2020 American Control Conference
Denver, CO, USA, July 1-3, 2020

Quaternion Based Nonlinear Trajectory Control of Quadrotors with Guaranteed Stability

Joo-Won Kang, Nader Sadegh, and Chase Urschel

***Abstract*—This paper investigates a dual loop quaternion-based nonlinear control scheme for a quadrotor: the inner feedback loop controls the attitude of the quadrotor while the outer control loop aims to control the linear velocity, and/or position, of the quadrotor. The paper shows that global exponential stability of a quadrotor can be achieved with the proposed control scheme. The presented controller includes feedback and feedforward compensation of the nonlinear dynamics of the quadrotor and gyroscopic torques to guarantee global stability of the attitude controller. The outer velocity control loop uses a PI feedback structure, where the proportional action is primarily used to control linear velocity, and the integral action can either be used for linear position and/or eliminating velocity steady-state error. The inner attitude control includes a PD feedback loop, where the proportional action is in terms of the vector part of the quaternion and the derivative action is in terms of the aircraft angular velocity. The proposed controller has been both simulated and tested experimentally with a commercial quadrotor.**

***Index Terms*—UAV, quadrotor, quaternion control, attitude stabilization, velocity control, position control, flight control, nonlinear control, lyapunov method.**

I. Introduction

Multirotors have seen a growing interest in the research community since the late 2000s. Multirotors make excellent unmanned aerial vehicles (UAVs) due to their capability of fixed position hovering and vertical take-off and landing (VTOL). Fixed-wing UAVs have longer flight times but need runways for take-off and landing and to move continuously to generate lift. Due to their unique characteristics, multirotor UAVs have become popular for military applications, photography and filming, and agricultural spraying and surveying.

Multirotor attitude control is commonly based on one of three different attitude representations: Euler angles, direction cosine matrices (DCM), and quaternions. Euler angles have the advantage of being the most intuitive, as each of the three Euler angles represents elementary rotations around the three principles axes of the aircraft: roll, pitch, and yaw. An alternative to the traditional Euler angles is the DCM representation. DCM representations are usually the combination of elementary rotation matrices that each corresponds to roll, pitch, and yaw rotations. Though commonly used, Euler angles have inherent geometric singularities, commonly known as "gimbal lock", while DCM representations are computationally expensive as all nine elements of the matrix are utilized [1]. Quaternion attitude control presents a singularity-free and less computationally expensive alternative. A more detailed comparison between the three can be found in [2]. The use of quaternion attitude representations can be found in [3], in which the paper proposes a quaternion-based PD^2 attitude controller of a quadrotor, as well as in [1], [4], [5], [6], and [7]. The attitude control problem, as well as the position control problem, has been addressed by several papers using various control techniques. Papers [8] and [9] propose an inner-outer loop PID structure to control the position of a quadrotor. Moreover, sliding mode control ([10], [11]), back-stepping control ([12]), adaptive control ([13], [7]) and feedback linearization ([10], [14]) are some of the nonlinear control techniques that have been explored in existing controllers. More recently, [15] proposed an interesting controller based on a combination of feedback linearization and embedded model control.

This paper introduces a globally stabilizing nonlinear controller that simultaneously controls the attitude and linear position/velocity of a quadrotor. The attitude controller includes PD feedback and nonlinear feed-forward compensation to track any arbitrary orientation. The controller is coupled with a PID-based velocity/position controller, which feeds the desired attitude quaternion to the inner attitude loop. A mixed velocity/attitude nonlinear feedback component is also included to ensure global stability. The proposed controller also utilizes quaternions to represent orientation, which eliminates the singularities possible in [8] and [9]; and, hence, do not need any limitation in attitude. Without any singularity, the proposed controller can recover from or reach any attitude at any velocity or position. In real-life situations, the controller has the potential to re-stabilize a free-falling quadrotor.

The following section covers the quaternion and modeling aspects of the proposed quadrotor control design. The specifics of the control design, with the inclusion of a complete Lyapunov stability analysis to prove that global exponential stability can be achieved, is subsequently given. And, finally, a controller is designed, simulated, and tested experimentally with a commercial quadrotor.

II. Background

A. Quaternion Math

A quaternion consists of two components - the scalar/real (q_0) and vector/imaginary (q_v) parts, each given by:

1)Joo-Won Kang is with the George W. Woodruff School of Mechanical Engineering, Georgia Institute of Technology, Atlanta, GA 30332, USA jkang86@gatech.edu 2)Nader Sadegh is with the Faculty of The George W. Woodruff School of Mechanical Engineering, Georgia Institute of Technology, Atlanta, GA 30332, USA sadegh@gatech.edu 3)Chase Urschel is with the George W. Woodruff School of Mechanical Engineering, Georgia Institute of Technology, Atlanta, GA 30332, USA curschel@gatech.edu

978-1-5386-8266-1/$31.00 ©2020 AACC

$$\begin{bmatrix} q_0 \\ q_v \end{bmatrix} = \begin{bmatrix} \cos\left(\frac{\theta}{2}\right) \\ \hat{r}\sin\left(\frac{\theta}{2}\right) \end{bmatrix} \tag{1}$$

where θ denotes the angle of rotation about a unit vector $\hat{r} \in \mathbb{R}^3$ [3]. Although quaternion representation avoids singularity, it has sign ambiguity; each rotation may be represented in two different sets of quaternions ([q_0,q_v] and [-q_0,-q_v]). This setback is commonly avoided by using quaternions that correspond to a positive q_0 ($0 \leq q_0 \leq 1$).

All quaternions representing attitude or rotation are unit quaternions; meaning that they are all subject to the condition:

$$q^T q = q_0^2 + q_v^T q_v = 1 \tag{2}$$

The inverse of a quaternion is

$$q^{-1} = \frac{q^*}{q^T q} \tag{3}$$

where $q^* = [q_0 \ \text{-}q_v]^T$ denotes the conjugate of the quaternion. The time derivative of any quaternion is given by,

$$\dot{q} = \frac{1}{2} q \bar{\omega} \tag{4}$$

where ω is the rate of rotation and $\bar{\mathbf{v}} = [0 \ \mathbf{v}^T]^T$ represents a quaternion equivalent of a vector $\mathbf{v} \in \mathbb{R}^3$ [2], [6]. Throughout the paper, we denote the rotation matrix corresponding to quaternion q by R_q:

$$\overline{R_q \mathbf{v}} = q \bar{\mathbf{v}} q^{-1}, \forall \mathbf{v} \in \mathbb{R}^3 \tag{5}$$

B. Quadrotor Model

Let $\mathcal{A} = \{e_{a,x} \ e_{a,y} \ e_{a,z}\}$ represent the aircraft body or local frame of the quadrotor, and $\mathcal{I} = \{e_x \ e_y \ e_z\}$ be the inertial or global frame, as shown in Fig. 1. Throughout the paper, we shall also denote the canonical basis vectors, $[1 \ 0 \ 0]^T$, $[0 \ 1 \ 0]^T$, and $[0 \ 0 \ 1]^T$, by e_x, e_y, and e_z, respectively, when no confusion is likely to arise. The dynamic model of the quadrotor, according to the two frames, is given as below [3]:

$$\dot{x} = v \tag{6}$$

$$m\dot{v} = f_t R_q e_z - g e_z \tag{7}$$

$$\dot{q} = \frac{1}{2} q \bar{\omega}_b \tag{8}$$

$$I_f \dot{\omega}_b = \tau_a + \omega_b \times I_f \omega_b - G_a \tag{9}$$

where $x \in \mathbb{R}^3$ and $v = \dot{x}$ represent the linear position and velocity vectors, respectively, of the origin of the aircraft frame $\mathcal{A}$ with respect to the inertial frame $\mathcal{I}$, $q = [q_0 \ q_v]^T$ denotes the quaternion representation of the aircraft attitude, $\omega_b = [\omega_{b,1} \ \omega_{b,2} \ \omega_{b,3}]^T$ represents the angular velocity of the quadrotor relative to the aircraft frame $\mathcal{A}$. m and I_f each denote the mass and inertial matrix of the quadrotor; g denotes the gravitational constant (i.e., $g \approx 9.806 \ m/s^2$). Gyroscopic torque, $G_a = [G_{a,1} \ G_{a,2} \ G_{a,3}]^T$, is the resulting torque produced due to the rotation of the aircraft and the four rotors, which is given by:

$$G_a = \sum_{i=1}^{4} I_r(\omega_b \times e_z)(-1)^{i+1}\Omega_i \tag{10}$$

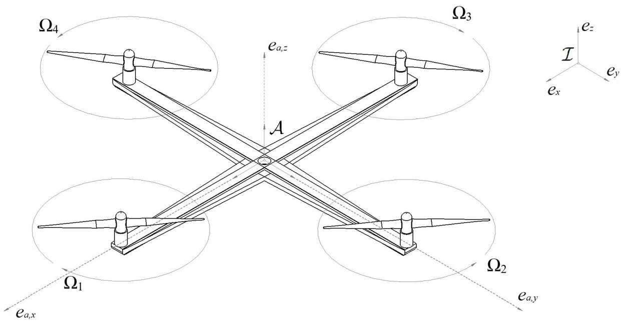


Figure 1: Depiction of a General Quadrotor

where I_r denotes the moment of inertia of the four rotors. f_t and $\tau_a = [\tau_{a,1} \ \tau_{a,2} \ \tau_{a,3}]^T$ are the total thrust and torque generated. Let $\mu = [\tau_a^T \ f_t]^T$ and $\Omega = [\Omega_1 \ \Omega_2 \ \Omega_3 \ \Omega_4]^T$, where $\Omega_i, i \in \{1,2,3,4\}$ is the angular speed of the four rotors. Then, τ_a and f_t are given by $\mu = M(\Omega^T\Omega)$, where the matrix M is defined as the following:

$$M = \begin{bmatrix} 0 & bl & 0 & -bl \\ bl & 0 & -bl & 0 \\ \kappa & -\kappa & \kappa & -\kappa \\ b & b & b & b \end{bmatrix} \tag{11}$$

b and κ each quantifies two dimensionless parameters, the drag and lift of the four rotor blades, respectively. l denotes the arm length of the torques created by each of the rotors [3].

III. Quadrotor Control Design

The control design for velocity or position of the quadrotor consists of two coupled feedback loops: an inner attitude feedback loop and an outer velocity/position feedback loop. The attitude controller features a nonlinear feedback plus feedforward control law that is capable of tracking a time-varying attitude trajectory. The outer feedback loop, on the other hand, controls the linear velocity and/or position of the quadrotor by adjusting the thrust force and supplying the desired attitude trajectories to the inner loop. Finally, the input torques in each direction of rotation (roll, pitch, and yaw) are determined by the attitude controller.

A. Attitude Control Law

To state the attitude control law, let q denote the actual attitude of the aircraft, and q_d denote the desired attitude of the quadrotor; both q and q_d are in quaternion representation and relative to the inertial frame $\mathcal{I}$. The attitude *error* is defined as the current attitude relative to the desired attitude:

$$\tilde{q} = q_d^{-1} q \tag{12}$$

Differentiating $\tilde{q}$ in (12) with respect to time yields the angular velocity *error* relative to the aircraft frame $\mathcal{A}$:

$$\tilde{\omega} = 2\tilde{q}^{-1}\dot{\tilde{q}} = \omega_b - \omega_d \tag{13}$$

where ω_d represents the desired angular velocity relative to the aircraft frame $\mathcal{A}$. The reference angular velocity used in the control law is defined to be:

$$\omega_r = \omega_d - \Lambda_q \tilde{q} \tag{14}$$

where $\Lambda_q = [0 \ \lambda_q I_{3\times 3}] \in \mathbb{R}^{3\times 4}$, wherein λ_q is a positive control gain. Finally, the reference angular velocity *error* is the difference between ω_b and ω_r:

$$\tilde{\omega}_r = \omega_b - \omega_r \tag{15}$$

Note that by inserting (14) into (15), an alternative expression for $\tilde{\omega}_r$ may be obtained:

$$\tilde{\omega}_r = \tilde{\omega} + \Lambda_q \tilde{q} \tag{16}$$

The attitude control law sets the quadrotor input torque vector, τ_a, according to:

$$\tau_a = I_f \dot{\omega}_r + \omega_r \times I_f \omega_b + G_a - K_\omega \tilde{\omega}_r - K_q \tilde{q} \tag{17}$$

where $\dot{\omega}_r$ is the time derivative of ω_r. K_ω is a 3×3. symmetric, positive definite matrix and $K_q = [0_{3\times 1} \ k_q I_{3\times 3}] \in \mathbb{R}^{3\times 4}$, $k_q > 0$ is a 3×4 matrix which each act as gains for the reference angular velocity error and attitude error, respectively. Define the *error* states of the attitude control loop as $e_1 = (\tilde{q}, \tilde{\omega}_r) \in \mathbb{R}^7$. Applying the control law (17) to the attitude subsystem (8)-(9) results in the following closed-loop error dynamics:

$$\dot{\tilde{q}} = \frac{1}{2} \tilde{q}\, \tilde{\omega}_r \tag{18}$$

$$I_f \dot{\tilde{\omega}}_r + \tilde{\omega}_r \times I_f \omega_b + K_\omega \tilde{\omega}_r + K_q \tilde{q} = 0 \tag{19}$$

The following Theorem, which is a generalization of the attitude controller in [3] to time-varying trajectories, proves the global exponential stability of the attitude control closed-loop error dynamics (18)-(19):

Theorem 1. *The attitude closed-loop error system described by* (18)-(19) *resulting from the control law* (17) *applied to the attitude subsystem* (8)-(9) *is globally exponentially stable for a twice continuously differentiable quaternion* q_d *with bounded derivatives.*

Proof. To prove the Theorem, consider the Lyapunov function candidate:

$$V(\tilde{q}, \tilde{\omega}_r) = \frac{1}{2} \tilde{\omega}_r^T I_f \tilde{\omega}_r + k_q (\tilde{q} - 1)^T (\tilde{q} - 1) \tag{20}$$

In view of Lemma 1 (Appendix A), the time derivative of V is given by:

$$\dot{V}(\tilde{q}, \tilde{\omega}_r) = \tilde{\omega}_r^T I_f \dot{\tilde{\omega}}_r + \tilde{\omega}^T K_q \tilde{q} \tag{21}$$

In view of (16) and (19), $\dot{V}$ can be simplified as follows:

$$\begin{aligned} \dot{V}(\tilde{q}, \tilde{\omega}_r) &= -\tilde{\omega}_r^T (\tilde{\omega}_r \times I_f \omega_b + K_\omega \tilde{\omega}_r + K_q \tilde{q}) + \tilde{\omega}^T K_q \tilde{q} \\ &= -\tilde{\omega}_r K_\omega \tilde{\omega}_r - k_q \tilde{q}^T \Lambda_q^T \tilde{q} \end{aligned} \tag{22}$$

Note that $\dot{V}(\tilde{q}, \tilde{\omega}_r)$ can also be expressed as:

$$\dot{V}(\tilde{q}, \tilde{\omega}_r) = -\tilde{\omega}_r K_\omega \tilde{\omega}_r - \lambda_q k_q \frac{1 + q_0}{2} (\tilde{q} - 1)^T (\tilde{q} - 1) \tag{23}$$

As a result, $\dot{V}(\tilde{q}, \tilde{\omega}_r) \leq -\alpha V$, where $\alpha = \min(\frac{2\lambda_{\min}(K_\omega)}{\lambda_{\max}(I_f)}, \lambda_q)$ and $\lambda_{\min/\max}(\mathbf{m})$ is the minimum or maximum eigenvalue of matrix $\mathbf{m}$, thus proving the global exponential stability of the proposed attitude control law (Theorem 4.10 in [16]). □

B. Velocity/Position Control Law

Let $u_d = [u_{d,1} \ u_{d,2} \ u_{d,3}]^T$ be the desired thrust vector, scaled by the aircraft mass, of the quadrotor relative to the inertial frame $\mathcal{I}$. Then, in view of (6) and (7), the position control law is given by:

$$u_d = \dot{v}_d - K_v \tilde{v} - K_x \tilde{x} - K_i \int_0^t \tilde{x} + g e_z \tag{24}$$

where $\dot{v}_d$ is the desired acceleration of the aircraft. $\tilde{x} = x - x_d$ and $\tilde{v} = v - v_d$, where x_d and v_d denote the desired linear position and velocity of the aircraft, respectively. The gain matrices K_v, K_x, and K_i are 3×3, symmetric, positive definite, and chosen to make the closed-loop characteristic equation, $|s^3 I + s^2 K_v + s K_x + K_i|$, Hurwitz. The full thrust vector u_d cannot be directly applied; hence, the thrust force f_t is set to the magnitude of u_d scaled by m (i.e., $f_t = m||u_d||$). Defining the *error* state of the velocity/position control loop as $e_2 = (w, \tilde{x}, \tilde{v}) \in \mathbb{R}^9$, the closed-loop error dynamics of the system can be described by:

$$\dot{w} = \tilde{x} \tag{25}$$

$$\dot{\tilde{x}} = \tilde{v} \tag{26}$$

$$\dot{\tilde{v}} = -K_v \tilde{v} - K_x \tilde{x} - K_i w + \tilde{u} \tag{27}$$

where $\tilde{u} = u - u_d$ represents the *error* between the desired and actual thrust vectors. Note that the closed-loop error dynamics can also be expressed as:

$$\dot{e}_2 = A e_2 + B \tilde{u} \tag{28}$$

where $A \in \mathbb{R}^{9\times 9}$ and $B \in \mathbb{R}^{9\times 3}$ denote the state and input matrices, respectively, of the error dynamics state-space representation in (25)-(27). The attitude controller and the velocity controller are coupled through $\tilde{u}$, where the u_d obtained through velocity/position control is passed to the attitude controller, in which the desired attitude quaternions are obtained. Note that $\tilde{u}$ may also be given by:

$$\tilde{u} = \frac{f_t}{m} R_{q_d} (\tilde{R}_q e_z - e_z) \tag{29}$$

where $\tilde{R}_q = R_{q_d}^T R_q$. In addition, Λ_q in (14) of the attitude controller must be modified according to (30) to guarantee global exponential stability of the overall control system.

$$\Lambda_q = (\lambda_q + \lambda_u \|u_d\|^2)[0_{3\times 1} \ I_{3\times 3}] \tag{30}$$

where λ_u is an additional positive control gain. Note that Λ_q in the new control law essentially serves as an additional gain of the attitude error $\tilde{q}$, which adjusts according to $\|u_d\|$.

Next, we describe how the desired and error quaternions, q_d and $\tilde{q}$, are generated from the desired thrust vector u_d.

Assuming $\|u_d\| \neq 0$, the normalization $\hat{u}_d$ of u_d relative to frame $\mathcal{A}$ is given by the following:

$$\hat{u}_d = \frac{R_q^T u_d}{\|u_d\|} \tag{31}$$

A vanishing thrust vector u_d is a rare occurrence and almost never happens in practice. Theorem 2 shows that for a suitably chosen desired trajectory $\|u_d(t)\|$ stays bounded away from zero for all t. The error quaternion now can be defined based on the normalized thrust vector $\hat{u}_d = [\hat{u}_{d,1} \;\; \hat{u}_{d,2} \;\; \hat{u}_{d,3}]^T$:

$$\tilde{q} = \frac{\begin{bmatrix}1+\hat{u}_{d,3} & -\hat{u}_{d,2} & \hat{u}_{d,1} & 0\end{bmatrix}^T}{\sqrt{2(1+\hat{u}_{d,3})}} \tag{32}$$

The error quaternion, $\tilde{q}$, represents the rotation about the common perpendicular of u and u_d that rotates u to u_d assuming $\hat{u}_{d,3} \neq -1$. The axis of rotation relative to frame $\mathcal{A}$ is simply $e_z \times \hat{u}_d$ with the corresponding rotation angle of $\theta = \cos^{-1}(e_z^T \hat{u}_d)$. The angular velocity error $\tilde{\omega} = 2\tilde{q}^{-1}\dot{\tilde{q}}$ and $\omega_d = \omega_b - \tilde{\omega}$ relative to $\mathcal{A}$ can be easily computed using

$$\dot{\hat{u}}_d = \frac{1}{\|u_d\|}(1 - \hat{u}_d\hat{u}_d^T)(R_q^T \dot{u}_d - \omega_b \times u_d) \tag{33}$$

where $\dot{u}_d = -KAe_2 + KB\tilde{u} + \ddot{v}_d$. To keep both $\|u_d\|$ and $|1+\tilde{u}_{d,3}|$ bounded away from zero, the desired trajectory $x_d(t)$ is chosen to be a smooth function satisfying the following.

DT: $x_d(0) = x(0)$, $v_d(0) = v(0)$, $\dot{v}_d(0) = R_{q(0)}e_z - ge_z$, $\ddot{v}_d(0) = 0$, and $\inf_{t\in\mathbb{R}^+} \|\dot{v}_d(t) + ge_z\| \neq 0$.

Note that assumption DT is not too restrictive as the original desired trajectory may be modified to satisfy it. For instance, adding two exponentially decaying terms to x_d

$$x_d(t) \leftarrow x_d(t) + \tilde{x}(0)(1+at)e^{-at} + t\tilde{v}(0)e^{-at}, \quad a > 0$$

guarantees that $x_d(0) = x(0)$ and $v_d(0) = v(0)$. The following Theorem, which is the main result of this paper, proves the global exponential convergence of the overall control system with respect to the initial position, velocity, and orientation:

Theorem 2. *Consider the closed-loop system described by (18)-(19) and (25)-(27) resulting from the control laws (17) and (24) with $f_t = m\|u_d\|$ and $\tilde{q}$ generated from q and u_d according to (32). The overall error system is exponentially stable for a desired trajectory $x_d(t)$ satisfying DT and sufficiently large λ_u and k_q. In particular, $\tilde{x}$, $\tilde{v}$, and $\tilde{\omega}$ converge to zero exponentially for an arbitrary initial position $x(0)$, velocity $v(0)$, and orientation $q(0)$ but sufficiently small $\|\omega_b(0)\|$. Furthermore, $\|u_d\| \neq 0$ and $\tilde{q}_0 \neq 0$, $\forall t \geq 0$.*

Proof. Consider the Lyapunov function candidate:

$$V(\tilde{q}, \tilde{\omega}_r, e_2) = \frac{1}{2}\tilde{\omega}_r^T I_f \tilde{\omega}_r + k_q(\tilde{q}-1)^T(\tilde{q}-1) + e_2^T P e_2 \tag{34}$$

where ω_r is given by (14), as before, and P is a 9×9 matrix that satisfies the Lyapunov equation:

$$A^T P + PA + Q = 0 \tag{35}$$

where A is given by (28), and Q is a symmetric, positive definite matrix (Theorem 4.6 and Equation 4.12 in [16]). In view of (22) and (35), the time derivative of the presented Lyapunov function is then given by:

$$\begin{aligned}\dot{V}(\tilde{q}, \tilde{\omega}_r, e_2) = &- \tilde{\omega}_r^T K_\omega \tilde{\omega}_r - \tilde{q}^T \Lambda_q^T K_q \tilde{q} - e_2^T Q e_2 \\ &+ 2e_2^T C\tilde{u}\end{aligned} \tag{36}$$

where $C = PB$ and B is given by (28). By Lemma 2 (Appendix A), $\dot{V}(\tilde{q}, \tilde{\omega}_r, e_2)$ can be bounded as follows:

$$\begin{aligned}\dot{V} \leq &- \alpha\|\tilde{\omega}_r\|^2 - (\lambda_q + \lambda_u\|u_d\|^2)k_q\|\tilde{q}_v\|^2 - \eta\|e_2\|^2 \\ &+ 4c\|e_2\|\,\|u_d\|\,\|\tilde{q}_v\| \\ \leq &- \alpha\|\tilde{\omega}_r\|^2 - \lambda_q k_q\|\tilde{q}_v\|^2 \\ &- \begin{bmatrix}\|e_2\| & \|u_d\|\,\|\tilde{q}_v\|\end{bmatrix}\begin{bmatrix}\eta & -2c \\ -2c & \lambda_u k_q\end{bmatrix}\begin{bmatrix}\|e_2\| \\ \|u_d\|\,\|\tilde{q}_v\|\end{bmatrix}\end{aligned} \tag{37}$$

where $\eta = \lambda_{\min}(Q)$ and $c = \|C\|$. Similarly to the proof of Theorem 1, it follows that if $\eta\lambda_u k_q > 4c^2$, the 2×2 matrix in the preceding equation is positive definite and consequently $\dot{V} \leq -\gamma V$ for some $\gamma > 0$.

Next we show that $u_d = \dot{v}_d - Ke_2 + ge_z$, given by (24) and $|1 + \hat{u}_{d,3}|$ stay away from zero. By the choice of the desired trajectory (Assumption DT), it can be seen that $e_2(0) = 0_{9\times 1}$, $\tilde{q}(0) = [1 \; 0 \; 0 \; 0]^T$, and $\omega_d(0) = 0$. Thus $V(\tilde{q}(0), \tilde{\omega}_r(0), e_2(0)) \leq \frac{1}{2}\|I_f\|\|\tilde{\omega}(0)\|^2$. Since $V(\tilde{q}, \tilde{\omega}_r, e_2)$ is non-increasing, it follows that $\|e_2\| \leq \|\tilde{\omega}(0)\|\sqrt{\|I_f\|/\lambda_{\min}(P)}$ and $|\tilde{q}_0| \geq 1 - k_q^{-1}\|I_f\|\|\tilde{\omega}(0)\|^2/4$. Therefore, if $\|\tilde{\omega}(0)\|$ is sufficiently small, both $\|u_d\| = \|\dot{v}_d + ge_z - Ke_2\| \geq \delta$ and $|1 + \hat{u}_{d,3}| = |\tilde{q}_0| \geq \delta$ for some $\delta > 0$ and all time t. Consequently, by the Laypunov's theorem (Theorem 4.10 in [16]) all the errors go to zero exponentially. □

IV. Simulation and Implementation

The following simulation and implementation results are based on the velocity control law, which is a modification of (24) given by setting $K_i = 0_{3\times 3}$:

$$u_d = \dot{v}_d - K_v\tilde{v} - K_x\tilde{x} + ge_z \tag{38}$$

Desired velocities, and not positions, were set as the reference inputs. This decision was made to allow the usage of RC (radio controlled) transmitters with the controller. Note that the horizontal velocity of the aircraft was controlled along the direction in which the aircraft was headed and not relative to the global frame $\mathcal{I}$ (i.e., along the direction of $e_{a,x}$ and $e_{a,y}$). Furthermore, the controller was designed according to the diagram shown in Fig. 2. The rotor velocities Ω, which are the final inputs the quadrotor receives, were obtained with matrix M defined by (11).

Simulations and implementations were both done by modifying an open source software provided by Ardupilot (http://ardupilot.org/ardupilot/); both under a discrete environment with a sampling frequency of 400 Hertz. Simulations were done in a SITL (Software in the Loop) environment, where the default aircraft model provided by the software was used; the mass properties of which are shown in Table I. The inertia properties are approximations, based on $I_{xx} \approx I_{yy} \approx \frac{1}{2}I_{zz}$. Compensation for gyroscopic torques were not included during simulations as the simulation software did

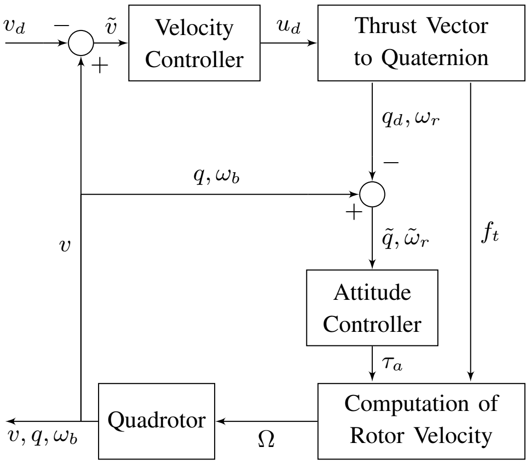


Figure 2: Schematic of Control Process

Table I: Mass Property of Simulation Aircraft

Property	Parameter	Value
Mass in kg	m	1.5
Inertia about x & y-axes in $kg \cdot m^2$	I_{xx} & I_{yy}	1.81×10^{-2}
Inertia about z-axis in $kg \cdot m^2$	I_{zz}	3.62×10^{-2}

not simulate the dynamics of the gyroscopic term. Actual testing of the controller was done outdoors to demonstrate the controllers ability to adapt to disturbances, mainly the effects of wind. Table II and III each show the individual equipments used during implementation and the mass properties of the assembled quadrotor. Furthermore, the controller gains for both simulation and implementation can be found in Table IV. Note that an additional lead compensation, in terms of $\dot{\omega}_r$, was added to the control law (17) during implementation. The additional term compensates for sensor lag, and adds additional damping to the transient response. For clarity, let $K_{\omega,p}$ be the original K_ω and $K_{\omega,d}$, which is also a 3×3, symmetric, positive definite matrix, be the gain of the added term.

Three types of inputs, which include a step input, a sinusoidal input, and finally a planned velocity input to demonstrate a 360° flip, were simulated to demonstrate the performance of the designed controller. The flip motion was simulated with high velocity step inputs in the e_z axis direction. Feeding the controller a large negative velocity input would result in a thrust vector which corresponds to a flipped attitude of the aircraft (i.e., $\frac{u_d}{\|u_d\|} = [0 \;\; 0 \;\; -1]^T$). Subsequently stabilizing the quadrotor back at zero (i.e., $\frac{u_d}{\|u_d\|} = [0 \;\; 0 \;\; 1]^T$) would complete the flip. Also, simultaneously feeding the controller with an initial horizontal step velocity input sets the direction of the flip; a step input was initially given along the direction of $e_{a,x}$ so that the flip would occur in the pitch direction. Figs. 3a and 3b each show the results of the simulated step and sinusoidal responses, and Fig. 3c shows the attitude response of the flip motion.

Table II: Experimental Equipment

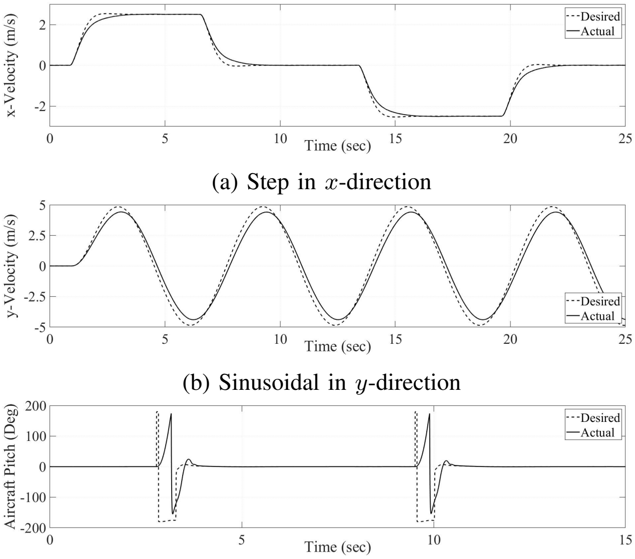
Type	Description
Flight Controller	Holybro Pix32 “Pixhawk”
GPS/Compass Module	Holybro Ublox Neo-M8N
Frame	DJI Flame Wheel F450
Motor	DJI 2312/960kV
ESC	DJI 420 Lite
Propeller	DJI 9450
Battery	Flouren Li-Po RC Battery, 4S, 5500 mAh

Table III: Quadrotor Model Parameters

Property	Parameter	Value
Mass in kg	m	1.207
Inertia about x-axis in $kg \cdot m^2$	I_{xx}	1.267×10^{-2}
Inertia about y-axis in $kg \cdot m^2$	I_{yy}	1.239×10^{-2}
Inertia about z-axis in $kg \cdot m^2$	I_{zz}	2.359×10^{-2}
Propeller Drag Coefficient	b	1.031×10^{-5}
Propeller Lift Coefficient	κ	1.973×10^{-6}

Table IV: Control Gains

Gain Description	Parameter	Value	
		Simulation	Implementation
Linear Velocity	K_v	$2.5I_{3\times3}$	$2.5I_{3\times3}$
Linear Position	K_x	$4I_{3\times3}$	$4I_{3\times3}$
Quaternion	k_q	1.9684	0.8846
ω_r Proportional	$K_{\omega,p}$	$0.2263I_{3\times3}$	$0.0663I_{3\times3}$
ω_r Derivative	$K_{\omega,d}$	-	$0.0141I_{3\times3}$
Variable Gain 1	λ_q	0.1250	0.012
Variable Gain 2	λ_u	0.3426	0.0343



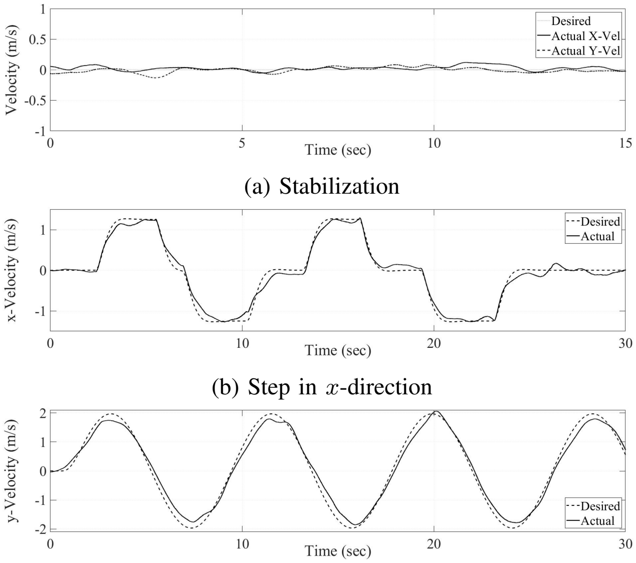
(a) Step in x-direction

(b) Sinusoidal in y-direction

(c) Flip

Figure 3: Simulation Results

The following implementation results demonstrate the controller stabilizing at zero reference, and tracking a step and sinusoidal reference. Fig. 4 shows implementation results of the three test cases. Fig. 4a shows the horizontal velocities as the quadrotor stabilizes about $v_d = [0 \;\; 0 \;\; 0]^T$ on a moderately windy day. The controller successfully rejected the disturbances and stayed close to v_d. Figs. 4b and 4c show the quadrotor's response to step and sinusoidal inputs, respectively. In each case, the controller closely tracked the desired states in the presence of some disturbance.



(a) Stabilization

(b) Step in x-direction

(c) Sinusoidal in y-direction

Figure 4: Implementation Results

V. Conclusion

This paper presents a dual-loop quaternion-based nonlinear controller, which aims to control the position or velocity of the quadrotor. The outer controller determines desired attitudes to satisfy a constant or time-varying desired velocity or position. The inner controller computes desired aircraft torques based on a desired orientation computed in the outer velocity/position controller. It is shown that adding a variable gain, Λ_q, leads to an exponentially stable nonlinear controller. By using quaternion, rather than Euler or DCM rotations, the attitude controller can track any arbitrary orientation without singularities. Note that the implementation results highly rely on measurements produced by "low-cost" sensors, which were filtered out at the expense of the speed of the controller. As a result, terms computed through numerical differentiation, though partly filtered, added some noise to the response. Though the noise did not hurt the stability of the controller, it caused minor attitude offsets. In fact, faster and more accurate tracking would be achieved if better sensor values were used. Even so, the use of these sensors can be justified, since the main objective of the experiments was not speed or attitude accuracy, but rather to demonstrate stability, and velocity trajectory tracking performance.

Appendix

A. Proof of Lemmas 1 and 2

Lemma 1. *The time derivative of the term* $(q-1)^T(q-1)$ *in (20) may be expressed as:* $\frac{d}{dt}(q-1)^T(q-1) = \bar{\omega}^T q$.

Proof. Keeping in mind of (2), the considered expression can be simplified as: $(q-1)^T(q-1) = 2(1-q_0)$ where q_0 is the scalar part of the quaternion. The proof is complete by noting that $\dot{q}_0 = -\frac{1}{2}\bar{\omega}^T q$. □

Lemma 2. *The norm of the input error* $\tilde{u}$ *satisfies* $\|\tilde{u}\| \leq 2\|u_d\|\|\tilde{q}_v\|$.

Proof. Consider $\|\tilde{u}\|^2$, which is given by:

$$\|\tilde{u}\|^2 = \tilde{u}^T\tilde{u} = (\frac{f_t}{m})^2(\tilde{e}_z - e_z)^T(\tilde{e}_z - e_z) \tag{39}$$

The term $(\tilde{e}_z - e_z)^T(\tilde{e}_z - e_z)$ can also be given by:

$$(\tilde{e}_z - e_z)^T(\tilde{e}_z - e_z) = 2(1 - \tilde{e}_z^T e_z) \tag{40}$$

In view of (2), $\tilde{e}_z^T e_z = 1 - 2[(\tilde{q}_{v,1})^2 + (\tilde{q}_{v,2})^2]$. Hence, one can conclude the following:

$$\|\tilde{u}\|^2 = 4(\frac{f_t}{m})^2[(\tilde{q}_{v,1})^2 + (\tilde{q}_{v,2})^2] \tag{41}$$

Consequently, in view of (41), the proof is complete. □

References

[1] E. Fresk and G. Nikolakopoulos, "Full quaternion based attitude control for a quadrotor," in *Proc. of the 2013 European Control Conference (ECC)*, July 2013, pp. 3864–3869.

[2] J. Diebel, "Representing attitude: Euler angles, unit quaternions, and rotation vectors," *Matrix*, vol. 58, 01 2006.

[3] A. Tayebi and S. McGilvray, "Attitude stabilization of a VTOL quadrotor aircraft," *IEEE Transactions on Control Systems Technology*, vol. 14, no. 3, pp. 562–571, May 2006.

[4] J. Cariño Escobar, H. Abaunza González, and P. Castillo Garcia, "Quadrotor quaternion control," in *Proc. of the 2015 International Conference on Unmanned Aircraft Systems (ICUAS)*, June 2015, pp. 825–831.

[5] H. Liu, X. Wang, and Y. Zhong, "Quaternion-based robust attitude control for uncertain robotic quadrotors," *IEEE Transactions on Industrial Informatics*, vol. 11, no. 2, pp. 406–415, April 2015.

[6] J. T.-Y. Wen and K. Kreutz-Delgado, "The attitude control problem," *IEEE Transactions on Automatic Control*, vol. 36, no. 10, pp. 1148–1162, Oct 1991.

[7] B. T. Costic, D. M. Dawson, M. S. De Queiroz, and V. Kapila, "A quaternion-based adaptive attitude tracking controller without velocity measurements," in *Proc. of the 39th IEEE Conference on Decision and Control (Cat. No.00CH37187)*, vol. 3, Dec 2000, pp. 2424–2429 vol.3.

[8] N. Cao and A. F. Lynch, "Inner–outer loop control for quadrotor UAVs with input and state constraints," *IEEE Transactions on Control Systems Technology*, vol. 24, no. 5, pp. 1797–1804, Sep. 2016.

[9] D. Xia, L. Cheng, and Y. Yao, "A robust inner and outer loop control method for trajectory tracking of a quadrotor," *Sensors*, vol. 17, p. 2147, 09 2017.

[10] D. Lee, H. Jin Kim, and S. Sastry, "Feedback linearization vs. adaptive sliding mode control for a quadrotor helicopter," *International Journal of Control, Automation and Systems*, vol. 7, no. 3, pp. 419–428, Jun 2009. [Online]. Available: https://doi.org/10.1007/s12555-009-0311-8

[11] M. Herrera, W. Chamorro, A. P. Gómez, and O. Camacho, "Sliding mode control: An approach to control a quadrotor," in *Proc. of the 2015 Asia-Pacific Conference on Computer Aided System Engineering*, July 2015, pp. 314–319.

[12] T. Madani and A. Benallegue, "Backstepping control for a quadrotor helicopter," in *Proc. of the 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Oct 2006, pp. 3255–3260.

[13] M. Schreier, "Modeling and adaptive control of a quadrotor," in *Proc. of the 2012 IEEE International Conference on Mechatronics and Automation*, Aug 2012, pp. 383–390.

[14] Z. Shulong, A. Honglei, Z. Daibing, and S. Lincheng, "A new feedback linearization LQR control for attitude of quadrotor," in *Proc. of the 2014 13th International Conference on Control Automation Robotics Vision (ICARCV)*, Dec 2014, pp. 1593–1597.

[15] M. A. Lotufo, L. Colangelo, and C. Novara, "Control design for UAV quadrotors via embedded model control," *IEEE Transactions on Control Systems Technology*, pp. 1–16, 2019.

[16] K. Hassan K, *Nonlinear systems; 3rd ed.* Upper Saddle River, NJ: Prentice-Hall, 2002, ch. 4.