J. Cent. South Univ. (2015) 22: 168−179
DOI: 10.1007/s11771-015-2507-9

Springer

Decentralized PID neural network control for a quadrotor helicopter subjected to wind disturbance

CHEN Yan-min(陈彦民), HE Yong-ling(何勇灵), ZHOU Min-feng(周岷峰)

School of Transportation Science and Engineering, Beihang University, Beijing 100191, China

Abstract: A decentralized PID neural network (PIDNN) control scheme was proposed to a quadrotor helicopter subjected to wind disturbance. First, the dynamic model that considered the effect of wind disturbance was established via Newton−Euler formalism. For quadrotor helicopter flying at low altitude in actual situation, it was more susceptible to be influenced by the turbulent wind field. Therefore, the turbulent wind field was generated according to Dryden model and taken into consideration as the disturbance source of quadrotor helicopter. Then, a nested loop control approach was proposed for the stabilization and navigation problems of the quadrotor subjected to wind disturbance. A decentralized PIDNN controller was designed for the inner loop to stabilize the attitude angle. A conventional PID controller was used for the outer loop in order to generate the reference path to inner loop. Moreover, the connective weights of the PIDNN were trained on-line by error back-propagation method. Furthermore, the initial connective weights were identified according to the principle of PID control theory and the appropriate learning rate was selected by discrete Lyapunov theory in order to ensure the stability. Finally, the simulation results demonstrate that the controller can effectively resist external wind disturbances, and presents good stability, maneuverability and robustness.

Key words: quadrotor helicopter; PID neural network (PIDNN); turbulent wind field; discrete Lyapunov theory

1 Introduction

Recently, as a member of vertical take-off and landing (VTOL) unmanned aerial vehicles (UAVs), the quadrotor helicopter has been more widely used in both military and civilian fields. Compared with fixed-wing aircrafts, quadrotors can fly at low altitude and hover at set point. Compared with traditional helicopters, quadrotors have several advantages including simple mechanical structure, good maneuverability, small size, low cost and strong concealment. These excellent features make quadrotors be able to perform in constrained area with more effectiveness and reliability.

The quadrotor helicopter is a highly nonlinear, strongly coupled and underactuated system; moreover, it is greatly influenced by external disturbances [1]. In the relevant literatures, a lot of work has been done to deal with the control problem of quadrotor helicopter. Adaptive control methods were used [2−4]. MOREL and LEONESSA [2] proposed a model reference adaptive control algorithm to solve the trajectory tracking problem. RASHID and AKHITAR [3] designed an adaptive feedback controller for asymptotic attitude and altitude stabilization. MOHAMMADI and SHAHR [4] developed a decentralized adaptive controller based on improved Lyapunov-based model reference adaptive control technique that did not need knowing the value of any physical parameter for generating appropriate control signals. Backstepping control approaches have been applied in Refs. [5−7]. The literature uses the step-by-step iteration characteristics of backstepping approach to reduce the on-line calculation time and solve the model uncertainties and coupling problems of quadrotor helicopter. The robust control strategies have been adopted in Refs. [8−9]. NICOL et al [9] designed a direct approximate-adaptive controller for an experimental prototype quadrotor so as to achieve both adaptation to unknown payloads and robustness to disturbances. Sliding mode control methods were introduced into quadrotor control [10−13]. This method has several advantages such as fast response, simple physical implementation and no need of system identification.

The PID control is one of the early developed control strategies. PID controller is still widely used in industrial process control nowadays for its simple structure, easy tuning, and excellent performance [14]. BOUABDALLAH et al [15] applied conventional PID method to quadrotor controller. SU et al [16] designed a globally stabilizing nonlinear PID attitude controller in order to reject the external disturbances. Though PID control is simple and dependable, it is difficult to be applied to the system with model uncertainty and unable

Foundation item: Project(2011ZA51001) supported by National Aerospace Science Foundation of China
Received date: 2013−09−06; **Accepted date:** 2013−12−25
Corresponding author: HE Yong-ling, Professor; Tel: +86−13718946286; E-mail: xkbhe@buaa.edu.cn

to obtain the transfer function. PID control is difficult to fulfill the performance requirements since quadrotor is a highly nonlinear and underactuated system.

Artificial neural network has several features such as nonlinear, self-learning and adaptability. Artificial neural network control is one of the intelligent control strategies. It takes artificial neural network as controller or identifier in order to solve the control problem of nonlinear and uncertain system in uncertain environment. DIERKS and JAGANNATHAN [17] proposed a output feedback control scheme using neural network in which the neural network was used to learn the nonlinear dynamics of the quadrotor online including uncertain nonlinear terms like aerodynamic friction and blade flapping. NICOL et al [18] developed an adaptive neural network controller to stabilize a quadrotor against modeling error and considerable wind disturbance. But the performance of neural network control is limited for its long learning time, uncertain network structure and large calculation.

Recently, the hybrid systems of PID control and artificial neural networks control have been widely studied [19−21]. Commonly, the artificial neural network is used to tune the control gains of PID controllers automatically. Therefore, the hybrid systems are able to improve the performance of conventional PID controller for it can deal with highly nonlinear and time-varying systems with system parameter variations and external disturbances. But the problems caused by artificial neural network such as long learning time, uncertain network structure and large calculation are produced at the same time. PID neural network (PIDNN) is a novel control strategy that includes the advantages of PID control and neural network control. It is not a simple combination of the two control schemes but defines neurons of proportional, integral, differential function so as to merge the PID algorithm into neural network control [22−23]. The advantages of PIDNN controller are as follows: 1) the response is faster than that of artificial neural network; 2) the online adaptability is stronger than that of conventional PID controller; and 3) the implementation is simple. According to above reasons, this study designs a controller based on PIDNN control scheme in order to solve the control problem of a quadrotor under influence of wind disturbance.

Due to friction, swirl and other reasons, wind often appears accompanied with turbulence [24]. There is usually turbulent wind field acting on quadrotor because the flying altitude is low in real conditions. To make the simulation results accord with the practical situation, the turbulent wind field is generated according to Dryden model and taken into consideration as the disturbance source of the quadrotor helicopter.

2 Dynamic model of quadrotor

The quadrotor is composed of body and four rotors, as presented in Fig. 1. The thrust force is generated by four rotors. The motion of quadrotor is controlled by varying the rotation speed of four rotors to change the thrust and torque produced by each one. Four rotors are divided into two pairs: pair (1, 3) and pair (2, 4). The rotate direction of the two pairs is contrary in order to counteract the aerodynamic torque generated by rotors' rotation. Increase or decreasing the rotation speed of the four rotors simultaneously will generate vertical motion. Independently varying the speed of the rotor pair (1, 3) can control the pitch angle about y-axis and the translational motion along x-axis. Accordingly, independently varying the speed of the rotor pair (2, 4) can control the roll angle about x-axis and the translational motion along y-axis. The yaw angle about z-axis is determined by the yaw torque which is the sum of the reaction torques generated by each rotor.

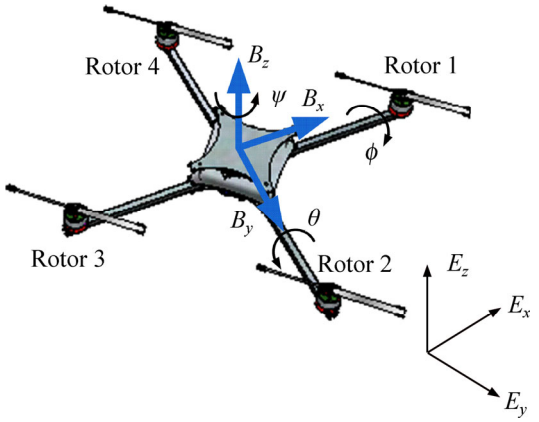

Fig. 1 Structure of quadrotor helicopter

It is assumed that the quadrotor is a rigid-body structure and is completely symmetrical. Two reference frames are set up: the earth-fixed reference frame $E=\{E_x, E_y, E_z\}$ and the body-fixed reference frame $B=\{B_x, B_y, B_z\}$. Attitude angle $\boldsymbol{\Theta}=[\phi, \theta, \psi]^{\mathrm{T}}$ is called roll angel ($-\pi/2<\phi<-\pi/2$), pitch angle ($-\pi/2<\theta<\pi/2$) and yaw angle ($-\pi<\psi<\pi$).

In real conditions, quadrotor is generally exposed to wind disturbance, and this disturbance would cause the aircraft subjected to an additional wind force F_{W}. Because the contact surface between quadrotor's body and air is small; therefore, the impact of wind disturbance on body is ignored and only considers the wind disturbance acting on rotor.

2.1 Rotor aerodynamics

The aerodynamic analysis of rotor subjected to wind disturbance is shown in Fig. 2, where $\boldsymbol{V}_{\mathrm{d}}$ is the induced

velocity; V_w is the wind velocity; $\hat{V}$ is the total induced velocity and could be expressed as

$$\hat{\boldsymbol{V}} = \boldsymbol{V}_d + \boldsymbol{V}_w \tag{1}$$

The induced velocity $\boldsymbol{V}_d$ is defined as

$$\|\boldsymbol{V}_d\| = \sqrt{\frac{\|\boldsymbol{F}_T\|}{2\rho A}} \tag{2}$$

where $\boldsymbol{F}_T$ is the rotor thrust; ρ is the air density; A is the propeller area.

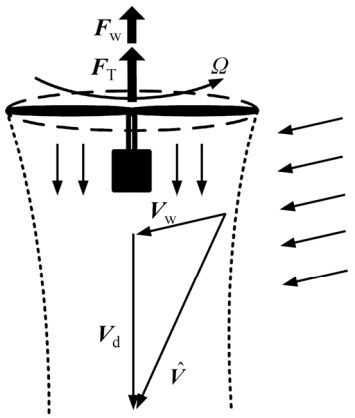

Fig. 2 Aerodynamic analysis of rotor subjected to wind disturbance

Rotor thrust $\boldsymbol{F}_T$ and torque $\boldsymbol{M}_T$ are proportional to the square of rotor speed $\boldsymbol{\Omega}$:

$$\begin{aligned} \boldsymbol{F}_T &= b\cdot\boldsymbol{\Omega}^2 \\ \boldsymbol{M}_T &= d\cdot\boldsymbol{\Omega}^2 \end{aligned} \tag{3}$$

where b and d are thrust coefficient and drag coefficient, respectively.

When the rotor is subjected to wind disturbance, the total aerodynamic force $\boldsymbol{F}_R$ is the sum of rotor thrust $\boldsymbol{F}_T$ and wind force $\boldsymbol{F}_W$.

$$\boldsymbol{F}_R = \boldsymbol{F}_T + \boldsymbol{F}_W = 2\rho A \boldsymbol{V}_d \hat{\boldsymbol{V}} \tag{4}$$

It can be noticed that when there is no wind disturbance, $\hat{\boldsymbol{V}} = \boldsymbol{V}_d$; then, this gives $\boldsymbol{F}_W = 0$, and Eq. (4) becomes $\boldsymbol{F}_R$=$\boldsymbol{F}_T$.

When the rotor is disturbed by wind field, the rotor's aerodynamic torque $\boldsymbol{M}_R$ is defined as

$$\boldsymbol{M}_R = \boldsymbol{M}_W + \boldsymbol{M}_T = k_{drag}\hat{\boldsymbol{V}}^2 \tag{5}$$

where k_{drag}>0 is a coefficient related to air density, rotor radius and rotor shape.

2.2 System general forces and moments

Translational dynamic equation and rotational dynamic equation is established according to Newton−Euler formalism.

The translational dynamic equation is established in reference frame $\boldsymbol{E}$:

$$m\ddot{\boldsymbol{X}} = \boldsymbol{R}\left(\sum_{i=1}^{4}\boldsymbol{F}_{T_i} + \sum_{i=1}^{4}\boldsymbol{F}_{W_i}\right) + m\boldsymbol{G} \tag{6}$$

where $\boldsymbol{X}$=[x, y, z]^T is the quadrotor's position; m is the mass; $\boldsymbol{R}$ is the transformation matrix; $\boldsymbol{F}_{T_i}$ and $\boldsymbol{F}_{W_i}$ are the thrust and wind forces of the i-th rotor; $\boldsymbol{G}$=[0, 0, $-g$]^T is the gravity acceleration.

The rotational dynamic equation in reference frame $\boldsymbol{B}$ is established as follows:

$$\boldsymbol{J}\dot{\boldsymbol{\omega}} = -\boldsymbol{\omega}\times\boldsymbol{J}\boldsymbol{\omega} + \boldsymbol{\omega}\times[0,0,J_r\Omega_r] + \boldsymbol{M}_B + \boldsymbol{M}_w \tag{7}$$

where $\boldsymbol{J}$ is the diagonal inertia matrix; $\boldsymbol{\omega}$=[p, q, r]^T is the angular velocity; $\boldsymbol{J}_r$ is the rotor inertia; Ω_r is the relative speed defined as

$$\Omega_r = -\Omega_1 + \Omega_2 - \Omega_3 + \Omega_4 \tag{8}$$

The term $\boldsymbol{\omega}\times[0, 0, J_r\Omega_r]$ in Eq. (7) represents the gyroscopic moment. $\boldsymbol{M}_B$ is the moments caused by rotor thrust:

$$\boldsymbol{M}_B = \begin{bmatrix} l(-F_{T_2}^2 + F_{T_4}^2) \\ l(F_{T_1}^2 - F_{T_3}^2) \\ M_{T_1} - M_{T_2} + M_{T_3} - M_{T_4} \end{bmatrix} \tag{9}$$

where l represents the distance between motor and aircraft center. $\boldsymbol{M}_w$ is the moment caused by wind force:

$$\boldsymbol{M}_w = \begin{bmatrix} l(-F_{w_2}^2 + F_{w_4}^2) \\ l(F_{w_1}^2 - F_{w_3}^2) \\ M_{W_1} - M_{W_2} + M_{W_3} - M_{W_4} \end{bmatrix} \tag{10}$$

From Eqs. (6−10), the dynamic equations are obtained:

$$\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \\ \hline \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} = \begin{bmatrix} U_1\cdot(C_\phi S_\theta C_\psi + S_\phi S_\psi)/m \\ U_1\cdot(C_\phi S_\theta S_\psi - S_\phi C_\psi)/m \\ U_1\cdot(C_\phi C_\theta)/m - g \\ \hline \dot{\theta}\dot{\psi}(I_{yy} - I_{zz})/I_{xx} + \dot{\theta}\Omega_r J_r/I_{xx} + U_2\cdot l/I_{xx} \\ \dot{\phi}\dot{\psi}(I_{zz} - I_{xx})/I_{yy} - \dot{\phi}\Omega_r J_r/I_{yy} + U_3\cdot l/I_{yy} \\ \dot{\phi}\dot{\psi}(I_{xx} - I_{yy})/I_{zz} + U_4\cdot l/I_{zz} \end{bmatrix} + \begin{bmatrix} W_1 \\ W_2 \\ W_3 \\ \hline W_4 \\ W_5 \\ W_6 \end{bmatrix} \tag{11}$$

where $\boldsymbol{U}$=[U_1, U_2, U_3, U_4]^T is the control input vector and described as follows:

$$\boldsymbol{U} = \begin{bmatrix} b(\Omega_1^2 + \Omega_2^2 + \Omega_3^2 + \Omega_4^2) \\ b(-\Omega_2^2 + \Omega_4^2) \\ b(\Omega_1^2 - \Omega_3^2) \\ d(\Omega_1^2 - \Omega_2^2 + \Omega_3^2 - \Omega_4^2) \end{bmatrix} \tag{12}$$

The wind disturbance vector $\boldsymbol{W}=[W_1, W_2, W_3, W_4, W_5, W_6]^{\mathrm{T}}$ is defined as

$$\boldsymbol{W}=\begin{bmatrix} R_x\cdot\sum_{i=1}^{4}F_{\mathrm{W}_i}\Big/m \\ R_y\cdot\sum_{i=1}^{4}F_{\mathrm{W}_i}\Big/m \\ R_z\cdot\sum_{i=1}^{4}F_{\mathrm{W}_i}\Big/m \\ l(-F_{\mathrm{w}_2}^2+F_{\mathrm{w}_4}^2)\Big/I_{xx} \\ l(F_{\mathrm{w}_1}^2-F_{\mathrm{w}_3}^2)\Big/I_{yy} \\ (M_{\mathrm{R}_1}-M_{\mathrm{R}_2}+M_{\mathrm{R}_3}-M_{\mathrm{R}_4})\Big/I_{zz} \end{bmatrix} \tag{13}$$

3 Model of turbulent wind field

Theoretically, the precise turbulent wind field can be obtained via solving N−S equation. But the computational cost is excessive, so it is almost impossible in real conditions. Therefore, the turbulent wind field should be described by establishing appropriate model based on random theory.

The turbulent wind field is described according to Dryden model in this work. The Dryden form of the power spectral densities for the turbulence velocities is as follows [25]:

$$\begin{aligned} \Phi_{\mathrm{u}}(\omega)&=\sigma_{\mathrm{u}}^2\frac{L_u}{\pi V_x}\frac{1}{1+\left(L_{\mathrm{u}}\omega/V_x\right)^2} \\ \Phi_{\mathrm{v}}(\omega)&=\sigma_{\mathrm{v}}^2\frac{L_v}{\pi V_y}\frac{1+12\left(L_{\mathrm{v}}\omega/V_y\right)^2}{\left[1+4\left(L_{\mathrm{v}}\omega/V_y\right)^2\right]^2} \\ \Phi_{\mathrm{w}}(\omega)&=\sigma_{\mathrm{w}}^2\frac{L_w}{\pi V_z}\frac{1+12\left(L_{\mathrm{w}}\omega/V_z\right)^2}{\left[1+4\left(L_{\mathrm{w}}\omega/V_z\right)^2\right]^2} \end{aligned} \tag{14}$$

where ω is the time-frequency; V is the air velocity; L is the scale length for power spectra; σ is the standard deviation. For quadrotor flying at low altitude, according to Ref. [24], the scale length and standard deviation are as follows:

$$\begin{cases} 2L_{\mathrm{w}}=h \\ L_{\mathrm{u}}=2L_{\mathrm{v}}=\dfrac{h}{\left(0.177+0.000823h\right)^{1.2}} \\ \sigma_{\mathrm{w}}=0.1u_{20} \\ \dfrac{\sigma_{\mathrm{u}}}{\sigma_{\mathrm{w}}}=\dfrac{\sigma_{\mathrm{v}}}{\sigma_{\mathrm{u}}}=\dfrac{1}{\left(0.177+0.000823h\right)^{0.4}} \end{cases} \tag{15}$$

where u_{20} is the 20 feet wind velocity and h is the flying altitude.

Let the unit intensity white noise $n(t)$ through a filter $G(s)$ and then generate signal $x(t)$, the spectrum function of $x(t)$ is:

$$\Phi(\omega)=\left|G(i\omega)\right|^2=G^*(i\omega)G(i\omega) \tag{16}$$

The transfer function of the shaping filter can be obtained from Eqs. (14) and (16):

$$\begin{cases} G_{\mathrm{u}}(s)=\dfrac{K_{\mathrm{u}}}{T_{\mathrm{u}}s+1} \\ G_{\mathrm{v}}(s)=\dfrac{K_{\mathrm{v}}}{T_{\mathrm{v}}s+1} \\ G_{\mathrm{w}}(s)=\dfrac{K_{\mathrm{w}}}{T_{\mathrm{w}}s+1} \end{cases} \tag{17}$$

where $K_{\mathrm{u}}=\sigma_{\mathrm{u}}\sqrt{L_{\mathrm{u}}/(\pi V)}$, $T_{\mathrm{u}}=L_{\mathrm{u}}/V$, $K_{\mathrm{v}}=\sigma_{\mathrm{v}}\sqrt{L_{\mathrm{v}}/(\pi V)}$, $T_{\mathrm{v}}=2L_{\mathrm{v}}/\sqrt{3}V$, $K_{\mathrm{w}}=\sigma_{\mathrm{w}}\sqrt{L_{\mathrm{w}}/(\pi V)}$, $T_{\mathrm{w}}=\dfrac{2L_{\mathrm{w}}}{\sqrt{3}V}$.

4 Control strategy

The controller is nested by inner loop and outer loop, as shown in Fig. 3. The PIDNN control approach is used in the inner loop in order to achieve attitude control, while the PID control algorithm is implemented in the

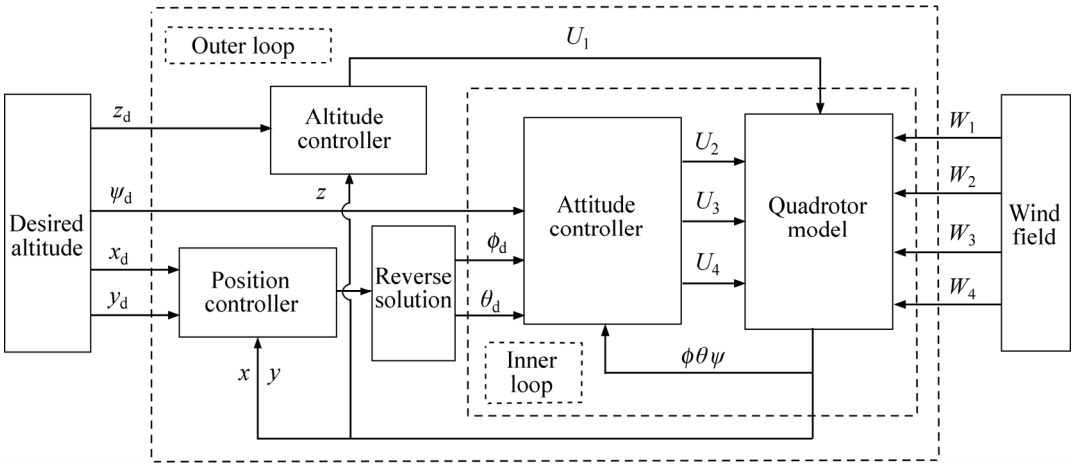

Fig. 3 Structure of controller

outer loop for the altitude and position control. It can be known from the flight characteristic of quadrotor that the position movement in outer loop is determined by the attitude angle in inner loop. Altitude controller outputs U_1 according to desired altitude (z_d) and current altitude (z). Position controller combines the desired position (x_d, y_d) and current position (x, y), outputting desired roll angle (ϕ_d) and desired pitch angle (θ_d) into attitude controller. And then, attitude controller outputs U_2, U_3 and U_4 to quadrotor model. Finally, the model integrates the disturbance from wind field, outputs the state of next time step and feeds back to all controllers.

4.1 Altitude control

From the altitude equation defined in Eq. (11):

$$\ddot{z} = U_1 \cdot (C_\phi C_\theta)/m - g + W_3 \tag{18}$$

the control input U_1 can be derived as

$$U_1 = \frac{\mathrm{m}}{C_\phi C_\theta}(P_z + g - W_3) \tag{19}$$

The necessary condition of Eq. (19) is $C_\phi C_\theta \neq 0$. P_z is selected as a PID controller:

$$P_z = k_{z\mathrm{P}}(z_\mathrm{d} - z) + k_{z\mathrm{P}}\sum_{i=1}^{k} t_\mathrm{s} \times (z_{\mathrm{d}i} - z_\mathrm{d}) + k_{z\mathrm{D}}(\dot{z}_\mathrm{d} - \dot{z}) \tag{20}$$

where $k_{z\mathrm{P}}$, $k_{z\mathrm{I}}$ and $k_{z\mathrm{D}}$ are the proportional, integral and differential coefficients, respectively; k represents the k th iteration; and t_s is the time step.

4.2 Position control

Taking U_1 into position equation of x- and y-directions defined in Eq. (11), and adopting the yaw angle approximate to 0, the dynamics can be obtained as

$$P_x = (P_z + g - W_3)\tan\theta + W_1 \tag{21}$$

$$P_y = -(P_z + g - W_3)\tan\phi + W_2 \tag{22}$$

The PID controller of x- and y-direction can be designed as

$$P_x = k_{x\mathrm{P}}(x_\mathrm{d} - x) + k_{x\mathrm{I}}\sum_{i=1}^{k} ts \times (x_{\mathrm{d}i} - x_i) + k_{x\mathrm{D}}(\dot{x}_\mathrm{d} - \dot{x}) \tag{23}$$

$$P_y = k_{y\mathrm{P}}(y_\mathrm{d} - y) + k_{y\mathrm{I}}\sum_{i=1}^{k} ts \times (y_{\mathrm{d}i} - y_i) + k_{y\mathrm{D}}(\dot{y}_\mathrm{d} - \dot{y}) \tag{24}$$

Reversely solving Eqs. (21) and (22), the desired attitude angles ϕ_d and θ_d can be derived as

$$\phi_\mathrm{d} = -\arctan\left(\frac{P_y - W_2}{P_z + g - W_3}\right) \tag{25}$$

$$\theta_\mathrm{d} = \arctan\left(\frac{P_x - W_1}{P_z + g - W_3}\right) \tag{26}$$

4.3 Attitude control

Attitude control is the core of quadrotor control whose performance would greatly affect the system stability. For PIDNN controller has good transient and steady performance, and is robust to external disturbance and inherent uncertainty, it is introduced as attitude controller.

Attitude control includes the control of pitch, roll and yaw angles. If a centralized controller is used, the algorithm would be very complicated. In order to achieve efficient control, a sub PIDNN controller is designed for each attitude angle respectively. These three PIDNN sub controllers construct the decentralized attitude controller and control the three attitude angles separately and simultaneously. Compared with centralized controller, decentralized controller can reduce computation burden, increase system response and simplify controller design.

4.3.1 Network structure

The network structure of PIDNN controller for roll angle is shown in Fig. 4. The structure is a three-layer feed-forward neural network and consists of input layer, hidden layer and output layer.

1) Input layer

The input layer has two neurons: $x_1 = \phi(k)$ and $x_2 = \phi_\mathrm{d}(k)$ where $\phi(k)$ is the roll angle; $\phi_\mathrm{d}(k)$ is the

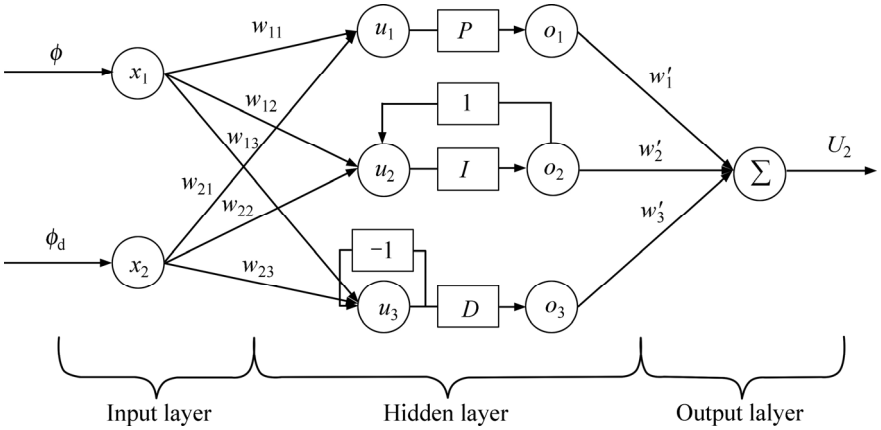

Fig. 4 Network structure of PIDNN controller for roll angle

desired roll angle and k represents the iteration times.

2) Hidden layer

The hidden layer includes three neurons which perform PID functions. P neuron performs the proportional function, I neuron performs the integral function, and D neuron performs derivative function. The inputs of the hidden layer are as follows:

$$u_j(k)=\sum_{i=1}^{2} w_{ij}(k)\cdot x_i(k),\ \ j=1,\ 2,\ 3 \tag{27}$$

where $w_{ij}(k)$ is the connective weight between input layer and hidden layer. The outputs of the hidden layer are as follows:

$$\begin{cases} o_1(k)=u_1(k)=\sum_{i=1}^{2} w_{i1}(k)x_i(k) \\ o_2(k)=u_2(k)+o_2(k-1)=\sum_{i=1}^{2} w_{i2}(k)x_i(k)+o_2(k-1) \\ o_3(k)=u_3(k)-u_3(k-1) \\ \qquad =\sum_{i=1}^{2} w_{i3}(k)x_i(k)-\sum_{i=1}^{2} w_{i3}(k-1)x_i(k-1) \end{cases} \tag{28}$$

3) Output layer

There is only one neuron in output layer. The function of this layer is to output the control input $U_2(k)$ for roll angle. $U_2(k)$ can be calculated as

$$U_2(k)=\sum_{j=1}^{3} w'_j(k)o_j(k) \tag{29}$$

where $w'_j(k)$ is the connective weight between hidden layer and output layer.

4.3.2 On-line learning algorithm

The network is trained on-line by error back-propagation algorithm. The algorithm is based on gradient descent method and modifies the connective weights to minimize target function $E(k)$. The target function $E(k)$ is defined as

$$E(k)=\frac{1}{2}e_\phi^2(k) \tag{30}$$

where $e_\phi(k)=\phi_{\mathrm{d}}(k)-\phi(k)$ is the tracking error of roll angle. The on-line learning algorithm is described as follows.

1) Hidden layer to output layer

The connective weight from hidden layer to output layer is updated according to

$$w'_j(k+1)=w'_j(k)-\eta_j\cdot\Delta w'_j(k) \tag{31}$$

where η_j is the learning rate of w'_j. According to BP algorithm, the weight is updated by

$$\begin{aligned}\Delta w'_j(k)&=\frac{\partial E(k)}{\partial w'_j(k)}=\frac{\partial E(k)}{\partial U_2(k)}\frac{\partial U_2(k)}{\partial w'_j(k)}\\ &=\frac{\partial E(k)}{\partial e_\phi(k)}\frac{\partial e_\phi(k)}{\partial \phi(k)}\frac{\partial \phi(k)}{\partial U_2(k)}\frac{\partial U_2(k)}{\partial w'_j(k)}\end{aligned} \tag{32}$$

Substituting Eq. (28) and (31) into Eq. (32), let $\delta(k)=\partial\phi(k)/\partial U_2(k)$, the equation can be rewritten as

$$\Delta w'_j(k)=-e_\phi(k)o_j(k)\delta(k) \tag{33}$$

2) Input layer to hidden layer

The connective weight from input layer to hidden layer is updated according to

$$w_{ij}(k+1)=w_{ij}(k)-\eta_i\cdot\Delta w_{ij}(k) \tag{34}$$

where η_i is the learning rate of w_{ij} and the weight is updated by

$$\begin{aligned}\Delta w_{ij}(k)&=\frac{\partial E(k)}{\partial w_{ij}(k)}\\ &=\frac{\partial E(k)}{\partial U_2(k)}\frac{\partial U_2(k)}{\partial o_j(k)}\frac{\partial o_j(k)}{\partial u_j(k)}\frac{\partial u_j(k)}{\partial w_{ij}(k)}\end{aligned} \tag{35}$$

From Eqs. (27), (28), (32) and (33), $\Delta w_{ij}(k)$ can be derived as

$$\Delta w_{ij}(k)=-e_\phi(k)w'_j(k)x_i(k)\delta(k) \tag{36}$$

The exact value of $\delta(k)$ in Eqs. (33) and (36) is difficult to be obtained, so a simplification is made:

$$\delta(k)=\mathrm{sgn}\frac{\phi(k)-\phi(k-1)}{U_2(k)-U_2(k-1)} \tag{37}$$

4.3.3 Stability analysis of PIDNN controller

The stability of PIDNN controller is determined by initial stability and system convergence. The initial stability is related to the initial connective weights of neural network and the system convergence is based on the selection of learning rates.

1) Selection of initial connective weights

The initial connective weights are selected based on PID control theory. The connective weights are chosen between input layer and hidden layer as

$$w_{1j}=+1,\ \ w_{2j}=-1,\ j=1,\ 2,\ 3 \tag{38}$$

The connective weights are chosen between hidden layer and output layer as

$$w'_1=K_{\mathrm{P}},\ w'_2=K_{\mathrm{I}},\ w'_3=K_{\mathrm{D}} \tag{39}$$

where K_{P}, K_{I} and K_{D} are the coefficients of a PID controller.

The input of hidden layer can be obtained according to the initial value:

$$u_j(k)=\sum_{i=1}^{2}w_{ij}x_i(k)$$
$$=\phi_{\mathrm{d}}(k)-\phi(k)=e_\phi(k),\ j=1,2,3 \tag{40}$$

The output of hidden layer is:

$$\begin{cases}o_1(k)=u_1(k)=e_\phi(k)\\ o_2(k)=\sum_{i=0}^{k}u_2(i)=\sum_{i=0}^{k}e_\phi(i)\\ o_3(k)=u_3(k)-u_3(k-1)=e_\phi(k)-e_\phi(k-1)\end{cases} \tag{41}$$

The initial network output can be derived based on Eq. (29):

$$U_2(k)=K_{\mathrm{P}}e_\phi(k)+K_{\mathrm{I}}\sum_{i=0}^{k}e_\phi(i)+K_{\mathrm{D}}\left[e_\phi(k)-e_\phi(k-1)\right] \tag{42}$$

It can be known from Eq. (42) that the PIDNN controller is equal to a conventional PID controller if the initial connective weights are selected as above. In this way, the PIDNN controller operates like a conventional PID control at the first time of operation and that will ensure the initial stability.

2) Selection of learning rates

The target $E(k)=\frac{1}{2}e_\phi^{\,2}(k)\geq 0$ is chosen as the discrete Lyapunov function. The update algorithm of connective weight is:

$$W(k+1)=W(k)-\eta\frac{\partial E}{\partial W} \tag{43}$$

where W represents the connective weight.

$e_\phi(k+1)$ can be described as

$$e_\phi(k+1)=e_\phi(k)+\Delta e_\phi(k) \tag{44}$$

where variable quantity $\Delta e_\phi(k)$ is defined as

$$\Delta e_\phi(k)=\frac{\partial e_\phi(k)}{\partial W}\Delta W \tag{45}$$

where

$$\Delta W=-\eta\frac{\partial E}{\partial W}=-\frac{1}{2}\eta\frac{\partial(e_\phi^{\,2}(k))}{\partial W}=-\eta e_\phi(k)\frac{\partial(e_\phi(k))}{\partial W} \tag{46}$$

From Eqs. (44)−(46) the variable quantity of $E(k)$ can be derived:

$$\Delta E(k)=\frac{1}{2}\left[e_\phi^{\,2}(k+1)-e_\phi^{\,2}(k)\right]$$
$$=\frac{1}{2}\eta\left[e_\phi(k)\frac{\partial(e_\phi(k))}{\partial W}\right]^2\left\{\eta\left[\frac{\partial(e_\phi(k))}{\partial W}\right]^2-2\right\} \tag{47}$$

In Eq. (47), $\eta>0$, let

$$\eta\left[\frac{\partial(e_\phi(k))}{\partial W}\right]^2-2<0 \tag{48}$$

This would make $\Delta E(k)<0$ ensure the system convergence. Solving the inequality (48), the range of η is derived as

$$0<\eta<\frac{2}{\left[\frac{\partial(e_\phi(k))}{\partial W}\right]^2} \tag{49}$$

where

$$\frac{\partial(e_\phi(k))}{\partial W}=\frac{\partial(\sqrt{2E})}{\partial W}=\frac{1}{2\sqrt{2E}}\frac{\partial E}{\partial W} \tag{50}$$

Let $\varepsilon=\frac{1}{2\sqrt{2E}}\frac{\partial E}{\partial W}$, η can be described as

$$0<\eta<\frac{2}{\varepsilon^2} \tag{51}$$

Substituting the connective weight w_j' between hidden layer and output layer into Eq. (50):

$$\varepsilon_j=\frac{1}{2\sqrt{2E}}\frac{\partial E}{\partial w_j'} \tag{52}$$

and combining with Eq. (32), we can obtain:

$$\varepsilon_j=-\frac{1}{2}o_j(k)\delta(k) \tag{53}$$

The learning rate of connective weight between hidden layer and output layer is as follows:

$$0<\eta_j<\frac{8}{\left[o_j(k)\delta(k)\right]^2} \tag{54}$$

Substituting the connective weight w_{ij} between input layer and hidden layer into Eq. (50):

$$\varepsilon_i=\frac{1}{2\sqrt{2E}}\frac{\partial E}{\partial w_{ij}} \tag{55}$$

and combining with Eq. (35), we can obtain:

$$\varepsilon_i=-\frac{1}{2}w_j'(k)x_i(k)\delta(k) \tag{56}$$

The learning rate of connective weight between input layer and hidden layer is as follows:

$$0<\eta_i<\frac{8}{\left[w_j'(k)x_i(k)\delta(k)\right]^2} \tag{57}$$

The tracking error $e_\phi\to 0$ as $t\to\infty$ if the learning rates are selected according to Eqs. (54) and (57) and the system convergence can be guaranteed.

The structure of PIDNN controller for roll angle is shown in Fig. 5. The controllers for pitch angle and yaw

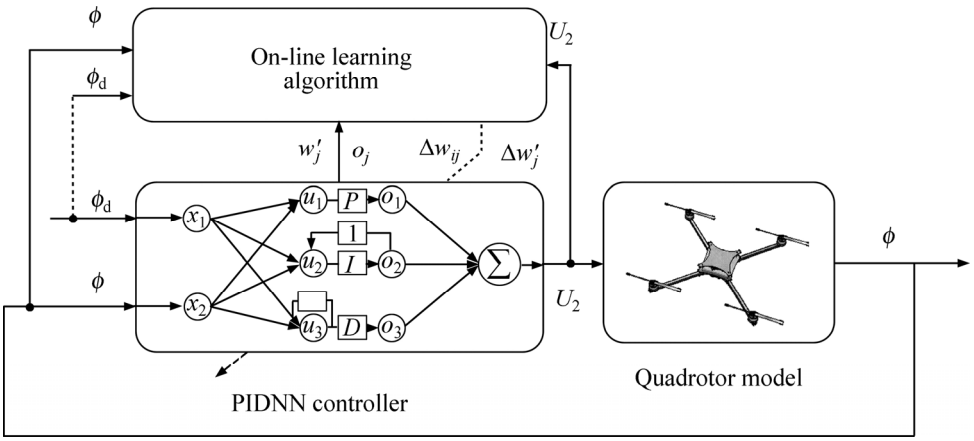

Fig. 5 Structure of roll angle controller

angle are similar to roll angle controller. If the appropriate initial connective weights are selected based on PID algorithm and the convergence is analyzed according to discrete Lyapunov theory, the tracking error of each attitude angle can vanish to zero; thereafter, the stability of whole system can be guaranteed.

5 Simulation results

The proposed control strategy has been tested by numerical simulation in order to check the control performance attained for set-point hovering flight and path following flight of a quadrotor helicopter subjected to external wind disturbance.

Due to the system uncertainties such as initial state error, attitude error and noise in the actual situation, a Gaussian white noise module is added to all feedback variables.

The turbulent wind field fragment is generated by simulation based on the model establish in section 3. Let simulation time be 60 s and the wind velocity be 20, feet altitude be 10 m/s, the flight altitude of quadrotor be 1 m. The parameters of the turbulent wind filed are summarized in Table 1 and the generated turbulent wind filed fragment is shown in Fig. 6.

Table 2 lists the dynamic parameter of the quadrotor helicopter.

5.1 Set-point hovering flight

The first simulation has been carried out to test the

Table 1 Turbulent wind field parameter

Parameter	Value	Parameter	Value
L_u/m	7.943	σ_u/(m·s^{-1})	1.995
L_v/m	3.971	σ_v/(m·s^{-1})	3.981
L_w/m	0.500	σ_w/(m·s^{-1})	1.000

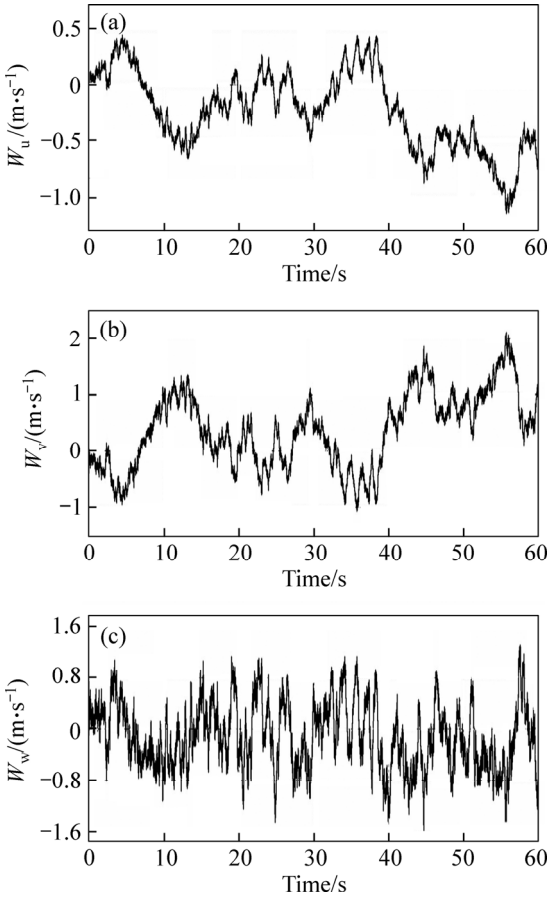

Fig. 6 Turbulent wind field fragment

Table 2 Dynamic parameter

Parameter	Definition	Unit	Value
m	Mass	kg	0.723
l	Arm length	m	0.310
J_r	Rotor inertia	kg·m^2	7.321×10^{-5}
I_{xx}	x inertia	kg·m^2	8.678×10^{-3}
I_{yy}	y inertia	kg·m^2	8.678×10^{-3}
I_{zz}	z inertia	kg·m^2	3.217×10^{-2}
b	Trust factor	N·s^2	5.324×10^{-5}
d	Drag factor	N·m·s^2	8.721×10^{-7}

control performance of set-point hovering flight, for the cases: 1) in the absence of wind disturbance and 2) in the presence of wind disturbance.

The hovering altitude h=1 m. The initial conditions are $\boldsymbol{X}_0$=[0, 0, 1] m and $\boldsymbol{\Theta}_0=\boldsymbol{0}_{3\times1}$ rad. The desired conditions are $\boldsymbol{X}_d$= [0, 0, 1] m and ψ_d=0 rad. Simulation results of set-point hovering flight are shown in Figs. 7−10.

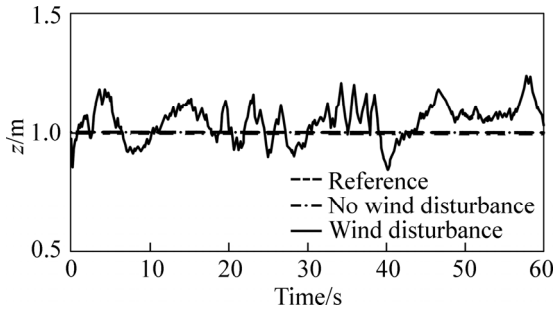

Fig. 7 Altitude output (set-point hovering flight)

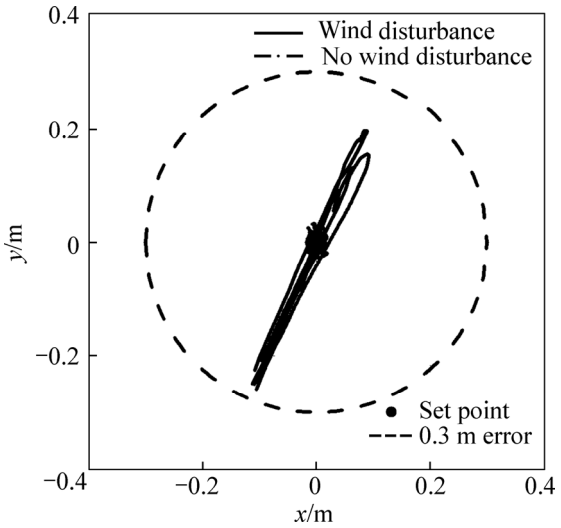

Fig. 8 Position output (set-point hovering flight)

As shown in Figs. 7 and 8, the quadrotor accurately achieved regulation in the absence of wind. In the presence of wind disturbance, the flight altitude is maintained in the range of 0.85−1.22 m, and the tracking error is within 0.2 m. Meanwhile, the position error is also below 0.3 m. The attitude angle tracks the reference well as depicted in Figs. 9 and 10.

In order to make a quantitative evaluation of the controller, the root-mean-square error (RMSE) indexes of all degrees of freedom are calculated. Table 3 summarizes the RSME indexes of set-point hovering flight. It can be derived from these data that all states are almost close to zero in the absence of wind disturbance. While in the presence of wind disturbance, the RSME indexes are increased; however, the values of them are still very small. From the results obtained, the proposed

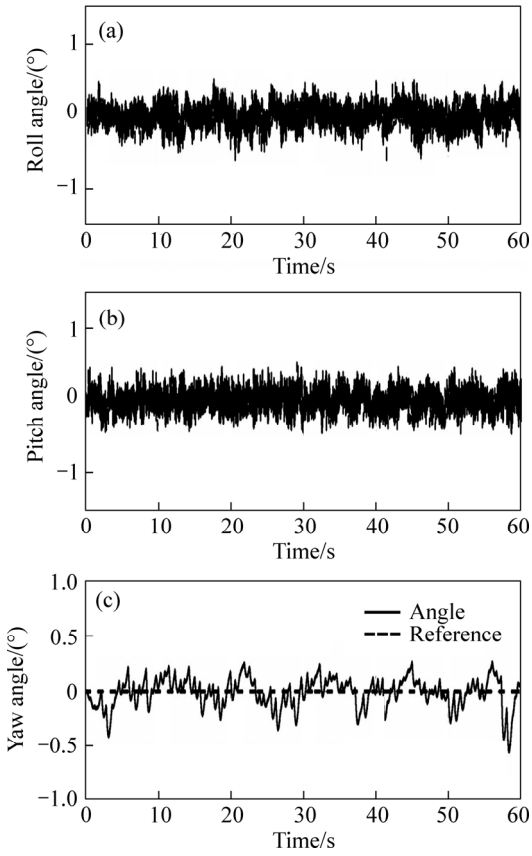

Fig. 9 Attitude angle output (set-point hovering flight, in the absence of wind disturbance)

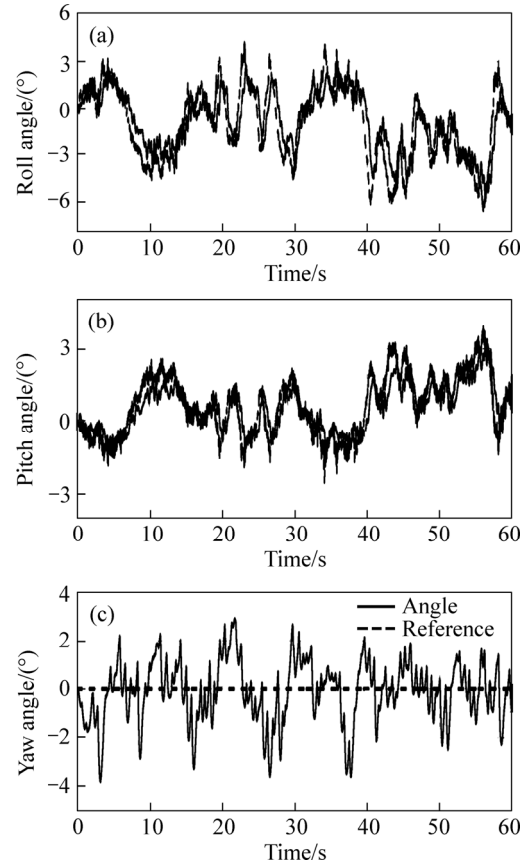

Fig. 10 Attitude angle output (set-point hovering flight, in the presence of wind disturbance)

Table 3 RMSE index performance analysis of set-point hovering flight

State	Index	Value	State	Index	Value
In the absence of wind disturbance	x/m	8.942×10^{-3}	In the presence of wind disturbance	x/m	9.072×10^{-2}
	y/m	8.534×10^{-3}		y/m	8.675×10^{-2}
	z/m	2.333×10^{-2}		z/m	0.124
	ϕ/(°)	3.394×10^{-2}		ϕ/(°)	0.373
	θ/(°)	3.357×10^{-2}		θ/(°)	0.392
	ψ/(°)	2.071×10^{-2}		ψ/(°)	0.143

control strategy effectively maintains the states in the neighborhood of the desired set-point and is robust to external wind disturbance.

5.2 Path following flight

The second simulation has been carried out for path following flight in order to test the maneuverability of the controller. The wind disturbance is similar to the set-point hovering flight. The reference path used is a spiral line evolving in the Cartesian space defined by $\boldsymbol{X}_d$=[sin(0.2094t), cos(0.2094t), 0.0333t] m. The initial conditions are $\boldsymbol{X}_0=\boldsymbol{0}_{3\times1}$ m and $\boldsymbol{\Theta}_0=\boldsymbol{0}_{3\times1}$ rad. The simulation results are shown in Figs. 11−14.

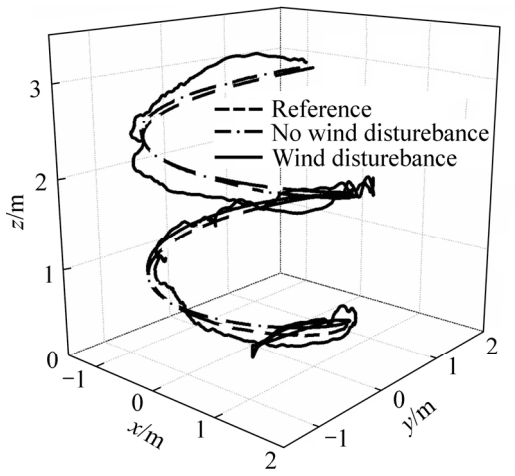

Fig. 11 Path following (3D)

The trajectory in which the quadrotor follows the reference path is depicted in Figs. 11 and 12. In the absence of wind disturbance, the quadrotor follows the reference path precisely. In the presence of wind disturbance, the quadrotor oscillates around the reference path, but the RMSE indexes of x, y and z are within 0.5 m as shown in Table 4. The attitude angles are able to track the reference in both cases as shown in Figs. 13 and 14. The RMSE indexes are below 0.4° (1.2°) in the absence (presence) of wind filed. The controller manages to accurately track the reference path, and provides fast and precise responses, and effectively attenuates the wind disturbance.

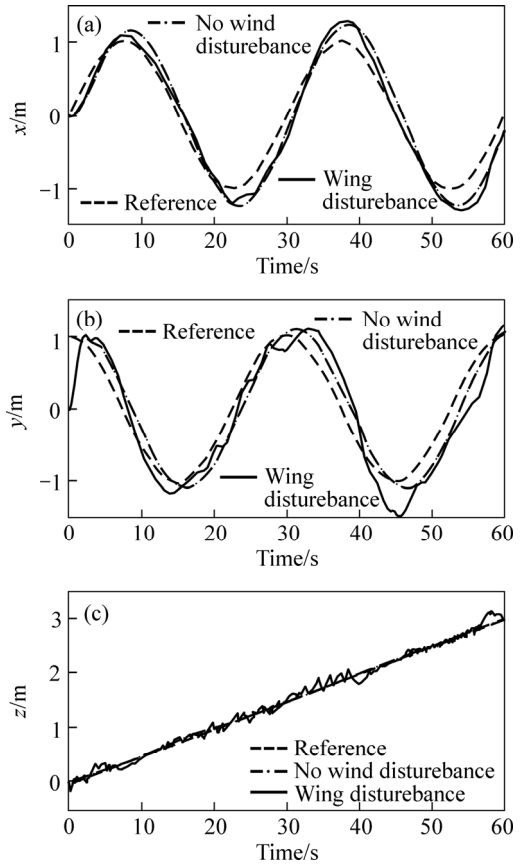

Fig. 12 Position output (path following)

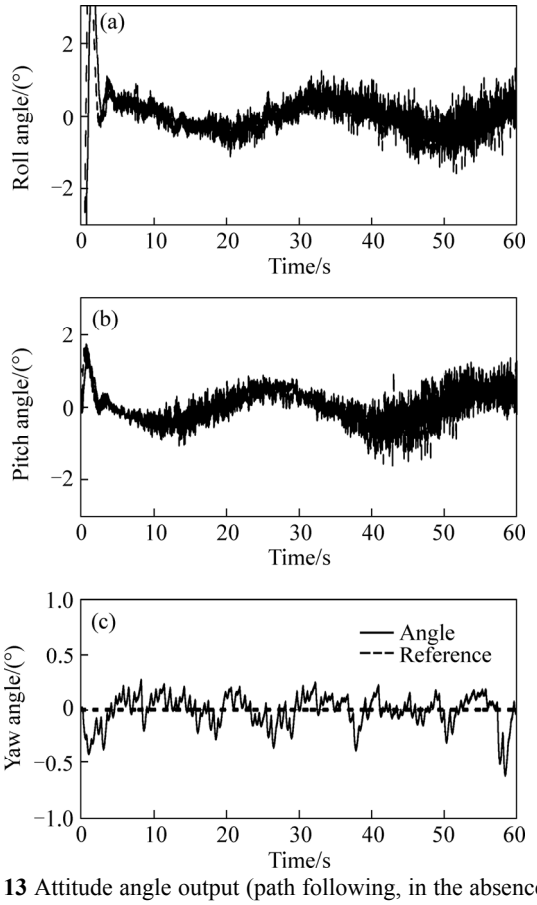

Fig. 13 Attitude angle output (path following, in the absence of wind disturbance)

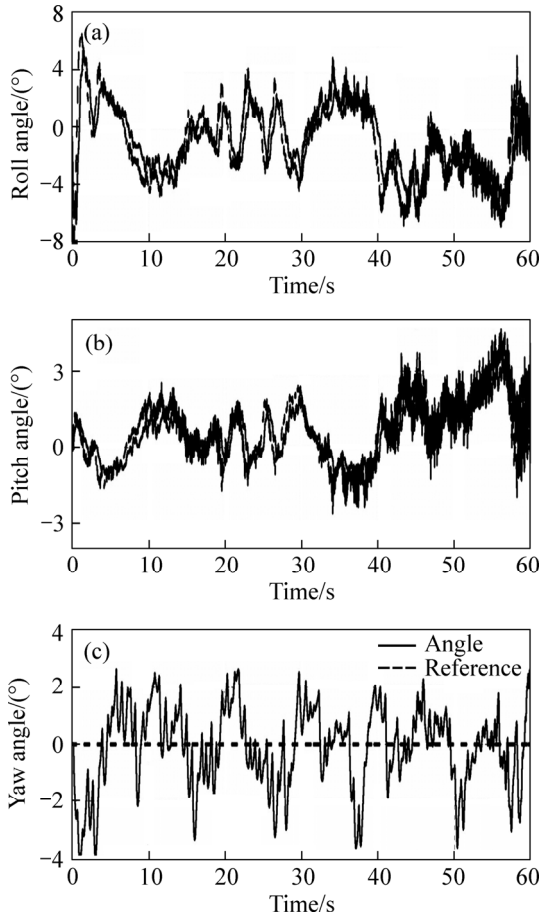

Fig. 14 Attitude angle output (Path following, in presence of wind disturbance)

Table 4 RMSE index performance analysis of path following flight

State	Index	Value	State	Index	Value
In the absence of wind disturbance	x/m	0.109	In the presence of wind disturbance	x/m	0.479
	y/m	8.074×10^{-2}		y/m	0.234
	z/m	0.164		z/m	0.409
	ϕ/(°)	0.369		ϕ/(°)	1.099
	θ/(°)	0.226		θ/(°)	1.178
	ψ/(°)	2.196×10^{-2}		ψ/(°)	0.236

6 Conclusions

A novel control strategy is proposed to solve the control problem of a quadrotor helicopter subjected to external wind disturbance. The controller consists of PID controller for the inner loop and decentralized PIDNN attitude controller for the outer loop. In order to accord with the practical situation, the turbulent wind field is generated based on Dryden model and is introduced as the disturbance source of the quadrotor helicopter. The performance of the proposed control algorithm is validated by simulation. The results demonstrate that the quadrotor achieves the desired conditions in both set-point hovering flight and path following flight, and the controller effectively attenuates the external wind disturbance. In summary, the controller presents good stability, maneuverability and robustness.

References

[1] ABHIJIT D, FRANK L, KAMESH S. Backstepping approach for controlling a quadrotor using Lagrange form dynamics [J]. Journal of Intelligent & Robotic Systems, 2009, 56(1/2): 127−151.

[2] MOREL Y, LEONESSA A. Direct adaptive tracking control of quadrotor aerial vehicles [C]// ASME International Mechanical Engineering Congress and Exposition. Chicago, 2006: 1−6.

[3] RASHID M I, AKHTAR S. Adaptive control of a quadrotor with unknown model parameters [C]// Proceedings of 9th International Bhurban Conference on Applied Sciences and Technology. Islamabad, 2012: 8−14.

[4] MOHAMMADI M, SHAHRI A M. Adaptive nonlinear stabilization control for a quadrotor UAV: Theory, simulation and experimentation [J]. Journal of Intelligent and Robotic Systems, 2013, 71(1): 1−18.

[5] ASHFAQ A M, WANG Dao-bo. Modeling and backstepping-based nonlinear control strategy for a 6 DOF quadrotor helicopter [J]. Chinese Journal of Aeronautics, 2008, 21(3): 261−268.

[6] RAFFO G V, ORTEGA M G, RUBIO F R. Backstepping & nonlinear ∞ control for path tracking of a quadrotor unmanned aerial vehicle [C]// American Control Conference. Seattle, 2008: 3356−3361.

[7] BOUCHOUCHA M, SEGHOUR S, H. OSMANI M B. Integral backstepping for attitude tracking of a quadrotor system [J]. Electronics and Electrical Engineering, 2011, 116(10): 75−80.

[8] RAFFO G, ORTEGA M, RUBIO F. An integral predictive nonlinear H(infinity) control structure for a quadrotor helicopter [J]. Automatic, 2010, 46(1): 29−39.

[9] NICOL C, MACNAB C J B, RAMIREZ-SERRANO A. Robust adaptive control of a quadrotor helicopter [J]. Mechatronics, 2011, 21(5): 927−938.

[10] DAEWON L, JIN K H, SHANKAR S. Feedback linearization vs. adaptive sliding mode control for a quadrotor helicopter [J]. International Journal of Control Automation And Systems, 2009, 7(3): 419−428.

[11] PATEL A R, PATEL M A, VYAS D R. Modeling and analysis of quadrotor using sliding mode control [C]// Proceedings of the Annual Southeastern Symposium on System Theory. FL, 2012: 111−114.

[12] BESNARDA L, SHTESSELB Y B, LANDRUM B. Quadrotor vehicle control via sliding mode controller driven by sliding mode disturbance observer [J]. Journal of the Franklin Institute, 2012, 349(2): 658−684.

[13] DERAFA L, BENALLEGUE A, FRIDMAN L. Super twisting control algorithm for the attitude tracking of a four rotors UAV [J]. Journal of the Franklin Institute, 2012, 349(2): 685−699.

[14] YANG Biao, PENG Jin-hui, GUO Sheng-hui, ZHANG Shi-min, LI Wei, HE Tao. Acid-pickling plates and strips speed control system by microwave heating based on self-adaptive fuzzy PID algorithm [J]. Journal of Central South University, 2012, 19(8): 2179−2186.

[15] BOUABDALLAH S, NOTH A, SIEGWART R. PID vs LQ control techniques applied to an indoor micro quadrotor [C]// Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems. Sendai, 2004: 2451−2456.

[16] SU Jing-ya, FAN Peng-hui, CAI Kai-yuan. Attitude control of quadrotor aircraft via nonlinear PID [J]. Journal of Beijing University of Aeronautics and Astronautics, 2011, 37(9): 1054−1058. (in Chinese)

[17] DIERKS T, JAGANNATHAN S. Neural network output feedback control of a quadrotor UAV [C]// 47th IEEE Conference on Decision and Control. Cancun, 2008: 3633−3639.

[18] NICOL C, MACNAB C J B, RAMIREZ-SERRANO A. Robust neural network control of a quadrotor helicopter [C]// IEEE Canadian Conference on Electrical and Computer Engineering. Canada, 2008: 1233−1237.

[19] CHEN J, HUANG T. Applying neural networks to on-line updated PID controllers for nonlinear process control [J]. Journal of Process Control, 2004, 14(2): 211−230.

[20] THANH T D C, AH K K. Nonlinear PID control to improve the control performance of 2 axes pneumatic artificial muscle manipulator using neural network [J]. Mechatronics, 2006, 16: 577−587.

[21] CONG S, LIANG Y. PID-like neural network nonlinear adaptive control for uncertain multivariable motion control systems [J]. Industrial Electronics, 2009, 56(10): 3872−3879.

[22] SHU H, PI Y. PID neural networks for time-delay systems [J]. Computers & Chemical Engineering, 2000, 24(2): 859−862.

[23] MARABA V A, KUZUCUOGLU A E. PID neural network based speed control of asynchronous motor using programmable logic controller [J]. Advances in Electrical and Computer Engineering, 2011, 11(4): 23−28.

[24] XIAO Ye-lun, JIN Chang-jiang. Flight theory in atmospheric disturbance [M]. Beijing: National Defense Industry Press, 1993: 73−74. (in Chinese)

[25] BEAL T R. Digital simulation of atmospheric turbulence [J]. Journal of Guidance, Control and Dynamics, 1993, 16(1): 132−138.

(Edited by YANG hua)