ICAS 2016
30th Congress of the International Council of the Aeronautical Sciences
DCC, Daejeon, Korea ; September 25-30, 2016

A NOVEL FLIGHT CONTORL ALGORITHM FOR MULTICOPTERS

Ozan Tekinalp*, Sarper Kumbasar**
Middle East Technical University
Keywords: *automatic flight control, nonlinear control, attitude parametrization, vertical take-off and landing, multicopter*

Abstract

A new nonlinear control algorithm is proposed for multi-copter type vertical takeoff vehicles. The algorithm uses linear controllers for the position control in the outer loop. The inner loop attitude control is carried out using quaternion representation. The necessary thrust vector requirements of the position control are converted to inner loop as attitude control commands. The inner loop uses a Lyapunov function based attitude controller utilizing the to-go quaternion calculated from thrust vector commands. The yaw commands are also added to the to-go quaternion to realize the attitude control in one step.

1 Introduction

VTOL aircraft has been attracting attention for many decades. Quadrotors or as sometimes called multi-copters, are uninhabited air vehicles mainly employed for rapid reconnaissance, aerial photography etc. Although they are not aerodynamically efficient as helicopters, they are cheaper than helicopters, and may be made more reliable with additional propellers and individual motor controllers.

The control of multi-copters on the other hand poses some difficulties due to the nonlinearities in the equations of motion. They are more maneuverable than aircraft, and steady flight conditions or trim conditions, are quite numerous. Consequently, linear controllers, although extensively used, may offer limited flight performance.

Use of quaternions for the attitude control of quadrotors have been addressed in the literature. For example, Tayebi [2] uses a Lyapunov function containing body angular velocities and quaternions, similar to Reference [3]. However, the attitude commands calculated from position control algorithm in terms of additional roll, pitch, yaw commands needs to be converted to a quaternion command for the attitude control system. Other example of quaternion utilization may also be given ([4][5][6]). However, all of them fall short in properly bridging the gap between the position control and obtaining the quaternion for attitude control. This manuscript addresses this issue.

In this manuscript a new flight control algorithm is proposed. It is a two loop algorithm, where the outer loop controls the position, and the inner loop controls the attitude. In the inner loop, it uses quaternion based attitude parametrization. The thrust vector commands of the outer loop is converted into attitude commands, to be realized by the nonlinear inner loop control.

In the following, the problem formulation and the control algorithms are given. It is followed by simulation results and discussion. Finally conclusions are given.

2 Formulation of the Problem

2.1 Multi-copter Equations of Motion

The translational equations of motion may be written with respect to the navigation frame, such as local vertical and local horizontal frame (i.e., North, East, Down, NED frame). Assuming that the NED frame is an inertial frame, the equations of motion may be written as,

$$\begin{aligned} m\dot{V}_X &= F_{XC} - D_X \\ m\dot{V}_Y &= F_{YC} - D_Y \\ m\dot{V}_Z &= F_{ZC} - D_Z - mg \end{aligned} \tag{1}$$

where, V_i , is the inertial velocity, F_{iC} control force, D_i, is the drag force associated with the i'th axis. Drag force may be taken proportional to the velocity vector [1],

$$\mathbf{D} = k_d (V_X, V_Y, V_Z)^T \tag{2}$$

The total angular momentum of the multi-copter, written in the body fixed frame is,

$$\mathbf{H} = \mathbf{J}\boldsymbol{\omega} + \mathbf{h} \tag{3}$$

where, $\mathbf{J}$ is the inertia tensor, $\boldsymbol{\omega}$ is the angular velocity of the vehicle with respect to the inertial frame, and $\mathbf{h}$, is the residual angular momentum of the rotors. Since the rotors are arranged to be counter rotating, the residual angular momentum of the rotors should normally be small. Taking the derivative of Eq. (3), the following equations of motion are obtained.

$$\dot{\boldsymbol{\omega}} = \mathbf{J}^{-1}\left\{-\dot{\mathbf{h}} - \boldsymbol{\omega}^{\times}\left(\mathbf{J}\boldsymbol{\omega} + \mathbf{h}\right) + \mathbf{T} + \mathbf{T}_d\right\} \tag{4}$$

where,

$$\boldsymbol{\omega}^{\times} = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix} \tag{5}$$

In Eq. (4), $\mathbf{T}$, is the control torque, while $\mathbf{T}_d$ is the disturbance torque, acting on the multi-copter. Since the multi-copter rotors are along the z-body axis direction, assuming that all the rotors are identical, the residual angular momentum of the rotors may approximately be written as,

$$\mathbf{h} = \mp J_{res} \Omega_{res} \mathbf{k} \tag{6}$$

2.2 Defining the To-Go Quaternion

A quaternion, q, with vector ($\mathbf{q}$), and scalar (q_4), parts may be written as, $q = (\mathbf{q} + q_4)$. Similarly using quaternion multiplication and defining the desired attitude, d , current attitude, q, the to-go quaternion, t, may be written as,

$$d = qt \tag{7}$$

Using the inverse or conjugate quaternion, the following may also be written:

$$t = q^{-1} d \tag{8}$$

Applying the quaternion multiplication rules, the *to-go* quaternion becomes,

$$\begin{aligned} t &= (-\mathbf{q} + q_4)(\mathbf{d} + d_4) \\ &= -\mathbf{q} \times \mathbf{d} - d_4 \mathbf{q} + q_4 \mathbf{d} + \mathbf{q} \cdot \mathbf{d} + d_4 q_4 \end{aligned} \tag{9}$$

Or in matrix vector form,

$$\begin{Bmatrix} t_1 \\ t_2 \\ t_3 \\ t_4 \end{Bmatrix} = \begin{bmatrix} -d_4 & -d_3 & d_2 & d_1 \\ d_3 & -d_4 & -d_1 & d_2 \\ -d_2 & d_1 & -d_4 & d_3 \\ d_1 & d_2 & d_3 & d_4 \end{bmatrix} \begin{Bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{Bmatrix} \tag{10}$$

or,

$$\begin{Bmatrix} \mathbf{t} \\ t_4 \end{Bmatrix} = \mathbf{D} \begin{Bmatrix} \mathbf{q} \\ q_4 \end{Bmatrix} \tag{11}$$

Then, the derivative of the *to-go* may be written as,

$$\begin{Bmatrix} \dot{\mathbf{t}} \\ \dot{t}_4 \end{Bmatrix} = \dot{\mathbf{D}} \begin{Bmatrix} \mathbf{q} \\ q_4 \end{Bmatrix} + \mathbf{D} \begin{Bmatrix} \dot{\mathbf{q}} \\ \dot{q}_4 \end{Bmatrix} \tag{12}$$

Normally, the desired attitude is fixed and its derivative is zero. The derivative of quaternion may be written as,

$$\begin{aligned} \begin{Bmatrix} \dot{\mathbf{q}} \\ \dot{q}_4 \end{Bmatrix} &= \frac{1}{2} \begin{bmatrix} -\tilde{\boldsymbol{\omega}} & \boldsymbol{\omega} \\ -\boldsymbol{\omega}^T & 0 \end{bmatrix} \begin{Bmatrix} \mathbf{q} \\ q_4 \end{Bmatrix} \\ &= \frac{1}{2} \boldsymbol{\Omega} \begin{Bmatrix} \mathbf{q} \\ q_4 \end{Bmatrix} \end{aligned} \tag{13}$$

Similarly, using Eq. (12) and Eq. (13), the derivative of the to-go quaternion may be found as,

$$\begin{aligned} \dot{\mathbf{t}} &= -\frac{1}{2} \tilde{\boldsymbol{\omega}} \mathbf{t} - \frac{1}{2} \boldsymbol{\omega} t_4 \\ \dot{t}_4 &= -\frac{1}{2} \boldsymbol{\omega}^T \mathbf{t} \end{aligned} \tag{14}$$

2.3 Control Algorithms

If the drag forces are neglected, the translational equations given in Eq. (1), are linear, as well as uncoupled. Thus, PID controllers may be used to calculate the desired control forces for position control. However, in this manuscript a linear quadratic tracking controller is used. The state vectors for each channel are taken as,

$$\begin{Bmatrix} X \\ V_X \end{Bmatrix}, \begin{Bmatrix} Y \\ V_Y \end{Bmatrix}, \begin{Bmatrix} Z \\ V_Z \end{Bmatrix} \tag{15}$$

Then, for a horizontal channel, system and input and measurement matrices becomes,

$$\mathbf{A} = \begin{bmatrix} 0 & 1 \\ 0 & -k_{fd}/m \end{bmatrix},\ \mathbf{B} = \begin{Bmatrix} 0 \\ 1/m \end{Bmatrix}, \mathbf{C} = \{1 \quad 0\} \quad (16)$$

Then, the feedback law may be written as [7],

$$\mathbf{u}(t) = \mathbf{K}\mathbf{x}(t) + \mathbf{K}_z\mathbf{z}(t) \quad (17)$$

where, $\mathbf{K}$ and $\mathbf{K}_z$ are the gains associated with the states and the reference input, $\mathbf{z}$, of the infinite horizon linear quadratic tracking controller, and the quadratic performance index is given as,

$$J = \frac{1}{2}\int_0^\infty \{\mathbf{e}^T(t)\mathbf{Q}(t)\mathbf{e}(t) + \mathbf{u}^T(t)\mathbf{R}(t)\mathbf{u}(t)\}dt \quad (18)$$

where, $\mathbf{e}(t) = \mathbf{z}(t) - \mathbf{y}(t)$, is the tracking error.

2.4 Attitude Control

For attitude control, a nonlinear controller is needed. Consider the following positive definite Lyapunov function,

$$V = \frac{1}{2}\boldsymbol{\omega}^T\mathbf{K}^{-1}\mathbf{J}\boldsymbol{\omega} + 2(1 + t_4) \quad (19)$$

Here, it is assumed that a positive definite, $\mathbf{K}^{-1}$ exists and $\mathbf{K}^{-1}\mathbf{J}$ is also a positive definite symmetric matrix. Then, the derivative of the Lyapunov function becomes:

$$\dot{V} = \boldsymbol{\omega}^T\mathbf{K}^{-1}\dot{\boldsymbol{\omega}} - \boldsymbol{\omega}^T\mathbf{t} \quad (20)$$

A decay rate selected as, $\dot{V} = -\boldsymbol{\omega}^T\mathbf{K}^{-1}\mathbf{D}\boldsymbol{\omega}$, where $(\mathbf{K}^{-1}\mathbf{D})$ is again a positive definite matrix, guarantees that the Lyapunov function, V, keeps decreasing. Substituting Eq. (4), to Eq. (20), the following control law may be obtained.

$$\mathbf{u} = -\dot{\mathbf{h}} - \boldsymbol{\omega}^\times(\mathbf{J}\boldsymbol{\omega} + \mathbf{h}) + \mathbf{K}\mathbf{t} - \mathbf{D}\boldsymbol{\omega} \quad (21)$$

Note that in Eq. (21), the disturbance torque is neglected. To realize the conditions on matrices $\mathbf{K}$ and $\mathbf{D}$, one may choose them proportional to the inertia matrix (i.e., $\mathbf{K} = k\mathbf{J}$ and $\mathbf{D} = d\mathbf{J}$).

2.5 Mechanization of the Attitude Control

To carry out the control, it must be realized that the multi-copter has all its propellers generating the thrust, perpendicular to the multi-copter plane. In the body fixed coordinates, the thrust vector direction is:

$$\boldsymbol{\gamma}_B = -\mathbf{k} \quad (22)$$

The, thrust direction is realized by rolling and pitching the quadrotor platform. The total force shall be equal to the total control force required,

$$F_C = \sqrt{F_{XC}^2 + F_{YC}^2 + F_{ZC}^2} \quad (23)$$

In the navigation frame, assume that the desired thrust direction dictated by the position control system is given as,

$$\boldsymbol{\lambda}_N = \frac{F_{XC}\mathbf{I} + F_{YC}\mathbf{J} + F_{ZC}\mathbf{K}}{\sqrt{F_{XC}^2 + F_{YC}^2 + F_{ZC}^2}} \quad (24)$$

In the body fixed coordinates, using the transformation matrix from NED frame to body fixed frame, C_N^B the trust vector direction may be written as,

$$\boldsymbol{\lambda}_B = \mathbf{C}_N^B\boldsymbol{\lambda}_N \quad (25)$$

From these two unit vectors, the to-go quaternion may be found. First, the axis of rotation may be calculated from the cross product,

$$\boldsymbol{\eta}_B = \boldsymbol{\gamma}_B \times \boldsymbol{\lambda}_B \quad (26)$$

The angle to be rotated may be found from the dot product of these two vectors,

$$\beta = \cos^{-1}(\boldsymbol{\gamma}_B \cdot \boldsymbol{\lambda}_B) \quad (27)$$

Then, from the definition of quaternions, the to-go quaternion may be found as follows:

$$\begin{aligned} \mathbf{s} &= \boldsymbol{\eta}_B \sin(\beta/2) \\ s_4 &= \cos(\beta/2) \end{aligned} \quad (28)$$

2.6 Adding a Yaw Rotation

To include a yaw control a sequence of rotations shall be defined. Going from the inertial (navigation) frame to the body fixed frame the sequence is normally defined as yaw-pitch-roll. In this case, we will use the opposite. Hence roll-pitch-yaw sequence will be used, making the yaw as the last rotation. While propagating the Euler equations in the navigation computer, the current Euler angles (ϕ,θ,ψ) are obtained. The transformation matrix from the NED frame to the body fixed frame, may be written using these Euler angles as,

$$\mathbf{C}_N^B = \begin{bmatrix} c\psi & s\psi & 0 \\ -s\psi & c\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c\theta & 0 & -s\theta \\ 0 & 1 & 0 \\ s\theta & 0 & c\theta \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\phi & s\phi \\ 0 & -s\phi & c\phi \end{bmatrix} \tag{29}$$

If an increment in the yaw angle ($\Delta\psi$) is desired (i.e., $\psi_d = \psi + \Delta\psi$), the transformation will matrix then be,

$$\mathbf{D}_N^B = \begin{bmatrix} c\psi_d & s\psi_d & 0 \\ -s\psi_d & c\psi_d & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c\theta & 0 & -s\theta \\ 0 & 1 & 0 \\ s\theta & 0 & c\theta \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\phi & s\phi \\ 0 & -s\phi & c\phi \end{bmatrix} \tag{30}$$

Assuming that we have already completed the desired yaw rotation, the desired thrust vector direction in the body fixed coordinates may be found from,

$$\boldsymbol{\lambda}_B = \mathbf{D}_N^B \boldsymbol{\lambda}_N \tag{31}$$

Then, the necessary rotation vector is again calculated using the Eq. (24)-(28). The, additional yaw angle to be rotated may be written as the following quaternion,

$$\begin{aligned} \mathbf{y} &= \mathbf{k}\sin\left(\Delta\psi/2\right) \\ y_4 &= \cos\left(\Delta\psi/2\right) \end{aligned} \tag{32}$$

Then, the total *to-go* quaternion is found as,

$$t = sy \tag{33}$$

2.7 Control Allocation

If we assume that there are four rotors (i.e. quad-rotor configuration), then the relation between the thrust of each rotor and the control torque may be written as,

$$\mathbf{u} = \begin{Bmatrix} \left(F_4 - F_2\right) l_y \\ \left(F_3 - F_1\right) l_x \\ T_l - T_3 + T_2 - T_4 \end{Bmatrix} \tag{34}$$

Where, F_i, is the thrust of i^{th} rotor, and T_i is the associated torque of the named rotor. Assuming that, $F_i = k_{fi}\Omega_i^2$, and $T_i = k_{ti}F_i$, then the following may be written,

$$\begin{Bmatrix} u_1 \\ u_2 \\ u_3 \\ F_C \end{Bmatrix} = \begin{bmatrix} 0 & -k_{f2}l_{y2} & 0 & k_{f4}l_{y4} \\ -k_{f1}l_{x1} & 0 & k_{f3}l_{x3} & 0 \\ k_{f4}k_{t1} & k_{f4}k_{t2} & -k_{f4}k_{t3} & -k_{f4}k_{t4} \\ k_{f1} & k_{f2} & k_{f3} & k_{f4} \end{bmatrix} \begin{Bmatrix} \Omega_1^2 \\ \Omega_2^2 \\ \Omega_3^2 \\ \Omega_4^2 \end{Bmatrix} \tag{35}$$

Then the rotor angular velocities to be realized may be found from Eq. (35) by inversion. Here a quadrotor is considered. For multi-copters with more than four rotors, a proper control allocation algorithm, such as Moore-Penrose pseudo-inverse may be used.

3. Simulation Results and Discussion

A nonlinear simulation for a quadrotor with properties given in Table 1 is used to carry out simulations. Various command are given to the simulation. The first command starts at 10 s and executed for 10 s commanding the quadrotor in the East direction, and increase its altitude by 10 m. The second maneuver is a heading maneuver initiated at 30 s. In this case the heading is changed from by some 40^0 starting at 30 s. Finally, another horizontal maneuver towards North starting at 60 s is given. The results of the simulation are presented in Figures 1-6. Figure 1, shows the horizontal motion of the quadrotor. Translational motion of the quadrotor is presented in Figure 2. Both figures indicate that the commands are very well followed by the feedback control system. Attitude histories are given in Figure 3. The realization of the commanded heading angle may also be observed from this figure. The total thrust presented in Figure 4 shows that it increases during the horizontal maneuver as expected since the propellers not only have to overcome the weight, but also create horizontal acceleration and overcome the drag force. The amount the thrust requested is within the limits of propellers. Propeller speeds are presented in Figure 5 also show that these speeds are within the limits of the propellers.

Conclusions

A new feedback control algorithm is developed to control quadrotors. The algorithm uses linear quadratic tracking controller in the outer loop, while it uses a nonlinear attitude controller in the inner loop. The nonlinear attitude control uses the to-go quaternion obtained from the force commands of the outer loop. The success of the controller is demonstrated through a simulation.

Table 1. Properties of the quadrotor used in the simulations [1].

m	0.468	kg
g	9.81	m/s^2
d	0.225	m
J_r	3.357×10^{-5}	$kg.m^2$
J_{xx}	4.856×10^{-3}	$kg.m^2$
J_{yy}	4.856×10^{-3}	$kg.m^2$
J_{zz}	8.801×10^{-3}	$kg.m^2$
b	2.98×10^{-6}	$N.s^2/rad^2$
k	1.14×10^{-7}	$N.m.s^2/rad^2$
k_{fd}	0.5	kg/s
k_{td}	0.05	$kg.m^2/(s.rad)$

References

[1] Zuo, Zungyu. ”Trajcetory tracking control design with command filtered compensation for a quadrotor,” *Control Theory and Applications*, Vol. 4, No. 11, pp. 2343-2355, 2010.

[2] Tayebi, Abdelhamid, and Stephen McGilvray. ”Attitude stabilization of a VTOL quadrotor aircraft.” Control Systems Technology, Vol. 14, No. 3, pp. 562-571, 2006.

[3] Wie, B., Weiss, H., and Arapostathis, A., “Quaternion Feedback Regulator for Spacecraft Eigenaxis Rotations,” *J Guidance*, Vol. 12, No. 3, pp. 375-380, 1989.

[4] Xiao, Bin, Diao, Chen, Zhaou, Bo., Yao, Zhang, ”Nonlinear Output Feedback Tracking Control of a Quadrotor UAV Using Quaternion Representation,” *Nonlinear Dynamics*, Vol. 79, pp. 2735-2752, 2015.

[5] Fresk, Emil, Nikolakopoulos, George, “Full Quaternion Based Attitude Control for a Quadrotor,” *European Control Conference*, July 17-19, Zurich, Switzerland, 2013.

[6] Reyes-Valeria, Elias, Enriques-Caldera, Rogeria, Camacho-Lara, Sergio, Guichard, Jose, “LQR Control of Quadrotor Using Quaternions: Modeling and Simulation,” *Proceedings of the 2013 International Conference on Electronics, Communications and Computing*, pp. 172-178, March 11-13, 2013.

[7] Naidu, Desineni S., *Optimal Control Systems*, CRC Press, Boca Ratton, 2002.

[8] S. Kayastha, O. Tekinalp, M. K. Ozgoren, M.K., “Quaternion Based State Dependent Riccati Equation Control of a Satellite Camera on Piezoelectric Actuators,” *Proceedings of the AIAA Astrodynamics Specialist Conference*, Toronto, Canada, 2010.

Appendix: Mechanization Using Quaternions

Quaternions may also be used for the above mechanization as well. If the current attitude is defined by q, then the desired attitude after the desired yaw rotation, $\Delta\psi$, may be written as,

$$q_1 = qy$$

The thrust direction in the body fixed coordinate frame may then be calculated using quaternion multiplication. Then, Eq. (31) may be written as,

$$g = q_1^{-1} f q_1$$

where,

$$\mathbf{f} = \boldsymbol{\lambda}_N, f_4 = 0$$

$$\boldsymbol{\lambda}_B = \mathbf{g}$$

The yaw angle may be obtained from the current quaternion easily. For the sequence described by Eq. (29),

$$\mathbf{C}_N^B = \begin{bmatrix} c\psi c\theta & c\psi s\theta s\phi + s\psi c\phi & -c\psi s\theta c\phi + s\psi s\phi \\ -s\psi c\theta & -s\psi s\theta s\phi - c\psi c\phi & s\psi s\theta c\phi + c\psi s\phi \\ s\theta & 0 & 0 \end{bmatrix}$$

Since the same transformation matrix written using quaternions is,

$$\mathbf{C}_N^B = \begin{bmatrix} 1-2(q_2^2+q_3^2) & 2(q_1q_2+q_3q_4) & 2(q_1q_3-q_2q_4) \\ 2(q_1q_2-q_3q_4) & 1-2(q_1^2+q_3^2) & 2(q_2q_3+q_1q_4) \\ 2(q_1q_3+q_2q_4) & 2(q_1q_3-q_2q_4) & 1-2(q_1^2+q_2^2) \end{bmatrix} \quad (36)$$

Then the, yaw angle may be found from,

$$\tan\psi = -\frac{c_{21}}{c_{11}} = \frac{1-2(q_2^2+q_3^2)}{2(q_1q_2-q_3q_4)}$$

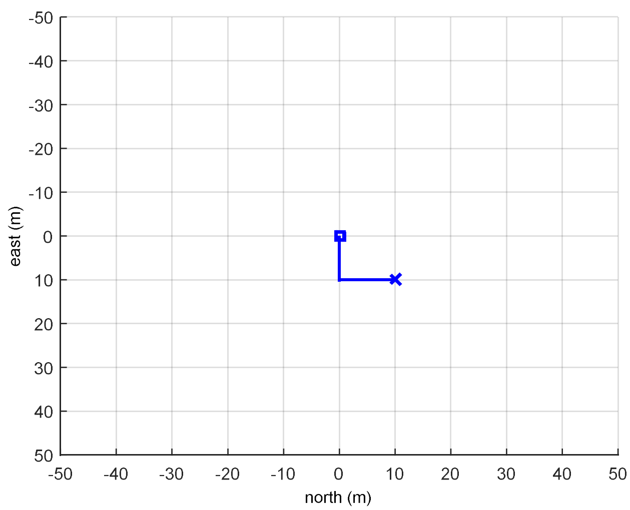

Figure 1 Motion of the quadrotor in the horizontal farme

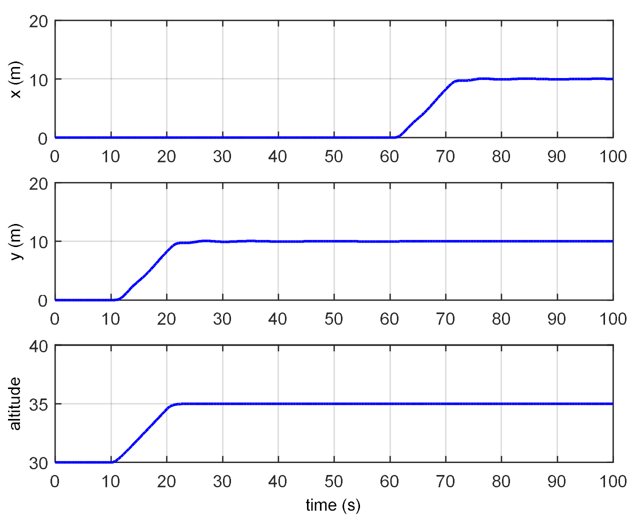

Figure 2 Quadrotor positon history in the navigation frame

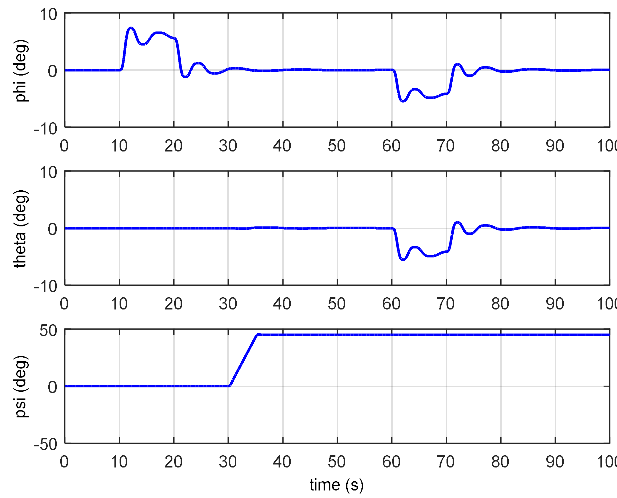

Figure 3 Atittude history

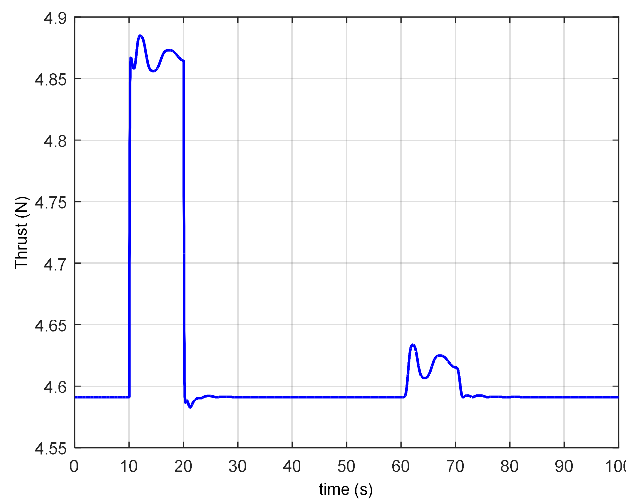

Figure 4 Total thrust history

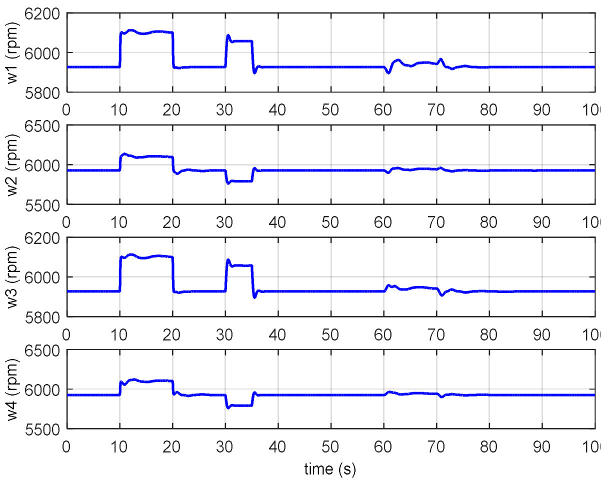

Figure 5 Propeller speeds

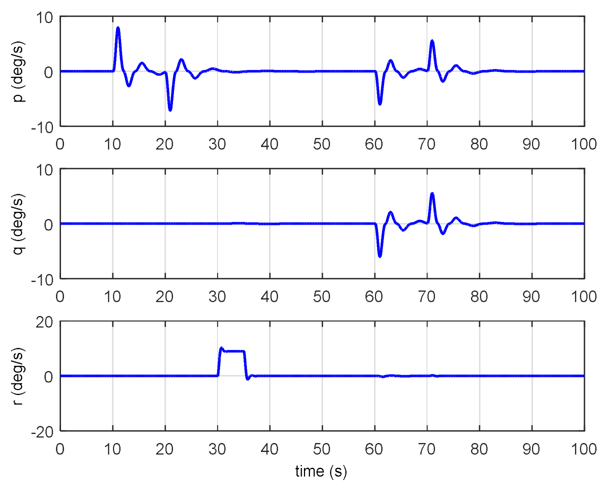

Figure 6 Angular velocities

Copyright Statement