High-precision prescribed-time path following for quadrotor

Yibo Ding®, Xiaogang Wang, Yuliang Bai and Naigang Cui

Abstract
A high-precision prescribed-time guidance law is developed for a quadrotor to accomplish precise path following at predesigned time $T_0$. Firstly, a new control structure is proposed for a quadrotor to perform fixed-velocity back-to-turn flying mode by introducing four controllers, which can realize a non-sideslip and bank-to-turn flight scheme just like a fixed-wing unmanned aerial vehicle. Then, prescribed-time guidance law is presented based on fixed-velocity back-to-turn flying mode via combining sliding mode control with a compensation function $p(t)$ to improve tracking precision of conventional methods. Compensation function $p(t)$ is designed to make state error achieve convergence at prescribed time exactly. Meanwhile, global sliding mode is established to enhance robustness of the system. Then, the stability and characteristic of prescribed-time convergence are proved strictly. Finally, simulations with a 6-degree-of-freedom quadrotor model are carried out to demonstrate the effectiveness and superiority of prescribed-time guidance law by comparing with traditional guidance law.

Keywords
Prescribed-time, high-precision tracking, back-to-turn flying mode, path following, quadrotor

Date received: 7 March 2019; accepted: 24 February 2020

Introduction
Quadrotors have numerous potential applications no matter for military or civil use. They present many advantages compared with fixed-wing unmanned aerial vehicle (UAV) including high maneuverability, the capability to hover, vertical takeoff, and landing in limited spaces. In recent years, the missions of quadrotor are becoming more complex such as close range surveillance, detection, and commercial package management. All the missions put forward a requirement of an accurate path following method. In addition, to execute time critical mission reliably for quadrotor, such as simultaneous arrival scenario, cooperative surveillance, cooperative target tracking, and spatial-temporal cooperative formation flying, it is necessary to investigate a novel path following method, which can drive quadrotor to achieve the desired path at pre-designed time exactly.

The existing guidance laws of quadrotor achieve position control by tracking a set of given waypoints or time-varying target point. However, such flying mode cannot maintain a constant flight speed, which is proportional to the distance between the quadrotor and the target point. In addition, it cannot achieve precise smooth turn with a constant speed. To improve the conventional flying mode, a new control structure is developed for quadrotor in this article to perform fixed-velocity back-to-turn flying mode, which can realize path following at any given velocity profile. Four controllers are introduced for the control structure, namely course hold controller, longitudinal

Department of aerospace engineering, School of Astronautics, Harbin Institute of Technology, Harbin, China

Corresponding author:
Xiaogang Wang, Department of aerospace engineering, School of Astronautics, Harbin Institute of Technology, No. 92, Xidazhi Street, Harbin 150001, China. Email: wangxiaogang@hit.edu.cn

Creative Commons CC BY: This article is distributed under the terms of the Creative Commons Attribution 4.0 License (https://creativecommons.org/licenses/by/4.0/) which permits any use, reproduction and distribution of the work without further permission provided the original work is attributed as specified on the SAGE and Open Access pages (https://us.sagepub.com/en-us/nam/open-access-at-sage).
controller, lateral controller, and height controller. By means of applying the novel flying mode, quadrotor not only has strong maneuverability but also can perform a non-sideslip and bank-to-turn flight scheme just like a fixed-wing UAV. Based on it, path following guidance law of fixed-wing UAV can be effectively applied to quadrotor. 

There have been many effective methods of path following developed for fixed-wing UAV. An approach named “vector field” is proposed by Miao and Fang,9 the purpose of which is not to make vehicle pursue a moving point but to get on the desired path at given velocity. An adaptive optimal path following is developed by Ratnou et al.10 It regards the path following of a vehicle as a general infinite-horizon nonlinear problem which is solved by combining a linear quadratic regulator with an adaptive gain. A nonlinear guidance law motivated by proportional navigation of missile is introduced by Park et al.11 and Curry et al.,12 which contains an element of anticipatory control to guarantee tight tracking. Sujit et al. performs simulations of some existing path following methods and compares the control efforts and tracking errors of them.13 The foregoing literature are all confirmed to be effective. However, they cannot achieve path following at predesigned time exactly and the tracking accuracies of them are not high enough.

To improve the conventional methods, sliding mode control is recommended for designing path following guidance law, due to its advantages of high precision and robustness. Sliding mode control includes reaching phase and sliding phase. Reaching phase drives system to sliding surface, while sliding phase makes state error slide to origin. There are several forms of sliding surfaces, like linear sliding surface, integral sliding surface, terminal sliding surface, fixed-time sliding surface, and so on.14 Linear sliding surface and integral sliding surface could only realize exponential convergence. Terminal sliding surface can achieve system stabilization in finite time, but the convergence time grows along with the increase of initial condition. The fixed-time sliding surface proposed by Polyakov can overcome this disadvantage.15 However, since the fixed convergence time obtained by theoretical derivation is too conservative compared with actual convergence time, it is difficult to carry out the actual design and parameter adjustment based on theoretical results.16 Therefore, a prescribed-time sliding surface is developed based on a compensation function $p(t)$ in this article, to realize convergence of state at a given predesigned time exactly. In addition, considering robustness of sliding mode control to disturbances exists only in sliding phase, $p(t)$ is designed to eliminate initial reaching phase and ensure that the whole process of system response is robust. By means of combining sliding mode control and the compensation function $p(t)$, a high-precision prescribed-time guidance law (PTGL) is proposed for path following of quadrotor in this article.

In summary, the contributions of this article are given as follows:

1. A novel control structure is designed for quadrotor to realize fixed-velocity back-to-turn flying mode. Based on it, quadrotor can perform a non-sideslip and bank-to-turn flight scheme just like a fixed-wing UAV.
2. A PTGL is proposed for quadrotor to realize path following at prescribed time exactly. Meanwhile, higher tracking precision can be ensured compared with the conventional path following guidance law.

An outline of the article is as follows: The second section presents a dynamic model of quadrotor. The third section introduces the four controllers used to make quadrotor realize fixed-velocity back-to-turn flying mode. The fourth section proposes the PTGL based on an accurate model and presents a strict proof for prescribed-time convergence. In the fifth section, simulations are performed to show effectiveness and superiority of the method proposed.

### Quadrotor modeling

In this section, the dynamic model of quadrotor is developed.17,18 The quadrotor structure is presented in Figure 1, with the definition of inertial frame, body frame, and positive direction of rotor rotation.

![Figure 1. The definition of (a) quadrotor model and (b) coordinate frames.](image-url)
The rotation matrix from inertial frame (I) to body frame (B) is modeled following Z-Y-X order. To transform from I to B, firstly we rotate around \( z_I \) by yaw angle \( \psi \), then rotate around the intermediate \( y_{in} \) by pitch angle \( \theta \), and eventually rotate around \( x_B \) by roll angle \( \phi \). The final rotation matrix is

\[
R_{BI} = \begin{bmatrix}
C_\phi C_\theta & S_\phi & 0 \\
S_\phi C_\theta & C_\phi C_\theta & -S_\theta \\
S_\theta & C_\theta & 0
\end{bmatrix}
\]

in which \( S_\phi = \sin(\phi) \) and \( C_\phi = \cos(\phi) \). The relation between the angular velocity and the attitude angle is shown as

\[
\begin{bmatrix}
\dot{\phi} \\
\dot{\theta} \\
\dot{\psi}
\end{bmatrix} =
\begin{bmatrix}
1 & S_\phi T_\theta & C_\phi T_\theta \\
0 & C_\phi & -S_\phi \\
0 & S_\phi/C_\theta & C_\phi/C_\theta
\end{bmatrix}
\begin{bmatrix}
p \\
q \\
r
\end{bmatrix}
\]

in which \( T_\phi = \tan(\phi) \) and \( [p \ q \ r]^T \) are coordinates of angular velocity in body frame. The translational kinematic relationship is shown below

\[
\begin{bmatrix}
x \\
y \\
z
\end{bmatrix} =
\begin{bmatrix}
V_{tx} \\
V_{ty} \\
V_{tz}
\end{bmatrix}
\]

where \([x \ y \ z]^T\) and \([V_{tx} \ V_{ty} \ V_{tz}]^T\) are coordinates of displacement and velocity in inertial frame, respectively.

The forces acting on quadrotor are gravity \( G \) and aerodynamic force produced by four rotors which is denoted by \( T_a \). The dynamic equations of vehicle are as follows

\[
m
\begin{bmatrix}
\dot{V}_{tx} \\
\dot{V}_{ty} \\
\dot{V}_{tz}
\end{bmatrix} =
R_{BI}^T
\begin{bmatrix}
0 \\
0 \\
-G
\end{bmatrix} + \begin{bmatrix}
0 \\
0 \\
0
\end{bmatrix}
\]

\[
\begin{bmatrix}
I_{xx} \dot{\phi}
I_{yy} \dot{\theta}
I_{zz} \dot{\psi}
\end{bmatrix} =
\begin{bmatrix}
M_0 \\
M_0 \\
M_0
\end{bmatrix} - \begin{bmatrix}
p \\
q \\
r
\end{bmatrix} \times \begin{bmatrix}
I_{xx} p \\
I_{yy} q \\
I_{zz} r
\end{bmatrix}
\]

where \( m \) is vehicle mass. Notations \( I_{xx}, I_{yy}, \) and \( I_{zz} \) are moments of inertia. Variables \([M_0 \ M_0 \ M_0]^T\) are coordinates of total aerodynamic torques in body frame generated by four rotors.

The relationship between forces, torques, and angular velocity \( \omega \) of each rotor is given below

\[
\begin{bmatrix}
T_a \\
M_\phi \\
M_\theta \\
M_\psi
\end{bmatrix} =
R_{xT}
\begin{bmatrix}
\omega_1^2 \\
\omega_2^2 \\
\omega_3^2 \\
\omega_4^2
\end{bmatrix}
\begin{bmatrix}
k & k & k & k \\
0 & -lk & 0 & lk \\
-lk & 0 & 0 & lk \\
-b & b & -b & b
\end{bmatrix}
\begin{bmatrix}
\omega_1^2 \\
\omega_2^2 \\
\omega_3^2 \\
\omega_4^2
\end{bmatrix}
\]

Fixed-velocity back-to-turn flying mode

To realize fixed-velocity back-to-turn flying mode, a new control structure is proposed in this section, composing of four controllers. By means of applying the novel control structure, quadrotor can perform a non-sideslip and bank-to-turn flight scheme just like a fixed-wing UAV.

Firstly, for the sake of achieving non-sideslip flight, course angle \( \chi \) is selected as desired yaw angle. Course hold controller is used to make yaw angle \( \psi \) reach course angle \( \chi \), so as to guarantee the plane determined by \( x_B \) axis and the resultant velocity vector is coincident with vertical plane. The course hold controller loop is shown in Figure 2 in which inner controller is defined applying PD control algorithm for practical engineering application

\[
M_{\psi \psi} = [k_{P\psi}(\chi - \psi) - k_{D\psi}\psi] \cdot I_{zz}
\]

The purpose of longitudinal controller is to make quadrotor achieve the desired velocity \( V_d \) by pitching to desired pitch angle \( \theta_d \). The longitudinal controller loop is shown in Figure 3 in which inner controller is defined applying PD algorithm

\[
M_{\theta \theta} = [k_{P\theta}(\theta_d - \theta) - k_{D\theta}\theta] \cdot I_{yy}
\]

Outer controller is given using proportional–integral–derivative (PID) algorithm as follows
The purpose of lateral controller is to make lateral acceleration $V_{1Bx}$ follow the lateral command acceleration $a_c$ by rolling to desired roll angle $\phi_d$. The lateral controller loop is shown in Figure 4 in which inner controller is defined applying PD algorithm

$$M_{ac} = [k_{Pc}(\phi_d - \phi) - k_{Dc}\dot{\phi}] \cdot I_{xx} \tag{12}$$

Outer controller is given using PI algorithm as follows

$$\phi_d = k_{Ps}(a_c - V_{\dot{X}}) + k_{Ia}\int(a_c - V_{1Bx}) dt \tag{13}$$

Height controller is used to make quadrotor arrive the expected altitude, which can be realized easily by making four rotors speed up or speed down at the same time. Therefore, height command $z_c$ can be imported to inner controller directly. The height controller loop is shown in Figure 5 in which inner controller is defined applying PD algorithm

$$T_c = \{m[k_{Pc}(z_c - z) - k_{Dc}V_{z}] + G\}/(\cos\phi\cos\theta) \tag{14}$$

Although the above four PID controllers are not able to achieve accurate and perfect tracking control as advanced control theories, they are more convenient for practical engineering application. The disturbance in quadrotor dynamics is mainly reflected in parametric uncertainties, such as mass, moments of inertia, lift constant, drag constant, and so on. The magnitude of the disturbance is generally small, which can be solved effectively by robustness of PID controller. Therefore, there is no need to apply the advanced but complex controllers with stronger robustness, such as $H_\infty$ robust control and active disturbance rejection control.

After calculating the four command signals $M_{ac}$, $M_{bc}$, $M_{gc}$, and $T_c$ using the four controllers, desired values of angular velocity for each rotor $\omega_{ec}$ are given as follows

$$\begin{bmatrix} \omega_{1c}^2 \\ \omega_{2c}^2 \\ \omega_{3c}^2 \\ \omega_{4c}^2 \end{bmatrix} = R_{\dot{c}}^T \begin{bmatrix} M_{ac} \\ M_{bc} \\ M_{gc} \\ M_{vc} \end{bmatrix} \tag{15}$$

Based on the new control structure mentioned above, advanced guidance law can be developed for quadrotor to realize high-precision and prescribed-time path following.

In the actual task of path following, desired velocity and desired height are given by the task in advance. To follow the desired path, guidance law designs lateral command acceleration to eliminate cross-track error. Therefore, in the process of performing a path following task, height controller drives quadrotor to the height of path and keeps it. Longitudinal controller adjusts velocity of quadrotor to the desired value and maintains it. Course hold controller is used to make yaw angle $\psi$ reach course angle $\chi$ in real time, to avoid the influence of yaw angle $\psi$ on lateral movement. Thus, lateral movement can only be achieved by rolling to track the designed lateral command acceleration, which is realized by lateral controller.

In summary, among the four controllers, height controller and longitudinal controller are independent, and they are less related to the other two controllers. The course hold controller and lateral controller are integrated. They cooperate with each other to realize back-to-turn flying mode for lateral tracking control. The interaction of the four controllers is illustrated in Figure 6.

**High-precision PTGL design**

In this section, a high-precision PTGL is proposed to decrease cross-track error and realize prescribed-time path following.

In the actual engineering application of path planning for fixed-wing UAV, the real-time characteristics and the small amount of calculation are very important. Therefore, the path planning method based on Dubins path has become an effective and widely used solution, which can realize high-efficient, real-time, and online path planning.\(^{20,21}\) The Dubins path is a shortest path from the starting point with initial prescribed heading to the final position and heading.\(^{22}\) The path consists of circular arcs and straight lines, which establishes optimal transitions between different waypoints.\(^{13}\) Compared with the complicated path, the Dubins path has the shortest length and is easy to implement in engineering. The demonstration of Dubins path is given in Figure 7.

To ensure the practicability of engineering, many existing path-following schemes are designed based on Dubins path. Considering Dubins path is composed of circular arcs and straight lines, the guidance laws designed by
researchers are all divided into two modes: straight line and circular path following modes. Therefore, the straight line and circular path-following modes are also designed, respectively, in this article.

The relationship of vehicle position, velocity, and desired linear path in horizontal plane is shown in Figure 8, where $d$ is the cross-track error. Desired course angle $\chi_p$ is defined as the angle between tangential direction of desired path and $x_I$ axis. The state variables are chosen as

$$
\begin{align*}
    x_1 &= d \\
    x_2 &= V_{1Bx} \sin(\chi - \chi_p)
\end{align*}
$$

Selecting lateral acceleration as control variable: $u = V_{1Bx} \dot{\chi}$. The state equations are satisfied with

$$
\begin{align*}
    \dot{x}_1 &= x_2 \\
    \dot{x}_2 &= u \cdot \cos(\chi - \chi_p) - V_{1Bx} \cos(\chi - \chi_p) \dot{\chi}_p + V_{1Bx} \sin(\chi - \chi_p)
\end{align*}
$$

It should be noted that velocity $V_{1Bx}$, course angle $\chi$, and lateral acceleration $V_{1Bx} \dot{\chi}$ in equation (17) are all independent of height motion. Therefore, the change in height does not have any influence on the effectiveness of guidance law.
Figure 9. Geometry relationship in circular path.

Defining $F(x) = -V_{1B} \cos(\chi - \chi_p) \chi_p$, $G(x) = \cos(\chi - \chi_p)$, and $D(x) = \dot{V}_{1B} \sin(\chi - \chi_p)$, equation (17) can be written as

$$
\begin{align*}
\dot{x}_1 &= x_2 \\
\dot{x}_2 &= F(x) + G(x)u + D(x)
\end{align*}
$$

(18)

Ratnoo et al. assumes that $\dot{\chi}_p = 0$, but it is not reasonable enough and will reduce tracking accuracy. The angle $\chi_p$ should take different values in different cases. If the desired path is a straight line, angle $\chi_p$ is a constant, which means $\dot{\chi}_p$ is equal to zero. However, if the desired path is circular, $\chi_p$ is no longer a constant. The geometry relationship in circular path is shown in Figure 9, where $\theta_c$ represents angular position of quadrotor relative to circular path. Thus, $\chi_p$ can be obtained as

$$
\begin{align*}
\chi_p &= \theta_c + \pi/2 \\
\dot{\chi}_p &= \dot{\theta}_c = V_{1B}/R
\end{align*}
$$

(19)

To enhance convergence precision, guidance law is designed for nonlinear system equation (17) based on sliding mode control. In addition, a compensation function $p(t)$ is introduced to realize prescribed-time convergence. Sliding mode variable $s$ is designed as

$$
s = ce_1 + e_2 - [c\dot{p}(t) + \dot{p}(t)]
$$

(20)

where $c$ must satisfy Hurwitz condition: $c > 0$. Error variables $e_1$ and $e_2$ are defined as

$$
\begin{align*}
e_1 &= x_1 - x_{1d} = d - 0 = x_1 \\
e_2 &= x_2 - x_{2d} = V_{1B} \sin(\chi - \chi_p) - 0 = x_2
\end{align*}
$$

(21)

in which $x_{1d}$ and $x_{2d}$ are the desired values of state variables. Therefore, sliding mode dynamics is satisfied with

$$
\dot{s} = c\dot{x}_1 + \dot{x}_2 - c\dot{p}(t) - \dot{p}(t)
$$

(22)

Combining sliding surface equation (20) and exponential reaching law, the PTGL is proposed as follows

$$
u = \frac{-1}{G(x)} [c[x_2 - \dot{p}(t)] + F(x) - \dot{p}(t) + \eta \text{sgn}(s) + \varepsilon \dot{s}]
$$

(23)

in which $\eta$ and $\varepsilon$ are positive design parameters of reaching law.

The aim of designing function $p(t)$ is to guarantee the errors of state variables converge to zero at prescribed time $T$. Therefore, $p(t)$ is defined as a piecewise polynomial function

$$
p(t) = \begin{cases} 
a_0 + a_1 t + a_2 t^2 + a_3 t^3 + a_4 t^4 & 0 \leq t \leq T \\
0 & T < t
\end{cases}
$$

(24)

In addition, for the sake of establishing global sliding mode, $p(t)$ should satisfy

$$
\begin{align*}
p(0) &= e_1(0) \\
\dot{p}(0) &= e_2(0)
\end{align*}
$$

(25)

To make the second-order derivative of $p(t)$ continuous at given time $T$, conditions below are essential

$$
p(T) = \dot{p}(T) = \ddot{p}(T) = 0
$$

(26)

Applying conditions (25) and (26) into (24), the following equations can be obtained

$$
\begin{align*}
a_0 &= e_1(0) \\
a_1 &= e_2(0) \\
a_0 + a_1 T + a_2 T^2 + a_3 T^3 + a_4 T^4 &= 0 \\
a_1 + 2a_2 T + 3a_3 T^2 + 4a_4 T^3 &= 0 \\
a_2 + 6a_3 T + 12a_4 T^2 &= 0
\end{align*}
$$

(27)

According to equation (27), coefficients are solved as

$$
\begin{align*}
a_0 &= e_1(0) \\
a_1 &= e_2(0) \\
a_2 &= -\frac{3(2e_1 + Te_2)}{T^2} \\
a_3 &= \frac{8e_1 + 3Te_2}{T^3} \\
a_4 &= -\frac{3e_1 + Te_2}{T^4}
\end{align*}
$$

(28)

**Theorem 1.** The errors of state variables $e_1$ and $e_2$ are able to converge to origin at prescribed time $T$ using the PTGL in equation.

**Proof.** The process of proof is divided into two steps. The first step is used to prove the convergence of sliding mode variable $s$, while the second one proves the exact convergence of tracking errors $e_1$ and $e_2$ at given time $T$.

Step 1: Convergence of traditional sliding mode control is composed of reaching phase and sliding phase. The former is used to drive the system converge to sliding surface, which also means to achieve the convergence of sliding mode variable...
\[ s(t) = \text{sign}(s(t)) \frac{|s(t)|}{\varepsilon}. \]

The latter is used to make states \( e_1 \) and \( e_2 \) converge to zero along the sliding surface. Therefore, the convergence time of sliding mode control also consists of reaching time and sliding time.

Combining the definition of sliding mode variable equation (20) and conditions in equation (25), it can be obtained that initial value of \( s \) is satisfied with

\[
\begin{align*}
    s(0) &= c e_1(0) + e_2(0) - (c p(0) + \dot{p}(0)) \\
    &= c e_1(0) - p(0) + [e_2(0) - \dot{p}(0)] \\
    &= c \times 0 + 0 = 0
\end{align*}
\]

which means that system using PTGL lies in sliding surface from the beginning. Therefore, compared with applying traditional sliding mode control, the system using PTGL does not have initial reaching phase and the convergence of system starts directly from the sliding phase.

Substituting equations (23) and (20) into equation (18), dynamics of \( s \) is satisfied with

\[
\begin{align*}
    \dot{s} &= c \dot{e}_1 + \dot{e}_2 - [c \dot{p}(t) + \dot{p}(t)] \\
    &= c x_2 + F(x) + G(x)a + D(x) - [c \dot{p}(t) + \dot{p}(t)] \\
    &= c x_2 + F(x) - c [x_2 - \dot{p}(t)] + F(x) - \dot{p}(t) + \eta \text{sgn}(s) + \varepsilon s \\
    &+ D(x) - [c \dot{p}(t) + \dot{p}(t)] \\
    &= -\eta \text{sgn}(s) - \varepsilon s + D(x)
\end{align*}
\]

(30)

A positive Lyapunov function for sliding mode variable \( s \) is defined as \( V_L = s^2 \). The derivative of \( V_L \) is satisfied with

\[
\begin{align*}
    \dot{V}_L &= 2 s \dot{s} = 2 s[-\eta \text{sgn}(s) - \varepsilon s + D(x)] \\
    &= -2\eta |s| - 2\varepsilon s^2 + 2s D(x) = -2\eta V_L^{1/2} - 2\varepsilon V_L + 2s D(x)
\end{align*}
\]

(31)

If \( V_{1bs} \) is equal to the desired velocity \( V_d \) initially, which means that \( D(x) = 0 \), \( V_L \) is satisfied with \( \dot{V}_L = -2\eta V_L^{1/2} - 2\varepsilon V_L \). Considering \( s \) is equal to zero initially, control law equation can keep \( s \) always equal to zero.

If \( V_{1bs} \) is not equal to the desired velocity \( V_d \), the initial dynamics of \( s \) is satisfied with: \( \dot{s}(0) = -\eta \text{sgn}(s(0)) - \varepsilon s(0) + D(x) = D(x) \). Therefore, \( s \) will diverge to leave origin under the influence of \( D(x) \). According to expression of \( \dot{V}_L \), the maximum value of divergence for \( s \) is \( \|D(x)\| - \eta / \varepsilon \). If \( s > \|D(x)\| - \eta / \varepsilon \), \( V_L \) is negative and \( s \) will converge to less than \( \|D(x)\| - \eta / \varepsilon \). Since the longitudinal controller can drive \( V_{1bs} \) to track the desired velocity \( V_d \) quickly, \( D(x) \) can be equal to zero once \( V_{1bs} = V_d \). When \( D(x) = 0 \), derivative of Lyapunov function \( V_L \) is satisfied with: \( \dot{V}_L = -2\eta V_L^{1/2} - 2\varepsilon V_L \). According to fast finite-time Lyapunov criteria,\(^{23} \) \( s \) is able to converge to zero in finite time.

Step 2: The following relationship is satisfied according to equation (20)

\[
\frac{de_1}{dt} + ce_1 = s + \dot{p}(t) + cp(t)
\]

Solving differential equation (32), \( e_1 \) can be obtained

\[
\begin{align*}
    e_1 &= e^{-\eta t} \left\{ e_1(0) + \int_0^t e^{\eta \tau} \dot{p}(\tau) + cp(\tau) + s d\tau \right\} \\
    &= e^{-\eta t} \left\{ e_1(0) + [e^\eta p(\tau)]|_0^t + \int_0^t e^{\eta \tau} s d\tau \right\} \\
    &= e^{-\eta t} \left\{ e_1(0) + e^\eta p(t) - e^\eta p(0) + e^{-\eta t} \int_0^t e^{\eta \tau} s d\tau \right\} \\
    &= p(t) + e^{-\eta t} [e_1(0) - p(0)] + e^{-\eta t} \int_0^t e^{\eta \tau} s d\tau
\end{align*}
\]

(33)

If \( V_{1bs} \) is equal to \( V_d \) initially, \( s \) will always be equal to zero according to the above analysis. Considering \( p(0) = e_1(0) \) in equation (25), the equation \( e_1 = p(t) \) is satisfied. According to equation (26), designing function \( p(t) \) and \( \dot{p}(t) \) are equal to zero at the prescribed time \( T \). Therefore, \( e_1 \) and \( e_2 \) can also converge to zero at prescribed time \( T \).

If \( V_{1bs} \) is not equal to the desired velocity \( V_d \), \( s \) will diverge to leave origin under the influence of \( D(x) \). Therefore, it can be obtained that \( e_1 = p(t) + e^{-\eta t} \int_0^t e^{\eta \tau} s d\tau \).

Without loss of generality, it is assumed that \( \sin(\chi - \gamma_p) > 0 \). If \( V_{1bs} \) is less than \( V_d \) initially, \( V_{1bs} \) is positive which makes \( D(x) \) be positive. Therefore, \( s \) will be larger than zero, which will make \( e^{-\eta t} \int_0^t e^{\eta \tau} s d\tau \) positive. Finally, \( e_1 \) will be slightly larger than \( p(t) \). Conversely, if \( V_{1bs} \) is larger than \( V_d \) initially, \( s \) will be less than zero. Therefore, \( e_1 \) will be slightly less than \( p(t) \).

According to the above analysis, the maximum value of divergence for \( s \) is \( \|D(x)\| - \eta / \varepsilon \). By means of selecting larger value of \( \varepsilon \) and \( \eta \), the maximum value of divergence \( \|D(x)\| - \eta / \varepsilon \) can be smaller. The value of \( e^{-\eta t} \int_0^t e^{\eta \tau} s d\tau \) is quite small. Therefore, the influence on convergence of \( e_1 \) is almost negligible, which means the prescribed-time convergence of \( e_1 \) can also be guaranteed.

**Remark 1.** Considering sliding mode variable \( s \) is initially equal to zero, system is located on sliding surface at the beginning. Therefore, system can eliminate initial reaching phase, so as to accelerate response speed and enhance robustness.

**Remark 2.** If compensation function \( p(t) \) is defined as zero, the PTGL in equation will degenerate into high-precision guidance law (HPGL) and lose its character of prescribed-time convergence but remain robust and accurate.

**Remark 3.** In order to restrain the chattering phenomenon in engineering application, a saturation function can be used to replace sign function \( \text{sgn}(s) \) in equation.
Table 1. Parameter table of quadrotor.

| Parameter | Value |
|-----------|-------|
| \( m \)   | 0.468 kg |
| \( k \)   | \( 2.98 \times 10^{-6} \) kg m |
| \( b \)   | \( 1.14 \times 10^{-7} \) kg m² |
| \( l \)   | 0.225 m |
| \( g \)   | 9.81 m s⁻² |
| \( \text{diag}[I_{xx} I_{yy} I_{zz}] \) | \( \text{diag}[4.856 4.856 8.801] \times 10^{-3} \) kg m² |

in which \( \Delta \) is a sufficiently small constant.

\[ \text{sat}(s) = \begin{cases} 
1 & s > \Delta \\
\frac{s}{\Delta} & |s| \leq \Delta \\
-1 & s < -\Delta 
\end{cases} \quad (34) \]

Remark 4. In the actual path following task, in order to track the desired Dubins path, quadrotor can switch between straight line and circular path-following modes automatically in real time, according to different desired path segments.

Simulation

In this section, a high-fidelity nonlinear 6-degree-of-freedom model of quadrotor is implemented with the proposed control structure and guidance law. The nominal parameters of quadrotor are given in Table 1.²⁴,²⁵ Three subsections are carried out to verify the effectiveness of method proposed in detail. “Straight line following” section compares HPGL and PTGL in linear path following. “Circular path following” section compares the method in this article with a conventional guidance law in circular path following, so as to highlight the advantages of high precision and prescribed-time convergence. “Complex path following” section verifies the validity of PTGL on a complex path following task.

The desired ground velocity and height command are \( V_d = 1 \) m/s and \( z_c = 3 \) m, respectively. The initial conditions are taken as \([x \ y \ z]^T = [10 \ -20 \ 3] \) m, \([V_{x_1} \ V_{y_1} \ V_{z_1}]^T = [1 \ -0.3 \ 0] \) m/s, and \([\phi \ \theta \ \psi]^T = [p \ q \ r]^T = [0 \ 0 \ 0] \).

Controllers of quadrotor are designed according to the “Fixed-velocity back-to-turn flying mode” section to realize fixed-velocity back-to-turn flying mode. To suppress the influence of high-frequency noise in practical applications, parameters of PID controllers are adjusted to minimize the value of amplitude–frequency curve in high frequency band. By means of designing appropriate bandwidth, the introduction of high-frequency noise can be avoided. Parameters for four controllers are selected as follows

\[ k_{P_v} = k_{P_\theta} = k_{P_{\psi}} = 15.8285 \]
\[ k_{D_v} = k_{D_\theta} = k_{D_{\psi}} = 8.8342 \]
\[ k_{P_V} = 0.984 \quad k_{P_{\theta}} = 0.001 \quad k_{P_{\psi}} = 0.4 \quad (35) \]
\[ k_{P_\phi} = -1.5 \quad k_{I_\phi} = -0.02 \]
\[ k_{P_z} = 70.5 \quad k_{D_z} = 20.5 \]

Remark 5. In order to avoid the accident of quadrotor caused by an excessive attitude angle, the magnitudes of desired angles \( \theta_\phi \) and \( \phi_\phi \) are limited between \(-20^\circ \) and \( 20^\circ \) in this article. Although the constraints are conservative, they are sufficient to implement the task scenario given in this article. If the task requirements of quadrotor are strict, the angle constraints can be amplified. For example, magnitudes of desired angles \( \theta_\phi \), \( \phi_\phi \) can be limited between \(-40^\circ \) and \( 40^\circ \). This will not affect the effectiveness of the algorithm in this article.

Straight line following

In this subsection, two cases of simulation are carried out. Case A utilizes the HPGL without compensation function \( p(t) \) as comparative simulation, while Case B applies the PTGL with compensation function to verify its effectiveness.

From a theoretical point of view, cross-track error \( e_1 \) can achieve precise convergence effectively. However, in practical applications, tracking error is inevitable due to the effect of sampling step. Therefore, the tracking task can be considered complete when error \( e_1 \) is less than \( 10^{-4} \) m for straight line following in this article. Symbol \( t_1 \) is defined as the time after which \( e_1 \) converges to less than \( 10^{-4} \) m.

The design parameters of the guidance law are selected as \( c = 1.5 \), \( \eta = 0.1, \varepsilon = 0.03 \), and \( \Delta = 0.035 \).

Case A:

It can be seen from Figures 10 and 11 that quadrotor tracks the desired path directly using HPGL and the final cross-track error is almost zero. In this case, \( t_1 \) is equal to \( 72.37 \) s.

Case B: The prescribed time \( T \) is set as \( 120 \) s for PTGL in this case.

Figure 12 shows that quadrotor does not fly directly to the path like Figure 10, to realize path following at given time. It can be observed from Figure 13 that \( e_1 \) and \( P(t) - e_1 \) can achieve convergence exactly at prescribed time. The convergence time \( t_1 \) is \( 120.15 \) s, which is approximately equal to prescribed time \( T \). Comparing Case A and Case B, the effectiveness of PTGL proposed is verified sufficiently in the aspect of prescribed-time convergence.

In Figure 14, tracking performance of course hold controller and longitudinal controller are displayed. It can be
seen that yaw angle $\psi$ can track the course angle $\chi$ accurately to realize no-sideslip flight under the effect of course hold controller. Longitudinal controller drives the pitch angle $\theta$ to reach a stable value quickly, so as to make velocity $V_{1Bx}$ track the desired value $V_d$ effectively. In Figure 15, tracking performance of lateral controller and height controller are demonstrated. It can be observed that lateral maneuver of quadrotor is realized by rolling motion. By means of generating roll angle $\phi$, quadrotor can enable lateral acceleration to track the lateral command acceleration $a_c$. Since cross-track error $e_1$ achieves convergence exactly at prescribed time, roll angle $\phi$, and lateral acceleration also converge to zero at prescribed time, so that no lateral motion will be generated after 120 s. In addition, height of quadrotor can be kept exactly equal to height command $z_c$, under the action of height controller.

To verify the effectiveness of prescribed-time convergence more comprehensively, two different simulations are performed using PTGL with prescribed time $T$ set as 90 and 130 s.

Figures 16 and 17 show that quadrotor can follow the desired path with convergence time $t_1 = 90.31$ s. Similarly, it can be observed from Figures 18 and 19 that convergence time $t_1$ is 129.78 s. The two convergence times are all approximately equal to given prescribed times. Therefore, the two simulations demonstrate that PTGL are able to adjust the convergence time freely according to task requirements while ensuring convergence.

However, it should be noted that the prescribed time $T$ of PTGL can only be set to be longer than convergence time of HPGL. When utilizing HPGL, quadrotor flies directly to the
desired path, to eliminate tracking error as quickly as possible. However, when utilizing PTGL, quadrotor changes its flight path, instead of flying directly to the desired path, so as to achieve path following at prescribed time $T$. Therefore, the convergence time of HPGL can be regarded as the lower bound of the convergence time of PTGL. It is unrealistic to make PTGL achieve faster convergence speed than HPGL just by setting shorter prescribed time $T$, and it is not in accordance with the actual physical constraints of quadrotor.

If the convergence speed of PTGL needs to be improved, the convergence speed of its basic method HPGL should be improved firstly. By means of increasing coefficients $\eta$, $\varepsilon$, and $c$, HPGL can achieve faster response speed, which means that the convergence time of HPGL is shorten. Therefore, the prescribed time $T$ of PTGL can be set to be shorter than before.

**Circular path following**

In this subsection, four cases of simulation are carried out. Case A utilizes a conventional guidance law as comparative simulation. Case B performs simulation using HPGL without compensation function to compare with Case A. The PTGL proposed in this article is applied in Case C. To verify the effectiveness of PTGL when height and velocity of quadrotor change dramatically, Case D performs simulation using PTGL under different initial height and velocity. Symbol $t_2$ is defined as the time after which the cross-track error $e_{1c}$ converges to steady state.

Case A: The nonlinear guidance law (NLGL) is given as

$$u = 2\frac{V^2}{L_1}\sin\mu$$  \hspace{1cm} (36)
in which $L_1$ is a line defined from quadrotor to a reference point on desired path and $\mu$ is the angle created from $V$ to the line $L_1$. In this case, value of $L_1$ is designed as 3 m to produce better results. If the initial relative distance is larger than 3 m, quadrotor is driven to fly directly toward the center of circular path.

Figures 20 and 21 show that the settling time is about 34.78 s, and the final tracking error is 0.0665 m using NLGL.

Case B: The design parameters of HPGL without compensation function are selected as $c = 1$, $\eta = 0.1$, $\varepsilon = 0.12$, and $\Delta = 0.04$.

Figures 22 and 23 demonstrate that the final cross-track error is 0.0055 m and $t_2$ is equal to 25.12 s approximately. It is obvious that HPGL can achieve higher tracking precision compared with NLGL, which confirms the descriptions in Remark 2.

Case C: The design parameters of PTGL in this article are selected as the same as that in Case B. The prescribed time $T$ is set as 40 s in this case.

Figure 24 shows that quadrotor takes longer time to settle on the desired path compared with Figure 22 in Case B. Figure 25 demonstrates that $t_2$ is 39.42 s, which is approximately equal to prescribed time $T$. Cross-track error $e_1$ is still equal to $-0.0055$ m. It can be summarized that PTGL not only can achieve convergence at prescribed time but also can retain the higher precision of HPGL, compared with conventional NLGL method. Although the tracking accuracies in numerical simulation are high enough, it should be noted that it is impossible to achieve such high accuracy in real world.
precision in practice due to the influences of sensor errors and wind variations.

Similarly to Case B in “Straight line following” section, tracking performance of course hold controller and longitudinal controller are displayed in Figure 26. Course hold controller drives yaw angle $\psi$ to track the course angle $\chi$ accurately to realize no-sideslip flight, while longitudinal controller makes quadrotor achieve the desired velocity $V_d$ by pitching to desired pitch angle. In Figure 27, lateral controller generates roll angle $\phi$, to follow the lateral command acceleration $a_{\theta}$. It can be seen that roll angle $\phi$ and lateral acceleration converge to stable values after prescribed time, because cross-track error $e_1$ achieves convergence exactly at prescribed time. However, different from Case B in “Straight line following” section, it should be noted that stable values of $\phi$ and lateral acceleration are not equal to zero, which is used to achieve exact circular path following. In addition, tracking performance of height value can also be guaranteed similarly to Case B in “Straight line following” section.

Case D: In this case, PTGL is applied in different initial conditions and disturbance, to demonstrate that effect of PTGL will almost be unaffected by drastic changes in height and velocity. The simulation conditions are listed in Table 2.

Compared with Case C, Case D.1 changes the initial height of quadrotor and Case D.2 changes the initial velocity of quadrotor. Case D.3 adds disturbance $D_{diss}$ to aerodynamic force $T_a$ during the flight of quadrotor, so as to induce sudden changes of velocity and height. Figures 28 to 31 show the simulation results of Case D.1. The results of Case D.2 are given in Figure 32 to 35, while the results
of Case D.3 are provide in Figures 36 to 39. By means of comparing the four cases of simulation, it can be seen that the effectiveness of PTGL proposed will not be affected, even if changing the initial velocity and altitude values dramatically. Similarly, the effectiveness of PTGL will also not be affected by the sudden changes of velocity and height. Not only that, it can be observed that tracking precision and convergence time of \( e_1 \) are almost unaffected.

In summary, the change in height does not have any influence on the effectiveness of PTGL whether in theory or in practice. The change in velocity will influence the convergence precision of \( e_1 \) slightly in theory, according to the analysis in proof of Theorem 1. However, this influence can be negligible in actual use.

**Remark 6.** Although the PTGL proposed could accomplish precise path following at predesigned time \( T \), the prescribed-time convergence is realized via changing its flight path, instead of flying directly to the desired path. It should be noted that PTGL will cost more spaces compared with conventional path following methods. Therefore, PTGL is suitable for path following tasks in more open areas. If the surrounding environment is relatively narrow, such as a narrow tunnel, it is not recommended to use PTGL, to ensure the safety of quadrotor.

### Complex path following

In this subsection, quadrotor is driven to follow a complex path, which is a combination of straight and circular path, just like a track. The positions of four waypoints of the complex path are set as

\[
A = C_0 \begin{bmatrix} 40 & 0 & 3 \\ 138 \end{bmatrix}, \\
B = C_0 \begin{bmatrix} 40 & 0 & 3 \\ 138 \end{bmatrix}, \\
C = C_0 \begin{bmatrix} 40 & 40 & 3 \\ 138 \end{bmatrix}, \\
D = C_0 \begin{bmatrix} 40 & 0 & 3 \\ 138 \end{bmatrix}
\]

while the radius of the arcs is equal to 20 m. The initial position and velocity of quadrotor are \( \begin{bmatrix} x \\ y \\ z \end{bmatrix}^T = \begin{bmatrix} 0 \\ -10 \\ 1 \end{bmatrix} \) m and \( \begin{bmatrix} V_{lx} \\ V_{ly} \\ V_{lz} \end{bmatrix}^T = \begin{bmatrix} 1 \\ 0.3 \\ 0 \end{bmatrix} \) m/s. Parameters of PTGL are selected as the same as those in “Circular path following” section.

To increase the persuasiveness of PTGL, two cases of simulation are performed. Case A.1 sets prescribed time \( T \) as 25 S, while Case A.2 sets \( T \) as 90 S.

Figures 40 and 41 show that PTGL is able to make quadrotor follow the desired complex path effectively with high tracking precision. The convergence time \( t_1 \) of Case A.1 is 25.12 s, which is approximately equal to

![Figure 26. (a to c) Tracking performance of course hold controller and longitudinal controller.](image)

![Figure 27. (a to c) Tracking performance of lateral controller and height controller.](image)

### Table 2. Simulation conditions of Case D.

| Case   | Initial height z | Initial velocity vector \( \begin{bmatrix} V_{lx} \\ V_{ly} \\ V_{lz} \end{bmatrix}^T \) | Disturbance                                      |
|--------|------------------|-----------------------------------------------|--------------------------------------------------|
| Case D.1 | 12 m             | \( \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \) m/s | None                                             |
| Case D.2 | 3 m              | \( \begin{bmatrix} 8 \\ 0 \\ 0 \end{bmatrix} \) m/s | None                                             |
| Case D.3 | 3 m              | \( \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \) m/s | Aerodynamic force \( T_a \) with disturbance \( D_{dis} = \begin{cases} 8\sin(0.7t) & 15 < t < 20 \\ 0 & \text{else} \end{cases} \) |
Figure 28. Result of circular path following using PTGL in Case D.1. PTGL: prescribed-time guidance law.

Figure 29. (a and b) Cross-track error of circular path following using PTGL in Case D.1. PTGL: prescribed-time guidance law.

Figure 30. (a to c) Tracking performance of course hold controller and longitudinal controller in Case D.1.

Figure 31. (a to c) Tracking performance of lateral controller and height controller in Case D.1.

Figure 32. Result of circular path following using PTGL in Case D.2. PTGL: prescribed-time guidance law.

Figure 33. (a and b) Cross-track error of circular path following using PTGL in Case D.2. PTGL: prescribed-time guidance law.
Figure 34. (a to c) Tracking performance of course hold controller and longitudinal controller in Case D.2.

Figure 35. (a to c) Tracking performance of lateral controller and height controller in Case D.2.

Figure 36. Result of circular path following using PTGL in Case D.3. PTGL: prescribed-time guidance law.

Figure 37. (a and b) Cross-track error of circular path following using PTGL in Case D.3. PTGL: prescribed-time guidance law.

Figure 38. (a to c) Tracking performance of course hold controller and longitudinal controller in Case D.3.

Figure 39. (a to c) Tracking performance of lateral controller and height controller in Case D.3.
prescribed time \( T \). The tracking errors of straight line and circular path are the same as the results in previous two subsections.

Tracking performance of course hold controller and longitudinal controller are displayed in Figure 42. In Figure 43, it can be seen that roll angle \( \phi \) and lateral acceleration converge to zero at prescribed time, so that \( e_1 \) achieves convergence exactly at prescribed time. When flight path switches between straight line and arc, the lateral acceleration command will generate a step. The reason for this phenomenon is that the \( F(x) \) used in PTGL has changed. Considering \( \chi_p \) is equal to zero in straight line and equal to \( V_{1Bx}/R \) in circular path, the value of \( F(x) \) will change stepwise once the path is switched. According to Figure 43, when lateral acceleration command changes, rolling controller responds quickly to produce a roll angle to achieve the desired acceleration. Therefore, such switches will not cause instability of quadrotor.

Similarly, Figures 44 and 45 display the results of Case A.2. The convergence time of Case A.2 is 90.26 s, which is approximately equal to prescribed time \( T \). Tracking performance of course hold controller, longitudinal controller, lateral controller, and height controller are shown in Figures 46 and 47, respectively, which are similar to Figures 42 and 43. It should be noted that due to the longer prescribed time set in Case A.2, quadrotor achieves prescribed-time convergence on circular path instead of on straight line path as in Case A.1. According to the results in Case A.1 and Case A.2, it can be concluded that PTGL can also adjust the
convergence time freely to achieve accurate path following even if the path is complex.

**Conclusion**

In this article, a new control structure is proposed for quadrotor, which can perform a non-sideslip and bank-to-turn flight scheme just like a fixed-wing UAV. Based on it, a PTGL is presented to realize path following at pre-designed time by means of combining sliding mode control and a compensation function \( p(t) \). Simulations demonstrate that the PTGL could make quadrotor converge to desired path at prescribed time precisely whether in linear or circular path following. Meanwhile, high-precision tracking of PTGL is verified by comparison with traditional guidance law. In the future research, the robustness of the method to wind disturbances will be verified and the implementation on an experimental testbed will be performed.

**Declaration of conflicting interests**

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

**Funding**

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work was supported by the National Nature Science Fund of China [No. 61703125].

**ORCID iD**

Yibo Ding  https://orcid.org/0000-0003-4111-9713
References

1. Mardan M, Esfandiari M, and Sepehri N. Attitude and position controller design and implementation for a quadrotor. Int J Adv Robot Syst 2017; 14: 172988141770924.

2. Cabecinhas D, Cunha R, and Silvestre C. A nonlinear quadrotor trajectory tracking controller with disturbance rejection. Contr EngPract 2014; 26: 1–10.

3. Thanh HLNN, Phi NN, and Hong SK. Simple nonlinear control of quadcopter for collision avoidance based on geometric approach in static environment. Int J Adv Robot Syst 2018; 15: 172988141876757.

4. Li Z, Ma X, and Li Y. Model-free control of a quadrotor using adaptive proportional derivative-sliding mode control and robust integral of the signum of the error. Int J Adv Robot Syst 2018; 15: 172988141880088.

5. Shirani B, Najafi M, and Izadi I. Cooperative load transportation using multiple UAVs. Aerosp Sci Technol 2019; 84: 158–169.

6. Liu H, Ma T, Lewis FL, et al. Robust formation control for multiple quadrotors with nonlinearities and disturbances. IEEE Trans Cybern 2020; 50(4): 1362–1371.

7. Dong W, Gu G-Y, Zhu X, et al. Development of a quadrotor test bed—modelling, parameter identification, controller design and trajectory generation. Int J Adv Robot Syst 2015; 12: 7.

8. Cabecinhas D, Cunha R, and Silvestre C. A globally stabilizing path following controller for rotocraft based on wind disturbance rejection. IEEE T Contr Syst T 2015; 23: 708–714.

9. Miao C-X and Fang J-C. An adaptive three-dimensional nonlinear path following method for a fix-wing micro aerial vehicle. Int J Adv Robot Syst 2017; 9: 206.

10. Ramoow A, Sujit PB, and Kothari M. Adaptive optimal path following for high wind flights. IFAC Proc 2011; 44: 12985–12990.

11. Park S, Deyst J, and How JP. Performance and Lyapunov stability of a nonlinear path following guidance method. J Guid Control Dynam 2007; 30: 1718–1728.

12. Curry R, Lizarraaga M, Mairs B, et al. L+2, an improved line of sight guidance law for UAVs. In: 2013 American Control Conference, Washington, DC, USA, 17–19 June, 2013. IEEE, pp. 1–6. New York, NY, USA: IEEE.

13. Sujit PB, Saripalli S, and Sousa JB. Unmanned aerial vehicle path following: a survey and analysis of algorithms for fixed-wing unmanned aerial vehicles. IEEE Contr Syst 2014; 34: 42–59.

14. Ding Y, Wang X, Bai Y, et al. Robust fixed-time sliding mode controller for flexible air-breathing hypersonic vehicle. ISA Trans 2019; 90: 1–18.

15. Polyakov A. Nonlinear feedback design for fixed-time stabilization of linear control systems. IEEE Trans Autom Contr 2012; 57: 2106–2110.

16. Tian B, Zuo Z, Yan X, et al. A fixed-time output feedback control scheme for double integrator systems. Automatica 2017; 80: 17–24.

17. Michael N, Mellinger D, Lindsey Q, et al. The GRASP multiple micro-UAV testbed. IEEE Robot Autom Mag 2010; 17: 56–65.

18. Mahony R, Kumar V, and Corke P. Multirotor aerial vehicles: modeling, estimation, and control of quadrotor. IEEE Robot Autom Mag 2012; 19: 20–32.

19. Chen Q, Wang X, Yang J, et al. Trajectory-following guidance based on a virtual target and an angle constraint. Aerosp Sci Technol 2019; 87: 448–458.

20. Beard RW, Ferrin J, and Humpherys J. Fixed wing UAV path following in wind with input constraints. IEEE T Contr Syst T 2014; 22: 2103–2117.

21. Jung W, Lim S, Lee D, et al. Unmanned aircraft vector field path following with arrival angle control. J Intell Robot Syst 2016; 84: 311–325.

22. Shah MZ, Samar R, and Bhatti AI. Lateral track control of UAVs using the sliding mode approach: from design to flight testing. T I Meas Control 2014; 37: 457–474.

23. Yu S, Yu X, Shirinzadeh B, et al. Continuous finite-time control for robotic manipulators with terminal sliding mode. Automatica 2005; 41: 1957–1964.

24. Tayebi A and McGilvray S. Attitude stabilization of a VTOL quadrotor aircraft. IEEE T Contr Syst T 2006; 14: 562–571.

25. Tayebi A and McGilvray S. Attitude stabilization of a four-rotor aerial robot. In: IEEE Conference on Decision & Control, Nassau, Bahamas, 14–17 December, 2004, pp. 1216–1221. New York, NY, USA: IEEE.