Chapter 8
Observer-based Feedback Linearization Control of a Quadrotor Subjected to Sensor Noise

Ahmet Ermeydan and Aziz Kaba

Acronyms

GSA   Gravitational search algorithm
H∞   H-infinity
LQR   Linear quadratic regulator
PD   Proportional derivative
PID   Proportional, integral, derivative
SNR   Signal-to-noise ratio
UAV   Unmanned aerial vehicle
VAR   Variance

8.1   Introduction

Nowadays, there are many application fields for unmanned aerial vehicles (UAVs) and drones such as surveillance, emergency operations, military applications, and package transportation (Metin and Aygün 2019). In particular, quadrotor becomes popular among researchers due to its hovering capacity and high maneuverability. Quadrotor is a highly nonlinear system and hard to control with linear control methods that operate around an equilibrium point (Zuo 2010). Although linear control techniques such as proportional derivative (PD) (Kıyak 2016), proportional integral derivative (PID) (Ermeydan and Kiyak 2017), linear quadratic regulator (LQR) (Khatoon et al. 2014), and H-infinity (H$_\infty$) (Araar and Aouf 2014) are applied

A. Ermeydan (✉) · A. Kaba
Eskisehir Technical University, Eskisehir, Türkiye
e-mail: aermeydan@eskisehir.edu.tr; azizkaba@eskisehir.edu.tr

to the quadrotor successfully, these control approaches may experience a performance drop when deviating from the equilibrium point (Kendoul 2012).

Recently, nonlinear control techniques have become widespread when it comes to the quadrotor. A PID-like speed controller is proposed by Yazar et al. (2018) for a nonlinear small-scale turbojet engine. Ai and Yu (2019) proposed a flatness-based sliding mode control is proposed for the trajectory tracking problem in a quadrotor in the presence of external disturbances. A backstepping-like feedback linearization method is suggested by Choi and Ahn (2014) for controlling the quadrotor. Köksal et al. (2016) presented a two-level nonlinear control design. An incremental sliding mode fault-tolerant flight controller is proposed by Wang et al. (2019). Xuan-Mung and Hong (2019) presented a new altitude algorithm that consists of nonlinear and linear controllers for quadrotors. A nonlinear PID controller is suggested by Najm and Ibraheem (2019) to stabilize a quadrotor system. To track the position and attitude of a quadrotor, a new control method is proposed by Razmi and Afshinfar (2019). A neural network is used to tune sliding mode controllers in the suggested method. The proposed method obtained better results for both attitude and position tracking in comparison to (Zheng et al. 2014) and (Xiong and Zhang 2017). A robust nonlinear control method is utilized to control the quadrotor's position and orientation (Labbadi and Cherkaoui 2019). A dynamic inversion based nonlinear controller is suggested for quadrotors by Prabhakaran et al. (2015). Nonlinear optimal backstepping control is applied to a quadrotor (Basri et al. 2015).

Literature survey showed that the control of the quadrotor can be insufficient when the dynamics are considered linear. Recent research efforts are shifting toward the nonlinear dynamics and control. Feedback linearization technique as one of the nonlinear controllers can transform the nonlinear plant dynamics to a linear dynamical form by using a proper feedback law (Özbek et al. 2016). In difference to linear controllers, feedback linearization control law is obtained through the exact state transformation rather than the state approximation approach (Gee et al. 1998). So, the nonlinear model of the plant can be converted into a fully or partly linear model with the cancellation of nonlinearities. The application of the feedback linearization control technique into the quadrotor control problem can be found in many research papers. Voos (2009) proposed the nonlinear control of a micro-quadrotor platform by using a feedback linearization technique. Saif (2017) proposed a feedback linearization control law for a tiltable quadrotor under wind gusts scenario. Bonna and Camino (2015) proposed the trajectory tracking control of a quadrotor with feedback linearization technique. Fault-tolerant control application of the feedback linearization approach is proposed by Ghandour et al. (2014) to a quadrotor testbed. Position-tracking control of the quadrotor is proposed by Kuantama et al. (2018) using the feedback linearization linear quadratic regulator (LQR) technique. Quadrotor control using feedback linearization with dynamic extension is proposed by Al-Hiddabi (2009). Attitude and altitude control of a quadrotor with feedback linearization technique is presented by Eltayeb et al. (2019).

Despite the advantages of the feedback linearization technique, the controller operates if all the state parameters are well-defined and do not have any uncertainties (Chen et al. 2016). It is a challenging but an essential task to control the quadrotor in

full degree-of-freedom space since quadrotors are both subjected to internal and external noise sources such as modeling inaccuracies, assumptions, wind gusts, measurement noises, and sensor errors. To overcome these disturbances, the feedback linearization control technique is enhanced with an observer to handle the control of the quadrotor subjected to internal and external disturbances.

Several disturbance-observer-based control techniques are available in the literature Chen et al. (2016). In this chapter, the enhancement of the feedback linearization control method is ensured by particle filter–based observer. The particle filter is a Monte Carlo–based suboptimal algorithm that deals with closed-loop integral problems of the recursive Bayesian estimation (Oh and Kim 2019). Unlike linear observers and simple robust controller structures, the particle filter can handle system nonlinearities and noise with both Gaussian and non-Gaussian distributions (Daroogheh et al. 2018).

Based on the given discussion, the originality of this manuscript is to propose a nonlinear observer–based feedback linearization control method of a quadrotor that solves the integration problems, overcomes the uncertainty issue of the feedback linearization technique, and not only reduces disturbance errors and minimizes sensor errors but also controls the quadrotor with nonlinear dynamics in a wide range of operation regions.

The organization of the chapter is as follows: Nonlinear kinematic and dynamic equations of the quadrotor are derived in Sect. 8.2. The proposed method and theoretical background are given in Sect. 8.3. The main results are covered in Sect. 8.4. Conclusions and discussions are given in Sect. 8.5.

8.2 Quadrotor

Quadrotor is a multirotor that consists of four motors and propellers. The two opposite propeller turn in the same direction and the other two turn in the opposite direction. The speeds of the rotors are equal in a balanced flight (Kose and Oktay 2019).

8.2.1 Euler Transformation

There are different methods for expressing a vector in one axis set in another axis set. Euler transformation is one of these methods. In this transformation, one set of axes is rotated three times to overlap the other. Here, the order of rotation of the axis assembly is important; since the rotation order changes, it leads to a different result (Pamadi 2015).

The inertial reference system is $\mathbf{I} = (x_i, y_i, z_i)$ and the body reference system is $\mathbf{B} = (x_b, y_b, z_b)$. So the following is obtained if $\mathbf{I}$ reference system is rotated around $\mathbf{B}$ reference system with angles ψ, θ, and ϕ sequentially as shown in Fig. 8.1.
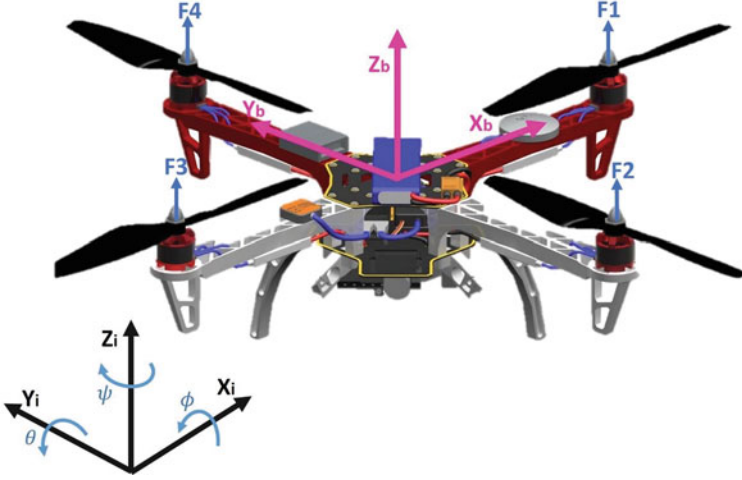
Fig. 8.1 Quadrotor coordinate system

$$R_3(\psi) = [\cos\psi \ \sin\psi \ 0 - \sin\psi \ \cos\psi \ 0 \ 0 \ 0 \ 1] \tag{8.1}$$

$$R_2(\theta) = [\cos\theta \ 0 - \sin\theta \ 0 \ 1 \ 0 \ \sin\theta \ 0 \ \cos\theta] \tag{8.2}$$

$$R_1(\phi) = [1 \ 0 \ 0 \ 0 \ \cos\phi \ \sin\phi \ 0 - \sin\phi \ \cos\phi] \tag{8.3}$$

These matrices are orthogonal matrices and multiplication of the matrices form the following transformation matrix, which moves any vector from **I** reference system to **B** reference system.

$$T_i^b = R_1(\phi)R_2(\theta)R_3(\psi) \tag{8.4}$$

$$T_i^b = [\cos\theta\cos\psi \ \cos\theta\sin\psi - \sin\theta \ \sin\theta\sin\phi\cos\psi$$
$$- \sin\psi\cos\phi \ \sin\psi\sin\theta\sin\phi + \cos\psi\cos\phi \ \sin\phi\cos\theta \ \sin\theta\cos\phi\cos\psi$$
$$+ \sin\psi\sin\phi \ \sin\psi\sin\theta\cos\phi - \cos\psi\sin\phi \ \cos\phi\cos\theta]$$
$$\tag{8.5}$$

It is not possible to measure Euler angles directly in the flight control system. However, P, Q, and R, which are the angular velocities around three axes in the body reference system, can be measured. It is possible to obtain Euler angular velocities by applying a transformation to these angular velocities (Corke 2013).

To move Euler angular velocities from **I** reference system to **B** reference system, the following matrix calculations are performed.

$$[P \ Q \ R] = R_1(\phi)R_2(\theta)[0 \ 0 \ \dot\psi] + R_1(\phi)[0 \ \dot\theta \ 0] + [\dot\phi \ 0 \ 0] \tag{8.6}$$

$[P\,Q\,R\,]$
$$= [\cos\theta\ 0 - \sin\theta\ \ \sin\phi\sin\theta\ \cos\phi\ \ \sin\phi\cos\theta\ \cos\phi\sin\theta - \sin\phi\ \ \cos\phi\cos\theta]$$
$$\times [0\ 0\ \dot{\psi}\,] + [1\ 0\ 0\ 0\ \cos\phi\ \ \sin\phi\ 0 - \sin\phi\ \ \cos\phi]\big[0\ \dot{\theta}\ 0\,\big] + \big[\dot{\phi}\ 0\ 0\,\big] \tag{8.7}$$

$$[P\,Q\,R\,] = [1\ 0 - \sin\theta\ 0\ \cos\phi\ \ \sin\phi\cos\theta\ 0 - \sin\phi\ \ \cos\phi\cos\theta]\big[\dot{\phi}\ \dot{\theta}\ \dot{\psi}\,\big] \tag{8.8}$$

However, P, Q, and R are generally given, and Euler angular velocities are desired. Therefore, **L** matrix is defined, and Euler angular velocities can be found using its inverse.

$$L = [1\ 0 - \sin\theta\ 0\ \cos\phi\ \ \sin\phi\cos\theta\ 0 - \sin\phi\ \ \cos\phi\cos\theta\,] \tag{8.9}$$

$$\big[\dot{\phi}\ \dot{\theta}\ \dot{\psi}\,\big] = L^{-1}[P\,Q\,R\,] \tag{8.10}$$

$\big[\dot{\phi}\ \dot{\theta}\ \dot{\psi}\,\big]$
$$= [1\ \ \sin\phi\tan\theta\ \ \cos\phi\tan\theta\ 0\ \ \cos\phi - \sin\phi\ 0\ \ \sin\phi\sec\theta\ \ \cos\phi\sec\theta\,]$$
$$\times [P\,Q\,R\,] \tag{8.11}$$

8.2.2   Equations of Motion

The following assumptions are made while gathering the equations of motion of the quadrotor (Musa 2017).

- Quadrotor is a rigid body.
- The center of gravity of the quadrotor is coincident with the center of the body reference frame.
- The ground effect of the quadrotor is neglected.
- Earth is an inertial reference.

To derive equations of motion according to the Newton–Euler method, it is essential to define forces and torques acting on a quadrotor (He and Zhao 2014). There are four control inputs of a quadrotor, which has six degrees of freedom.

$$u_\phi = l(F_4 - F_2) \tag{8.12}$$

$$u_\theta = l(F_3 - F_1) \tag{8.13}$$

$$u_\psi = d(F_1 - F_2 + F_3 - F_4) \tag{8.14}$$

where l is the distance (m) between the center of gravity of the quadrotor and the center of the propeller, and d is the ratio between the drag and the thrust coefficients of the propeller.

The net moment on the quadrotor is the sum of the moments resulting from the speed differences of the propellers and rotational drag.

$$M_{\text{net}} = M_{\text{quad}} + M_k \tag{8.15}$$

The net moment acting on the quadrotor is equal to the change in angular momentum.

$$M_{\text{net}} = I\frac{d}{dt}(w) + w \times (Iw) \tag{8.16}$$

I_x, I_y, and I_z are moments of inertia, and since the quadrotor is symmetrical along the x and y axes, the product of inertia is zero.

$$M_{\text{net}} = \left[I_x\dot{P} \; I_y\dot{Q} \; I_z\dot{R}\right] + \left[(I_z - I_y)QR \; (I_x - I_z)RP \; (I_y - I_x)PQ\right] \tag{8.17}$$

Roll, pitch, and yaw inputs are part of the moments on the quadrotor.

$$M_k = [l(F_4 - F_2) \; l(F_3 - F_1) \; d(F_1 - F_2 + F_3 - F_4)] = \left[u_\phi \; u_\theta \; u_\psi\right] \tag{8.18}$$

If $\mathbf{M}_{\text{quad}}$ and $\mathbf{M}_k$ are summed and arranged, the following equation is obtained:

$$\left[\dot{P} \; \dot{Q} \; \dot{R}\right] = \left[\frac{(I_y - I_z)}{I_x}QR \; \frac{(I_z - I_x)}{I_y}RP \; \frac{(I_x - I_y)}{I_z}PQ\right] + \left[\frac{u_\phi}{I_x} \; \frac{u_\theta}{I_y} \; \frac{u_\psi}{I_z}\right] - \left[\frac{k_rP}{I_x} \; \frac{k_rQ}{I_y} \; \frac{k_rR}{I_z}\right] \tag{8.19}$$

where k_r is the rotational drag coefficient.

A summary of the equations of motion for the rotational subsystem of the quadrotor is given below:

$$\dot{P} = \frac{1}{I_x}\left[(I_y - I_z)QR + u_\phi - k_rP\right] \tag{8.20}$$

$$\dot{Q} = \frac{1}{I_y}[(I_z - I_x)RP + u_\theta - k_rQ] \tag{8.21}$$

$$\dot{R} = \frac{1}{I_z}\left[(I_x - I_y)PQ + u_\psi - k_rR\right] \tag{8.22}$$

$$\dot{\phi} = P + Q\sin\phi\tan\theta + R\cos\phi\tan\theta \tag{8.23}$$

$$\dot{\theta} = Q\cos\phi - R\sin\phi \tag{8.24}$$

$$\dot{\psi} = Q\sin\phi\sec\theta + R\cos\phi\sec\theta \tag{8.25}$$

8.3   Scheme of the Proposed Method

8.3.1   Input–Output Linearization

We consider the following nonlinear system with n states, m inputs, and m outputs (Freddi 2012):

$$\dot{x} = f(x) + g(x)u = f(x) + \sum_{i=1}^{m} g_j x u_j \qquad (8.26)$$

$$y = [h_1(x), \ldots, h_m(x)]^T \qquad (8.27)$$

where $x \in R^n$ is the state vector, $u \in R^m$ is the control input, $y \in R^m$ is the output vector, and $f(x)$, $g_1(x), \ldots, g_m(x)$, y are smooth vector fields described on an open subset of R^m.

If the system is single input single output (SISO) ($m = 1$), it has a relative degree ρ, $1 \leq \rho \leq n$ in a region $D_0 \subset D$ if

$$L_g L_f^{i-1} h(x) = 0, \quad i = 1, 2, \ldots, \rho - 1, L_g L_f^{\rho-1} h(x) \neq 0 \quad \text{for all } x \epsilon D_0 \qquad (8.28)$$

If the system is multiple input multiple output (MIMO) ($m \neq 1$), the relative degree definition changes, and the system is said to have a vector relative degree $\{r_1, \ldots, r_m\}$ at a point x^0 if:

- The following condition holds for all $1 \leq j \leq m$, for all $k \leq r_i - 1$, for all $1 \leq i \leq m$, and for all x in the neighborhood of x^0 (Freddi 2012)

$$L_{g_j} L_f^k h_i(x) = 0 \qquad (8.29)$$

- The following matrix is nonsingular at $x = x^0$:

$$M(x) = \left[L_{g_1} L_f^{r_1-1} h_1(x) \cdots L_{g_m} L_f^{r_1-1} h_1(x) L_{g_1} L_f^{r_2-1} h_2(x) \ldots L_{g_m} L_f^{r_2-1} h_2(x) \vdots \ldots \vdots L_{g_1} L_f^{r_m-1} h_m(x) \cdots L_{g_m} L_f^{r_m-1} h_m(x) \right] \qquad (8.30)$$

The nonlinear system above must hold the conditions below:

- Sum of elements of the vector relative degree $\{r_1, \ldots, r_m\}$ must be equal to n.
- The matrix $M(x)$ is invertible.

If these conditions are satisfied, a linearizing control law can be found as follows:

$$u_c(x, w) = \alpha_c(x) + \beta_c(x)w \qquad (8.31)$$

where

$$\alpha_c(x) = -M^{-1}(x)\left[L_f^{r_1}h_1(x)\ldots L_f^{r_m}h_m(x)\right]^{\mathrm{T}} \tag{8.32}$$

$$\beta_c = M^{-1}(x) \tag{8.33}$$

A diffeomorphism can also be defined as:

$$x_c = \phi_c(x) \tag{8.34}$$

where

$$\phi_c^{\mathrm{T}} = \left[\phi_{c_1}^{\mathrm{T}}(x)\ldots\phi_{c_m}^{\mathrm{T}}(x)\right] \tag{8.35}$$

$$\phi_{c_i}^{\mathrm{T}} = \left[h_i(x)\ L_f h_i(x)\ldots L_f^{r_i-1}h_i(x)\right] \tag{8.36}$$

A linear system can be obtained if the diffeomorphism is applied to the nonlinear system.

$$\dot{x} = A_c x_c + B_c w \tag{8.37}$$

where $\mathbf{w}$ is linear control law and $\mathbf{A_c}$ and $\mathbf{B_c}$ represent the canonical Brunowski form.

$$A_c = \mathrm{diag}(A_1\ldots A_m) \tag{8.38}$$

$$B_c = \mathrm{diag}(b_1\ldots b_m) \tag{8.39}$$

where

$$A_i = [0\ 1\ 0\cdots 0\ 0\ 0\ 1\ldots 0\ldots\cdots 0\ 0\ 0\ldots 1\ 0\ 0\ 0\ldots 0\,] \tag{8.40}$$

$$b_i = [0\ldots 1\,] \tag{8.41}$$

8.3.2 Control System Structure

Control system consists of two control loops: the inner loop is feedback lineariza-tion, and the outer loop is LQR. State variables are selected as follows:

$$x = [x_1\ x_2\ x_3\ x_4\ x_5\ x_6] = [\phi\ \theta\ \psi\ P\ Q\ R] \tag{8.42}$$

x_1, x_2, x_3, x_4, x_5, and x_6 are chosen for feedback linearization. The nonlinear system is represented by:

$$x = [x_1 \ x_2 \ x_3 \ x_4 \ x_5 \ x_6] = [\phi \ \theta \ \psi \ P \ Q \ R] \tag{8.43}$$

$$y = [h_1 \ h_2 \ h_3]^{\mathrm{T}} = [x_1 \ x_2 \ x_3]^{\mathrm{T}} \tag{8.44}$$

where

$f(x)$

$$
\begin{aligned}
&[x_4 + x_5\mathrm{sinsin}\ x_1 \ \mathrm{tantan}\ x_2 + x_6\mathrm{coscos}\ x_1 \ \mathrm{tantan}\ x_2 \ x_5\mathrm{coscos}\ x_1 \\
&= - x_6\mathrm{sinsin}\ x_1 (x_5\mathrm{sinsin}\ x_1 + x_6\mathrm{coscos}\ x_1)/\mathrm{coscos}\ x_2 (-k_r x_4 - x_5 x_6 (I_z - I_y)) \\
&/I_x(-k_r x_5 - x_4 x_6 (I_x - I_z))/I_y(-k_r x_6)/I_z]
\end{aligned}
\tag{8.45}
$$

and

$$g(x) = \left[0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ \frac{1}{I_x}\ 0\ 0\ 0\ \frac{1}{I_y}\ 0\ 0\ 0\ \frac{1}{I_z}\right] \tag{8.46}$$

The summation of relative degrees ($r_1 = 2$, $r_2 = 2$, and $r_3 = 2$) equals to the vector relative degree ($r = 6$) that is equal to the number of states. The matrix $\mathbf{M}(x)$ that is nonsingular is calculated as follows:

$$M(x) = \left[\frac{1}{I_x}\ \frac{\mathrm{sinsin}\ x_1 \ \mathrm{tantan}\ x_2}{I_y}\ \frac{\mathrm{coscos}\ x_1 \ \mathrm{tantan}\ x_2}{I_z}\ 0\ \frac{\mathrm{coscos}\ x_1}{I_y}\ \frac{-\mathrm{sinsin}\ x_1}{I_z}\ 0\ \frac{\mathrm{sinsin}\ x_1}{I_y\mathrm{coscos}\ x_2}\ \frac{\mathrm{coscos}\ x_1}{I_z\mathrm{coscos}\ x_2}\right] \tag{8.47}$$

So, two conditions required for feedback linearization are satisfied. Diffeomorphism converts the state variables in the nonlinear system into the state variables of the new system that is linearized in the feedback linearization method.

$\phi_c(x)$

$$
\begin{aligned}
&[x_1 \ x_4 + x_6\mathrm{coscos}\ x_1 \ \mathrm{tantan}\ x_2 + x_5\mathrm{sinsin}\ x_1 \ \mathrm{tantan}\ x_2 \ x_2 \ x_5\mathrm{coscos}\ x_1 \\
&= - x_6\mathrm{sinsin}\ x_1 \ x_3 \frac{(x_5\mathrm{coscos}\ x_1 + x_5\mathrm{sinsin}\ x_1)}{\mathrm{coscos}\ x_2}]
\end{aligned}
\tag{8.48}
$$

The feedback linearizing law is computed as:

$$u_c(x, w) = \alpha_c(x) + \beta_c(x)w \tag{8.49}$$

where

$$\alpha_c(x) = \left[k_r x_4 + I_z x_5 x_6 - \frac{I_x x_5^2 \sin\sin 2x_1}{2} + \frac{I_x x_6^2 \sin\sin 2x_1}{2} \right.$$

$$- 2I_x x_5 x_6 \cos\cos x_1{}^2 \frac{k_r x_5 \cos\cos x_2 - I_y x_5^2 \cos\cos x_1 \ \sin\sin x_2 + 2I_y \cos x_6^2 \cos \ x_1 \ \sin\sin x_2 + I_x x_4 x_6 \cos\cos x_2 + a}{\cos\cos x_2}$$

$$\left. - \frac{(I_z x_5^2 \sin\sin x_1 \ \sin\sin x_2 - k_r x_6 \cos\cos x_2 + I_z x_4 x_5 \cos\cos x_2 + b)}{\cos\cos x_2} \right]$$

$$(8.50)$$

$$a = I_y x_4 x_6 \cos\cos x_2 - I_z x_4 x_6 \cos\cos x_2 + I_y x_5^2 \cos\cos x_1{}^3 \sin\sin x_2$$

$$- I_y x_6^2 \cos\cos x_1{}^3 \sin\sin x_2 \tag{8.51}$$

$$b = I_z x_5^2 \cos\cos x_1{}^2 \sin\sin x_1 \ \sin\sin x_2 - I_z x_6^2 \cos\cos x_1{}^2 \sin\sin x_1 \ \sin\sin x_2$$

$$+ 2I_z x_5 x_6 \cos\cos x_1{}^3 \sin\sin x_2 \tag{8.52}$$

$$\beta_c(x) = \begin{bmatrix} I_x \ 0 - I_x \sin\sin x_2 \ 0 \ I_y \cos\cos x_1 \ I_y \cos\cos x_2 \ \sin\sin x_1 \ 0 \\ - I_z \sin\sin x_1 \ I_z \cos\cos x_1 \ \cos\cos x_2 \end{bmatrix} \tag{8.53}$$

If the control law is applied, the following linearized system is obtained:

$$\dot{x}_c = A_c x_c + B_c w$$
$$= [0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ ] x_c$$
$$+ [0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ ] w$$

$$(8.54)$$

Since the system is now linear, a linear control method can be applied. Therefore, linear quadratic regulator (LQR) is preferred because it has easy design and application for linear control of the system. A control law:

$$w = -Kx_c \tag{8.55}$$

is designed by selecting an appropriate K value. This is achieved by choosing Q and R matrices and using MATLAB command lqr, where Q and R matrices are given as follows:

$$Q = [100\ 0\ 0\ 0\ 0\ 0\ 0\ 100\ 0\ 0\ 0\ 0\ 0\ 0\ 100\ 0\ 0\ 0\ 0\ 0\ 0\ 100\ 0\ 0\ 0\ 0\ 0\ 0\ 100\ 0\ 0\ 0\ 0\ 0\ 0\ 100]$$

$$(8.56)$$

$$R = [0.01\ 0\ 0\ 0\ 0.01\ 0\ 0\ 0\ 0.01\ ] \tag{8.57}$$

In the SISO case, if the dimension of the nonlinear system is not equal to the relative degree, zero dynamics appear. So, the stability of the system depends on the
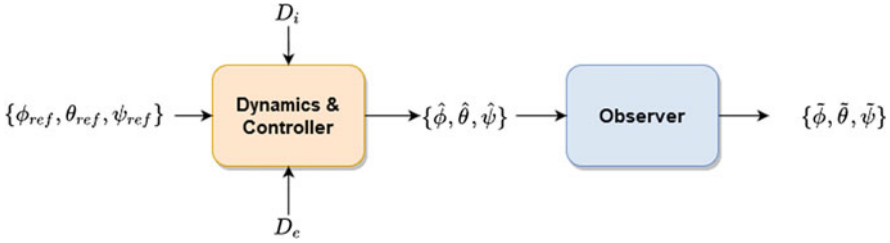
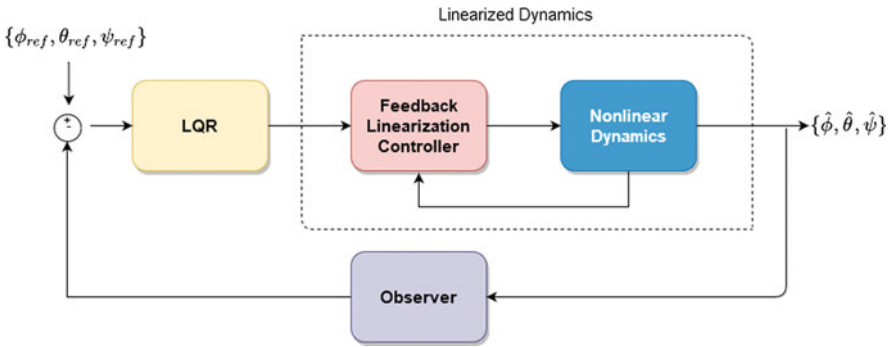**Fig. 8.2** General diagram of the proposed scheme

Fig. 8.3 Block diagram of the controller structure

stability of the zero dynamics, but this condition is not valid for MIMO systems. In addition, there are no zero dynamics that appear into the system since relative degree is equal to the dimension of the system. So, since our model is MIMO, it is globally asymptotically stable. Sufficient conditions for the system to be stable can be found in (Seborg and Henson 1996).

The general diagram of the proposed observer-based feedback controller is given in Fig. 8.2. $\{\phi_{ref}, \theta_{ref}, \psi_{ref}\}$ is the reference attitude input. The proposed scheme is based on the quadrotor attitude dynamics given in Eqs. (8.20) and (8.25). The internal disturbance D_i is added to the model since the physical and mathematical modeling of the quadrotor contains unknown errors and uncertainty. The external disturbance D_e is also added due to environmental disturbances and sensor errors. $\{\hat{\phi}, \hat{\theta}, \hat{\psi}\}$ is the set of outputs that contain noise terms. Lastly, $\{\tilde{\phi}, \tilde{\theta}, \tilde{\psi}\}$ is the set of estimated attitude signals of the quadrotor.

The sub-diagram of the dynamics and controller block is given in Fig. 8.3. Nonlinear dynamics are linearized by the feedback linearization controller. Then, LQR controller is designed using the linearized dynamics. The sensor produces noisy $\{\phi, \theta, \psi\}$ states, and reference states are subtracted from these noisy states. The error signal is fed to the LQR controller, which computes the control signal of the linearized system. The feedback linearization controller calculates roll, pitch, and yaw inputs according to LQR output. Nonlinear dynamics generate a state vector that goes to feedback linearization and observer.

Fig. 8.4 Flowchart of the
proposed controller
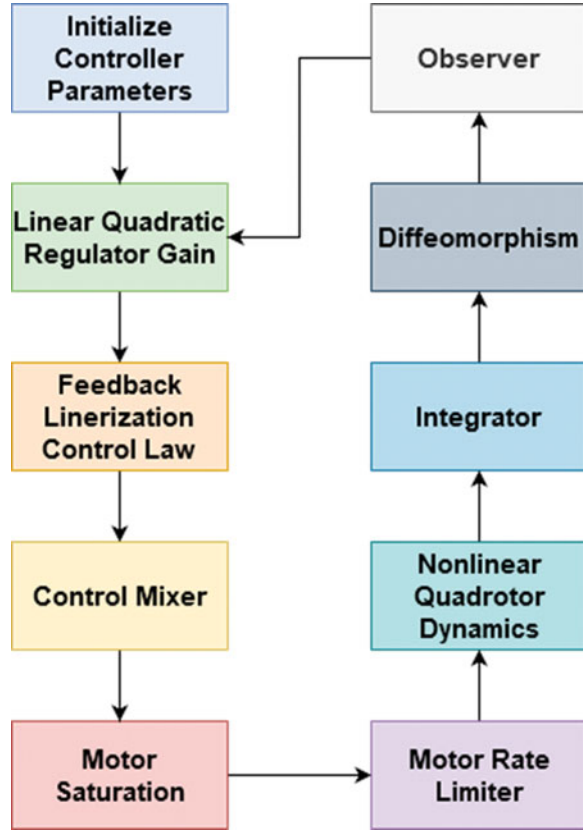
The observer employed in this method is chosen as particle filter. Particle filter is a Monte Carlo–based iterative importance sampling method. Particle filter finds a suboptimal solution to the Bayesian estimation. This suboptimal solution is based on the approximation of the integral terms in the probability density functions. The density functions mentioned before are represented by a vector of random particles and weights. If the number of particle size increases, representation converges to the optimal solution. Particle filter–based scheme is chosen as observer since particle filter has nonlinear framework and can deal with both Gaussian and non-Gaussian distributed noises. Thus, the performance of the proposed controller can be increased and enhanced in real-world scenarios.

The flowchart shown in Fig. 8.4 demonstrates the flow of the proposed controller. First, controller parameters are initialized. Then the loop starts with LQR controller gain. The output of the diffeomorphism is multiplied by LQR gain, and the gain is fed to feedback linearization law. The control mixer produces motor (1–4) inputs from roll, pitch, and yaw inputs. Motor dynamics consist of motor saturation and motor rate limiter. The motor saturation limits motor thrust between 0 N and 5 N. The motor rate limiter restricts the motor rate of thrust by ±50 N/s. Nonlinear

quadrotor dynamics generate a derivative of the states using motor (1–4) outputs. The integrator integrates the derivative of the state vector to obtain the states to be used in the control system. Diffeomorphism calculates the linearized system state vector that is observed and fed to LQR gain. The loop continues during the simulation time in this way.

8.4   Results and Discussion

The proposed observer-based nonlinear control of the quadrotor is implemented in both MATLAB and Simulink environments. Simulations are carried out for 20 [s] with a discrete time step of 0.1 [s]. The parameters of the nonlinear quadrotor model are given in Table 8.1 (Lanzon et al. 2014), where m is the weight, g is the gravity, (I_x, I_y, I_z) are the moments of inertia, l is the length of the arm, d is the drag coefficient of the propellers, and (k_r, k_t) are rotational and translational drag coefficients. The attitude response of the feedback controller is tested from negative initial angles to positive reference angles for each attitude angle. The initial state vectors are set as [0, 0, 0], [$-\pi/4$, $-\pi/8$, $-\pi/12$] and reference vectors are set as [$\pi/6$, $\pi/4$, $\pi/3$], [0, 0, 0] for first and second scenarios where states are [ϕ, θ, ψ] in [rad], respectively.

8.4.1   Noisy Reference Tracking

Since quadrotors are prone to low-cost and unstable sensor errors, it is an essential task to support the controller with an observer/estimator block. Thus, sensor errors can be eliminated through the observer scheme. To evaluate the sensor noise performance of the proposed method, an internal noise source with a variance of 0.01 [rad^2] and an external sensor noise with a variance of 0.02 [rad^2] are injected to the attitude states of the quadrotor as disturbances. The disturbances are modeled as Gaussian random variable with zero mean and a total of 0.03 [rad^2] variance component. The measured overall variance values of the states are 0.0302 for ϕ,

Table 8.1 Quadrotor parameters

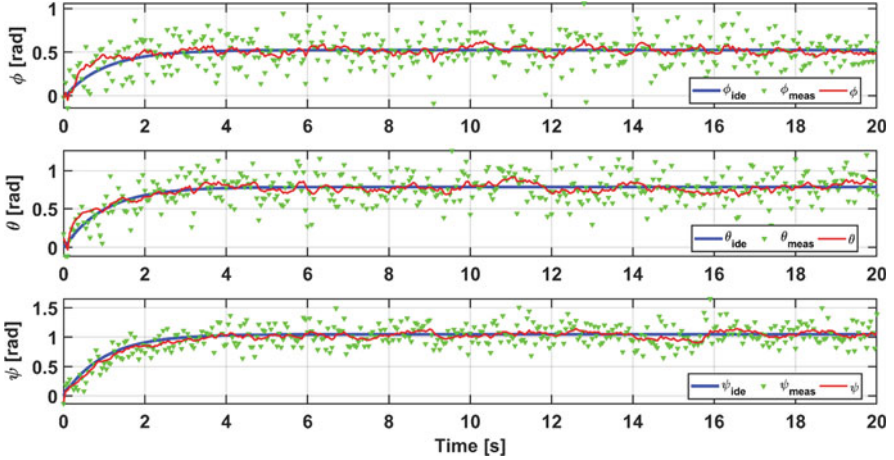| Variable	Value	Unit
m	0.5	[kg]
g	9.81	[m/s^2]
I_x	5.90E$-$03	[kg.m^2]
I_y	5.90E$-$03	[kg.m^2]
I_z	1.16E$-$03	[kg.m^2]
l	2.55E$-$01	[m]
d	2.40E$-$03	[kg.m^2]
kr	1.00E$-$02	[kg.m^2/s]
kt	1.00E$-$02	[kg/s]

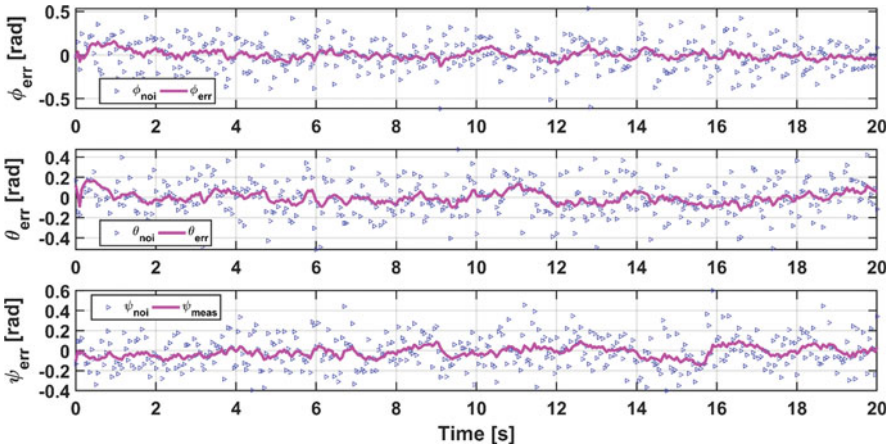**Fig. 8.5** Noisy response of the proposed controller: from 0 [rad] to arbitrary states

Fig. 8.6 A comparison of sensor noise and error of the proposed method: from 0 [rad] to arbitrary states

0.0327 for θ, and 0.0283 for ψ. The responses of the controller for the first scenario are given in Fig. 8.5 and the error plot is given in Fig. 8.6. The responses of the controller for the second trajectory is given in Fig. 8.7 and the error plot is given in Fig. 8.8. In Fig. 8.5, green points show the measurement of the states subjected to sensor noise, the blue line shows the ideal controller response without any noise component, and, lastly, the red line shows the response of the proposed method under the sensor noise. A comparative analysis of the performance of the proposed method and total noise is seen in Fig. 8.6 where the magenta line denotes the error of the method and blue points show the total error of internal and external disturbances.
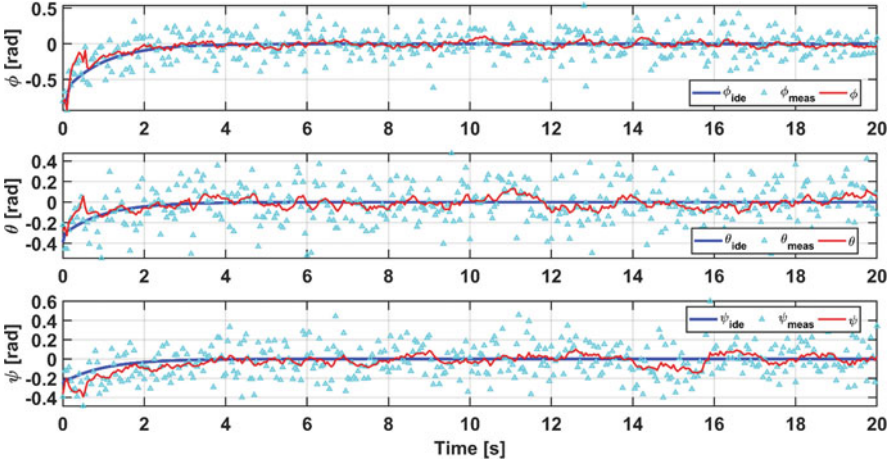
Fig. 8.7 Noisy response of the proposed controller: from arbitrary states to 0 [rad]
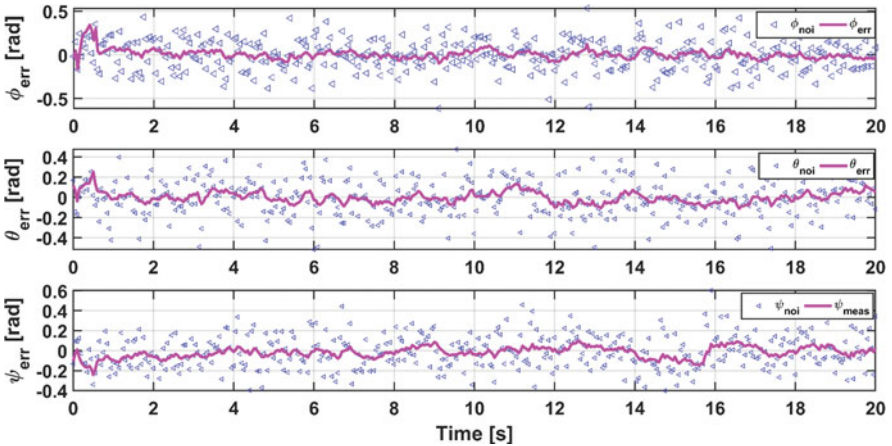

Fig. 8.8 A comparison of sensor noise and error of the proposed method: from arbitrary states to 0 [rad]

The notation is similar for Fig. 8.7 and Fig. 8.8 except sensor noise is given with cyan points for the second scenario. It can be clearly deduced from Figs. 8.5, 8.6, 8.7 and 8.8 that the proposed method is well performed and tracks the reference trajectory even under the sensor noise with a low error ratio for all states.

The statistical analyses of the proposed method under sensor noise are given in Tables 8.2, 8.3, and 8.4 for standard deviation, signal-to-noise ratio (SNR), and absolute error, respectively. The performance of the proposed method is compared against the total noise to show the enhancements of the algorithm. Standard deviations of the errors for all states are seen in Table 8.2 for both trajectories. The

Table 8.2 Standard deviation analysis of the proposed method

	Noise			Method		
	ϕ	Θ	ψ	ϕ	θ	ψ
First trajectory	0.173971	0.181029	0.168277	0.046902	0.055575	0.049403
Second trajectory	0.173971	0.181029	0.168277	0.054185	0.054281	0.054268

Table 8.3 Signal-to-noise ratio analysis of the proposed method

	Noise			Method		
	ϕ	θ	ψ	ϕ	Θ	ψ
First trajectory	9.6586	12.5690	15.5428	20.6373	22.6393	25.5759
Second trajectory	1.3092	0.3413	0.4372	5.7035	1.1488	3.3781

Table 8.4 Absolute error analysis of the proposed method

	Noise			Method		
	ϕ	θ	ψ	ϕ	Θ	ψ
First trajectory	0.6141	0.5173	0.6012	0.1573	0.1710	0.1452
Second trajectory	0.6141	0.5173	0.6012	0.3421	0.2534	0.2428

proposed method is capable of controlling and reducing sensor errors. The standard deviation of the error for the attitude angles are reduced from nearly 0.17 [rad] to nearly 0.05 [rad]. The standard deviation performance of the proposed method is no more than 0.056 [rad], which is approximately reduced three times with respect to the total error. The signal-to-noise ratio analysis of the algorithm is given in Table 8.3. The method is capable of increasing the SNR for all states and trajectories. The SNR is increased as high as 25.5 with the lowest increment of 0.34 to 1.14. Lastly, the absolute error performance of the algorithm is given in Table 8.4. According to the results, the length of the error is reduced from nearly 0.6 [rad] up to 0.145 [rad] with a reduction of four times.

8.5 Conclusion

In this work, an observer-based feedback linearization controller for a quadrotor is proposed to control the nonlinear attitude dynamics subjected to sensor noise. The proposed method is tested for noisy references. Two scenarios are considered: from 0 [rad] to any arbitrary state and from any arbitrary state to 0 [rad] to cover more circumstances. According to the results, the proposed method is capable of tracking the given trajectories for noisy references. Numerical analysis showed that the maximum standard deviation of the error of the proposed method is reduced by three times, signal-to-noise ratio is increased by seven times, and the absolute error is reduced by four times.

The SNR is increased from 0.34 to 1.14, absolute error is reduced from 0.6 [rad] up to 0.145 [rad], and standard deviation is reduced from 0.17 [rad] to 0.05 [rad]. According to the abovementioned results, it can be easily concluded that the proposed controller has good control, estimation, and reference tracking responses regarding internal and external noise sources with a low standard deviation of error, absolute error, and high signal-to-noise ratio statistics, respectively.

References

Ai X, Yu J (2019) Fixed-time trajectory tracking for a quadrotor with external disturbances: a flatness-based sliding mode control approach. Aerosp Sci Tech 89:58–76

Al-Hiddabi S (2009) Quadrotor control using feedback linearization with dynamic extension. In: Proceeding of the 6th international symposium on mechatronics and its applications, Sharjah

Araar O, Aouf N (2014) Full linear control of a quadrotor UAV, LQ vs H∞. In: UKACC international conference on control, Loughborough

Basri MAM, Husain AR, Danapalasingam KA (2015) Nonlinear control of an autonomous quadrotor unmanned aerial vehicle using Backstepping controller optimized by particle swarm optimization. J Eng Sci Technol 8(3):39–45

Bonna R, Camino JF (2015) Trajectory tracking control of a quadrotor using feedback linearization, Proceedings of the XVII international symposium on dynamic problems of mechanics, Brazil

Chen W, Yang J, Guo L, Li S (2016) Disturbance-observer-based control and related methods-an overview. IEEE Trans Ind Electron 63(2):1083–1095

Choi Y-C, Ahn H-S (2014) Nonlinear control of quadrotor for point tracking: actual implementation and experimental tests. IEEE/ASME Trans Mechatron 20(3):1179–1192

Corke P (2013) Robotics, vision and control: fundamental algorithms in MATLAB. Springer-Verlag Berlin Heidelberg, Berlin

Daroogheh N, Meskin N, Khorasani K (2018) A dual particle filter-based fault diagnosis scheme for nonlinear systems. IEEE Trans Control Syst Technol 26(4):1317–1334. https://doi.org/10.1109/TCST.2017.2704959

Eltayeb A, Rahmat MF, Musa MJ (2019) Feedback linearization and sliding mode control design for quadrotor's attitude and altitude. In: Proceedings of the IEEE 1st international conference on mechatronics, automation and cyber-physical computer system

Ermeydan A, Kiyak E (2017) Fault tolerant control against actuator faults based on enhanced PID controller for a quadrotor. Aircraft Eng Aerospace Tech 89(3):468–476

Freddi A (2012) Model-based diagnosis and control of unmanned aerial vehicles: application to the quadrotor system. Universita Politechnica Delle Marche, Ancona

Gee SS, Lee TH, Tan EG (1998) Adaptive neural network control of flexible joint robots based on feedback linearisation. Int J Syst Sci 29:623–635

Ghandour J, Aberkane S, Ponsart J-C (2014) Feedback linearization approach for standard and fault tolerant control: application to a quadrotor UAV testbed. J Phys Conf Ser 570 pp. 1–11

He Z, Zhao L (2014) A simple attitude control of quadrotor helicopter based on Ziegler-Nichols rules for tuning PD parameters. Sci World J 2014:743869. https://doi.org/10.1155/2014/743869

Kendoul F (2012) Survey of advances in guidance, navigation, and control of unmanned rotorcraft systems. J Field Robotics 29(2):315–378

Khatoon S, Gupta D, Das LK (2014) PID & LQR control for a quadrotor: modeling and simulation. In: 2014 international conference on advances in computing, communications and informatics (ICACCI), New Delhi

Kıyak E (2016) Tuning of controller for an aircraft flight control system based on particle swarm optimization. Aircraft Eng Aerospace Tech 88(6):799–809

Köksal N, An H, Fidan B (2016) Two-level nonlinear tracking control of a quadrotor unmanned aerial vehicle. IFAC-PapersOnLine 49(17):254–259

Kose O, Oktay T (2019) Dynamic modeling and simulation of quadrotor for different flight. Eur J Sci Technol 15:132–142. https://doi.org/10.31590/ejosat.546867

Kuantama E, Tarca I, Tarca R (2018) Feedback linearization LQR control for quadcopter position tracking. In: Proceedings of the 5th international conference on control, decision and information technologies, Thessaloniki

Labbadi M, Cherkaoui M (2019) Robust adaptive backstepping fast terminal sliding mode controller for uncertain quadrotor UAV. Aerosp Sci Technol 93

Lanzon A, Freddi A, Longhi S (2014) Flight control of a quadrotor vehicle subsequent to a rotor failure. J Guid Control Dyn 37(2):580–591. https://doi.org/10.2514/1.61494

Metin EY, Aygün H (2019) Energy and power aspects of an experimental target drone engine at non-linear controller loads. Energy 185:981–993

Musa S (2017) Techniques for quadcopter modeling and design: a review. J Unmann Syst Technol 5(3):66–75. https://doi.org/10.21608/just.2017.543

Najm AA, Ibraheem IK (2019) Nonlinear PID controller design for a 6-DOF UAV quadrotor system. Eng Sci Technol 22(4):1087–1097

Oh H, Kim S (2019) Persistent standoff tracking guidance using constrained particle filter for multiple UAVs. Aerosp Sci Technol 84:257–264. https://doi.org/10.1016/j.ast.2018.12.022

Özbek NS, Önkol M, Efe MÖ (2016) Feedback control strategies for quadrotor-type aerial robots: a survey. Trans Inst Meas Control 38(5):529–554

Pamadi BN (2015) Performance, stability, dynamics and control of airplanes. AIAA Education Series, Reston

Prabhakaran B, Kothari M, Abhishek (2015) Nonlinear control design for quadrotors, 2015 IEEE Workshop on Computational Intelligence: Theories, Applications and Future Directions (WCI), Kanpur, pp. 1–6

Razmi H, Afshinfar S (2019) Neural network-based adaptive sliding mode control design for position and attitude control of a quadrotor UAV. Aerosp Sci Technol 91:12–27

Saif A-WA (2017) Feedback linearization control of quadrotor with tiltable rotors under wind gusts. Int J Adv Appl Sci 4(10):150–159

Seborg DE, Henson MA (1996) Nonlinear process control. Prentice Hall, Upper Saddle River

Voos H (2009) Nonlinear control of a quadrotor micro-UAV using feedback-linearization. In: Proceedings of the 2009 IEEE international conference on mechatronics, Malaga

Wang X, Kampen E-J, Chu Q, Lu P (2019) Incremental sliding-mode fault-tolerant flight control. J Guidance Control Dyn 42(2)

Xiong J-J, Zhang G-B (2017) Global fast dynamic terminal sliding mode control for a quadrotor UAV. ISA Trans 66:233–240

Xuan-Mung N, Hong S-K (2019) Improved altitude control algorithm for quadcopter unmanned aerial vehicles. Appl Sci 9(10)

Yazar I, Kıyak E, Çalışkan F, Karakoç TH (2018) Simulation-based dynamic model and speed controller design of a small-scale turbojet engine. Aircraft Eng Aerospace Tech 90(2):351–358

Zheng E-H, Xiong J-J, Luo J-L (2014) Second order sliding mode control for a quadrotor UAV. ISA Trans 53(4):1350–1356

Zuo Z (2010) Trajectory tracking control design with command-filtered compensation for a quadrotor. IET Control Theory Appl 4(11):2343–2355