Nonlinear Dyn (2017) 89:1289–1308
DOI 10.1007/s11071-017-3516-z

ORIGINAL PAPER

Nonlinear adaptive tracking control for a small-scale unmanned helicopter using a learning algorithm with the least parameters

Bin Zhou · Zhibin Li · Zhiqiang Zheng · Shuai Tang

Received: 14 July 2016 / Accepted: 31 March 2017 / Published online: 8 April 2017

Abstract This paper puts forward a new nonlinear adaptive controller for a small-scale unmanned helicopter with unknown mass. The controller is developed under the framework of backstepping technique, with the unknown mass estimated by a novel identifier and the internal and external uncertainties approximated by radial basis function neural networks (RBFNNs). The overall closed-loop system, which consists of three parts: longitudinal–lateral subsystem, heave subsystem, and heading subsystem, is proved to be semi-globally uniformly ultimately bounded by the strict Lyapunov stability theory. Furthermore, the proposed method is more practical in actual applications with an improved online learning algorithm of the least parameters used in the RBFNNs. Finally, the effectiveness and the robustness of the proposed strategy are exhibited through two simulations compared with the classic PID method.

Keywords Nonlinear adaptive control · RBFNNs · Robust control · Small-scale unmanned helicopter

B. Zhou (✉) · Z. Li · Z. Zheng
College of Mechatronic Engineering and Automation, National University of Defense Technology, Changsha 410073, People's Republic of China
e-mail: zhoubin@nudt.edu.cn

S. Tang
Department of Machinery and Electrical Engineering, Logistical Engineering University, Chongqing 401311, People's Republic of China

1 Introduction

With the developments of integrated circuits and the advances in the computing ability of processors promoting the development of the unmanned helicopter airborne hardware system significantly, the autonomous flight technology of the unmanned helicopter has attracted more and more attention [1,9,11,14,19,28,36]. Compared with full-size manned helicopters, unmanned helicopters can be made much smaller in size, which can greatly reduce the cost of industrial production. Moreover, small-scale unmanned helicopters can replace full-size manned helicopters in carrying out potentially life-threatening tasks, such as monitoring of active volcanoes, forest fire-fighting, and topographic surveys of remote areas, completely eliminating the potential risks borne by human pilots. In addition, compared with fixed-wing unmanned aerial vehicles (UAVs), an unmanned helicopter has unique advantages. For example, it does not need any runways used to take off or land, because it has the maneuver ability of vertical takeoff and landing. Besides, it can hover in the air and move in all directions in the Cartesian space. In spite of the above-described advantages of small-scale unmanned helicopters, they are much more difficult to be controlled for autonomous flight, because they are more vulnerable to aerodynamic interferences [26].

Although the autonomous flight of a small-scale unmanned helicopter has been basically achieved [11,29,32,35,38], the performances of those controllers are relatively conservative compared to a skilled pilot. For instance, basic attitude control has been achieved by [32] and [38], but the position control is not considered in both two literatures. In addition, the controller in [32] is designed based on a linear model, which is only effective in the vicinity of the equilibrium point, and the control performance is relatively conservative in [38] because the experiment is implemented on a test bed and the effect of the test bed is disregarded without any serious analysis. Thus, it is still a challenging problem to design a nonlinear controller with high tracking accuracy and strong robustness, which should fully consider the nonlinear characteristics of the unmanned helicopter and is supposed to be comparable with skilled pilots. The small-scale unmanned helicopter is a high-order complicated model with high nonlinearity and strong coupling among different channels. How to deal with the nonlinearity and the coupling is a very critical issue in the process of controller design. Since obtaining its exact nonlinear model is very difficult, many researchers use PID control method to design flight control system for an autonomous helicopter [11,29,36]. However, the selection of control parameters mainly relies on the designers' experiences, resulting in a relatively cumbersome adjustment process. And what is worse, the robustness of the PID controller is quite weak in flight environments with external interferences, e.g., wind gusts. In addition, many controllers are designed based on the linearized model around the equilibrium point in which dozens of nonlinear characteristics are ignored to obtain simple linearization controllers [9,10,19,20]. But the linear controllers are effective only when the unmanned helicopter flies near the balanced point. The control performance may be severely degraded, even leading to the instability of the closed-loop system, when the helicopter states deviate from the equilibrium point. As a result, linearization has brought about the application conditions of the controller, and the flight envelope of the unmanned helicopter is restricted. Although combining gain scheduling technique and the linearization method based on multi operating points can expand the flight envelope of the unmanned helicopter [10] to a certain degree, it needs a number of tedious calculations and extra experiments.

In contrast, nonlinear control techniques are more suitable for control systems with large flight envelope, and plentiful research achievements have been brought out in recent years [6,7,17,24,28,33,34,37,41]. Second-order sliding mode control is adopted to alleviate the chattering phenomena of the traditional sliding model method in both [7] and [34]. In [7], the multivariable super twisting control is combined with backstepping control based on a sliding mode observer to control an unmanned helicopter. While in [34], the second-order sliding mode control method is combined with an extended state observer to tackle the tracking problem of small-scale helicopters. Unfortunately, like the most of the previous nonlinear controllers [13,24,28,33,41], the second-order sliding mode flight controller in [34] is constructed under the hypothesis of two-timescale separation, which assumes that the response of the outer-loop is much slower than that of the inner-loop. But this assumption is probably not established when the maneuver of the unmanned helicopter is large. A novel attitude kinematics representation using a rotation matrix is proposed in [31] to transform the nonlinear model into a strict feedback form for applying backstepping control without the assumption of two-timescale separation. However, neither internal uncertainties nor external uncertainties are taken into account. Considering the advantages of the NNs with the ability of online learning over the NNs trained offline [30], the researches in [4] use NNs to learn the dynamics of an UAV online while assuming full state feedback. To further remove the assumption, Dierks [5] proposes a novel virtual control scheme with a NN observer. The authors in [5,16,27,39] all adopt NNs with online learning algorithms to compensate small coupling forces and aerodynamic uncertainties under backstepping control framework. Unfortunately, so many updating laws of weight matrices may lead to the calculation explosion of the airborne computer, making the control system hard to be applied in real flights. Calvo-Rolle [3] proposes an adaptive inverse control using an online learning algorithm for system identification, and the advantage of the method is that it has the ability to deal with non-stationary environments. Unfortunately, it needs $K \times (K + J)$ (K is the number of hidden neurons and J is the number of outputs) times of iterative calculation at each sampling point to obtain the optimal weights, and the calculation is obviously not small. In [15], the researchers combine backstepping control and NNs to design a flight control system and propose a new identifier which can identify the inertial matrix asymptotically. However, the flapping dynamics of the main rotor is totally neglected, which may lead to the deterioration of control performance. Besides, the inertial matrix identifier has nine adaptive laws, which results in huge computation workload despite using an optimized updating law. Zou [42] also investigates the tracking problem of an unmanned helicopter by combining backstepping control approach with an adaptive RBFNN. Simulations show the effectiveness of the method, nevertheless, the fact that the weights of the adaptive RBFNN are directly estimated increases the computational burden greatly and hinders the implementation of the approach. To surmount this difficulty, an efficient learning algorithm with less learning parameters is introduced in [18,22,23] where not only the computation amount is reduced but also good control performances are obtained.

The purpose of this paper is to propose a robust and nonlinear adaptive backstepping controller which needs no assumptions about multi-timescale separation and has a low computational complexity so as to facilitate the practical implementation. Inspired by [18,21–23], new adaptive RBFNNs with online learning algorithms of the least parameters are designed in this paper to enhance the robustness of the flight control system of an unmanned helicopter. In order to complete the backstepping control design smoothly, the attitude kinematics representation proposed in [31] is used. And to avoid complex analytical solutions, the time derivatives of some virtual controls in this paper are calculated through command filters [8]. The main highlights of this paper are summarized as follows: First, a novel identifier is designed to estimate the mass of the unmanned helicopter, which is considered as a constant by many previous researches [31,35,37,39]. However, the mass of an unmanned helicopter decreases with the reduction of fuel in real flights. Furthermore, when an unmanned helicopter performs transport tasks, the mass changes with loading or unloading of goods in the air. The mass is a very important model parameter for the control method based on model and a good adaptive law for the mass can effectively improve the robustness of the controller. Second, unlike the prior literatures [5,16,27,39], an improved neural network learning algorithm with the least parameters is proposed to reduce computation load [21]. With the increase in the neurons in the hidden layer, the number of the updating laws proposed in [5,16,27,39] increases exponentially, resulting in a heavy computational load. For the learning algorithm of the least parameters introduced in this paper, it only needs to estimate one parameter, i.e., the square of the induced Euclidean norm of an ideal weight matrix.

The remaining parts of this paper are arranged as follows: Some preliminaries are introduced in Sect. 2. Section 3 describes the complete nonlinear model of a small-scale unmanned helicopter. The detailed design process of the nonlinear adaptive controller including the stability analysis is elaborated in Sect. 4. Two simulations and some discussions are carried out in Sect. 5. At last, Sect. 6 concludes the paper.

2 Preliminaries

2.1 Mathematical notation

In this paper, the operands $\|\cdot\|$ and $\|\cdot\|_2$ denote Euclidean norm of column vectors and induced Euclidean norm of matrices, respectively; the notation $|\cdot|$ represents absolute value for real numbers; the abbreviations $c(\cdot)$ and $s(\cdot)$ indicate the trigonometric function $\cos(\cdot)$ and $\sin(\cdot)$, respectively; $\lambda_{\min}(\cdot)$ is the minimum eigenvalue of a matrix; $\exp(\cdot)$ means an exponential function.

2.2 Modified RBFNNs

RBFNNs can approximate any nonlinear function with an arbitrary precision and have a fast learning convergence speed. A three-layered architecture of a modified RBFNN is illustrated in Fig. 1, whose output is different from the traditional RBFNN, where $\hat{\Gamma}$ is the estimation of Γ which is defined by $\Gamma = \|W\|_2^2$ and

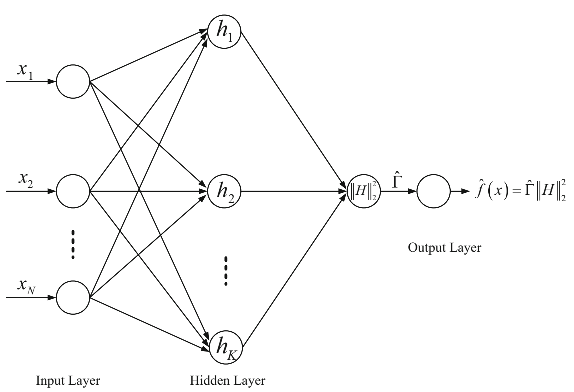


Fig. 1 Modified RBFNN structure adopted in this paper

the output is $\hat{f}(x) = \hat{\Gamma} \|H\|_2^2$. There are N, K, and M nodes in the input layer, the hidden layer, and the output layer, respectively, where N, K, and M are positive integers. And $W \in R^{K \times M}$ is the idealized weight matrix between the hidden layer and the output layer; $H(x) = [h_1(x), h_2(x), \ldots, h_K(x)]^T \in R^K$. $h_j(x)$ is the Gaussian basis function as the following form

$$h_j(x) = \exp\left(-\frac{1}{2b_j}\left\|x - c_j\right\|^2\right), \quad j = 1, 2, \ldots, K, \tag{1}$$

where $c_j = [c_{1j}, c_{2j}, \ldots, c_{Nj}]^T \in R^N$ and $b_j \in R$ are the center and the width of the radial basis function $h_j(x)$.

The advantage of this modified RBFNN is that there is only one parameter, $\hat{\Gamma}$, which needs to be updated, lessening the calculation burden dramatically. More details about the modified RBFNN learning algorithm are presented in Remarks 2 and 3.

3 Helicopter model

A nonlinear small-scale unmanned helicopter dynamics model has been developed using the first principles modeling approach, including twenty states and four inputs. The system states and the control inputs are summarized in Table 1. Two reference frames, i.e., the body frame and the inertial frame, are depicted in Fig. 2. The small-scale unmanned helicopter can be described by a hybrid nonlinear model, which is mainly composed of the following three parts: fuselage rigid body kinematics and dynamics, main rotor dynamics, and simplified yaw dynamics.

3.1 Fuselage rigid body kinematics and dynamics

The nonlinear fuselage rigid body kinematics and dynamics can be derived by Newton–Euler equation [31] as follows

$$\dot{P} = V, \tag{2}$$

$$m\dot{V} = R(\Theta)F + mgE_z, \tag{3}$$

$$\dot{R}(\Theta) = R(\Theta)Sk(\omega), \tag{4}$$

$$I\dot{\omega} = -\omega \times I\omega + M, \tag{5}$$

Table 1 Physical descriptions of states and inputs

Variable (unit)	Physical description
x, y, z (m)	Position vector along the inertial frame x-, y-, and z-axes
v_x, v_y, v_z (m/s)	Velocity vector along the inertial frame x-, y-, and z-axes
u, v, w (m/s)	Velocity vector along the body frame x-, y-, and z-axes
ϕ, θ, ψ (rad)	Roll, pitch, and yaw attitude angles in the inertial frame
p, q, r (rad/s)	Roll, pitch, and yaw angular rates in the body frame
a, b (rad)	Flapping angles of the main blade in longitudinal and lateral directions
c, d (rad)	Flapping angles of the stabilizer in longitudinal and lateral directions
$\delta_{ped,int}$	Intermediate state in the yaw dynamics
δ_{lat}	Aileron servo input
δ_{lon}	Elevator servo input
δ_{col}	Collective pitch servo input
δ_{ped}	Rudder servo input

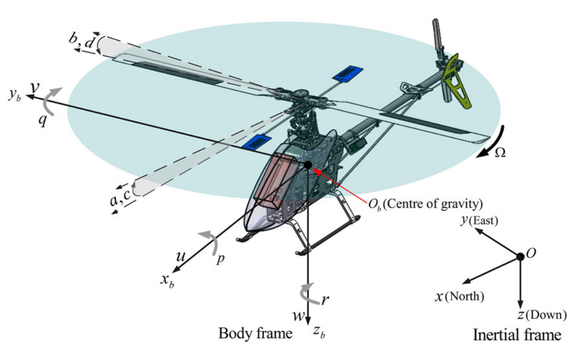


Fig. 2 Sketch of the body frame and the inertial frame

where $P = [x, y, z]^T$ and $V = [v_x, v_y, v_z]^T$ are the position vector and the linear velocity vector of the centre of gravity of the unmanned helicopter in the inertial frame, respectively; m is the mass of the helicopter; $\Theta = [\phi, \theta, \psi]^T$ denotes the attitude angle vector of the fuselage; $R(\Theta)$ indicates the transform matrix from the body frame to the inertial frame, specified by

$$R(\Theta) = \begin{bmatrix} c(\theta\psi) & s(\phi\theta)c(\psi) - c(\phi)s(\psi) & c(\phi\psi)s(\theta) + s(\phi\psi) \\ c(\theta)s(\psi) & s(\phi\theta\psi) + c(\phi\psi) & c(\phi)s(\theta\psi) - s(\phi)c(\psi) \\ -s(\theta) & s(\phi)c(\theta) & c(\phi\theta) \end{bmatrix},$$

where $c(\phi\theta\psi) = c(\phi)c(\theta)c(\psi)$, $s(\phi\theta\psi) = s(\phi)s(\theta)s(\psi)$, and so on; E_z is a unit vector along the z axis of the inertial frame, i.e., $E_z = [0, 0, 1]^T$; $\omega = [p, q, r]^T$ denotes the angular velocity vector of the unmanned helicopter represented in the body frame; $Sk(\omega)$ is the skew-symmetric matrix defined as

$$Sk(\omega) = \begin{bmatrix} 0 & -r & q \\ r & 0 & -p \\ -q & p & 0 \end{bmatrix}; \tag{6}$$

$I = diag\{I_x, I_y, I_z\}$ is the inertial matrix; F and M denote all the external forces except the gravity and all the external moments expressed in the body frame, respectively.

To be more exact, F and M can be, respectively, specified by

$$F = \begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix} = \begin{bmatrix} X_{mr} + X_{fd} \\ Y_{mr} + Y_{fd} + Y_{vt} \\ Z_{mr} + Z_{fd} \end{bmatrix} \tag{7}$$

and

$$M = \begin{bmatrix} M_x \\ M_y \\ M_z \end{bmatrix} = \begin{bmatrix} L_{mr} + L_{vt} \\ M_{mr} \\ N_{mr} \end{bmatrix}, \tag{8}$$

where X_{mr}, Y_{mr}, Z_{mr} and X_{fd}, Y_{fd}, Z_{fd} are, respectively, the components of the main rotor force and the fuselage drag force along the body frame x-, y-, and z-axes; Y_{vt} and L_{vt} are the vertical tail force and moment; L_{mr}, M_{mr}, and N_{mr} are the components of the main rotor moment along the body frame x-, y-, and z-axes. Because the yaw control in manual flight of a small-scale helicopter is very challenging due to the high sensitivity to the control input and the strong coupling effect between the main rotor and the tail rotor, an angular vector control system (AVCS) is usually equipped to facilitate the pilot for heading hold. Thus, the forces and moments of the tail rotor can be disregarded.

It is known that the dominant forces and moments are mainly produced by the main rotor. According to [2], disregarding the effect of the flapping angles and assuming there exist no twists at the blades of the main rotor, the thrust of the main rotor can be expressed as follows

$$\begin{cases} T = (w_b - v_{im})\dfrac{\rho\Omega R^2 C_{l\alpha} b_m c_m}{4}, \\ v_{im}^2 = \sqrt{(\dfrac{v_{int}^2}{2})^2 + (\dfrac{T}{2\rho\pi R^2})^2} - \dfrac{v_{int}^2}{2}, \\ v_{int}^2 = \bar{u}^2 + \bar{v}^2 + \bar{w}(\bar{w} - 2v_{im}), \end{cases} \tag{9}$$

with $w_b = \bar{w} + a\bar{u} - b\bar{v} + \frac{2}{3}\Omega R\theta_{col}$, $\theta_{col} = K_{col}K_a\delta_{col}$, where v_{int} is an intermediate variable in the calculation process of the main rotor thrust; v_{im} indicates the induced velocity of the main rotor; w_b is the resultant velocity at the main rotor in vertical direction; b_m denotes the number of the main blades; c_m is the chord length of the main blade; θ_{col} is the collective pitch angle controlled by the collective actuator; $V_w = [u_w, v_w, w_w]^T$ denotes the wind velocity vector in the body frame; $\bar{u} = u - u_w$, $\bar{v} = v - v_w$, $\bar{w} = w - w_w$; K_{col} is the linkage gain from the collective actuator to the main blade; δ_{col} indicates the control signal of the collective actuator. Subsequently, the force components and the moment components generated by the main rotor can be written as follows

$$\begin{cases} X_{mr} &= -T\sin(a), \\ Y_{mr} &= T\sin(b), \\ Z_{mr} &= -T\cos(a)\cos(b), \\ L_{mr} &= (k_\beta + TH_{mr})\sin(b), \\ M_{mr} &= (k_\beta + TH_{mr})\sin(a), \end{cases} \tag{10}$$

where k_β and H_{mr} denote the main rotor spring constant and the vertical distance between the main rotor hub and the centre of gravity, respectively.

The fuselage drag forces are calculated [1,12] as follows

$$X_{fd} = \begin{cases} -0.5\rho S_{fux}\bar{u}v_{im}, & \text{if } |\bar{u}| \le v_{im}; \\ -0.5\rho S_{fux}\bar{u}\,|\bar{u}|, & \text{if } |\bar{u}| > v_{im}. \end{cases} \tag{11}$$

$$Y_{fd} = \begin{cases} -0.5\rho S_{fuy}\bar{v}v_{im}, & \text{if } |\bar{v}| \le v_{im}; \\ -0.5\rho S_{fuy}\bar{v}\,|\bar{v}|, & \text{if } |\bar{v}| > v_{im}. \end{cases} \tag{12}$$

$$Z_{fd} = -0.5\rho S_{fuz}(\bar{w} - v_{im})\,|\bar{w} - v_{im}|, \tag{13}$$

where S_{fux}, S_{fuy}, and S_{fuz} are longitudinal, lateral, and vertical effective fuselage areas, respectively.

The vertical tail force is defined by

$$Y_{vt} = \begin{cases} -0.5\rho C_{l\alpha,vt} S_{vt} v_{vt}\,|\bar{u}|, & \text{if } |v_{vt}| \le 0.3\,|\bar{u}|; \\ -0.5\rho S_{vt} v_{vt}\,|v_{vt}|, & \text{if } |v_{vt}| > 0.3\,|\bar{u}|. \end{cases} \tag{14}$$

Here $v_{vt} = \bar{v} - v_{it} - rD_{vt}$, and v_{it} denotes the induced velocity of the tail rotor. Then the vertical tail moment is given by $L_{vt} = Y_{vt}H_{vt}$, where H_{vt} means the vertical distance between the vertical tail and the centre of gravity.

Remark 1 In spite of the fact that the fuselage drag forces and the vertical tail force and moment are taken into account in the modeling process of the small-scale unmanned helicopter to improve the precision of the model, they are omitted in the design procedure of the flight control system for simplicity, testing the robustness of the controller to some extent.

3.2 Main rotor dynamics

Rotor flapping dynamics is a unique feature of a small-scale unmanned helicopter. The dynamics of the stabilizer mounted on the main rotor to facilitate manual operation should also be considered. Then the main rotor flapping dynamics and the stabilizer flapping dynamics are modeled, respectively, as follows

$$\begin{aligned}\begin{bmatrix}\dot{a}\\ \dot{b}\end{bmatrix} &= \frac{\gamma^m \Omega}{(\gamma^m)^2+64}\begin{bmatrix}-\dfrac{4k_\beta}{I_\beta \Omega^2}-8 & \gamma^m - \dfrac{32k_\beta}{\gamma^m I_\beta \Omega^2}\\ \dfrac{32k_\beta}{\gamma^m I_\beta \Omega^2}-\gamma^m & -\dfrac{4k_\beta}{I_\beta \Omega^2}-8\end{bmatrix}\begin{bmatrix}a\\ b\end{bmatrix}\\ &\quad -\begin{bmatrix}q\\ p\end{bmatrix}+\frac{\gamma^m \Omega}{(\gamma^m)^2+64}\begin{bmatrix}-\gamma^m & 8\\ 8 & \gamma^m\end{bmatrix}\begin{bmatrix}B_1^m + K_{mix}d\\ A_1^m + K_{mix}c\end{bmatrix},\end{aligned} \tag{15}$$

$$\begin{aligned}\begin{bmatrix}\dot{c}\\ \dot{d}\end{bmatrix} &= \frac{\gamma^s \Omega}{(\gamma^s)^2+64}\begin{bmatrix}-8 & \gamma^s\\ -\gamma^s & -8\end{bmatrix}\begin{bmatrix}c\\ d\end{bmatrix}\\ &\quad -\begin{bmatrix}q\\ p\end{bmatrix}+\frac{\gamma^s \Omega}{(\gamma^s)^2+64}\begin{bmatrix}-\gamma^s & 8\\ 8 & \gamma^s\end{bmatrix}\begin{bmatrix}B_1^s\\ A_1^s\end{bmatrix},\end{aligned} \tag{16}$$

where (a, b) and (c, d) are the flapping angles of the main rotor and the stabilizer in the longitudinal direction and the lateral direction, respectively; (A_1^m, B_1^m) and (A_1^s, B_1^s) indicate the cyclic pitch angles of the main blade and the stabilizer, determined by pitch angles of swash plate as follows

$$\begin{cases}A_1^m = K_{bel}A_1,\\ B_1^m = K_{bel}B_1,\\ A_1^s = K_{sb}A_1,\\ B_1^s = K_{sb}B_1,\end{cases} \tag{17}$$

where K_{bel} and K_{sb} are the mechanical gains from the swash plate to the main blade and the stabilizer, respectively; A_1 and B_1 denote the pitch angles of the swash plate in longitudinal and lateral directions, respectively, which can be calculated from

$$\begin{cases}A_1 = K_{lon}K_a\delta_{lon} + \theta_{lon0},\\ B_1 = K_{lat}K_a\delta_{lat} + \theta_{lat0},\end{cases} \tag{18}$$

θ_{lon0} is the initial tilting angle in the longitudinal direction and θ_{lat0} is the initial tilting angle in the lateral direction; K_a denotes the gain of the actuators; K_{lon} and K_{lat} are the linkage gains from the elevator actuator and the aileron actuator to the swash plate, respectively.

3.3 Yaw dynamics

The fundamental yaw dynamics can be written as follows

$$\dot{r} = N_r r + N_{ped}\bar{\delta}_{ped}, \tag{19}$$

where N_r and N_{ped} are model parameters; $\bar{\delta}_{ped}$ is the output of the AVCS mentioned in Sect. 3.1, which is composed of a low-cost rate gyro and an embedded PI controller. Therefore, $\bar{\delta}_{ped}$ can be further expressed as follows

$$\bar{\delta}_{ped} = k_P\left(K_c\delta_{ped} - r\right) + k_I\int\left(K_c\delta_{ped} - r\right)dt, \tag{20}$$

where k_P and k_I are the proportional coefficient and the integral coefficient of the embedded PI controller, respectively.

Then, the integrated yaw dynamics including the dynamics of the AVCS can be modeled as follows

$$\begin{cases}\dot{\delta}_{ped,\mathrm{int}} = K_c\delta_{ped} - r,\\ \dot{r} = (N_r - N_1)r + N_2\delta_{ped,\mathrm{int}} + N_1K_c\delta_{ped},\end{cases} \tag{21}$$

where $N_1 = N_{ped}k_P$; $N_2 = N_{ped}k_I$.

4 RBF neural networks-based nonlinear adaptive backstepping control design

The design of a robust and nonlinear adaptive flight controller is conducted in this section. The flight control system is divided into three parts: longitudinal–lateral subsystem, heave subsystem, and heading subsystem. Its structure diagram is portrayed in Fig. 3. The control

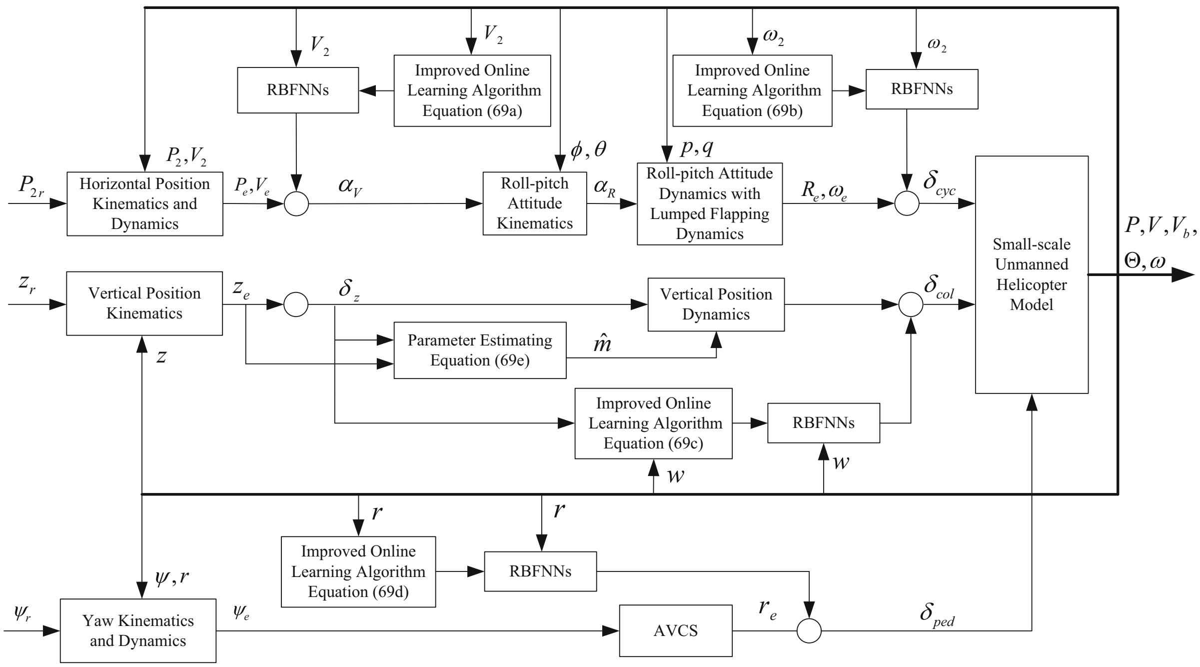


Fig. 3 Diagram of the nonlinear adaptive backstepping control system based on RBF neural networks

design for longitudinal–lateral subsystem is described in Sect. 4.1; the design procedures for heave subsystem and heading subsystem are presented in Sects. 4.2 and 4.3, respectively; the stability analysis of the closed-loop system is revealed in Sect. 4.4, in which the relative updating laws are also derived by means of the Lyapunov stability theory.

4.1 Longitudinal–lateral subsystem

Since the dominating force is the thrust component along the z axis of the body frame, for simplicity, X_{mr} and Y_{mr} are disregarded and included in the lumped disturbance, which also contains other neglected forces in the design procedure. Generally, it can be assumed that $c(a) \approx 1$ and $c(b) \approx 1$ because the flapping angles are ordinarily very small. Then the external force F is divided into the following two parts

$$F = \underbrace{[0, 0, -T]^T}_{F_0} + \underbrace{\left[d_x, d_y, d_z\right]^T}_{d_F}, \tag{22}$$

where d_F is the lumped disturbance resulting from other neglected forces, such as the fuselage drag force and the vertical tail force. Similarly, assuming $s(a) \approx a$ and $s(b) \approx b$, the external torque exerted on the unmanned helicopter can be written as

$$M = \underbrace{\left[\left(k_\beta + T H_{mr}\right) b, \left(k_\beta + T H_{mr}\right) a, 0\right]^T}_{M_0} + \underbrace{\left[d_p, d_q, d_r\right]^T}_{d_M}, \tag{23}$$

where d_M is the lumped disturbance produced by other neglected torques. The flapping dynamics composed of Eqs. (15) and (16) is too complicated for control design. For convenience, under the assumption of $\dot{c} \approx 0$ and $\dot{d} \approx 0$, the flapping dynamics is reduced to the following lumped form

$$\dot{\beta}_m = A_m \beta_m + A_\omega \omega_2 + B \delta_{cyc}, \tag{24}$$

where $\beta_m = [a, b]^T$; $\omega_2 = [p, q]^T$; $\delta_{cyc} = [\delta_{lat}, \delta_{lon}]^T$; A_m and A_ω are the coefficient matrices calculated from Eqs. (15) and (16).

Input formulas (22) and (23) into the fuselage rigid body kinematics and dynamics (2)–(5), one can extract the longitudinal–lateral subsystem as follows

$$\dot{P}_2 = V_2, \tag{25}$$

$$\dot{V}_2 = \frac{1}{m} R_2 (d_z - T) + \frac{1}{m} R_N d_{\bar{F}}, \tag{26}$$

$$\dot{R}_2 = R_M \omega_2 \tag{27}$$

$$\dot{\omega}_2 = R_\omega \omega_2 + R_\beta \beta_m + d_{\bar{M}}, \tag{28}$$

$$\dot{\beta}_m = A_m \beta_m + A_\omega \omega_2 + B \delta_{cyc}, \tag{29}$$

where $P_2 = [x, y]^T$; $V_2 = [v_x, v_y]^T$; $R_2 = [R_{13}, R_{23}]^T$;
$d_{\bar{F}} = [d_x, d_y]^T$; $R_N = \begin{bmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \end{bmatrix}$; $R_M = \begin{bmatrix} -R_{12} & R_{11} \\ -R_{22} & R_{21} \end{bmatrix}$;
$d_{\bar{M}} = [d_p, d_q]^T + \bar{d}_T H_{mr} \begin{bmatrix} 0 & 1/I_x \\ 1/I_y & 0 \end{bmatrix} \beta_m$;
$R_\omega = \begin{bmatrix} 0 & r(I_y - I_z)/I_x \\ r(I_z - I_x)/I_y & 0 \end{bmatrix}$;
$R_\beta = (k_\beta + T_{int} H_{mr}) \begin{bmatrix} 0 & 1/I_x \\ 1/I_y & 0 \end{bmatrix}$; R_{ij} indicates the element in the ith row and the jth column of the transformation matrix $R(\Theta)$; T is expressed as

$$T = T_{int} + \bar{d}_T, \tag{30}$$

where T_{int} is the intermediate control in the heave subsystem and $\bar{d}_T$ is the lumped disturbance.

Step 1 Define $P_e = P_2 - P_{2r}$, and differentiate P_e with respect to the time variable to get

$$\dot{P}_e = V_2 - \dot{P}_{2r}. \tag{31}$$

By defining the horizontal velocity error as $V_e = V_2 - \alpha_P$, and substituting it into Eq. (31), one can obtain

$$\dot{P}_e = V_2 - \dot{P}_{2r} = V_e + \alpha_P - \dot{P}_{2r}, \tag{32}$$

where α_P is the pseudocontrol. To stabilize the horizontal position error dynamics (32), the pseudocontrol is chosen as

$$\alpha_P = -K_P P_e + \dot{P}_{2r}, \tag{33}$$

where K_P is a positive definite diagonal matrix. Applying the pseudocontrol (33) into equation (32), the horizontal position error dynamics becomes

$$\dot{P}_e = V_e - K_P P_e. \tag{34}$$

Now, define the Lyapunov function candidate for P_e as follows

$$L_P = \frac{1}{2} P_e^T P_e. \tag{35}$$

Differentiating (35) and substituting (34) into the differential equation of (35) yield

$$\dot{L}_P \leq -\lambda_{\min} (K_P) \left\| P_e \right\|^2 + P_e^T V_e. \tag{36}$$

Step 2 For convenience, the horizontal translational dynamics (26) can be rewritten as

$$\dot{V}_2 = -\frac{1}{\hat{m}} R_2 T_{int} + \underbrace{\frac{1}{\hat{m}} R_2 \left(T - \bar{d}_T\right) + \frac{1}{m} (R_2 d_z + R_N d_{\bar{F}} - R_2 T)}_{\Delta_F}, \tag{37}$$

where $\hat{m}$ is the estimation of the unmanned helicopter's mass, which is derived by the adaptive law (69e) presented in Sect. 4.4. Δ_F denotes the lumped disturbance. Then, the time derivative of V_e can be denoted as follows

$$\dot{V}_e = -\frac{1}{\hat{m}} (R_e + \alpha_V) T_{int} + K_P \dot{P}_e - \ddot{P}_{2r} + \Delta_F, \tag{38}$$

where $\Delta_F = W_1^T H_1(V_2) + \sigma_1(V_2)$; $W_1 \in R^{K_1 \times 2}$ is the ideal neural network weight matrix; K_1 is the number of the nodes in the hidden layer; $\sigma_1(V_2) \in R^2$ is the bounded neural network approximation error, and $H_1(V_2) \in R^{K_1}$ is the Gaussian basis function of V_2; R_e is specified by $R_e = R_2 - \alpha_V$.

Subsequently, to stabilize the horizontal velocity error dynamics (38), the pseudocontrol α_V is designed as

$$\alpha_V = \frac{\hat{m}}{T_{int}} \left(K_V V_e + P_e + K_P \dot{P}_e - \ddot{P}_{2r}\right) + \frac{\hat{m}}{2 T_{int} \varepsilon_1^2} V_e \hat{\Gamma}_1 H_1^T (V_2) H_1(V_2), \tag{39}$$

where K_V is a positive definite diagonal matrix and ε_1 is a designed positive constant; $\hat{\Gamma}_1$ is the estimation of Γ_1, which is the square of the induced Euclidean norm about W_1, defined as $\Gamma_1 = \|W_1\|_2^2$.

Remark 2 A point worth emphasizing is that in some previous literatures [16,27,39] the ideal neural network weight matrix W_1 is directly estimated without

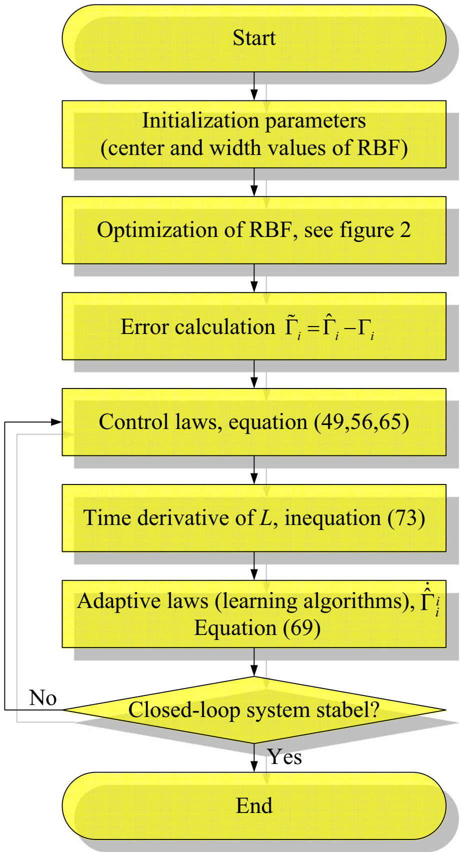


Fig. 4 Sequential flowchart of the learning algorithm

any optimization, leading to as many as $K_1 \times 2$ adaptive laws. With the increase in the nodes in the hidden layer or the output layer, the number of the updating laws grows exponentially, resulting in the calculation explosion of the online embedded computer system. Differently from the foregoing documents, this paper adopts an improved learning algorithm of the least parameters, which compensates the uncertainties utilizing only one parameter to estimate the square of the induced Euclidean norm regarding W_1 indirectly, bringing about a more efficient control system in practical applications. In addition, a sequential flowchart illustrated in Fig. 4 is provided to further explain this algorithm.

Remark 3 In [18] the dynamic surface control approach is combined with "minimal learning parameters" algorithm for MIMO systems. While in [23] and [22] a reinforcement learning algorithm with less learning parameters is put forward to tackle the adaptive tracking control and fault-tolerant control issue for MIMO discrete-time systems. Although the similar RBFNN is used in [18,23] and [22], the focus of this paper is to propose an adaptive backstepping controller with a learning algorithm of the least parameters for unmanned helicopters, which is different from the works in the above papers. Moreover, the learning algorithms used here are very different from those in [18,23] and [22]. What they estimated indirectly in the RBF neural network is the Euclidean norm of unknown weights, which is actually different from the modified adaptive neural network designed in this paper. In order to combine the modified RBFNN and the backstepping control technique, and ensure the asymptotic stability of the whole closed-loop system, it is necessary to estimate the square of the norm of the weights rather than the norm itself and this is the difference in dealing with adaptive neural networks between this paper and the literatures mentioned above.

Then, the Lyapunov function candidate for V_e is defined by

$$L_V = \frac{1}{2} V_e^T V_e. \tag{40}$$

Combining Eqs. (38) and (39), one obtains the time derivative of L_V

$$\begin{aligned}
\dot{L}_V =& -V_e^T K_V V_e - V_e^T P_e - \frac{1}{\hat{m}} T_{int} V_e^T R_e \\
&+ V_e^T \sigma_1(V_2) + V_e^T W_1^T H_1(V_2) \\
&- \frac{1}{2\varepsilon_1^2} V_e^T V_e \hat{\Gamma}_1 H_1^T(V_2) H_1(V_2) \\
\leq& -(\lambda_{\min}(K_V) - \varsigma_V)\|V_e\|^2 + V_e^T \sigma_1(V_2) \\
&- \|V_e\|\,\|\sigma_1(V_2)\| - \left(\sqrt{\varsigma_V}\,\|V_e\| - \frac{\|\sigma_1(V_2)\|}{2\sqrt{\varsigma_V}}\right)^2 \\
&- V_e^T P_e - \frac{1}{\hat{m}} T_{int} V_e^T R_e + V_e^T W_1^T H_1(V_2) \\
&- \frac{1}{2\varepsilon_1^2} V_e^T V_e \hat{\Gamma}_1 H_1^T(V_2) H_1(V_2) + \frac{\|\sigma_1(V_2)\|^2}{4\varsigma_V} \\
\leq& -(\lambda_{\min}(K_V) - \varsigma_V)\|V_e\|^2 - V_e^T P_e \\
&- \frac{1}{\hat{m}} T_{int} V_e^T R_e + V_e^T W_1^T H_1(V_2) \\
&- \frac{1}{2\varepsilon_1^2} V_e^T V_e \hat{\Gamma}_1 H_1^T(V_2) H_1(V_2) + \frac{\|\sigma_1(V_2)\|^2}{4\varsigma_V},
\end{aligned} \tag{41}$$

where ς_V is a constant and $0 < \varsigma_V < \lambda_{\min}(K_V)$.

Step 3 Taking the time derivative of R_e and combining (27) bring about

$$\dot{R}_e = R_M(\omega_e + \alpha_R) - \hat{\dot{\alpha}}_V + \tilde{\dot{\alpha}}_V, \tag{42}$$

where $\omega_e = \omega_2 - \alpha_R$ is the roll-pitch angular velocity error, α_R is the pseudocontrol. A second-order command filter [8] is used here to estimate the time derivative of α_V due to the extremely tedious solving process of analytic solution for the time derivative of α_V. $\hat{\dot{\alpha}}_V$ is the estimation of $\dot{\alpha}_V$, and $\tilde{\dot{\alpha}}_V = \hat{\dot{\alpha}}_V - \dot{\alpha}_V$ is the estimation error of the second-order command filter.

Then, design the pseudocontrol α_R as

$$\alpha_R = R_M^{-1}\left(-K_R R_e + \frac{1}{\hat{m}}T_{int}V_e + \hat{\dot{\alpha}}_V\right), \tag{43}$$

where K_R is a positive definite diagonal matrix.

The Lyapunov function candidate including R_e is specified by

$$L_R = \frac{1}{2}R_e^T R_e. \tag{44}$$

Taking the time derivative of L_R and utilizing Eqs. (42) and (43), it generates

$$\begin{aligned}\dot{L}_R &= -R_e^T K_R R_e + \frac{1}{\hat{m}}T_{int}R_e^T V_e \\ &\quad + R_e^T R_M \omega_e + R_e^T \tilde{\dot{\alpha}}_V \\ &\le -\lambda_{\min}(K_R)\|R_e\|^2 + \frac{1}{\hat{m}}T_{int}R_e^T V_e \\ &\quad + R_e^T R_M \omega_e + R_e^T \tilde{\dot{\alpha}}_V \\ &\le -(\lambda_{\min}(K_R) - \varsigma_R)\|R_e\|^2 \\ &\quad + \frac{1}{\hat{m}}T_{int}R_e^T V_e + R_e^T R_M \omega_e + \frac{\left\|\tilde{\dot{\alpha}}_V\right\|^2}{4\varsigma_R},\end{aligned} \tag{45}$$

where ς_R is a constant that satisfies the following constraints: $0 < \varsigma_R < \lambda_{\min}(K_R)$.

Step 4 Because of the much faster response of the main rotor compared with the fuselage, the flapping dynamics is approximated by a steady state equation, which is proved to be acceptable and effective for control design in [40]. Then, the simplified flapping dynamics is denoted by

$$\beta_m = -A_m^{-1}\left(A_\omega \omega_2 + B\delta_{cyc}\right). \tag{46}$$

With Eq. (46) substituted into (28), the roll-pitch angular velocity dynamics becomes

$$\dot{\omega}_2 = \left(R_\omega - R_\beta A_m^{-1}A_\omega\right)\omega_2 - R_\beta A_m^{-1}B\delta_{cyc} + d_{\bar{M}}. \tag{47}$$

Subsequently, the time derivative of ω_e can be rewritten as follows

$$\begin{aligned}\dot{\omega}_e &= \left(R_\omega - R_\beta A_m^{-1}A_\omega\right)\omega_2 \\ &\quad - R_\beta A_m^{-1}B\delta_{cyc} + d_{\bar{M}} - \dot{\alpha}_R,\end{aligned} \tag{48}$$

where $d_{\bar{M}} = W_2^T H_2(\omega_2) + \sigma_2(\omega_2)$ is the total torture disturbance, which will be estimated by the RBF neural network learning algorithm of the least parameters indirectly; $\sigma_2(\omega_2)$ is the bounded approximation error. Thus, the actual longitudinal–lateral control is designed as follows

$$\begin{aligned}\delta_{cyc} = -\left(R_\beta A_m^{-1}B\right)^{-1}\Big(&-K_\omega \omega_e - R_M^T R_e \\ &- R_\omega \omega_2 + R_\beta A_m^{-1}A_\omega \omega_2 + \hat{\dot{\alpha}}_R \\ &- \frac{1}{2\varepsilon_2^2}\omega_e \hat{\Gamma}_2 H_2^T(\omega_2)H_2(\omega_2)\Big),\end{aligned} \tag{49}$$

where K_ω is a positive definite diagonal matrix; ε_2 is a positive number; $\hat{\Gamma}_2$ is the estimation of Γ_2, which is specified by $\Gamma_2 = \|W_2\|_2^2$; $\hat{\dot{\alpha}}_R$ is the estimation of $\dot{\alpha}_R$ through the command filter.

Then the Lyapunov function candidate containing ω_e is selected as

$$L_\omega = \frac{1}{2}\omega_e^T \omega_e, \tag{50}$$

which gives the time derivative of L_ω

$$\begin{aligned}\dot{L}_\omega &= -\omega_e^T K_\omega \omega_e - \omega_e^T R_M^T R_e + \omega_e^T W_2^T H_2(\omega_2) \\ &\quad - \frac{1}{2\varepsilon_2^2}\omega_e^T \omega_e \hat{\Gamma}_2 H_2^T(\omega_2)H_2(\omega_2) + \omega_e^T d_\omega \\ &\le -(\lambda_{\min}(K_\omega) - \varsigma_\omega)\|\omega_e\|^2 \\ &\quad - \omega_e^T R_M^T R_e + \omega_e^T W_2^T H_2(\omega_2) \\ &\quad - \frac{1}{2\varepsilon_2^2}\omega_e^T \omega_e \hat{\Gamma}_2 H_2^T(\omega_2)H_2(\omega_2) + \frac{\|d_\omega\|^2}{4\varsigma_\omega},\end{aligned} \tag{51}$$

where $0 < \varsigma_\omega < \lambda_{\min}(K_\omega)$; $d_\omega = \sigma_2(\omega_2) + \tilde{\dot{\alpha}}_R$, and $\tilde{\dot{\alpha}}_R$ is the approximation error of the command filter.

4.2 Heave subsystem

As mentioned in Sect. (4.1), the thrust T in (30) can be rewritten as follows

$$T = \underbrace{k_1 w + k_1 k_2 K_a \delta_{col} - k_1 \nu_{i0}}_{T_{int}} + \bar{d}_T, \tag{52}$$

where $k_1 = \dfrac{\rho\Omega R^2 C_{l\alpha} b_m c_m}{4}$; $k_2 = \dfrac{2}{3}\Omega R$; $\bar{d}_T = d_T - k_1(\nu_{im} - \nu_{i0})$; d_T is the disturbance including system uncertainty and external disturbance. Moreover, $\bar{d}_T$ is given by $\bar{d}_T = -W_3^T H_3(w)(w) - \sigma_3(w)$, where $W_3 \in R^{K_3}$ and $\sigma_3(w) \in R$ are the ideal neural network weight matrix and the bounded approximation error.

According to Eqs. (2) and (3), the heave subsystem can be specified by the following form

$$\begin{cases} \dot{z} = v_z, \\ m\dot{v}_z = -Tc(\theta)c(\phi) + mg. \end{cases} \tag{53}$$

Substituting Eq. (52) into (53), the heave dynamics can be rewritten as follows

$$m\ddot{z} = -c(\theta)c(\phi)\left(T_{int} + \bar{d}_T\right) + mg. \tag{54}$$

Then the intermediate control T_{int} is designed as

$$T_{int} = -\frac{1}{c(\theta)c(\phi)}\hat{m}(\nu - g) + \frac{1}{2\varepsilon_3^2}\delta_z \hat{\Gamma}_3 H_3(w)^T H_3(w), \tag{55}$$

where $\hat{m}$ is the estimation of the actual mass; $\hat{\Gamma}_3$ is the estimation of Γ_3, which is specified by $\Gamma_3 = \|W_3\|^2$; ν is specified by $\nu = \ddot{z}_r - 2k_z\dot{z}_e - k_z^2 z_e$ and δ_z is defined as $\delta_z = \dot{z}_e + k_z z_e$, with $z_e = z - z_r$.

Presently the actual heave control can be derived as follows by combining Eqs. (52) and (55)

$$\begin{aligned}\delta_{col} = &-\frac{\hat{m}(\nu - g)}{k_1 k_2 K_a c(\theta)c(\phi)} + \frac{\nu_{i0} - w}{k_2 K_a} \\ &- \frac{1}{2k_1 k_2 K_a \varepsilon_3^2}\delta_z \hat{\Gamma}_3 H_3(w)^T H_3(w).\end{aligned} \tag{56}$$

With Eq. (55) substituted into (54), it yields

$$\begin{aligned} m(\ddot{z} - \nu) = &\ \tilde{m}(\nu - g) + \sigma_3(w)c(\theta)c(\phi) \\ &+ W_3^T H_3(w)c(\theta)c(\phi) \\ &- \frac{1}{2\varepsilon_3^2}\delta_z \hat{\Gamma}_3 H_3(w)^T H_3(w)c(\theta)c(\phi). \end{aligned} \tag{57}$$

Through the attentive analysis of ν and δ_z, the following expression can be found

$$\begin{aligned}(\ddot{z} - \nu) &= \ddot{z} - \ddot{z}_r + 2k_z\dot{z}_e + k_z^2 z_e \\ &= \dot{\delta}_z + k_z\delta_z.\end{aligned} \tag{58}$$

Then, define the Lyapunov function candidate for δ_z as follows

$$L_z = \frac{1}{2}m\delta_z^2. \tag{59}$$

With Eqs. (57) and (58) substituted into the time derivative equation of (59), the time derivative of L_z can be calculated as follows

$$\begin{aligned}\dot{L}_z = &-mk_z\delta_z^2 + \tilde{m}\delta_z(\nu - g) \\ &+ W_3^T H_3(w)c(\theta)c(\phi)\delta_z + \sigma_3(w)c(\theta)c(\phi) \\ &- \frac{1}{2\varepsilon_3^2}\delta_z^2 \hat{\Gamma}_3 H_3(w)^T H_3(w)c(\theta)c(\phi) \\ \le &-(mk_z - \varsigma_z)\delta_z^2 + \tilde{m}\delta_z(\nu - g) \\ &+ W_3^T H_3(w)\delta_z c(\theta)c(\phi) \\ &- \frac{1}{2\varepsilon_3^2}\delta_z^2 \hat{\Gamma}_3 H_3(w)^T H_3(w)c(\theta)c(\phi) \\ &+ \frac{(\sigma_3(w)c(\theta)c(\phi))^2}{4\varsigma_z},\end{aligned} \tag{60}$$

where ς_z is a positive number which satisfies the following constraints: $0 < \varsigma_z < k_z$.

4.3 Heading subsystem

In this section, the design of the heading control subsystem is carried out. First, the yaw kinematics can be extracted from the attitude kinematics representation (4) as follows

$$\dot{\psi} = \frac{s(\phi)}{c(\theta)}q + \frac{c(\phi)}{c(\theta)}r. \tag{61}$$

By defining $\psi_e = \psi - \psi_r$ and $r_e = r - \alpha_\psi$, the time derivative of ψ_e can be expressed by

$$\dot{\psi}_e = \frac{s(\phi)}{c(\theta)}q + \frac{c(\phi)}{c(\theta)}\left(r_e + \alpha_\psi\right) - \dot{\psi}_r. \tag{62}$$

Choose the virtual control α_ψ as

$$\alpha_\psi = \frac{c(\theta)}{c(\phi)}\left(-k_\psi\psi_e + \dot{\psi}_r - \frac{s(\phi)}{c(\theta)}q\right), \tag{63}$$

where k_ψ is a positive number.

Take the time derivative of r_e and combine Eq. (21) to get

$$\dot{r}_e = (N_r - N_1)r + N_1 K_c \delta_{ped} - \dot{\alpha}_\psi + d_r, \tag{64}$$

where $d_r = N_2\delta_{ped,\mathrm{int}}$, which is to be estimated by the RBFNN through $d_r = W_4^T H_4(r) + \sigma_4(r)$; $\sigma_4(r) \in R$ indicates the bounded approximation error; $W_4 \in R^{K_4}$ denotes the ideal weight matrix, and K_4 is the number of the nodes in the hidden layer.

Finally, the actual control is chosen as

$$\begin{aligned}\delta_{ped} = \frac{1}{N_1 K_c}\Bigg(&-k_r r_e - (N_r - N_1)r - \frac{c(\phi)}{c(\theta)}\psi_e \\ &-\frac{1}{2\varepsilon_4^2} r_e \hat{\Gamma}_4 H_4^T(r) H_4(r) + \hat{\dot{\alpha}}_\psi\Bigg),\end{aligned} \tag{65}$$

where k_r and ε_4 are both positive numbers; $\hat{\Gamma}_4$ is the estimation of Γ_4, while Γ_4 is given by $\Gamma_4 = \|W_4\|^2$.

The Lyapunov function candidate including ψ_e and r_e is selected as

$$L_\psi = \frac{1}{2}\psi_e^2 + \frac{1}{2}r_e^2. \tag{66}$$

Substituting Eq. (63) into (62), (65) into (64), and differentiating (66) yield

$$\begin{aligned}\dot{L}_\psi &= -k_\psi\psi_e^2 - k_r r_e^2 + r_e\left(\tilde{\dot{\alpha}}_\psi + \sigma_4(r)\right) \\ &\quad + r_e W_4^T H_4(r) - \frac{1}{2\varepsilon_4^2} r_e^2 \hat{\Gamma}_4 H_4^T(r) H_4(r) \\ &\le -k_\psi\psi_e^2 - (k_r - \varsigma_r) r_e^2 \\ &\quad + r_e W_4^T H_4(r) - \frac{1}{2\varepsilon_4^2} r_e^2 \hat{\Gamma}_4 H_4^T(r) H_4(r) \\ &\quad + \frac{\left(\tilde{\dot{\alpha}}_\psi + \sigma_4(r)\right)^2}{4\varsigma_r},\end{aligned} \tag{67}$$

where ς_r is a constant and $0 < \varsigma_r < k_r$; $\tilde{\dot{\alpha}}_\psi$ is the estimation error of the command filter.

4.4 Stability analysis

Assumption 1 During the whole process of the autonomous flight, the roll angle and the pitch angle of the unmanned helicopter's fuselage always satisfy the following constraints

$$-\pi/2 < \phi < \pi/2, \quad -\pi/2 < \theta < \pi/2. \tag{68}$$

Assumption 2 The desired trajectories $P_{2r}(t)$, $z_r(t)$, $\psi_r(t)$, and their second-order time derivatives $\ddot{P}_{2r}(t)$ and $\ddot{z}_r(t)$ are bounded and continuous.

The relative adaptive laws are designed as the following forms

$$\dot{\hat{\Gamma}}_1 = \frac{\zeta_1}{2\varepsilon_1^2} V_e^T V_e H_1^T(V_2) H_1(V_2) - \xi_1\hat{\Gamma}_1, \tag{69a}$$

$$\dot{\hat{\Gamma}}_2 = \frac{\zeta_2}{2\varepsilon_2^2} \omega_e^T \omega_e H_2^T(\omega_2) H_2(\omega_2) - \xi_2\hat{\Gamma}_2, \tag{69b}$$

$$\dot{\hat{\Gamma}}_3 = \frac{\zeta_3}{2\varepsilon_3^2} \delta_z^2 H_3^T(w) H_3(w) c(\theta) c(\phi) - \xi_3\hat{\Gamma}_3, \tag{69c}$$

$$\dot{\hat{\Gamma}}_4 = \frac{\zeta_4}{2\varepsilon_4^2} r_e^2 H_4^T(r) H_4(r) - \xi_4\hat{\Gamma}_4, \tag{69d}$$

$$\dot{\hat{m}} = \gamma\left(g\delta_z - \nu\delta_z - \eta\hat{m}\right), \tag{69e}$$

where ζ_i, ε_i and γ are designed positive parameters; ξ_i and η are small correction parameters which are greater than zero. With the appropriate parameters selected for Eq. (69), the estimation errors $\tilde{m} = \hat{m} - m$ and $\tilde{\Gamma}_i = \Gamma_i - \hat{\Gamma}_i$ are bounded $(i = 1, 2, 3, 4)$.

Theorem 1 *Under* Assumption 1 *and* Assumption 2*, considering a small-scale unmanned helicopter specified by* Eqs. (2), (3), (4), (5), (15), (16) *and* (21)*, the control inputs* (49), (56) *and* (65) *can drive the position and yaw tracking errors to converge to an arbitrarily small compact of the origin asymptotically with the supporting of the updating laws* (69a), (69b), (69c), (69d) *and* (69e)*. And all the error variables* P_e, V_e, R_e, ψ_e, ω_e, z_e, δ_z, r_e, $\tilde{m}$, *and* $\tilde{\Gamma}_i$ *are guaranteed to be semi-globally uniformly ultimately bounded.*

Proof To prove the stability of the overall closed-loop system, the composite Lyapunov function candidate is defined as follows

$$L = L_1 + L_2, \tag{70}$$

where $L_1 = L_P + L_V + L_R + L_\omega + L_z + L_\psi$ and $L_2 = \sum_{i=1}^{4} \frac{1}{2\zeta_i}\tilde{\Gamma}_i^2 + \frac{1}{2\gamma}\tilde{m}^2$. Combining Eqs. (36), (41), (45), (51), (60), and (67), the time derivative of L becomes

$$\begin{aligned}
\dot{L} \le & -\lambda_{\min}(K_P)\|P_e\|^2 - (\lambda_{\min}(K_V) - \varsigma_V)\|V_e\|^2 \\
& - (\lambda_{\min}(K_R) - \varsigma_R)\|R_e\|^2 \\
& - (\lambda_{\min}(K_\omega) - \varsigma_\omega)\|\omega_e\|^2 \\
& - (mk_z - \varsigma_z)\delta_z^2 - k_\psi \psi_e^2 \\
& - (k_r - \varsigma_r) r_e^2 + V_e^T W_1^T H_1(V_2) \\
& - \frac{1}{2\varepsilon_1^2} V_e^T V_e \hat{\Gamma}_1 H_1^T(V_2) H_1(V_2) \\
& + \omega_e^T W_2^T H_2(\omega_2) + \tilde{m}\delta_z(\nu - g) \\
& - \frac{1}{2\varepsilon_2^2}\omega_e^T \omega_e \hat{\Gamma}_2 H_2^T(\omega_2) H_2(\omega_2) \\
& + W_3^T H_3(w)\delta_z c(\theta)c(\phi) \\
& - \frac{1}{2\varepsilon_3^2}\delta_z^2 \hat{\Gamma}_3 H_3^T(w) H_3(w) c(\theta)c(\phi) \\
& + r_e W_4^T H_4(r) - \frac{1}{2\varepsilon_4^2} r_e^2 \hat{\Gamma}_4 H_4^T(r) H_4(r) \\
& + \frac{\|\sigma_1(V_2)\|^2}{4\varsigma_V} + \frac{\left\|\tilde{\dot{\alpha}}_V\right\|^2}{4\varsigma_R} + \frac{\|d_\omega\|^2}{4\varsigma_\omega} \\
& + \frac{(\sigma_3(w)c(\theta)c(\phi))^2}{4\varsigma_z} + \frac{\left(\tilde{\dot{\alpha}}_\psi + \sigma_4(r)\right)^2}{4\varsigma_r} \\
& - \sum_{i=1}^{4}\frac{1}{\zeta_i}\tilde{\Gamma}_i\dot{\hat{\Gamma}}_i + \frac{1}{\gamma}\tilde{m}\dot{\hat{m}}.
\end{aligned} \tag{71}$$

According to the Cauchy–Schwarz inequality and Assumption 1, Eq. (71) can be rewritten as follows

$$\begin{aligned}
\dot{L} \le & -\lambda_{\min}(K_P)\|P_e\|^2 - (\lambda_{\min}(K_V) - \varsigma_V)\|V_e\|^2 \\
& - (\lambda_{\min}(K_R) - \varsigma_R)\|R_e\|^2 \\
& - (\lambda_{\min}(K_\omega) - \varsigma_\omega)\|\omega_e\|^2 \\
& - (mk_z - \varsigma_z)\delta_z^2 - k_\psi \psi_e^2 - (k_r - \varsigma_r) r_e^2 \\
& + \|V_e\|\,\|W_1\|_2\,\|H_1(V_2)\| \\
& - \frac{1}{2\varepsilon_1^2} V_e^T V_e \hat{\Gamma}_1 H_1^T(V_2) H_1(V_2) \\
& + \|\omega_e\|\,\|W_2\|_2\,\|H_2(\omega_2)\| \\
& - \frac{1}{2\varepsilon_2^2}\omega_e^T \omega_e \hat{\Gamma}_2 H_2^T(\omega_2) H_2(\omega_2) \\
& + |\delta_z|\,\|W_3\|\,\|H_3(w)\|\,c(\theta)c(\phi) \\
& - \frac{1}{2\varepsilon_3^2}\delta_z^2 \hat{\Gamma}_3 H_3^T(w) H_3(w) c(\theta)c(\phi) \\
& + \tilde{m}\delta_z(\nu - g) + |r_e|\,\|W_4\|\,\|H_4(r)\| \\
& - \frac{1}{2\varepsilon_4^2} r_e^2 \hat{\Gamma}_4 H_4^T(r) H_4(r) + \frac{\|\sigma_1(V_2)\|^2}{4\varsigma_V} \\
& + \frac{\left\|\tilde{\dot{\alpha}}_V\right\|^2}{4\varsigma_R} + \frac{\|d_\omega\|^2}{4\varsigma_\omega} + \frac{(\sigma_3(w)c(\theta)c(\phi))^2}{4\varsigma_z} \\
& + \frac{\left(\tilde{\dot{\alpha}}_\psi + \sigma_4(r)\right)^2}{4\varsigma_r} - \sum_{i=1}^{4}\frac{1}{\zeta_i}\tilde{\Gamma}_i\dot{\hat{\Gamma}}_i + \frac{1}{\gamma}\tilde{m}\dot{\hat{m}}.
\end{aligned} \tag{72}$$

Then, using the Young's inequality, Eq. (72) can be further expressed by

$$\begin{aligned}
\dot{L} \le & -\lambda_{\min}(K_P)\|P_e\|^2 - (\lambda_{\min}(K_V) - \varsigma_V)\|V_e\|^2 \\
& - (\lambda_{\min}(K_R) - \varsigma_R)\|R_e\|^2 \\
& - (\lambda_{\min}(K_\omega) - \varsigma_\omega)\|\omega_e\|^2 \\
& - (mk_z - \varsigma_z)\delta_z^2 - k_\psi \psi_e^2 - (k_r - \varsigma_r) r_e^2 \\
& + \frac{\|W_1\|_2^2\,\|V_e\|^2\,\|H_1(V_2)\|^2}{2\varepsilon_1^2} \\
& - \frac{1}{2\varepsilon_1^2} V_e^T V_e \hat{\Gamma}_1 H_1^T(V_2) H_1(V_2) \\
& + \frac{\|W_2\|_2^2\,\|\omega_e\|^2\,\|H_2(\omega_2)\|^2}{2\varepsilon_2^2} \\
& - \frac{1}{2\varepsilon_2^2}\omega_e^T \omega_e \hat{\Gamma}_2 H_2^T(\omega_2) H_2(\omega_2) \\
& + \frac{\|W_3\|^2\,|\delta_z|^2\,\|H_3(w)\|^2}{2\varepsilon_3^2} c(\theta)c(\phi) \\
& - \frac{1}{2\varepsilon_3^2}\delta_z^2 \hat{\Gamma}_3 H_3^T(w) H_3(w) c(\theta)c(\phi) \\
& + \frac{\|W_4\|^2\,|r_e|^2\,\|H_4(r)\|^2}{2\varepsilon_4^2} \\
& - \frac{1}{2\varepsilon_4^2} r_e^2 \hat{\Gamma}_4 H_4^T(r) H_4(r) + \tilde{m}\delta_z(\nu - g) \\
& + \frac{\|\sigma_1(V_2)\|^2}{4\varsigma_V} + \frac{\left\|\tilde{\dot{\alpha}}_V\right\|^2}{4\varsigma_R} + \frac{\|d_\omega\|^2}{4\varsigma_\omega} \\
& + \frac{(\sigma_3(w)c(\theta)c(\phi))^2}{4\varsigma_z} + \frac{\left(\tilde{\dot{\alpha}}_\psi + \sigma_4(r)\right)^2}{4\varsigma_r} \\
& + \frac{1}{2}\sum_{i=1}^{4}\varepsilon_i^2 s_i - \sum_{i=1}^{4}\frac{1}{\zeta_i}\tilde{\Gamma}_i\dot{\hat{\Gamma}}_i + \frac{1}{\gamma}\tilde{m}\dot{\hat{m}} \\
\le & -\lambda_{\min}(K_P)\|P_e\|^2 - (\lambda_{\min}(K_V) - \varsigma_V)\|V_e\|^2 \\
& - (\lambda_{\min}(K_R) - \varsigma_R)\|R_e\|^2 \\
& - (\lambda_{\min}(K_\omega) - \varsigma_\omega)\|\omega_e\|^2
\end{aligned}$$

$$
\begin{aligned}
&-\left(mk_z-\varsigma_z\right)\delta_z^2-k_\psi\psi_e^2-\left(k_r-\varsigma_r\right)r_e^2\\
&+\tilde{\varGamma}_1\left(\frac{1}{2\varepsilon_1^2}V_e^TV_eH_1^T(V_2)H_1(V_2)-\frac{1}{\zeta_1}\dot{\hat{\varGamma}}_1\right)\\
&+\tilde{\varGamma}_2\left(\frac{1}{2\varepsilon_2^2}\omega_e^T\omega_eH_2^T(\omega_2)H_2(\omega_2)-\frac{1}{\zeta_2}\dot{\hat{\varGamma}}_2\right)\\
&+\tilde{\varGamma}_3\left(\frac{1}{2\varepsilon_3^2}\delta_z^2H_3^T(w)H_3(w)c(\theta)c(\phi)-\frac{1}{\zeta_3}\dot{\hat{\varGamma}}_3\right)\\
&+\tilde{\varGamma}_4\left(\frac{1}{2\varepsilon_4^2}r_e^2H_4^T(r)H_4(r)-\frac{1}{\zeta_4}\dot{\hat{\varGamma}}_4\right)\\
&-\tilde{m}\left(g\delta_z-\nu\delta_z-\frac{1}{\gamma}\dot{\hat{m}}\right)+\frac{1}{2}\sum_{i=1}^4\varepsilon_i^2s_i\\
&+\frac{\|\sigma_1(V_2)\|^2}{4\varsigma_V}+\frac{\left\|\tilde{\ddot{\alpha}}_V\right\|^2}{4\varsigma_R}+\frac{\|d_\omega\|^2}{4\varsigma_\omega}\\
&+\frac{(\sigma_3(w)c(\theta)c(\phi))^2}{4\varsigma_z}+\frac{\left(\tilde{\ddot{\alpha}}_\psi+\sigma_4(r)\right)^2}{4\varsigma_r},
\end{aligned}
\tag{73}
$$

where $i=1,2,3,4$; $s_i=\begin{cases}1, & \text{if } i=1,2,4;\\ c(\theta)c(\phi), & \text{if } i=3.\end{cases}$

Input the adaptive laws (69a), (69b), (69c), (69d) and (69e) into (73) and combine the following inequalities

$$
\begin{cases}
\sum_{i=1}^4\frac{\xi_i}{\zeta_i}\tilde{\varGamma}_i\hat{\varGamma}_i\le-\sum_{i=1}^4\frac{\xi_i}{2\zeta_i}\tilde{\varGamma}_i^2+\sum_{i=1}^4\frac{\xi_i}{2\zeta_i}\varGamma_i^2,\\
\eta\tilde{m}\hat{m}\le-\frac{\eta}{2}\tilde{m}^2+\frac{\eta}{2}m^2,
\end{cases}
\tag{74}
$$

to get

$$
\begin{aligned}
\dot{L}\le&-\lambda_{\min}(K_P)\|P_e\|^2\\
&-\left(\lambda_{\min}(K_V)-\varsigma_V\right)\|V_e\|^2\\
&-\left(\lambda_{\min}(K_R)-\varsigma_R\right)\|R_e\|^2\\
&-\left(\lambda_{\min}(K_\omega)-\varsigma_\omega\right)\|\omega_e\|^2\\
&-\left(mk_z-\varsigma_z\right)\delta_z^2-k_\psi\psi_e^2-\left(k_r-\varsigma_r\right)r_e^2\\
&-\sum_{i=1}^4\frac{\xi_i}{2\zeta_i}\tilde{\varGamma}_i^2+\sum_{i=1}^4\frac{\xi_i}{2\zeta_i}\varGamma_i^2-\frac{\eta}{2}\tilde{m}^2+\frac{\eta}{2}m^2\\
&+\frac{\|\sigma_1(V_2)\|^2}{4\varsigma_V}+\frac{\left\|\tilde{\ddot{\alpha}}_V\right\|^2}{4\varsigma_R}\\
&+\frac{\|d_\omega\|^2}{4\varsigma_\omega}+\frac{(\sigma_3(w)c(\theta)c(\phi))^2}{4\varsigma_z}\\
&+\frac{\left(\tilde{\ddot{\alpha}}_\psi+\sigma_4(r)\right)^2}{4\varsigma_r}+\frac{1}{2}\sum_{i=1}^4\varepsilon_i^2s_i\\
\le&-2\varSigma L+\Delta,
\end{aligned}
\tag{75}
$$

where

$$
\begin{aligned}
\varSigma=\min\Bigg(&\lambda_{\min}(K_P),\left(\lambda_{\min}(K_V)-\varsigma_V\right),\\
&\left(\lambda_{\min}(K_R)-\varsigma_R\right),\left(\lambda_{\min}(K_\omega)-\varsigma_\omega\right),\\
&\frac{(mk_z-\varsigma_z)}{m},k_\psi,\left(k_r-\varsigma_r\right),\frac{\xi_i}{2},\frac{\eta\gamma}{2}\Bigg),
\end{aligned}
\tag{76}
$$

and

$$
\begin{aligned}
\Delta=&\sum_{i=1}^4\frac{\xi_i}{2\zeta_i}\varGamma_i^2+\frac{\eta}{2}m^2+\frac{\|\sigma_1(V_2)\|^2}{4\varsigma_V}\\
&+\frac{\left\|\tilde{\ddot{\alpha}}_V\right\|^2}{4\varsigma_R}+\frac{\|d_\omega\|^2}{4\varsigma_\omega}+\frac{(\sigma_3(w)c(\theta)c(\phi))^2}{4\varsigma_z}\\
&+\frac{\left(\tilde{\ddot{\alpha}}_\psi+\sigma_4(r)\right)^2}{4\varsigma_r}+\frac{1}{2}\sum_{i=1}^4\varepsilon_i^2s_i.
\end{aligned}
\tag{77}
$$

Thus, solving the differential inequality (75) yields

$$
L(t)\le L(t_0)e^{-2\varSigma(t-t_0)}+\frac{\Delta}{2\varSigma}\left(1-e^{-2\varSigma(t-t_0)}\right),
\tag{78}
$$

where t_0 is the initial time. And inequality (78) means that

$$
L(t)\to\frac{\Delta}{2\varSigma},\quad \text{as } t\to\infty.
\tag{79}
$$

Defining $\varPi(t)=L(t_0)e^{-2\varSigma(t-t_0)}+\frac{\Delta}{2\varSigma}\left(1-e^{-2\varSigma(t-t_0)}\right)$, then one can get

$$
\|P_e\|\le\sqrt{2\varPi(t)},
\tag{80}
$$

$$
\delta_z\le\sqrt{\frac{2}{m}\varPi(t)},
\tag{81}
$$

$$
\psi_e\le\sqrt{2\varPi(t)}.
\tag{82}
$$

Since $\delta_z=\dot{z}_e+k_zz_e$, the following inequality can be obtained

$$z_e \leq z_e(t_0)\, e^{-k_z(t-t_0)} + \frac{\sqrt{\frac{2}{m}\Pi(t)}}{k_z}\left(1 - e^{-k_z(t-t_0)}\right), \tag{83}$$

which implies that $z_e \to \frac{\sqrt{\frac{2}{m}\Pi(t)}}{k_z}$, as $t \to \infty$.

Therefore, the position and the yaw angle tracking errors of the overall closed-loop system are semi-globally uniformly ultimately bounded and can converge to an arbitrarily small neighborhood of the original point by selecting the proper parameters. □

Remark 4 As shown in inequality (80), (82), and (83), the size of the neighborhood is adjustable by choosing the parameters Σ and k_z. Since $c(\phi)$ and $c(\theta)$ are used in the denominator of the formula (56), Assumption 1 is necessary to guarantee the validity of the formula and the analysis on the stability of the closed-loop system.

Remark 5 Since the autonomous flight of the small-scale helicopter is under the Assumption 1 and the transient response of the dynamics of sensing and actuation systems is much faster than that of the helicopter dynamics [25], the sensing and actuation systems are assumed to be ideal and the dynamics of which can be disregarded during developing a controller.

5 Simulation results and discussion

In this section, the effectiveness and the robustness of the proposed nonlinear adaptive controller augmented by the RBFNNs are evaluated through two simulation experiments. For convenience, the numerical values of the main physical parameters of the small-scale unmanned helicopter are provided in Table 2. Readers interested can refer to the literature [35] for more detailed information. The main controller parameters are the same in the two simulations, which are summarized in Tabel 3. Since the PID controller needs no knowledge of the helicopter model, it is widely used in practical applications in spite of the fact that the controller gains can only be tuned by experience. Thus, in order to be more persuasive, the proposed controller is compared with the PID controller proposed in [25] in both two simulations, which is consist of four single-input single-output PID control loops.

Some model parameter variations and external disturbances are considered in the simulations to verify the robustness of the proposed controller. More specifically, a heavy object is dropped during the flight to test the robustness to the internal disturbances in the first simulation, and the wind gusts of 9 m/s along the body frame x-, y-, and z-axes are added to test the robustness to the external disturbances in the second simulation. Besides, other simulation conditions (e.g., a fixed step size of 0.01 s) are the same for both two simulations.

Table 2 Physical parameters of the unmanned helicopter

Parameter	Value (unit)	Parameter	Value
R	0.785 m	K_{bel}	0.4825
Ω	172.788 rad/s	K_{sb}	1.1959
c_m	0.060 m	K_{mix}	0.8709
D_{vt}	0.90 m	K_a	9.4248
H_{vt}	0.065 m	K_{lat}	0.4434
H_{mr}	0.275 m	K_{lon}	0.4667
g	9.81 N/kg	K_{col}	0.3813
m	7.495 kg	N_r	−14.1674
I_x	0.1895 kg m^2	N_1	2.1629
I_y	0.4515 kg m^2	N_2	137.4115
I_z	0.3408 kg m^2	K_c	−184.6632
I_β	0.0913 kg m^2	k_β	167.6592
I_β^s	0.0058 kg m^2	γ^m	1.3112
ρ	1.290 kg/m^3	γ^s	0.3282

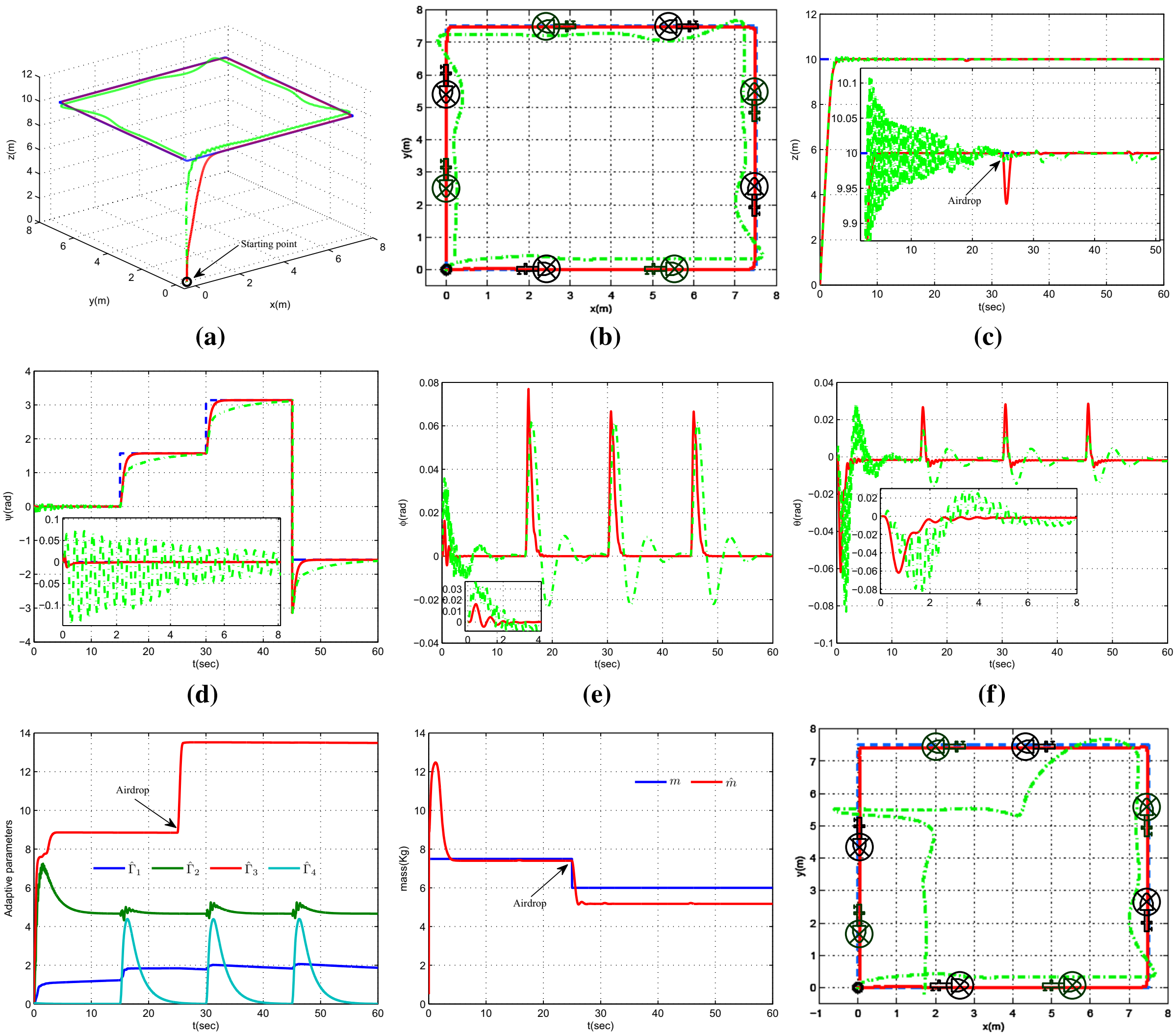
Table 3 Control parameters of the proposed controller

Parameter	Value	Parameter	Value
K_P	diag(1, 1)	k_z	5
K_V	diag(1, 1)	k_ψ	10
K_R	diag(3, 3)	k_r	10
K_ω	diag(150,160)	γ	1

5.1 Simulation 1: trajectory tracking with airdrop

The reference trajectory in the first simulation is denoted by the following square curve

$$P_r = \begin{cases} (0.5t,\ 0,\ 10)^T, & \text{if} \quad t \leq 15; \\ (7.5,\ 0.5(t-15),\ 10)^T, & \text{if} \quad 15 < t \leq 30; \\ (0.5(45-t),\ 7.5,\ 10)^T, & \text{if} \quad 30 < t \leq 45; \\ (0,\ 0.5(60-t),\ 10)^T, & \text{if} \quad 45 < t \leq 60. \end{cases} \tag{84}$$



(a) (b) (c) (d) (e) (f) (g) (h) (i)

Fig. 5 First simulation: the reference trajectory (*blue dashed line*) and the actual trajectory of the proposed nonlinear adaptive controller (*red solid line*), and the PID controller (*green dashed-dotted line*). (Color figure online)

The purpose of the first simulation is mainly to evaluate the tracking accuracy and the robustness to the internal parameter variation considering the intrinsic coupling among different control channels. To enhance the coupling effect, the nose of the small-scale unmanned helicopter is kept forward during the whole flight, producing the following reference trajectory for ψ

$$\psi_r = \begin{cases} 0, & \text{if} \quad t \le 15; \\ \dfrac{1}{2}\pi, & \text{if} \quad 15 < t \le 30; \\ \pi, & \text{if} \quad 30 < t \le 45; \\ \dfrac{3}{2}\pi, & \text{if} \quad 45 < t \le 60. \end{cases} \tag{85}$$

Since the mass is used as a denominator in the control design process, the initial mass cannot be zero or too small. For the sake of fairness, the initial mass is set as 0.5 kg. The actual total mass of the unmanned helicopter is assumed to be 7.495 kg, including an object of 1.495 kg, which is to be dropped off during the flight to trigger an abrupt mass variation. The adaptive parameters are selected as: $\epsilon_1 = 0.75$, $\epsilon_2 = \epsilon_3 = 1$, $\epsilon_4 = 5$, $\zeta_1 = \zeta_2 = 200$, $\zeta_3 = \zeta_4 = 100$, $\xi_1 = 0.005$, $\xi_2 = \xi_4 = 0.5$, $\xi_3 = 0.0001$, and $\eta = 0.005$. The simulation results are shown in Fig. 5.

As shown in Fig. 5a–d, the red solid lines of the proposed controller obviously have higher tracking accuracy than those green dashed-dotted lines of the PID controller despite the airdrop at the 25th second. It is observed from Fig. 5e, f, the roll angle and the pitch angle both deviate from their equilibrium values when the yaw angle changes, resulting from the coupling effect with the yaw channel. However, the roll angle and the pitch angle of the proposed strategy are both stabler than those of the PID method. In addition, Fig. 5g shows the adaptive parameters and Fig. 5h demonstrates that the estimated mass can keep up with the actual value well. The mass estimation is quite valid although there exists a bounded estimation error. Figure 5i depicts the tracking result including the airdrop at the 25th second and the wind gusts of 9 m/s along the body frame x-, y-, and z-axes rising up at the 30th second to further reveal the gap between the two methods more clearly.

Remark 6 As shown in Fig. 5c, when the airdrop happens, the altitude variation of the PID controller is relatively smaller than that of the proposed controller. The result emerges mainly due to the common structure characteristics of the PID controllers which do not need precise model information, while the mass is quite an important model parameter in the proposed controller. However, the proposed strategy makes a rapid recovery from the abrupt height variation and gets a better tracking accuracy than the PID method. This just validates the effectiveness of the mass adaption (69e) and the robustness of the proposed controller.

Remark 7 Fig. 5d depicts that the yaw angle of the nonlinear adaptive controller has a quicker response and a higher tracking precision. It is worth mentioning that the yaw angle changes from π to $-\pi$ at the 45th second in the simulation although the desired yaw angle is $3\pi/2$. This is because that the actual yaw angle range of the small-scale unmanned helicopter is between $-\pi$ and π, and the heading control loop takes it into account, making the simulation closer to a real flight.

Remark 8 As observed in Fig. 5g, h, the adaptive parameter $\hat{\Gamma}_3$ changes most fiercely when the abrupt mass variation happens. This is owing to the fact that the heave subsystem is most seriously affected by the mass variation among all the control channels. Then the adaptive law (69c) reacts fleetly to hold the height of the unmanned helicopter, leading to an abrupt increase in $\hat{\Gamma}_3$ and maintaining the altitude stable again very quickly, which is consistent with the situation in Fig. 5c.

5.2 Simulation 2: trajectory tracking with wind gusts

In this simulation, the desired position trajectory is selected as a spiral path descending along the inertial frame z-axis, which is more aggressive than what is used in the first simulation, defined as follows:

$$P_r = \begin{cases} (0,\ 0,\ -3)^T, & \text{if } t \leq 5; \\ \begin{bmatrix} 5(1-cos(\frac{\pi}{5}(t-5))) \\ -5sin(\frac{\pi}{5}(t-5)) \\ -23+20e^{(-0.06(t-5))} \end{bmatrix}, & \text{if } t \geq 5. \end{cases} \tag{86}$$

And the reference trajectory of the yaw angle is set as $\psi_r = 0$. Moreover, the wind gusts of 9 m/s are exerted along the body frame x-, y-, and z-axes of the unmanned helicopter at the 30th second for both two methods. With the adaptive parameters chosen as: $\epsilon_1 = 0.5, \epsilon_2 = 1.5, \epsilon_3 = 1, \epsilon_4 = 5, \zeta_1 = 300, \zeta_2 = 200, \zeta_3 = \zeta_4 = 100, \xi_1 = 0.1, \xi_2 = \xi_4 = 0.5, \xi_3 = 0.001, \eta = 0.001$, and other parameters selected as the same with Simulation 1, the performance results of Simulation 2 are depicted in Fig. 6.

Figure 6a–f illustrates that the tracking errors of the PID method are bigger than those of the proposed strategy in the absence of wind gusts before the 30th second in the simulation. Once the wind gusts are exerted at the 30th second, the PID controller suffers much larger tracking errors and trajectory deviations, while the proposed controller withstands the test of the wind gusts and keeps highly accurate tracking trajectories throughout the total flight process. In addition, as depicted in Fig. 6g, h, the PID controller needs more aggressive roll and pitch angles than the proposed method to track the same reference trajectory, which means that the proposed controller can obtain more efficient and robust tracking results with less aggressive controls. Furthermore, the adaptive parameters are illustrated in Fig. 6i.

Remark 9 It should be stressed that in Fig. 6d 1 radian is about equal to 57.3°. The yaw angle tracking error of the PID method is more than 5.73° when there are no wind gusts. Once the wind gusts are exerted, the error increases to about 10°, which is quite a large tracking error for a small-scale unmanned helicopter.

Remark 10 From Eq. (86), it can be derived that the time required for a cycle is 10 seconds, which is much faster than the reference trajectory used in [16]. Con-

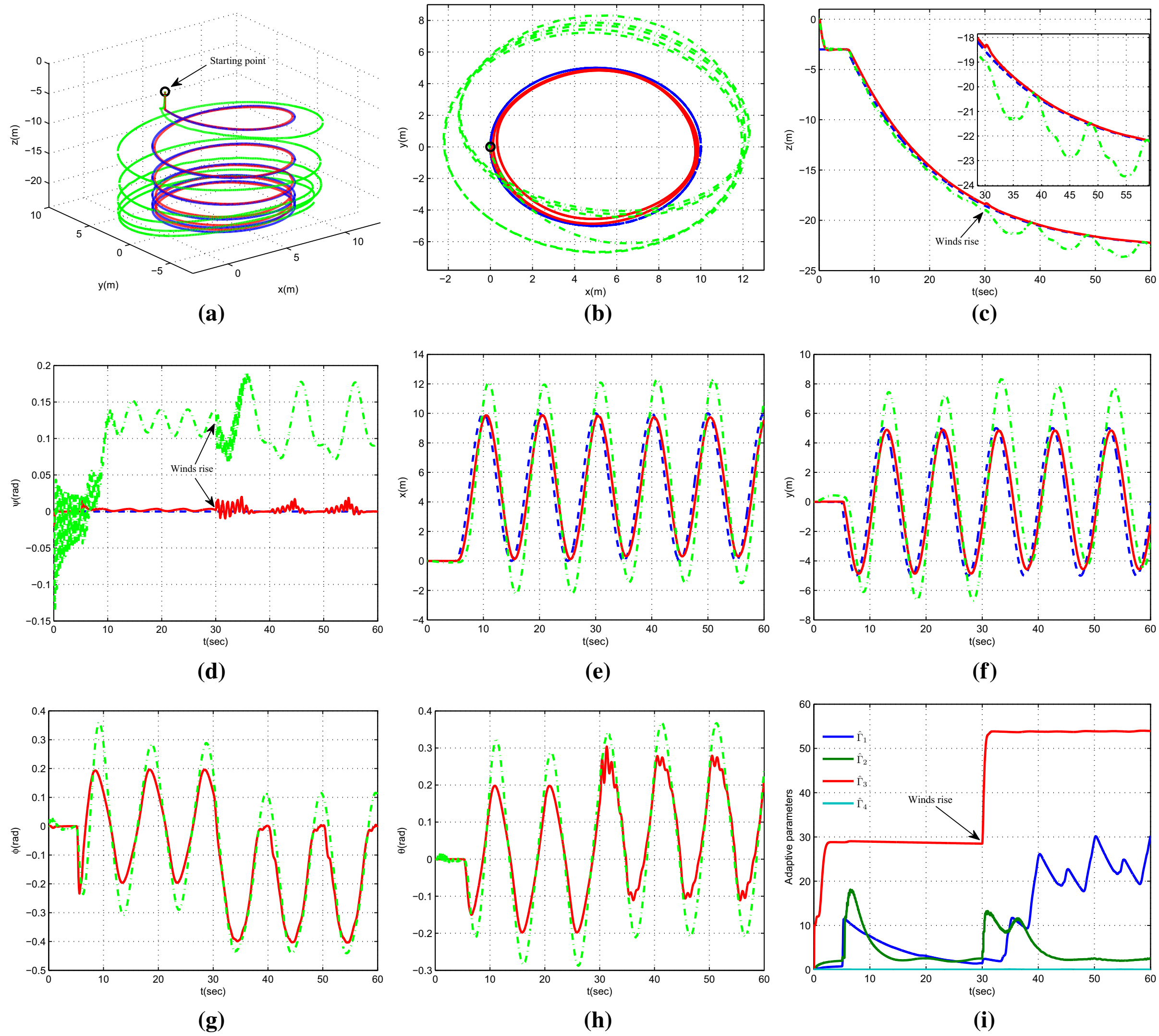


Fig. 6 Second simulation: the reference trajectory (*blue dashed line*) and the actual trajectory of the proposed nonlinear adaptive controller (*red solid line*), and the PID controller (*green dashed-dotted line*). (Color figure online)

sequently, the proposed control system certainly needs bigger roll and pitch angles than those in [16] to maintain a circular motion of such a high frequency.

Remark 11 Fig. 6i shows that the updating law in the heave subsystem experiences the intensest transition, followed by the adaptive laws in the longitudinal–lateral subsystem, and the adaption in the heading subsystem undergoes the slightest change. That is because the area of the rotating main rotor at high speed is much larger than that of the fuselage, while the area of the rotating tail rotor is the smallest, almost negligible. Correspondingly, the larger the area is, the bigger effect the wind gusts produce, then the intenser change the adaption has.

Remark 12 It is worth mentioning that, in an ideal simulation environment, the PID controller can only withstand up to 9 m/s wind gusts along the body frame x-, y-, and z-axes with terrible tracking errors, while the proposed nonlinear adaptive controller can withstand up to 13 m/s wind gusts with perfectly acceptable tracking errors. Moreover, it has to be pointed out that the approximation ability of the RBFNN learning algorithm with the least parameters is limited, and it is the price that must be paid to reduce the number of the updating laws and the amount of computation.

6 Conclusion

This paper addresses the position and yaw tracking problem of a small-scale unmanned helicopter with unknown mass. To estimate the unknown mass, internal uncertainties, and external uncertainties, a novel nonlinear adaptive controller augmented by the RBFNNs is proposed, including a novel mass adaption and four improved adaptive neural network learning algorithms of the least parameters. All the adaptive laws are derived in the process of the Lyapunov stability analysis, by which the position and yaw tracking errors of the overall closed-loop system are proved to converge to an arbitrarily small compact of the origin asymptotically. Moreover, two simulations confirm that the proposed method is not only effective but also robust to the internal and external uncertainties and is suitable for practical implementation. And the future work is to implement the proposed controller on a specific unmanned helicopter experimental system.

Acknowledgements This work was supported in part by the National Natural Science Foundation of China (Grant No. 61403470) and the Basic and Advanced Research Project of ChongQing (Grant No. cstc2016jcyjA0563). The authors of this paper owe great thanks to Dr. Peng Li and Dr. Yadong Liu for their constructive suggestions.

References

1. Cai, G., Chen, B.M., Dong, X., Lee, T.H.: Design and implementation of a robust and nonlinear flight control system for an unmanned helicopter. Mechatronics **21**(5), 803–820 (2011)
2. Cai, G., Chen, B.M., Tong, H.L., Kai, Y.L.: Comprehensive nonlinear modeling of an unmanned-aerial-vehicle helicopter. In: Proceedings of the 2008 AIAA Guidance, Navigation and Control Conference (2008)
3. Calvo-Rolle, J.L., Fontenla-Romero, O., Pérez-Sánchez, B., Guijarro-Berdiñas, B.: Adaptive inverse control using an online learning algorithm for neural networks. Informatica **25**(3), 401–414 (2014)
4. Das, A., Lewis, F., Subbarao, K.: Neural network based robust backstepping control approach for quadrotors. In: Proceedings of the 2008 AIAA Guidance, Navigation and Control Conference, pp. 1–17 (2008)
5. Dierks, T., Jagannathan, S.: Output feedback control of a quadrotor uav using neural networks. IEEE Trans. Neural Netw. **21**(1), 50–66 (2010)
6. Fang, X., Wu, A., Shang, Y., Dong, N.: A novel sliding mode controller for small-scale unmanned helicopters with mismatched disturbance. Nonlinear Dyn. **83**(1), 1053–1068 (2016)
7. Fang, X., Wu, A., Shang, Y., Du, C.: Multivariable super twisting based robust trajectory tracking control for small unmanned helicopter. Math. Probl. Eng. **2015**, 1–13 (2015)
8. Farrell, J.A., Polycarpou, M., Sharma, M., Dong, W.: Command filtered backstepping. IEEE Trans. Autom. Control **54**(6), 1391–1395 (2009)
9. Gadewadikar, J., Chen, B.M., Subbarao, K.: Structured h_∞ command and control-loop design for unmanned helicopters. J, Guid. Control Dyn. **31**(4), 1093–1102 (2008)
10. Gavrilets, V., Mettler, B., Feron, E.: Human-inspired control logic for automated maneuvering of miniature helicopter. J. Guid. Control Dyn. **27**(5), 752–759 (2004)
11. Godbolt, B., Vitzilaios, N.I., Lynch, A.F.: Experimental validation of a helicopter autopilot design using model-based pid control. J. Intell. Robot. Syst. **70**(1), 385–399 (2013)
12. Heffley, R.K., Mnich, M.A.: Minimum-complexity helicopter simulation math model. Technical report usaavscom technical report 87-a-7, NASA, Moffett Field, CA (1988)
13. Islam, S., Liu, P.X., El Saddik, A.: Nonlinear adaptive control for quadrotor flying vehicle. Nonlinear Dyn. **78**(1), 117–133 (2014)
14. Jca, V., Brogliato, B., Dzul, A., Lozano, R.: Nonlinear modelling and control of helicopters. Automatica **39**(9), 1583–1596 (2003)
15. Lai, G., Liu, Z., Zhang, Y., Chen, C.L.: Adaptive position/attitude tracking control of aerial robot with unknown inertial matrix based on a new robust neural identifier. IEEE Trans. Neural Netw. Learn. Syst. **27**(1), 18–31 (2016)
16. Lee, C.T., Tsai, C.C.: Nonlinear adaptive aggressive control using recurrent neural networks for a small scale helicopter. Mechatronics **20**(4), 474–484 (2010)
17. Leonard, F., Martini, A., Abba, G.: Robust nonlinear controls of model-scale helicopters under lateral and vertical wind gusts. IEEE Trans. Control Syst. Technol. **20**(1), 154–163 (2012)
18. Li, T., Feng, G., Zou, Z., Liu, Y.: Robust adaptive fuzzy tracking control for a class of mimo systems: a minimal-learning-parameters algorithm. In: Proceedings of the 2009 American Control Conference, pp. 3106–3111 (2009)
19. Liu, H., Derawi, D., Kim, J., Zhong, Y.: Robust optimal attitude control of hexarotor robotic vehicles. Nonlinear Dyn. **74**(4), 1155–1168 (2013)
20. Liu, H., Lu, G., Zhong, Y.: Robust lqr attitude control of a 3-dof laboratory helicopter for aggressive maneuvers. IEEE Trans. Ind. Electron. **60**(10), 4627–4636 (2013)
21. Liu, J.: Radial Basis Function (RBF) Neural Network Control for Mechanical Systems: Design. Analysis and Matlab Simulation. Springer-Verlag, Berlin Heidelberg (2013)
22. Liu, L., Wang, Z., Zhang, H.: Adaptive fault-tolerant tracking control for mimo discrete-time systems via reinforcement learning algorithm with less learning parameters. IEEE Trans. Autom. Sci. Eng. (2016). doi:10.1109/TASE.2016.2517155
23. Liu, Y.J., Tang, L., Tong, S., Chen, C.L.P., Li, D.J.: Reinforcement learning design-based adaptive tracking control with less learning parameters for nonlinear discrete-time

mimo systems. IEEE Trans. Neural Netw. Learn. Syst. **26**(1), 165–176 (2015)
24. Marconi, L., Naldi, R.: Robust full degree-of-freedom tracking control of a helicopter. Automatica **43**(11), 1909–1920 (2007)
25. Mettler, B.: Identification Modeling and Characteristics of Miniature Rotorcraft. Kluwer Academic Publishers, Norwell (2003)
26. Mettler, B., Dever, C., Feron, E.: Scaling effects and dynamic characteristics of miniature rotorcraft. J Guid. Control Dyn. **27**(3), 466–478 (2004)
27. Nodland, D., Zargarzadeh, H., Jagannathan, S.: Neural network-based optimal adaptive output feedback control of a helicopter uav. IEEE Trans. Neural Netw. Learn. Syst. **24**(7), 1061–1073 (2013)
28. Peng, K., Cai, G., Chen, B.M., Dong, M., Lum, K.Y., Lee, T.H.: Design and implementation of an autonomous flight control law for a uav helicopter. Automatica **45**(10), 2333–2338 (2009)
29. Pounds, P.E.I., Dollar, A.M.: Stability of helicopters in compliant contact under pd-pid control. IEEE Trans. Robot. **30**(6), 1472–1486 (2014)
30. Puttige, V.R., Anavatti, S.G.: Comparison of real-time online and offline neural network models for a uav. In: Proceedings of the 2007 international joint conference on neural networks, pp. 412–417 (2007)
31. Raptis, I.A., Valavanis, K.P., Moreno, W.A.: A novel nonlinear backstepping controller design for helicopters using the rotation matrix. IEEE Trans. Control Syst. Technol. **19**(2), 465–473 (2011)
32. Sheng, S., Sun, C.: An adaptive attitude tracking control approach for an unmanned helicopter with parametric uncertainties and measurement noises. Int. J. Control Autom. Syst. **14**(1), 217–228 (2016)
33. Simplcio, P., Pavel, M., Kampen, E.V., Chu, Q.P.: An acceleration measurements-based approach for helicopter nonlinear flight control using incremental nonlinear dynamic inversion. Control Eng. Pract. **21**(8), 1065–1077 (2013)
34. Tang, S., Zhang, L., Wang, L., Jiang, M.: Nonlinear robust control design of a small-scale helicopter. In: Proceedings of the 2015 Chinese control conference (CCC), pp. 2854–2859 (2015)
35. Tang, S., Zheng, Z., Qian, S., Zhao, X.: Nonlinear system identification of a small-scale unmanned helicopter. Control Eng. Pract. **25**(1), 1–15 (2014)
36. Wang, X., Chen, Y., Lu, G., Zhong, Y.: Robust attitude tracking control of small-scale unmanned helicopter. Int. J. Syst. Sci. **46**(8), 1472–1485 (2015)
37. Xian, B., Diao, C., Zhao, B., Zhang, Y.: Nonlinear robust output feedback tracking control of a quadrotor uav using quaternion representation. Nonlinear Dyn. **79**(4), 2735–2752 (2015)
38. Zhou, B., Zheng, Z., Li, Z., Tang, S.: Implementation of a robust and nonlinear attitude control system for a small-scale unmanned helicopter. In: Proceedings of the 2015 IEEE international conference on information and automation, pp. 2487–2492 (2015)
39. Zhu, B.: Nonlinear adaptive neural network control for a model-scaled unmanned helicopter. Nonlinear Dyn. **78**(3), 1695–1708 (2014)
40. Zhu, B., Zuo, Z.: Approximate analysis for main rotor flapping dynamics of a model-scaled helicopter with bell-hiller stabilizing bar in hovering and vertical flights. Nonlinear Dyn. **85**(3), 1705–1717 (2016)
41. Zou, Y.: Adaptive trajectory tracking control approach for a model-scaled helicopter. Nonlinear Dyn. **83**(4), 2171–2181 (2016)
42. Zou, Y., Zheng, Z.: A robust adaptive rbfnn augmenting backstepping control approach for a model-scaled helicopter. IEEE Trans. Control Syst. Technol. **23**(6), 2344–2352 (2015)