Proceedings of the 5th International Conference on Computing and Data Science
DOI: 10.54254/2755-2721/20/20231101

# A flight control system for quadrotor plant protection UAV based on RBF-PID algorithm

**Chenyu Liu[1, 3, †], Jiawang Xu[2, †]**

[1]School of Physical Sciences, University of Science and Technology of China, Hefei, China
[2]School of Automation, Chengdu University of Information Technology, Chengdu, China

[3]liuchenyu@mail.ustc.edu.cn
[†] All the authors contributed equally and their names were listed in alphabetical order

**Abstract.** As a powerful tool for modern agricultural protection, unmanned aerial vehicle (UAV) has the advantages of good adaptability and high efficiency in spraying and other operations compared with traditional manual operations The attitude and control of plant protection UAV are greatly affected by external factors during operation. However, the traditional proportional-integral-derivative (PID) controller commonly used at present cannot adapt to the complex environment because of the fixed parametersTherefore, a more sensitive method is needed to achieve adaptive balance of the aircraft attitude. In this paper, we establish the dynamic mathematical model of quadrotor UAV, and the influence of air resistance and rotational torque is considered in the process. Adopting the combined control method of Radial Basis Function (RBF) neural network and PID, the dynamic tuning of system control parameters is achieved through the self-learning of neural network and the feature of nonlinear mapping. The control system not only has the advantages of simple PID control structure and clear physical meaning, but also has the ability of neural network self-learning and adaptation. Matlab/Simulink was used as the experimental platform to respectively simulate the RBF-PID control system and the traditional PID control system. The simulation results show that the control system based on RBF-PID algorithm has a shorter response time and stronger robustness, which enhances the system's self-adaptability.

**Keywords:** quadrotor UAVs, RBF neural network, PID control.

## 1. Introduction

According to incomplete statistics on agricultural production, the economic losses caused by diseases and pests in the entire agricultural production can reach 20% to 30% of its profits. In addition, due to the large area of farmland, scattered plots, and diverse crop varieties, different models of plant protection equipment are needed to provide assistance for their operations under different production processes [1]. In this context, the application of plant protection unmanned aerial vehicle (UAV) in agricultural field is increasingly common. Compared with traditional manual or mechanical, agricultural unmanned aerial vehicle has the advantages of low cost, high efficiency, adaptability to different crop varieties and less

DOI: 10.54254/2755-2721/20/20231101

damage to crops [2]. Therefore, agricultural unmanned aerial vehicle is playing an increasingly important role in crop sowing, growth monitoring, and pesticide spraying [3].

Currently, plant protection drones are mainly operated remotely by professional personnel. Due to significant differences in the level of operators, inconsistent crop heights, and complex terrain, there are problems when operating plant protection drones, such as poor accuracy in height determination, waste of medication, and instability. Therefore, achieving stable and reliable autonomous terrain following flight of unmanned aerial vehicle is of great significance [4].

When the plant protection unmanned aerial vehicle operating, the external interference and the drone's own parameter changes and other factors will affect the drone's attitude control, so the adjustment of attitude position is the key link of drone flight. At present, the common control algorithms of aircraft mainly include: single loop proportional-integral-derivative (PID) control [5], cascade PID control [6], linear quadratic regulator (LQR) control [7], backstepping method [8], sliding mode control (SMC) [9], fuzzy logic control (FLC) [10], neural network control [11-13] and other control methods [14]. The traditional PID controller commonly used now has poor dynamic performance with fixed parameters and weak anti-interference ability, which cannot be well adapted to the impact of system parameter changes [15]. The RBF-PID control algorithm based on radial basis function and PID control integrates neural network, adaptive learning, PID control technology, etc. It has advantages such as high sensitivity, fast dynamic response, small overshoot, and strong robustness.

The rest of this paper is arranged as follows: The kinematics and dynamics mathematical model of the quadrotor drone is studied and established in Section 2. Section 3 designs a flight control system algorithm based on RBF-PID control technology and conducts simulation verification in Matlab software. Finally, a summary and discussion were conducted on the simulation results and algorithm effectiveness, and prospects were proposed for the research shortcomings.

## 2. Mathematical model of quadrotor UAV

The main structure of the four-rotor unmanned aerial vehicle is mainly based on the flight control board as the center, to the four symmetrical directions of the flight control board to build four arms of the same length, equal mass and in the shape of "X" or cross arms. At the end of each arm, a brushless DC motor is installed and connected to the blades of the unmanned aerial vehicle. Its flight power lies in the fact that the flight control board receives instructions to change the expected attitude of the aircraft, and then controls the battery to supply power to the motor, which drives the blades to rotate at high speed to generate upward lift.

### *2.1. Establishment and transformation of coordinate systems*

In order to accurately describe the spatial attitude information of the quadrotor drone, establish a good mathematical model, and facilitate the design of the flight control system, the reference coordinate system is first defined: body coordinate system ($O_bX_bY_bZ_b$) and fixed geographic coordinate system ($O_eX_eY_eZ_e$). There are two main movements of the quadrotor unmanned aerial vehicle: translation and rotation. In the process of translational motion, the unmanned aerial vehicle is analysed as a particle model. When rotating, it is studied as a rigid body model.

According to Euler's theorem, the rotation of a rigid body around a fixed point in three-dimensional coordinate space can be regarded as the combination of three finite rotations around that point. The rotation order XYZ is defined, that is, the attitude vector of the quadrotor drone is rotated to the geographical coordinate system by first rotating around the X-axis, then the Y-axis and finally the Z-axis. The rotation matrix $R_b^e$ can be calculated from Equation 2.1:

$$R_b^e = R_z(\Psi)\, R_y(\theta)\, R_x(\phi) \tag{1}$$

In the equation, the roll angle is $\phi$, the pitch angle is $\theta$, and the yaw angle$\Psi$ is. The rotation matrix $R_b^e$ can be obtained as

DOI: 10.54254/2755-2721/20/20231101

$$R_{\mathrm{b}}^{\mathrm{e}}=\begin{bmatrix}\cos\Psi\cos\theta & -\sin\Psi\cos\phi+\cos\Psi\sin\theta\sin\phi & \sin\Psi\sin\phi+\cos\Psi\sin\theta\cos\phi\\ \sin\Psi\cos\theta & \cos\Psi\cos\phi+\sin\Psi\sin\theta\sin\phi & -\cos\Psi\sin\phi+\sin\Psi\sin\theta\cos\phi\\ -\sin\theta & \cos\theta\sin\phi & \cos\theta\cos\phi\end{bmatrix} \quad (2)$$

*2.2. Establishment of mathematical model of quadrotor UAV*

Figure 1 shows the structure of the simplified quadrotor unmanned aerial vehicle. Assuming that the aircraft is a rigid body, and the centre of gravity is consistent with the centre, the mass and moment of inertia are unchanged, the propellers 2 and 4 rotate clockwise, and the propellers 1 and 3 rotate counter clockwise. It is defined that the length $l$ of the drone arm and the rotational speed $\Omega_1$, $\Omega_2$, $\Omega_3$, $\Omega_4$ of the four propellers. The angular velocity of the drone along the three axes in the body coordinate axis is $\omega=(\omega_x,\omega_y,\omega_z)^{\mathrm{T}}$. Euler angular velocity in geographic coordinate system is $\dot{\alpha}=(\dot{\phi},\dot{\theta},\dot{\psi})^{\mathrm{T}}$ and the speed is $v=(v_x,v_y,v_z)^{\mathrm{T}}$.

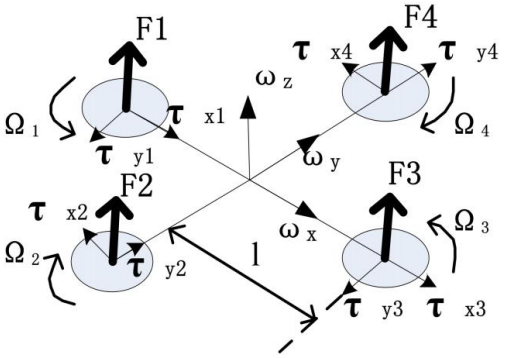


**Figure 1.** Simplified kinematics model of quadrotor UAV.

The angular velocity of the body can be represented by Euler's angular velocity through three-axis rotation:

$$\omega=\mathrm{R_x}(\phi)\ \mathrm{R_y}(\theta)\begin{bmatrix}0\\0\\\dot{\psi}\end{bmatrix}+\mathrm{R_x}(\phi)\begin{bmatrix}0\\\dot{\theta}\\0\end{bmatrix}+\begin{bmatrix}\dot{\phi}\\0\\0\end{bmatrix} \quad (3)$$

Therefore, we can get:

$$\begin{cases}\omega_{\mathrm{x}}=\dot{\phi}-\dot{\Psi}\ \sin\theta\\ \omega_{\mathrm{y}}=\dot{\theta}\cos\phi+\dot{\Psi}\cos\theta\sin\phi\\ \omega_{\mathrm{z}}=-\dot{\theta}\sin\phi+\dot{\Psi}\cos\theta\cos\phi\end{cases} \quad (4)$$

Considering the air resistance during translational motion and the influence of rolling, pitching, and yaw torque on the model, the motion equation of the drone model can be written as follows according to the Newton's law of motion and Euler equation.

$$\begin{cases}\mathrm{m}\dot{\mathrm{v}}=\mathrm{F_f}+\mathrm{F_d}+\mathrm{F_g}\\ \mathrm{J}\dot{\omega}+\omega\times\mathrm{J}\omega=\mathrm{M}+\tau\end{cases} \quad (5)$$

In the equation, $\mathrm{F_f}$ is the translational air resistance; $\mathrm{F_d}$ is translational force; $\mathrm{F_g}$ is gravity; k is translational air resistance coefficient; ω is the angular velocity of the body; J is the moment of inertia of the body; M is the aerodynamic torque generated when the rotor rotates; τ is the gyroscopic effect torque during rotation.

Due to the assumption that the drone is a symmetric rigid body, there is no inertia product. Then the inertia matrix is given:

DOI: 10.54254/2755-2721/20/20231101

$$J = I = \begin{bmatrix} I_x & 0 & 0 \\ 0 & I_y & 0 \\ 0 & 0 & I_z \end{bmatrix} \tag{6}$$

According to the working principle of quadrotor UAV, the control input is defined as U, that is

$$U = \begin{cases} U_1 = b(\Omega_1^2 + \Omega_2^2 + \Omega_3^2 + \Omega_4^2) \\ U_2 = b(\Omega_4^2 - \Omega_2^2) \\ U_3 = b(\Omega_3^2 - \Omega_1^2) \\ U_4 = d(\Omega_2^2 + \Omega_4^2 - \Omega_1^2 - \Omega_3^2) \end{cases} \tag{7}$$

Where, b is the lift coefficient and d are the torque coefficient. $U_1$ controls the climbing motion of unmanned aerial vehicle, controls the climbing motion, $U_2$ controls the rolling motion, $U_3$ controls the pitching motion, and $U_4$ controls the yawing motion [16].

The aerodynamic moment $M_x$ of quadrotor unmanned aerial vehicle in roll motion (moving around the x-axis) can be expressed as

$$M_x = l(F_4 - F_2) = b(\Omega_4^2 - \Omega_2^2) = U_2 \tag{8}$$

At this point, propeller 1 is analysed and its spiral effect torque is $\tau_{x1} = \omega_y \times J_r \Omega_1$. The same can be said for other propellers, then the gyroscopic effect torque of the four propellers in rolling motion is:

$$\tau_x = J_r\, \omega_y(\Omega_1 + \Omega_3 - \Omega_2 - \Omega_4) \tag{9}$$

The torque applied to pitch motion and yaw motion can be deduced similarly, with the exception of yaw aerodynamic moment $M_z$ is:

$$M_z = d(\Omega_2^2 + \Omega_4^2 - \Omega_1^2 - \Omega_3^2) = U_4 \tag{10}$$

Combined with the above formula, the mathematical model of dynamics and kinematics of quadrotor drone can be obtained:

$$\begin{cases} \ddot{x} = \frac{U_1(\cos\Psi \sin\theta \cos\phi + \sin\Psi \sin\phi) - k\dot{x}}{m} \\ \ddot{y} = \frac{U_1(\sin\Psi \sin\theta \cos\phi + \cos\Psi \sin\phi) - k\dot{y}}{m} \\ \ddot{z} = \frac{U_1(\cos\theta \cos\phi) - k\dot{z}}{m} - g \\ \dot{\omega_x} = \frac{(I_y - I_z)\omega_y\omega_z + U_2 l + J_r\, \omega_y(\Omega_1 + \Omega_3 - \Omega_2 - \Omega_4)}{I_x} \\ \dot{\omega_y} = \frac{(I_z - I_x)\omega_x\omega_x + U_3 l + J_r\, \omega_x(\Omega_2 + \Omega_4 - \Omega_1 - \Omega_3)}{I_y} \\ \dot{\omega_z} = \frac{(I_x - I_y)\omega_x\omega_y + U_4 l}{I_z} \end{cases} \tag{11}$$

## 3. Combining a PID controller with an RBF neural network

At present, most of the drone control algorithms are classical PID control, which has simple principles and mature technology. However, unmanned aerial vehicle is a typical multi - input - output, strong nonlinear, strong coupling system. Classical PID has certain limitations in some cases. For example, when realizing the flipping of the drone's body, it is a little bit hard for the classic PID controller to make it. Therefore, this chapter explores the application of RBF-PID control algorithm in drone attitude control from the perspective of improving the control effect and improving the adaptability of PID control parameters.

DOI: 10.54254/2755-2721/20/20231101

### *3.1. Proportional-integral-derivative (PID) controller structure*

PID control is currently the most extensively used control algorithm. It performs corresponding proportional, integral, and differential operations based on the error between the currently accurate output of the controlled system and the target output to achieve the goal of improving the control effect of the controlled system. The core idea of PID control is to perform proportional, integral, and differential operations on the error between the currently accurate output and the expected output at each sampling time of the system, and then adjust the next control output of the controlled system based on the results of the operations. Thus, the actual output can gradually approach the target output.

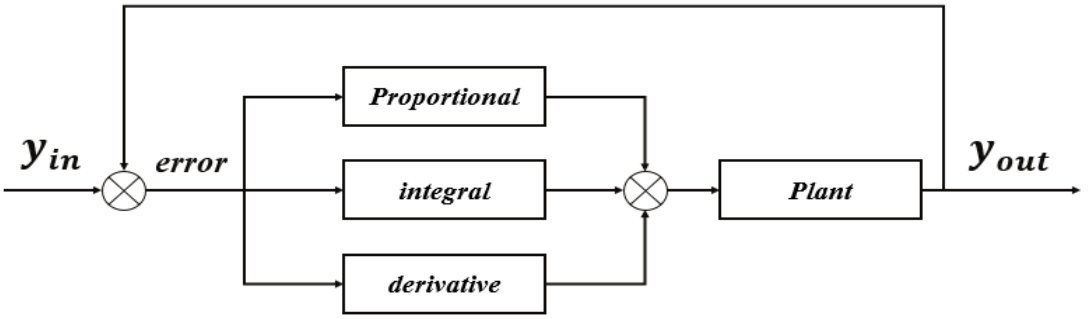


**Figure 2.** PID controller structure.

The structure diagram of the PID controller formed based on the PID control algorithm is shown in Figure 2. The PID controller has a simple structure and belongs to a closed-loop control method that continuously adjusts the input based on the real-time output feedback of the controlled object. When there is an error between the actual output of the system and the target output, the PID controller will correct the error of the controlled object through a combination of proportional, integral, and differential control algorithms according to norms or standards, maintaining the output in a stable state [17].

Discrete system PID control is divided into two types: positional and incremental, and incremental control is used for motors. Its control algorithm is

$$\Delta u(k) = K_p[e(k) - e(k-1)] + K_i\, e(k) + K_d[e(k) - 2e(k-1) + e(k-2)] \quad (12)$$

$$K_i = K_p \frac{T}{T_i} \quad (13)$$

$$K_d = K_p \frac{T_d}{T} \quad (14)$$

In the equation, $K_p$ is the proportional coefficient, the integral coefficient is $K_i$, and $K_d$ is the differential coefficient. $T$ is the continuous signal sampling period and $e(k)$ is the control deviation.

### *3.2. Radial basis function (RBF) neural network*

Radial basis function neural networks are generally divided into three layers. The input layer is the first layer, composed of input signals; The hidden layer is the second, whose activation function is a radial basis function, often using Gaussian basis function, which is a locally distributed nonnegative nonlinear function with central radially symmetric decay; The third layer is the output layer.

DOI: 10.54254/2755-2721/20/20231101

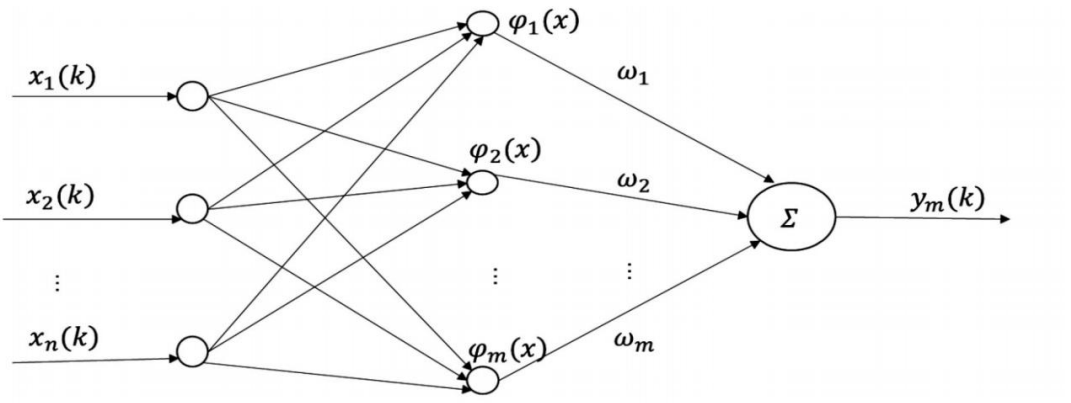


**Figure 3.** RBF neural network structure.

According to the structure analysis of 3.2. Radial basis function neural network (as shown in Figure 3), the basic idea of radial basis function neural network is: in the hidden layer, the basis function is used as the activation function of neurons, and the input is mapped from the input layer to the hidden layer. The output layer is the weighted sum of the output of the hidden layer. In general, the input and output of the network are mapped nonlinearly, while the network is linear for tunable parameters. Such a network has a faster learning rate than a basic network and does not create local minimum problems. [18].

### *3.3. RBF-PID control algorithm principle*

To achieve good control results in PID control, it is necessary to find appropriate proportional, integral, and differential coefficients according to the characteristics of the controlled object, so that the coefficients cooperate with each other. But the process of finding the optimal parameters often takes a lot of time. The fit between coefficients is not a simple linear relationship. The radial basis function neural network has the ability to represent any nonlinear relationship and does not have local optimization problems. Therefore, machine learning can be used to adjust the parameters of the PID controller to seek the best coefficient combination and achieve the best control effect [18].

The controller mainly consists of two parts (as shown in Figure 4). The incremental PID controller performs closed-loop control on the controlled object, and its proportional, integral, and differential coefficients are adjusted through a neural network. The radial basis function neural network identifies the Jacobian information of the system and outputs the increment of proportional, integral, and differential coefficients. In this way, self-tuning of coefficients is achieved.

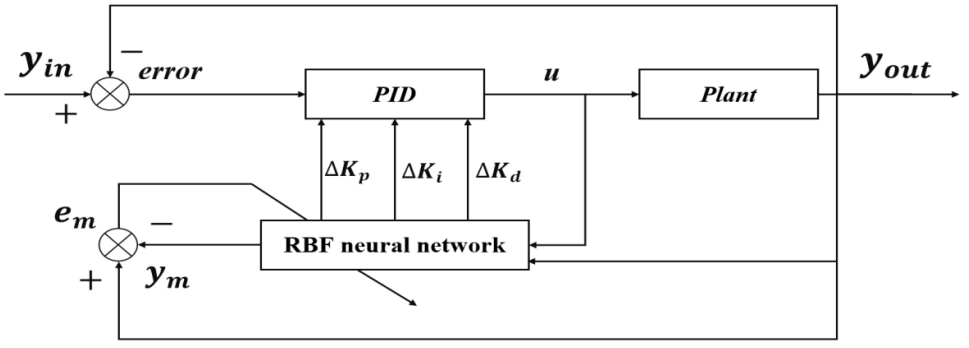


**Figure 4.** RBF-PID controller structure.

DOI: 10.54254/2755-2721/20/20231101

### *3.4. Parameters adjustment of RBF neural networks*

In radial basis function neural networks, the input variable $x = [x_1, x_2, \dots x_n]^{\mathrm{T}}$; The hidden layer activation function is Gauss function $h_j$.

$$h_j = \exp\left(-\frac{\|x-c_j\|^2}{2b_j^2}\right) \ (j = 1,2, \dots m) \tag{15}$$

The knot vector of Gauss function $b_j = [b_1, b_2, \dots b_m]^{\mathrm{T}}$. The center vector of the $j$th node in the network $c_j = \left[c_{j1}, c_{j2}, \dots c_{ji}, \dots c_{jn}\right]^{\mathrm{T}} (i = 1, 2, \dots n)$. The weight vector of the network is $W = [w_1, w_2, \dots w_m]^{\mathrm{T}}$.

The output of the radial basis function neural network is:

$$y_m(k) = \sum_{i=1}^{m} w_i h_{\mathrm{i}} (\mathrm{i} = 1,2, \dots \mathrm{m}) \tag{16}$$

In the calculation process of radial basis function neural network, Jacobian information identification formula of the controlled object is very important, its performance index function is:

$$J = \frac{1}{2}[y_{out}(k) - y_M(k)]^2 \tag{17}$$

During the network training process, gradient descent algorithm is used to update the parameters of node center, node width, and weight. The specific iteration calculation is as follows:

$$\Delta \mathrm{w_j(k)} = \eta[\mathrm{y_{out}(k) - y_m(k)}]\mathrm{h_j} \tag{18}$$

$$\mathrm{w_j(k)} = \mathrm{w_j(k-1)h_j} + \Delta \mathrm{w_j(k)} + \beta\left[\mathrm{w_j(k-1) - \Delta w_j(k-2)}\right] \tag{19}$$

$$\Delta \mathrm{b_j(k)} = \eta[\mathrm{y_{out}(k) - y_m(k)}]\mathrm{h_j w_j} \frac{\|\mathrm{x-c_j}\|}{\mathrm{b_j^3}} \tag{20}$$

$$\mathrm{b_j(k)} = \mathrm{b_j(k-1)} + \Delta \mathrm{b_j(k)} + \beta\left[\mathrm{b_j(k-1) - b_j(k-2)}\right] \tag{21}$$

$$\Delta \mathrm{c_{ji}(k)} = \eta[\mathrm{y_{out}(k) - y_m(k)}]\mathrm{w_j} \frac{\mathrm{x_j - c_{ji}}}{\mathrm{b_j^2}} \tag{22}$$

$$\mathrm{c_{ji}(k)} = \mathrm{c_{ji}(k-1)} + \Delta \mathrm{c_{ji}(k)} + \beta\left[\mathrm{c_{ji}(k-1) - c_{ji}(k-2)}\right] \tag{23}$$

In the equation, $\beta$ is the momentum factor ($\beta > 0$), and $\eta$ is the learning rate ($\eta > 0$).

The sensitivity of the controlled object's output to the control input is the system's Jacobian information. The specific calculation method is:

$$\frac{\partial y_{out}(k)}{\partial \Delta u(k)} \approx \frac{\partial y_m(k)}{\partial \Delta u(k)} = \sum_{j=1}^{m} w_j h_j \frac{c_{ji} - x_1}{b_j^2} \tag{24}$$

In actual systems, $x_1 = \Delta u(k)$.

### *3.5. Parameters adjustment of PID controller*

Adopting an incremental PID controller, the control error is:

$$error(k) = y_{in}(k) - y_{out}(k) \tag{25}$$

The tuning index of radial basis function neural network are:

$$\mathrm{E(k)} = \frac{1}{2}\mathrm{error(k)}^2 \tag{26}$$

Gradient descent method is adopted in the adjustment of $K_p$, $K_i$, $K_d$:

$$xc(1) = error(k) - error(k-1) \tag{27}$$

DOI: 10.54254/2755-2721/20/20231101

$$xc(2) = error(k) \tag{28}$$

$$xc(3) = error(k) - 2error(k-1) + error(k-2) \tag{29}$$

$$\Delta K_p = -\eta \frac{\partial E}{\partial k_p} = -\eta \frac{\partial E}{\partial y}\frac{\partial y}{\partial u}\frac{\partial u}{\partial k_p} = \eta error(k)\frac{\partial y}{\partial u}xc(1) \tag{30}$$

$$\Delta K_i = -\eta \frac{\partial E}{\partial k_i} = -\eta \frac{\partial E}{\partial y}\frac{\partial y}{\partial u}\frac{\partial u}{\partial k_i} = \eta error(k)\frac{\partial y}{\partial u}xc(2) \tag{31}$$

$$\Delta K_d = -\eta \frac{\partial E}{\partial k_d} = -\eta \frac{\partial E}{\partial y}\frac{\partial y}{\partial u}\frac{\partial u}{\partial k_d} = \eta error(k)\frac{\partial y}{\partial u}xc(3) \tag{32}$$

In the equation, $\frac{\partial y}{\partial u}$ represents the Jacobian information of the controlled object, which can be obtained through identification of the neural network.

## 4. Simulation of RBF-PID control algorithm

Now the RBF-PID controller is designed for a nonlinear object. First, its discretization mathematical model is assumed to be:

$$\mathrm{y(k)} = \frac{-0.3\mathrm{y(k-1)}+\mathrm{u(k-1)}}{6+\mathrm{y(k-1)}^2} \tag{33}$$

In order to comprehensively analyse the dynamic adjustment ability of incremental PID and RBF-PID control algorithms, the objective functions were selected as sine wave and square wave, and sine wave and square wave tracking simulations were conducted separately. The simulation time is 10 seconds, and the sampling interval is 0.001 seconds. The input sine wave signal and square wave signal have amplitudes of 1 and a period of 4 seconds.

Simulation results of RBF-PID control algorithm were shown in the Figure 5. The sine wave tracking results indicate that the incremental PID has overshoot and static error. During the tracking process, the response is slow, and the average error is large. The RBF-PID controller has significant errors in the initial stage. However, after a short period of time (about 0.7 second), the parameters are automatically adjusted, and the error significantly decreases. The overshoot of the system is less than 15% and the entire tracking process is also smoother. Therefore, it can be said that the control effect of RBF-PID algorithm has significantly improved compared to incremental PID.

DOI: 10.54254/2755-2721/20/20231101

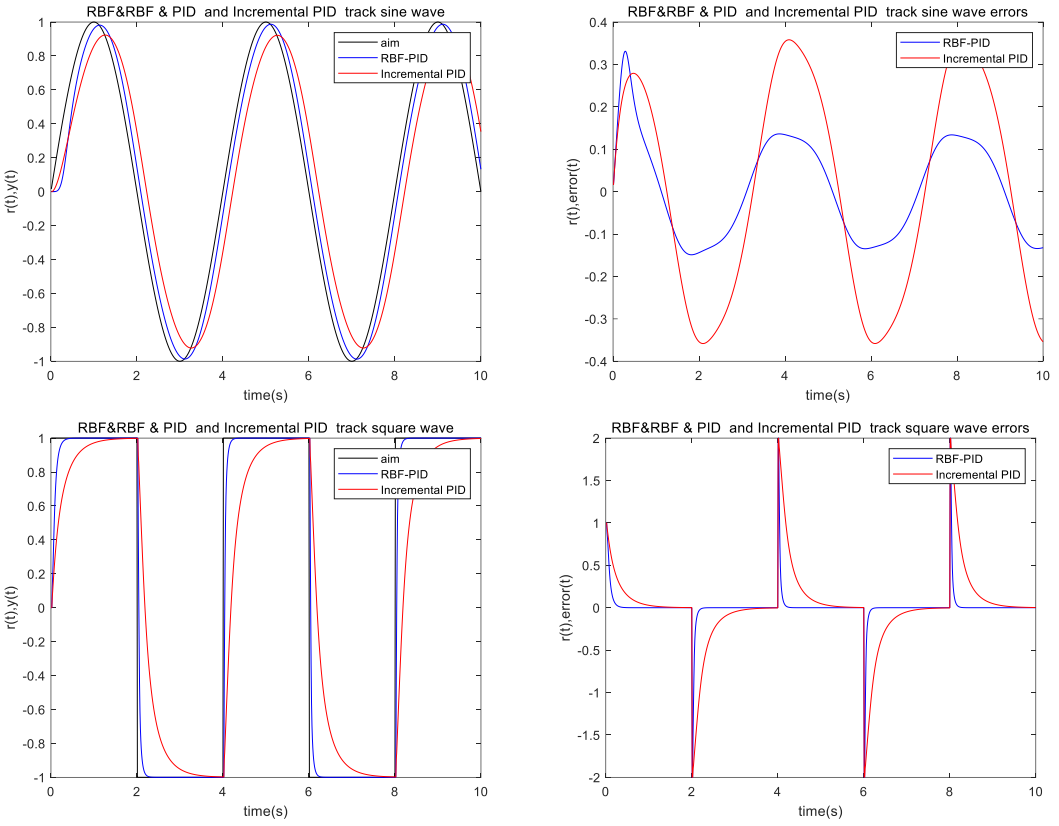


**Figure 5.** Simulation results.

The square wave tracking results indicate that the response speed of incremental PID is significantly slower than that of RBF-PID. The RBF-PID controller has faster response speed. Moreover, it can be clearly seen that with the growth of time and the continuous iteration of parameters, the response speed of RBF-PID is still improving.

In summary, RBF-PID controller has smaller overshoot and shorter adjustment time than PID controller in attitude control and can achieve the target state more quickly. When the system input changes, the neural network PID control can self-learning to find the appropriate PID parameters and can recover to the specified trajectory more rapidly after encountering disturbances in the flight of quadrotor unmanned aerial vehicle.

## 5. Discussion

Since the plant protection unmanned aerial vehicle is loaded with medicine box, its own parameters are constantly changing in the process of operation, and it is more susceptible to external interference. The proposed RBF-PID controller not only overcomes the shortcoming of traditional PID controller with fixed parameters, but also ensures accurate direction of parameter correction and stable process. The momentum factor is introduced into the gradient descent method to ensure that the correction results are carried out along the direction of parameter convergence. At the same time, when the weight is corrected too fast, the updating of the weight has a certain inertia, which can overcome the nonlinear system instability caused by the excessive learning rate. Of course, since the object of the research system is a simple low-order system in the simulation, and the influence of external disturbance is not taken into account, so the superiority of RBF-PID controller cannot be fully reflected.

Compare the relevant studies of other researchers in this field. Yu W et al. [19] proposed a fuzzy adaptive stable PID controller. Although their simulation verification is more complete, the response speed is still insufficient. The RBF-PID control proposed in this paper not only has a fast response speed

DOI: 10.54254/2755-2721/20/20231101

and can be gradually improved due to its characteristics of self-learning, but also requires a larger amount of calculation.

## 6. Conclusion

This study utilized the self-learning ability of radial basis function (RBF) neural networks to optimize and adjust the key parameters of incremental PID controllers, in order to achieve online adaptive adjustment of PID parameters. The problems of poor real-time performance and low control accuracy in conventional PID control methods are solved. A comparative analysis of sine wave and square wave tracking was conducted on the incremental PID and RBF-PID controllers on the Matlab simulation platform. The software simulation results show that the RBF-PID controller has fast response speed, small error, and follows closely when the target value changes. The controller can rapidly reach steady state, has good fast response ability and stability, and can improve the anti-disturbance ability and adaptive ability of quadrotor. The control effect of RBF-PID algorithm is significantly better than that of traditional incremental PID. The efficient controller scheme for Quadrotor Plant Protection Unmanned Aerial Vehicle is successfully achieved.

In the research process, there are still several aspects that can be further studied and improved. In the process of dynamic modeling of quadrotor drone, this paper ignored the influence of the change of characteristic parameters caused by the change of plant protection drone's own parameters and obtained an approximate linear model. Therefore, the modeling is not rigorous enough, and further study is needed to establish an accurate model. In addition, only simulation verification has been conducted on the RBF-PID algorithm, proving its superiority. In the future, we will further apply it to the drone model for further application research.

## References

[1] Liu J, Zu J, Tao D and Li C. 2022. Optimization and segmentation method for plant protection drone operation route based on minimum spraying unit. Computer Systems & Applications (03), 340-344.

[2] Chen H, Lan Y, Fritz BK, Hoffmann WC and Liu S. 2021. Review of agricultural spraying technologies for plant protection using unmanned aerial vehicle (UAV). International Journal of Agricultural and Biological Engineering, 14(1) , 38-49.

[3] Li J, Hu X, Lan Y and Deng X. 2021, Research advance on worldwide agricultural UAVs in 2001-2020 based on bibliometrics[J]. Transactions of the Chinese Society of Agricultural Engineering (Transactions of the CSAE), 37(9), 328-339.

[4] Zang Y, Wu H, Zhou Z, Zang Y, Zhao L, Zhou J and Lv Z. 2020. Terrain Following Technology of Unmanned Aerial Vehicle for Plant Protection: a review. Journal of Shenyang Agricultural University, (02), 250-256.

[5] Shehzad MF, Bilal A and Ahmad H. 2019. Position & Attitude Control of an Aerial Robot (Quadrotor) With Intelligent PID and State feedback LQR Controller: A Comparative Approach. 2019 16th International Bhurban Conference on Applied Sciences and Technology (IBCAST).

[6] Gomez N, Gomez V, Paiva E, JER Benítez and Gregor R. 2020. Flight Controller Optimization of Unmanned Aerial Vehicles using a Particle Swarm Algorithm. The 2020 International Conference on Unmanned Aircraft Systems: ICUAS'20.

[7] Ma M, Xu Z, Chang C and Xue J. 2016. Research on Four-Rotor UAV Control System Based on PID and LQR. Measurement & Control Technology, 35(10), 42-55.

[8] Matouk D, Gherouat O, Abdessemsd F and Hassam A. 2016. Quadrotor Position and Attitude Control via Backstepping Approach. 8th International Conference on Modelling, Identification and Control (ICMIC 2016). 672-677.

[9] Huang T, Li B, Shah A, Qin N and Huang D. 2019. Fuzzy Sliding Mode Control for a Quadrotor UAV. IEEE 8th Data Driven Control and Learning Systems Conference (DDCLS).

DOI: 10.54254/2755-2721/20/20231101

[10] Sheng G and Gao G. 2019. Research on the Attitude Control of Civil Quad-Rotor UAV Based on Fuzzy PID Control. 31st Chinese Control and Decision Conference (CCDC), 4566-4569.
[11] Wang X, Sun C, Lin X and Yu Y. 2020. Simulation Study on Adaptive Attitude Control for a Quadrotor Based on Neural Network. Computer Integrated Manufacturing Systems, 37(03) ,37-129.
[12] Liang C, LIU X, Zhang X and Huang J. 2021. Design of control law for quadrotor UAV based on reinforcement learning. Computer Measurement & Control, 29(02), 71-91.
[13] Yu R, Zhao W and Cheng R. 2019. Research on neural network PID control algorithm for quadrotor aircraft. modern electronics technique, 42(10), 108-112.
[14] Huo X, Huo M and Karimi HR. 2014. Attitude Stabilization Control of a Quadrotor UAV by Using Backstepping Approach. Mathematical Problems in Engineering (Pt.3).
[15] Zhou Z, Chen Z and Zhao Z. 2019. Overview of Flight Control Algorithms for quadrotor UAV. Network Security Technology & Application, (09), 33-36
[16] Alexis K, Papachristos C, Nikolakopoulos, G and Tzes A. 2011. Model predictive quadrotor indoor position control. Control & Automation. IEEE.
[17] Zhao Y, Yu X and Qi J. 2022. Speed Control System of Stepping Motor Based on BP Neural Network PID. 2022 IEEE 10th Joint International Information Technology and Artificial Intelligence Conference (ITAIC).
[18] Guo X, Li Z and Sun G. 2019. The Robot Arm Control Based on RBF with Incremental PID and Sliding Mode Robustness. 2019 WRC Symposium on Advanced Robotics and Automation (WRC SARA), pp. 97-102
[19] Yu W, Li J and Yang K. 2018. Research on Fuzzy Adaptive Stabilization PID Control System. 2018 IEEE 3rd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), pp. 2037-2043