applied sciences

Article

Deep Reinforcement Learning Based Active Disturbance Rejection Control for ROV Position and Attitude Control

Gaosheng Luo [1], Dong Zhang [1], Wei Feng [2], Zhe Jiang [1,3,*] and Xingchen Liu [1]

[1] Shanghai Engineering Research Center of Hadal Science and Technology, College of Engineering Science and Technology, Shanghai Ocean University, Shanghai 201306, China; gsluo@shou.edu.cn (G.L.); zhangdong58496@163.com (D.Z.); 15012648516@163.com (X.L.)
[2] Shenzhen Institute for Advanced Study, University of Electronic Science and Technology of China, Shenzhen 518110, China; wei_feng@oit-sz.com
[3] Lanqi Robot Co., Ltd., Wuxi 214000, China
* Correspondence: zjiang@shou.edu.cn

Abstract: Remotely operated vehicles (ROVs) face challenges in achieving optimal trajectory tracking performance during underwater movement due to external disturbances and parameter uncertainties. To address this issue, this paper proposes a position and attitude control strategy for underwater robots based on a reinforcement learning active disturbance rejection controller. The linear active disturbance rejection controller has achieved satisfactory results in the field of underwater robot control. However, fixed-parameter controllers cannot achieve optimal control performance for the controlled object. Therefore, further exploration of the adaptive capability of control parameters based on the linear active disturbance rejection controller was conducted. The deep deterministic policy gradient (DDPG) algorithm was used to optimize the linear extended state observer (LESO). This strategy employs deep neural networks to adjust the LESO parameters online based on measured states, allowing for more accurate estimation of model uncertainties and environmental disturbances, and compensating the total disturbance into the control input online, resulting in better disturbance estimation and control performance. Simulation results show that the proposed control scheme, compared to PID and fixed parameter LADRC, as well as the double closed-loop sliding mode control method based on nonlinear observers (NESO-DSMC), significantly improves the disturbance estimation accuracy of the linear active disturbance rejection controller, leading to higher control precision and stronger robustness, thus demonstrating the effectiveness of the proposed control strategy.

Keywords: ROV; linear active disturbance rejection control; deep reinforcement learning; deep deterministic policy gradient (DDPG) algorithm; resilience to disturbance

check for updates

Academic Editor: Douglas O'Shaughnessy

Received: 11 March 2025
Revised: 13 April 2025
Accepted: 15 April 2025
Published: 17 April 2025

Citation: Luo, G.; Zhang, D.; Feng, W.; Jiang, Z.; Liu, X. Deep Reinforcement Learning Based Active Disturbance Rejection Control for ROV Position and Attitude Control. *Appl. Sci.* **2025**, *15*, 4443. https://doi.org/10.3390/app15084443

1. Introduction

Remotely operated vehicles (ROVs) play an important role in underwater inspection, marine salvage, and deep-sea mining. Currently, the requirements for the control accuracy and robustness of underwater robots are also increasing. However, due to the nonlinear dynamics, external disturbances, and parameter uncertainties present in the underwater movement of ROVs, designing a reliable tracking controller is challenging. In recent decades, an increasing number of scholars have conducted extensive research on the control stability of ROVs. The control methods they used are elaborated upon below.

Proportional-integral-derivative (PID) control is the most widely used approach in industrial control. Guerrero et al. proposed a saturation function-based nonlinear PID controller, effectively addressing the control instability issues in underwater vehicles caused by actuator saturation and complex environmental disturbances [1]. Sarhadi et al. proposed a model reference adaptive PID control structure with an anti-saturation compensator to address the issue of model uncertainty in autonomous underwater vehicle systems [2].

Fuzzy control is a control method similar to expert systems. Han et al. proposed a fuzzy logic system to address the issue of unknown inertia matrices in AUV systems [3]. Li et al. proposed a fuzzy adaptive controller that considers the dynamics of ROV thrusters to improve the trajectory tracking performance of work-class ROVs, using a fuzzy adaptive control algorithm to compensate for changes in system parameters and disturbances [4]. Yang et al. proposed a fuzzy logic system (FLS) to replace the discontinuous switching terms in CSMC to reduce chattering phenomena [5].

Sliding Mode Control (SMC) is often used for trajectory tracking of underwater robots due to its resistance to external disturbances and parameter variations. Cheng et al. proposed a method that combines a finite-time observer with adaptive sliding mode control to achieve high-precision robust tracking of underwater vehicles [6]. Long et al. proposed an Adaptive Sliding Mode Control (ASMC) to construct a dynamic controller that calculates the optimal force and torque based on the output virtual speed. This approach is robust to parameter uncertainties and addresses the issue of flutter [7]. Luo et al. proposed an improved sliding surface has been proposed to address the finite selection problem of exponential parameters, resulting in a controller with better robustness [8]. Huang et al. introduced a double closed-loop sliding mode controller for trajectory tracking control of working-class ROVs, which uses the arctangent function as the switching function of the controller, effectively reducing chattering phenomena [9].

Neural network control (NNC) has emerged as a potent tool for crafting controllers for nonlinear and uncertain systems. Wen et al. proposed a predefined time control strategy using Radial Basis Function Neural Networks (RBFNNs), which effectively approximates external disturbances, thereby enhancing the robustness of the system [10]. Chu et al. proposed an adaptive control scheme based on radial basis function (RBF) neural networks for ROV trajectory tracking. To ensure system stability under actuator saturation, a first-order auxiliary state system was constructed [11]. Shojaei et al. proposed a neural network-enhanced feedback linearization control framework, which effectively addresses the performance guarantee issues of underactuated AUVs under model uncertainties and disturbances [12].

Each of the methodologies mentioned above has specific limitations. As the complexity of the Remotely Operated Vehicle (ROV) model increases, the effectiveness of proportional-integral-derivative (PID) control decreases significantly. Fuzzy control heavily relies on a fuzzy rule base constructed from expert experience, while SMC (Sliding Mode Control) has a very high dependency on the model and is prone to high-frequency chattering issues [13,14]. Neural network control (NNC) is particularly influenced by the number of nodes in the neural network; while increasing the nodes can improve control accuracy, it also leads to a significant rise in computational complexity, posing challenges for practical engineering applications [15]. Furthermore, these approaches do not adequately address the constraints related to the ROV's state, potentially compromising tracking precision and risking damage to the thrusters.

To overcome the limitations associated with the status of the Remotely Operated Vehicle (ROV), this research employs Linear Active Disturbance Rejection Control (LADRC), an optimal control approach. LADRC preserves the essential characteristics of the proportional-integral-derivative (PID) algorithm without requiring an accurate model of the controlled system [16]. Instead, it treats the unmodeled dynamics and external disturbances as "total disturbances," which are then estimated and compensated for. This methodology offers significant advantages in engineering applications due to its ease of use and robust resistance to interference [17,18]. Zhao et al. introduced a trajectory tracking control method for a dual-joint robotic arm system, integrating an extended state observer to estimate both the disturbances and states of the system. Additionally, they applied a state error feedback controller, and experimental findings indicate that the proposed control approach effectively meets control requirements under various conditions, including low-frequency, high-frequency, load, and disturbance scenarios [19,20]. Despite the successful implementation of LADRC in nonlinear systems, its control effectiveness is limited by its fixed structure and parameters.

The adaptive adjustment of parameters for control methods has been a prominent subject of interest, with various optimization algorithms being utilized to improve the robustness and control efficacy of these methods [21,22]. Different operational contexts necessitate varying control parameters, posing challenges for controllers with fixed optimization parameters to achieve optimal control performance. Drawing inspiration from artificial intelligence technologies, reinforcement learning (RL) algorithms have been amalgamated with control theory to devise novel control strategies that augment the adaptability of control systems and uphold optimal control performance in real time. Chen introduced a Q-learning-based adaptive tuning technique for LADRC parameters [23], which identifies optimal control parameters through iterative updates of the Q-value table and applies it to the heading control of ships. Nevertheless, the Q-learning algorithm mandates manual partitioning of the states of the controlled object and the specification of discrete actions, rendering it arduous to train and learn efficiently as the number of states and specified actions escalates. Furthermore, due to the discrete actions specified, the controller parameters can only assume predetermined values rather than varying continuously, thereby constraining the controller's flexibility. To solve this problem [24], this study employs the Deep Deterministic Policy Gradient (DDPG) RL algorithm to dynamically generate optimal control gains online for the designed LADRC within the Linear Extended State Observer (LESO), thereby determining the optimal parameters of the extended state observer under diverse unknown disturbance conditions. This methodology circumvents the issue of inaccurate disturbance estimation stemming from fixed parameters, and the efficacy of the algorithm is ultimately corroborated through simulations.

The main contributions of this article include:

1. This article presents a nonlinear model for underwater robots that considers parameter uncertainty in the dynamic model. It also proposes a linear active disturbance rejection controller for controlling the position and attitude of the underwater robot based on this model. The convergence of the extended state observer in the active disturbance rejection controller and the stability of the closed-loop control system are demonstrated using the Lyapunov method.
2. A novel control method, named DDPG-LADRC, has been introduced to address disturbances in linear systems by integrating the Deep Deterministic Policy Gradient (DDPG) algorithm with an active disturbance rejection control approach. This method focuses on optimizing the extended state observer through the DDPG algorithm, enabling the observer to sustain optimal performance under varying external disturbances during both position and attitude control of Remotely Operated Vehicles (ROVs). Through real-time adjustments of control parameters, the performance of the extended state observer (LESO) is enhanced, thereby improving the system's resilience to disturbances and enhancing control accuracy in intricate underwater settings.
3. Based on a nonlinear underwater robot model, numerical simulations have confirmed the efficacy of the approach. The simulation results first compared three control algorithms: PID, fixed-parameter LADRC, and DDPG-LADRC, and finally included

NESO-DSMC for comparison. Through analysis, the proposed method has been verified to have significant advantages in terms of transient performance, control accuracy, and robustness.

The subsequent sections of this article are structured as follows: Section 2 introduces a nonlinear model for the underwater robot, Section 3 outlines the design of the LADRC controller for the robot, and Section 4 introduces the DDPG-LADRC algorithm and analyzes the convergence of the extended state observer and the stability of the closed-loop control. Section 5 presents the results and analysis of numerical simulations, and lastly, a conclusion is offered.

2. Dynamics Model of a Robot

This section investigates the dynamic model of an ROV for offshore underwater structure marine growth cleaning and structural inspection independently developed by the Hadal Science and Technology Center of Shanghai Ocean University. The robot is equipped with eight thrusters, allowing it to execute three-dimensional spatial maneuvers. Figure 1 shows the coordinate system of the robot and defines the inertial coordinate system $(x_0,\ y_0, z_0)$ and the motion coordinate system $(x,\ y, z)$. The state variables relative to the motion coordinate system are represented as $V = [u, v, w, p, q, r]^T$, where u, v, w represent linear velocity, and p, q, r represent angular velocity. The state variables relative to the inertial coordinate system are represented as $[x, y, z, \phi, \theta, \psi]^T$, where x, y, z indicate the position of the ROV, and ϕ, θ, ψ represent the attitude of the ROV. The kinematic equations of the ROV can be expressed as $\dot{\eta} = J(\eta)\nu$ [25]. The roll and pitch are passively stabilized by the buoyancy system, requiring no active control, so $\phi = \theta = 0$. Therefore, the six degrees of freedom motion of the ROV can be simplified to four degrees of freedom motion.

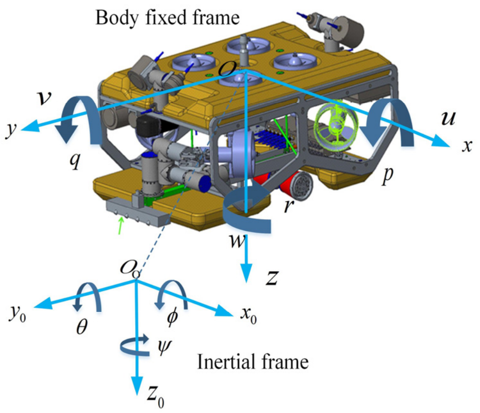


Figure 1. Remotely operated vehicle coordinate system.

The dynamic equations in the Remotely Operated Vehicle (ROV) motion coordinate system are presented below [25]:

$$M\dot{\nu} + C(v)v + D(v)v + g(\eta) = \tau + \Delta f \tag{1}$$

In the dynamics Equation (1) of the ROV, M represents the inertia matrix of the ROV, v represents the linear velocities and angular velocities of the ROV, $\dot{\nu}$ represents the linear and angular accelerations of the ROV, $C(v)$ represents the Coriolis-centrifugal matrix, $D(v)$ represents the hydrodynamic damping matrix, $g(\eta)$ represents the restoring force vector, τ is the control input provided by the main thrust and torque from the ROV's thrusters, and Δf represents external water flow disturbances and uncertainties such as unmodeled dynamics. The movement of an ROV can be conceptualized as the general motion of a rigid body influenced by gravity and hydrodynamics in a water flow. Usually, the fluid dynamics parameters are derived from experiments or fluid simulations. However, due to the complexity of real-world ocean conditions, ensuring the accuracy of these parameters is a significant challenge. Consequently, the parameters M, $C(v)$, and $D(v)$ in the equation remain indeterminate. The $M, C(v), D(v)$ in Equation (1) can be represented as the combination of the nominal parameters $M_0, C_0(\nu), D_0(\nu)$ and the dynamic uncertainties ΔM_0, $\Delta C_0(v)$, $\Delta D_0(v)$, as follows:

$$\begin{cases} M = M_0 + \Delta M \\ C(v) = C_0(v) + \Delta C(v) \\ D(v) = D_0(v) + \Delta D(v) \end{cases} \tag{2}$$

Then, Equation (1) can be rewritten as:

$$M_0\dot{\nu} + C_0(v)v + D_0(v)v + g(\eta) = \tau + \Delta f + \tau_\Delta \tag{3}$$

$$M_{RB} = \begin{bmatrix} m & 0 & 0 & -my_G \\ 0 & m & 0 & mx_G \\ 0 & 0 & m & 0 \\ -my_G & mx_G & 0 & I_z \end{bmatrix} \qquad M_A = -\begin{bmatrix} X_{\dot{u}} & X_{\dot{v}} & X_{\dot{w}} & X_{\dot{r}} \\ Y_{\dot{u}} & Y_{\dot{v}} & Y_{\dot{w}} & Y_{\dot{r}} \\ Z_{\dot{u}} & Z_{\dot{v}} & Z_{\dot{w}} & Z_{\dot{r}} \\ N_{\dot{u}} & N_{\dot{v}} & N_{\dot{w}} & N_{\dot{r}} \end{bmatrix} \tag{4}$$

By assuming that the origin O of the dynamic system coincides with the centroid G and that the coordinate axes coincide with the three principal inertia axes, and ignoring the off-diagonal elements in the M_{RB} and M_A matrices [25], we can obtain the M_0 matrix:

$$M_0 = M_{RB} + M_A = \begin{bmatrix} m - X_{\dot{u}} & 0 & 0 & 0 \\ 0 & m - Y_{\dot{v}} & 0 & 0 \\ 0 & 0 & m - Z_{\dot{w}} & 0 \\ 0 & 0 & 0 & I_Z - N_{\dot{r}} \end{bmatrix} \tag{5}$$

In Formula (5), $M_0 \in R^{4\times4}$ represents the inertia matrix, which is composed of the sum of the rigid body mass matrix M_{RB} and the added mass matrix, M_A. m is the mass of the ROV, I is the moment of inertia, and $X_{\dot{u}}, Y_{\dot{v}}, Z_{\dot{w}}$ represent the hydrodynamic forces induced by the added mass in the x, y, and z directions, respectively, with unit acceleration along the $\dot{u}$, $\dot{v}$, and $\dot{w}$ axes. $N_{\dot{r}}$ represents the additional inertial force generated by the unit angular acceleration $\dot{r}$ in the direction of the z-axis.

$$C_A(\nu) = \begin{bmatrix} 0 & 0 & 0 & Y_{\dot{v}}\nu \\ 0 & 0 & 0 & -X_{\dot{u}}u \\ 0 & 0 & 0 & 0 \\ -Y_{\dot{v}}\nu & X_{\dot{u}}u & 0 & 0 \end{bmatrix} \qquad C_{RB}(\nu) = \begin{bmatrix} 0 & 0 & 0 & -m\nu \\ 0 & 0 & 0 & mu \\ 0 & 0 & 0 & 0 \\ m\nu & -mu & 0 & 0 \end{bmatrix}$$
$$C_0(\nu) = C_{RB} + C_A = \begin{bmatrix} 0 & 0 & 0 & -(m - Y_{\dot{v}})\nu \\ 0 & 0 & 0 & (m - X_{\dot{u}})u \\ 0 & 0 & 0 & 0 \\ (m - Y_{\dot{v}})\nu & -(m - X_{\dot{u}})u & 0 & 0 \end{bmatrix} \tag{6}$$

In Formula (6), $C_0(v) \in R^{4\times4}$, $C_0(\nu) = C_{RB} + C_A$, where C_{RB} represents the matrix encompassing the rigid body Coriolis force and centripetal force, while C_A denotes the matrix accounting for the Coriolis force and centripetal force resulting from the added mass of the inertial fluid dynamics [25].

$$D_0(v) = -\begin{bmatrix} X_u + X_{u|u|}|u| & 0 & 0 & 0 \\ 0 & Y_v + Y_{v|v|}|v| & 0 & 0 \\ 0 & 0 & Z_w + Z_{w|w|}|w| & 0 \\ 0 & 0 & 0 & N_r + N_{r|r|}|r| \end{bmatrix} \tag{7}$$

In the Formula (7), $D_0(V) \in R^{4\times4}$ denotes the damping matrix, which arises from the effects of viscous fluid dynamics on the robot. The symbols X_u, Y_v, Z_w, N_r and $X_{u|u|}, Y_{v|v|}, Z_{w|w|}, N_{r|r|}$ correspondingly denote the primary and secondary hydrodynamic damping coefficients that emerge during the motion of the underwater robot [25].

$$g(\eta) = \begin{bmatrix} 0 & 0 & -(W-B) & 0 \end{bmatrix}^T \tag{8}$$

In Formula (8), $g(\eta)$ is represented as the restoring force and moment caused by gravity and buoyancy [25]. W is the gravity of the underwater robot, and B is the buoyancy. In the physical structure design of the ROV, buoyancy is equal to gravity, so it can be further expressed as: $g(\eta) = \begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix}^T$.

In Formula (3), $\tau \in R^{4\times4}$ represents the thrust generated by the propeller, expressed as $\tau = \left[F_x, F_y, F_Z, N_Z\right]^T$, where F_x, F_y, F_Z are the thrusts generated by the propeller along the three coordinate axes, and N_Z are the moments generated by the propeller around the coordinate axes.

In Formula (3), $\Delta f \in R^{4\times4}$ represents external disturbances such as ocean currents in the working environment, and $\tau_\Delta \in R^{4\times4}$ represents the uncertainty of dynamic lumped parameters, where $\tau_\Delta = -\Delta M\dot{v} - \Delta C(v)v - \Delta D(v)v$. For the subsequent design of LADRC, we unify the model parameter uncertainty terms $(\Delta M, \Delta C(V), \Delta D(V))$ and external disturbances as total disturbances.

3. LADRC Controller Design

In addressing the challenges posed by uncertainties in ocean current disturbances, process noise, and hydrodynamic damping coefficients, the Linear Active Disturbance Rejection Control (LADRC) method is utilized. LADRC comprises a Linear Extended State Observer (LESO) and Linear State Error Feedback Control Law (LSEFC). Of particular significance is the development of the LESO, which is implemented to estimate and counteract external disturbances and uncertainties in model parameters. The LSEFC determines the virtual control signal u_0 by evaluating the system's state error. The control block diagram is illustrated in Figure 2.

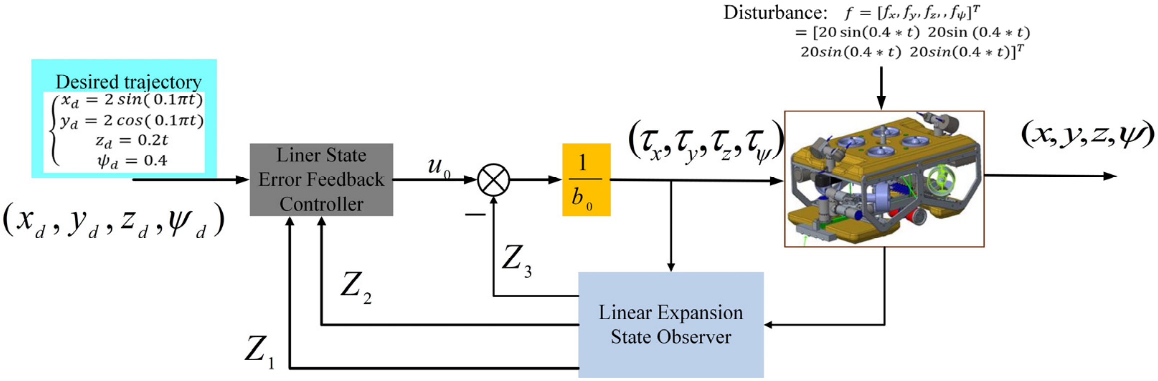

Disturbance: $f = [f_x, f_y, f_z, , f_\psi]^T$
$= [20\sin(0.4*t)\ 20\sin(0.4*t)$
$20sin(0.4*t)\ 20sin(0.4*t)]^T$
Desired trajectory
$x_d = 2\,sin(0.1\pi t)$
$y_d = 2\,cos(0.1\pi t)$
$z_d = 0.2t$
$\psi_d = 0.4$
(x_d, y_d, z_d, ψ_d)
Liner State Error Feedback Controller
u_0
$-$
$\frac{1}{b_0}$
$(\tau_x, \tau_y, \tau_z, \tau_\psi)$
(x, y, z, ψ)
Z_3
Z_2
Z_1
Linear Expansion State Observer

Figure 2. Structure diagram of linear active disturbance rejection controller.

3.1. Linear Extended State Observer

LESO improves the performance of control systems by estimating and compensating for disturbances and unknown state variables. This section outlines the design process of the LESO scheme based on the mathematical model of the ROV system. The kinematics expression of the ROV is $\dot{\eta} = J(\eta)\nu$ [25], from which we can derive Formula (9):

$$\begin{gathered}\ddot{\eta} = J(\eta)\dot{\nu} + \dot{J}(\eta)\nu \\ = J(\eta)M_0^{-1}(\Delta f + \tau_\Delta - C_0(\nu)v - D_0(\nu)v + g(\eta)) + \dot{J}(\eta)\nu + J(\eta)M_0^{-1}\tau \\ = f + X + J(\eta)M_0^{-1}\tau\end{gathered} \tag{9}$$

In Formula (9), $X = \dot{J}(\eta)\nu - J(\eta)M_0^{-1}(C_0(\nu)v - D_0(\nu)v + g(\eta)), f = J(\eta)M_0^{-1}(\Delta f + \tau_\Delta)$ represents the total uncertainty of the unmodeled dynamics and external disturbances. For the convenience of designing the LESO, let $f = [f_x, f_y, f_z, f_\psi]$, $\tau = \left[\tau_x, \tau_y, \tau_z, \tau_\psi\right]$, then Formula (9) can be written as [26]:

$$\begin{gathered}\ddot{x} = a_1\tau_x + X_1 + f_x \\ \ddot{y} = a_2\tau_y + X_2 + f_y \\ \ddot{z} = a_3\tau_z + X_3 + f_z \\ \ddot{\psi} = a_6\tau_\psi + X_6 + f_\psi\end{gathered} \tag{10}$$

$$\left\{\begin{gathered}X_1 = \tfrac{cos\psi}{m-X_{\dot{u}}}\left(X_u + X_{u|u|} \mid u \mid -(m - Y_{\dot{v}})ur\right) \\ -\tfrac{sin\psi}{m-Y_{\dot{v}}}\left(Y_v + Y_{v|v|} \mid v \mid -(m - X_{\dot{u}})ur\right) \\ -usin\psi - vcos\psi \\ a_1 = \left(\tfrac{cos\psi}{m-X_{\dot{u}}} - \tfrac{sin\psi}{m-Y_{\dot{v}}}\right) \\ X_2 = \tfrac{sin\psi}{m-X_{\dot{u}}}\left(X_u + X_{u|u|} \mid u \mid -(m - Y_{\dot{v}})ur\right) + \tfrac{cos\psi}{m-Y_{\dot{v}}} \\ \left(Y_v + Y_{v|v|} \mid v \mid -(m - X_{\dot{u}})ur\right) + ucos\psi - vsin\psi \\ a_2 = \left(\tfrac{sin\psi}{m-X_{\dot{u}}} + \tfrac{cos\psi}{m-Y_{\dot{v}}}\right) \\ X_3 = \tfrac{\left(Z_w + Z_{w|w|}|w|\right)w}{m-Z_{\dot{w}}} \\ a_3 = \tfrac{1}{m-Z_{\dot{w}}} \\ X_4 = \tfrac{\left(N_r + N_{r|r|}|r|\right)r}{I_z-N_{\dot{r}}}, a_4 = \tfrac{1}{I_z-N_{\dot{r}}}\end{gathered}\right. \tag{11}$$

The total uncertainty f, which represents the unmodeled dynamics and external disturbances, is defined as the total disturbance. To achieve an accurate estimation of the total disturbance f experienced in the control of underwater robots [19], the dynamics model of the ROV is rewritten as follows.

$$\ddot{\psi} = \frac{1}{I_z - N_{\dot{r}}}\tau_\psi + \frac{\left(N_r + N_{r|r|}|r|\right)r}{I_z - N_{\dot{r}}} + f_\psi \tag{12}$$

Using the dynamic expression of heading angle ψ from Equations (10) and (11) as an example for controller design, we provide a detailed design explanation for LESO and LSEFC.

In Formula (12), $\frac{1}{I_z-N_{\dot{r}}}$ is a constant term whose value is determined by the system's inertia parameter I_z and damping coefficient $N_{\dot{r}}$. To simplify the formula, $b = \frac{1}{I_z-N_{\dot{r}}}$ is used to replace this complex coefficient. The second term $\frac{\left(N_r+N_{r|r|}|r|\right)r}{I_z-N_{\dot{r}}}$ describes the nonlinear disturbances caused by hydrodynamics, which are typically regarded as a type of interference or unmodeled dynamics. In the design of LESO, this part is incorporated into the total uncertainty f_ψ for unified treatment. Therefore, Formula (12) can be simplified to:

$$\ddot{\psi} = b\tau_\psi + f_\psi \tag{13}$$

Taking the state variables ψ_1, ψ_2, ψ_3, among them $\psi_3 = f_\psi$ as the extended state, the ROV heading angle ψ control model can be expressed as:

$$\begin{cases} \psi = \psi_1 \\ \dot{\psi}_1 = \psi_2 \\ \dot{\psi}_2 = b\tau_\psi + \psi_3 \\ \dot{\psi}_3 = D \end{cases} \tag{14}$$

Let $D = \dot{f}_\psi$, $\psi_2 = \dot{\psi}$. A linear expansion state observer can be established for the system (13) [19]:

$$\begin{cases} e_1 = \psi - z_1 \\ \dot{z}_1 = z_2 + \beta_1 e_1 \\ \dot{z}_2 = z_3 + \beta_2 e_1 + b\tau_\psi \\ \dot{z}_3 = \beta_3 e_1 \end{cases} \tag{15}$$

In reference Formula (15), z_1, z_2 represent the estimated values of the state variables of the controlled object (in this example, z_1 represents the observation value of ψ, and z_2 is the observation value of the ψ derivative), while z_3 represents the real-time estimated value of the total disturbance (unknown external disturbances and uncertain models). $\beta_1, \beta_2, \beta_3$ are the gains of the LESO. If the observer gains are chosen appropriately, LESO can achieve precise tracking of each state variable of the controlled object. To facilitate parameter tuning, the values of $\beta_1, \beta_2, \beta_3$ are determined by ω_0. By reasonably selecting the parameter ω_0, the observed value of the "total disturbance" can be made closer to the true value. The Laplace transform of the LESO equation yields [19]:

$$\begin{aligned} z_1 &= \tfrac{\beta_1 s^2 + \beta_2 s + \beta_3}{L'(s)} Y(s) + \tfrac{bs}{L'(s)} U(s) \\ z_2 &= \tfrac{\beta_2 s^2 + \beta_3 s}{L'(s)} Y(s) + \tfrac{bs(s+\beta_1)}{L'(s)} U(s) \\ z_3 &= \tfrac{\beta_3 s^2}{L'(s)} Y(s) - \tfrac{b\beta_3}{L'(s)} U(s) \end{aligned} \tag{16}$$

$Y(S)$ is the Laplace transform of the system output y(t) in the time domain, and $U(S)$ is the Laplace transform of the system input u(t) in the time domain. The characteristic equation corresponding to LESO is [18]:

$$L'\left(s\right) = s^3 + \beta_1 s^2 + \beta_2 s + \beta_3 \tag{17}$$

To stabilize the system, the roots of the characteristic equation must be located in the left half of the s-plane. Therefore, the three poles of the observer are uniformly placed on the left half of the real axis at $-\omega_o$ (where ω_o is the bandwidth of the observer, and $\omega_o > 0$). Therefore, the observer gain can be obtained as $\beta_1 = 3\omega_o$, $\beta_2 = 3\omega_o^2$, $\beta_3 = \omega_o^3$. The estimated error of LESO can be expressed as [18]:

$$\begin{cases} \dot{e}_1 = e_2 - \beta_1 e_1 \\ \dot{e}_2 = e_3 - \beta_2 e_1 \\ \dot{e}_3 = D - \beta_3 e_1 \end{cases} \tag{18}$$

Among them, $e_i = \psi_i - z_i$ $(i = 1, 2, 3)$ provides the conditions for the estimation error of LESO, which will be used for the stability proof of LESO in the following text.

3.2. Linear State Error Feedback Controller

Traditional PID controllers use error integration to eliminate static errors, but the feedback from error integration can make the system prone to oscillation. In contrast, the LESO (Linear Extended State Observer) employs real-time compensation for total disturbances, avoiding the negative effects of integral feedback. The result is shown in Equation (19) [18]:

$$\begin{cases} e_1 = \psi_d - z_1 \\ e_2 = \dot{\psi}_d - z_2 \\ u_0 = k_p e_1 + k_d e_2 \end{cases} \tag{19}$$

In the equation, ψ_d is the reference heading angle input, u_0 is the error feedback control quantity, and k_p, k_d are the controller gains. According to Equation (19), the transfer function of u_0 concerning r can be obtained:

$$\frac{U_0(s)}{R(s)} = \frac{k_p + k_d s}{s^2 + k_d s + k_p} \tag{20}$$

By setting both poles of the controller on the real axis at $-\omega_c$ (where ω_c represents the control bandwidth) in the left half of the s-plane, the controller gain can be determined as [18].

$$\begin{cases} k_p = \omega_c^2 \\ k_d = 2\omega_c \end{cases} \tag{21}$$

Based on u_0, an additional compensation term for the total disturbance estimate is added, so the control quantity can be taken as:

$$\tau_\psi = \frac{-z_3 + u_0}{b_0} \tag{22}$$

In Formula (22), u_0 is the error feedback control quantity, and τ_ψ is the actual input of the controlled object.

4. DDPG Optimization of Control Parameters for Active Disturbance Rejection Controller

For the controlled system, when the range of unknown disturbances is too large and the rate of change is too fast, using a fixed-parameter active disturbance rejection controller results in low control accuracy. Therefore, by combining deep reinforcement learning, an improved active disturbance rejection controller has been designed, in which the active disturbance rejection control parameters will vary with the environment.

4.1. Deep Deterministic Policy Gradient-Active Disturbance Rejection Controller Algorithm Framework

Reinforcement learning is an algorithm that allows an agent to adjust its behavioral strategies based on observations made during interactions with the environment, to maximize cumulative rewards. The schematic diagram is shown in Figure 3.

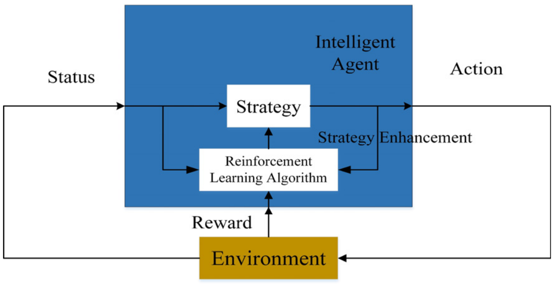


Figure 3. Reinforcement learning diagram.

In response to the online adjustment problem of fixed parameter active disturbance rejection controllers, this paper employs the Deep Deterministic Policy Gradient algorithm (DDPG), which can handle continuous action control. The proposed control strategy, DDPG-LADRC, treats the entire underwater robot control system as the environment, using the system's control performance as the reward evaluation criterion. The DDPG-LADRC agent determines actions based on the current environment, and then the environment provides a new state based on the output value and calculates the reward value. The DDPG-LADRC agent makes judgments, optimizes and updates the next action, and interacts with the environment until the reward converges.

4.2. Deep Deterministic Policy Gradient Algorithm Principles

DDPG is a deep reinforcement learning algorithm based on the actor-critic framework. The actor-network outputs deterministic actions in a continuous action space based on environmental state feedback, while the critic-network calculates the corresponding *Q*-value based on the current state and action, which is used to evaluate the long-term expected return of the action. By adjusting the weights of the critic network according to the error between the reward output by the critic network and the actual received reward, the output estimates of the critic network can become more accurate. Using the policy gradient algorithm, the parameters of the actor-network are updated in the direction of increasing the action value. During the interaction between the agent and the environment, the learning parameters of both networks will be continuously updated until the policy converges [27].

At time t, the mapping from state s to action a is referred to as policy π.

$$a_t = \pi(s_t) \tag{23}$$

According to the actions generated by strategy π, new states and reward values r are continuously obtained. The formula for calculating cumulative rewards is:

$$G_t = \sum_{t=1}^{T} \gamma^t r^t \tag{24}$$

The Bellman equation for the state value function is represented as:

$$V_\pi(s) = E_\pi[\textstyle\sum_{k=0}^{\infty} G_t | s = s_t] = E_\pi[r_{t+1} + \gamma V_\pi(s_{t+1}) | s = s_t] \tag{25}$$

Considering the impact of actions on the value function, the Bellman equation for the state-action value function is represented as:

$$Q_\pi(s, a) = E_\pi[r_{t+1} + \gamma Q_\pi(s_{t+1}, a_{t+1}) | s = s_t, a = a_t] \tag{26}$$

The optimal Bellman equation can be expressed as:

$$Q^*(s_t, a_t) = r_{s_t}^{a_t} + \gamma Q_\pi^*(s_{t+1}, a_{t+1}) \tag{27}$$

The optimal strategy π^* is obtained by maximizing the cumulative reward and its corresponding optimal Bellman equation:

$$\pi(s) \to a_t = argmax Q_\pi^*(s_t, a_t) \tag{28}$$

We calculate the loss function for the target Q value, using y_t to represent it:

$$\begin{cases} y_t = r_t + \gamma Q'\left(s_{t+1}, \pi'\left(s_{t+1} \mid \theta^{\pi'}\right) \mid \theta^{Q'}\right) \\ L\left(\theta^Q\right) = E\left(y_t - Q\left(s_t, a_t \mid \theta^Q\right)\right)^2 \end{cases} \tag{29}$$

By calculating the gradient of the loss function, we update the current value network [28].

$$\begin{cases} \theta_k^Q = \theta_{k-1}^Q - \mu_Q \nabla_{\theta^Q} L\left(\theta_{k-1}^Q\right) \\ \nabla_{\theta^Q} L\left(\theta_{k-1}^Q\right) = E\left(2\left(y_t - Q\left(s, a \middle| \theta_{k-1}^Q\right) |_{s=s_t, a=a_t}\right)\right. \\ \nabla Q_{\theta^Q}\left(s, a \middle| \theta_{k-1}^Q\right) |_{s=s_t, a=a_t}) \end{cases} \tag{30}$$

The strategy network uses the Q function output from the value network as the loss function. By taking the policy gradient of the Q function, the update formula is obtained [28].

$$\begin{cases} \theta_k^\pi = \theta_{k-1}^\pi - \mu_\pi \nabla_{\theta^\pi} L\left(\theta_{k-1}^\pi\right) \\ \nabla_{\theta_{k-1}^\pi} \pi = \nabla_a Q\left(s, a \mid \theta_{k-1}^\pi\right) \\ |_{s=s_t, a_t=\pi(s_t)} \nabla_{\theta^\pi} \pi\left(s \mid \theta_{k-1}^\pi\right) |_{s=s_t}) \end{cases} \tag{31}$$

The target network uses a soft update method as follows [29].

$$\begin{cases} \theta_k^{Q'} = \tau \theta_{k-1}^Q + (1-\tau)\theta_{k-1}^{Q'} \\ \theta_k^{\pi'} = \tau \theta_{k-1}^\pi + (1-\tau)\theta_{k-1}^{\pi'} \end{cases} \tag{32}$$

The DDPG algorithm is a deterministic policy that adds noise to the deterministic policy, as shown in Equation (33), allowing the agent to explore the environment more effectively and preventing it from getting stuck in local optima. The deterministic policy gradient helps the critic converge and updates the network parameters [29]. The meanings of the various parameters in the above analysis are shown in Table 1.

$$a_t = \mu_\theta(s_t) + N \tag{33}$$

Table 1. DDPG algorithm parameter meaning.

Algorithm Parameters	Meaning
$Q\left(s_t, a_t \mid \theta^Q\right)$	The Q value output by the current value network at time t
$Q'\left(s_{t+1}, \pi'\left(s_{t+1} \mid \theta^{\mu'}\right) \mid \theta^{Q'}\right)$	Input Q value of the target network
$\pi'\left(S_{t+1} \mid \theta^{\pi'}\right)$	Action variables output by the target strategy network
θ_k^Q, θ_k^π	The parameters of the network at the k round of learning
μ_Q	Learning rate of value networks
$\nabla_{\theta^Q} L\left(\theta_{k-1}^Q\right)$	A gradient of the loss function concerning the parameters
μ_π	Learning rate of the strategy network
θ_{k-1}^π	Strategy gradient
$\theta_k^{Q'}, \theta_k^{\pi'}$	The parameters of the target network at the k learning iteration
τ	Soft update coefficient

The structure of the DDPG algorithm model is shown in Figure 4. The DDPG agent stores the sample data obtained from interacting with the LADRC control system in the experience pool. During the learning process, it randomly samples m pieces of data from the experience pool and continuously iterates to update the network gradient values to optimize the algorithm [29,30].

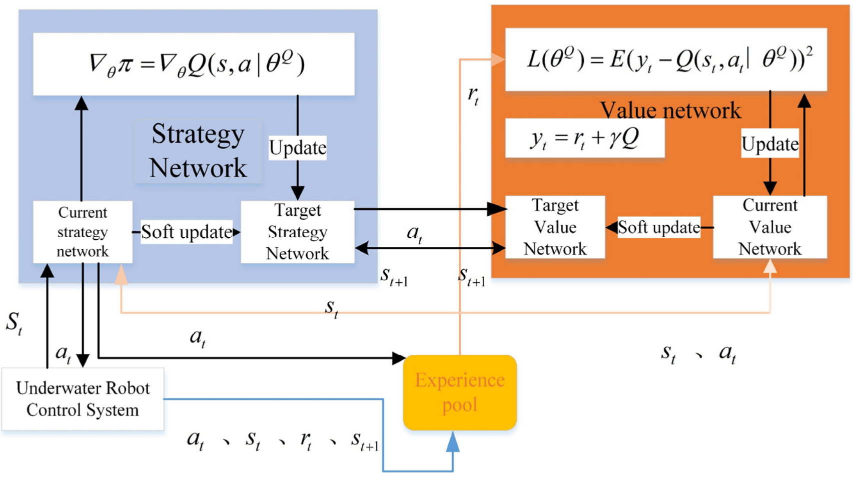


Figure 4. DDPG algorithm framework diagram.

Combining deep reinforcement learning with an active disturbance rejection controller, the control system is designed to obtain environmental state data through the interaction between the agent and the environment (underwater robot control system). Here, LSEFC represents the state error feedback controller, and LESO represents the linear expanded observer. The structural block diagram of the active disturbance rejection controller based on deep reinforcement learning is shown in Figure 5.

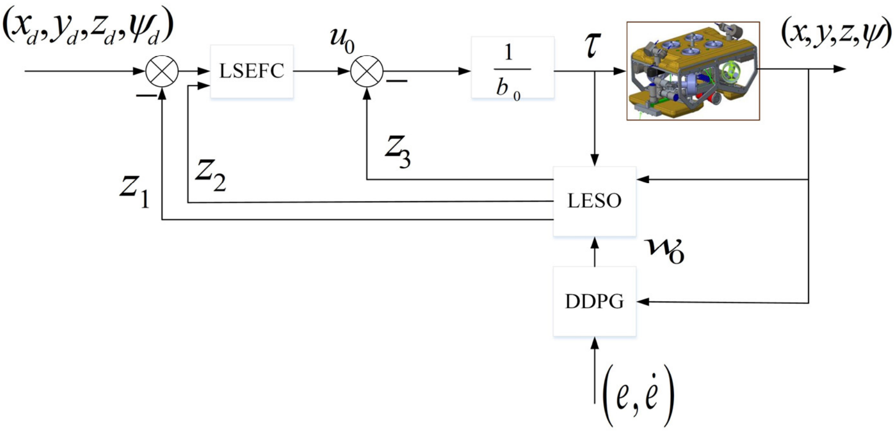


Figure 5. Block diagram of a self-disturbance rejection controller based on reinforcement learning.

According to the control system structure block diagram designed in the above figure, we set the various parameters of the deep reinforcement learning agent.

If the LADRC control has the Markov property, then when optimizing with DDPG, the future state transitions of the system depend only on the current state and control actions, without the need to explicitly model the state transition probabilities. The optimization problem of the LESO observation capability is modeled as a reinforcement learning task in a continuous action space. The DDPG algorithm is used to dynamically adjust the key parameter w_0 of the LESO, enabling it to adapt to changes in external disturbances, thereby improving the disturbance estimation accuracy of the LESO and the robustness of the controller. The actor network is responsible for generating the adjustment of the LESO observer bandwidth w_0, based on the current state. The critic network outputs the action value Q based on the current state and the action generated by the actor network, guiding the policy update of the actor network. The ROV studied in this paper is under umbilical cable control, effectively avoiding the issue of limited resources for the ROV, and the ROV's controller can meet the high computational demands of DDPG training.

For the state space, select $(e, \dot{e})$, corresponding to the errors (e_x, e_y, e_z, e_ψ), and the differential of the error in each degree of freedom.

For the action space, select the poles of the expanded state observer, that is w_0, $\beta_1 = 3\omega_o, \beta_2 = 3\omega_o^2, \beta_3 = \omega_o^3$.

To reduce the final error, a reward function is set based on the error between the output of each degree of freedom and the expected value: $R = -\left[\left(\sqrt{{e_x}^2 + {e_y}^2 + {e_z}^2 + {e_\psi}^2}\right)\right]$.

In terms of the discount factor, the degree to which future rewards influence current decisions is determined, and in this article, the chosen discount factor γ = 0.98 is used to ensure that accurate trajectory tracking is given high priority. After multiple adjustments, the final selected DDPG parameters are shown in Table 2.

Table 2. DDPG algorithm parameters.

Hyperparameter	Value
Actor-network learning rate	0.001
Critics' online learning rate	0.0005
Small batch sampling sample size	64
Discount factor	0.98
Noise variance	0.2
Noise attenuation coefficient	0.00001
Experience pool size	100,000

The reward curve of the Deep Deterministic Policy Gradient (DDPG) algorithm is generally used to determine whether the agent's training has converged. The curve showing the change in rewards after training over the training iterations is shown in Figure 6.

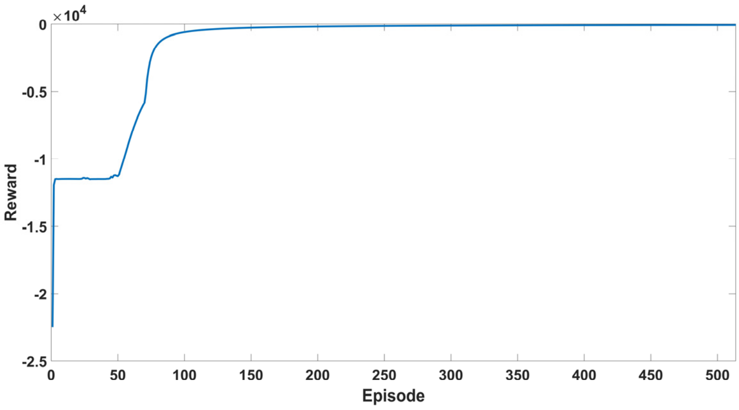


Figure 6. Training reward change curve.

4.3. Stability Analysis

To conduct the analysis, the following hypothesis is proposed based on engineering practice: the total disturbance observed by the observer in the self-disturbance rejection control is bounded within H, $H = \{D||D| \leqslant F_h\}$, where F_h is a positive constant.

Theorem 1. *The estimation error of the observer constructed in Equation (15) is bounded [18].* $\underset{\omega_0 \to \infty, t \to \infty}{lim} ||e|| = 0$.

Proof. Let $q_i = \frac{e_i}{\omega_o^i} (i = 1, 2, 3)$, then Equation (18) can be rewritten as:

$$\begin{cases} \dot{q}_1 = \omega_o(q_2 - 3q_1) \\ \dot{q}_2 = \omega_o(q_3 - 3q_1) \\ \dot{q}_3 = \omega_o\left(\frac{D}{\omega_o^4} - q_1\right) \end{cases} \tag{34}$$

The reference Formula (34) can be rewritten as:

$$\dot{\eta} = \omega_o A q + B\frac{D}{\omega_o^3} \tag{35}$$

For simplicity, let $q = [q_1, q_2, q_3]^T$, therefore:

$$A = \begin{bmatrix} -3 & 1 & 0 \\ -3 & 0 & 1 \\ -1 & 0 & 0 \end{bmatrix},\ B = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \tag{36}$$

Observing the above equation, for any positive ω_o, A is Hurwitz, therefore, there exists a unique positive definite symmetric matrix P_η that satisfies the Lyapunov equation $A^T P_\eta + P_\eta A = -Q_\eta$. By choosing the Lyapunov function as $V(q) = q^T P_\eta q$, we can derive the following for $V(q)$:

$$\begin{aligned} &\dot{V}(q) = q^T P_\eta \dot{q} + \dot{q}^T P_\eta q \\ &= q^T P_\eta (\omega_0 A q + B\tfrac{D}{\omega_0^3}) + (\omega_0 A q + B\tfrac{D}{\omega_0^3})^T P_\eta q \\ &= -\omega_0 q^T Q_\eta q + 2q^T P_\eta B\tfrac{D}{\omega_0^3} \end{aligned} \tag{37}$$

Referring to Equation (37), using the Cauchy inequality and the property of the minimum eigenvalue of positive definite matrices, $\dot{V}(q)$ can be rewritten as:

$$\begin{aligned} &\dot{V}(q) = -\omega_0 q^T Q_\eta q + 2q^T P_\eta B\tfrac{D}{\omega_o^3} \\ &\leq \omega_o \lambda_{min}(Q_\eta) \parallel q \parallel^2 + \tfrac{2F_h \lambda_{max}(P_\eta)\|q\|}{\omega_o^3} \end{aligned} \tag{38}$$

Using the eigenvalues of the matrix, we can obtain the following bounds on the quadratic form: $\lambda_{\min}(P_\eta) \parallel q \parallel^2 \leq q^T P_\eta q \leq \lambda_{\max}(P_\eta) \parallel q \parallel^2$. This can be rewritten as: $\frac{V(q)}{\lambda_{\max}(P_\eta)} \leq \parallel q \parallel^2 \leq \frac{V(q)}{\lambda_{\min}(P_\eta)}$. Therefore, inequality (38) can be rewritten as:

$$\begin{aligned} &\dot{V}(q) \leq \omega_o \tfrac{\lambda_{min}(Q_\eta)}{\lambda_{max}(P_\eta)} V(q) \\ &+ \tfrac{2F_h \lambda_{max}(P_\eta)}{\omega_o^3 \sqrt{\lambda_{min}(P_\eta)}} \sqrt{V(q)} \end{aligned} \tag{39}$$

To obtain the linear differential inequality, let $W = \sqrt{V(q)}$, then $\dot{W} = \frac{\dot{V}(q)}{2\sqrt{V(q)}}$ can be obtained, and inequality (39) can be rewritten as:

$$\dot{W} \leq \omega_o \frac{\lambda_{min}(Q_\eta)}{2\lambda_{max}(P_\eta)} W + \frac{F_h \lambda_{max}(P_\eta)}{\omega_o^3 \sqrt{\lambda_{min}(P_\eta)}} \tag{40}$$

When studying the state equation $\dot{W}$, it is often necessary to obtain the boundary of its solution W, rather than the solution itself. The Gronwall–Bellman method is one of the approaches used for this purpose [31]. First, let $\beta = -w_0 \frac{\lambda min(Q_\eta)}{2\lambda max(P_\eta)}, \alpha = \frac{F_h \lambda_{max}(P_\eta)}{\omega_o^3 \sqrt{\lambda_{min}(P_\eta)}}$. By applying the inequality, we can obtain the following.

Assume $\dot{W} = \beta W + \alpha$. Thus, there is $W \leq (W(t_0) - \frac{\alpha}{\beta})e^{\beta(t-t_0)} + \frac{\alpha}{\beta}$. By organizing, we can obtain the following expression:

$$\begin{aligned} W \leq (&\frac{2F_h\lambda_{\max}^2(P_\eta)}{\omega_o^4\sqrt{\lambda_{\min}(P_\eta)}\lambda_{\min}(Q_\eta)} - W(t_0))e^{-\omega_0\frac{\lambda_{\min}(Q_\eta)}{2\lambda_{\max}(P_\eta)}(t-t_0)} \\ &+ \frac{2F_h\lambda_{\max}^2(P_\eta)}{\omega_o^4\sqrt{\lambda_{\min}(P_\eta)}\lambda_{\min}(Q_\eta)} \end{aligned} \tag{41}$$

From $W = \sqrt{V(q)} = \sqrt{q^T P_\eta q}$, $\frac{V(q)}{\lambda_{max}(P_n)} \leq \parallel q \parallel^2 \leq \frac{V(q)}{\lambda_{min}(P_n)}$, we can obtain Equation (42):

$$\parallel q \parallel \leq \frac{\sqrt{V}}{\sqrt{\lambda_{min}(P_\eta)}} = \frac{W}{\sqrt{\lambda_{min}(P_\eta)}} \tag{42}$$

When $t \to \infty$, the Expression (43) can be obtained:

$$\begin{aligned} \parallel q \parallel &\leq \frac{\sqrt{V(q)}}{\sqrt{\lambda_{min}(P_\eta)}} \\ &\leq \frac{2F_h\lambda_{max}^2(P_\eta)}{\omega_o^4\lambda_{min}(P_\eta)\lambda_{min}(Q_\eta)} \\ &= \frac{k}{\omega_0^4} \end{aligned} \tag{43}$$

In Formula (43), $k = \frac{2F_h\lambda_{max}^2(P_\eta)}{\lambda_{min}(P_\eta)\lambda_{min}(Q_\eta)}$ is a normal constant, and because P_η, Q_η are independent of ω_0, the Equation (43) can demonstrate that $\lim\limits_{\omega_0\to\infty,t\to\infty}||q|| = 0$, which, together with $q_i = \frac{e_i}{\omega_o^i}(i = 1,2,3)$, is produced. Therefore, $\lim\limits_{\omega_0\to\infty,t\to\infty}||e|| \leqslant ||q||\omega_0^3 \leqslant \frac{k}{\omega_o} = 0$, thus completing the proof. Define H_e as $H_e = \{e|||e||\leqslant E\}$, where E is a positive constant. By adjusting ω_0 to ensure $\frac{k}{\omega_o} \leqslant E$, the estimated error of LESO will remain within H_e, Theorem 1 has been proven. □

Theorem 2. *According to the error feedback control law given by Equation (19), we can ensure the closed-loop stability of the control system. According to Theorem 1, the convergence of LESO can be guaranteed by carefully selecting* ω_0,*b, and the estimation error of LESO will be constrained within* H_e. *Translate Equation (19) into the Equation (14) to obtain:*

$$\begin{cases} \dot{e}_{\psi_1} = e_{\psi_2} \\ \dot{e}_{\psi_2} = -k_p e_{\psi_1} - k_d e_{\psi_2} + k_p e_1 + k_d e_2 + e_3 \end{cases} \tag{44}$$

Referencing Equation (44), $e_{\psi 1} = \psi_r - \psi_1$ and $e_{\psi 2} = \dot{\psi}_r - \psi_2$ are defined as tracking errors. The above equation can be rewritten as:

$$\begin{gathered} \dot{e}_\psi = Ce_\psi + Ge \\ e_\psi = \left[e_{\psi 1}, e_{\psi 2}\right]^T, C = \begin{bmatrix} 0 & 1 \\ -k_p & -k_d \end{bmatrix} \\ G = \begin{bmatrix} k_p & k_d & 1 \end{bmatrix} \end{gathered} \tag{45}$$

Since the controller output is positive, ensure that $k_p, k_d \geqslant 0$. Then, the characteristic roots of C can be expressed as:

$$\lambda_{1,2} = -\frac{k_d}{2} \pm \sqrt{-k_p + \frac{k_d^2}{4}} \tag{46}$$

Therefore, the designed linear active disturbance rejection controller is stable, and Theorem 2 has been proven. The tracking error of the observer is also bounded.

5. Simulation Analysis

The underwater intelligent cleaning and inspection robot is specifically designed for the safety inspection of marine oil platform risers and the removal of marine organisms attached to the risers. It is equipped with an Ultra-Short Baseline positioning system (USBL), an attitude sensor, a depth sensor, and a compass, enabling precise positioning, attitude awareness, and depth perception. In addition, its propulsion system includes four horizontal thrusters and four vertical thrusters. The model parameters of the underwater robot are shown in Table 3, and the physical prototype and thruster layout are illustrated in Figure 7.

Table 3. ROV model parameters.

Parameter	Values	Parameter	Values
m	197 kg	$Y_{v\|v\|}$	$-245.2\ \mathrm{N^2/m^2}$
I_z	25.1 $\mathrm{Nms^2}$	Z_w	-12.6 N/S
X_u	-5.24 N/S	$Z_{\dot{w}}$	-367.8 kg
$X_{\dot{u}}$	-135.1 kg	$Z_{w\|w\|}$	$-547.4\ \mathrm{N^2/m^2}$
$X_{u\|u\|}$	$-109.1\ \mathrm{N^2/m^2}$	N_r	-1.52 N/S
Y_v	-11.1 N/S	$N_{\dot{r}}$	-34.3 kg
$Y_{\dot{v}}$	-390.6 kg	$N_{r\|r\|}$	$-26.2\ \mathrm{N^2/m^2}$

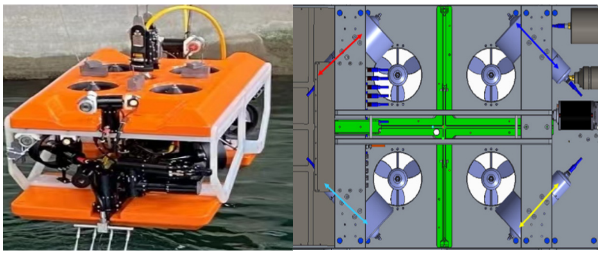


Figure 7. Underwater robot (ROV) physical prototype, 3D arrangement of thrusters.

To verify that DDPG-LADRC has stronger robustness, this paper proposes two experimental simulation scenarios.

(a) Section 5.1 introduces a simple time-varying external disturbance, and the tracked trajectory is also relatively simple, to evaluate the improvement of the DDPG-LADRC control strategy on the transient performance during the motion of the ROV.

(b) The time-varying disturbances introduced in Section 5.2 are related to the motion state of the ROV and track different trajectories, aiming to verify that the DDPG-LADRC control strategy has stronger robustness when the ROV is in a dynamic marine environment.

5.1. Scenario 1

To verify the enhanced effect of combining reinforcement learning DDPG with a linear active disturbance rejection controller in terms of disturbance suppression capability and control accuracy, the position and attitude of the underwater robot are tracked under time-varying external disturbances. The transient performance of the control system under perturbations is evaluated to validate the disturbance rejection and robustness of the DDPG-

LADRC control scheme. Disturbances are introduced during the movement of the ROV as follows:

$$\begin{aligned} f &= \left[f_x, f_y, f_z, f_\psi\right]^T \\ &= [20sin(0.4*t)\ 20\ sin(0.4*t) \\ &\quad 20\ sin(0.4*t)\ \ 20\ sin(0.4*t)]^T \end{aligned} \tag{47}$$

The initial conditions for the underwater robot are set as $[X(0), Y(0), Z(0), Phi(0)] = 0$, with the velocity and angular velocity set as $u(0) = v(0) = w(0) = r(0) = 0$. Additionally, for the controller parameters, the PID parameters are set as:

$$\begin{cases} K_p = \{150, 150, 300, 370\} \\ K_i = \{15, 15, 60, 15\} \\ K_d = \{300, 300, 150, 300\} \end{cases} \tag{48}$$

The parameters for the Active Disturbance Rejection Control are set as follows: $b_0 = 10$, $w_0 = 5$, because $\beta_1 = 3\omega_o$, $\beta_2 = 3\omega_o^2, \beta_3 = \omega_o^3$ which means β_1 = 15, $\beta_2 = 75$, $\beta_3 = 125$. The relevant DDPG setting parameters are shown in Table 2 above. The underwater robot simulation is designed to run for 100 s, with a simulation step size of 0.01 s. The proposed control algorithm is mainly compared with PID and LADRC under fixed parameters through three-dimensional trajectory tracking, and planar tracking, to verify the degree of improvement in the system's transient performance by the DDPG-LADRC control strategy. The trajectory tracking curve in the inertial coordinate system is:

$$\begin{cases} x_d = 2sin(0.1\pi t)\ \text{m} \\ y_d = 2cos(0.1\pi t)\ \text{m} \\ z_d = 0.2t\ \text{m} \\ \psi_d = 0.03\pi t\ \text{rad} \end{cases} \tag{49}$$

First, a feasibility analysis of the parameter optimization for LESO is conducted. Figure 8 compares the observation errors of the LESO optimized by DDPG with those of the fixed-parameter LESO. It can be observed that the fixed-parameter LADRC controller is not precise in tracking total disturbances. In contrast, the DDPG-LADRC can maintain better performance with a shorter time under the constraints of model parameter uncertainty and strong unknown external disturbances in underwater robot trajectory tracking control. The DDPG-LADRC can quickly respond to changes in disturbances and adjust its control strategy promptly to adapt to these changes, thereby enhancing the system's dynamic performance. This indicates that the optimized observer parameters of DDPG-LADRC are effective.

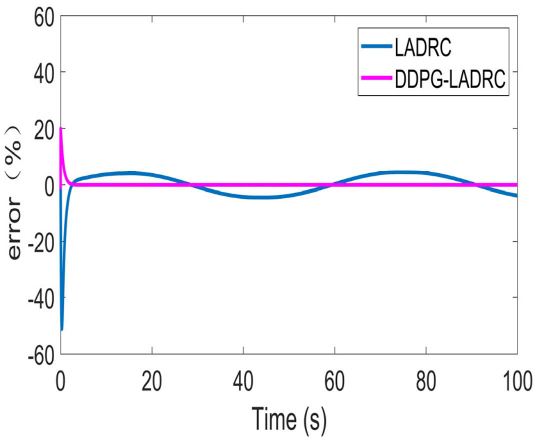
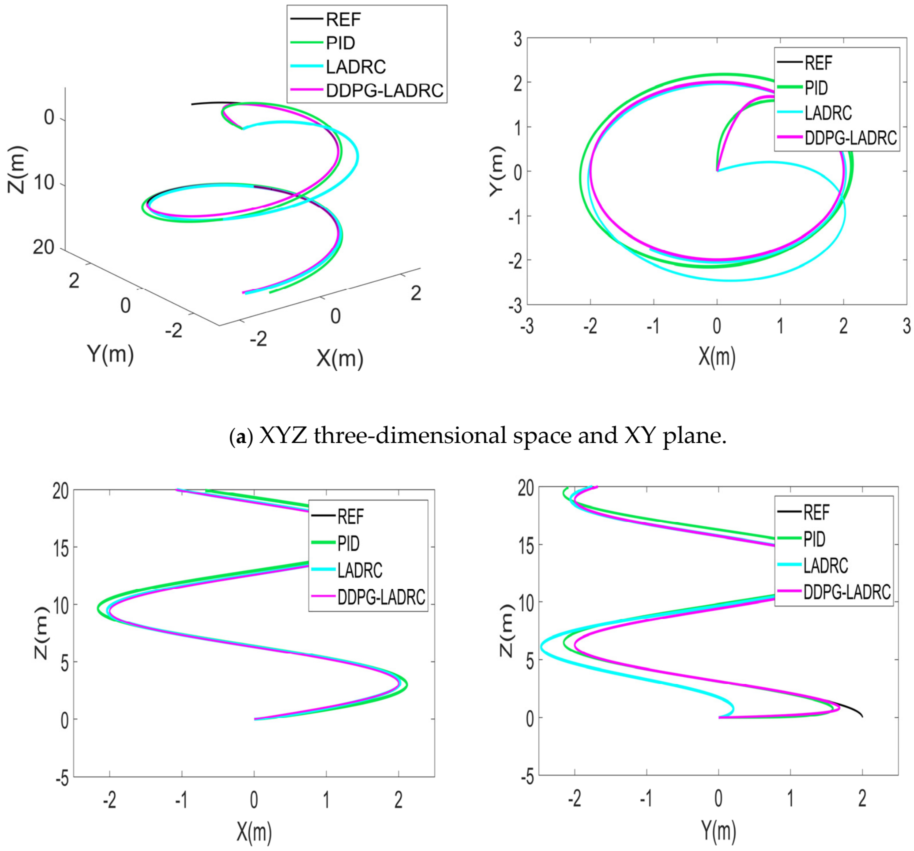


Figure 8. Comparison of total disturbance observation errors of two types of observers.

The three-dimensional trajectory tracking performance of the ROV under different control schemes, as well as the tracking curves in the XY, XZ, and YZ planes shown in Figure 9, can be observed. It can be seen that even in the presence of disturbances, the DDPG-LADRC control scheme can achieve precise trajectory tracking, with control performance superior to that of the PID controller and the fixed parameter LADRC controller, demonstrating stronger robustness. Therefore, parameter optimization based on DDPG can enhance the control performance of LADRC.



(**a**) XYZ three-dimensional space and XY plane.

(**b**) XZ plane and YZ plane.

Figure 9. Trajectory tracking results of the ROV under three control schemes.

The selected evaluation indicators for transient performance are overshoot, settling time, and peak time.

In underwater robot control, overshoot is an important indicator used to describe the dynamic performance of a system. Overshoot is typically measured by the difference between the maximum output value and the steady-state value, and it can also be expressed as a percentage of this difference relative to the steady-state value. The system without overshoot typically stabilizes at the setpoint without deviating too much from the target value, indicating that there is no significant overreaction or oscillation during the response process. From Table 4, we can see that DDPG-LADRC maintains response speed without overshoot, while PID and LADRC exhibit overshoot. When the overshoot is too large, the control system is prone to oscillation. The results indicate that DDPG-LADRC ensures the dynamic response process of the system, maintaining high robustness even in the face of model uncertainty or external disturbances. The parameter optimization effect of DDPG-LADRC is evident, effectively meeting the dynamic performance requirements of the system.

Table 4. Comparison of transient performance of different control methods: overshoot.

Comparison	*X*	*Y*	*Z*	*Phi*
PID	6%	12.5%	10%	5%
LADRC	2%	25%	5%	0
DDPG-LADRC	0	0	0	0

In underwater robot trajectory tracking control, the adjustment time is an important dynamic performance indicator. It reflects the robot's sensitivity to changes in control signals and its ability to respond quickly, defined as the time required for the ROV to respond and maintain within a certain allowable error range (usually ±2% or ±5% of the final value) after initially reaching the target value. A shorter adjustment time means that the ROV can stabilize more quickly around the target value, reducing oscillations or instability during the transition process. Additionally, a rapid response can better handle external disturbances and changes in the internal parameters of the ROV, enhancing the system's robustness and stability. Referring to Table 5, it can be seen that the adjustment time of DDPG-LADRC for the ROV in the X-direction is significantly better than the other two control strategies, reducing by 93% and 98%, respectively. In the Y-direction, the reductions are 93% and 86%, respectively, and in the Z-direction, the reductions are 66.7% and 90%, respectively. The attitude angles *Phi* were reduced by 64% and 89%, respectively.

Table 5. Comparison of transient performance of different control methods: settling time/s.

Comparison	*X*	*Y*	*Z*	*Phi*
PID	50	71	10	11
LADRC	14	40	3	35
DDPG-LADRC	1	5	1	4

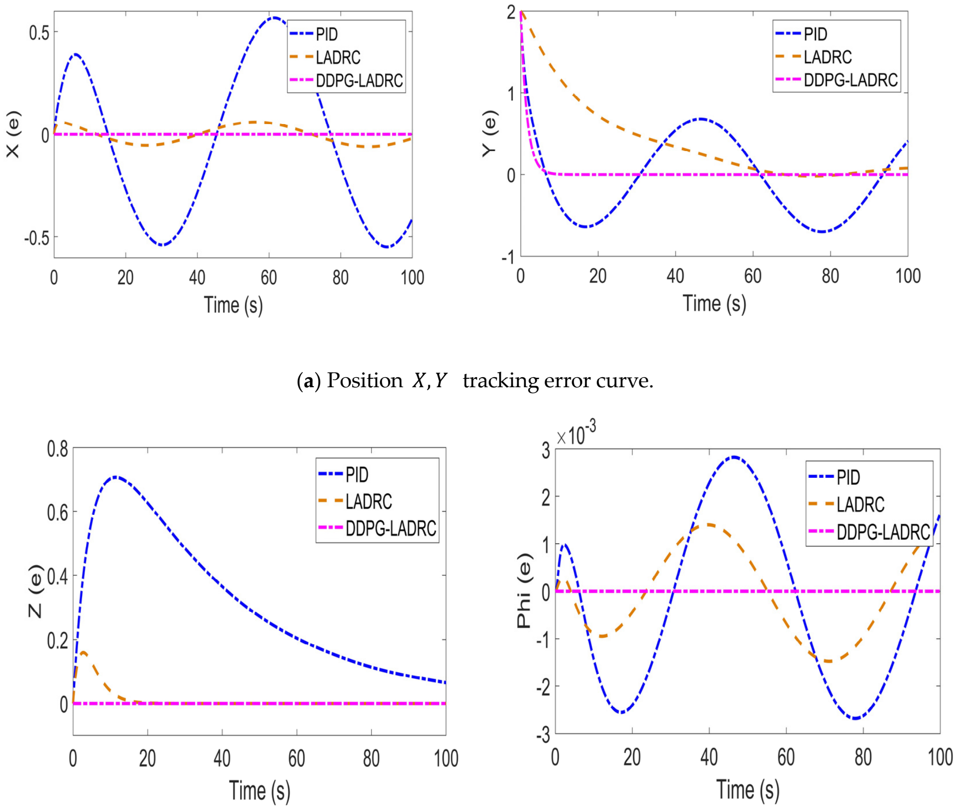
Even if the overshoot is 0, the system response may still have a "peak," which does not refer to a deviation exceeding the steady-state value, but rather to the maximum value during the response process. In the underwater robot trajectory tracking control system, the peak time is an important dynamic performance indicator that describes the time required for the system response to exceed its steady-state value and reach the first peak. Referring to Table 6, the comparison of peak times shows that in the X-direction, DDPG-LADRC significantly outperforms the other two control strategies, reducing by 93% and 98%, respectively. In the Y-direction, it reduces by 82% and 90%, respectively, and in the Z-direction, it reduces by 80% and 98%, respectively. The attitude angles *Phi* are reduced by 93% and 89%, respectively.

Table 6. Comparison of transient performance of different control methods: peak time/s.

Comparison	*X*	*Y*	*Z*	*Phi*
PID	50	34	5	61
LADRC	15	62	50	35
DDPG-LADRC	1	6	1	4

In summary, through a comparative analysis of transient performance under different control methods, the results indicate the superiority of the DDPG-LADRC control strategy in terms of transient performance. Compared to PID controllers and traditional LADRC controllers, the proposed DDPG-LADRC is more suitable for underwater robotic systems that are multivariable, strongly coupled, have significant randomness, and are subject to unknown disturbances.

The tracking error of the ROV trajectory tracking in Figure 9 is shown in Figure 10. Compared to the PID controller and the fixed parameter LADRC controller, the proposed DDPG-LADRC controller has a smaller steady-state error. The PID and fixed-parameter LADRC control schemes are unable to eliminate steady-state errors in a short time, which leads to an inability to track the desired trajectory. However, the DDPG-LADRC significantly improves the control accuracy of the system by introducing DDPG to achieve online tuning of LADRC parameters in response to environmental changes. This ensures that the ROV can maintain satisfactory control performance even in the presence of inaccurate model parameters and significant uncertain disturbances.



(**a**) Position X, Y tracking error curve.

(**b**) Position Z, attitude angle Phi tracking error curve.

Figure 10. Comparison of tracking errors of different control methods for ROV.

After 60 s, data from 1000 sampling points should be collected to calculate the root mean square error for determining the stable accuracy of the control method, as presented in Table 7.

Table 7. Stable accuracy (X, Y, Z = m, *Phi* = rad).

Comparison	X	Y	Z	*Phi*
PID	0.351	0.497	0.381	0.004
LADRC	0.0285	0.3587	1.43×10^{-5}	0.003
DDPG-LADRC	5.43×10^{-5}	1.14×10^{-4}	1.29×10^{-14}	6.56×10^{-9}

In underwater robot control, better stability accuracy means that the robot can precisely reach the target position. Simulation results indicate that the designed DDPG-LADRC controller not only has robust performance but also possesses the ability to quickly track commands and suppress disturbances. Further comparisons show that the performance of

DDPG-LADRC surpasses that of PID and conventional fixed-parameter LADRC. Therefore, parameter optimization based on DDPG can enhance the control performance of LADRC.

5.2. Scenario 2

To further verify the robustness of the controller, the anti-interference capability of different control methods under strong interference conditions was compared. The most representative tracking trajectory during the ROV's motion was selected (Formula (51)). A dual closed-loop sliding mode control scheme based on a nonlinear extended state observer (NESO-DSMC) was added for the comparison of control methods [26], to validate the superiority of the DDPG-LADRC controller's performance.

The parameters for the Active Disturbance Rejection Control are set as follows: $b_0 = 10$, $w_0 = 5$, Because $\beta_1 = 3\omega_o,\ \beta_2 = 3\omega_o^2,\ \beta_3 = \omega_o^3$ which means $\beta_1 = 15$, $\beta_2 = 75$, $\beta_3 = 125$. The relevant DDPG setting parameters are shown in Table 2 above. In addition, the controller parameters proposed in the NESO-DSMC are chosen as follows: $\delta = 0.01$, $\epsilon_{11} = \epsilon_{21} = \epsilon_{31} = 0.5,\ \beta = 0.1,\ \rho_{11} = 100,\ \rho_{21} = 300,\ \rho_{31} = 1000,\ \gamma_1 = \gamma_2 = \gamma_3 = \gamma_4 = 0.01,$ $m = 5,\ q = 2,\ K_\eta = diag\{0.3,\ 0.3,\ 0.3,\ 0.3\}, K_\nu = \mathrm{diag}\{10,\ 10,\ 10,\ 10\}$ [26].

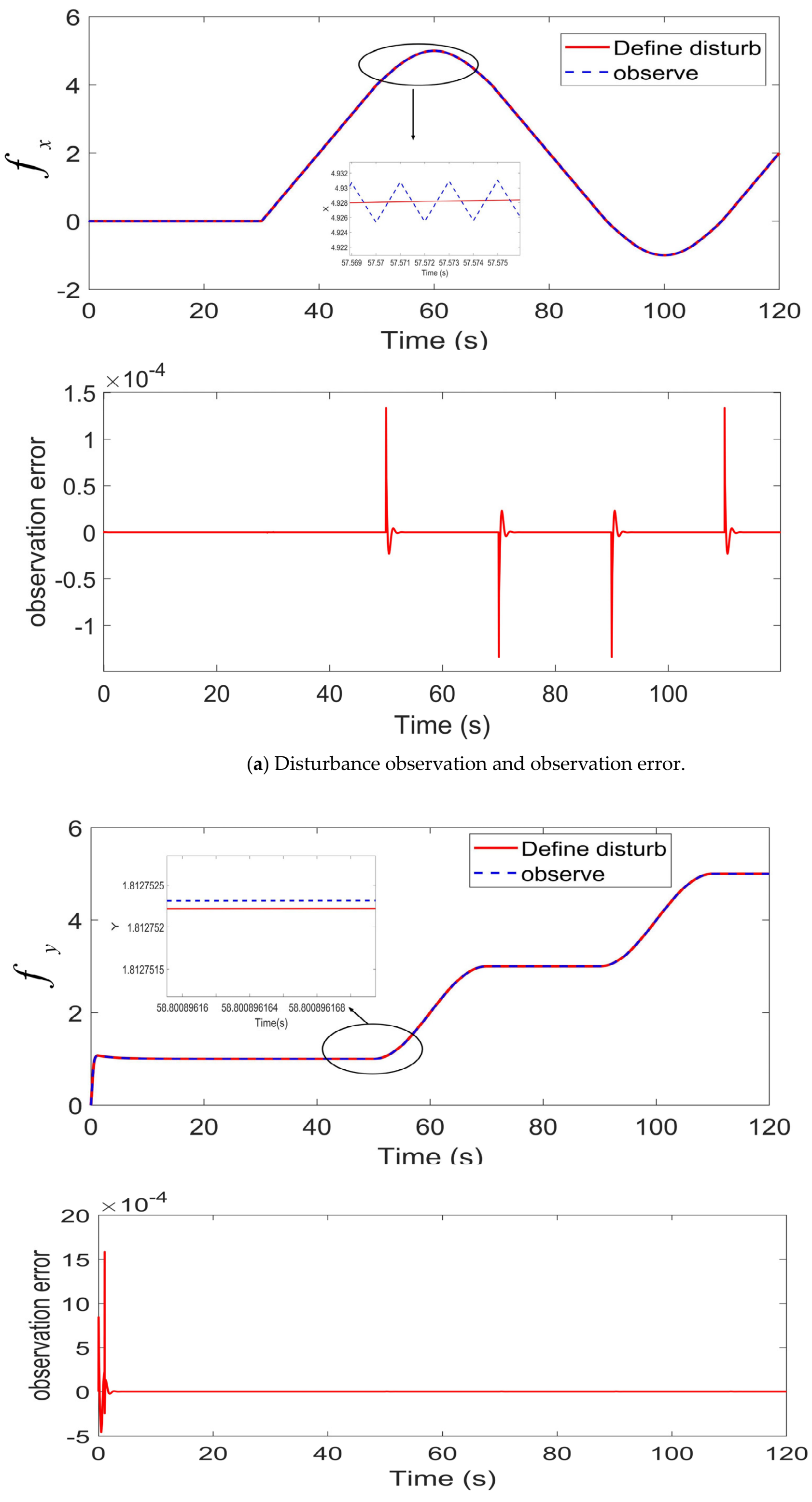
The external interference added is shown in Equation (50). The added disturbance signal is related to the state of the ROV, and this signal is constantly changing.

$$\begin{cases} f_x = 40 - 1.65X - 0.3Y^2 - 1.22Z^2 - 8Z \\ f_y = 2.2X^2 - 2.5Y + 0.3Z \\ f_z = 18 - 2.1X^2 - 0.88Y^2 - 0.5Z^2 \end{cases} \tag{50}$$

The tracked trajectory is shown in Formula (51). This trajectory indicates that the ROV first descends vertically, then performs linear back-and-forth and spiral movements on a horizontal plane, accompanied by changes in depth and adjustments in heading, ultimately returning to a horizontal straight path. The initial position and attitude of the ROV are set as: $X(0) = 0\ \mathrm{m},\ Y(0) = 1\ \mathrm{m},\ Z(0) = 0\ \mathrm{m},\ Phi(0) = 0\ \mathrm{rad}$.

$$\begin{aligned} x_d(t) &= \begin{cases} 0\ \mathrm{m}, 0 \le t \le 20\ \mathrm{s} \\ 0.2(t-20)\ \mathrm{m}, 20 \le t < 40\ \mathrm{s} \\ \sin(0.04\pi(t-40)) + 4\ \mathrm{m}, 40 \le t < 60\ \mathrm{s} \\ -0.2(t-60) + 4\ \mathrm{m}, 60 \le t < 80\ \mathrm{s} \\ -\sin(0.05\pi(t-80)), 80 \le t < 100\ \mathrm{s} \\ 0.2(t-100)\ \mathrm{m}, 100 \le t \le 120\ \mathrm{s} \end{cases} \\ y_d(t) &= \begin{cases} 1\ \mathrm{m}, 0 \le t \le 20\ \mathrm{s} \\ 1\ \mathrm{m}, 20 \le t < 40\ \mathrm{s} \\ -\mathrm{co\ s}(0.05\pi(t-40)) + 2\ \mathrm{m}, 40 \le t < 60\ \mathrm{s} \\ 3\ \mathrm{m}, 60 \le t < 80\ \mathrm{s} \\ -\cos(0.05\pi(t-80)) + 4\ \mathrm{m}, 80 \le t < 100\ \mathrm{s} \\ 5\ \mathrm{m}, 100 \le t \le 120\ \mathrm{s} \end{cases} \\ z_d(t) &= \begin{cases} 0.3t\ \mathrm{m}, 0 \le t < 20\ \mathrm{s} \\ 6 - 4\mathrm{co\ s}(0.1\pi) + 5\sin(0.1\pi x) + 4\cos(0.1\pi y)\mathrm{m}, 20 \le t \le 120\ \mathrm{s} \end{cases} \\ \psi_\mathrm{d}(t) &= \begin{cases} 0\ \mathrm{rad}, 0 \le t < 20\ \mathrm{s} \\ 0\ \mathrm{rad}, 20 \le t < 40\ \mathrm{s} \\ 0.05\pi(t-40)\ \mathrm{rad}, 40 \le t < 60\ \mathrm{s} \\ \pi\ \mathrm{rad}, 60 \le t < 80\ \mathrm{s} \\ \pi - 0.05\pi(t-80)\ \mathrm{rad}, 80 \le t < 100\ \mathrm{s} \\ 0\ \mathrm{rad}, 100 \le t \le 120\ \mathrm{s} \end{cases} \end{aligned} \tag{51}$$

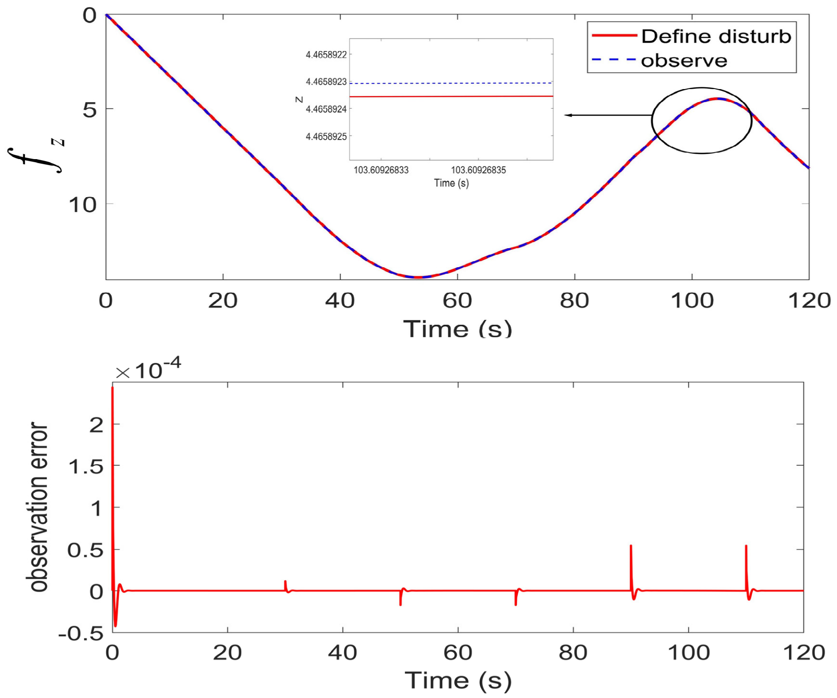
The simulation results shown in Figure 11 demonstrate that the DDPG-LADRC can achieve accurate disturbance estimation for the perturbation observations and corresponding disturbance observation error curves of the three state variables f_x, f_y, f_z. The maximum observation error value for the observer in the X-direction is 0.00141, the maximum observation error in the Y-direction is 0.0016, and the maximum observation error in the Z-direction is 0.0021. DDPG-optimized LESO has achieved the estimation accuracy for disturbances that meet our requirements.



(**a**) Disturbance observation and observation error.

(**b**) Disturbance observation and observation error.

Figure 11. *Cont.*



(**c**) Disturbance observation and observation error.

Figure 11. DDPG-LADRC disturbance observation and corresponding error.

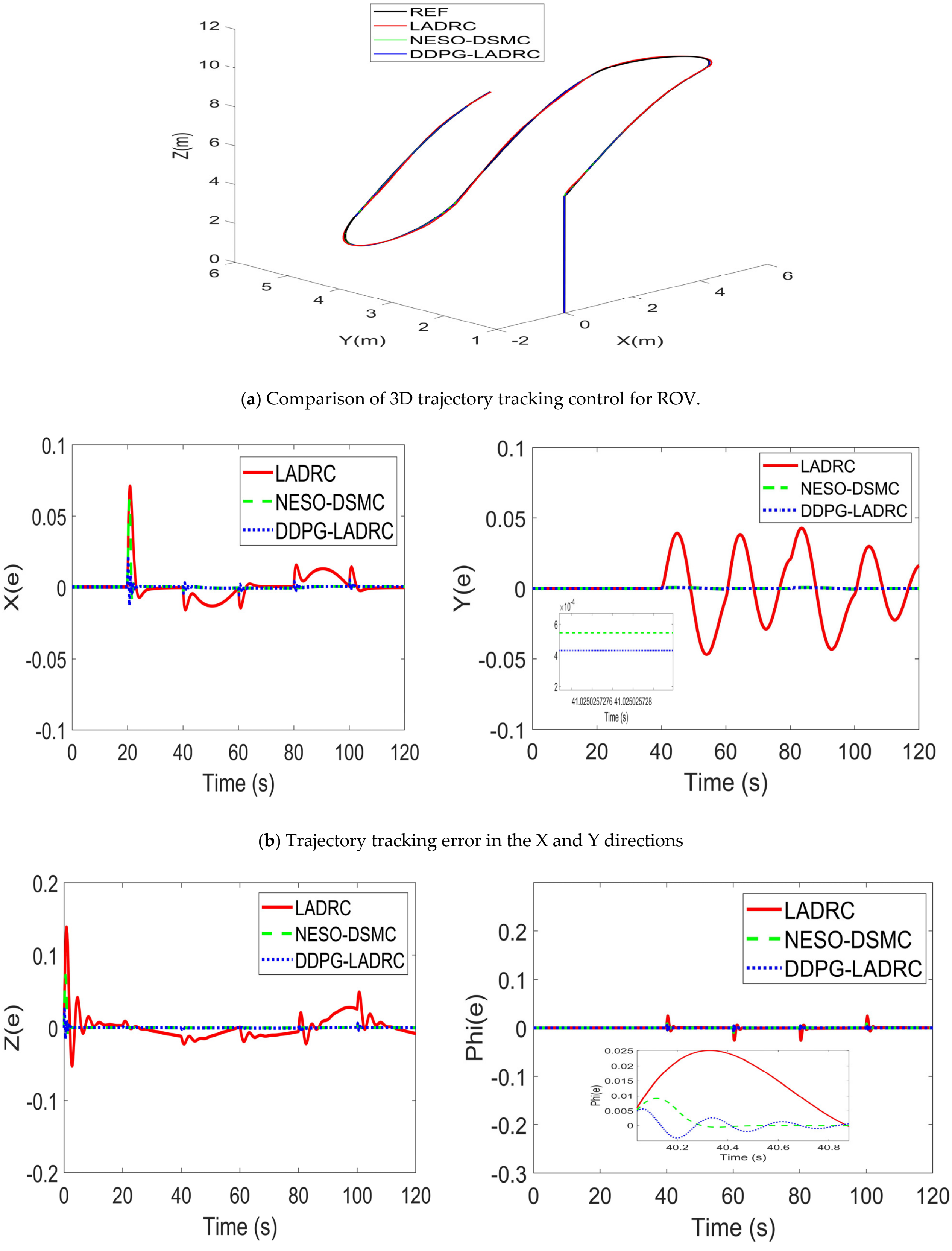
From Figure 12, it can be seen that LADRC, due to the issue of fixed parameters in the controller, is unable to eliminate steady-state errors in a short time. Under continuously changing external disturbances, LADRC cannot achieve optimal control performance. In the presence of significant uncertain disturbances, NESO-DSMC cannot reach the same level of error convergence accuracy as DDPG-LADRC. Tables 8 and 9 show the RMSE and MAE under different control methods, indicating that DDPG-LADRC has better robustness compared to LADRC and NESO-DSMC. DDPG-LADRC can eliminate steady-state errors within 5 s because it incorporates DDPG for online adjustment of LADRC parameters in response to uncertain disturbances caused by environmental changes, significantly improving the control accuracy of the system. This ensures that the ROV can maintain satisfactory control performance even in the presence of inaccurate model parameters and significant uncertain disturbances.

Table 8. Root Mean Square Error (*X*, *Y*, *Z* = m, *Phi* = rad).

Comparison	*X*	*Y*	*Z*	*Phi*
LADRC	0.01	0.021	0.014	0.003
NESO-DSMC	0.005	0.0003	0.006	0.0006
DDPG-LADRC	0.0011	0.0002	0.0012	0.00029

Table 9. Mean Absolute Error (*X*, *Y*, *Z* = m, *Phi* = rad).

Comparison	*X*	*Y*	*Z*	*Phi*
LADRC	0.005	0.015	0.012	0.008
NESO-DSMC	0.001	0.0003	0.0007	0.00028
DDPG-LADRC	0.0006	0.0001	0.0005	0.00011



(**a**) Comparison of 3D trajectory tracking control for ROV.

(**b**) Trajectory tracking error in the X and Y directions

(**c**) Trajectory tracking error in the Z and *Phi* directions.

Figure 12. ROV 3D trajectory and error.

6. Conclusions

In response to the issue of underwater robots facing difficulties in determining model parameters and external disturbances, and the inability of traditional fixed-parameter controllers to achieve optimal control performance for the controlled object, an online parameter tuning strategy based on active disturbance rejection control has been proposed: the DDPG-LADRC algorithm.

1. Based on the nonlinear model of underwater robots, dynamic parameter uncertainty was considered, and a linear active disturbance rejection controller was designed. The convergence of the extended state observer in the linear active disturbance rejection controller and the stability of the closed-loop control were proven using the Lyapunov method. To address the issue that fixed parameter controllers in nonlinear systems cannot achieve optimal control performance, a DDPG-LADRC control strategy was designed, which improved the performance of the LESO by online adjusting control parameters, resulting in the reward curve of DDPG. A feasibility analysis of parameter optimization for LESO was conducted in numerical simulations, demonstrating the effectiveness of the DDPG-LADRC strategy.
2. Compared to PID, fixed-parameter LADRC, and the latest nonlinear observer-based double closed-loop sliding mode control method (NESO-DSMC), the DDPG-LADRC method can generate optimal parameters for the controller, thereby improving control accuracy. Experiments show that this control strategy outperforms PID, fixed-parameter LADRC, and NESO-DSMC control strategies in terms of transient performance and anti-interference capability. Therefore, it can be said that DDPG-LADRC has significant advantages in tracking and anti-interference capabilities.
3. The algorithm, although demonstrating good performance in simulations, still faces significant challenges when being translated into practical engineering applications. For instance, the accurate determination of an ROV's rotational inertia and hydrodynamic coefficients presents a notable challenge. In the future, the parameter adaptation concept based on DDPG can be combined with other control methods to achieve asymptotic stability and optimal control performance.

Author Contributions: G.L.: Writing—review and editing, Writing—original draft, Validation, Supervision, Software, Methodology, Investigation, Funding acquisition, Formal analysis, Data curation, Conceptualization. D.Z.: Writing—review and editing, Writing—original draft, Visualization, Validation, Software, Project administration, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. W.F.: Conceptualization, Funding acquisition, Investigation, Resources, Supervision, Writing—review and editing. Z.J.: Data curation, Investigation, Resources, Supervision, Validation, Writing—review and editing. X.L.: Data curation, Investigation, Supervision, Visualization. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the project [2024]52 funded by the Guangdong Provincial Department of Natural Resources and the Guangdong Provincial Natural Resources Research Committee. We appreciate the support you provided. It was also funded by the Shanghai Municipal Industrial Collaborative Innovation Technology Project (XTCX-KJ-2023-2-15). We sincerely thank you for your valuable help and support.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author. The data are not publicly available due to copyright issues with co-developers.

Conflicts of Interest: Author Zhe Jiang was employed by the company Lanqi Robot Co., Ltd. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

1. Guerrero, J.; Torres, J.; Creuze, V.; Chemori, A.; Campos, E. Saturation based nonlinear PID control for underwater vehicles: Design, stability analysis and experiments. *Mechatronics* **2019**, *61*, 96–105. [CrossRef]
2. Sarhadi, P.; Noei, A.R.; Khosravi, A. Model reference adaptive PID control with anti-windup compensator for an autonomous underwater vehicle. *Robot. Auton. Syst.* **2016**, *83*, 87–93. [CrossRef]
3. Han, Y.; Liu, J.; Yu, J.; Sun, C. Adaptive fuzzy quantized state feedback control for AUVs with model uncertainty. *Ocean Eng.* **2024**, *313*, 119496. [CrossRef]
4. Li, M.; Yu, C.; Zhang, X.; Liu, C.; Lian, L. Fuzzy adaptive trajectory tracking control of work-class ROVs considering thruster dynamics. *Ocean Eng.* **2023**, *267*, 113232. [CrossRef]
5. Yang, M.; Sheng, Z.; Yin, G.; Wang, H. A recurrent neural network based fuzzy sliding mode control for 4-DOF ROV movements. *Ocean Eng.* **2022**, *256*, 111509. [CrossRef]
6. Chen, B.; Hu, J.; Zhao, Y.; Ghosh, B.K. Finite-time observer based tracking control of uncertain heterogeneous underwater vehicles using adaptive sliding mode approach. *Neurocomputing* **2022**, *481*, 322–332. [CrossRef]
7. Long, C.; Hu, M.; Qin, X.; Bian, Y. Hierarchical trajectory tracking control for ROVs subject to disturbances and parametric uncertainties. *Ocean Eng.* **2022**, *266 Pt 1*, 112733. [CrossRef]
8. Luo, W.; Liu, S. Disturbance observer based nonsingular fast terminal sliding mode control of underactuated AUV. *Ocean Eng.* **2023**, *279*, 114553. [CrossRef]
9. Huang, B.; Yang, Q. Double-loop sliding mode controller with a novel switching term for the trajectory tracking of work-class ROVs. *Ocean Eng.* **2019**, *178*, 80–94. [CrossRef]
10. Wen, J.; Zhang, J.; Yu, G. Predefined-Time Three-Dimensional Trajectory Tracking Control for Underactuated Autonomous Underwater Vehicles. *Appl. Sci.* **2025**, *15*, 1698. [CrossRef]
11. Chu, Z.; Xiang, X.; Zhu, D.; Luo, C.; Xie, D. Adaptive trajectory tracking control for remotely operated vehicles considering thruster dynamics and saturation constraints. *ISA Trans.* **2020**, *100*, 28–37. [CrossRef] [PubMed]
12. Shojaei, K. Neural network feedback linearization target tracking control of underactuated autonomous underwater vehicles with a guaranteed performance. *Ocean Eng.* **2022**, *258*, 111827. [CrossRef]
13. Bao, H.; Zhang, Y.; Song, M.; Kong, Q.; Hu, X.; An, X. A review of underwater vehicle motion stability. *Ocean Eng.* **2023**, *287*, 115735. [CrossRef]
14. Zheng, J.; Song, L.; Liu, L.; Yu, W.; Wang, Y.; Chen, C. Fixed-time sliding mode tracking control for autonomous underwater vehicles. *Appl. Ocean Res.* **2021**, *117*, 102928. [CrossRef]
15. Xia, T.; Yang, Q.; Huang, B.; Ouyang, Y.; Zheng, Y.; Mao, P. Enhanced trajectory tracking control algorithm for ROVs considering actuator saturation, external disturbances, and model parameter uncertainties. *Ocean Eng.* **2024**, *311*, 118973. [CrossRef]
16. Han, J. Auto disturbance rejection controller and its applications. *Control Decis.* **1998**, *13*, 19–23.
17. Gao, J.; Liang, X.; Chen, Y.; Zhang, L.; Jia, S. Hierarchical image-based visual serving of underwater vehicle manipulator systems based on model predictive control and active disturbance rejection control. *Ocean Eng.* **2021**, *229*, 108814. [CrossRef]
18. Gao, Z. Scaling and bandwidth-parameterization based controller tuning. In Proceedings of the 2003 American Control Conference, Denver, CO, USA, 4–6 June 2003. [CrossRef]
19. Li, S.; Chen, Z.; Ju, Y.; Jia, Y.; Tang, W.; Wang, Y. Transverse vibration analysis and active disturbance rejection decoupling control of vector propulsion shaft system for underwater vehicles. *Ocean Eng.* **2023**, *298*, 117158. [CrossRef]
20. Zhao, L.; Liu, X.; Wang, T. Trajectory tracking control for double-joint manipulator systems driven by pneumatic artificial muscles based on a nonlinear extended state observer. *Mech. Syst. Signal Process.* **2019**, *122*, 307–320. [CrossRef]
21. Zheng, Y.; Chen, Z.; Huang, Z.; Sun, M.; Sun, Q. Active disturbance rejection controller for multi-area interconnected power system based on reinforcement learning. *Neurocomputing* **2021**, *425*, 149–159. [CrossRef]
22. Huang, Z.; Chen, Z.; Zheng, Y.; Sun, M.; Sun, Q. Optimal design of load frequency active disturbance rejection control via double chains quantum genetic algorithm. *Neural Comput. Appl.* **2020**, *33*, 3325–3345. [CrossRef]
23. Chen, Z.; Qin, B.; Sun, M.; Sun, Q. Q-learning-based parameters adaptive algorithm for active disturbance rejection control and its application to ship course control. *Neurocomputing* **2020**, *408*, 51–63. [CrossRef]
24. Sehgal, A.; Ward, N.; La, H.M.; Papachristos, C.; Louis, S. GA+DDPG+HER: Genetic algorithm-based function optimizer for deep reinforcement learning in robotic manipulation tasks. In Proceedings of the 2022 6th IEEE International Conference on Robotic Computing (IRC), Naples, Italy, 5–7 December 2022; pp. 85–86. [CrossRef]
25. Xu, W.; Liu, J.; Yu, J.; Han, Y. Low complexity adaptive neural network three-dimensional tracking control for autonomous underwater vehicles considering uncertain dynamics. *Eng. Appl. Artif. Intell.* **2025**, *142*, 109860. [CrossRef]

26. Luo, G.; Gao, S.; Jiang, Z.; Luo, C.; Zhang, W.; Wang, H. ROV trajectory tracking control based on disturbance observer and combinatorial reaching law of sliding mode. *Ocean Eng.* **2024**, *304*, 117744. [CrossRef]
27. Liang, Y.; Guo, C.; Ding, Z.; Hua, H. Agent-based modeling in electricity market using deep deterministic policy gradient algorithm. *IEEE Trans. Power Syst.* **2020**, *35*, 4180–4192. [CrossRef]
28. Qi, G.; Li, Y. Reinforcement learning control for robot arm grasping based on improved DDPG. In Proceedings of the 2021 40th Chinese Control Conference (CCC), Shanghai, China, 26–28 July 2021; pp. 4132–4137. [CrossRef]
29. Mousavifard, R.; Alipour, K.; Najafqolian, M.A.; Zarafshan, P. Quadrotor trajectory tracking using combined stochastic model-free position and DDPG-based attitude control. *ISA Trans.* **2025**, *156*, 240–252. [CrossRef]
30. Wu, D.; Dong, X.; Shen, J.; Hoi, S.C.H. Reducing estimation bias via triplet-average deep deterministic policy gradient. *IEEE Trans. Neural Netw. Learn. Syst.* **2020**, *31*, 4933–4945. [CrossRef]
31. Khalil, H.K. *Nonlinear Systems*, 3rd ed.; Prentice Hall: Hoboken, NJ, USA, 2002.

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.