Hindawi
International Journal of Aerospace Engineering
Volume 2022, Article ID 9322534, 18 pages
https://doi.org/10.1155/2022/9322534

Research Article

3-DOF Position and Orientation Control of an Air Flotation Platform for Spacecraft Ground Microgravity Simulation by Using Double Closed-Loop Cascade PIDnn

Siyue Wang,[1,2] Qing Gao,[3] Jinguo Liu,[2,4] Jingkai Feng,[2,4,5] and Jingshu Liufu[2,6]

[1]*Shenyang Ligong University, Shenyang 110158, China*
[2]*State Key Laboratory of Robotics, Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang 110016, China*
[3]*Institute of Robotics and Intelligent Manufacturing & School of Science and Engineering, The Chinese University of Hong Kong, Shenzhen 518172, China*
[4]*Institute of Robotics and Intelligent Manufacturing, Chinese Academy of Sciences, Shenyang 110169, China*
[5]*University of Chinese Academy of Sciences, Beijing 100049, China*
[6]*University of Science and Technology Liaoning, Anshan 110325, China*

Correspondence should be addressed to Jinguo Liu; liujinguo@sia.cn

Received 24 April 2022; Revised 5 July 2022; Accepted 1 August 2022; Published 22 August 2022

Academic Editor: Adel Ghenaiet

Space assistant robots can help astronauts or their assistants perform certain tasks. A ground microgravity simulation environment is built for the space assistant robot AAR-2. The hardware requirements of the ground simulation by the 3-DOF microgravity air flotation platform. An algorithm is designed for this simulation system. By using momentum and RMSprop methods to improve the PID neural network, the challenging problem of strong coupling between system nonlinearity and variables is solved. Firstly, the paper introduces the hardware system and deduces the dynamic model of the system. Then, the algorithm is calculated and simulated. Through simulation, the effectiveness and feasibility of the algorithm are compared and proved. Finally, the control system is simulated by MATLAB/Simulink and compared with other advanced algorithms. The simulation results show that the designed neural network controller can quickly and accurately control the 3-DOF of freedom motion of AAR-2.

1. Introduction

With the continuous development of aerospace, many countries have carried out the research and development of space robots to replace astronauts to perform some specific tasks, for example, space rendezvous and docking, space debris avoidance [1], fuel supply, on orbit maintenance [2], on orbit component reconstruction, etc. In the past, most of these tasks were completed by astronauts, mainly due to the complexity of the space environment and the arduousness of the tasks. The characteristics of space greatly limit the astronauts' activities inside and outside the capsule, and the complex and cumbersome operations greatly increase the astronauts' work and psychological pressure. Therefore, in recent years, more and more space operation tasks gradually turn to space robots.

For the design, manufacture, and operation of space robots, simulation experiments are required in the ground microgravity environment for overall performance evaluation, component testing, key parameter determination, and various system verifications [3]. Therefore, it directly promotes the research on building a space microgravity environment. If you want to conduct a space robot simulation experiment, you need to build a microgravity experiment environment on the ground [4].

The methods of constructing space microgravity environment mainly include tower dropping method [5, 6], water floating method, and air floating method [7, 8]. The falling tower method is a method to generate microgravity experimental environment by performing free falling motion in microgravity tower (well). It is usually carried out by building high towers or digging deep wells on the ground. Because of the special experimental environment and the limited weightlessness time, this method has great limitations, so it is not commonly used as a method to build a microgravity environment in the laboratory. The water floating method refers to a state in which when the density of the object is the same as that of the liquid. The object can be suspended at any point in the liquid. However, this method does not have real-time operability in the experimental process and needs to constantly adjust the counterweight to ensure balance and stability. On the other hand, it also requires that the experimental equipment can work normally under the liquid without affecting the accuracy. Therefore, this method will bring higher cost and experimental operation requirements to the test. However, the air flotation platform has the advantages of low cost, low operation difficulty, and sustainable real-time operation, so most experiments use the air flotation platform to simulate the microgravity environment [9, 10]. The core component of air flotation platform is gas bearing, which can be divided into plane gas bearing and spherical gas bearing. The compressed air flows out through the small hole or slit of the gas bearing and forms an air film between the workbench and the gas bearing, so as to achieve the purpose of frictionless movement. The air flotation platform has no contact with the worktable and produces a suspended effect, so it overcomes the influence of gravity to a great extent. Therefore, this paper selects the air flotation platform as the ground microgravity simulation platform of the space assistant robot and simulates the motion of the space assistant robot through the 3-DOF of air flotation platform.

Many scholars have different views and studies on the position and attitude control of spacecraft in microgravity environment. Terui et al. proposed a sliding mode control and state-dependent Riccati equation method to achieve 6-DOF position and attitude maneuverability without uncertainty and disturbance [11, 12]. In the follow-up, high-pass filter and output feedback control law are used to solve the problem of translational and attitude coupling, but the stability and convergence time are unconstrained [13]. In order to solve the problem of model uncertainty and environmental disturbance, Wu et al. proposed a finite time controller based on nonsingular terminal sliding mode control technology, which can control the translation and attitude change of spacecraft [14]. Liu et al. realized the 3-DOF free attitude control of AAR-1. In order to meet the characteristics of multivariable, strong coupling, and nonlinear systems, the traditional PIDnn controller is used to quickly achieve stability and control effect [15]. Gao et al. upgraded the functions of AAR-1, designed AAR-2, and realized high-precision control of AAR-2 by combining synovial control and fuzzy control [16]. Malladi et al. adopted the nonlinear model predictive control (NMPC) method to solve the highly coupled and nonlinear problem of position and attitude [17]. Aiming at the fast attitude dynamics and slow position dynamics caused by time scale separation, a nonlinear hierarchical control law was proposed [18].

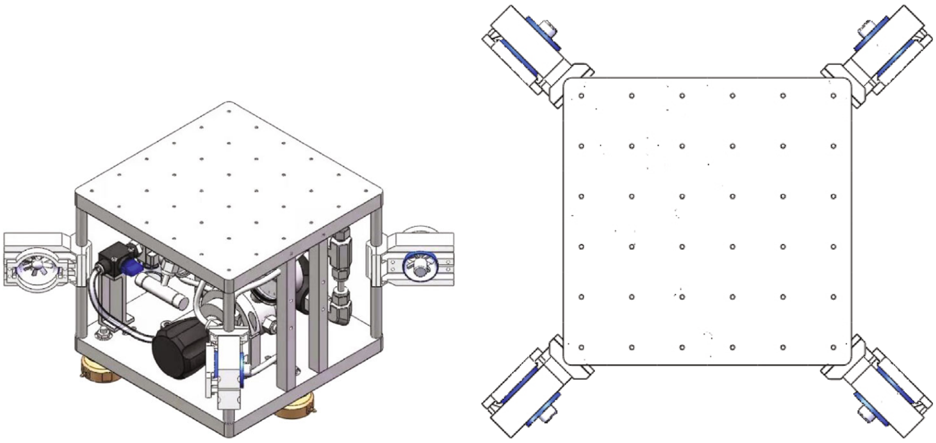
The attitude and translational motion of spacecraft are coupled and highly nonlinear, so the model parameters of spacecraft cannot be accurately obtained, and the spacecraft is always affected by environmental disturbances. All these problems make it difficult for the spacecraft control system to achieve the desired control performance. Due to the high coupling and nonlinearity of the dynamic model, this paper uses the double closed-loop structure to solve the influence of strong coupling on the system and uses the improved PID neural network to solve the problem of nonlinear system. Because the uncertain environmental disturbance always exists in the spacecraft movement, the neural network in machine learning can better cater to the uncertainty to a certain extent and match the changing environment through continuous learning.

The main innovations and contributions of this paper are as follows: (a) we use a double closed-loop system to solve the problem of the strong coupling and environmental disturbance in the model. This control structure can better overcome the impact of uncertainty on the system by matching with the neural network. (b) We optimized the PID neural network algorithm to solve the variable coupling problem and nonlinear problem when the spacecraft position and attitude change, so that the microgravity air flotation system can converge faster and reach the equilibrium position, better simulate the spacecraft motion and meet the experimental needs. (c) A complete control system and algorithm are built on the microgravity air flotation platform, which makes the flight simulation experiment of ground simulation spacecraft more efficient and convenient, and promote the development of ground microgravity environment simulation equipment.

The rest of this paper is organized as follows: in Section 2, the control system of the actual microgravity air flotation platform is designed and the motion principle is analyzed. In Section 3, the dynamic model of the microgravity air flotation platform is built and its motion characteristics are analyzed. In addition, the model is optimized to a certain extent according to the ideal environment of the actual space station. In Section 4, the control algorithm of the microgravity air flotation platform is improved on the basis of PIDnn. Meanwhile, the stability of the system is proved and the feasibility of the algorithm is simulated. In Section 5, the whole improved microgravity air flotation system is simulated. The feasibility of the scheme is verified by comparing other advanced algorithms and combining the theoretical simulation with the actual platform.

2. Design of the Control System Based on Air Flotation Platform

2.1. Control System Hardware Design. The microgravity simulation air flotation platform involved in this paper is shown in Figure 1(a). It can realize two-dimensional microgravity simulation motion on the marble plane through three air flotation pads, gas cylinders, and air pressure transmission system. Four ducted fans are mounted on it to control the movement of the air flotation platform. The installation position is shown in Figure 1(b). Four ducted fans at 45° angles to the side of the air flotation platform are installed at the four corners of the platform. Microspacecraft, such



(a) Air flotation platform model

(b) Top view of model

Figure 1: Structural model of air flotation platform.

as astronaut assistance robots AAR-1, AAR-2 [19, 20], etc., can be installed on the air flotation platform, as shown in Figure 2(a). By controlling the changes of the thrust of the four ducted fans, the two-dimensional plane motion of the spacecraft can be simulated in a microgravity environment.

In this paper, the actual platform is shown in Figure 2(b). The STM series chip STM32f1 is selected as the main chip, equipped with two ultrasonic ranging modules HC-SR04, a six-axis gyroscope sensor MPU6050 (built-in DMP attitude calculation), and four ducted fans (brush less motor HL3508).

Figure 3 shows the top view of the air flotation platform, $O-X_nY_nZ_n$ is the inertial coordinate system, $O-X_bY_bZ_b$ is the geometric coordinate system, the arrow direction is the wind direction output by the culvert fan, and the four culvert fans are combined in pairs to control the forward and rotational degrees of freedom of the air flotation platform. The system forms a closed-loop structure through sensors, so as to achieve more efficient control effect.

2.2. Principles of Movement. To correctly describe the motion of the microgravity simulated air flotation platform, the two coordinate systems involved in the system are analyzed and processed. The motion pose of the air flotation platform is determined by the angle between the base point o and the axes X_b and X_n (the initial state is that the x-axis and the y-axis are parallel to each other), the coordinate of the base point is (x_0, y_0), the rigid body gestures are determined by x_0, y_0, and θ. The basic motion unit is composed of translation and rotation in the directions of X_n and Y_n (3-DOF), and the rotation is the rotation around the center of mass. F_1, F_2, F_3, and F_4 are the thrust (wind force) of the four ducted fans.

$$F_{X_b} = m\ddot{X}_b = \frac{\sqrt{2}}{2}(F_1 + F_4 - F_2 - F_3), \tag{1}$$

$$F_{Y_b} = m\ddot{Y}_b = \frac{\sqrt{2}}{2}(F_1 + F_2 - F_3 - F_4), \tag{2}$$

$$m_z^b = J_c\ddot{\theta} = \frac{\sqrt{2}}{2}(F_1 + F_4 - F_2 - F_3)(r + r_s). \tag{3}$$

It is known that the aerial view of the actual air flotation platform is a square, r is the side length, and r_s is the radius of the ducted fan. Let clockwise to be the positive direction, and m_z^b is the torque rotating around Z_b with the center of mass of the air flotation platform.

3. System Dynamics Model

3.1. Coordinate and Attitude Transformation. The pose of the air flotation platform is expressed as

$$\eta_{b,n}^n = \begin{bmatrix} p_{b,n}^n \\ \theta_{nb} \end{bmatrix}. \tag{4}$$

The instantaneous speed is expressed as

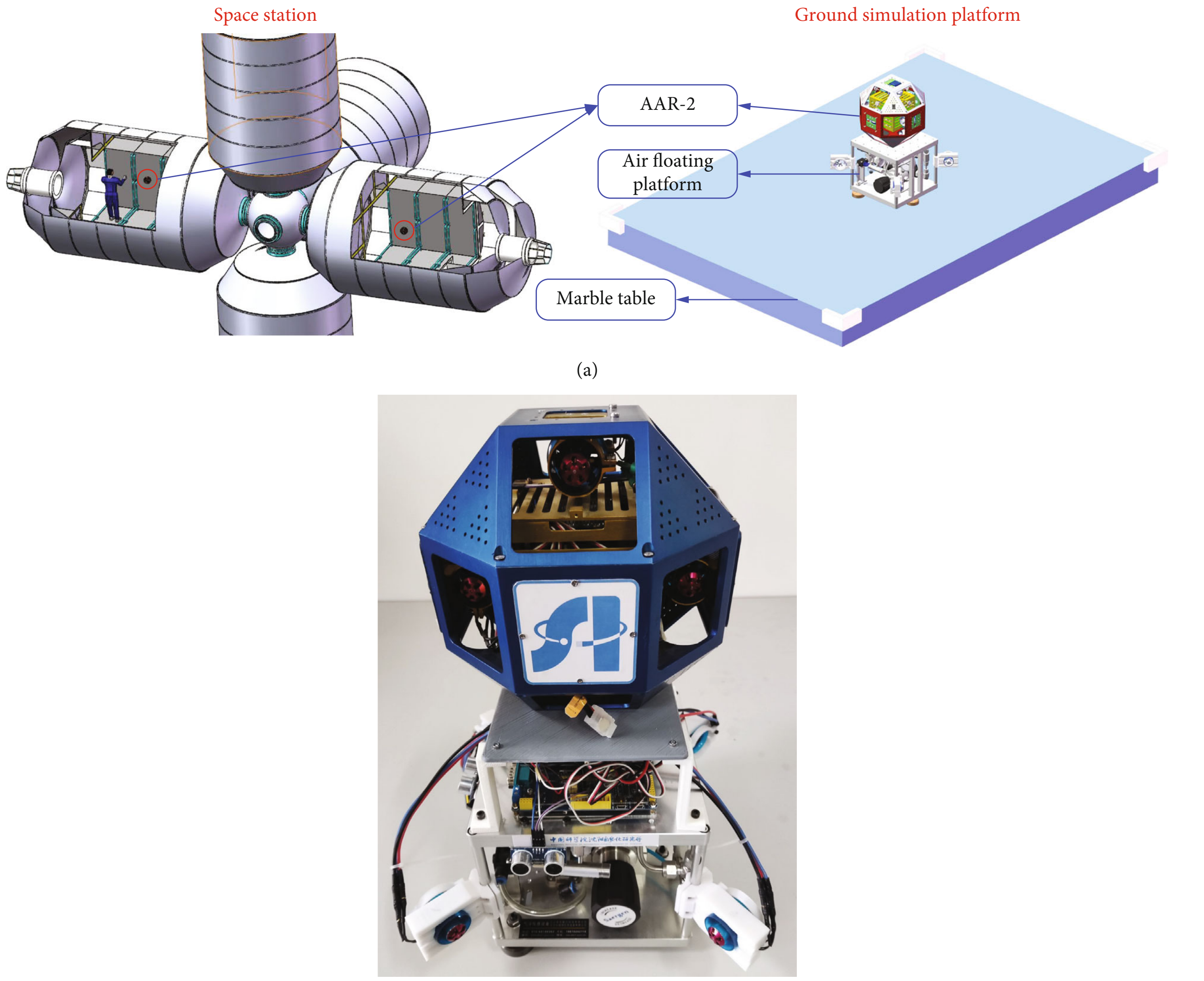
$$V_{b,n}^n = \begin{bmatrix} v_{b,n}^n \\ w_{b,n}^n \end{bmatrix}. \tag{5}$$

Then, we can get

$$\acute{\eta}_{b,n}^n = \begin{bmatrix} \acute{p}_{b,n}^n \\ \acute{\theta}_{nb}^n \end{bmatrix} = \begin{bmatrix} R_b^n(\theta_{nb}) & 0_{3*3} \\ 0_{3*3} & T_\theta(\theta_{nb}) \end{bmatrix} \begin{bmatrix} v_{b,n}^n \\ w_{b,n}^n \end{bmatrix}, \tag{6}$$

where $\eta_{b,n}^n$ represents the attitude of b relative to the inertial system n under the inertial system n, $R_b^n(\theta_{nb})$ represents the rotation matrix, and $\theta_{nb} = [\phi \quad \theta \quad \psi]^T$ is the geometric position b Euler angles to inertial position n.

$R_b^n(\theta_{nb})$ and $T_\theta(\theta_{nb})$ are the results of the transformation from the inertial coordinate system to the geometric coordinate system, and there are many ways to obtain them, which are explained in the order of Z-Y-X [21].



(a)

(b)

Figure 2: (a) Schematic diagram of space station and ground simulation platform. The AAR-2 is a free flying robot used in the space station cabin to assist astronauts in some space missions. This robot can use the microgravity simulation air flotation platform to simulate the 3-DOF plane motion on the marble table. (b) Actual experimental platform.

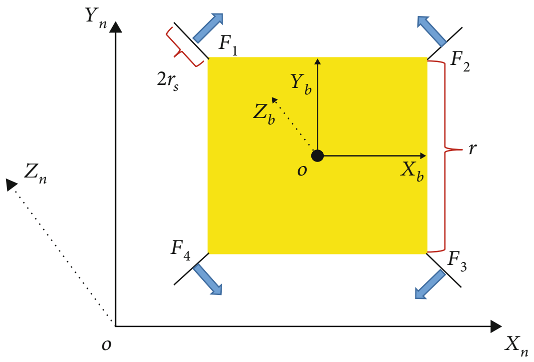


Figure 3: Top view of air flotation platform.

$$R_b^n(\theta_{nb}) = R_\phi R_\theta R_\phi = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\phi & s\phi \\ 0 & -s\phi & c\phi \end{bmatrix} \begin{bmatrix} c\theta & s\theta & 0 \\ -s\theta & c\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c\psi & 0 & -s\psi \\ 0 & 1 & 0 \\ s\psi & 0 & c\psi \end{bmatrix}$$
$$= \begin{bmatrix} c\theta c\psi & s\theta & -c\theta s\psi \\ s\phi s\psi - c\phi s\theta c\psi & c\phi c\theta & s\phi c\psi - c\phi s\theta s\psi \\ c\phi s\psi + c\phi s\theta s\psi & -s\phi c\theta & c\phi c\psi - s\phi s\theta s\psi \end{bmatrix}, \tag{7}$$

$$T_\theta(\theta_{nb}) = \begin{bmatrix} 1 & 0 & -s\theta \\ 0 & c\phi & s\phi c\theta \\ 0 & -s\phi & c\phi c\theta \end{bmatrix}^{-1} = \begin{bmatrix} 1 & s\phi t\theta & c\phi t\theta \\ 0 & c\phi & -s\phi \\ 0 & \dfrac{s\phi}{c\theta} & \dfrac{c\phi}{c\theta} \end{bmatrix}. \tag{8}$$

The above sine, cosine, and tangent functions are replaced by s, c, and t. When the motion mode of the air flotation platform is fixed as translation and rotation, then

$$v^n_{b,n} = \left[\dot{X}_b, \dot{Y}_b, 0\right]^T, \tag{9}$$

$$\theta_{nb} = [0 \quad 0 \quad \psi]^T, \tag{10}$$

$$R^n_b(\theta_{nb}) = \begin{bmatrix} c\psi & 0 & -s\psi \\ 0 & 1 & 0 \\ s\psi & 0 & c\psi \end{bmatrix}, \tag{11}$$

$$T_\theta(\theta_{nb}) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{12}$$

3.2. Dynamics Model. To further accurately analyze the motion state of the microgravity simulated air flotation platform, the dynamic model of the air flotation platform is deduced and simulated in this paper. The dynamic model of the air flotation platform can be deduced as follows [22–24]:

$$m\left(\dot{v}^b_{g,n} + w^b_{g,n} \times v^b_{g,n}\right) = F^g + f^g, \tag{13}$$

$$I\dot{w}^b_{g,n} + w^b_{g,n} \times \left(Iv^b_{g,n}\right) = M^g + m^g. \tag{14}$$

Among them, g represents the center of gravity and $F^g = [F^g_x, F^g_y, F^g_z]^T$ is used to represent gravity, buoyancy, and other external forces acting on an air flotation platform. $f^g = [f^g_x, f^g_y, f^g_z]^T$ is used to represent the thrust generated by the rotation of the ducted fan. $M^g = [M^g_x, M^g_y, M^g_z]^T$ is used to represent all external moments acting on the air flotation platform except the moment generated by the ducted fan. $m^g = [m^g_x, m^g_y, m^g_z]^T$ indicates the moment acting on the air flotation platform generated by the rotation of the ducted fan.

Since the air flotation platform is a rigid body and the additional mass is the same in all directions, the total inertia matrix about the center of gravity is

$$I = \begin{bmatrix} I_x & 0 & 0 \\ 0 & I_y & 0 \\ 0 & 0 & I_z \end{bmatrix}. \tag{15}$$

Since the air flotation platform needs to move freely on the marble tabletop, its motion mode is 3-DOF motion, and the yaw motion can be freely performed while translating on the two-dimensional marble tabletop:

$$v^b_{g,n} = \left[v^b_{x_{g,n}}, v^b_{y_{g,n}}, 0\right]^T, \tag{16}$$

$$w^b_{g,n} = \left[0, 0, w^b_{z_{g,n}}\right]^T. \tag{17}$$

When the air flotation platform rotates and flies forward at the same time, a lateral force is generated, which is represented by $w^b_{g,n} \times v^b_{g,n}$. This term is offset by the air resistance created when the microgravity simulated air flotation platform moves. This force can also be ignored under the assumption that the spin speed is almost zero in forward flight and almost zero in spinning motion.

In the inertia matrix, we can get $I_x = I_y$, if the roll and pitch velocities are zero, no rotational moment will be generated. When the rolling and pitching moments are not zero, $w^b_{g,n} \times (Iv^b_{g,n})$ has little effect on the roll and pitch moments, so they are easily offset by the resistance of gravity, so they are also ignored.

The model can thus be simplified to

$$m\dot{v}^b_{x_{g,n}} = f^g_x + F^g_x = f^g_x - k_x\dot{x}, \tag{18}$$

$$m\dot{v}^b_{y_{g,n}} = f^g_y + F^g_y = f^g_y - k_y\dot{y}, \tag{19}$$

$$I_z\dot{w}^b_{z_{g,n}} = m^g_z + M^g_z = m^g_z + M^g_{z_{\text{cor}}} + M^g_{z_d} - k_z w_z(r + r_s). \tag{20}$$

Among them, $M^g_{z_{\text{cor}}}$ refers to the Coriolis torque in the direction of rotation around the z-axis, $M^g_{z_d}$ refers to the average disturbance torque on the z-axis, and k_z represents the average torque coefficient in the air resistance on the z-axis.

It is easy to design the controller with negligible Coriolis torque, disturbance torque, and air resistance. In the motion state, we are only interested in the dynamic equations of Euler angles in the geometric coordinate system in motion. Therefore, the dynamic model of the air flotation platform is further simplified as follows:

$$m\ddot{X}_b = c\psi f^b_x, \tag{21}$$

$$m\ddot{Y}_b = f^b_y, \tag{22}$$

$$I_z\ddot{\psi} = m^b_z. \tag{23}$$

4. Algorithmic Controller Design

4.1. PIDnn Controller Design. PIDnn is a kind of PID-type controller that relies on the self-adaptation and learning ability of the neural network algorithm [25]. There are various neural network structures that can be designed. This paper adopts a single-layer forward propagation network, in which the input layer, hidden layer, and output layer contain two, three, and one neurons (perceptrons), respectively [26].

The input of the PID neural network system has two parts, which are the expected input and the real-time output of the system. After each iteration, the neural network outputs new values of P, I, and D as three parameters of the

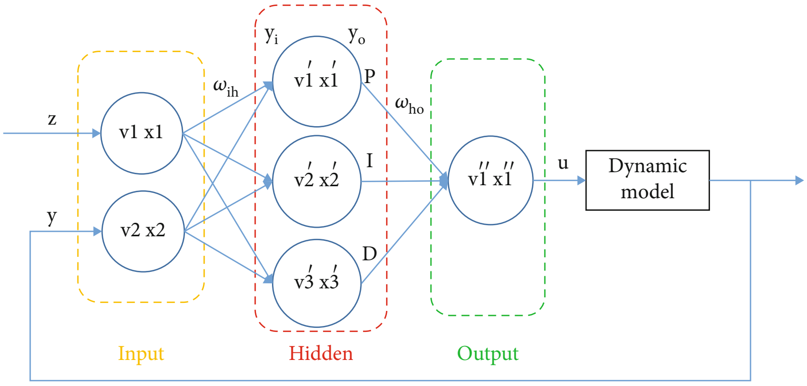


Figure 4: PIDnn control algorithm structure.

PID neural network controller [27, 28]. After processing by the dynamic model, as the new input of the system continues to iterate, the system tends to stabilize after the number of sampling N approaches a larger value [29]. The structure of the PIDnn control algorithm is shown in Figure 4.

4.1.1. Forward Algorithm. When iterating to the nth time, the input and output of neurons in each layer are expressed as follows:

Input layer:

$$v_1(n) = z(n), \tag{24}$$

$$v_2(n) = y(n), \tag{25}$$

$$x_i(n) = \mathrm{NET}[v_i(n)], i = 1, 2, \tag{26}$$

where $\mathrm{NET}(n)$ is an activation function. To reduce the actual hardware system code calculation amount, it can be set as an identity activation function here, namely,

$$\mathrm{NET}[v_i(n)] = v_i(n). \tag{27}$$

Hidden layer:

$$v_h'(n) = \sum_{i=1}^{2} \omega_{ih} x_i(n), h = 1, 2, 3. \tag{28}$$

The activation function of the hidden layer is special. Considering that the neural network serves the PID system, the three neurons here perform their proportional, integral, and differential functions, respectively. After discretizing the continuous system, we get the following:

Proportion:

$$x_1'(n) = v_1'(n). \tag{29}$$

Integral:

$$x_2'(n) = x_2'(n-1) + v_2'(n). \tag{30}$$

Derivative:

$$x_3'(n) = v_3'(n) - v_3'(n-1). \tag{31}$$

Output layer:

$$v_1''(n) = \sum_{h=1}^{3} W_{ho} x_h'(n), o = 1, \tag{32}$$

$$x_o''(n) = v_o''(n). \tag{33}$$

4.1.2. Back-Propagation Algorithm. The back-propagation algorithm can achieve the effect of learning and memory by modifying the weight value (ω_{ih} and W_{ho}) of the neural network. First, calculate the systematic error (the ideal value minus the actual value) and then update the weight of each layer of the neural network according to the conventional gradient descent method. When taking the $(n+1)$th sampling, the error $E = z(n) - y(n)$. Let learning step size be η, after the nth learning, the weight value between the hidden layer and the output layer is changed to

$$W_{ho}(n+1) = W_{ho}(n) - \eta \frac{dE}{dW_{ho}}, \tag{34}$$

$$\frac{dE}{dW_{ho}} = \frac{dE}{dy_o} \frac{dy_o}{dx_1''} \frac{dx_1''}{dv_1''} \frac{dv_1''}{dW_{ho}}. \tag{35}$$

The weight value between the input layer and the hidden layer is changed to

$$\omega_{ih}(n+1) = W_{ih}(n) - \eta \frac{dE}{d\omega_{ih}}, \tag{36}$$

$$\frac{dE}{d\omega_{ih}} = \frac{dE}{dy_o} \frac{dy_o}{dx_1''} \frac{dx_1''}{dv_1''} \frac{dv_1''}{dx_h'} \frac{dx_h'}{dv_h'} \frac{dv_h'}{d\omega_{ih}}. \tag{37}$$

The weights between hidden layer neurons and output layer neurons are real-time P, I, and D coefficients.

4.2. Momentum and RMSprop Jointly Improve the Algorithm. The iterative principle of the standard gradient descent method is

$$W(n) = W(n-1) - \eta \Delta W(n), \tag{38}$$

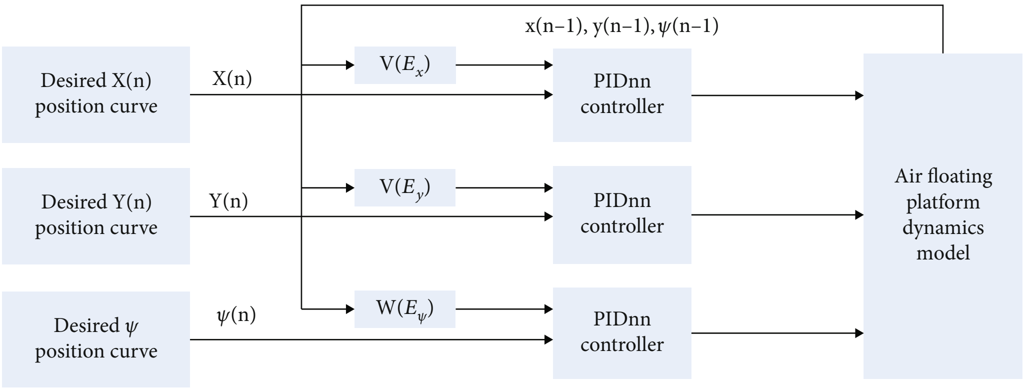


Figure 5: Air flotation platform control system.

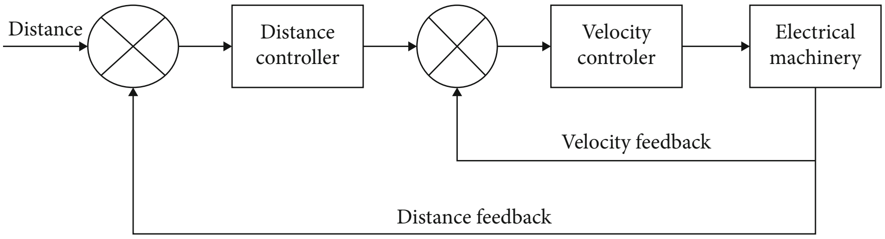


Figure 6: Cascade control structure.

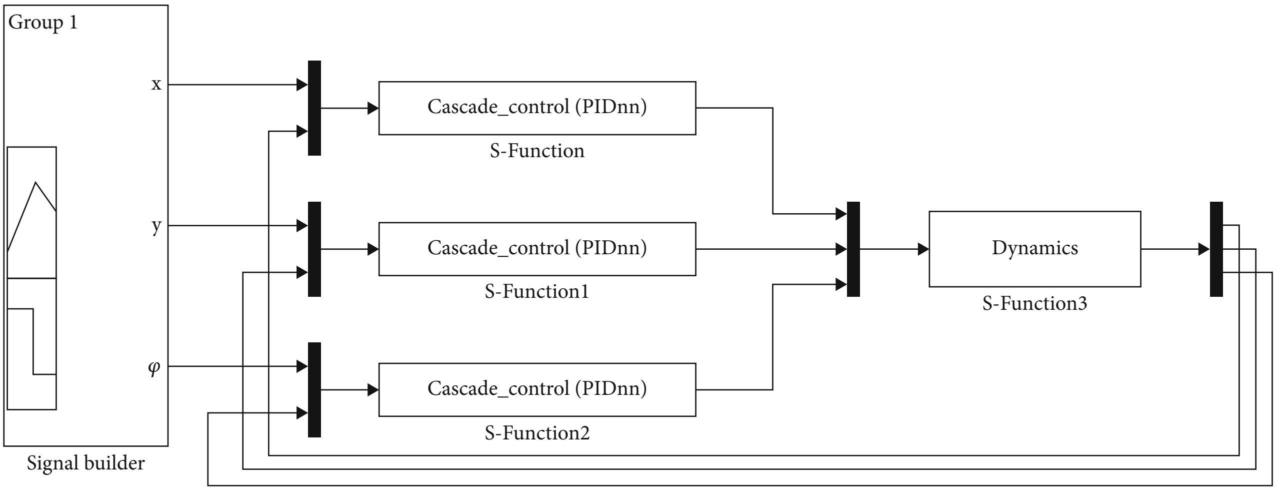


Figure 7: Simulink simulation structural diagram.

$$\Delta W(n) = \frac{\partial E[W(n-1)]}{\partial W}, \mathrm{n} = 1, 2, 3 \cdots . \tag{39}$$

The momentum method changes the weight update law in the standard gradient descent method and introduces an intermediate quantity:

$$V(n) = \beta V(n-1) + (1-\beta)\Delta W(n). \tag{40}$$

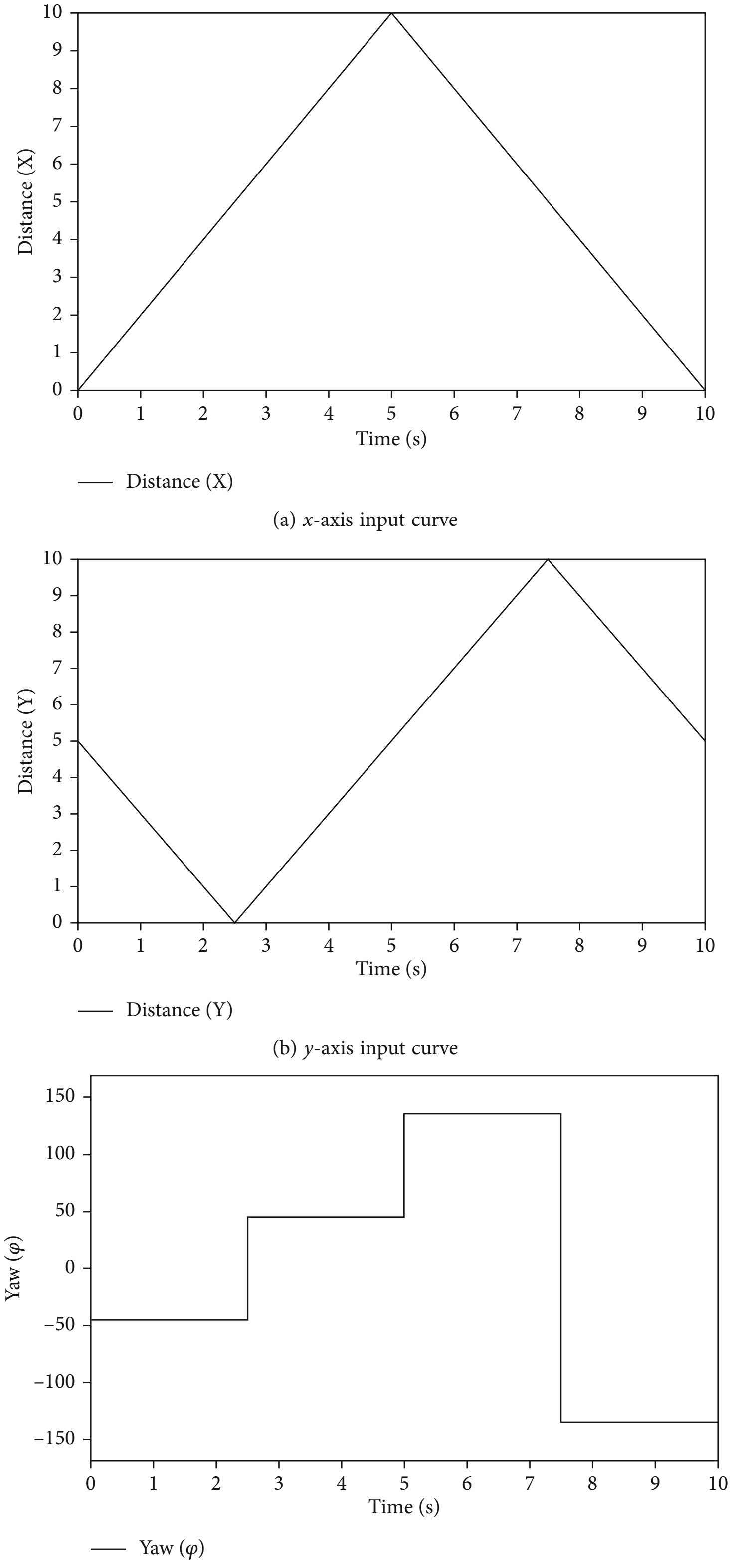
And replacing the original $\Delta W(n)$ with $V(n)$ can get

$$\begin{aligned} W(n) &= W(n-1) - \eta V(n) \\ &= W(n-1) - \eta[\beta V(n-1) + (1-\beta)\Delta W(n)], \end{aligned} \tag{41}$$

$$\begin{aligned} W(n) = W(n-1) - \eta(1-\beta)[\Delta W(n) + \beta \Delta W(n-1) \\ + \beta^2 \Delta W(n-2) + \cdots + \beta^{n-1} \Delta W(1)]. \end{aligned} \tag{42}$$

The value of β around 0.9 can make the algorithm achieve better results. Doing so is equivalent to a weighted sum of historically processed data. From the nth iteration, the farther historical data has a relatively small impact on the weight update, but the weight updated in the nearby iterations has a greater impact.

The method of adaptive adjustment of Adagrad learning rate is to convert η (learning rate) in Equation (43) into $\eta / \sqrt{S(n)} + \varepsilon$.

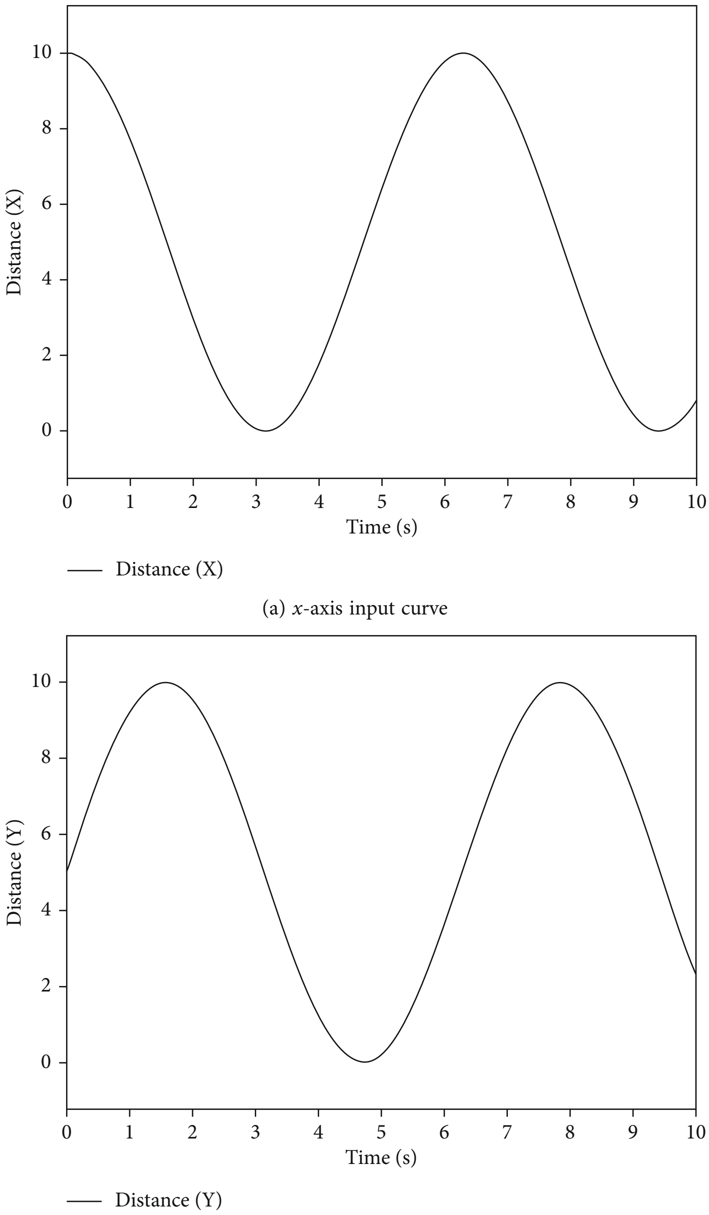


(a) *x*-axis input curve

(b) *y*-axis input curve

(c) Yaw-axis input curve

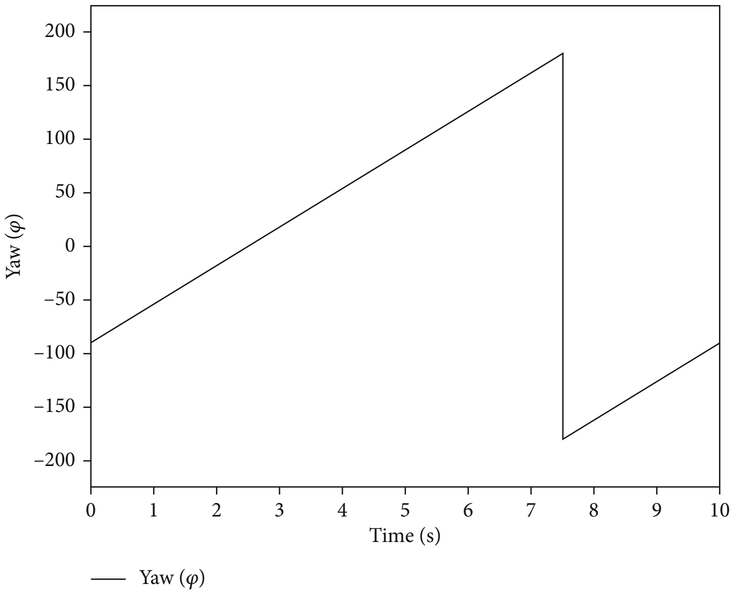
Figure 8: Square motion curve design.



(a) x-axis input curve

(b) y-axis input curve

Figure 9: Continued.



(c) Yaw-axis input curve

Figure 9: Circular curve design.

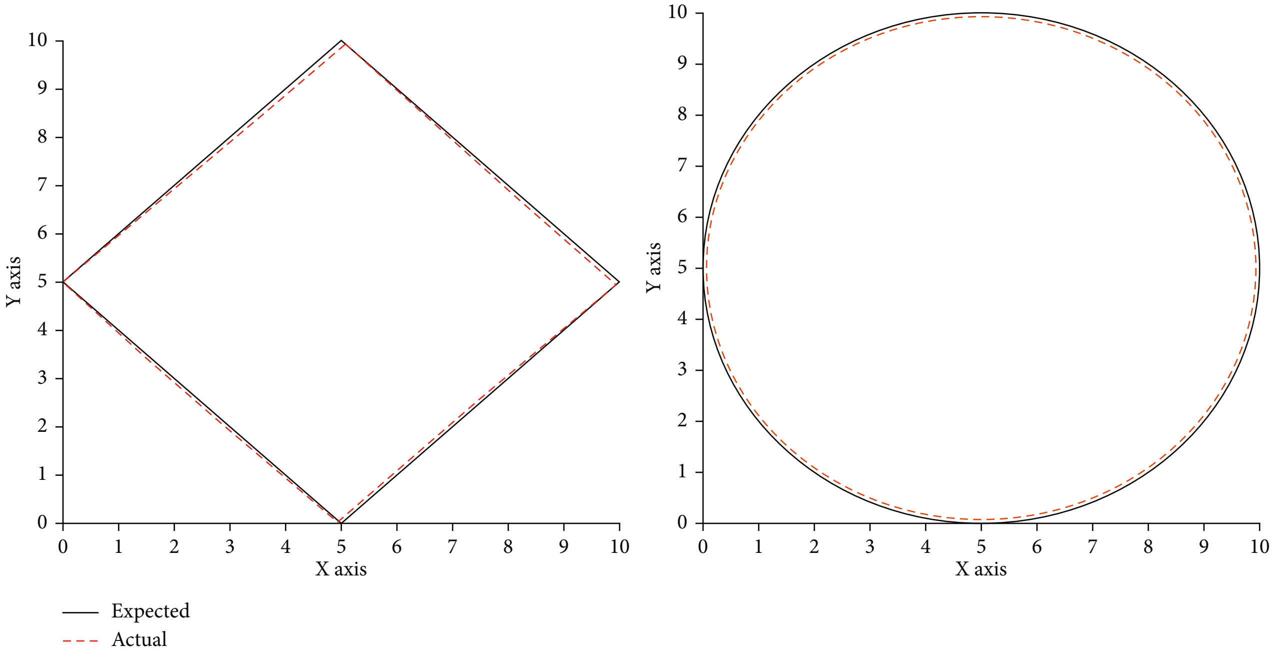


Figure 10: Comparison between control effect and expected effect.

$$W(n) = W(n-1) - \eta \Delta W(n). \tag{43}$$

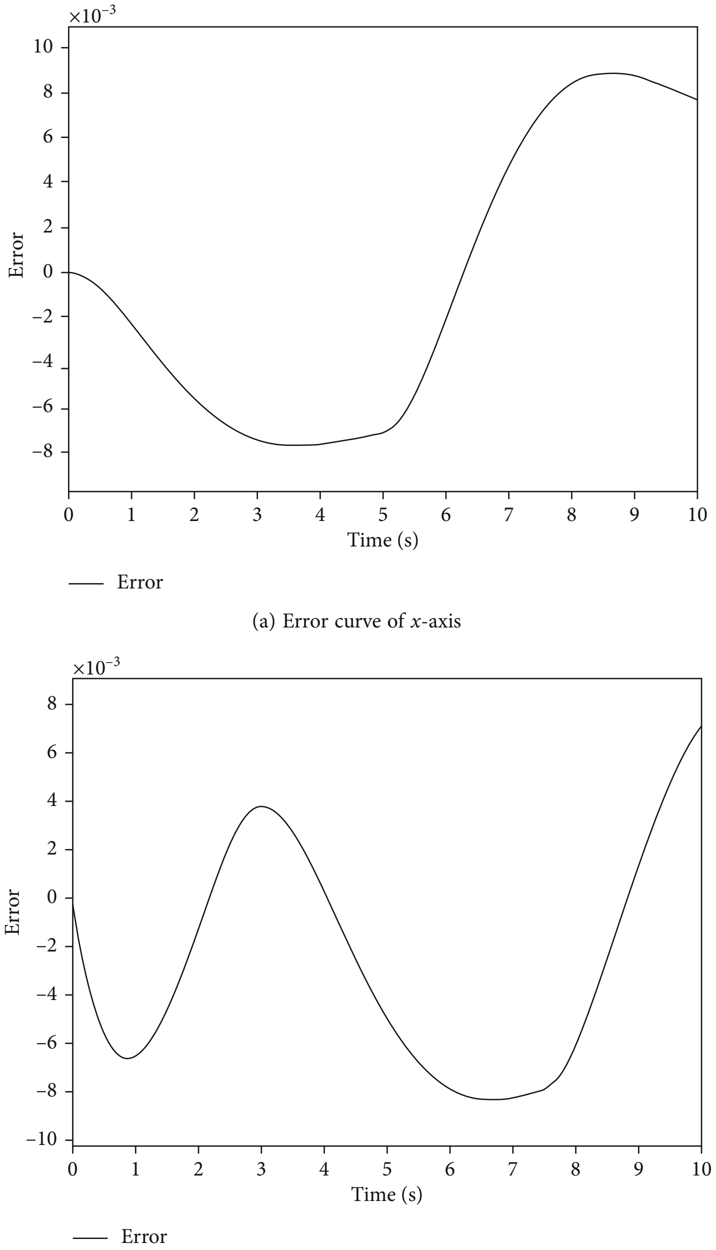
To prevent the denominator from being zero, set ε to a particularly small number.

$$S(n) = S(n-1) + \Delta W(n) \cdot \Delta W(n). \tag{44}$$

Then, we can get

$$W(n) = W(n-1) - \frac{\eta}{\sqrt{[\Delta W(1)]^2 + \cdots + [\Delta W(n)]^2 + \varepsilon}} \Delta W(n). \tag{45}$$

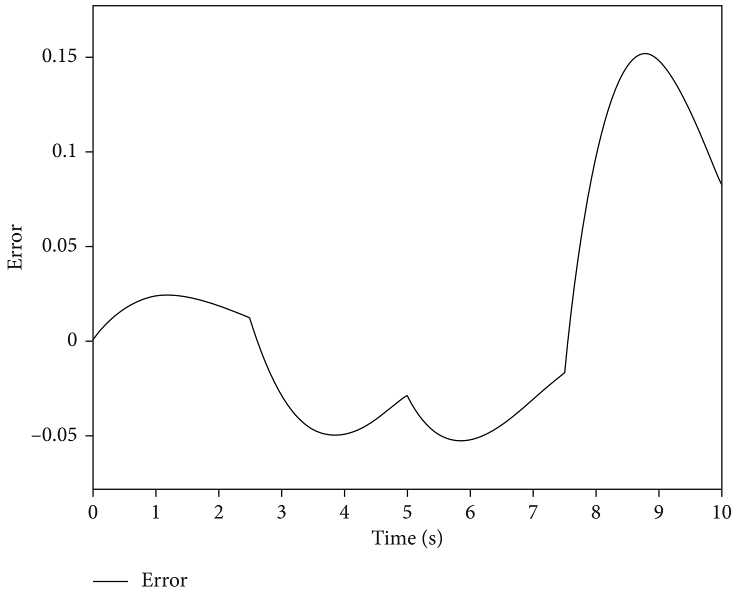
If the weights are modified too much, the learning rate



(a) Error curve of *x*-axis

(b) Error curve of *y*-axis

Figure 11: Continued.



(c) Error curve in yaw-axis

Figure 11: Error tracking of the first motion curve.

will decrease more. Then, in the process of gradient descent training, after a rapid change period, and then a platform area, the change of the learning rate in the platform area will be very slow. When it enters the fast period again after the plateau period, the decline is still very slow and cannot be modified, and all the historical data of the weights will be taken into account, and then, there is a problem.

So, the RMSprop method introduces Momentum on the learning rate, so that the formula becomes

$$S(n) = \alpha S(n-1) + (1-\alpha)\Delta W(n) \cdot \Delta W(n). \quad (46)$$

It is derived from Equation (46) that the gradient descent method in the PID neural network is finally optimized as

$$W(n) = W(n-1) - \frac{\eta(1-\beta)\left[\beta^{n-1}\Delta W(1)+\cdots+\beta^{0}\Delta W(n)\right]}{\sqrt{(1+\alpha)\left\{\alpha^{n-1}[\Delta W(1)]^2+\cdots+\alpha^{0}[\Delta W(n)]^2\right\}+\varepsilon}}, \quad (47)$$

where α and β have the same effect, so we let $\alpha = \beta = 0.9$.

4.3. Proof of System Stability. The system control input of the PID neural network controller is

$$u(n) = u(n-1) + \eta(n)\sum_{\mathrm{h}=1}^{3} x_h'(n)\, W_{ho}(n), \quad (48)$$

$$E(n) = \frac{1}{2}(r(n) - Y(n))^2, \quad (49)$$

$$\begin{aligned} \frac{dE(n)}{dx_h'(n)} &= \frac{dE(n)}{de(n)}\frac{de(n)}{dY(n)}\frac{dY(n)}{dz(n)}\frac{dz(n)}{dx_h'(n)} \\ &= -e(n)\frac{dy(n)}{dz(n)}W_{ho}(n)\eta(n). \end{aligned} \quad (50)$$

To analyze the stability of the system, a Lyapurov function $V(n) = (1/2)e(n)^2$ is constructed, where $e(n) = z(n) - y(n)$.

$$\Delta \mathrm{V(n)} = \mathrm{V(n+1)} - \mathrm{V(n)} = \frac{1}{2}\left(e(n+1)^2 - e(n)^2\right). \quad (51)$$

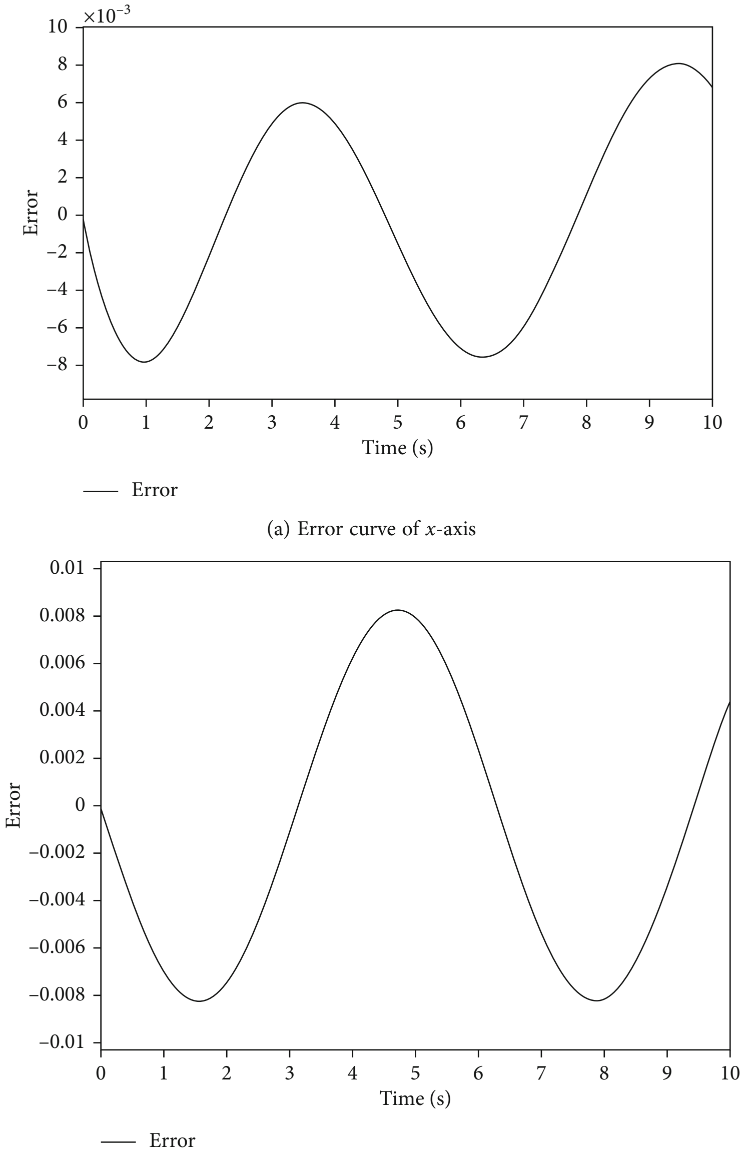
According to the differential theorem,

$$\Delta e(n) = \sum_{j=1}^{3}\frac{de(n)}{dx_h'(n)}\Delta x_h'(n). \quad (52)$$

According to Lyapurov function,

$$\begin{aligned} \frac{de(n)}{dx_h'(n)} &= \frac{de(n)}{dY(n)}\frac{dY(n)}{du(n)}\frac{du(n)}{dx_h'(n)} \\ &= -\operatorname{sgn}\left(\frac{dy(n)}{du(n)}\right)\eta(n)W_{ho}(n), \end{aligned} \quad (53)$$

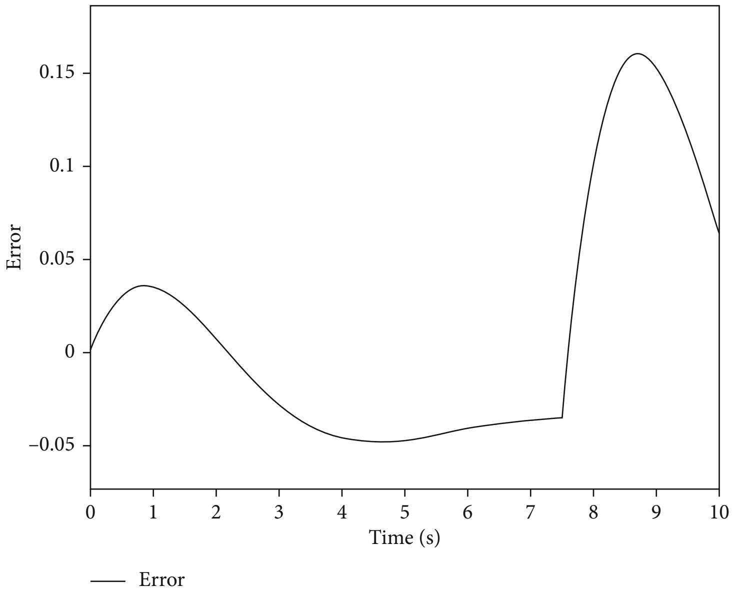
$$\begin{aligned} \Delta e(n) = -\sum_{j=1}^{3}\operatorname{sgn}\left(\frac{dY(n)}{du(n)}\right)\eta(n)W_{ho}(n) \times \eta(k)e(k)\operatorname{sgn} \\ \cdot\left(\frac{dy(n)}{du(n)}\right)W_{ho}(n). \end{aligned} \quad (54)$$



(a) Error curve of *x*-axis

(b) Error curve of *y*-axis

Figure 12: Continued.



(c) Error curve in yaw-axis

Figure 12: Error tracking of the second motion curve.

Because of $(\mathrm{sgn}\ (dy(n)/du(n)))^2 = 1$,

$$\Delta e(n) = -\sum_{j=1}^{3} (\eta(n))^2 (W_{ho}(n))^2 e(n), \tag{55}$$

$$e(n+1) = e(n) + \Delta e(n), \tag{56}$$

$$\begin{aligned}\Delta V(n) &= \frac{1}{2}\left(e(n+1)^2 - e(n)^2\right) = \frac{1}{2}\Delta e(n)(\Delta e(n) + 2e(n)) \\ &= e(n)\Delta e(n) + \frac{1}{2}\Delta e(n)^2.\end{aligned} \tag{57}$$

So we can come to a conclusion. The closed loop system is stable at $\Delta V(n) < 0$.

$$e(n)\Delta e(n) < -\frac{1}{2}\Delta e(n)^2, \tag{58}$$

$$\begin{aligned}&-e(n)\sum_{h=1}^{3} (\eta(n))^2 (W_{ho}(n))^2 e(n) \\ &< -\frac{1}{2}\left(-\sum_{h=1}^{3} (\eta(n))^2 (W_{ho}(n))^2 e(n)\right)^2,\end{aligned} \tag{59}$$

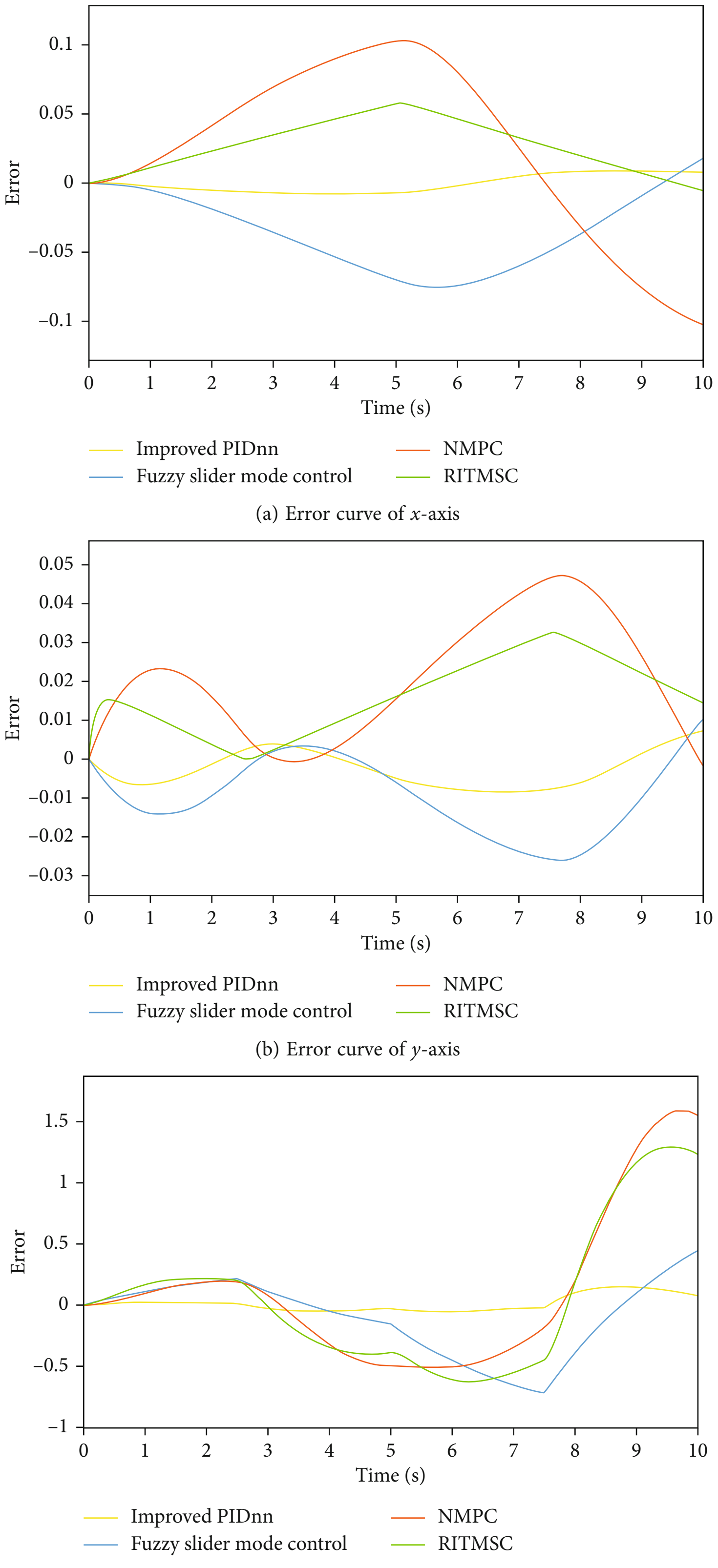
$$2\left(\sum_{h=1}^{3} (W_{ho}(n))^2\right)^{-1} > (\eta(n))^2. \tag{60}$$

Because the Lyapunov function has a lower bound and $\Delta V(n) < 0$, the function monotonically decreases, then when $n \longrightarrow \infty$, $V(n+1) = V(n) = 0, \Delta V(n) = 0$. Therefore, the neural network learning rate is set to a small value to make the system stable. Since the learning rate will affect the optimization speed of the gradient descent method, we have improved the neural network by momentum and RMSprop. By constantly changing the learning rate, the system can be stable and quickly reduced to the best.

5. Control System and Simulation Experiment

5.1. Control System. According to the actual experimental requirements of fixed-point and fixed-speed, the desired air flotation platform position (x, y), angle (ψ), and speed-error curve $[V(E_x), V(E_y), W(E_\psi)]$ are designed. For three control variables $[x(n), y(n), \psi(n)]$ are designed three cascade double-loop control models. The curve is sent to the algorithm controller for processing and then sent to the algorithm controller through the dynamic model for repeated iterations. The experiment combined the sensor to form a closed loop and combined with the cascade double-loop control to realize the stable control of the air flotation platform to move freely. The air flotation platform control system and cascade control structure are illustrated in Figures 5 and 6.

Considering that the actual trajectory may have sharp turns or circular motion, the following two motion models are designed. The first is a right-angled square and the second is a circular motion. In the first one, the motion curve of the air flotation platform starts from point $(0, 5)$ and returns to the origin $(0, 5)$ after moving counterclockwise for a week, and the overall consumption time is 10 seconds. And the initial direction of the air flotation platform is -90°, and the angle calculation is based on the angle between the direction of the air flotation platform and the x-axis. In the second one, the starting point, ending point, and elapsed time conditions of the movement curve of the air flotation platform remains unchanged. The motion curve presents a circular structure with a center of $(5, 5)$ and a radius of 5.

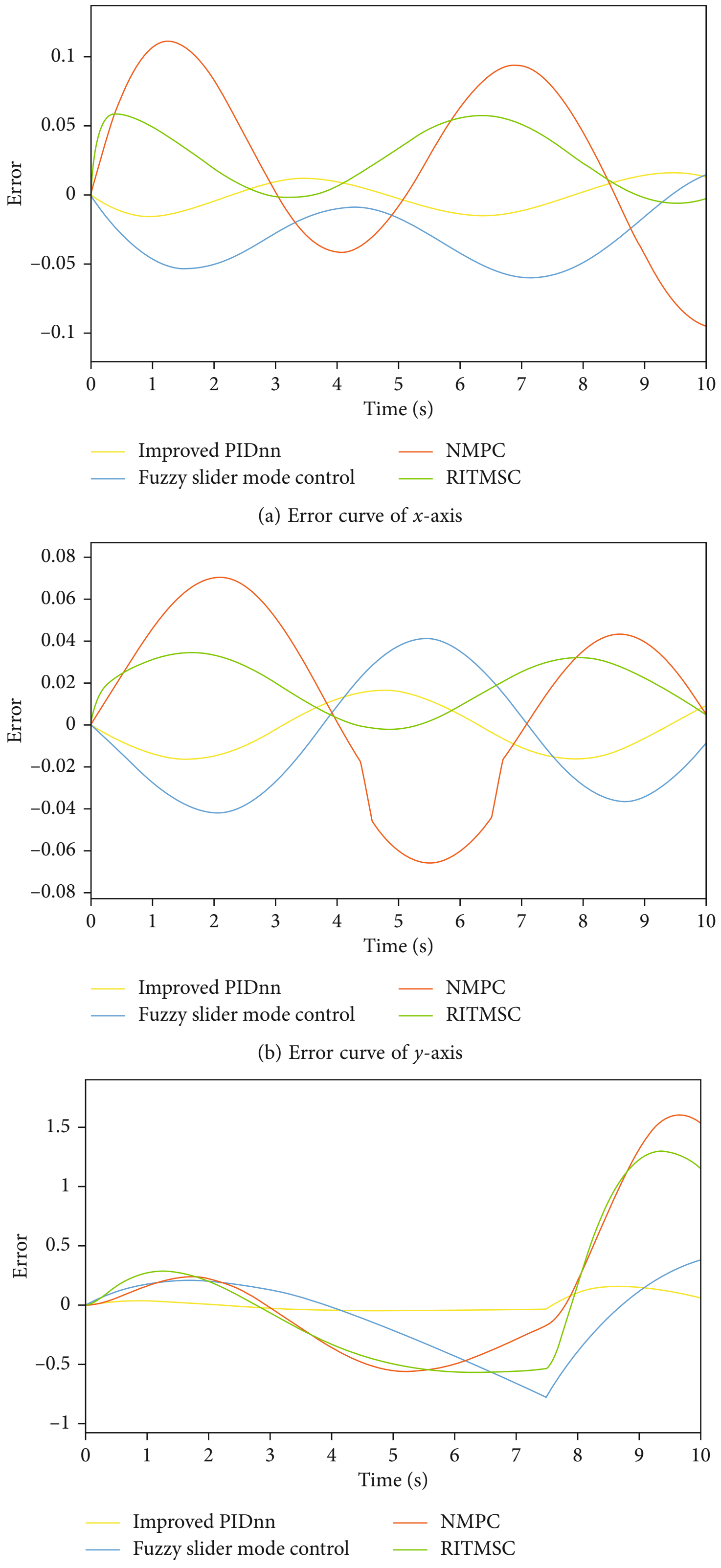


(a) Error curve of x-axis

(b) Error curve of y-axis

(c) Error curve in yaw-axis

FIGURE 13: Motion trajectory comparison of advanced algorithms (square).



(a) Error curve of x-axis

(b) Error curve of y-axis

(c) Error curve in yaw-axis

Figure 14: Motion trajectory comparison of advanced algorithms (circle).

5.2. Simulation. To verify the effectiveness of the control algorithm designed for the 3-DOF air flotation platform, MATLAB/Simulink is used for simulation in this paper, and the overall effect is shown in Figure 7. The predicted position curve of the trajectory was input into the system, and then, the algorithm was controlled by the cascade double closed-loop PIDnn system. The obtained control results are introduced into the dynamics model and then returned to the controller for further calculation.

Considering the various movement and work of space robot, there will be a variety of movement curves such as straight line, curve, or sharp angle. So, we designed two kinds of curves, one is the motion curve of the square, and the other is the motion curve of the circle.

As shown in Figures 8 and 9, for each curve, we give the system's 3-DOF input.

After effective control by the controller, the system simulation curve is obtained as shown in Figure 10.

In the simulation experiment, as shown in Figures 11 and 12, we get the error tracking of the three degrees of freedom curve.

We can see from the error curve that the error effect of moving distance can reach the level of 10^{-3} on average in the simulation experiment. In yaw, the maximum error in motion is only 0.17°. It can be observed that using the improved PIDnn based on momentum and RMSprop, the position and attitude control of the air flotation platform has a good control effect. It can quickly and accurately track the input signal.

5.3. Comparison of Advanced Algorithms. In order to analyze the optimality of space robot control system, we compared other advanced control algorithms according to the above two trajectory models. We can see the error comparison curve in Figures 13 and 14. We compare it with three other advanced control algorithms in recent years. The blue line represents the fuzzy slider mode control algorithm [16], the red line represents the nonlinear model predictive control algorithm (NMPC) [17], and the green line represents the robot integral terminal sliding mode control law (RITSMC) [30].

The improvement and variation of the control effect can be obtained from the error curve. The average control error of position control can reach 7.59×10^{-3} unit distance, and the error control accuracy of yaw can reach $8.36 \times 10^{-2\circ}$. The improved PID neural network using momentum and RMSprop can be proved by simulation results. It can effectively control the high-precision movement of the space robot, so that the space robot can have smaller error with the preset curve and achieve accurate control.

6. Conclusion and Future Work

In this paper, according to the needs of the ground preexperiment, the control system is designed and constructed in combination with the 3-DOF air flotation platform. The design allows it to move freely on the marble table. The design of the overall dynamic model and control algorithm is introduced, and the model is simplified and deduced according to the actual experimental environment. The control algorithm adopts the PID neural network to realize the 3-DOF motion of the air flotation platform. And the double-loop cascade control method is adopted to achieve better control effect. Finally, the designed control system is simulated by MATLAB/Simulink. The overall control effect is simulated, and it is known from the simulation effect that the system can control the air flotation platform quickly and effectively. In the era of rapid development of science and technology, the development of space operation technology and robot-assisted technology is gradually improving the hard power of human space exploration. As a ground-assisted experimental configuration, the air flotation platform plays an important role, and the improvement and research of experimental equipment will be further improved and further advance the development and progress of automation.

In the future work, we mainly improve the following contents: (a) optimize the overall control structure and specific hardware equipment of the microgravity air flotation platform and (b) optimize the computing power and sensor accuracy of the single-chip microcomputer, so that it can perform more complex control algorithms and obtain data with higher precision.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no competing interests regarding the publication of this paper.

Authors' Contributions

Siyue Wang and Qing Gao contributed equally to this work. Qing Gao is the co-first author.

Acknowledgments

This work was supported in part by the National Key R & D Program of China (Grant No. 2018YFB1304600), the Natural Science Foundation of China (Grant Nos. 51775541 and 62006204), and CAS Interdisciplinary Innovation Team (Grant No. JCTD-2018-11).

References

[1] M. Shan, J. Guo, and E. Gill, "Review and comparison of active space debris capturing and removal methods," *Progress in Aerospace Sciences*, vol. 80, pp. 18–32, 2016.

[2] X. Ge, Q. Zhou, and Z. Liu, "Assessment of space station on-orbit maintenance task complexity," *Reliability Engineering & System Safety*, vol. 193, article 106661, 2020.

[3] K. Yoshida, "Engineering test satellite VII flight experiments for space robot dynamics and control: theories on laboratory test beds ten years ago, now in orbit," *The International Journal of Robotics Research*, vol. 22, no. 5, pp. 321–335, 2003.

[4] R. Herranz, R. Anken, J. Boonstra et al., "Ground-based facilities for simulation of microgravity: organism-specific recommendations for their use, and recommended terminology," *Astrobiology*, vol. 13, no. 1, pp. 1–17, 2013.

[5] T. Könemann, U. Kaczmarczik, A. Gierse et al., "Concept for a next-generation drop tower system," *Advances in Space Research*, vol. 55, no. 6, pp. 1728–1733, 2015.

[6] J. Liu, Y. Li, W. Li, H. Zhai, and L. Chen, "Testing liquid distribution in a vane-type propellant tank under conditions of microgravity using a drop tower test," *International Journal of Aerospace Engineering*, vol. 2020, Article ID 6402083, 13 pages, 2020.

[7] C. Nieto-Peroy, G. Palmerini, É. J. de Oliveira, P. Gasbarri, M. Sabatini, and M. Milz, "Simulation of spacecraft formation maneuvers by means of floating platforms," in *2021 IEEE Aerospace Conference (50100)*, pp. 1–10, Big Sky, MT, USA, 2021.

[8] L. W. Deng, Z. M. Liu, C. C. Wang, B. W. Chen, and S. M. Song, "The design for position and attitude control system of 5-DOF air-floated platform based on ADRC," in *2011 2nd International Conference on Intelligent Control and Information Processing*, pp. 1061–1065, Harbin, China, 2011.

[9] T. Rybus and K. Seweryn, "Planar air-bearing microgravity simulators: review of applications, existing solutions and design parameters," *Acta Astronautica*, vol. 120, pp. 239–259, 2016.

[10] J. Prado, G. Bisiacchi, L. Reyes et al., "Three-axis air-bearing based platform for small satellite attitude determination and control simulation," *Journal of Applied Research and Technology*, vol. 3, no. 3, pp. 222–237, 2005.

[11] F. Terui, "Position and attitude control of a spacecraft by sliding mode control," in *Proceedings of the American Control Conference*, pp. 217–221, Phildelphia, Pa, USA, 1998.

[12] D. T. Stansbery and J. R. Cloutier, "Position and attitude control of a spacecraft using the state-dependent Riccati equation technique," in *Proceedings of the American Control Conference*, pp. 1867–1871, Chicago, Ill, USA, 2000.

[13] Y. G. H. A. D. Z. Cheng, *Nonlinear System Analysis and Control*, Science Press, Beijing, China, 1st edition, 2005.

[14] G. Q. Wu, S. N. Wu, and Z. G. Wu, "Robust finite-time control for spacecraft with coupled translation and attitude dynamics," *Mathematical Problems in Engineering*, vol. 2013, Article ID 707485, 7 pages, 2013.

[15] J. Liu, Q. Gao, Z. Liu, and Y. Li, "Attitude control for astronaut assisted robot in the space station," *International Journal of Control, Automation and Systems*, vol. 14, no. 4, pp. 1082–1095, 2016.

[16] Q. Gao, J. Liu, T. Tian, and Y. Li, "Free-flying dynamics and control of an astronaut assistant robot based on fuzzy sliding mode algorithm," *Acta Astronautica*, vol. 138, pp. 462–474, 2017.

[17] B. P. Malladi, S. Di Cairano, and A. Weiss, "Nonlinear model predictive control of coupled rotational-translational spacecraft relative motion," in *2019 American Control Conference (ACC)*, pp. 3581–3586, Philadelphia, PA, USA, 2019.

[18] Y. K. Nakka, R. C. Foust, E. S. Lupu et al., "A six degree-of-freedom spacecraft dynamics simulator for formation control research," in *2018 AAS/AIAA Astrodynamics Specialist Conference*, Snowbird, UT, USA, 2018.

[19] J. Zhao, S. Yan, and J. Wu, "Analysis of parameter sensitivity of space manipulator with harmonic drive based on the revised response surface method," *Acta Astronautica*, vol. 98, pp. 86–96, 2014.

[20] J. L. Zhao, S. Z. Yan, J. N. Wu, W. Ma, and Z. Y. Han, "Thermodynamic analysis of a space station remote manipulator with a harmonic drive that considers an integrated thermal protection layer," *SCIENCE CHINA Technological Sciences*, vol. 58, no. 11, pp. 1884–1893, 2015.

[21] H. Liu, L. Guo, and Y. Zhang, "An anti-disturbance PD control scheme for attitude control and stabilization of flexible spacecrafts," *Nonlinear Dynamics*, vol. 67, no. 3, pp. 2081–2088, 2012.

[22] C. Nie, Z. Zheng, and M. Zhu, "Three-dimensional path-following control of a robotic airship with reinforcement learning," *International Journal of Aerospace Engineering*, vol. 2019, Article ID 7854173, 12 pages, 2019.

[23] P. C. Hughes, *Spacecraft attitude dynamics*, Courier Corporation, 2012.

[24] T. Yang, S. Yan, W. Ma, and Z. Han, "Joint dynamic analysis of space manipulator with planetary gear train transmission," *Robotica*, vol. 34, no. 5, pp. 1042–1058, 2016.

[25] Z. B. Zhang, X. H. Li, J. P. An, W. X. Man, and G. H. Zhang, "Model-free attitude control of spacecraft based on PID-guide TD3 algorithm," *International Journal of Aerospace Engineering*, vol. 2020, Article ID 8874619, 13 pages, 2020.

[26] H. Shu, X. Guo, and H. Shu, "PID neural networks in multivariable systems," in *Proceedings of the IEEE Internatinal Symposium on Intelligent Control*, pp. 440–444, Vancouver, BC, Canada, 2002.

[27] H. Yin, W. Yi, J. Wu, K. Wang, and J. Guan, "Adaptive fuzzy neural network PID algorithm for BLDCM speed control system," *Mathematics*, vol. 10, no. 1, p. 118, 2022.

[28] R. Ma, S. Li, B. Zhang, and Z. Li, "Towards fast and robust real image denoising with attentive neural network and PID controller," *IEEE Transactions on Multimedia*, vol. 24, pp. 2366–2377, 2022.

[29] J. Kang, W. Meng, A. Abraham, and H. Liu, "An adaptive PID neural network for complex nonlinear system control," *Neurocomputing*, vol. 135, pp. 79–85, 2014.

[30] M. Labbadi and M. Cherkaoui, "Robust integral terminal sliding mode control for quadrotor UAV with external disturbances," *International Journal of Aerospace Engineering*, vol. 2019, Article ID 2016416, 10 pages, 2019.