mathematics

MDPI

Article

Fixed-Point Control of Airships Based on a Characteristic Model: A Data-Driven Approach

Yanlin Chen †, Shaoping Shen *,†, Zikun Hu and Long Huang

School of Aerospace Engineering, Xiamen University, Xiamen 361104, China
* Correspondence: shen_shaoping@xmu.edu.cn
† These authors contributed equally to this work.

Abstract: Factors such as changes in the external atmospheric environment, volatility in the external radiation, convective heat transfer, and radiation between the internal surfaces of the airship skin will cause a series of changes in the motion model of an airship. The adaptive control method of the characteristic model is proposed to extract the relationship between input and output in the original system, without relying on an accurate dynamic model, and solves the problem of inaccurate modeling. This paper analyzes the variables needed for two-dimensional path tracking and combines the guidance theory and the method of wind field state conversion to determine specific control targets. Through the research results, under the interference of wind, the PD control method and the reinforcement learning-based method are compared with a characteristic model control method. The response speed of the characteristic model control method surpasses the PD control method, and it reaches a steady state earlier than the PD control method does. The overshoot of the characteristic model control method is smaller than that of the PD control method. Using the control method of the characteristic model, the process of an airship flying to a target point will be more stable under the influence of an external environment. The modeling of the characteristic model adaptive control method does not rely on a precise model of the system, and it automatically adjusts when the parameters change to maintain a consistent performance in the system, thus reflecting the robustness and adaptability of the characteristic model adaptive control method in contrast with reinforcement learning.

Keywords: characteristic model; high-altitude airship; fixed-point control; data-driven; nonlinear system

MSC: 93C40

check for updates

Citation: Chen, Y.; Shen, S.; Hu, Z.; Huang, L. Fixed-Point Control of Airships Based on a Characteristic Model: A Data-Driven Approach. *Mathematics* **2023**, *11*, 310. https://doi.org/10.3390/math11020310

Academic Editors: António Lopes and Asier Ibeas

Received: 23 November 2022
Revised: 31 December 2022
Accepted: 4 January 2023
Published: 6 January 2023

Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

The high-altitude airship is a kind of unmanned aerial vehicle lighter than air. It uses electric drive propellers or new ion propulsion. It can hover or fly at low speed for a long time at a fixed point in the high-altitude range from 20 to 30 km. The high altitude airship has the characteristic of hovering at a fixed point for a long time, which makes it play an irreplaceable role in the mission [1]. Moreover, the airship has the characteristics of a low production cost, a long dwell time, and a large load capacity. In the military, the stratospheric airship can be used for military reconnaissance, gathering important intelligence, and monitoring secret targets. In terms of civilian use, it can carry out observations on the ground and measurements in land meteorology. It can also be used as an ideal platform for high-altitude communication, with the advantage of a wide coverage area for transmitted and received signals [2,3]. When studying the motion control of the airship, the change in airship position caused by atmospheric circulation is generally considered, while the changes in parameter changes brought about by the external environment of the airship are ignored [4]. The external atmospheric environment,

external radiation, convective heat transfer, and radiant heat transfer inside the airship skin can cause changes in the skin's temperature [5]. In order to achieve the safe flight of the airship, the difference between the gas pressure in the airbag and the outside atmospheric pressure is maintained within a certain range [6]. However, the adaptive control method based on the characteristic model extracts the main relationship between input and output on the original dynamic model without relying on the precise dynamic model [7]. This design of the controller is more in line with the actual engineering. Liang Dong et al. used the Lyapunov stability theorem to design the fixed point mode of a stratospheric airship, but this method required an accurate dynamic model and did not consider the influence of the wind field on the airship [6]. Wang Xiaoliang et al. achieved the fixed point control of an airship based on sliding mode variable structure control, but only considered the influence of the horizontal wind field [5]. Gao Wei et al. used the neural network dynamic inverse control method to design the fixed-point control law of an airship's horizontal plane and only considered the one-dimensional motion model on the horizontal plane [3].

The uncertainty of airship parameters in control has been studied extensively in the past few years. Considering the special structure of some nonlinear objects, such as the upper triangle and the lower triangle, relevant research has solved the adaptive control of these systems [8,9]. In order to facilitate online execution, decentralized adaptive control of these special structures has been investigated [10]. Practical factors (such as structural damage and unmeasurable states) that affect robustness have also been considered [11–13]. Reinforcement learning (RL) algorithms are being increasingly applied to control airships [14,15]. Obviously, all of the above relevant adaptive methods involve prior system information.

In this paper, a six-degree-of-freedom dynamic model of an airship is established based on the Newton–Euler equation. By analyzing the variables required by the airship model in two-dimensional path tracking, the dynamic model of an airship is simplified into a three-degree-of-freedom motion model, and specific control targets are given by integrating guidance theory with the wind field state conversion method. Based on characteristic modeling of the kinematics model, combined with the golden section adaptive control law and the logical differential control law, this work achieves the fixed-point control of an airship. For comparison purposes, a PD controller is often selected as an object of comparison [16–21]. In the mathematical simulation, the control result of the controller is compared with the control result of a traditional PD controller, which shows the effectiveness and superiority of the controller designed in this study.

With the improvement of hardware computing power and the emergence of various numerical calculation methods, data-driven approaches have emerged in a variety of fields [22–24]. Their advantage lies in the ability to interact with the environment online, which reduces the involvement of prior information [25]. In a control field, the datasets will be part of the controller and participate in the control in an adaptive manner [26]. Offline methods compute a control policy in the form of a parametric form or a table lookup prior to applying it to the process. Although the online computing needed might be minimal, robustness against uncertainty decreases. From the perspective of control theory, ML or RL achieves stability and a transient performance of the system by determining the dynamic programming problem, which shows a fitting process that identifies the HJB (Hamilton–Jacobi–Bellman) equation with a posteriori approach [27]. Making the most feasible choice in accordance with certain desired criteria is never easy. In airship control, relevant research can be found in [28–31]. Such solutions can only be made when the system model is unknown or there are uncertainties in the system dynamics, which make them even more difficult and sometimes impossible [32]. Nevertheless, owing to the difficulty of the computational problem (even if existence and other theoretical questions are settled), this is not always possible. In addition, the practical control problem is not usually posed as a well-formulated mathematical optimization problem. In this work, we will provide a data-driven approach without pursuing an optimal index by identifying the characteristic model online, which is instrumental in practice. Meanwhile, we provide the results of comparison with the RL algorithm, which can be found in [33,34].

The organization of this paper is as follows. In Section 2, we provide a description of the dynamic system. In Section 3, some control loops are provided, and corresponding characteristic models are divided. The main results of this paper are then presented. Based on the model given in Section 3, the fixed-point control method is presented in Sections 4 and 5. Some necessary proofs are provided. In Section 6, we compare some experiments by numerical simulation. For a comparison with existing data-driven methods, in Section 7, we provide a policy iteration algorithm via reinforcement learning, and a comprehensive comparative analysis is given. Finally, we provide some concluding remarks in Section 8.

2. The Dynamic Model of a Stratospheric Airship

The structure of the airship is shown in Figure 1. The thrust system of the airship is composed of two propellers at the rear of the airship. The direction of the airship is controlled by changing the deflection of the rudder of the airship. In this paper, the ground coordinate system and airship coordinate system are established. In the ground coordinate system, the origin of coordinate o_e is selected as any point on the ground. The axis $o_e x_e$ points to the east direction, and the axis $o_e y_e$ points to the north direction. The origin of the airship coordinate system o_b is set at the center of the airship's body. The coordinate axis $o_b x_b$ is arranged in the plane of symmetry, parallel to the axis of the body of the airship and pointing to the head of the airship. The coordinate axis $o_b z_b$ is set in the plane of symmetry perpendicular to the coordinate axis $o_b x_b$, pointing downwards. $o_b y_b$ is set perpendicular to the plane $x_b o_b y_b$ and points to the right. This article studies the fixed-point control of the stratosphere airship in the horizontal plane. The roll motion and the motion in the vertical plane need not be considered. Therefore, there is no need to consider the pitch angle θ and the roll angle Φ only the change in the yaw angle ψ. T is the thrust vector. The thrust components in the directions of $o_b x_b$, $o_b y_b$ can be expressed as $T_x = T\cos\mu$, $T_y = T\sin\mu$, where μ is used as the angle between the thrust vector and the plane $x_b o_b y_b$ The six-degree-of-freedom motion model of the airship can be simplified into a three-degree-of-freedom dynamic equation. The three-degree-of-freedom dynamic equation can be expressed as follows:

$$\begin{cases} \dot{u}_b = \frac{m+m_{22}}{m+m_{11}} v_b r + \frac{QV_f^{\frac{2}{3}}}{m+m_{11}}(-C_x\cos\beta + C_y\sin\beta) + \frac{T_x}{m+m_{11}} \\ \dot{v}_b = \frac{m+m_{11}}{m+m_{22}} u_b r + \frac{QV_f^{\frac{2}{3}}}{m+m_{22}}(C_x\sin\beta + C_y\cos\beta) + \frac{T_y}{m+m_{22}} \\ \dot{r} = \frac{1}{I_z+m_{66}} QV_f C_n + \frac{l_x}{I_z+m_{66}} T_y \end{cases} \tag{1}$$

The simplified kinematic equation of the airship in the horizontal plane can be expressed as

$$\begin{cases} \dot{x}_e = u_b\cos\psi - v_b\sin\psi + V_\omega\cos\psi_\omega \\ \dot{y}_e = u_b\sin\psi + v_b\cos\psi + V_\omega\sin\psi_\omega \\ \dot{\psi} = r \end{cases} \tag{2}$$

where m is the central mass of the airship, $m_{ii}(i = 1, 2, 3, \cdots, 6)$ is additional mass; V_f is the volume of the airship; Q is the dynamic pressure, $Q = \frac{1}{2}\rho_a V_f^2$, where ρ_a is the atmospheric density; u_b, v_b, r are the forward, lateral, yaw speed, respectively; C_x, C_y, and C_n are the drag coefficient, side force coefficient, and yaw moment coefficient, respectively; l_z is the distance from the thrust point p_0 to the coordinate axis $o_b x_b$; β is expressed as the side slip angle. I_z is expressed as the moment of inertia about the coordinate axis $o_b z_b$; x_e and y_e are, respectively, expressed as the axial displacement and lateral displacement of the airship.

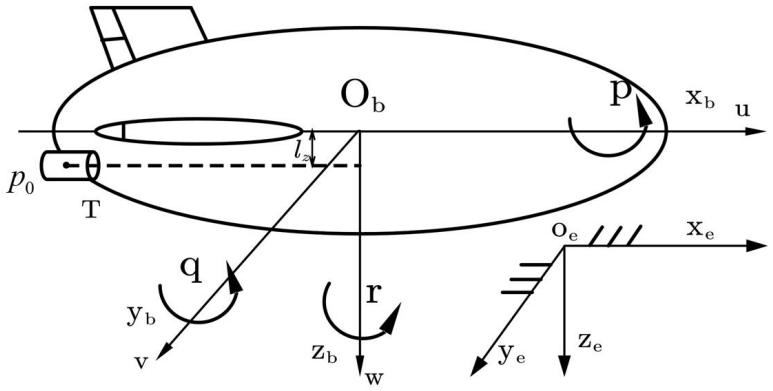


Figure 1. The coordinate diagram of the airship.

Remark 1. *For the idea of the characteristic model, it is assumed that the dynamic characteristics of the controlled system are known. Based on these characteristics, one can establish a mathematical model that is different from ordinary differential equations, and then use appropriate parameter estimation methods to obtain the coefficients of the characteristic model equation, i.e., determine quantitatively the system model of the characteristic equation in a data-driven way. Its goal is to facilitate the control objectives and performance to achieve the controller design. We cannot acquire precise system matrices, so a characteristic model is a good method for describing the system evolution, which is different from the dynamics model identified by the first principle.*

3. The Target of the Control

The speed of the airship relative to the ground is equal to the speed of the airship relative to the air plus the wind speed, so the plane position kinematics equation in the stratospheric airfield can be expressed as

$$\dot{\zeta}_p = \begin{bmatrix} \cos\psi & -\sin\psi \\ \sin\psi & \cos\psi \end{bmatrix} \begin{bmatrix} u_b \\ v_b \end{bmatrix} + \begin{bmatrix} V_\omega \cos\psi_\omega \\ V_\omega \sin\psi_\omega \end{bmatrix} \tag{3}$$

where ζ_p represents the position of the airship, V_ω represents the wind speed, and $V_\omega > 0$; ψ_ω represents the wind direction. The position of the airship in the wind field can be represented as

$$\begin{cases} x_b = x_e - \int_0^t V_\omega \cos\psi_\omega dt \\ y_b = y_e - \int_0^t V_\omega \sin\psi_\omega dt \end{cases} \tag{4}$$

Thus, Equation (3) can be expressed as

$$\dot{\zeta}_a = \begin{bmatrix} \dot{x}_b \\ \dot{y}_b \end{bmatrix} = \begin{bmatrix} \cos\psi & -\sin\psi \\ \sin\psi & \cos\psi \end{bmatrix} \begin{bmatrix} u_b \\ v_b \end{bmatrix} \tag{5}$$

where ζ_a represents the position error of the airship. In this paper, the straight path that passes through the desired point $\zeta_d = [x_d, y_d]^T$ and has a slope of ψ_ω which can be given as

$$\zeta_c(\varpi) = \begin{bmatrix} x_c(\varpi) \\ y_c(\varpi) \end{bmatrix} = \begin{bmatrix} -\varpi\cos\psi_\omega + x_d \\ -\varpi\sin\psi_\omega + y_d \end{bmatrix} \tag{6}$$

where ϖ is the path parameter. Based on the definition of ζ_c, we can see that, if ζ_a tracks ζ_c, when $\dot{\varpi} > 0$, the airship heading angle is directly opposite to the wind direction. At the same time, if the control airship's forward airspeed u_a is equal to the wind speed, fixed-point hovering can be achieved.

3.1. The Design of the Guidance Loop

In this paper, guidance theory is applied to solve the fixed point problem of an airship on a two-dimensional plane. We define the path coordinate system with the point $\zeta_c(\varpi)$ on

the desired path as the origin and with the axis x_c along the tangential direction of $\zeta_c(\varpi)$. y_c is perpendicular to x_c and points to the right. The rotation angle from the airship's body coordinate system to the path coordinate system can be expressed as

$$\varphi_p = \arctan(\dot{y}_c/\dot{x}_c) = \psi_\pi - \pi \tag{7}$$

Differentiating Equation (6) and combining Equation (7), we have

$$\dot{\zeta}_c = \begin{bmatrix} \cos\psi_p & -\sin\psi_p \\ \sin\psi_p & \cos\psi_p \end{bmatrix} \begin{bmatrix} \dot{\varpi} \\ 0 \end{bmatrix} = R(\psi_p) \begin{bmatrix} \dot{\varpi} \\ 0 \end{bmatrix} \tag{8}$$

In the path coordinate system, the position error between ζ_a and ζ_c can be expressed as

$$\varepsilon = \begin{bmatrix} \cos\psi_p & -\sin\psi_p \\ \sin\psi_p & \cos\psi_p \end{bmatrix} \begin{bmatrix} x_b - x_d \\ y_b - y_d \end{bmatrix} = R^T(\psi_p)(\zeta_a - \zeta_c(\varpi)) \tag{9}$$

Therefore, the target of the guidance loop is converted to the design expected yaw angle such that $\varepsilon \to 0$. The definition of the Lyapunov function can be expressed as

$$V_\varepsilon = \frac{1}{2}\varepsilon^T\varepsilon \tag{10}$$

We assume that, when $u_a = V_\omega$, $\psi = \psi_c$, ψ_c is a given value. the stratosphere airship reaches the desired path $\zeta_c(\varpi)$. Differentiating V_ε with Equations (5) and (8), we have

$$\dot{V}_\varepsilon = \varepsilon^T(\dot{R}^T(\psi_p)(\zeta_a - \zeta_c) + R^T(\psi_p)(\dot{\zeta}_a - \dot{\zeta}_c)) \tag{11}$$

$\dot{\varpi} = V_\omega\cos(\psi_c - \psi_p) - v_a\sin(\psi_c - \psi_p) + k_s s$, $\psi_c = \psi_p + \arctan(\frac{-e}{k_e}) - \arctan(\frac{v_b}{V_\omega})$, where $\{k_s, k_e\} > 0$ are the control parameters, which can be substituted into Equation (11) to obtain

$$\dot{V}_\varepsilon = -k_s s^2 - \frac{\sqrt{V_\omega^2 + v_a^2}}{\sqrt{e_2 + k_e^2}}e^2 \le 0 \tag{12}$$

where s is the forward tracking error, and e is the lateral tracking error. Therefore, the path following error ε is convergent.

3.2. The Design of Expectation Speed

To achieve the control goal, the airship has the same speed as the wind speed and flies along the path. The defined expected forward airspeed can be expressed as

$$u_{bc} = V_\omega - k_p(\cos\psi_p(x_b - x_d) + \sin\psi_p(y_b - y_d)) \tag{13}$$

where $k_p > 0$ is the controller parameter. The desired airspeed value can be adjusted by the position error. If the airship is in front of the desired position, the desired airspeed is reduced; otherwise, the airship is accelerated. Therefore, the desired forward speed can be expressed as

$$u_c = u_{bc} - V_\omega = -k_p s_p \tag{14}$$

where s_p is the forward tracking error between ζ_p and ζ_d.

4. Fixed-Point Control of an Airship Based on a Multiple-Input Multiple-Output Characteristic Model

The stratospheric airship is a nonlinear coupling system with multiple inputs and outputs. The analyses for a precise dynamic model is not a trivial task, even if a Lyapunov equation can be found by trial and error. In this section, We apply a characteristic model approach which is built on a discrete state space. Adaptive learning method will be employed in the following policy design, which includes feedforward compensation and

feedback stabilization in the method of characteristic model. Specifically, golden section adaptive controller is used to learn airship parameters online, which is subject to prescribed coefficients $L_1 = 0.382$ and $L_2 = 0.618$. It is worth mentioning that minimum variance combined with L_1, L_2 can guarantee system stability at given equilibrium state under some mild limits on the range of parameters learnt. For avoiding redundancies, we recommend the literature [4,35] for the details of the characteristic model. In addition, we ignore the feedforward compensation part of the controller design. Firstly, a four channel coupling model of forward velocity, lateral velocity, yaw velocity, and yaw angle are established. We then parameterize these known models by the characteristic model. In practice, one can adopt it directly.

4.1. The Characteristic Model Derivation of the Forward Velocity Channel

According to Equation (1), where $Q = \frac{1}{2}\rho_a(u_b^2 + v_b^2)$ is dynamic pressure, $\beta = \arctan(\frac{v_b}{u_b})$ in Equation (1) is the slip angle, we then obtain a more compact expression:

$$\begin{aligned}\dot{u}_b &= \frac{1}{2(m+m_{11})}\rho_a \cdot V_f^{\frac{2}{3}}(u_b^2+v_b^2)[-C_x \cdot \cos(\arctan\frac{v_b}{u_b}) \\ &+ (C_{y0} + C_{ybeta} \cdot \arctan\frac{v_b}{u_b} + C_{yr} \cdot r)\sin(\arctan\frac{v_b}{u_b})] \\ &+ \frac{1}{m+m_{11}}T_x + \frac{m+m_{22}}{m+m_{11}}v_b r \\ &= f_1(u_b, v_b, r) + \frac{1}{m+m_{11}}T_x\end{aligned} \tag{15}$$

where C_{y0}, C_{ybeta}, C_{yr} are the different drag model parameters. One can then obtain the derivative of Equation (15), which will be used for the parameterization design of the characteristic model:

$$\ddot{u}_b = f'_{1,u_b} \cdot \dot{u}_b + f'_{1,v_b} \cdot \dot{v}_b + f'_{1,r} \cdot \dot{r} + \frac{1}{m+m_{11}}\dot{T}_x \tag{16}$$

The expression of f'_{1,u_b} in Equation (16) can be expressed as

$$\begin{aligned}f'_{1,u_b} &= \frac{1}{m+m_{11}}\rho_a \cdot V_f^{\frac{2}{3}} \cdot u_b[-C_x\cos\beta + C_y\sin\beta] \\ &+ \frac{1}{2(m+m_{11})}\rho_a \cdot V_f^{\frac{2}{3}} \\ &\times [-C_x\sin\beta \cdot v_b - C_{ybeta}\sin\beta \cdot v_b - C_y\cos\beta \cdot v_b] \\ &- C_{ybeta}\sin\beta \cdot v_b - C_y\cos\beta \cdot v_b]\end{aligned} \tag{17}$$

In the same way, the expression of f'_{1,v_b} in Equation (16) can be expressed as

$$\begin{aligned}f'_{1,v_b} &= \frac{1}{m+m_{11}}\rho_a \cdot V_f^{\frac{2}{3}} \cdot v_b[-C_x\cos\beta + C_y\sin\beta] \\ &+ \frac{1}{2(m+m_{11})}\rho_a \cdot V_f^{\frac{2}{3}} \\ &\times [-C_x\sin\beta \cdot u_b - C_{ybeta}\sin\beta \cdot u_b - C_y\cos\beta \cdot u_b] \\ &+ \frac{m+m_{22}}{m+m_{11}}r\end{aligned} \tag{18}$$

As f'_{1,u_b} and f'_{1,v_b} above, $f'_{1,r}$ in Equation (16) can be acquired:

$$f'_{1,r} = \frac{m+m_{22}}{m+m_{11}}v_b + \frac{\rho_a \cdot V_f^{\frac{2}{3}}}{2(m+m_{11})}(u_b^2+v_b^2) \cdot C_{yr}\sin\beta \tag{19}$$

Equation (15) can then be simplified to

$$\ddot{u}_b = {f'}_{1,u_b} \cdot \dot{u}_b + \frac{1}{m + m_{11}} \dot{T}_x + \Delta_u \tag{20}$$

where $\Delta_u = {f'}_{1,v_b} \cdot \dot{v}_b + {f'}_{1,r} \cdot \dot{r}$. Equation (20) after discretization can be expressed as

$$\begin{aligned} \frac{u_b(k+1) - 2u_b(k) + u_b(k-1)}{h^2} &= {f'}_{1,u_b} \frac{u_b(k) - u_b(k-1)}{h} \\ &+ \frac{1}{m + m_{11}} \frac{T_x(k) - T_x(k-1)}{h} \\ &+ \Delta_u(k) \end{aligned} \tag{21}$$

Equation (21) is written in the form of a characteristic model [36]:

$$\begin{aligned} u_b(k+1) = f_{11} \cdot u_b(k) + f_{12} \cdot u_b(k-1) \\ + g_{11}(k) \cdot T_x(x) + g_{12}(k) \cdot T_x(k-1) + \Delta_u(k) \end{aligned} \tag{22}$$

For normalization, Equation (22) is written as a parameter estimation equation:

$$\begin{aligned} u_b(k+1) = \bar{f}_{11}(k) \cdot u_b(k) + \bar{f}_{12}(k) u_b(k-1) + \bar{g}_{11}(k) \cdot T_x(x) \\ + \bar{g}_{12}(k) \cdot T_x(k-1) \end{aligned} \tag{23}$$

Here,

$$\begin{cases} \bar{f}_{11}(k) = 2 + h \cdot {f'}_{1,u_b} + \frac{h^2 \Delta_u(k) u_b(k)}{u_b^2(k) + u_b^2(k-1) + T_x(k)^2 + T_x(k-1)^2} \\ \bar{f}_{12}(k) = -1 - h \cdot {f'}_{1,u_b} + \frac{h^2 \Delta_u(k) u_b(k-1)}{u_b^2(k) + u_b^2(k-1) + T_x(k)^2 + T_x(k-1)^2} \\ \bar{g}_{11}(k) = h \cdot \frac{1}{m+m_{11}} + \frac{h^2 \Delta_u(k) T_x(k)}{u_b^2(k) + u_b^2(k-1) + T_x(k)^2 + T_x(k-1)^2} \\ \bar{g}_{12}(k) = -h \cdot \frac{1}{m+m_{11}} + \frac{h^2 \Delta_u(k) T_x(k-1)}{u_b^2(k) + u_b^2(k-1) + T_x(k)^2 + T_x(k-1)^2} \end{cases} \tag{24}$$

where h is the sampling period.

4.2. The Characteristic Model Derivation of the Lateral Velocity Channel

By the same token, according to Equation (1), the lateral velocity of the airship can be derived as

$$\dot{v}_b = f_2(u_b, v_b, r) + \frac{1}{m + m_{22}} T_y \tag{25}$$

By taking the derivative of Equation (25), we have

$$\ddot{v}_b = {f'}_{2,v_b} \cdot \dot{v}_b + {f'}_{2,u_b} \cdot \dot{u}_b + {f'}_{2,r} \cdot \dot{r} + \frac{1}{m + m_{22}} \dot{T}_y \tag{26}$$

Similarly, the expression of ${f'}_{2,\dot{v}_b}$ can be expressed as

$$\begin{aligned} f'_{2,v_b} = \frac{1}{m + m_{22}} \rho_a \cdot v_f^{\frac{2}{3}} \cdot v_b (C_x \sin\beta + C_y \cos\beta) \\ + \frac{\rho_a v_f^{\frac{2}{3}} \cdot u_b}{2(m + m_{22})} (C_x \cos\beta - C_y \sin\beta + C_{ybeta} \cos\beta) \end{aligned} \tag{27}$$

Further, the expression of f'_{2,u_b} can be given as

$$\begin{aligned} f'_{2,u_b} &= \frac{1}{m+m_{22}}\rho_a \cdot v_f^{\frac{2}{3}} \cdot u_b(C_x \sin\beta + C_y \cos\beta) \\ &+ \frac{\rho_a v_f^{\frac{2}{3}} \cdot u_b}{2(m+m_{22})}(-C_x \cos\beta + C_y \sin\beta - C_{ybeta} \cos\beta) \\ &- \frac{m+m_{11}}{m+m_{22}} \cdot r \end{aligned} \tag{28}$$

The expression of $f'_{2,r}$ can be expressed as

$$f'_{2,r} = \frac{\rho_a v_f^{\frac{2}{3}} \cdot u_b}{2(m+m_{22})})(u_b^2 + v_b^2)C_{yr} \cos\beta - \frac{m+m_{11}}{m+m_{22}} \cdot u_b \tag{29}$$

Equation (26) can be expressed as

$$\ddot{v}_b = f'_{2,v_b} \cdot \dot{v}_b + \frac{1}{m+m_{22}}\dot{T}_y + \Delta_v \tag{30}$$

where $\Delta_v = f'_{2,u_b} \cdot \dot{u}_b + f'_{2,r} \cdot \dot{r}$. The above formula after discretization, like Equation (20), can be expressed as

$$\begin{aligned} \frac{v_b(k+1) - 2v_b(k) + v_b(k-1)}{h^2} &= f'_{2,v_b} \cdot \frac{v_b(k) - v_b(k-1)}{h} \\ &+ \frac{1}{m+m_{22}} \frac{T_y(k) - T_y(k-1)}{h} \\ &+ \Delta_v(k) \end{aligned} \tag{31}$$

Similarly, Equation (31) is written in the form of a characteristic model [36]:

$$\begin{aligned} v_b(k+1) &= f_{21}(k) \cdot v_b(k) + f_{22}(k)v_b(k-1) \\ &+ g_{21}(k) \cdot T_y(k) + g_{22}(k) \cdot T_y(k-1) + \Delta_v(k) \end{aligned} \tag{32}$$

For normalization, Equation (32) is written as a parameter estimation equation:

$$\begin{aligned} v_b(k+1) &= \bar{f}_{21}(k) \cdot v_b(k) + \bar{f}_{22}(k)v_b(k-1) \\ &+ g_{21}(k) \cdot T_y(k) + g_{22}(k) \cdot T_y(k-1) \end{aligned} \tag{33}$$

Here,

$$\begin{cases} \bar{f}_{21}(k) = 2 + h \cdot f'_{2,v_b} + \frac{h^2\Delta_v(k)v_b(k)}{v_b^2(k)+v_b^2(k-1)+T_y(k)^2+T_y(k-1)^2} \\ \bar{f}_{12}(k) = -1 - h \cdot f'_{2,v_b} + \frac{h^2\Delta_v(k)v_b(k-1)}{v_b^2(k)+v_b^2(k-1)+T_y(k)^2+T_y(k-1)^2} \\ \bar{g}_{21}(k) = h \cdot \frac{1}{m+m_{22}} + \frac{h^2\Delta_v(k)T_y(k)}{v_b^2(k)+v_b^2(k-1)+T_y(k)^2+T_y(k-1)^2} \\ \bar{g}_{22}(k) = -h \cdot \frac{1}{m+m_{22}} + \frac{h^2\Delta_v(k)T_y(k-1)}{v_b^2(k)+v_b^2(k-1)+T_y(k)^2+T_y(k-1)^2} \end{cases} \tag{34}$$

where h is the sampling period.

4.3. The Characteristic Model Derivation of the Yaw Angular Velocity Channel

Analogously to forward and lateral velocity channel parameterization process, according to Equation (1), one can represent the yaw angular velocity channel as follows:

$$\dot{r} = f_3(u_b, v_b, r) + \frac{l_x}{I_z + m_{66}}T_y \tag{35}$$

where I_z, l_x is the regularized mass about the corresponding axis. The derivative of Equation (35) can be obtained:

$$\ddot{r} = f'_{3,r} \cdot \dot{r} + f'_{3,u_b} \cdot \dot{u}_b + f'_{3,v_b} \cdot \dot{v}_b + \frac{l_x}{I_z + m_{66}} \dot{T}_y \tag{36}$$

Similarly, the expression of $f'_{3,r}$ can be expressed as

$$f'_{3,r} = \frac{1}{I_z + m_{66}} V_f \cdot Q \cdot C_{nr} \tag{37}$$

The expression of f'_{3,u_b} can be expressed as

$$f'_{3,u_b} = \frac{\rho_a \cdot V_f \cdot u_b}{I_z + m_{66}} - \frac{1}{2(I_z + m_{66})} \rho_a \cdot V_f \cdot C_{nbeta} \cdot v_b \tag{38}$$

The expression of f'_{3,v_b} can be expressed as

$$f'_{3,v_b} = \frac{\rho_a \cdot V_f \cdot v_b}{I_z + m_{66}} + \frac{1}{2(I_z + m_{66})} \rho_a \cdot V_f \cdot C_{nbeta} \cdot u_b \tag{39}$$

Finally, Equation (36) can be simplified to

$$\ddot{r} = f'_{3,r} \cdot \dot{r} + \frac{l_x}{I_z + m_{66}} \dot{T}_y + \Delta_r \tag{40}$$

where $\Delta_r = f'_{3,u_b} \cdot \dot{u}_b + f'_{3,v_b} \cdot \dot{v}_b$. Equation (40) after discretization can be expressed as

$$\begin{aligned} \frac{r(k+1) - 2r(k) + r(k-1)}{h^2} &= f'_{3,r} \cdot \frac{r(k) - r(k-1)}{h} \\ &+ \frac{l_x}{I_z + m_{66}} \frac{T_y(k) - T_y(k-1)}{h} \\ &+ \Delta_r(k) \end{aligned} \tag{41}$$

The above equation is written in the form of a characteristic model [36]:

$$\begin{aligned} r(k+1) = f_{31}(k) \cdot r(k) + f_{32}(k) r(k-1) \\ + g_{31}(k) \cdot T_y(k) + g_{32}(k) \cdot T_y(k-1) + \Delta_r(k) \end{aligned} \tag{42}$$

For normalization, Equation (42) is written as a parameter estimation equation:

$$\begin{aligned} r(k+1) = \bar{f}_{21}(k) \cdot r(k) + \bar{f}_{32}(k) r(k-1) \\ + g_{31}(k) \cdot T_y(k) + g_{32}(k) \cdot T_y(k-1) \end{aligned} \tag{43}$$

Here,

$$\begin{cases} \bar{f}_{31}(k) = 2 + h \cdot f'_{3,r} + \frac{h^2 \Delta_r(k) r(k)}{r^2(k) + r^2(k-1) + T_y(k)^2 + T_y(k-1)^2} \\ \bar{f}_{32}(k) = -1 - h \cdot f'_{3,v_b} + \frac{h^2 \Delta_r(k) r(k-1)}{r^2(k) + r^2(k-1) + T_y(k)^2 + T_y(k-1)^2} \\ \bar{g}_{31}(k) = h \cdot \frac{l_x}{I_z + m_{66}} + \frac{h^2 \Delta_r(k) T_y(k)}{r^2(k) + r^2(k-1) + T_y(k)^2 + T_y(k-1)^2} \\ \bar{g}_{32}(k) = -h \cdot \frac{l_x}{I_z + m_{66}} + \frac{h^2 \Delta_r(k) T_y(k-1)}{r^2(k) + r^2(k-1) + T_y(k)^2 + T_y(k-1)^2} \end{cases} \tag{44}$$

where h is the sampling period.

4.4. The Characteristic Model Derivation of Yaw Angle Channel

According to Equations (1) and (2), we have the following:

$$\ddot{\psi} = f'_{41}\dot{\psi} + g'_{41}T_y + \Delta_\psi \tag{45}$$

Here,

$$\begin{cases} f'_{41} = 0 \\ g'_{41} = \frac{l_x}{I_z + m_{66}} \\ \Delta_\psi = \frac{l_x}{I_z + m_{66}} Q V_f C_n \end{cases} \tag{46}$$

Equation (45) can be written in the form of a characteristic model as shown above:

$$\begin{aligned} \psi(k+1) = f_{41}(k) \cdot \psi(k) + f_{42}(k)\psi(k-1) \\ + g_{41}(k) \cdot T_y(k) + \Delta_\psi(k) \end{aligned} \tag{47}$$

For normalization, the above equation can be written as a parameter estimation equation:

$$\psi(k+1) = \bar{f}_{41}(k) \cdot \psi(k) + \bar{f}_{42}(k)\psi(k-1) + \bar{g}_{41}(k) \cdot T_y(k) \tag{48}$$

Here,

$$\begin{cases} \bar{f}_{41}(k) = 2 + h \cdot f'_{41} + \frac{h^2 \Delta_\psi(k)\psi(k)}{\psi^2(k) + \psi^2(k-1) + T_y(k)^2} \\ \bar{f}_{42}(k) = -1 - h \cdot f'_{41} + \frac{h^2 \Delta_\psi(k)\psi(k)}{\psi^2(k) + \psi^2(k-1) + T_y(k)^2} \\ \bar{g}_{41}(k) = h^2 \cdot g'_{41} + \frac{h^2 \Delta_\psi(k)\psi(k)}{\psi^2(k) + \psi^2(k-1) + T_y(k)^2} \end{cases} \tag{49}$$

5. The Design of a Fixed-Point Control Law for an Airship

A golden section adaptive controller and a logic differential controller are proposed in [12]. In this section, we focus only on the stability issues of the airship via an adaptive method.

The golden section adaptive control law of the forward speed channel can be expressed as

$$\begin{aligned} u_1(k) = \frac{-1}{\hat{g}_{11}(k) + \lambda_1} [L_1 \hat{f}_{11}(k)(u_b(k) - u_c(k)) \\ + L_2 \hat{f}_{12}(k)(u_b(k-1) - u_c(k-1))] \end{aligned} \tag{50}$$

where $L_1 = 0.382$, $L_2 = 0.618$. λ_1 is a small positive number. $\hat{f}_{11}(k)$, $\hat{f}_{12}(k)$, and $\hat{g}_{11}(k)$, respectively, represent the identification values of the characteristic model parameters. The logic differential control law of the forward speed channel can be expressed as

$$ud_1(k) = k_{1,d}[u_b(k) - u_c(k-1)] \tag{51}$$

where $k_{1,d} = c_1 \sqrt{\sum_{i-1}^{N} |u_b(k-i)|}$, and c_1 is an adjustable parameter.

The golden section adaptive control law of the yaw angle channel can be expressed as

$$\begin{aligned} u_2(k) = \frac{-1}{\hat{g}_{41}(k) + \lambda_2} [L_1 \hat{f}_{41}(k)(\psi(k) - \psi_c(k)) \\ + L_2 \hat{f}_{42}(k)(\psi(k-1) - \psi_c(k-1))] \end{aligned} \tag{52}$$

where $L_1 = 0.382$,$L_2 = 0.618$. λ_2 is a small positive number. $\hat{f}_{41}(k)$, $\hat{f}_{42}(k)$, and $\hat{g}_{41}(k)$, respectively, represent the identification values of the characteristic model parameters. The logic differential control law of the forward speed channel can be expressed as

$$ud_2(k) = k_{2,d}[\psi(k) - \psi_c(k-1)] \tag{53}$$

$k_{2,d} = c_2\sqrt{\sum_{i-1}^{N}|\psi(k-i)|}$. c_2 is an adjustable parameter.

Remark 2. *In off-policy optimal control, to maintain system stabilization, PI (policy iteration) needs an initial controller that can stabilize the system [34,37]. Similarly, in the initial transition stage of adaptive control, when the parameter estimation has not yet converged to the "true value", the above methods cannot easily guarantee the stability of the closed-loop system theoretically. The golden section adaptive controller restricts f to the specified parameter range. The feedback control law can guarantee the stability of the closed-loop system regardless of whether the parameter estimation converges to the "true value". Next, we show that the stabilization of the controlled system can be guaranteed under some conditions.*

Online Parameter Learning

The difference of Equation (23), taken as an example, can be transformed into

$$u_b(k+1) = U^T(k)\theta(k+1) \tag{54}$$

where $U(k) = \begin{bmatrix} u_b(k) & u_b(k-1) & T_x(k) & T_x(k-1) \end{bmatrix}$, and input parameters $\theta(k) = \begin{bmatrix} \bar{f}_{11}(k) & \bar{f}_{12}(k-1) & \bar{g}_{11}(k) & \bar{g}_{12}(k-1) \end{bmatrix}$. The least square method can be used to estimate the parameters of the difference equation.

Remark 3. *The main characteristics of the system are compressed into several characteristic variables. We can obtain a characteristic model with unknown coefficients, and this depends on the basic structural characteristics of the system. The input and output variable information data are obtained through online or offline operations, and the coefficients of the characteristic model equation are obtained by using appropriate parameter estimation methods. The least-squares problem can be solved in real time after a sufficient number of data points are collected along a single state trajectory under the regular presence of an excitation requirement.*

$$\begin{cases} K(k+1) = \frac{P(k)\psi(k)}{\rho+\psi^T(k)P(k)\psi(k)} \\ \hat{\theta}(k+1) = \hat{\theta}(k) + K(k+1)[u_b(k+1) - \psi^T(k)\hat{\theta}(k)] \\ P(k+1) = \frac{1}{\rho}[I - K(k+1)\psi^T(k+1)]P(k) \end{cases} \tag{55}$$

where $P(k) = \alpha I$, I is the identity matrix, and α is a large positive number, i.e., $0.95 \leq \rho \leq 0.99$.

Remark 4. *In order to ensure that the datasets are sufficiently rich (i.e., informative) and linearly independent, there is a need to add probing noise (excitation signal) to the control input in RL. However, the persistent excitation condition will lead to a bias away from true value if output feedback is employed. In practice, it is necessary to collect a sufficient amount of data samples for ensuring the existence of a recursive least squares solution, but our method obviate extra input singal.*

6. Numerical Experiments

In this section, we compare our approach with a frequently chosen PID controller. Furthermore, trajectory tracking of the airship is converted to corresponding fixed point control. Numerical simulation will be performed on the Matlab platform to simulate the fixed-point hovering control of the stratosphere airship. Controller parameters are k_s, k_e, and k_p. In simulation, $k_s = 0.01$ (when disturbance exists), $k_e = 100$, and $k_p = 0.1$. We select the desired hover trajectory $\zeta_d = [sin(t), cos(t)]^T$, and the initial position is $\zeta_p = [0,0]^T$. We define the airship initial state as $u_{b0} = 1(m/s)$, $v_{b0} = 1(m/s)$, and $r_0 = 0(rad/s)$, respectively. Wind speed is $V_\omega = 5(m/s)$, and wind direction is $\psi_\omega = \frac{7\pi}{6}(rad)$. The reference trajectory is a set of triangular signals. The parameters and initial values of the airship are shown in Tables 1 and 2. In simulation results, the notion "ideal" represents ideal fixed-point (trajectory) and "Characteristic model" and "RL" denote corresponding controller tracking results.

Table 1. Airship model parameters.

Parameter	Value
m	329 kg
I_z	5264 kg·m^2
ρ_a	1.29 kg/m^3
m_{11}	16.45 kg
m_{22}	493.5 kg
m_{66}	7896 kg

In practice, the displacement of the airship can be measured by GPS. The traditional PD control method and the adaptive control proposed in this paper are used to compare the control effects. In the ground coordinate system, the simulation results of the airship position are shown in Figures 2 and 3. The simulation result of the speed of the airship relative to the speed of the air are shown in Figure 4. According to the above research results, under the interference of wind, both the PD and the characteristic model control schemes can achieve the control targets. It can be seen from the first result shown in Figure 4 that the characteristic model control method responds slightly more slowly than the PD control method in the initial stage. After a period of time, the response speed of the characteristic model exceeds that of the PD and reaches the steady state before the PD method. As shown by Figure 5 and the second result shown in Figure 4, the overshoot amount of the characteristic model control method is smaller than that of the PD control, and the PD control has a stronger shock than the characteristic model control method. An integrated system trajectory is shown in Figure 6. Using the control method of the characteristic model, the process of the airship flying to the target point will be more stable.

We have tried our best to find the optimal parameters of the PD controller. Parameter tuning is a difficult process in practice, impossible in some cases, especially with some complex equipment. Thus, an adaptive controller is necessary for a changing environment.

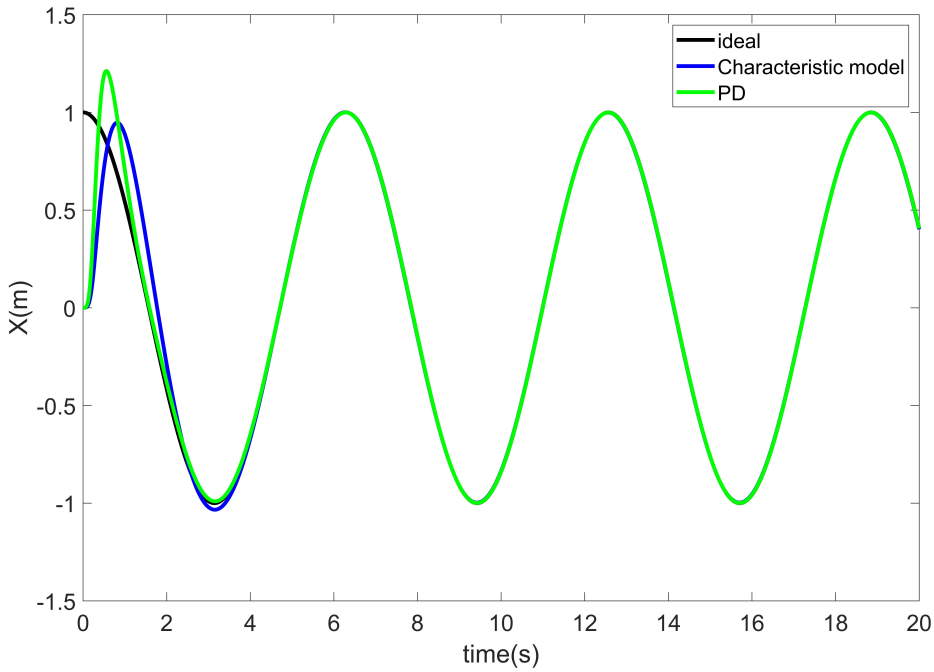


Figure 2. The position (X) of the airship in the ground coordinate system.

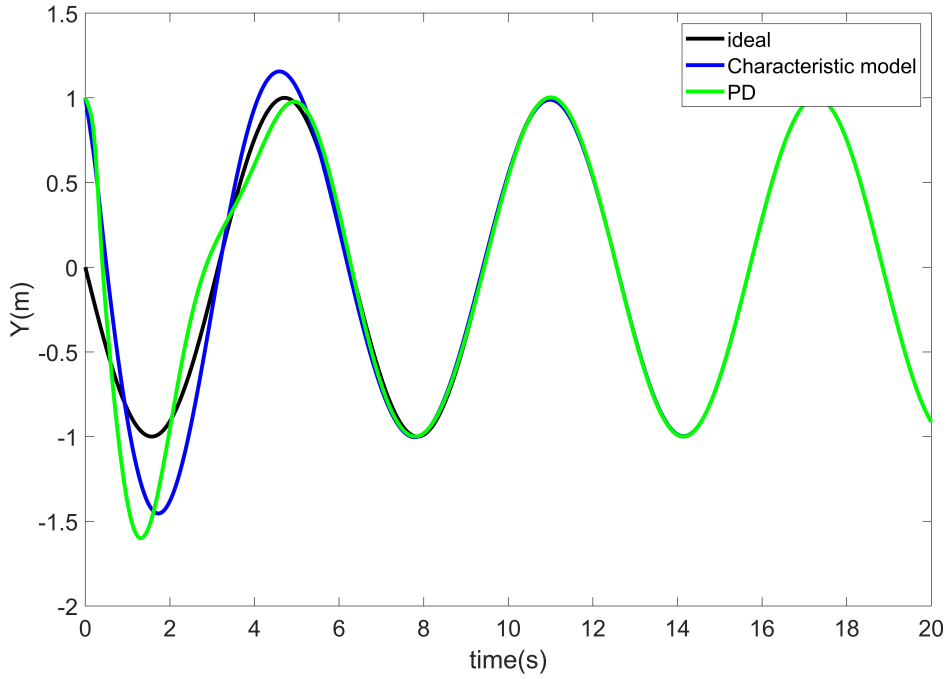


Figure 3. The position (Y) of the airship in the ground coordinate system.

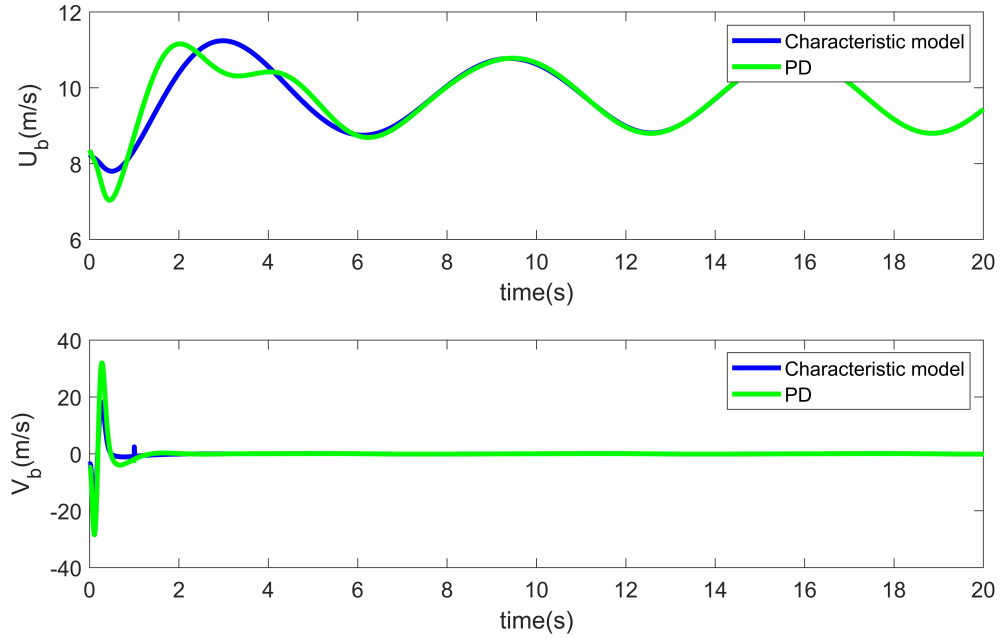


Figure 4. The forward speed of the airship (top); The lateral speed of the airship (bottom).

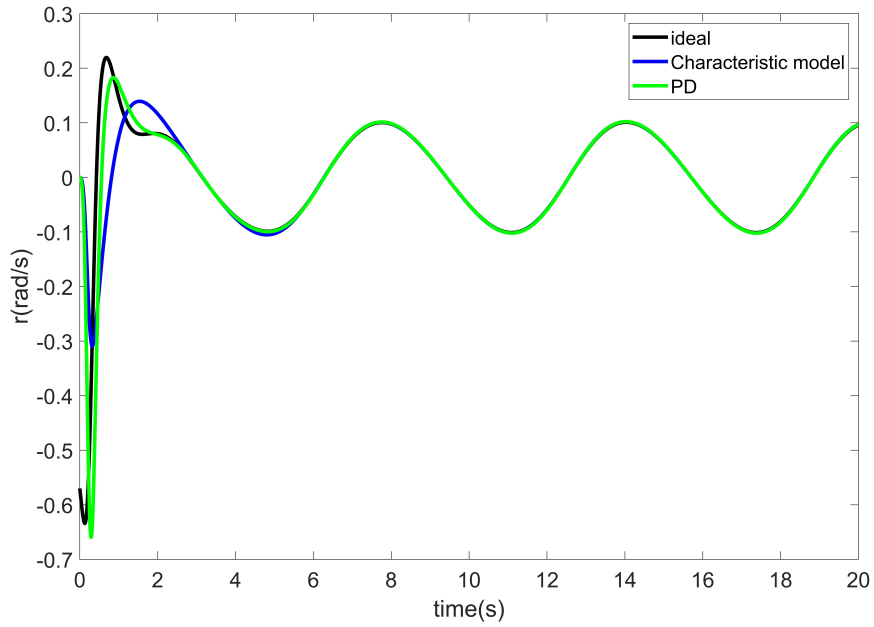


Figure 5. The yaw speed of the airship.

Table 2. Airship model parameters.

Coefficients	Value
c_x	0.0437
c_{y_0}	0
c_{ybeta}	-1.2534
c_{yr}	0.644
c_{n0}	0
c_{nbeta}	0.323
c_{nr}	-0.1659

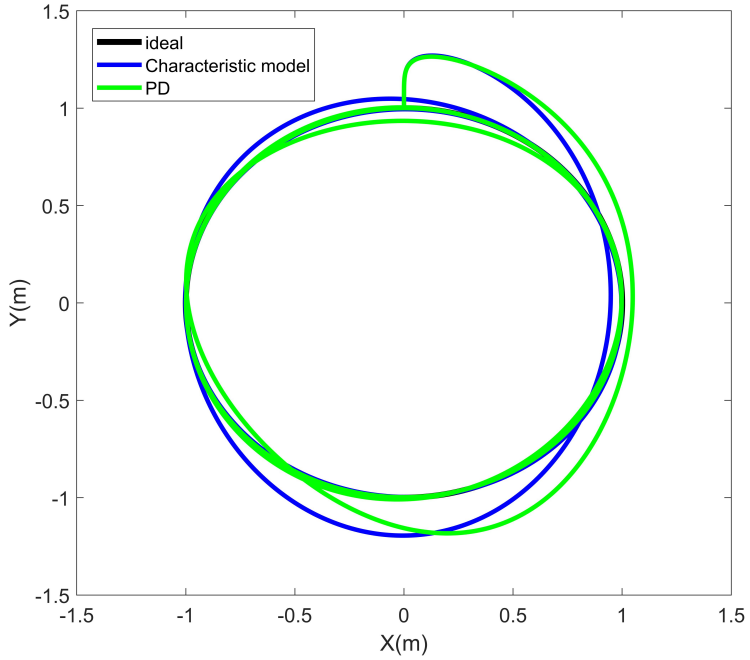


Figure 6. The tracking trajectory of the airship.

7. Fix-Point Control by RL

In this section, we solve the airship fixed-point control problem using well-known adaptive dynamic programming (ADP) with the goal of achieving stability and optimality and show results, as in [15]. The constraint of the performance index on the system is the main characteristic of this algorithm (RL, ADP) [38,39]. For simplicity, we assume the plant operates in a small neighborhood of the operating point. We then linearize the original system on this fixed point. Namely, one can acquire an optimal policy online with the linear model of the original system. On the other hand, a nonlinear optimal policy can also be determined with the help of a neural network. To this end, we have the following system, which can be acquired by calculating the Jacobian matrix on the system.

$$\dot{x} = Ax + Bu \tag{56}$$

The design objective is to find a linear optimal control law $u = -kx$ that can stabilize the system and minimize the following performance index:

$$J(t, x_0) = \int_t^{\infty} \left[x(t)^T Q x(t) + u(t)^T R u(t) \right] dt \tag{57}$$

where $Q = Q^T \geq 0$, with $(A, \sqrt{Q})$ observable, and $R = R^T > 0$.

By optimal control theory and the reinforcement learning method, one can obtain optimal gain without accessing both A and B by identifying the Lyapunov equation $A_k^T P_k + P_k A_k = -Q - K_k^T R K_k$, with a symmetric positive definite P_k, where $A_k = A - BK_k$, and a policy update equation $K_{k+1} = R^{-1} B^T P_k$ at iteration step k, which is known as policy iteration, shown in Algorithm 1. For this purpose, some datasets are collected along the single system trajectory before the system is stable. The following datasets are needed:

$$\Lambda = [\bar{x}(t_1) - \bar{x}(t_0), \bar{x}(t_2) - \bar{x}(t_1), \cdots, \bar{x}(t_l) - \bar{x}(t_{l-1})]^T,$$

$$\Phi_{xx} = \left[\int_{t_0}^{t_1} x(t) \otimes x(t) dt, \int_{t_1}^{t_2} x(t) \otimes x(t) dt, \cdots, \int_{t_{l-1}}^{t_l} x(t) \otimes x(t) dt \right]^T,$$

$$\Phi_{xu} = \left[\int_{t_0}^{t_1} x(t) \otimes u(t) dt, \int_{t_1}^{t_2} x(t) \otimes u(t) dt, \cdots, \int_{t_{l-1}}^{t_l} x(t) \otimes u(t) dt \right]^T.$$

where symbol $\otimes$ denotes the Kronecker product, and we obtain the following operators $\bar{x} = \left[x_1^2, x_1 x_2, \cdots, x_1 x_n, x_2^2, x_2 x_3, \cdots, x_{n-1} x_n, x_n^2 \right]^T$.

Algorithm 1 On-Policy Iteration for Airship Control.

1: **Initialization:**
   Select any initial admissible control policy $u_0 = -K_0 x$
2: Let $k = 0$ and choose stop condition $\varepsilon > 0$
3: **repeat**
4: Update P_k based on Equation
   $(A - BK_k)^T P_k + P_k (A - BK_k) + Q + K_k^T R K_k = 0$
5: Update K_{i+1} based on equation
   $K_k = R^{-1} B^T P_{k-1}$
6: Update u based on Equation
   $u = -K_k x$
7: $k \leftarrow k + 1$
8: **until** $\left\| P^{k+1} - P^k \right\| < \varepsilon$

Output: Obtain the optimal control policy u^* and the optimal value function V^*

Enough datasets with sufficient information about system dynamics are obtained in order to determining Equation (57). It is worth pointing out that the algorithm requires data with a high quality; i.e., measuring errors will spoil the system's robustness to some extent. When rank condition [34] is satisfied, the following key equation in RL admits a unique solution:

$$\begin{aligned} & x(t+\Delta t)^T P_k x(t+\Delta t) - x(t)^T P_k x(t) \\ & = -\int_t^{t+\Delta t} x(t)^T (Q + K_k^T R K_k) x(t) dt \\ & + 2\int_t^{t+\Delta t} [u(t) + K_k x(t)]^T R K_{k+1} x(t) dt \end{aligned} \tag{58}$$

In (58), an exploration input v will generally be added in $u(t)$ so as to make the datasets linearly independent.

Remark 5. *Exploration conditions are essential in RL to learn improved policies [12]. However, the choice of exploration noise v is not a trivial task for general reinforcement learning issues and machine learning, particularly for high-dimensional systems. There are some peculiarities in the control of the airship, such as a large volume, a special environment, communication delay, and so on, and this requires that the controller is able to handle a wide variety of problems. RL can overcome the majority of complicated control problems when we do not have a model, but some prior information is necessary. For example, the dimensions of the system matrix (A, B) need to be identified beforehand. Moreover, an initial stable policy may be difficult to acquire in practice.*

In numerical stimulation, we compared RL policy and PD controller with our work in the uncertainty environment. The unpredicted wind speed is the main disturbance that the airship needs to reject. To this end, we will consider wind speeds that have different amplitudes and data uncertainties that are caused by sensor errors. From the perspective of practice, the considerations of these uncertainties are relatively sufficient to meet the robustness requirements of the controller. It should be noted that exploration input v can make datasets informative, but RL shows worse robustness against sensor errors under input v. Surprisingly, the data-driven strategy performs better than both the high-order model-based design and the model-based control method utilizing a low-order model, as predicted by the theory of this study. In fact, this comparison result with the low-order model-based approach is application-specific rather than universal. However, it can be argued that the data-driven approach is a reliable alternative for controller design to address many real-world issues, because even small modeling errors can have a large impact on the overall control performance. We can see that the controller based on the characteristic model outperforms the RL-based control policy and the PD controller.

The results of the simulation in the absence of uncertainty are displayed in Figures 7–11. It is obvious that the characteristic model outperforms the RL and PD in terms of accuracy and output smoothness in the presence of disturbance. Figure 11 displays the comparison of the complete trajectory among PD, RL, and the proposed method. As a result of our research on adaptive control, we discovered that, while the optimal control algorithm has strong stability, its output is not smooth. This shows that the characteristic model-based control suggested in this study is superior. To verify the adaptability to different trajectories, we simulated a relatively complex trajectory (astroid $x = \cos^3 t$, $y = \sin^3 t$) as shown in Figures 12 and 13. One can see that the proposed approach is insensitive to certain disturbances from Figure 12.

Remark 6. *Integral reinforcement learning (IRL) plays an important role in the control theory community, as it makes it possible to obtain the optimal and stable control policy online. However, there are many integral operations in plants, such as IMU and the calculation of (58), so errors will accumulate unpredictably. Although some filter methods can be employed to improve data quality, it will inevitably put higher demands on computing power. When there are no disturbances, one can see learning convergence as shown by Figures 14 and 15.*

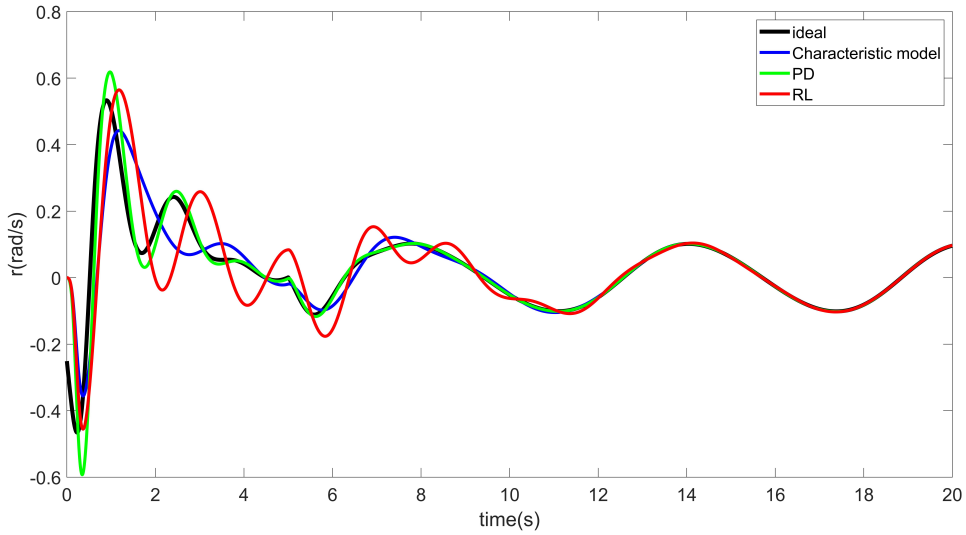


Figure 7. The yaw speed of the airship in the presence of disturbances.

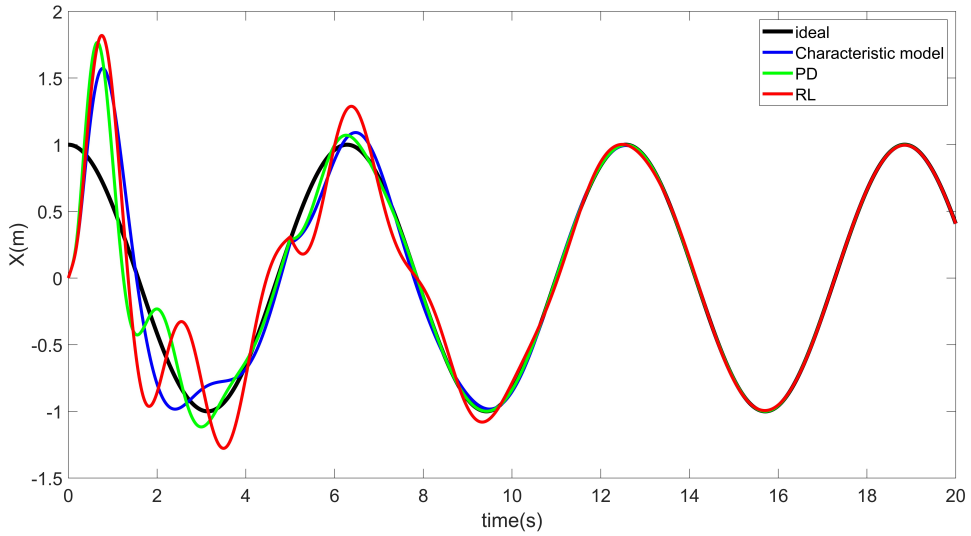


Figure 8. The position (X) of the airship in the ground coordinate system in the presence of disturbances.

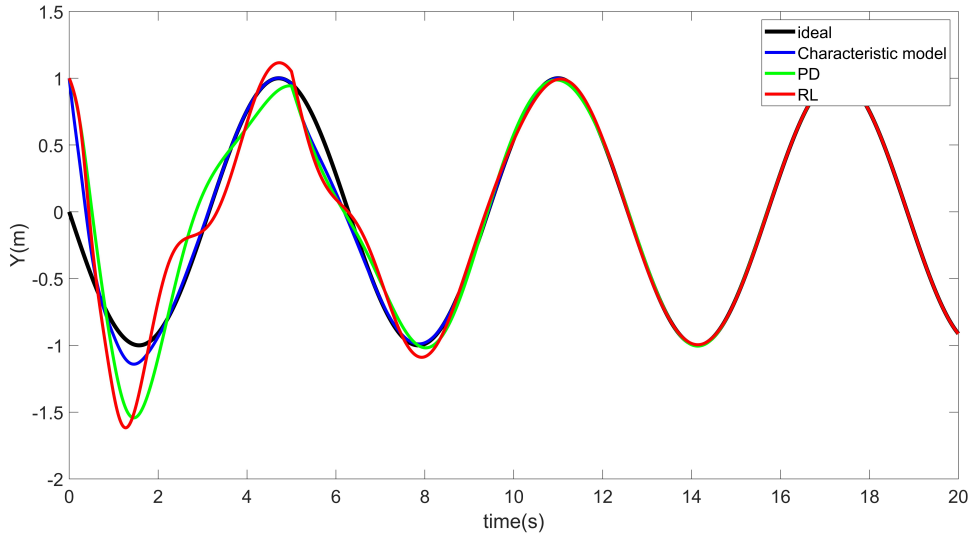


Figure 9. The position (Y) of the airship in the ground coordinate system in the presence of disturbances.

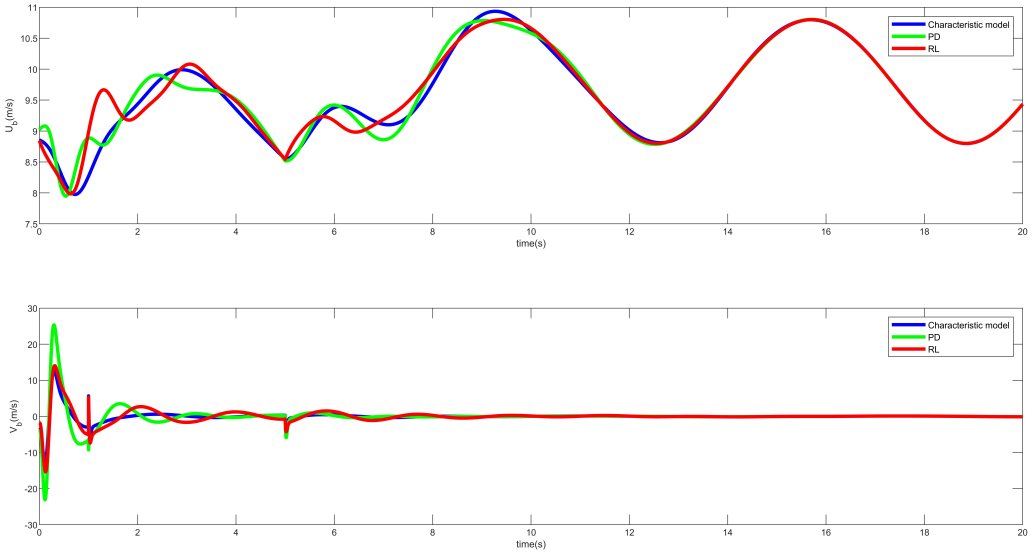


Figure 10. The forward speed of the airship (top); The lateral speed of the airship (bottom) (in the presence of disturbances).

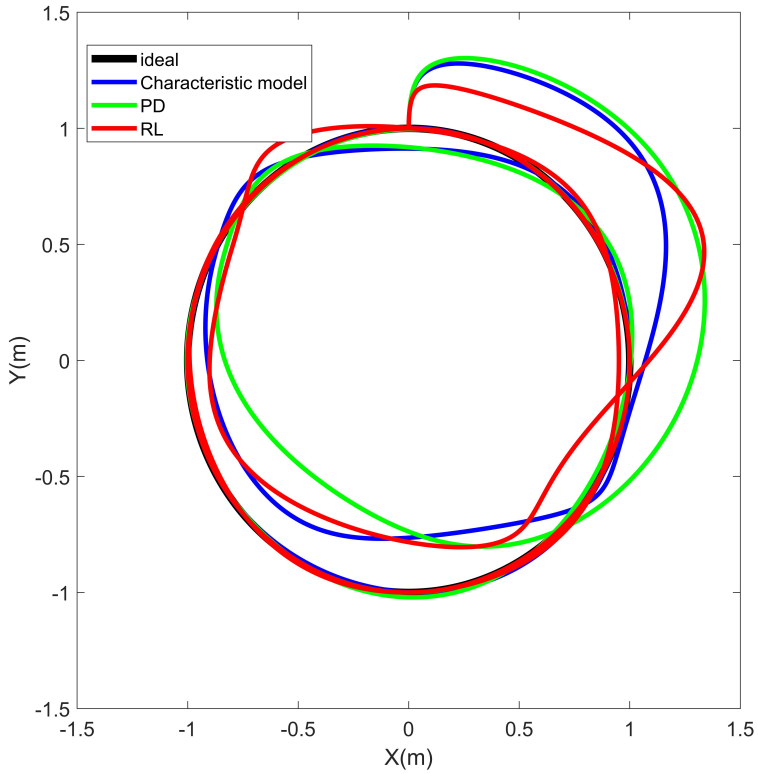


Figure 11. The tracking trajectory of the airship in the presence of disturbances.

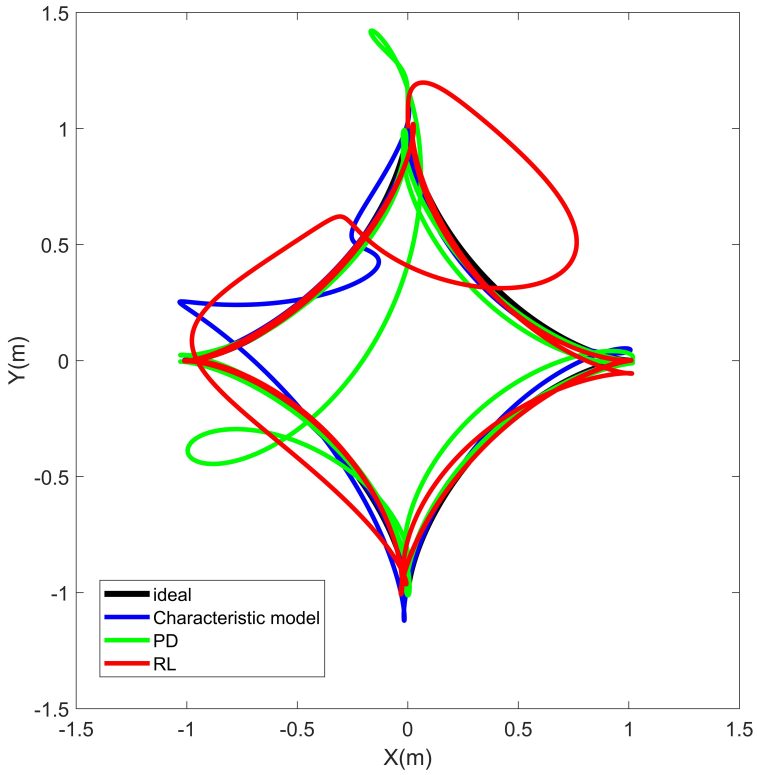


Figure 12. Simulation results for complex trajectories in the absence of disturbances.

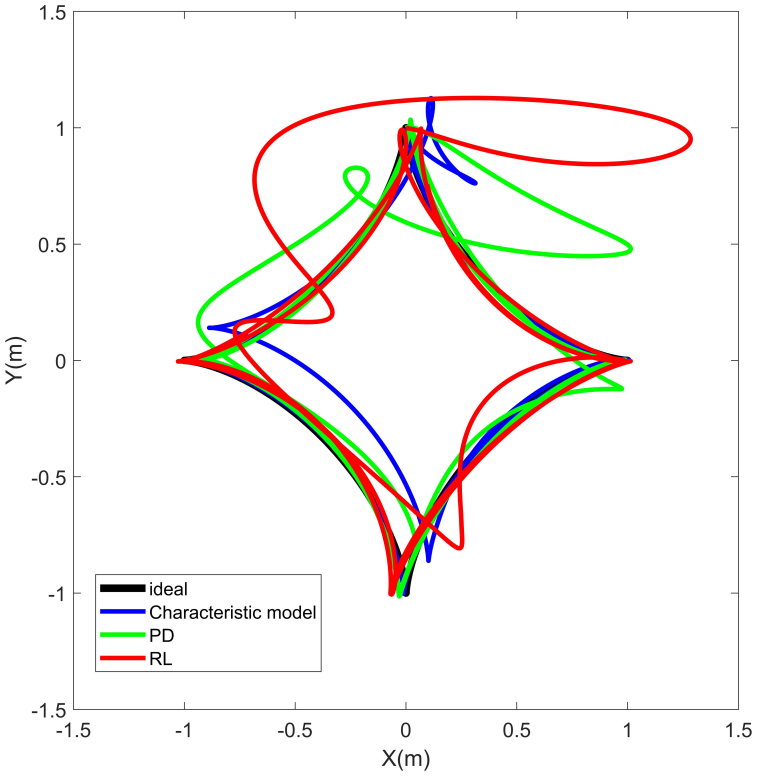


Figure 13. Simulation results for complex trajectories in the presence of disturbances.

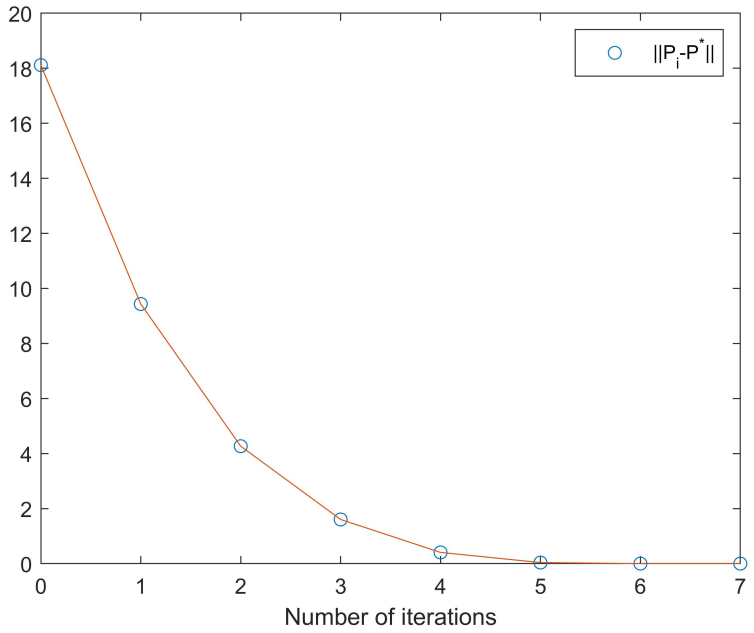


Figure 14. Evolution of the parameters (value function) of the P matrix for the duration of the experiment.

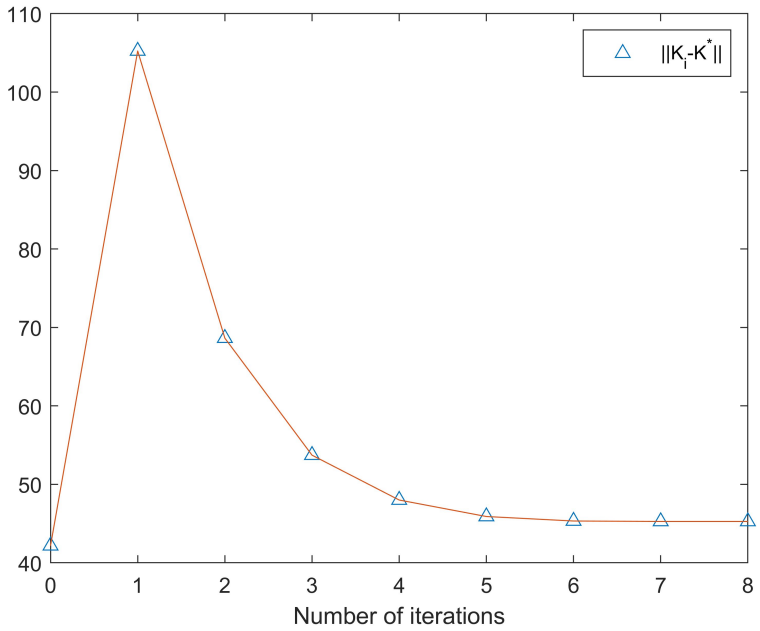


Figure 15. Evolution of the parameters of the control policy K matrix for the duration of the experiment.

8. Conclusions

In this paper, we propose a new golden section adaptive controller based on path tracking theory. The characteristic motion model of a high-altitude airship is build and achieves control targets of fixed point hovering. Through the study, it is shown that characteristic model control achieves a superior control effect when solving the tracking control problem of a complex system, with a fast response, a small overshoot, and a high robustness, which is helpful for the further popularization and application of the characteristic model control method. This method was also compared to reinforcement learning algorithms to a relatively conservative degree and showed advantages in online running from a practical point of view. In contrast with the classic dynamic adaptive controllers, the proposed algorithm shows a superior performance. Issues such as the

neural network approximation methods and the choice of a persistent excitation condition for RL will be addressed in future work.

Author Contributions: Conceptualization, S.S. and Y.C.; methodology, Y.C.; software, Z.H.; validation, Y.C., Z.H. and L.H.; formal analysis, S.S. and Y.C.; investigation, Z.H.; resources, Z.H.; data curation, Z.H.; writing—original draft preparation, S.S. and Y.C.; writing—review and editing, S.S. and Y.C.; visualization, Z.H.; supervision, S.S.; project administration, S.S.; funding acquisition, S.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research is supported by the National Natural Science Foundation (NNSF) of China under Grants 61333008, 61603320, 61733017, and 61673327 and by the Xiamen Key Lab of Big Data Intelligent Analysis and Decision.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare that they have no conflict of interest.

References

1. Wang, H.F.; Song, B.F.; Wang, H.P. Key Technology and the Preliminary Exploration for the Position Control of High Altitude Airships. *Flight Dyn.* **2005**, *23*, 5.
2. Mayrhofer, M.; Wagner, O.; Sachs, G.; Callies, R.; Dinkelmann, M.; Wchter, M.; Stich, R.; Cox, T.H.; Zhringer, C. *Flight Mechanics and Control*; Basic Research and Technologies for Two-Stage-to-Orbit Vehicles; Wiley: Hoboken, NJ, USA, 2006.
3. Liu, G.; Zhang, Y.J. Analysis and Simulation of Stratospheric Airship's Fixed-Point Resident Control. *Ordnance Ind. Autom.* **2008**, *27*, 64–66.
4. Shen, S.P. Fixed point control of high altitude airship based on state estimations and the characteristic model. *Electron. Des. Eng.* **2017**, *25*, 1–5.
5. Chen, W.J.; Dong, S.L. Research and development of airship and high altitude long endurance platform in Germany (Europe). *Spat. Struct.* **2006**, *12*, 3–7.
6. Liang, D. Modeling and Stability Analysis for a Stratospheric Airship in Position-Attitude Keeping Mode. *Spacecr. Eng.* **2007**, *16*, 108–113.
7. Gao, W.; Wang, S.; Jiang, L.H. Fixed-Point Control Of Airship Based On Neural Network Dynamic Inversion. *Microcomput. Inf.* **2010**, *13*, 56–57+60.
8. Su, C.Y.; Stepanenko, Y. Adaptive variable structure set-point control of underactuatedrobots. *IEEE Trans. Autom. Control* **1999**, *44*, 2090–2093.
9. Seto, D.; Annaswamy, A.M.; Baillieul, J. Adaptive control of nonlinear systems with a triangular structure. *IEEE Trans. Autom. Control* **1994**, *39*, 1411–1428. [CrossRef]
10. Bai, W.; Li, T.; Tong, S. NN Reinforcement Learning Adaptive Control for a Class of Nonstrict-Feedback Discrete-Time Systems. *IEEE Trans. Cybern.* **2020**, *50*, 4573–4584. [CrossRef]
11. Asadi, D.; Ahmadi, K. Nonlinear robust adaptive control of an airplane with structural damage. *Proc. Inst. Mech. Eng. Part G J. Aerosp. Eng.* **2020**, *234*, 2076–2088. [CrossRef]
12. Yue, L. and Tong, S. and Li, Y. Observer-Based Adaptive Fuzzy Backstepping Control for a Class of Stochastic Nonlinear Strict-Feedback Systems. *IEEE Trans. Cybern.* **2011**, *44*, 1693–1704.
13. Oh, C.S.; Bang, H.; Park, C.S. Attitude control of a flexible launch vehicle using an adaptive notch filter: Ground experiment. *Control Eng. Pract.* **2008**, *16*, 30–42. [CrossRef]
14. Zhang, H.; Qin, C.; Jiang, B.; Luo, Y. Online Adaptive Policy Learning Algorithm for H-infinity State Feedback Control of Unknown Affine Nonlinear Discrete-Time Systems. *IEEE Trans. Cybern.* **2014**, *44*, 2706–2718. [CrossRef]
15. Puriel-Gil, G.; Wen, Y.; Sossa, H. Reinforcement Learning Compensation based PD Control for Inverted Pendulum. In Proceedings of the 2018 15th International Conference on Electrical Engineering, Computing Science and Automatic Control (CCE), Mexico City, Mexico, 5–7 September 2018.
16. Rubio, J.D.J.; Orozco, E.; Cordova, D.A.; Islas, M.A.; Pacheco, J.; Gutierrez, G.J.; Zacarias, A.; Soriano, L.A.; Meda-Campaña, J.A.; Mujica-Vargas, D. Modified linear technique for the controllability and observability of robotic arms. *IEEE Access* **2022**, *10*, 3366–3377. [CrossRef]
17. Balcazar, R.; Rubio, J.; Orozco, E.; Cordova, D.A.; Ochoa, G.; Garcia, E.; Pacheco, J.; Gutierrez, G.J.; Mujica-Vargas, D.; Aguilar-Ibaez, C. The Regulation of an Electric Oven and an Inverted Pendulum. *Symmetry* **2022**, *14*, 759. [CrossRef]
18. Soriano, L.A.; Zamora, E.; Vazquez-Nicolas, J.M.; Hernández, G.; Balderas, D. PD Control Compensation Based on a Cascade Neural Network Applied to a Robot Manipulator. *Front. Neurorobot.* **2020**, *14*, 577749. [CrossRef]
19. Silva-Ortigoza, R.; Hernandez-Marquez, E.; Roldan-Caballero, A.; Tavera-Mosqueda, S.; Silva-Ortigoza, G. Sensorless Tracking Control for a "Full-Bridge Buck Inverter–DC Motor" System: Passivity and Flatness-Based Design. *IEEE Access* **2021**, *9*, 132191–132204. [CrossRef]

20. Meda-Campaña, J.A.; Rodríguez-Manzanarez, R.A.; Ontiveros-Paredes, S.D.; de Jesús Rubio, J.; Tapia-Herrera, R.; Hernández-Cortés, T.; Obregón-Pulido, G.; Aguilar-Ibáñez, C. An Algebraic Fuzzy Pole Placement Approach to Stabilize Nonlinear Mechanical Systems. *IEEE Trans. Fuzzy Syst.* **2021**, *30*, 3322–3332. [CrossRef]
21. Lughofer, E.; Skrjanc, I. Evolving Error Feedback Fuzzy Model for Improved Robustness under Measurement Noise. *IEEE Trans. Fuzzy Syst.* **2022**. [CrossRef]
22. Lewis, F.L.; Vrabie, D. Reinforcement learning and adaptive dynamic programming for feedback control. *Circuits Syst. Mag. IEEE* **2009**, *9*, 32–50. [CrossRef]
23. Sutton, R.; Barto, A. *Reinforcement Learning: An Introduction*; MIT Press: Cambridge, MA, USA, 1998.
24. Hafner, R.; Riedmiller, M. Reinforcement learning in feedback control. *Mach. Learn.* **2011**, *84*, 137–169. [CrossRef]
25. Ferrari, S.; Steck, J.E.; Chandramohan, R. Adaptive Feedback Control by Constrained Approximate Dynamic Programming. *IEEE Trans. Syst. Man Cybern. Part B Cybern.* **2008**, *38*, 982–987. [CrossRef] [PubMed]
26. Kiumarsi, B.; Lewis, F.L.; Naghibi-Sistani, M.B.; Karimpour, A. Optimal Tracking Control of Unknown Discrete-Time Linear Systems Using Input–Output Measured Data. *IEEE Trans. Cybern.* **2015**, *45*, 2770. [CrossRef] [PubMed]
27. Liu, Y.J.; Tang, L.; Tong, S.; Chen, C.P.; Li, D.J. Reinforcement learning design-based adaptive tracking control with less learning parameters for nonlinear discrete-time MIMO systems. *IEEE Trans. Neural Netw. Learn. Syst.* **2015**, *26*, 165–176. [CrossRef]
28. Liu, Y.; Wu, Y.L.; Hu, Y.M. Autonomous dynamics-modeling and feedback control for an airship. *Control. Theory Appl.* **2010**, *27*, 991–1000.
29. Nie, C.; Zheng, Z.; Zhu, M. Three-Dimensional Path-Following Control of a Robotic Airship with Reinforcement Learning. *Int. J. Aerosp. Eng.* **2019**, *2019*, 7854173.1–7854173.12. [CrossRef]
30. Zhang, J.; Yang, X.; Deng, X.; Lin, H. Trajectory control method of stratospheric airships based on model predictive control in wind field. *Proc. Inst. Mech. Eng.* **2019**, *233*, 418–425. [CrossRef]
31. Zhen, Y.; Hao, M. Aircraft Control Method Based on Deep Reinforcement Learning. In Proceedings of the 2020 IEEE 9th Data Driven Control and Learning Systems Conference (DDCLS), Liuzhou, China, 19–21 June 2020.
32. Formentin, S.; Van Heusden, K.; Karimi, A. A comparison of model-based and data-driven controller tuning. *Int. J. Adapt. Control. Signal Process.* **2014**, *28*, 882–897. [CrossRef]
33. Lewis, F.L.; Vamvoudakis, K.G. Reinforcement Learning for Partially Observable Dynamic Processes: Adaptive Dynamic Programming Using Measured Output Data. *IEEE Trans. Syst. Man Cybern. Part B Cybern.* **2010**, *41*, 14–25. [CrossRef]
34. Jiang, Y.; Jiang, Z.P. Computational adaptive optimal control for continuous-time linear systems with completely unknown dynamics. *Autom. Oxf.* **2012**, *48*, 2699–2704. [CrossRef]
35. Long, H. Study on the Fixed Point of the Airship Based on Characteristic Model. Master's Thesis, Xiamen University, Xiamen, China, 2020.
36. Xin, W.H.; Wang, Y.; Xie, Y.C. Nonlinear golden-section adaptive control. *J. Astronaut.* **2002**, *23*, 1–8.
37. Khan, S.G.; Herrmann, G.; Lewis, F.L.; Pipe, T.; Melhuish, C. Reinforcement learning and optimal adaptive control: An overview and implementation examples. *Annu. Rev. Control* **2012**, *36*, 42–59. [CrossRef]
38. Zhang, Y.; Guo, L.; Gao, B.; Qu, T.; Chen, H. Deterministic Promotion Reinforcement Learning Applied to Longitudinal Velocity Control for Automated Vehicles. *IEEE Trans. Veh. Technol.* **2019**, *69*, 338–348. [CrossRef]
39. Hwangbo, J.; Sa, I.; Siegwart, R.; Hutter, M. Control of a Quadrotor with Reinforcement Learning. *arXiv* **2017**, arXiv:1707.05110.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.