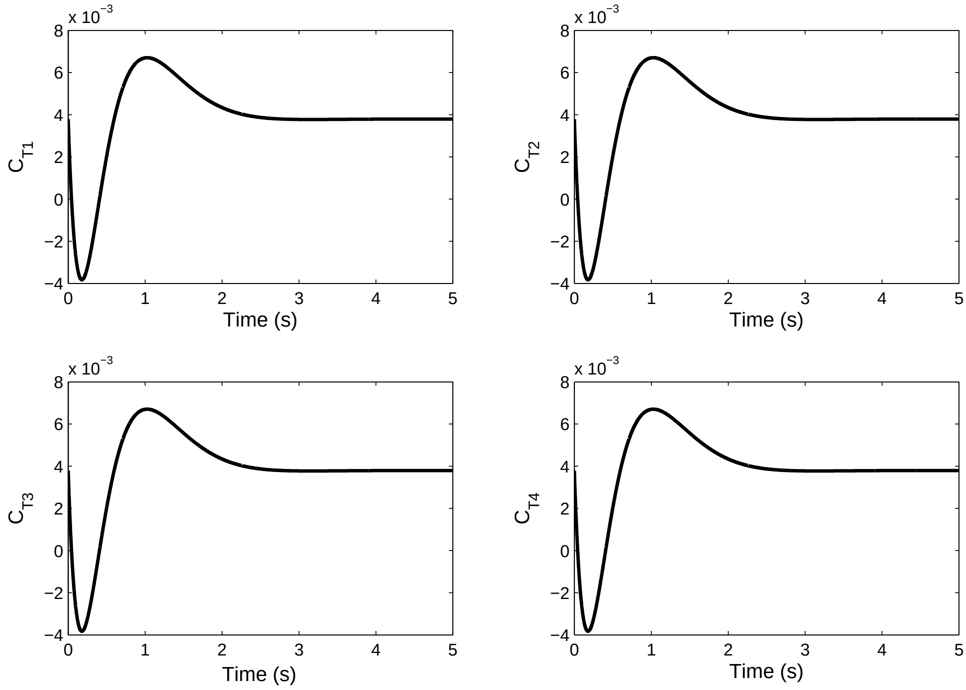


Fig. 10: Thrust coefficient variation for the four rotors of variable pitch quadrotor

A. Hardware

The proposed control algorithm is implemented on a commercially available fixed pitch H-King Color 250 Class quadrotor shown in Fig 12a. We use PX4 [21] opensource codebase running on Pixhawk autopilot hardware. Pixhawk uses ST Micro L3GD20H 16 bit gyroscope, ST Micro LSM303D 14 bit accelerometer / magnetometer and Invensense MPU 6000 3-axis accelerometer/gyroscope as sensors. These sensors are used in a non-linear filter to estimate the attitude of the quadrotor.The codebase uses a PID controller for attitude which is totally replaced with our quaternion based controller. The attitude loop is run as a priority thread with the actuation commands being given to the speed controllers via a rail. The position loop is run slower and processes the estimated position of the quadrotor to pass the computed attitude setpoint to the attitude controller. A 915MHz telemetry module is used to remotely communicate with the ground station. An FrSky X9D transmitter is used as the radio controller with FrSky X8R receiver placed onboard the vehicle and paired with the transmitter. The overall takeoff weight for the quadcopter is 630 gms. Moment of Inertia along I_{xx}= 2.2686E-3 kgm^2, I_{yy}=2.24557E-3 kgm^2, I_{zz}=3.19819E-3 kgm^2. The cross

*

Controller	Attitude								Position					
Gains	ζ_q				ω_q				ζ_{xyz}			ω_{xyz}		
Parameter	ζ_{q0}	ζ_{q1}	ζ_{q2}	ζ_{q3}	ω_{q0}	ω_{q1}	ω_{q2}	ω_{q3}	ζ_x	ζ_y	ζ_z	ω_x	ω_y	ω_z
Value	0.85	0.86	0.86	0.86	10.65	10.65	11	13	0.45	0.45	0.7	1	1	2

Table 2: Experimentally obtained tuning parameters for Attitude and Position Controller

moment of inertia are assumed to be zero as it is a nearly symmetrical structure. The experiments are performed outdoor on the IIT Kanpur airstrip with wind speed of around 3m/s. The values of ζ and ω_n are tuned for both position and attitude controllers so as to achieve an acceptable tracking performance.

B. Attitude controller validation

The quadrotor is flown in manual mode in which the pilot, generates the (T ,θ , ϕ , ψ) as control inputs using the transmitter. The control input was converted to a quaternion representation (q_d). Although, there are no specific guidelines to tune the frequency and damping ratio in the controller, we first try to obtain nominal values for the parameters involved (decided by the response and sensitivity of the controller as observed virtually) followed by the damping part(ζ). The system is able to track attitude commands with no position constraints. This is verified with a series of test flights with manual radio controller inputs. One such flight data is shown in Fig.11. The gain values are tabulated in Table 2. Small roll oscillations are given as input to showcase tracking. Note that in contrast to standard PD controllers based on Euler angles which have 6 parameters to tune, we have 8 parameters to tune corresponding to the 4 quaternion components. The tuning is also less intuitive but we avoid the singularity problem and once tuned the controller is more efficient computationally. For small angles, the vector part of the quaternion resembles the roll, pitch and yaw responses of the quadrotor however, the scalar part still has an effect coupled with all three rotational degrees of freedom. We observed that properly tuning the scalar part (q_0) helps in smoothing the response of the controller and reducing the tweaky response.

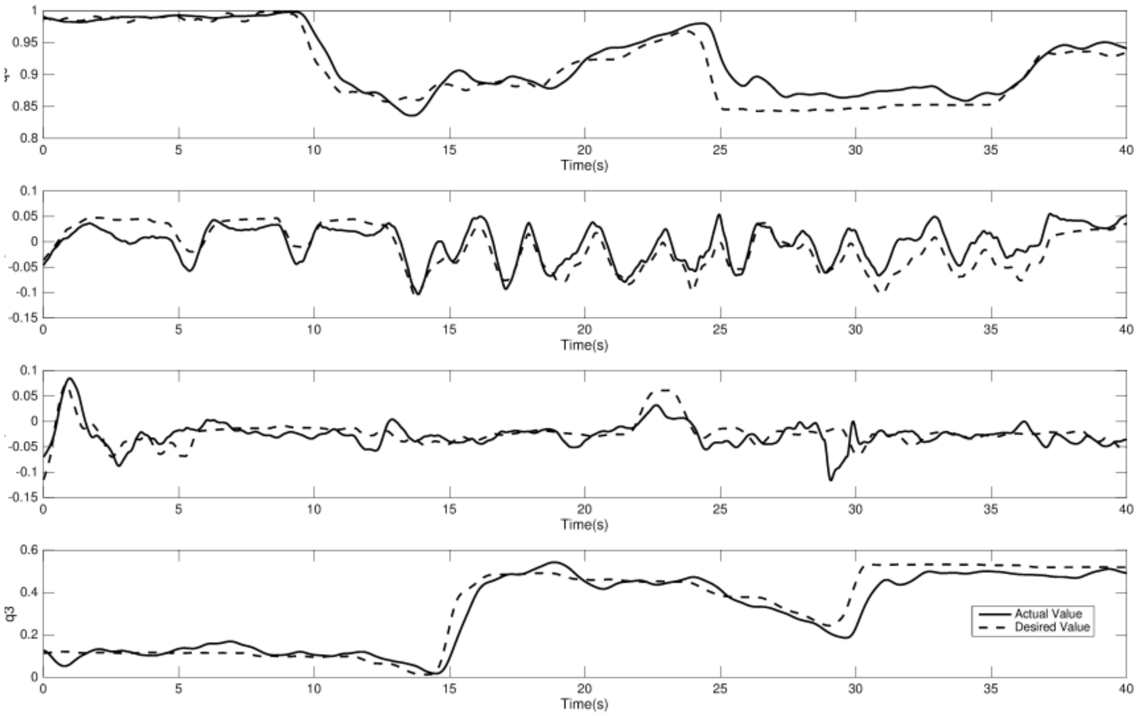
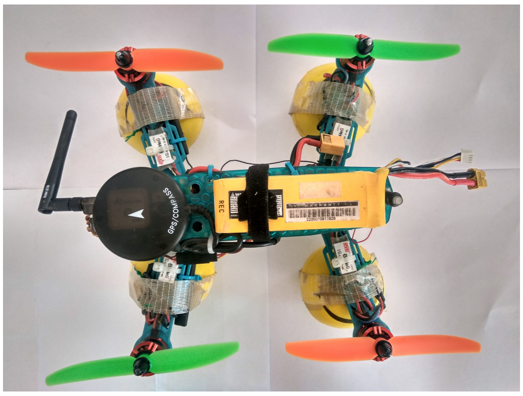


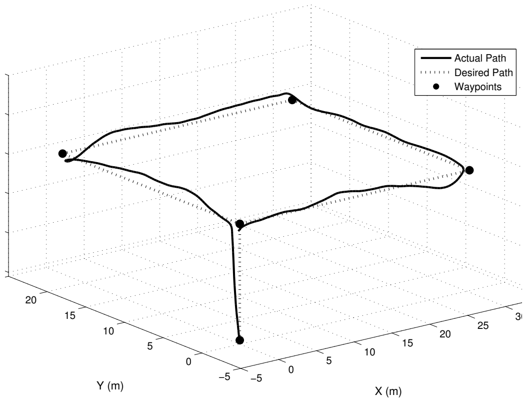
Fig. 11: Attitude controller experimental results for a fixed pitch quadrotor

C. Position Controller validation

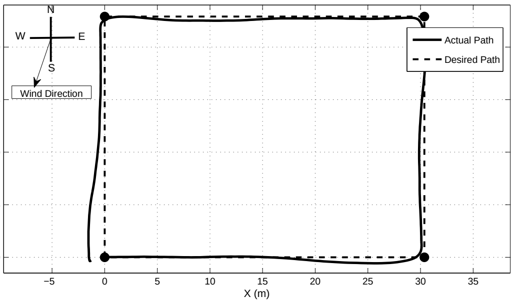
The position controller simulated earlier is validated by performing waypoint navigation. The desired trajectory consists of take-off to an altitude and then reaching 4 waypoints defining a square along straight lines. The waypoints are input through the ground control station (QGroundControl). The computed desired attitude is passed to the attitude controller tuned earlier. The yaw angle is specified so that the quadrotor always faces the next waypoint. The tuning is performed in a similar fashion to the attitude controller by performing a series of flight tests. The system is able to successfully track the desired trajectory and the results are shown in Fig.12b. Fig.12c shows the XY trajectory following and its convergence over time. Fig.12d shows the yaw angle tracking. The gain values are tabulated in Table.2. Note that due to the presence of 3 m/s wind towards South, the quadrotor has a tendency to deviate from its trajectory but our implemented controller corrects for it eventually.



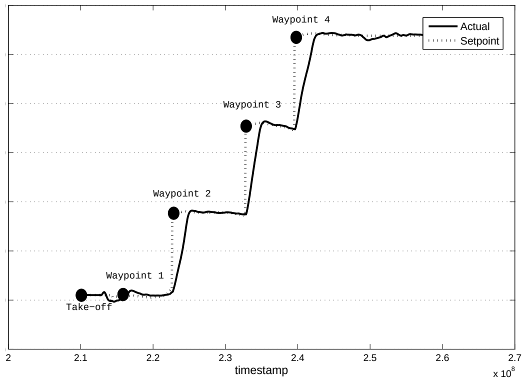
(a) Fixed pitch quadrotor used for experiments



(b) Waypoint Navigation in Mission mode



(c) XY trajectory tracking in Waypoint Navigation



(d) Yaw angle tracking in Waypoint Navigation

Fig. 12: Position controller experimental results for a fixed pitch quadrotor

VIII. Conclusions

In this paper, we have presented theory, implementation, and experimental results for a novel second order stable error dynamics controller using a fully quaternion based approach. The suitability of the generic controller for highly nonlinear systems like quadrotors and satellites is examined by designing controllers that efficiently track the desired trajectories. The controller enables the exploitation of highly maneuver systems like variable pitch quadrotors which are capable of generating negative thrust. It also avoids the problem of singularity often encountered in designing controllers for satellites with the added benefit of a second order system generating smoother control inputs. A framework for designing attitude controllers for rigid bodies and position controller for quadrotors is discussed and numerical simulation results are presented. The controllers are validated through a series of flight tests performed on quadrotor platform. The results of attitude and position tracking are shown and discussed.

Acknowledgments

The authors would like to thank Mahathi Bhargavapuria, Graduate student, Department of Electrical Engineering, Indian Institute of Technology for his invaluable support and inputs during the experiments carried out to validate the proposed controller.

References

[1] Choset, H. M., *Principles of robot motion: theory, algorithms, and implementation*, MIT press, 2005. Chapter 3.

[2] Lupashin, S., Schöllig, A., Sherback, M., and D'Andrea, R., "A simple learning strategy for high-speed quadrocopter multi-flips," in "Robotics and Automation (ICRA), 2010 IEEE International Conference on," IEEE, 2010, pp. 1642–1648. DOI:10.1109/ROBOT.2010.5509452.

[3] Mellinger, D., Michael, N., and Kumar, V., "Trajectory generation and control for precise aggressive maneuvers with quadrotors," *The International Journal of Robotics Research*, Vol. 31, No. 5, 2012, pp. 664–674. DOI:10.1177/0278364911434236.

[4] Fresk, E. and Nikolakopoulos, G., "Full quaternion based attitude control for a quadrotor," in "Control Conference (ECC), 2013 European," IEEE, 2013, pp. 3864–3869. ISBN: 978-3-033-03962-9, INSPEC: 13936305.

[5] Tayebi, A. and McGilvray, S., "Attitude stabilization of a four-rotor aerial robot," in "Decision and Control, 2004. CDC. 43rd IEEE Conference on," Ieee, Vol. 2, 2004, pp. 1216–1221. DOI:10.1109/CDC.2004.1430207.

[6] Xian, B., Diao, C., Zhao, B., and Zhang, Y., "Nonlinear robust output feedback tracking control of a quadrotor UAV using quaternion representation," *Nonlinear Dynamics*, Vol. 79, No. 4, 2015, pp. 2735–2752. DOI:10.1007/s11071-014-1843-x.

[7] Bangura, M., Lim, H., Kim, H. J., and Mahony, R., "An open-source implementation of a unit quaternion based attitude and trajectory tracking for quadrotors," in "Australasian Conference on Robotics and Automation," , 2014, pp. 1–10.

[8] Cutler, M. and How, J. P., "Analysis and control of a variable-pitch quadrotor for agile flight," *Journal of Dynamic Systems, Measurement, and Control*, Vol. 137, No. 10, 2015, p. 101002. DOI: 10.1115/1.4030676.

[9] Gupta, N., Kothari, M., et al., "Flight dynamics and nonlinear control design for variable-pitch quadrotors," in "American Control Conference (ACC), 2016," IEEE, 2016, pp. 3150–3155. DOI:10.1109/ACC.2016.7525402.

[10] Leishman, G. J., *Principles of helicopter aerodynamics with CD extra*, Cambridge university press, 2006. Chapter 3.

[11] Borenstein, J., "The hoverbot, an electrically powered flying robot," *Ann Arbor*, Vol. 1001, 1992, pp. 48109–2110. URL: http://ftp.eecs.umich.edu/people/johannb/paper99.pdf.

[12] Costic, B., Dawson, D., De Queiroz, M., and Kapila, V., "Quaternion-based adaptive attitude tracking controller without velocity measurements," *Journal of Guidance, Control, and Dynamics*, Vol. 24, No. 6, 2001, pp. 1214–1222. DOI:10.1109/CDC.2000.914164.

[13] Diebel, J., "Representing attitude: Euler angles, unit quaternions, and rotation vectors," *Matrix*, Vol. 58, No. 15-16, 2006, pp. 1–35.

[14] Graf, B., "Quaternions and dynamics," *arXiv preprint arXiv:0811.2889*.

[15] Sreenath, K., Lee, T., and Kumar, V., "Geometric control and differential flatness of a quadrotor UAV with a cable-suspended load," in "Decision and Control (CDC), 2013 IEEE 52nd Annual Conference on," IEEE, 2013, pp. 2269–2274. DOI:10.1109/CDC.2013.6760219.

[16] Slotine, J.-J. E., Li, W., et al., *Applied nonlinear control*, Vol. 199, Prentice hall Englewood Cliffs, NJ, 1991. Chapter 6.

[17] Khalil, H. K., *Noninear Systems*, Prentice-Hall, New Jersey, 1996. Chapter 8.

[18] Enns, D., Bugajski, D., Hendrick, R., and Stein, G., "Dynamic inversion: an evolving methodology for flight control design," *International Journal of control*, Vol. 59, No. 1, 1994, pp. 71–91.

[19] Prabhakaran, B., Kothari, M., et al., "Nonlinear control design for quadrotors," in "Computational Intelligence: Theories, Applications and Future Directions (WCI), 2015 IEEE Workshop on," IEEE, 2015, pp. 1–6. DOI:10.1109/WCI.2015.7495504.

[20] Markley, F. L., "Fast quaternion attitude estimation from two vector measurements," *Journal of Guidance, Control, and Dynamics*, Vol. 25, No. 2, 2002, pp. 411–414. DOI: 10.2514/2.4897.

[21] Meier, L., Honegger, D., and Pollefeys, M., "PX4: A node-based multithreaded open source robotics framework for deeply embedded platforms," in "Robotics and Automation (ICRA), 2015 IEEE International Conference on," IEEE, 2015, pp. 6235–6240. DOI:10.1109/ICRA.2015.7140074.