Published in IET Control Theory and Applications
Received on 9th July 2009
Revised on 6th November 2009
doi: 10.1049/iet-cta.2009.0336

ISSN 1751-8644

Trajectory tracking control design with command-filtered compensation for a quadrotor

Z. Zuo

National Key Laboratory of Science and Technology on Holistic Control, Beijing University of Aeronautics and Astronautics, Beijing 100191, People's Republic of China
The Seventh Research Division, Beijing University of Aeronautics and Astronautics, Beijing 100191, People's Republic of China
E-mail: zzybobby@ss.buaa.edu.cn

Abstract: The design of a flight control system capable of not only stabilising attitude but also tracking a trajectory accurately for an under-actuated quadrotor aircraft is quite challenging. This study constructs the relationship between the attitude and linear acceleration of a small quadrotor unmanned aircraft and proposes a trajectory tracking control design algorithm, based on the relationship, using a new command-filtered backstepping technique to stabilise the attitude and a linear tracking differentiator to eliminate the classical inner/outer-loop structure. Finally, the validity and the effectiveness of this algorithm are demonstrated by various numerical simulations under different conditions.

1 Introduction

Quadrotor unmanned aircraft, consisting of four individual rotors of 'X' arrangement, is an attractive vertical take-off and landing unmanned aerial vehicle for both military and civilian usages. In order to compensate the effect of the reactive torques, the four rotors are divided into two pairs of (1, 3) and (2, 4) turning in the opposite direction, as depicted in Fig. 1. Hence, the quadrotor is suitable for hover and pseudo-static flight conditions.

The quadrotor is a typical under-actuated, non-linear coupled system, and Table 1 shows the quadrotor flight mechanism: vertical motion created by collectively increasing and decreasing the speed of all four rotors; pitch or roll motion is achieved by the differential speed of the front–rear set or the left–right set of rotors, coupled with lateral motion; yaw motion is realised by the different reactive torques between the (1, 3) and the (2, 4) rotors. So the number of individual manipulating variables cannot instantaneously set the accelerations in all directions of the configuration space. In spite of the four rotors, the quadrotor is still an under-actuated and non-linear coupled system.

McKerrow [1] studied the dynamical model of the four-rotor VTOL vehicles in the literature. Hamel *et al.* [2] made a slight modification of the gyroscopic torque expression on the dynamical model in [1] and designed a backstepping-based trajectory tracking controller. Tayebi and McGilvray [3] designed an attitude stabilisation PD^2 controller based on the unit quaternion description frequently adopted by attitude control problem (cf. [4, 5]) of a rigid body. Bouabdallah *et al.* [6] applied the classical PID and LQ algorithm, respectively, to the quadrotor attitude stabilisation. However, the preferable flight condition, under-actuated and strong coupled properties of this kind aircraft render the trajectory tracking control law design much more challenging. In [7], Bouabdalla achieved the quadrotor trajectory tracking by combining classical inner/outer loop with backstepping and sliding-mode techniques, respectively, but the two-time scale separation assumption needs large inner-loop gain to guarantee closed-loop stability [8]. Madani and Benallegue [9] divided the quadrotor dynamics into three subsystems using the same design methodology to track the desired trajectory via full state backstepping approach. In view of the uncertainty and unknown dynamics, Madani and Benallegue [10] presented a trajectory tracking controller

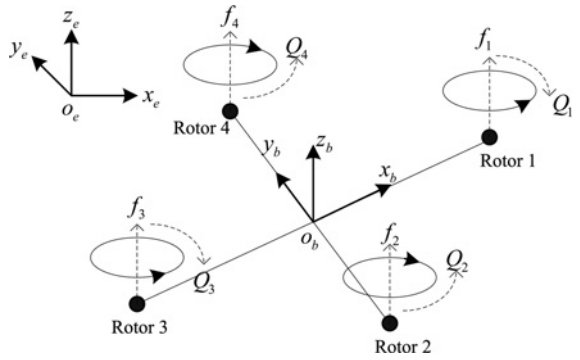

Figure 1 *Quadrotor aircraft*

Table 1 Quadrotor unmanned aircraft rotor speed control

	Rotor1	Rotor2	Rotor3	Rotor4
up	↑	↑	↑	↑
pitch	↑	–	↓	–
roll	–	↑	–	↓
yaw	↓	↑	↓	↑

based on backstepping and sliding-mode techniques which was robust to the unknown aerodynamic friction integrated in the dynamic modelling. Pollini and Metrangolo [11] proposed a robust backstepping controller based on the idea of practical stability instead of Lyapunov stability to deal with non-linearities, measurement disturbances and noise, sensor biases and actuator failure.

The motion equations of a rigid body contain translation and rotation of two components, and satisfy the strict feedback form. That is why the backstepping technique (cf. [12, 13]) is so popular in flight control systems. However, the analytic derivative expressions of pseudo control variables are usually overly complicated or unknown especially for the high-order systems or uncertain systems, which limits the backstepping technique in practical applications. This paper employs the command-filtered backstepping technique [14] to stabilise the quadrotor attitude without calculating the pseudo control signal derivative analytically, and to decrease the dependent degree on the analytic model. The trajectory tracking controller employs PD linear feedback control methodology to construct the attitude command signals. In order to avoid introducing inner and outer loop under the time scale separation assumption between the attitude and linear dynamics of the quadrotor aircraft, the paper proposes a linear tracking differentiator [15] to extract the attitude command derivative signal without tedious computation. Such a design methodology can avoid the complication to obtain the analytic expressions of the commanded signals and save the control energy through compensating these command signal dynamics in the control law.

This paper is organised as follows: Section 2 presents the detailed dynamical model of the quadrotor. Sections 3 and 4 present the trajectory tracking control design algorithm. In Section 5, numerical simulation results of a quadrotor trajectory tracking are compared and discussed. And finally the conclusion of the proposed trajectory tracking control design algorithm is presented in Section 5.

2 Quadrotor mathematical model

In order to simplify the modelling of the quadrotor and therefore make the controller design easier, several reasonable assumptions are made:

Assumption 1: Quadrotor is a rigid body. Then the quadrotor non-linear dynamics can be derived by using Newton–Euler formulas.

Assumption 2: Aerodynamic forces and moments can be ignored at low speed. Actually the aerodynamic effects are very complicated and hardly modelled, including vortex ring state, blade flapping caused by the differing inflow velocities and the effect caused by the quadrotor body in the slip stream of the rotors (cf. [16]). However, it is reasonable to neglect the aerodynamic effects at low speed for simplicity.

Assumption 3: The quadrotor is symmetrical with respect to the axes Ox, Oy and Oz.

Let $\mathcal{I} = \{O_e x_e y_e z_e\}$ denote an earth-fixed inertial frame and $\mathcal{A} = \{Oxyz\}$ a body-fixed frame whose origin O is at the centre of mass of the quadrotor, as shown in Fig. 2. The absolute position of the quadrotor is defined by $\boldsymbol{p} = (x, y, z)^{\mathrm{T}}$ and the attitude by three Euler angles $\boldsymbol{\Theta} = (\phi, \theta, \psi)^{\mathrm{T}}$. $\boldsymbol{R} \in SO(3)$ is the orthogonal rotation matrix to orient the quadrotor (see equation at the bottom of page).

The quadrotor is a six-degree-of-freedom rigid body described by three translations $\boldsymbol{v} = (v_x, v_y, v_z)^{\mathrm{T}}$ and three rotations $\boldsymbol{\Omega} = (p, q, r)^{\mathrm{T}}$. And the quadrotor motion equations [3] can be expressed as

$$\dot{\boldsymbol{p}} = \boldsymbol{v} \tag{1}$$

$$\dot{\boldsymbol{v}} = -g\boldsymbol{z}_{\mathrm{e}} + \frac{T}{m}\boldsymbol{R}\boldsymbol{z}_{\mathrm{e}} \tag{2}$$

$$\boldsymbol{R} = \begin{pmatrix} \cos\theta\cos\psi & \sin\theta\cos\psi\sin\phi - \sin\psi\cos\phi & \sin\theta\cos\psi\cos\phi + \sin\psi\sin\phi \\ \cos\theta\sin\psi & \sin\theta\sin\psi\sin\phi + \cos\psi\cos\phi & \sin\theta\sin\psi\cos\phi - \cos\psi\sin\phi \\ -\sin\theta & \cos\theta\sin\phi & \cos\theta\cos\phi \end{pmatrix}$$

doi: 10.1049/iet-cta.2009.0336

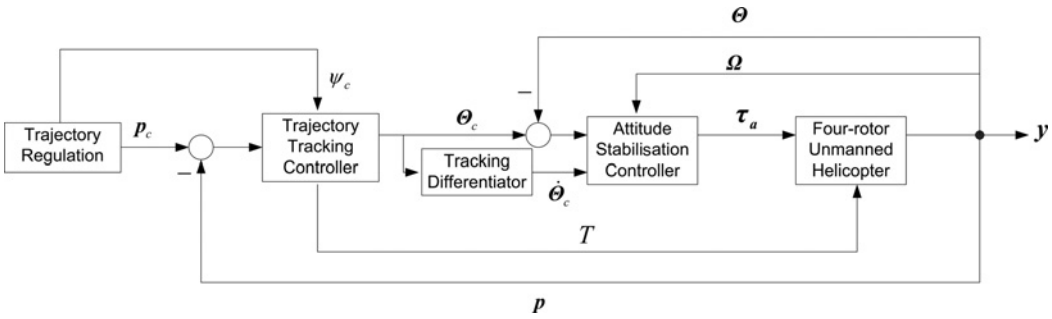

Figure 2 *Control block diagram*

$$\dot{\boldsymbol{R}} = \boldsymbol{R}\boldsymbol{S}(\boldsymbol{\Omega}) \tag{3}$$

$$\boldsymbol{I}_f\dot{\boldsymbol{\Omega}} = -\boldsymbol{\Omega}\times\boldsymbol{I}_f\boldsymbol{\Omega} - \boldsymbol{G}_a + \boldsymbol{\tau}_a \tag{4}$$

where $\boldsymbol{S}(\boldsymbol{\Omega})$ is a skew-symmetric matrix and defined as follows:

$$\boldsymbol{S}(\boldsymbol{\Omega}) = \begin{pmatrix} 0 & -r & q \\ r & 0 & -p \\ -q & p & 0 \end{pmatrix}$$

and m denotes the quadrotor mass, g the gravity acceleration, $\boldsymbol{z}_e = (0, 0, 1)^{\mathrm{T}}$ the unit vector expressed in the frame $\mathcal{I}$ and T the total thrust produced by the four rotors in free air:

$$T = \sum_{i=1}^{4} f_i = b\sum_{i=1}^{4}\omega_i^2 \tag{5}$$

The rotor dynamics can be expressed as

$$I_r\dot{\omega}_i = \tau_i - Q_i,\ (i = 1, 2, 3, 4) \tag{6}$$

where I_r and ω_i denote, respectively, the moment of inertial and the speed of the rotor i, τ_i is the electrical torque of DC motor i and Q_i is the reactive torque caused by air drag and given by

$$Q_i = k\omega_i^2 \tag{7}$$

where the parameters b and k are positive proportionality constants relative to the density of air, the shape of the blades, the number of the blades, the chord length of the blades, the pitch angle of the blade airfoil and the drag constant.

Remark 1: It is reasonable to assume that the scalars b and k are constants when the quadrotor is performed at low speed in free and still air.

Under Assumption 3, the total inertial matrix $\boldsymbol{I}_f \in \Re^{3\times3}$ is a symmetric positive definite constant matrix expressed in the frame $\mathcal{A}$. The vector $\boldsymbol{G}_a$ denotes the gyroscopic torque which is given by

$$\boldsymbol{G}_a = \sum_{i=1}^{4} I_r(\boldsymbol{\Omega}\times\boldsymbol{z}_e)(-1)^{i+1}\omega_i \tag{8}$$

If the distance from the rotors to the centre of mass is denoted by l, the control torques generated by the four rotors are

$$\boldsymbol{\tau}_a = \begin{pmatrix} \tau_a^1 \\ \tau_a^2 \\ \tau_a^3 \end{pmatrix} = \begin{pmatrix} bl(\omega_4^2 - \omega_2^2) \\ bl(\omega_3^2 - \omega_1^2) \\ k(\omega_2^2 + \omega_4^2 - \omega_1^2 - \omega_3^2) \end{pmatrix} \tag{9}$$

In order to facilitate the computation of the real control inputs, that is, $\omega_i (i = 1, 2, 3, 4)$, (9) and (5) are put together:

$$\begin{pmatrix} T \\ \tau_a^1 \\ \tau_a^2 \\ \tau_a^3 \end{pmatrix} = \begin{pmatrix} b & b & b & b \\ 0 & -bl & 0 & bl \\ -bl & 0 & bl & 0 \\ -k & k & -k & k \end{pmatrix}\begin{pmatrix} \omega_1^2 \\ \omega_2^2 \\ \omega_3^2 \\ \omega_4^2 \end{pmatrix} \triangleq \boldsymbol{M}\bar{\boldsymbol{\omega}} \tag{10}$$

After simple algebraic calculation, (3) can be rewritten as

$$\dot{\boldsymbol{\Theta}} = \boldsymbol{W}\boldsymbol{\Omega} \tag{11}$$

where

$$\boldsymbol{W} = \begin{pmatrix} 1 & \sin\phi\tan\theta & \cos\phi\tan\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi\sec\theta & \cos\phi\sec\theta \end{pmatrix}$$

and $\det(\boldsymbol{W}) = \sec\theta$. Thus, when the pitch angle satisfies $\theta \neq (2k-1)\pi/2$, $(k \in \mathcal{Z})$ the matrix $\boldsymbol{W}$ is invertible.

3 Trajectory tracking control design methodology

In consideration of the under-actuated and strong coupled properties of a quadrotor aircraft, this paper builds the relationship between the attitude and the linear acceleration based on the position error PD closed-loop equation of the quadrotor, and then proposes a control design algorithm that employs the command-filtered backstepping technique to track the attitude commanded signal produced by the position controller and uses a linear tracking differentiator to extract the attitude command derivative signals required by the backstepping control law without introducing inner/outer-loop structure, as shown in Fig. 2.

Note 1: The outer loop in this paper refers to the position control loop while the inner one refers to the attitude control loop. Therefore the linear tracking differentiator is applied here to eliminate the time scale separation assumption between attitude and linear dynamics which usually requires the significant response in the DC motor.

3.1 Position control design

Define the position error as

$$\boldsymbol{p}_{\mathrm{e}} = \boldsymbol{p}_{\mathrm{c}} - \boldsymbol{p}$$

where the vector $\boldsymbol{p}_{\mathrm{c}} = (x_{\mathrm{c}}, y_{\mathrm{c}}, z_{\mathrm{c}})^{\mathrm{T}}$ is the position-commanded vector. Then construct the position closed-loop equation as

$$\ddot{\boldsymbol{p}}_{\mathrm{e}} + \boldsymbol{K}_d \dot{\boldsymbol{p}}_{\mathrm{e}} + \boldsymbol{K}_p \boldsymbol{p}_{\mathrm{e}} = 0 \tag{12}$$

where $\boldsymbol{K}_d$ and $\boldsymbol{K}_p$ are two positive definite matrices. According to the Routh–Hurwitz criterion, the position error $\boldsymbol{p}_{\mathrm{e}}$ converges to zero exponentially. Continuously, (12) can be rewritten as

$$\ddot{\boldsymbol{p}} = \ddot{\boldsymbol{p}}_{\mathrm{c}} + \boldsymbol{K}_d(\dot{\boldsymbol{p}}_{\mathrm{c}} - \dot{\boldsymbol{p}}) + \boldsymbol{K}_p(\boldsymbol{p}_{\mathrm{c}} - \boldsymbol{p}) \tag{13}$$

Define the virtual control vector as $\boldsymbol{U} = \ddot{\boldsymbol{p}} = (U_1, U_2, U_3)^{\mathrm{T}}$. Replacing $\dot{\boldsymbol{v}}$ with $\boldsymbol{U}$ in (2) yields

$$\boldsymbol{U} = -g\boldsymbol{z}_e + \frac{1}{m} T \boldsymbol{R} \boldsymbol{z}_{\mathrm{e}}$$

Move the gravity acceleration item $-g\boldsymbol{z}_{\mathrm{e}}$ to the left-hand side and left multiply rotation matrix $\boldsymbol{R}^{\mathrm{T}}$ on both sides of the preceding equation. Then we obtain

$$\boldsymbol{R}^{\mathrm{T}}(\boldsymbol{U} + g\boldsymbol{z}_{\mathrm{e}}) = \frac{1}{m} T \boldsymbol{z}_{\mathrm{e}} \tag{14}$$

That is

$$\begin{pmatrix} c_\theta c_\psi & c_\theta s_\psi & -s_\theta \\ s_\theta c_\psi s_\phi - s_\psi c_\phi & s_\theta s_\psi s_\phi + c_\psi c_\phi & c_\theta s_\phi \\ s_\theta c_\psi c_\phi + s_\psi s_\phi & s_\theta s_\psi c_\phi - c_\psi s_\phi & c_\theta c_\phi \end{pmatrix} \begin{pmatrix} U_1 \\ U_2 \\ U_3 + g \end{pmatrix} = \frac{1}{m} \begin{pmatrix} 0 \\ 0 \\ T \end{pmatrix}$$

where $s_{(\cdot)} \triangleq \sin(\cdot)$, $c_{(\cdot)} \triangleq \cos(\cdot)$. After simple algebraic computation, the following equations are acquired

$$U_1 \cos\theta\cos\psi + U_2 \cos\theta \sin\psi - (U_3 + g)\sin\theta = 0 \tag{15}$$

$$\begin{aligned} &U_1(\sin\theta\cos\psi\sin\phi - \sin\psi\cos\phi) \\ &\quad + U_2(\sin\theta\sin\psi\sin\phi + \cos\psi\cos\phi) \\ &\quad + (U_3 + g)\cos\theta\sin\phi = 0 \end{aligned} \tag{16}$$

$$\begin{aligned} &U_1(\sin\theta\cos\psi\cos\phi + \sin\psi\sin\phi) \\ &\quad + U_2(\sin\theta\sin\psi\cos\phi - \cos\psi\sin\phi) \\ &\quad + (U_3 + g)\cos\theta\cos\phi = \frac{1}{m} T \end{aligned} \tag{17}$$

In virtue of the fact that $\cos\theta$ is non-zero, divide both sides of (15) by $\cos\theta$ and then the pitch attitude can be computed as

$$\theta = \arctan\left(\frac{U_1\cos\psi + U_2\sin\psi}{U_3 + g}\right) \tag{18}$$

Equation (17) $\times$ $\sin\phi$−Equation (16) $\times$ $\cos\phi$ yields

$$\frac{1}{m} T \sin\phi = U_1 \sin\psi - U_2 \cos\psi \tag{19}$$

Employing (14), we have the following relationship

$$\left(\frac{1}{m} T \boldsymbol{z}_{\mathrm{e}}\right)^{\mathrm{T}} \left(\frac{1}{m} T \boldsymbol{z}_{\mathrm{e}}\right) = (\boldsymbol{U} + g\boldsymbol{z}_{\mathrm{e}})^{\mathrm{T}} (\boldsymbol{U} + g\boldsymbol{z}_{\mathrm{e}})$$

which is equivalent to

$$U_1^2 + U_2^2 + (U_3 + g)^2 = \left(\frac{1}{m} T\right)^2 \tag{20}$$

Combining (20) with (19) yields the roll attitude expression:

$$\phi = \arcsin\left(\frac{U_1\sin\psi - U_2\cos\psi}{\sqrt{U_1^2 + U_2^2 + (U_3 + g)^2}}\right) \tag{21}$$

To this end, (18) and (21) can be used to compute the attitude-commanded inputs to the inner loop, and the virtual control signal can be computed from (13). Hence, the closed-form expressions for pitch and roll attitudes are

$$\begin{cases} \theta_c = \arctan\left(\dfrac{U_1\cos\psi_c + U_2\sin\psi_c}{U_3 + g}\right) \\ \phi_c = \arcsin\left(\dfrac{U_1\sin\psi_c - U_2\cos\psi_c}{\sqrt{U_1^2 + U_2^2 + (U_3 + g)^2}}\right) \end{cases}$$

where ψ_c is the commanded yaw attitude. Besides, the required total thrust generated by the four rotors can be

IET Control Theory Appl., 2010, Vol. 4, Iss. 11, pp. 2343–2355
doi: 10.1049/iet-cta.2009.0336

calculated from (17)

$$T = m[U_1(\sin\theta\cos\psi\cos\phi + \sin\psi\sin\phi) + U_2(\sin\theta\sin\psi\cos\phi - \cos\psi\sin\phi) + (U_3 + g)\cos\theta\cos\phi]$$

3.2 Attitude control design

Define the following attitude and angular velocity errors as

$$\boldsymbol{Z}_1 = \boldsymbol{\Theta} - \boldsymbol{\Theta}_c \tag{22}$$

$$\boldsymbol{Z}_2 = \boldsymbol{\Omega} - \boldsymbol{\Omega}_c \tag{23}$$

where $\boldsymbol{\Theta}_c$ and $\boldsymbol{\Omega}_c$ are commanded attitude and angular velocity, respectively. Let us consider the candidate Lyapunov function

$$V_1 = \frac{1}{2}\boldsymbol{Z}_1^{\mathrm{T}}\boldsymbol{Z}_1 \tag{24}$$

The derivative of the Lyapunov function along the trajectory of (11) is given by

$$\dot{V}_1 = \boldsymbol{Z}_1^{\mathrm{T}}\dot{\boldsymbol{Z}}_1 = \boldsymbol{Z}_1^{\mathrm{T}}(\boldsymbol{W}\boldsymbol{\Omega} - \dot{\boldsymbol{\Theta}}_c)$$

If $\theta \neq (2k-1)\pi/2, (k \in \mathcal{Z})$, the angular velocity $\boldsymbol{\Omega}$ can be seen as virtual control signal and extracted to satisfy $\dot{V}_1 < 0$ as

$$\boldsymbol{\Omega}_d = \boldsymbol{W}^{-1}(\dot{\boldsymbol{\Theta}}_c - \boldsymbol{\Gamma}_1\boldsymbol{Z}_1) \tag{25}$$

where $\boldsymbol{\Omega}_d$ is the desired angular velocity and $\boldsymbol{\Gamma}_1$ is a positive definite matrix. In order to avoid tediously taking time derivative of $\boldsymbol{\Omega}_d$ and alleviate the dependence on analytic model, the first-order command filter

$$\dot{\boldsymbol{\Omega}}_c = -\bar{\boldsymbol{T}}(\boldsymbol{\Omega}_c - \boldsymbol{\Omega}_d) \tag{26}$$

is employed to track the desired angular velocity $\boldsymbol{\Omega}_d$. The filter time constant matrix $\bar{\boldsymbol{T}} = \mathrm{diag}\{t_1, t_2, t_3\} > 0$ should be as large as possible to promise the fast tracking. In consideration of the influence caused by the command filter, a new vector ε is introduced to compensate the tracking error in the rigorous stability analysis

$$\dot{\varepsilon} = -\boldsymbol{\Gamma}_1\varepsilon + \boldsymbol{W}(\boldsymbol{\Omega}_c - \boldsymbol{\Omega}_d) \tag{27}$$

So the attitude tracking error is redefined as

$$\bar{\boldsymbol{Z}}_1 = \boldsymbol{Z}_1 - \varepsilon = \boldsymbol{\Theta} - \boldsymbol{\Theta}_c - \varepsilon \tag{28}$$

Choose the candidate compound Lyapunov function as

$$V_2 = \frac{1}{2}\bar{\boldsymbol{Z}}_1^{\mathrm{T}}\bar{\boldsymbol{Z}}_1 + \frac{1}{2}\boldsymbol{Z}_2^{\mathrm{T}}\boldsymbol{Z}_2$$

Similarly, taking the time derivative of V_2 along the vector field of (11) and (4) yields

$$\begin{aligned}\dot{V}_2 &= \bar{\boldsymbol{Z}}_1^{\mathrm{T}}\dot{\bar{\boldsymbol{Z}}}_1 + \frac{1}{2}\boldsymbol{Z}_2^{\mathrm{T}}\dot{\boldsymbol{Z}}_2 \\ &= \bar{\boldsymbol{Z}}_1^{\mathrm{T}}[\boldsymbol{W}\boldsymbol{\Omega} - \dot{\boldsymbol{\Theta}}_c + \boldsymbol{\Gamma}_1\varepsilon - \boldsymbol{W}(\boldsymbol{\Omega}_c - \boldsymbol{\Omega}_d)] \\ &\quad + \boldsymbol{Z}_2^{\mathrm{T}}[-\boldsymbol{I}_f^{-1}(\boldsymbol{\Omega}\times\boldsymbol{I}_f\boldsymbol{\Omega}) - \boldsymbol{I}_f^{-1}\boldsymbol{G}_a + \boldsymbol{I}_f^{-1}\boldsymbol{\tau}_a - \dot{\boldsymbol{\Omega}}_c] \\ &= \bar{\boldsymbol{Z}}_1^{\mathrm{T}}(-\boldsymbol{\Gamma}_1\bar{\boldsymbol{Z}}_1 + \boldsymbol{W}\boldsymbol{Z}_2) \\ &\quad + \boldsymbol{Z}_2^{\mathrm{T}}[-\boldsymbol{I}_f^{-1}(\boldsymbol{\Omega}\times\boldsymbol{I}_f\boldsymbol{\Omega}) - \boldsymbol{I}_f^{-1}\boldsymbol{G}_a + \boldsymbol{I}_f^{-1}\boldsymbol{\tau}_a - \dot{\boldsymbol{\Omega}}_c] \\ &= -\bar{\boldsymbol{Z}}_1^{\mathrm{T}}\boldsymbol{\Gamma}_1\bar{\boldsymbol{Z}}_1 \\ &\quad + \boldsymbol{Z}_2^{\mathrm{T}}[-\boldsymbol{I}_f^{-1}(\boldsymbol{\Omega}\times\boldsymbol{I}_f\boldsymbol{\Omega}) - \boldsymbol{I}_f^{-1}\boldsymbol{G}_a + \boldsymbol{I}_f^{-1}\boldsymbol{\tau}_a - \dot{\boldsymbol{\Omega}}_c + \boldsymbol{W}^{\mathrm{T}}\bar{\boldsymbol{Z}}_1]\end{aligned}$$

If the control torque $\boldsymbol{\tau}_a$ is extracted as

$$\begin{aligned}\boldsymbol{\tau}_a &= (\boldsymbol{\Omega}\times\boldsymbol{I}_f\boldsymbol{\Omega}) + \boldsymbol{G}_a + \boldsymbol{I}_f\dot{\boldsymbol{\Omega}}_c - \boldsymbol{I}_f\boldsymbol{W}^{\mathrm{T}}\bar{\boldsymbol{Z}}_1 - \boldsymbol{I}_f\boldsymbol{\Gamma}_2\boldsymbol{Z}_2 \\ &= (\boldsymbol{\Omega}\times\boldsymbol{I}_f\boldsymbol{\Omega}) + \boldsymbol{G}_a - \boldsymbol{I}_f\bar{\boldsymbol{T}}\boldsymbol{\Omega}_c + \boldsymbol{I}_f\bar{\boldsymbol{T}}\boldsymbol{W}^{-1}\dot{\boldsymbol{\Theta}}_c \\ &\quad - \boldsymbol{I}_f\bar{\boldsymbol{T}}\boldsymbol{W}^{-1}\boldsymbol{\Gamma}_1(\boldsymbol{\Theta} - \boldsymbol{\Theta}_c) \\ &\quad - \boldsymbol{I}_f\boldsymbol{W}^{\mathrm{T}}(\boldsymbol{\Theta} - \boldsymbol{\Theta}_c - \varepsilon) - \boldsymbol{I}_f\boldsymbol{\Gamma}_2(\boldsymbol{\Omega} - \boldsymbol{\Omega}_c)\end{aligned} \tag{29}$$

the time derivative of Lyapunov function V_2 satisfies $\dot{V}_2 = -\bar{\boldsymbol{Z}}_1^{\mathrm{T}}\boldsymbol{\Gamma}_1\bar{\boldsymbol{Z}}_1 - \boldsymbol{Z}_2^{\mathrm{T}}\boldsymbol{\Gamma}_2\boldsymbol{Z}_2 < 0(\boldsymbol{\Gamma}_2 > 0)$. Thus, the tracking errors $\bar{\boldsymbol{Z}}_1$ and $\boldsymbol{Z}_2$ converge to zero exponentially. Combining (26) and (27) with singular perturbation theory [12], the attitude $\boldsymbol{\Theta}$ converges to $\boldsymbol{\Theta}_c$ ultimately. However, there exists a commanded attitude derivative signal in the explicit expression of $\boldsymbol{\tau}_a$. Although the commanded attitude $\boldsymbol{\Theta}_c$ is known, the analytic computation of derivative expression is quite boring and intricate. And thus a linear tracking differentiator is adopted to obtain the derivative signal $\dot{\boldsymbol{\Theta}}_c$

$$\begin{cases}\dot{\boldsymbol{X}}_1 = \boldsymbol{X}_2 \\ \dot{\boldsymbol{X}}_2 = -2\boldsymbol{\Lambda}\boldsymbol{X}_1 - \boldsymbol{\Lambda}^2(\boldsymbol{X}_1 - \boldsymbol{\Theta}_c)\end{cases} \tag{30}$$

where $\boldsymbol{\Lambda} = \mathrm{diag}\{\gamma_1, \gamma_2, \gamma_3\} > 0$. Similar to previous first-order filter, $\gamma_i (i = 1, 2, 3)$ shall be large enough to guarantee the fast tracking to commanded attitude derivative signal. Thus, $\dot{\boldsymbol{\Theta}}_c$ can be replaced with $\boldsymbol{X}_2$ accordingly

$$\begin{aligned}\boldsymbol{\tau}_a &= (\boldsymbol{\Omega}\times\boldsymbol{I}_f\boldsymbol{\Omega}) + \boldsymbol{G}_a - \boldsymbol{I}_f\bar{\boldsymbol{T}}\boldsymbol{\Omega}_c \\ &\quad + \boldsymbol{I}_f\bar{\boldsymbol{T}}\boldsymbol{W}^{-1}\boldsymbol{X}_2 - \boldsymbol{I}_f\bar{\boldsymbol{T}}\boldsymbol{W}^{-1}\boldsymbol{\Gamma}_1(\boldsymbol{\Theta} - \boldsymbol{\Theta}_c) \\ &\quad - \boldsymbol{I}_f\boldsymbol{W}^{\mathrm{T}}(\boldsymbol{\Theta} - \boldsymbol{\Theta}_c - \varepsilon) - \boldsymbol{I}_f\boldsymbol{\Gamma}_2(\boldsymbol{\Omega} - \boldsymbol{\Omega}_c)\end{aligned} \tag{31}$$

The block diagram of the proposed attitude stabilisation controller is depicted as Fig. 3.

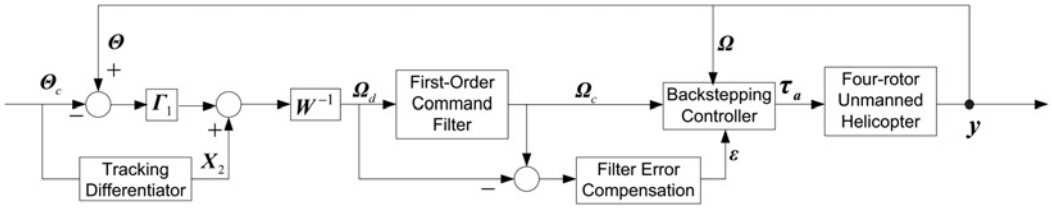

Figure 3 *Backstepping attitude control block diagram*

4 Control algorithm

Step 1: Trajectory planning: given the desired trajectory

$$\boldsymbol{p}_{\mathrm{c}}(\alpha(t)) = (x_{\mathrm{c}}(\alpha(t)), y_{\mathrm{c}}(\alpha(t)), z_{\mathrm{c}}(\alpha(t))) \tag{32}$$

and the desired yaw attitude $\psi_{\mathrm{c}}(\alpha(t))$, where $\alpha(t)$ is a function of time, and usually $\alpha(t) = t$.

Step 2: Calculate the virtual control vector $\boldsymbol{U}$

$$\begin{pmatrix} U_1 \\ U_2 \\ U_3 \end{pmatrix} = \begin{pmatrix} \ddot{x}_{\mathrm{c}} \\ \ddot{y}_{\mathrm{c}} \\ \ddot{z}_{\mathrm{c}} \end{pmatrix} + \begin{pmatrix} k_{d1} & 0 & 0 \\ 0 & k_{d2} & 0 \\ 0 & 0 & k_{d3} \end{pmatrix} \times \begin{pmatrix} \dot{x}_{\mathrm{c}} - \dot{x} \\ \dot{y}_{\mathrm{c}} - \dot{y} \\ \dot{z}_{\mathrm{c}} - \dot{z} \end{pmatrix} + \begin{pmatrix} k_{p1} & 0 & 0 \\ 0 & k_{p2} & 0 \\ 0 & 0 & k_{p3} \end{pmatrix} \begin{pmatrix} x_{\mathrm{c}} - x \\ y_{\mathrm{c}} - y \\ y_{\mathrm{c}} - y \end{pmatrix} \tag{33}$$

Step 3: Calculate the commanded pitch and roll attitudes

$$\begin{cases} \theta_{\mathrm{c}} = \arctan\left(\dfrac{U_1 \cos\psi_{\mathrm{c}} + U_2 \sin\psi_{\mathrm{c}}}{U_3 + g}\right) \\ \phi_{\mathrm{c}} = \arcsin\left(\dfrac{U_1 \sin\psi_{\mathrm{c}} - U_2 \cos\psi_{\mathrm{c}}}{\sqrt{U_1^2 + U_2^2 + (U_3 + g)^2}}\right) \end{cases} \tag{34}$$

Step 4: Calculate the commanded angular velocity $\boldsymbol{\Omega}_{\mathrm{c}}$ and the filter error $\boldsymbol{\varepsilon}$

$$\boldsymbol{\Omega}_{\mathrm{c}} = -\bar{\boldsymbol{T}} \int (\boldsymbol{\Omega}_{\mathrm{c}} - \boldsymbol{\Omega}_{\mathrm{d}}) \tag{35}$$

$$\boldsymbol{\varepsilon} = -\boldsymbol{\Gamma}_1 \int \boldsymbol{\varepsilon} + \boldsymbol{W} \int (\boldsymbol{\Omega}_{\mathrm{c}} - \boldsymbol{\Omega}_{\mathrm{d}}) \tag{36}$$

Step 5: Extract the commanded attitude derivative signal $\dot{\boldsymbol{\Theta}}_{\mathrm{c}}$

$$\begin{cases} \dot{\boldsymbol{X}}_1 = \boldsymbol{X}_2 \\ \dot{\boldsymbol{X}}_2 = -2\boldsymbol{\Lambda}\boldsymbol{X}_1 - \boldsymbol{\Lambda}^2(\boldsymbol{X}_1 - \boldsymbol{\Theta}_{\mathrm{c}}) \end{cases} \tag{37}$$
$$\dot{\boldsymbol{\Theta}}_{\mathrm{c}} = \boldsymbol{X}_2$$

Step 6: Calculate the control inputs T and $\boldsymbol{\tau}_a$

$$\begin{aligned} T = m[&U_1(\sin\theta\cos\psi\cos\phi + \sin\psi\sin\phi) \\ &+ U_2(\sin\theta\sin\psi\cos\phi - \cos\psi\sin\phi) \\ &+ (U_3 + g)\cos\theta\cos\phi] \end{aligned} \tag{38}$$

$$\begin{aligned} \boldsymbol{\tau}_a = &(\boldsymbol{\Omega} \times \boldsymbol{I}_f\boldsymbol{\Omega}) + \boldsymbol{G}_a - \boldsymbol{I}_f\bar{\boldsymbol{T}}\boldsymbol{\Omega}_{\mathrm{c}} + \boldsymbol{I}_f\bar{\boldsymbol{T}}\boldsymbol{W}^{-1}\boldsymbol{X}_2 \\ &- \boldsymbol{I}_f\bar{\boldsymbol{T}}\boldsymbol{W}^{-1}\boldsymbol{\Gamma}_1(\boldsymbol{\Theta} - \boldsymbol{\Theta}_{\mathrm{c}}) \\ &- \boldsymbol{I}_f\boldsymbol{W}^{\mathrm{T}}(\boldsymbol{\Theta} - \boldsymbol{\Theta}_{\mathrm{c}} - \boldsymbol{\varepsilon}) - \boldsymbol{I}_f\boldsymbol{\Gamma}_2(\boldsymbol{\Omega} - \boldsymbol{\Omega}_{\mathrm{c}}) \end{aligned} \tag{39}$$

Remark 2: The parameter $\alpha(t)$ in (32) is a generic function of time and used to define the desired trajectory with the most appropriate scale of time, such as $r(t) = at$. The selection of this scalar a usually depends on the control energy and the external disturbances.

Remark 3: The time scale separation is eliminated and actually it is required that the design parameters $\boldsymbol{K}_d$, $\boldsymbol{K}_p$ and $\boldsymbol{\Gamma}_i (i = 1, 2)$ to be positive definite in theory. Hence, the design parameters $\boldsymbol{K}_i (i = p, d)$ in the position controller and $\boldsymbol{\Gamma}_i (i = 1, 2)$ in the attitude controller can be selected independently to satisfy their own bandwidths, respectively.

Remark 4: The command filter (35) and the linear tracking differentiator (37) can be also interpreted as arranging the transition of tracking derivative signal of attitude and angular velocity command. The selection of parameter $\bar{\boldsymbol{T}}$ and $\boldsymbol{\Lambda}$ should be a compromise between the control energy and the tracking speed, that is, the larger $\bar{\boldsymbol{T}}$ and $\boldsymbol{\Lambda}$ leads to faster tracking but larger control energy. If the combination of these design parameters still cannot satisfy the quadrotor physical constraints like magnitude, rate and bandwidth, one way is to integrate the constraint into the tracking filter to guarantee the whole stability (cf. [17]).

5 Numerical simulation

In order to verify the validity and efficiency of the control algorithm proposed in this paper, simulations of typical trajectory tracking under certain conditions are performed on Matlab/Simulink using data taken from [3], as listed in Table 2.

doi: 10.1049/iet-cta.2009.0336

Table 2 Quadrotor unmanned mini-helicopter model parameters

Parameter	m	g	l	I_r	I_x	I_y	I_z	b	k
value	0.468	9.81	0.225	3.357×10^{-5}	4.856×10^{-3}	4.856×10^{-3}	8.801×10^{-3}	2.98×10^{-6}	1.14×10^{-7}
units	kg	m/s^2	m	kg m^2	kg m^2	kg m^2	kg m^2	N s^2/rad^2	Nm s^2/rad^2

5.1 Comparison with a two-time scale-based controller

For simplicity the dynamics of DC motor is ignored temporarily at first, because the commercial DC motor usually has its own motion controller with excellent performance. The controller parameters in simulations are fixed at $\boldsymbol{K}_d = \text{diag}\{2, 2, 2\}$, $\boldsymbol{K}_p = \text{diag}\{1, 1, 1\}$, $\boldsymbol{\Gamma}_1 = \text{diag}\{0.1, 0.1, 0.1\}$ and $\boldsymbol{\Gamma}_2 = \text{diag}\{0.1, 0.1, 0.1\}$. The filter time constant matrices are chosen as $\bar{\boldsymbol{T}} = \text{diag}\{15, 15, 15\}$ and $\boldsymbol{\Lambda} = \text{diag}\{10, 10, 10\}$, respectively. In view of the limitations of real measuring device, the sampling time is fixed to $\Delta t = 0.01$ s. The initial positions and Euler angles are $\boldsymbol{p}_0 = (0, 0, 0)^{\mathrm{T}}$, $\boldsymbol{\Theta}_0 = (0, 0, 0)^{\mathrm{T}}$, respectively, and so are linear and angular velocities, respectively. The yaw angle command is fixed at zero and the considered desired trajectories are a vertical and a horizontal rectangle, given by

$$\begin{cases} x_c = t\text{fsg}(t, 0, 20) + 20\text{fsg}(t, 20, 40) + (60 - t)\text{fsg} \\ \qquad \times(t, 40, 60) \\ y_c = 0 \\ z_c = 5\text{fsg}(t, 20, 60) \end{cases} \tag{40}$$

and

$$\begin{cases} x_c = t\text{fsg}(t, 0, 20) + 20\text{fsg}(t, 20, 40) + (60 - t) \\ \qquad \times\text{fsg}(t, 40, 60) \\ y_c = (t - 20)\text{fsg}(t, 20, 40) + 20\text{fsg}(t, 40, 60) \\ z_c = 0 \end{cases} \tag{41}$$

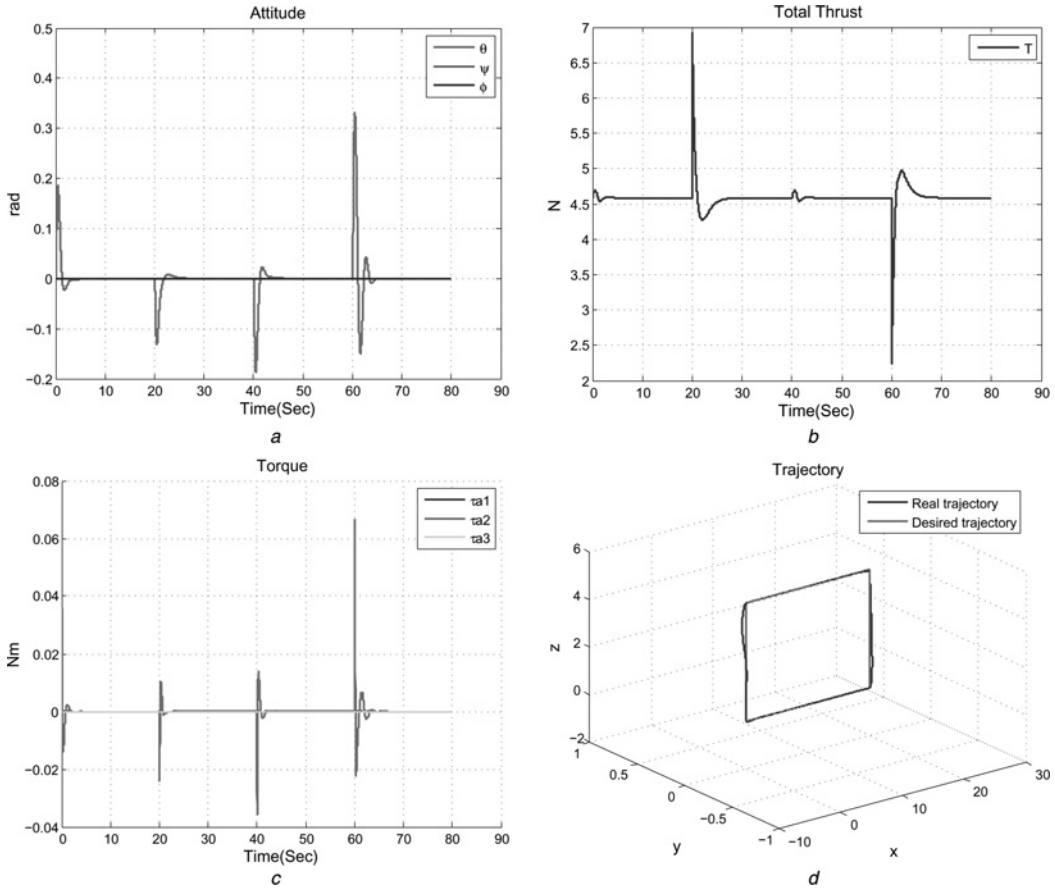

Figure 4 *Vertical rectangle flight*

a Euler angles
b Control thrust
c Control torques
d Flight trajectory

doi: 10.1049/iet-cta.2009.0336

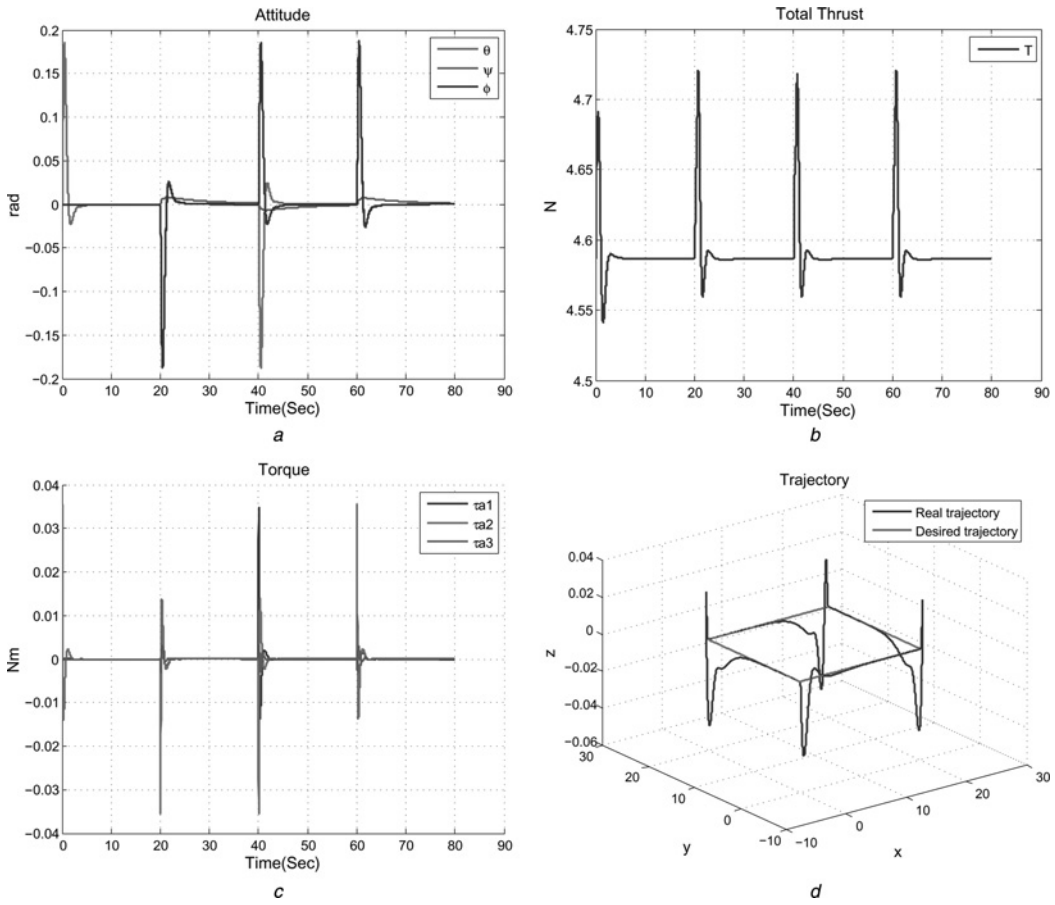

Figure 5 *Horizontal rectangle flight*

a Euler angles
b Control thrust
c Control torques
d Flight trajectory

where fsg is an interval function and is expressed as

$$\text{fsg} = \frac{\text{sign}(x-a) - \text{sign}(x-b)}{2}$$

Simulation results are presented in Figs. 4 and 5 which depict the time histories of Euler angles, control inputs and flight trajectories. As expected, the design methodology makes the quadrotor track the desired trajectories in a satisfactory

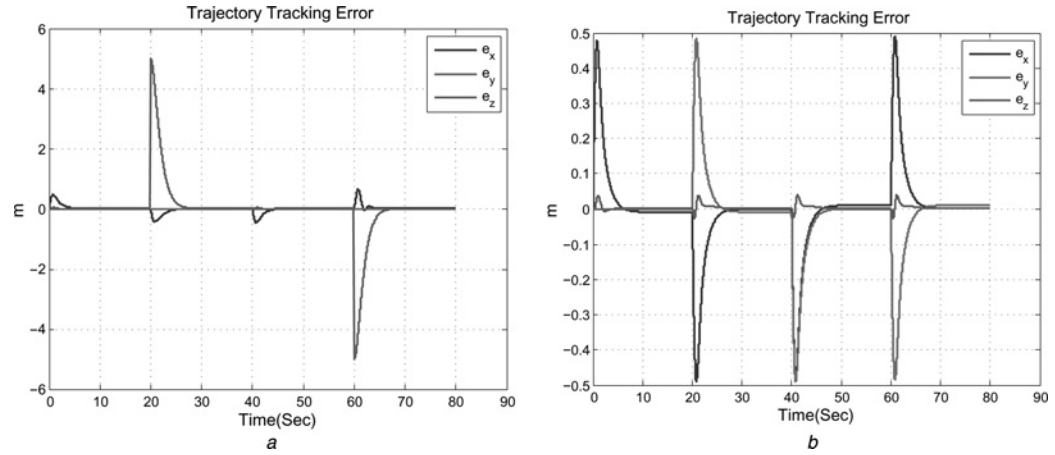

Figure 6 *Trajectory tracking error*

a Tracking error of vertical flight
b Tracking error of horizontal flight

doi: 10.1049/iet-cta.2009.0336

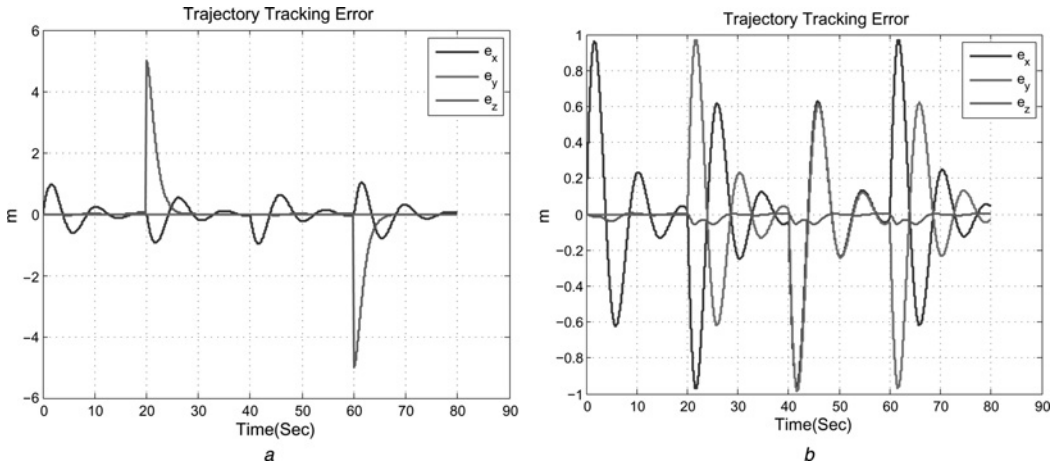

Figure 7 *Trajectory tracking error without compensation of the commanded attitude dynamics*

a Tracking error of vertical flight
b Tracking error of horizontal flight

way, seeing Fig. 6. But in order to make a comparison with the inner/outer-loop control structure, it is assumed that the dynamics $\dot{\boldsymbol{\Theta}}_c = 0$ in (29) within the small-enough sampling time (that is two-time scale separation assumption), and therefore the linear tracking differentiator (37) can be removed. Fig. 7 depicts the time response of tracking errors $(x_c - x, y_c - y, z_c - z)$ in the case that the controller parameters remain unchanged. As indicated, the

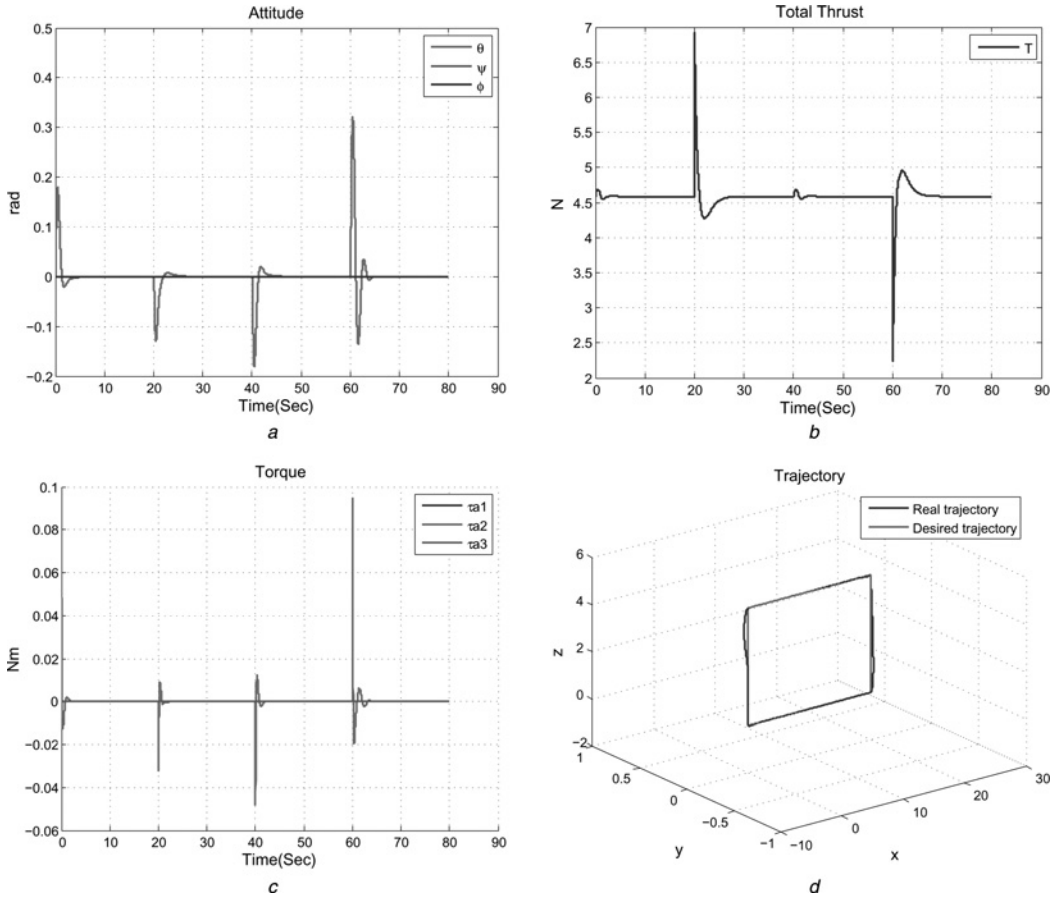

Figure 8 *Vertical rectangle flight without compensation of the commanded attitude dynamics*

a Euler angles
b Control thrust
c Control torques
d Flight trajectory

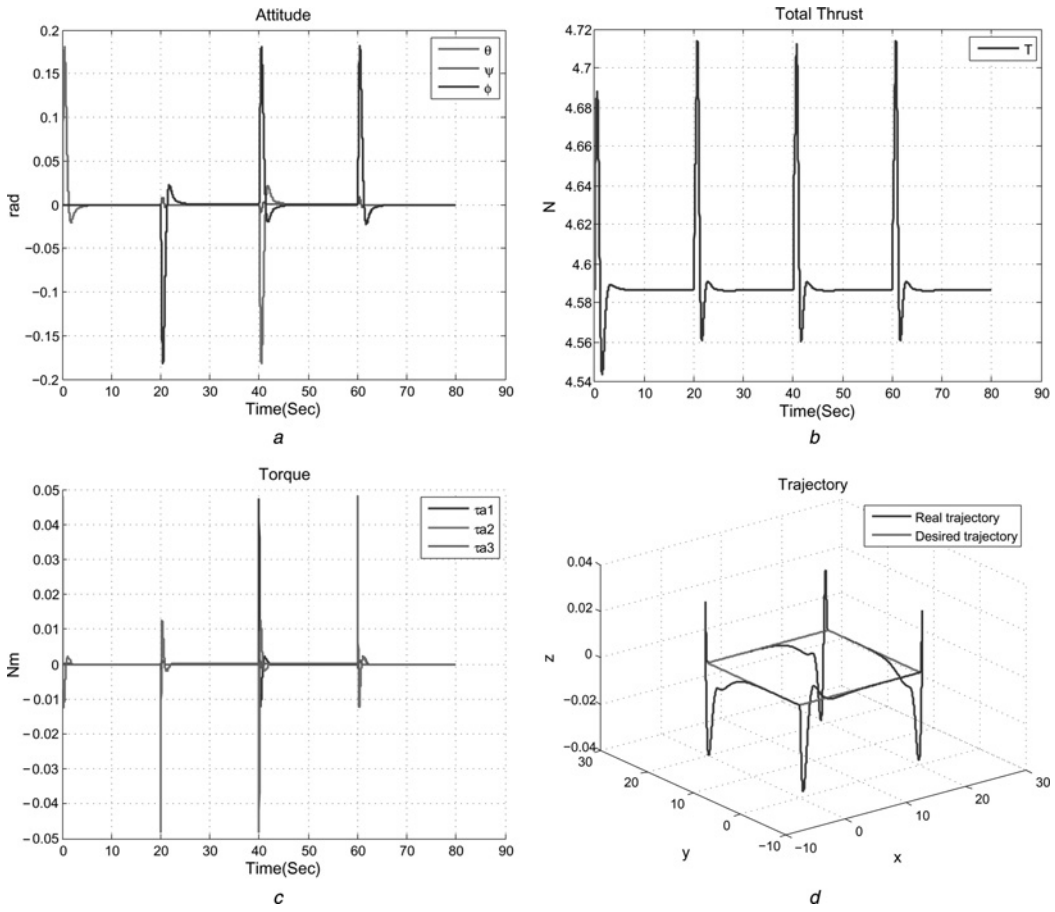

Figure 9 *Horizontal Rectangle Flight without compensation of the commanded attitude dynamics*

a Euler angles
b Control thrust
c Control torques
d Flight Trajectory

system tracking errors become large or even unstable. In order to guarantee the system stability, the difference of time scale between the outer and inner loop should be separated obviously by selecting the larger inner loop gain matrix, for example, $\mathbf{\Gamma}_1 = \text{diag}\{3, 3, 3\}$. The simulation results of tracking the same trajectories (40) and (41) are presented in Figs. 8 and 9. Although Fig. 10 presents the similar tracking performance with that of the algorithm without

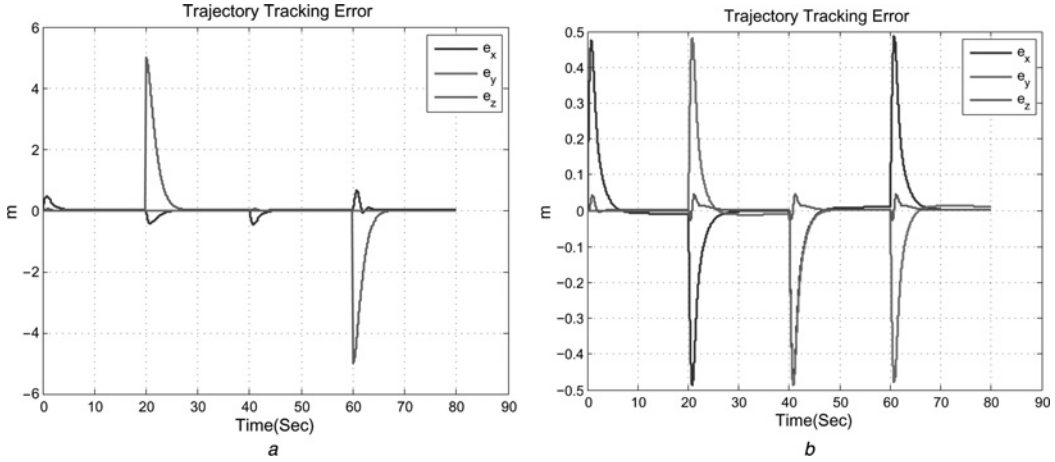

Figure 10 *Trajectory Tracking Error without compensation of the commanded attitude dynamics*

a Tracking error of vertical flight
b Tracking error of horizontal flight

doi: 10.1049/iet-cta.2009.0336

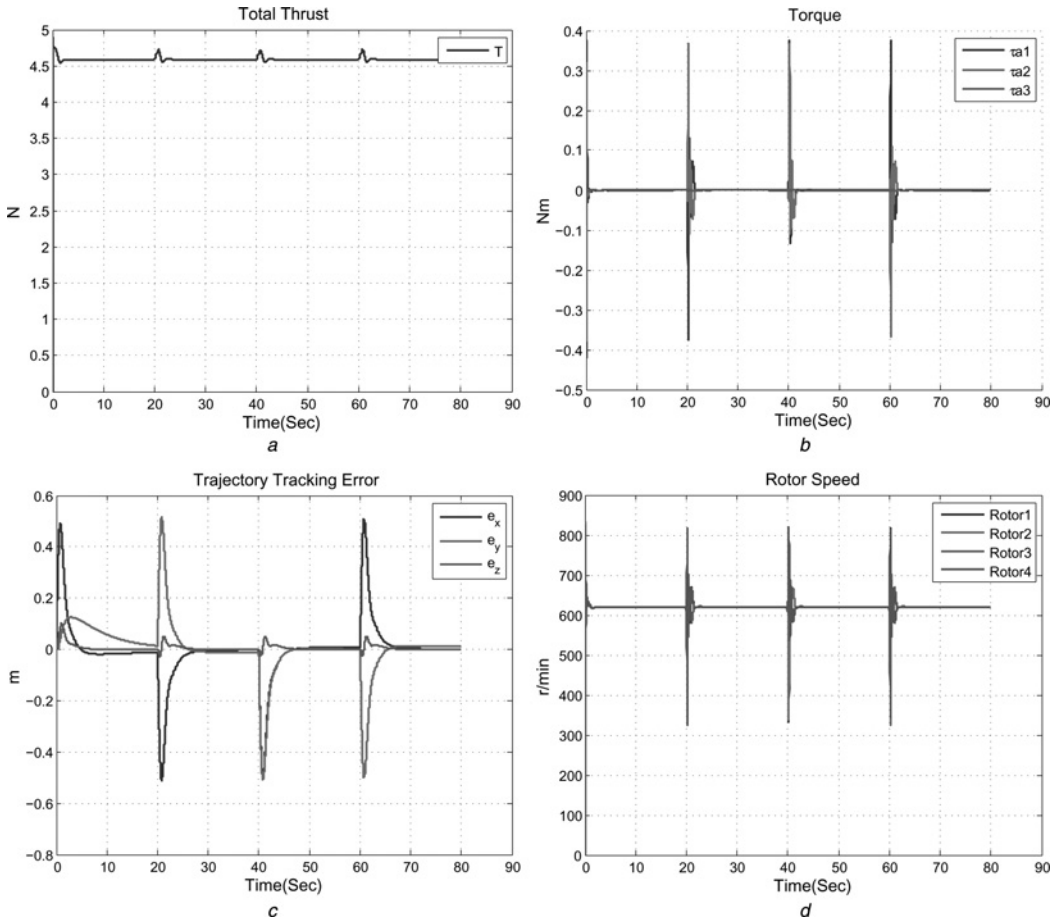

Figure 11 *Horizontal rectangle flight with acuator dynamics*

a Real control thrust
b Real control torque
c Tracking error
d Real rotor speed

time scale separation designed in this paper, the control energy consumption, especially the control torque, calculated by the latter control scheme with time scale separation is obviously larger.

5.2 Actuator dynamics

This subsection explores the performance under actuator dynamics which is given by (6). According to (10), the matrix $\boldsymbol{M}$ is non-singular for $bkl \neq 0$ and therefore the square of the desired rotor speed can be calculated as $\bar{\boldsymbol{\omega}}_{\mathrm{d}} = \boldsymbol{M}^{-1}(T, \boldsymbol{\tau}_a)^{\mathrm{T}}$. Let $\bar{\boldsymbol{\omega}} = (\bar{\omega}_1, \bar{\omega}_2, \bar{\omega}_3, \bar{\omega}_4)^{\mathrm{T}}$ denote the square of the real rotor speed. The actuator dynamic expression has been given in [2, 3], and here the analytical expression of the rotor speed is directly written as follows:

$$\bar{\omega}_i = \bar{\omega}_{i,d} + \exp\left(-\frac{k_i}{I_r}t\right)(\bar{\omega}_{i0} - \bar{\omega}_{i,d}) \tag{42}$$

where $k_i > 0 (i = 1, 2, 3, 4)$ is the control parameter of DC motor i and $\bar{\omega}_{i0}$ is the square of the initial rotor speed. Thus, the real control input is $(T, \boldsymbol{\tau}_a)^{\mathrm{T}} = \boldsymbol{M}\bar{\boldsymbol{\omega}}$. Intuitively, the tracking stability can be guaranteed as long as $\exp((-k_i/I_r)\Delta t)$ converges to zero within every sampling time. In order to investigate the controller performance under the actuator dynamics, the parameter k_i is fixed at 0.002 to lower the response of the motors on purpose. Accordingly, the gain matrix of the command filter is selected as $\bar{\boldsymbol{T}} = \mathrm{diag}\{10, 10, 10\}$ to slacken the transition of tracking the angular velocity command, while the other control parameters remain the same. Fig. 11 depicts the time histories of real control inputs, tracking error as well as real rotor speeds. The quadrotor is stabilised to the desired trajectory by the controller, although the tracking error is obvious at the first segment of the horizontal square trajectory because of the start-up process of the motor and the control torque is larger than that in Fig. 5. Moreover, the total control thrust is much more smooth because of the low-pass property of the DC motor.

5.3 Robustness test

The quadrotor modelling in Section II ignores the aerodynamic effects. Although this assumption is reasonable at low speed, it is still necessary to investigate

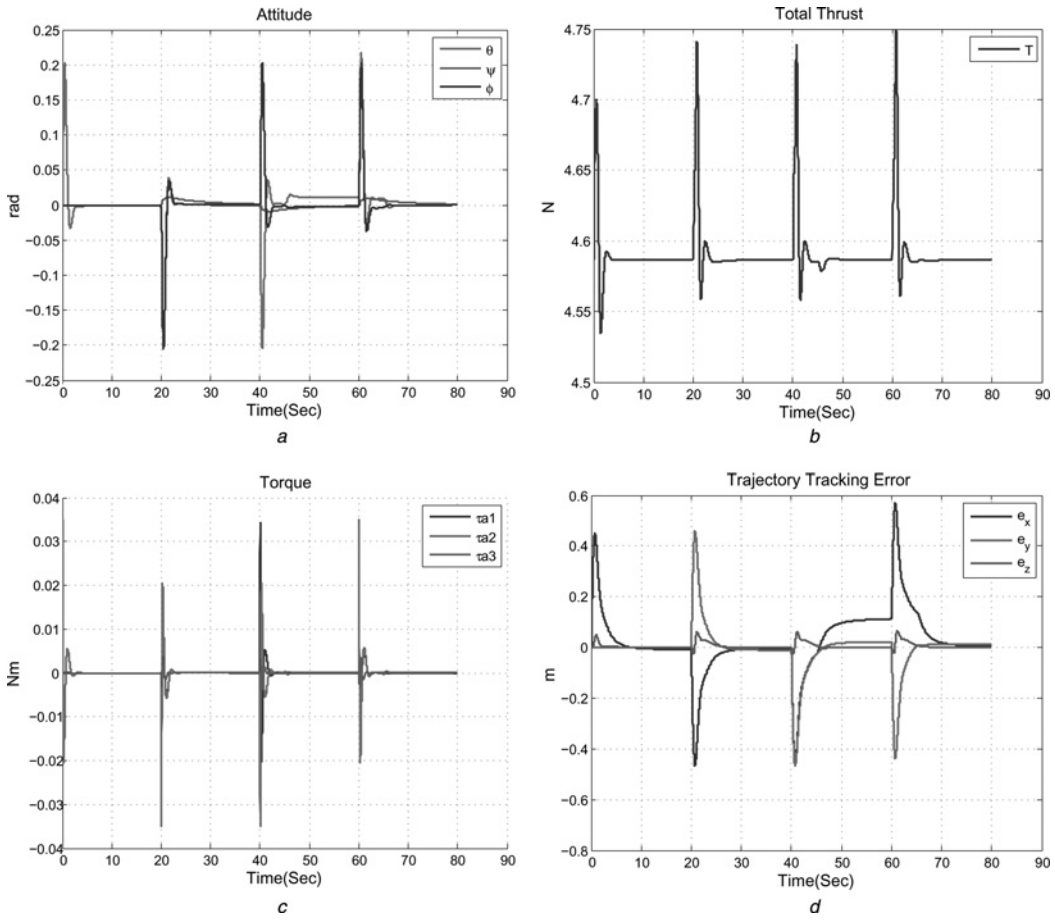

Figure 12 *Vertical rectangle flight with modelling error and external disturbance*

a Euler angles
b Control thrust
c Control torques
d Flight trajectory

whether the control system is robust to the influence of the aerodynamics. Therefore aerodynamic friction [10] is integrated into the simulation plant model

$$\begin{cases} \boldsymbol{F}_{\text{aero}} = \boldsymbol{K}_t(\boldsymbol{v} - \boldsymbol{v}_{\text{air}}) \\ \boldsymbol{T}_{\text{aero}} = \boldsymbol{K}_r(\boldsymbol{\Omega} - \boldsymbol{R}^{\text{T}}\boldsymbol{\Omega}_{\text{air}}) \end{cases} \tag{43}$$

where $\boldsymbol{K}_t$ and $\boldsymbol{K}_r$ are two diagonal aerodynamic matrices and $\boldsymbol{v}_{\text{air}}$ and $\boldsymbol{\Omega}_{\text{air}}$ are the translation and the rotation velocity of the air with respect to the earth-fixed inertial frame. The coefficient matrices of the aerodynamic friction are fixed at $\boldsymbol{K}_t = \text{diag}\{10^{-2}, 10^{-2}, 10^{-2}\}\,\text{N s/m}$ and $\boldsymbol{K}_r = \text{diag}\{10^{-3}, 10^{-3}, 10^{-3}\}\,\text{Nm s/rad}$ given by Madani and Benallegue [10]. The external wind disturbances $\boldsymbol{v}_{\text{air}} = \{0.5, 0.1, 0\}\,\text{m/s}$ and $\boldsymbol{\Omega}_{\text{air}} = \{0, 0, 0\}\,\text{rad/s}$ occur between $t = 45$ sec and $t = 65$ sec. The parameters of the controller still remain the same as Subsection 5.1 and the simulation result of the robustness test is shown in Fig. 12, from which we can see that the tracking errors are very small under the existence of modelling error before the occurrence of the external wind disturbance, and the quadrotor still keeps stable in the wind albeit the obvious tracking errors appear during this period. Thus, the performance shows the stability robustness of the proposed controller. The explanation of the controller robustness is that (i) there exists a PD component in the position controller; (ii) the command filter alleviates the dependence on the mathematical model and (iii) the two-time scale separation assumption between the outer and inner loop is eliminated by the linear tracking differentiator.

6 Conclusions

This paper has presented a trajectory tracking design algorithm for a small quadrotor aircraft. A main motivation is driven by the requirement of constructing the analytical relationship between attitude and linear acceleration as well as eliminating the two-time scale separation assumption between attitude and linear dynamics. Since in the classical flight control system the dynamics of the commanded attitude generated by the outer position loop is neglected through choosing large inner attitude loop gains to separate the time scale, which may deteriorate the system performance, low-pass filters are employed in the backtepping-based attitude stabilisation control law design

doi: 10.1049/iet-cta.2009.0336

to extract and compensate the desired angular velocity and the commanded attitude derivative dynamics. In the end, the numerical simulations for a quadrotor under several conditions are performed and the results demonstrate the controller efficiency of tracking ground trajectory and the robustness to modelling error and external disturbance.

7 Acknowledgments

This work was partially supported by Research Foundation for Key Disciplines of Beijng Municipal Commission of Education (XK100060422). The author sincerely acknowledges the motivating discussions with his supervisor Prof. Huo Wei of the Seventh Research Division, Beijing University of Aeronautics and Astronautics.

8 References

[1] MCKERROW P.: 'Modelling the Draganflyer four-rotor helicopter'. Proc. 2004 Int. Conf. Robotics and Automation, New Orleans, LA, April 2004, pp. 3596–3601

[2] HAMEL T., MAHONY R., LOZANO R., OSTROWSKI J.: 'Dynamic modelling and configuration stabilization for an X4-Flyer'. Proc. IFAC World Congress, Barcelona, Spain, 2002

[3] TAYEBI A., MCGILVRAY S.: 'Attitude stabilization of a four-rotor aerial robot'. 43rd IEEE Conf. Decision and Control, Atlantis, Paradise Island, Bahamas, December 2004, pp. 1216–1221

[4] JOSHI S.M., KELKAR A.G., WEN J.T.: 'Robust attitude stabilization of spacecraft using nonlinear quaternion feedback', *IEEE Trans. Autom. Control*, 1995, **40**, (10), pp. 1800–1803

[5] LIZARRALDE F., WEN J.T.: 'Attitude control without angular velocity measurement: a passivity approach', *IEEE Trans. Autom. Control*, 1996, **41**, (3), pp. 468–472

[6] BOUABDALLAH S., NOTH A., SIEGWART R.: 'PID vs LQ control techniques applied to an indoor micro quadrotor'. Proc. 2004 IEEE/RSJ Int. Conf. Intelligent Robots and Systems, Sandal, Japan, September 2004, pp. 2451–2456

[7] BOUABDALLAH S., SIEGWART R.: 'Backstepping and sliding-mode techniques applied to an indoor micro quadrotor'. Proc. 2005 IEEE Int. Conf. Robotics and Automation, Barcelona, Spain, April 2005, pp. 2247–2252

[8] SCHUMACHER C., KHARGONEKAR P.P.: 'Stability analysis of a missile control system with a dynamic inversion controller', *J. Guid. Control Dyn.*, 1998, **21**, (3), pp. 508–515

[9] MADANI T., BENALLEGUE A.: 'Control of a quadrotor, mini-helicopter via full state backstepping technique'. Proc. 45th IEEE Conf. Decision and Control, Manchester Grand Hyatt Hotel, San Diego, CA, USA, December 2006, pp. 1515–1520

[10] MADANI T., BENALLEGUE A.: 'Backstepping sliding mode control applied to a miniature quadrotor flying robot'. Proc. 32nd IEEE Annual Conf. Industrial Electronics, November 2006, pp. 700–705

[11] POLLINI L., METRANGOLO A.: 'Simulation and robust backstepping control of a quadrotor aircraft'. AIAA Modeling and Simulation Technologies Conf. Exhibit, Honolulu, Hawaii, August 2008

[12] KHALIL H.: 'Nonlinear control systems' (Prentice Hall, 2002, 3nd edn.), Chs. 11, 14

[13] KRSTIĆ M., KANELLAKOPOULOS I., KOKOTOVIĆ P.: 'Nonlinear and adaptive control design' (Wiley, New York, 1995)

[14] FARRELL J., POLYCARPOU M., SHARMA M., DONG W.: 'Command filtered backstepping'. 2008 American Control Conf., Westin Seattle Hotel, Seattle, Washington, USA, June 2008, pp. 1923–1927

[15] GUO B., HAN J.: 'A linear tracking-differentiator and applications to the online estimation of the frequency of a sinusoidal signal'. Proc. 2000 IEEE Int. Conf. Control Applications, Anchorage, Alaska, USA, September 2000, pp. 9–13

[16] HOFFMANN G.M., HUANG H., WASLANDER S.L., TOMLIN C.J.: 'Quadrotor helicopter flight dynamics and control: theory and experiment'. AIAA Guidance, Navigation and Control Conf. Exhibit, Hilton Head, South Carolina, August 2007

[17] FARREL J., SHARMA M., POLYCARPOU M.: 'Backstepping-based flight control with adaptive function approximation', *J. Guid. Control Dyn.*, 2005, **28**, (6), pp. 1089–1102