

Article

Research on Motion Control of Hydraulic Manipulator Based on Prescribed Performance and Reinforcement Learning

Yuhe Li ^{1,2,*} and Xiaowen Qi ¹¹ School of Mechanical Engineering, Yanshan University, Qinhuangdao 066004, China; qxw@ysu.edu.cn² Tianjin Research Institute of Construction Machinery Co., Ltd., Tianjin 300409, China

* Correspondence: liyh@stumail.ysu.edu.cn

Abstract

Achieving high-precision motion control for hydraulic manipulators presents a challenging task. Addressing the issue of low motion control accuracy caused by the strong electromechanical-hydraulic coupling characteristics of hydraulic manipulator systems, this paper innovatively introduces an RBF neural network and an Actor–Critic reinforcement learning architecture within a performance-based control framework designed using the inverse method. This approach enables dual compensation for both internal uncertainties and external disturbances within the manipulator, thereby enhancing the system's control performance. First, within the control architecture, the performance function ensures system transient performance while employing an RBF neural network to estimate and compensate for internal unmodeled errors caused by mechanical coupling and hydraulic parameter uncertainties. Stability proofs are used to derive the network weight update rate. Second, a disturbance compensator is designed based on reinforcement learning. Deployed into the controller through offline training and online adaptation, it compensates for external system disturbances, further improving control accuracy. Finally, comparative and ablation experiments conducted on a hydraulic manipulator testbed demonstrate the effectiveness of the disturbance compensator. Compared to PID control, the proposed approach achieves a 60–65% improvement in control accuracy.

Keywords: hydraulic machinery; reinforcement learning; neural networks; performance function

1. Introduction

Manipulators serve as indispensable high-tech equipment in energy and public welfare sectors such as mining operations, emergency rescue, urban and rural infrastructure development, and industrial applications. They represent high-value-added, multifunctional flagship products [1,2]. They increasingly meet market demands for high-precision motion control amid the trend toward intelligentization. Not only do they significantly enhance operational efficiency and work quality, but they also reduce human workload, improve construction safety and automation levels, and lay the foundation for intelligent development [3].

The primary methods currently applied to the motion control of manipulators include Proportional Integral Derivative (PID) control [4], adaptive robust control [5,6], fuzzy logic control [7], improved Linear Quadratic Regulator [8], and sliding mode control [9]. Although PID, as a typical model-free controller, offers advantages such as convenient parameter tuning and simple structure, its inability to effectively address the strong



Academic Editor: Zhuming Bi

Received: 22 November 2025

Revised: 20 December 2025

Accepted: 4 January 2026

Published: 6 January 2026

Copyright: © 2026 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 \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

electromechanical-hydraulic coupling characteristics of manipulators results in low control accuracy, making it difficult to meet current market demands [10]. Furthermore, unlike electrically driven manipulators, hydraulically driven manipulators present unpredictable, complex operating conditions and external disturbances. Their inherent joint strong coupling characteristics and uncertainties in hydraulic system parameters, specifically, that manipulators commonly exhibit nonlinear differential equation structures, friction nonlinearities, and uncertainties in parameters such as load, hydraulic oil volumetric elastic modulus, and viscous friction coefficients. Additionally, external disturbances, oil leakage rates, and dynamic friction forces present difficult-to-model uncertain nonlinearities [11,12]. These factors compromise the accuracy of model-based control algorithms. There is an urgent need for a control algorithm that can overcome the challenges inherent in hydraulic manipulators and enhance their motion control precision.

To enhance control accuracy, prior work introduced extended state observers (ESOs) and nonlinear disturbance observers (NDOs) to monitor system uncertainties and disturbances [13]. A classic approach integrates these observers into adaptive robust control strategies, enabling both precise estimation of unknown disturbances and effective handling of parameter uncertainties [14]. Building upon this foundation, Zhang et al. proposed an adaptive robust control method based on a dual extended state observer to observe the uncertainties in a hydraulic manipulator system, achieving good tracking accuracy [15]. Shi et al. effectively resolved the trade-off between tracking accuracy and disturbance rejection capability by employing a disturbance compensation mechanism based on NDO to address external disturbances and unmodeled dynamics [16]. Although the aforementioned methods effectively compensate for parameter and unmodeled nonlinearities, they remain sensitive to sensor noise. The high computational demands for real-time operation introduce significant control feedback delays, hindering practical implementation.

In manipulator motion control, neural networks are widely employed as feedforward compensators to address system nonlinearity and model uncertainty due to their strong approximation capabilities and learning capacity [17,18]. Asko Rouvinen et al. approximate the uncertainties in the system by using different methods of neural network, and obtain good control performance [19]. Tran et al. designed an adaptive control method for multi-degree-of-freedom hydraulic manipulators, employing RBF neural networks to approximate and compensate for uncertainties, achieving high control performance [20]. Su et al. proposed a manipulator control method combining adaptive neural networks with backstepping control to compensate for unknown system dynamics, enhancing control accuracy [21].

The aforementioned control algorithms primarily focus on nonlinear compensation and steady-state tracking performance improvement, without considering the transient performance of the control system. In current research, prescriptive performance control is achieved by setting different performance functions [22,23], keeping the tracking error within an adjustable range and demonstrating significant transient and steady-state performance. Wang Shubo et al. proposed an adaptive funnel control scheme for manipulators with unknown blind zones. Error transformation was applied in controller design [24]. Yang et al. introduced a novel adaptive dynamic surface asymptotic tracking controller based on neural networks, approximating unknown dynamics while ensuring tracking error remains within predetermined boundaries via performance functions, achieving high transient tracking performance [25]. Liang Xianglong et al. proposed an adaptive neural network admittance control method based on integral obstacle Lyapunov functions, achieving excellent position and force tracking control while ensuring system outputs remain within preset limits [26].

In recent years, researchers have explored applying reinforcement learning to hydraulic manipulators. As an intelligent algorithm applicable to both model-based and model-free scenarios, its Actor–Critic architecture has seen the widest industrial adoption [27]. Yao et al. combined the Actor–Critic reinforcement learning architecture with PID control, achieving high motion accuracy through online PID parameter tuning [28]. Additionally, when integrated with RISE controllers, reinforcement learning estimated and compensated for unmodeled errors in the system, similarly yielding high control precision [29]. These studies demonstrate the feasibility of applying reinforcement learning to hydraulic manipulators.

Building on the above research, this paper addresses the current challenge in hydraulic manipulator control algorithms, which struggle to simultaneously accommodate both system transient response and internal/external uncertainties. By employing a performance-based controller as the foundational architecture, the system’s position output error can be constrained within predetermined boundaries, thereby ensuring transient performance. Furthermore, an RBF neural network is introduced to estimate and compensate for unmodeled internal errors. Simultaneously, a disturbance compensator based on the Actor–Critic reinforcement learning architecture is designed to handle external disturbance uncertainties. This dual compensation mechanism significantly enhances the system’s motion control accuracy.

The rest of this paper is as follows. Section 2 constructs the excavator model; Section 3 designs the proposed control algorithm; Section 4 analyzes the algorithm’s stability and convergence; Section 5 experimentally validates the algorithm’s application; Section 6 summarizes the paper.

2. Hydraulic Manipulator Model

The hydraulic manipulator studied in this paper is modified from an excavator, with its structure shown in Figure 1. This paper only considers the compound motion of the boom and arm joints within a single plane. The dynamics of the hydraulic manipulator in the joint space can be described as follows:

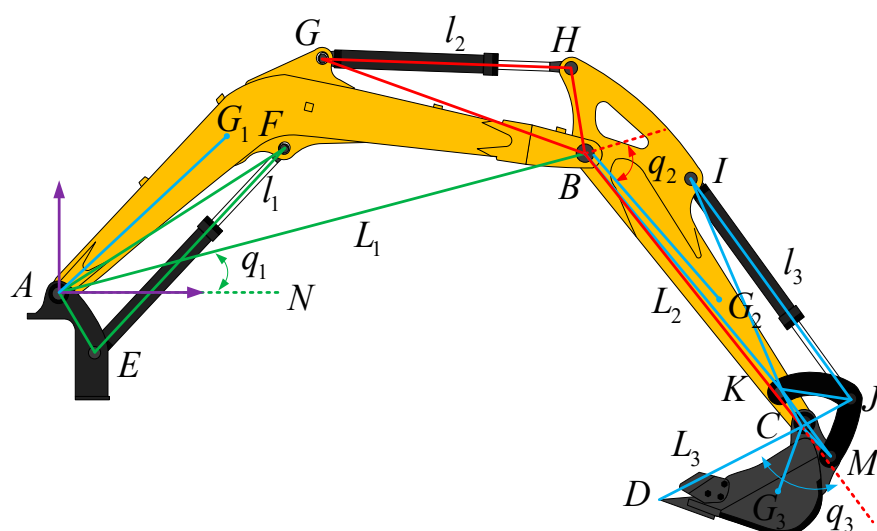


Figure 1. Hydraulic manipulator structure diagram.

The force balance equation of inertial load can be expressed as:

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + B\dot{q} + G(q) + \Delta M(q)\ddot{q} + \Delta C(q, \dot{q})\dot{q} + \Delta B\dot{q} + \Delta G(q) = \tau \quad (1)$$

where $M(q) \in \mathbb{R}^{3 \times 3}$ represents the inertia matrix; $C(q, \dot{q}) \in \mathbb{R}^{3 \times 3}$ denotes the Coriolis and centrifugal force matrix; $G(q) \in \mathbb{R}^3$ signifies the gravitational force matrix; $B \in \mathbb{R}^{3 \times 3}$ indicates the viscous friction coefficient matrix; $q \in \mathbb{R}^3$, $\dot{q} \in \mathbb{R}^3$, and $\ddot{q} \in \mathbb{R}^3$ respectively denote the joint angular displacement, angular velocity, and angular acceleration vectors; $\Delta M(q)$, $\Delta C(q, \dot{q})$, $\Delta G(q)$, ΔB represent the unmodeled portions of joint coupling dynamics; τ denotes the torque vector acting on the joint.

The motion of this hydraulic manipulator is driven by a single-rod hydraulic cylinder, as shown in Figure 2. Let $x_{ha} = [x_{ha1}, x_{ha2}]^T$ denote the displacement vector of the boom and arm hydraulic cylinder, i.e., the displacement vector in the actuator space. Then, the velocity vector $\dot{x}_{ha}$ can be expressed in terms of the joint vectors q and $\dot{q}$ as follows:

$$\dot{x}_{ha} = \frac{\partial x_{ha}}{\partial q} \dot{q} = \text{diag} \left\{ \frac{\partial x_{ha1}}{\partial q_1}, \frac{\partial x_{ha2}}{\partial q_2} \right\} \dot{q} \quad (2)$$

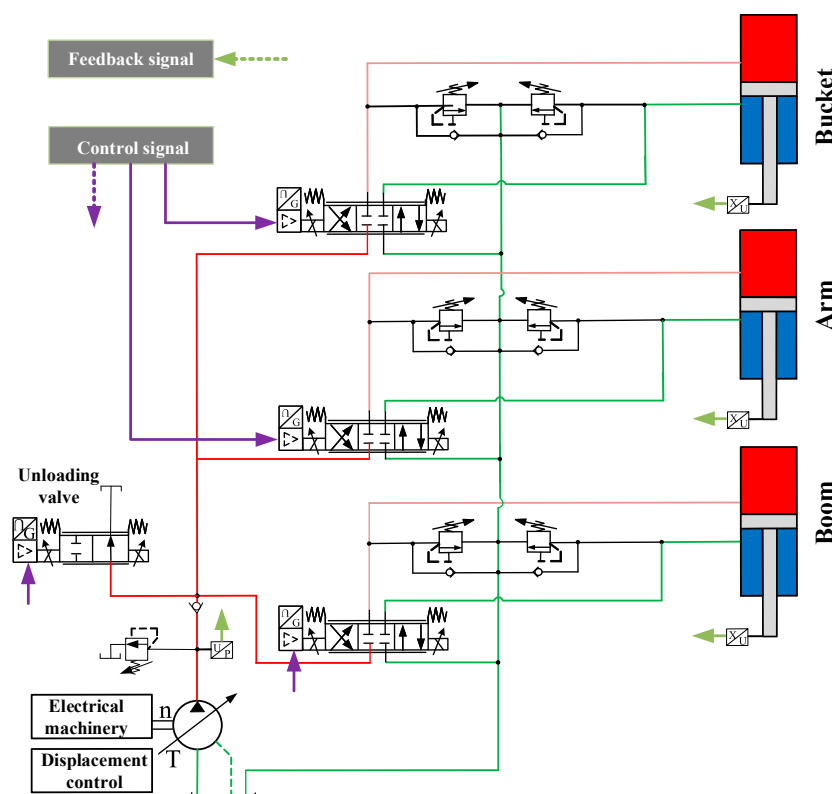


Figure 2. Manipulator hydraulic schematic diagram.

The torque vector acting on the manipulator joint can be expressed as:

$$\tau = J_{ha} (A_1 P_1 - A_2 P_2) = \frac{\partial x_{ha}}{\partial q} (A_1 P_1 - A_2 P_2) \quad (3)$$

where A_1 is the piston area of the rodless chamber of the hydraulic cylinder; A_2 is the piston area of the rod chamber of the hydraulic cylinder; P_1 is the pressure in the rodless chamber of the hydraulic cylinder; P_2 is the pressure in the rod chamber of the hydraulic cylinder;

Neglecting internal leakage, the dynamic pressure equations for the rod chamber and rodless chamber of the hydraulic cylinder can be expressed as [30]:

$$\dot{P}_1 = \beta_e (-A_1 \dot{x}_{ha} + Q_1) / V_1 \quad \dot{P}_2 = \beta_e (A_2 \dot{x}_{ha} - Q_2) / V_2 \quad (4)$$

where β_e is the effective volumetric modulus of the oil; $V_1 = V_{01} + A_1 \cdot x_{ha}$ is the rodless chamber volume of the hydraulic cylinder; V_{01} is the initial volume of the rodless chamber; $V_2 = V_{02} - A_2 \cdot x_{ha}$ is the rod chamber volume; V_{02} is the initial volume of the rod chamber; Q_1 is the supply flow rate to the hydraulic cylinder; Q_2 is the return flow rate from the hydraulic cylinder.

The hydraulic cylinder flow equation is:

$$\begin{aligned} Q_1 &= k_v u \left[s(u) \sqrt{(P_s - P_1)} + s(-u) \sqrt{(P_1 - P_r)} \right] \\ Q_2 &= k_v u \left[s(u) \sqrt{(P_2 - P_r)} + s(-u) \sqrt{(P_s - P_2)} \right] \end{aligned} \quad (5)$$

where k_v represents flow gain; u denotes the proportional valve control signal; P_s indicates the supply pressure; and P_r signifies the return pressure.

Define state variable $x = [x_1, x_2, x_3]^T$, where $x_1 = q$, $x_2 = \dot{q}$, $x_3 = A_1 P_1 - A_2 P_2$. The state equation is:

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= M^{-1} (J_{ha} x_3 - N_1(x_2) + N_2(x_2)) \\ \dot{x}_3 &= g_1(x_3) u - g_2(x_2) \end{aligned} \quad (6)$$

where

$$\begin{cases} N_1(x_2) = Cx_2 + Bx_2 + G & N_2(x_2) = -\Delta M \dot{x}_2 - \Delta Cx_2 - \Delta Bx_2 - \Delta G \\ g_1(x_3) = \beta_e k_v (A_1 R_1 / V_1 + A_2 R_2 / V_2) & g_2(x_2) = \beta_e J_{ha} (A_1^2 / V_1 + A_2^2 / V_2) x_2 \\ R_1 = s(u) \sqrt{P_s - P_1} + s(-u) \sqrt{P_1 - P_r} & R_2 = s(u) \sqrt{P_2 - P_r} + s(-u) \sqrt{P_s - P_2} \end{cases} \quad (7)$$

3. Controller Design

This section first designs a performance-based controller that employs an RBF neural network to estimate and compensate for unmodeled mechanical and hydraulic errors in the manipulator. Subsequently, building upon this performance controller, a disturbance compensator is designed using the Actor–Critic reinforcement learning architecture to counteract external disturbances, thereby further enhancing control accuracy.

3.1. RBF Neural Network

The RBF neural network consists of an input layer, a hidden layer, and an output layer. Due to its simple structure, fast learning convergence, and ability to approximate continuous nonlinear functions on compact sets with arbitrary precision, it is commonly used in controllers for real-time fitting of unmodeled errors. Adaptive laws designed based on Lyapunov stability theory can adjust network weights online, enabling the controller to compensate for unmodeled errors while ensuring the stability and tracking performance of the closed-loop system [31]. The transformation from the input layer to the hidden layer is nonlinear, while the transformation from the hidden layer to the output layer is linear. The activation function of the neurons in the hidden layer is the radial basis function. Any continuous unknown function can be approximated as:

$$F(X) = W^{*T} \phi(X) + \varepsilon \quad (8)$$

where $W^{*T} \in \mathbb{R}^{c \times v}$ is the weight vector, c and v represent the number of outputs and the number of hidden layer neurons, respectively; $X \in \mathbb{R}^{z \times l}$ is the input to the neural network, z is the number of inputs; ε is the neural network approximation error satisfying $\|\varepsilon\| \leq \bar{\varepsilon}$, $\bar{\varepsilon}$ is a positive constant; $\phi(X) = [\phi_1(X), \phi_2(X), \dots, \phi_l(X)]$ is the activation function, expressed as:

$$\phi_i(X) = \exp \left(-(X - \mu_i)^T (X_1 - \mu_i) / \sigma_i^2 \right), i = 1, 2, l \quad (9)$$

where μ_i and σ_i represent the center and width of the Gaussian function for the i th neuron, respectively.

3.2. Performance Function

The time-varying logarithmic Lyapunov function is designed as follows:

$$V = \frac{1}{2} \ln \frac{k_a^2(t)}{k_a^2(t) - z^T z} \quad (10)$$

where $k_a(t)$ represents the error constraint boundary, $k_a(t) > 0$; z denotes tracking errors.

As shown in Equation (10), when the system tracking error variable exceeds the time-varying boundary, the function value of the time-varying log-type Lyapunov function tends toward infinity. Based on this characteristic, by rigorously proving the boundedness of this Lyapunov function under the controller's action, it is ensured that the closed-loop system tracking error strictly remains within the preset event constraint boundary $(-k_a(t), k_a(t))$, where $k_{ai}(t)$ is expressed as:

$$k_a(t) = (k_{a0} - k_{a\infty}) \exp^{-nt} + k_{a\infty} \quad (11)$$

where k_{a0} , $k_{a\infty}$, n are constants greater than zero and satisfy $0 < k_{a\infty} < k_{a0}$, $\|z(0)\| < k_{a0}$; $z(0)$ is the initial value of z . The image of $k_a(t)$ is shown in Figure 3:

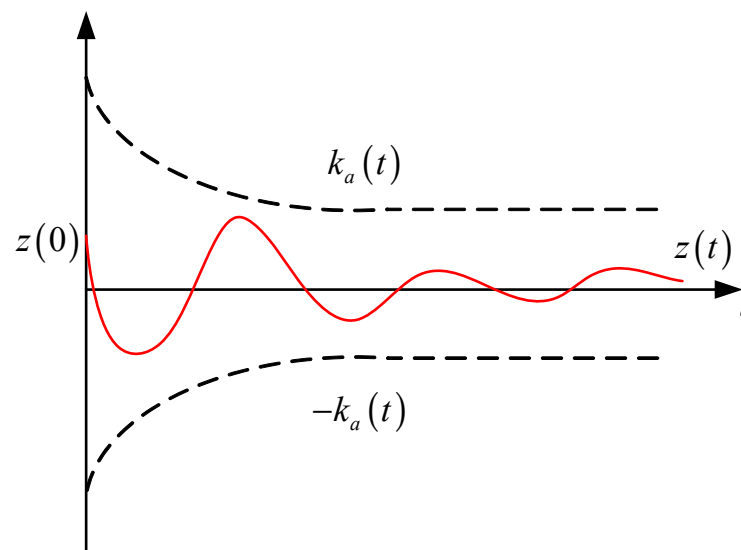


Figure 3. Performance function boundary and tracking error.

3.3. Prescribed Performance Controller Design

Define the error variable as:

$$z_1 = x_1 - x_{1d}, \quad z_2 = x_2 - \alpha_1, \quad z_3 = x_3 - \alpha_2 \quad (12)$$

where α_1 , α_2 are virtual control variables.

Define the first Lyapunov function as:

$$V_1 = \frac{1}{2} \ln \frac{k_{b1}^2(t)}{k_{b1}^2(t) - z_1^T z_1} \quad (13)$$

The derivation of V_1 yields:

$$\dot{V}_1 = \frac{z_1^T}{k_{b1}^2 - z_1^T z_1} \left(x_2 - \dot{x}_{1d} - \frac{\dot{k}_{b1}}{k_{b1}} z_1 \right) = \frac{z_1^T}{k_{b1}^2 - z_1^T z_1} \left(z_2 + \alpha_1 - \dot{x}_{1d} - \frac{\dot{k}_{b1}}{k_{b1}} z_1 \right) \quad (14)$$

For the convenience of representation, in the following, $\ln(\cdot)_i$ is used to denote $\ln(k_{bi}^2(t)/k_{bi}^2(t) - z_i^T z_i)$, $i = 1, 2, 3$, and the virtual control rate α_1 is designed based on Equation (14) and the fixed time theory as:

$$\alpha_1 = \dot{x}_{1d} + \frac{\dot{k}_{b1}}{k_{b1}} z_1 - k_{11}^* \left(\frac{\ln^{\gamma_1}(\cdot)_1 (k_{b1}^2 - z_1^T z_1)}{z_1^T} \right) - k_{12}^* \left(\frac{\ln^{\gamma_2}(\cdot)_1 (k_{b1}^2 - z_1^T z_1)}{z_1^T} \right) \quad (15)$$

where k_{11}^* and k_{12}^* are constants greater than zero; $0 < \gamma_1 < 1$, $1 < \gamma_2 < \infty$.

Substituting Equation (15) into Equation (14) yields:

$$\dot{V}_1 = \frac{z_1^T z_2}{k_{b1}^2 - z_1^T z_1} - k_{11}^* \ln^{\gamma_1}(\cdot)_1 - k_{12}^* \ln^{\gamma_2}(\cdot)_1 \quad (16)$$

Further, the second Lyapunov function is designed:

$$V_2 = V_1 + \frac{1}{2} \ln \frac{k_{b2}^2(t)}{k_{b2}^2(t) - z_2^T z_2} + \frac{1}{2\eta_{w1}} \tilde{W}_1^T \tilde{W}_1 + \frac{1}{2\eta_{\theta 1}} \tilde{\theta}_1^T \tilde{\theta}_1 \quad (17)$$

where $\tilde{W}_1 = W_1^* - \hat{W}_1$, $\tilde{\theta}_1 = \theta_1 - \hat{\theta}_1$, η_{w1} and $\eta_{\theta 1}$ are constants greater than zero.

The derivation of Equation (17) is obtained:

$$\begin{aligned} \dot{V}_2 &= \dot{V}_1 + \frac{z_2^T}{k_{b2}^2 - z_2^T z_2} \left[\frac{J_{ha}}{M} x_3 - \frac{N_1(x_2)}{M} + \frac{N_2(x_2)}{M} - \dot{\alpha}_1 - \frac{\dot{k}_{b2}}{k_{b2}} z_2 \right] - \frac{1}{\eta_{w1}} \tilde{W}_1^T \dot{\tilde{W}}_1 - \frac{1}{\eta_{\theta 1}} \tilde{\theta}_1^T \dot{\tilde{\theta}}_1 \\ &= -k_{11}^* \ln^{\gamma_1}(\cdot)_1 - k_{12}^* \ln^{\gamma_2}(\cdot)_1 + \frac{z_2^T}{k_{b2}^2 - z_2^T z_2} \left[\frac{J_{ha}}{M} z_3 + \frac{J_{ha}}{M} \alpha_2 - \right. \\ &\quad \left. \frac{N_2(x_2)}{M} + f_1 - \frac{\dot{k}_{b2}}{k_{b2}} z_2 \right] - \frac{1}{\eta_{w1}} \tilde{W}_1^T \dot{\tilde{W}}_1 - \frac{1}{\eta_{\theta 1}} \tilde{\theta}_1^T \dot{\tilde{\theta}}_1 \end{aligned} \quad (18)$$

where $f_1 = \frac{N_2(x_2)}{M} + \frac{d_1}{M} - \dot{\alpha}_1 + \frac{(k_{b2}^2 - z_2^T z_2) z_1}{k_{b1}^2 - z_1^T z_1}$.

The virtual control volume α_2 is designed according to Equation (18) as:

$$\alpha_2 = \frac{M}{J_{ha}} \left[-\hat{f}_1 + \frac{N_1(x_2)}{M} + \frac{\dot{k}_{b2}}{k_{b2}} z_2 \right] - \frac{M}{J_{ha}} \frac{k_{a2}^2 - z_2^T z_2}{z_2^T} [k_{21}^* \ln^{\gamma_1}(\cdot)_2 + k_{22}^* \ln^{\gamma_2}(\cdot)_2] \quad (19)$$

where k_{21}^* and k_{22}^* are constants greater than zero.

Let $\hat{f}_1 = \hat{W}_1^T \phi(X_1) + \hat{\theta}_1$ and substituting together with Equation (19) into Equation (18) yields:

$$\begin{aligned} \dot{V}_2 &= \sum_{i=1}^2 [-k_{i1} \ln^{\gamma_1}(\cdot)_i - k_{i2} \ln^{\gamma_2}(\cdot)_i] + \frac{z_2^T}{k_{b2}^2 - z_2^T z_2} [\tilde{W}_1^T \phi(X_1) + \tilde{\theta}_1] \\ &\quad - \frac{1}{\eta_{w1}} \tilde{W}_1^T \dot{\tilde{W}}_1 + \frac{z_2^T}{k_{b2}^2 - z_2^T z_2} \frac{J_{ha}}{M} z_3 - \frac{1}{\eta_{\theta 1}} \tilde{\theta}_1^T \dot{\tilde{\theta}}_1 \end{aligned} \quad (20)$$

According to Equation (20), the neural network weight update rate can be obtained as:

$$\dot{\tilde{W}}_1 = \eta_{w1} \frac{z_2^T}{k_{b2}^2 - z_2^T z_2} \phi(X_1) - \sigma_{w1} \tilde{W}_1 \quad \dot{\tilde{\theta}}_1 = \eta_{\theta 1} \frac{z_2^T}{k_{b2}^2 - z_2^T z_2} - \sigma_{\theta 1} \tilde{\theta}_1 \quad (21)$$

where both σ_{w1} and $\sigma_{\theta 1}$ are constants greater than zero; $\mathbf{X}_1 = [\mathbf{q}, \dot{\mathbf{q}}, \mathbf{q}_d, \dot{\mathbf{q}}_d, \boldsymbol{\alpha}_1]$. Substituting Equation (21) has:

$$\dot{\mathbf{V}}_2 = \sum_{i=1}^2 [-k_{i1} \ln^{\gamma_1}(\cdot)_i - k_{i2} \ln^{\gamma_2}(\cdot)_i] + \frac{\sigma_{w1}}{\eta_{w1}} \tilde{\mathbf{W}}_1^T \dot{\tilde{\mathbf{W}}}_1 + \frac{\sigma_{\theta 1}}{\eta_{\theta 1}} \tilde{\boldsymbol{\theta}}_1^T \dot{\tilde{\boldsymbol{\theta}}}_1 + \frac{\mathbf{z}_2^T}{k_{b2}^2 - \mathbf{z}_2^T \mathbf{z}_2} \mathbf{J}_{ha} \mathbf{z}_3 \quad (22)$$

According to Young's inequality,

$$\tilde{\mathbf{W}}_1^T \dot{\tilde{\mathbf{W}}}_1 \leq -\frac{1}{2} \tilde{\mathbf{W}}_1^T \tilde{\mathbf{W}}_1 + \frac{1}{2} \mathbf{W}_1^{*T} \mathbf{W}_1^* \quad \tilde{\boldsymbol{\theta}}_1^T \dot{\tilde{\boldsymbol{\theta}}}_1 \leq -\frac{1}{2} \tilde{\boldsymbol{\theta}}_1^T \tilde{\boldsymbol{\theta}}_1 + \frac{1}{2} \boldsymbol{\theta}_1^{*T} \boldsymbol{\theta}_1^* \quad (23)$$

Further, the deflation of Equation (22) yields:

$$\dot{\mathbf{V}}_2 \leq \sum_{i=1}^2 [-k_{i1} \ln^{\gamma_1}(\cdot)_i - k_{i2} \ln^{\gamma_2}(\cdot)_i] - \frac{\sigma_{w1}}{2\eta_{w1}} \tilde{\mathbf{W}}_1^T \tilde{\mathbf{W}}_1 - \frac{\sigma_{\theta 1}}{2\eta_{\theta 1}} \tilde{\boldsymbol{\theta}}_1^T \tilde{\boldsymbol{\theta}}_1 + \frac{\sigma_{w1}}{2\eta_{w1}} \mathbf{W}_1^{*T} \mathbf{W}_1^* + \frac{\sigma_{\theta 1}}{2\eta_{\theta 1}} \boldsymbol{\theta}_1^{*T} \boldsymbol{\theta}_1^* + \frac{\mathbf{z}_2^T}{k_{b2}^2 - \mathbf{z}_2^T \mathbf{z}_2} \mathbf{J}_{ha} \mathbf{z}_3 \quad (24)$$

It can be shown that if $\mathbf{z}_3$ converges, then $\mathbf{V}_2$ eventually reaches consistent boundedness. Define the third Lyapunov function as:

$$\mathbf{V}_3 = \mathbf{V}_2 + \frac{1}{2} \ln \frac{k_{b3}^2(t)}{k_{b3}^2(t) - \mathbf{z}_3^T \mathbf{z}_3} + \frac{1}{2\eta_{w2}} \tilde{\mathbf{W}}_2^T \dot{\tilde{\mathbf{W}}}_2 + \frac{1}{2\eta_{\theta 2}} \tilde{\boldsymbol{\theta}}_2^T \dot{\tilde{\boldsymbol{\theta}}}_2 \quad (25)$$

Derivation of Equation (25) yields:

$$\begin{aligned} \dot{\mathbf{V}}_3 &= \dot{\mathbf{V}}_2 + \frac{\mathbf{z}_3^T}{k_{b3}^2 - \mathbf{z}_3^T \mathbf{z}_3} \left[\mathbf{g}_1(\mathbf{x}_3) \mathbf{u} - \mathbf{g}_2(\mathbf{x}_2) - \dot{\mathbf{a}}_2 - \frac{\dot{k}_{b3}}{k_{b3}} \mathbf{z}_3 \right] - \frac{1}{\eta_{w2}} \tilde{\mathbf{W}}_2^T \dot{\tilde{\mathbf{W}}}_2 - \frac{1}{\eta_{\theta 2}} \tilde{\boldsymbol{\theta}}_2^T \dot{\tilde{\boldsymbol{\theta}}}_2 \\ &\leq \sum_{i=1}^2 [-k_{i1} \ln^{\gamma_1}(\cdot)_i - k_{i2} \ln^{\gamma_2}(\cdot)_i] - \frac{\sigma_{w1}}{2\eta_{w1}} \tilde{\mathbf{W}}_1^T \tilde{\mathbf{W}}_1 - \frac{\sigma_{\theta 1}}{2\eta_{\theta 1}} \tilde{\boldsymbol{\theta}}_1^T \tilde{\boldsymbol{\theta}}_1 + \frac{\sigma_{w1}}{2\eta_{w1}} \mathbf{W}_1^{*T} \mathbf{W}_1^* + \\ &\quad \frac{\sigma_{\theta 1}}{2\eta_{\theta 1}} \boldsymbol{\theta}_1^{*T} \boldsymbol{\theta}_1^* + \frac{1}{2} \|\boldsymbol{\varepsilon}\|^2 + \frac{\mathbf{z}_3^T}{k_{b3}^2 - \mathbf{z}_3^T \mathbf{z}_3} \left[\mathbf{g}_1(\mathbf{x}_3) \mathbf{u} - \mathbf{g}_2(\mathbf{x}_2) + \mathbf{f}_2 - \frac{\dot{k}_{b3}}{k_{b3}} \mathbf{z}_3 \right] \\ &\quad - \frac{1}{\eta_{w2}} \tilde{\mathbf{W}}_2^T \dot{\tilde{\mathbf{W}}}_2 - \frac{1}{\eta_{\theta 2}} \tilde{\boldsymbol{\theta}}_2^T \dot{\tilde{\boldsymbol{\theta}}}_2 \end{aligned} \quad (26)$$

$$\text{where } \mathbf{f}_2 = -\dot{\mathbf{a}}_2 + \frac{(k_{b3}^2 - \mathbf{z}_3^T \mathbf{z}_3) \mathbf{z}_2 \mathbf{J}_{ha} \mathbf{M}^{-1}}{k_{b2}^2 - \mathbf{z}_2^T \mathbf{z}_2}.$$

The control input $\mathbf{u}$ is designed according to Equation (26) as:

$$\mathbf{u} = \frac{1}{\mathbf{g}_1(\mathbf{x}_3)} \left[\mathbf{g}_2(\mathbf{x}_2) - \hat{\mathbf{f}}_2 + \frac{\dot{k}_{b3}}{k_{b3}} \mathbf{z}_3 - \frac{k_{b3}^2 - \mathbf{z}_3^T \mathbf{z}_3}{\mathbf{z}_3^T} (-k_{31} \ln^{\gamma_1}(\cdot)_3 - k_{32} \ln^{\gamma_2}(\cdot)_3) \right] \quad (27)$$

where k_{31}^* and k_{32}^* are constants greater than zero;

Let $\hat{\mathbf{f}}_2 = \hat{\mathbf{W}}_2^T \boldsymbol{\phi}(\mathbf{X}_2) + \hat{\boldsymbol{\theta}}_2$ and substituting together with Equation (27) into Equation (26) has:

$$\begin{aligned} \dot{\mathbf{V}}_3 &\leq \sum_{i=1}^3 [-k_{i1} \ln^{\gamma_1}(\cdot)_i - k_{i2} \ln^{\gamma_2}(\cdot)_i] - \frac{\sigma_{w1}}{2\eta_{w1}} \tilde{\mathbf{W}}_1^T \tilde{\mathbf{W}}_1 - \frac{\sigma_{\theta 1}}{2\eta_{\theta 1}} \tilde{\boldsymbol{\theta}}_1^T \tilde{\boldsymbol{\theta}}_1 + \\ &\quad \frac{\sigma_{w1}}{2\eta_{w1}} \mathbf{W}_1^{*T} \mathbf{W}_1^* + \frac{\sigma_{\theta 1}}{2\eta_{\theta 1}} \boldsymbol{\theta}_1^{*T} \boldsymbol{\theta}_1^* - \frac{1}{\eta_{w2}} \tilde{\mathbf{W}}_2^T \dot{\tilde{\mathbf{W}}}_2 - \frac{1}{\eta_{\theta 2}} \tilde{\boldsymbol{\theta}}_2^T \dot{\tilde{\boldsymbol{\theta}}}_2 + \\ &\quad \frac{\mathbf{z}_3^T}{k_{b3}^2 - \mathbf{z}_3^T \mathbf{z}_3} (\tilde{\mathbf{W}}_2^T \boldsymbol{\phi}(\mathbf{X}_2) + \tilde{\boldsymbol{\theta}}_2) \end{aligned} \quad (28)$$

According to Equation (28), the neural network weight update rate can be obtained as:

$$\dot{\tilde{\mathbf{W}}}_2 = \eta_{w2} \frac{\mathbf{z}_3^T}{k_{b3}^2 - \mathbf{z}_3^T \mathbf{z}_3} \boldsymbol{\phi}(\mathbf{X}_2) - \sigma_{w2} \tilde{\mathbf{W}}_2 \quad \dot{\tilde{\boldsymbol{\theta}}}_2 = \eta_{\theta 2} \frac{\mathbf{z}_3^T}{k_{b3}^2 - \mathbf{z}_3^T \mathbf{z}_3} - \sigma_{\theta 2} \tilde{\boldsymbol{\theta}}_2 \quad (29)$$

where both σ_{w2} and $\sigma_{\theta2}$ are constants greater than zero; $\mathbf{X}_2 = [\mathbf{q}, \dot{\mathbf{q}}, \mathbf{q}_d, \dot{\mathbf{q}}_d, \boldsymbol{\alpha}_2]$.

3.4. Disturbance Compensator

This subsection designs the disturbance compensator based on the reinforcement learning Actor–Critic architecture to compensate for external disturbances based on Equation (27), i.e., [32]:

$$\mathbf{u} = \frac{1}{\mathbf{g}_1(\mathbf{x}_3)} [\mathbf{g}_2(\mathbf{x}_2) - \hat{\mathbf{f}}_2 + \frac{\dot{\mathbf{k}}_{b3}}{\mathbf{k}_{b3}} \mathbf{z}_3 - \frac{\mathbf{k}_{b3}^2 - \mathbf{z}_3^T \mathbf{z}_3}{\mathbf{z}_3^T} (-\mathbf{k}_{31} \ln^{\gamma_1}(\cdot)_3 - \mathbf{k}_{32} \ln^{\gamma_2}(\cdot)_3)] + \mathbf{F}_D \quad (30)$$

where $\mathbf{F}_D = \mathbf{F}_D(\mathbf{x}(k))$, is the disturbance compensator designed in this paper.

Define the stage cost function as:

$$\delta(k) = \mathbf{Z}_1^T(k) \mathbf{M} \mathbf{Z}_1(k) + \mathbf{u}^T(k) \mathbf{P} \mathbf{u}(k) \quad (31)$$

where $\mathbf{M} \in \mathbb{R}^{3 \times 3}$, $\mathbf{P} \in \mathbb{R}^{3 \times 3}$, are semipositive definite matrices. Parameter $\mathbf{M}$ serves to fully account for the impact of error on the system's control accuracy. Meanwhile, to prevent the output signal from becoming excessively large, parameter $\mathbf{P}$ is added to constrain the output signal.

According to reinforcement learning theory, the long-term reward function is defined as:

$$J(\mathbf{x}(k), \mathbf{F}_D(k)) = \sum_{j=k}^{\infty} \lambda^{j-k} \delta(k) \quad (32)$$

where $0 < \lambda < 1$, representing the reward discount factor, according to Equation (32), the following Bellman equation holds:

$$J^*(\mathbf{x}(k), \mathbf{F}_D(k)) = \delta(k) + \lambda J^*(\mathbf{x}(k+1), \mathbf{F}_D^*(k+1)) \quad (33)$$

where $J^*(\mathbf{x}(k), \mathbf{F}_D(k))$ and $\mathbf{F}_D^*(k)$ are the optimal value function and optimal strategy, respectively, and $\mathbf{F}_D^*(k)$ is expressed as:

$$\mathbf{F}_D^*(k) = \arg \min J^*(\mathbf{x}(k), \mathbf{F}_D(k)) \quad (34)$$

This paper employs the Q-learning algorithm to solve the equations, where the actor–critic neural network utilizes a 2×64 fully connected layer with a hyperbolic tangent activation function.

The state action function $Q(k)$ is defined as based on Equation (32):

$$Q(\mathbf{x}(k), \mathbf{F}_D(k)) = \delta(k) + \sum_{j=1}^{\infty} \lambda^j \delta(k+j) \quad (35)$$

This paper starts with $Q_0(\mathbf{x}(k), \mathbf{F}_D(k)) = 0$ and solves the initial control strategy $\pi_0(k)$ using the following equation:

$$\mathbf{F}_{D0}(k) = \arg \min Q_0(\mathbf{x}(k), \mathbf{F}_D(k)) \quad (36)$$

Then, Q_1 can be computed in the next iteration using the determined strategy $\mathbf{F}_{D0}(k)$:

$$Q_1(\mathbf{x}(k), \mathbf{F}_D(k)) = \delta(k) + \lambda Q_0(\mathbf{x}(k+1), \mathbf{F}_{D0}(k+1)) \quad (37)$$

By means of mathematical induction, there is the following:

$$F_{Di}(k) = \arg \min Q_i(x(k), F_D(k)) \quad (38)$$

$$Q_i(x(k), F_D(k)) = \delta(k) + \lambda Q_i(x(k+1), F_{Di}(k+1)) \quad (39)$$

where i is the iteration index.

By means of Equations (38) and (39), there is the following Bellman equation:

$$Q_{i+1}(x(k), F_D(k)) = r(k) + \lambda \min Q_i(x(k+1), F_D(k+1)) \quad (40)$$

In this paper, critic neural network is used to approximate the $Q(k)$. There is:

$$\hat{Q}_i(k) = \hat{w}_{c2,i}(k) * \sigma(\hat{w}_{c1,i}(k) * x_c(k)) = \hat{w}_{c2,i}(k) * \sigma_{c,i}(k) \quad (41)$$

where $\hat{w}_{c1,i}$ and $\hat{w}_{c2,i}$ represent neural network weight matrices, and $*$ denotes the inner product.

The input of the critic neural network is:

$$x_c(k) = [x_a(k), u(k)]^T \quad (42)$$

where $x_a(k) = [x_1(k), x_2(k), x_3(k), e_1(k), x_d(k)]^T$.

From the Bellman Equation (39), the prediction error of the critic neural network is:

$$e_{c,i+1}(k) = \delta(k) + \lambda \hat{Q}_i(k+1) - \hat{Q}_{i+1}(k) \quad (43)$$

To design the weight update rate, the following minimization approximation cost equation is designed:

$$E_{c,i+1}(k) = e_{c,i+1}^T(k) e_{c,i+1}(k) / 2 \quad (44)$$

In this paper, the gradient descent method is used to update the weights as:

$$\begin{aligned} \Delta \hat{w}_{c1,i+1}(k) &= K_c \left[-\frac{\partial E_{c,i+1}(k)}{\partial \hat{w}_{c1,i+1}(k)} \right] \\ \frac{\partial E_{c,i+1}(k)}{\partial \hat{w}_{c1,i+1}(k)} &= \frac{\partial E_{c,i+1}(k)}{\partial \hat{Q}_i(k)} \frac{\partial \hat{Q}_i(k)}{\partial \sigma_{c,i}(k)} \frac{\partial \sigma_{c,i}(k)}{\partial \hat{w}_{c1,i}(k)} = \lambda e_{c,i+1}(k) \hat{w}_{c2,i}(k) \left[\frac{1}{2} (1 - (\sigma_{c,i}(k))^2) \right] x_c(k) \\ \Delta \hat{w}_{c2,i+1}(k) &= K_c \left[-\frac{\partial E_{c,i+1}(k)}{\partial \hat{w}_{c2,i+1}(k)} \right] \quad \frac{\partial E_{c,i+1}(k)}{\partial \hat{w}_{c2,i+1}(k)} = \frac{\partial E_{c,i+1}(k)}{\partial \hat{Q}_i(k)} \frac{\partial \hat{Q}_i(k)}{\partial \hat{w}_{c2,i}(k)} = \lambda e_{c,i+1}(k) \sigma_{c,i}(k) \end{aligned} \quad (45)$$

where $K_c > 0$ is the weighted learning step.

In this paper, using actor neural network to output the strategy function $F_D(k)$.

$$F_{Di}(k) = \hat{w}_{a2,i}(k) * \sigma(\hat{w}_{a1,i}(k) * x_a(k)) = \hat{w}_{a2,i}(k) * \sigma_{a,i}(k) \quad (46)$$

where $\hat{w}_{a1,i}$ and $\hat{w}_{a2,i}$ represent neural network weight matrices, and $*$ denotes the inner product.

The purpose of the Actor neural network is to minimize the value function $\hat{Q}_i(k)$. Therefore, the following minimization approximation equation is designed:

$$E_{a,i+1}(k) = \hat{Q}_{i+1}(k) \hat{Q}_{i+1}(k) / 2 \quad (47)$$

The weights are updated using the gradient descent method as follows:

$$\begin{aligned}
\Delta \hat{w}_{a1,i+1}(k) &= K_a \left[-\frac{\partial E_{a,i+1}(k)}{\partial \hat{w}_{a1,i+1}(k)} \right] \\
\frac{\partial E_{a,i+1}(k)}{\partial \hat{w}_{a1,i+1}(k)} &= \frac{\partial E_{a,i+1}(k)}{\partial \hat{Q}_i(k)} \left[\frac{\partial \hat{Q}_i(k)}{\partial \pi_i(k)} \right]^T \frac{\partial \pi_i(k)}{\sigma_{a,i}(k)} \frac{\sigma_{a,i}(k)}{\partial s_{a,i}(k)} \frac{\partial s_{a,i}(k)}{\partial \hat{w}_{a1,i}(k)} = \hat{Q}_i(k) \sum_{l=1}^n \sum_{r=1}^N [\hat{w}_{c2,i}(k) \frac{1}{2} (1 - (\sigma_{a,i}(k))^2) \hat{w}_{c1,i}(k)] \frac{1}{2} (1 - (\sigma_{a,i}(k))^2) x_a(k) \\
\Delta \hat{w}_{a2,i+1}(k) &= K_a \left[-\frac{\partial E_{a,i+1}(k)}{\partial \hat{w}_{a2,i+1}(k)} \right] \\
\frac{\partial E_{a,i+1}(k)}{\partial \hat{w}_{a2,i+1}(k)} &= \frac{\partial E_{a,i+1}(k)}{\partial \hat{Q}_i(k)} \left[\frac{\partial \hat{Q}_i(k)}{\partial \pi_i(k)} \right]^T \frac{\partial \pi_i(k)}{\partial \hat{w}_{a2,i}(k)} \\
&= e_a(k) \sum_{l=1}^n \sum_{r=1}^N [\hat{w}_{c2,i}(k) \frac{1}{2} (1 - (\sigma_{a,i}(k))^2) \hat{w}_{c1,i}(k)] \sigma_{a,i}(k)
\end{aligned} \tag{48}$$

where $K_a > 0$ is the weighted learning step.

4. Stability and Optimality Analysis

This subsection first analyzes the global stability of the prescribed-performance controller based on Lyapunov theory, and second, since the disturbance compensator is an externally attached compensation signal, this subsection analyzes the optimality and suboptimality of its compensation signal.

4.1. Global Stability Analysis of Prescribed Performance Controller

Theorem 1. Let a continuous function $V(t)$ be positive definite and satisfy the following differential equation:

$$\dot{V}(t) \leq -\vartheta_a V + \vartheta_b \tag{49}$$

where $\vartheta_a > 0$, $0 < \vartheta_b < 1$, at this time, $V(t)$ satisfies consistent eventual bounded convergence.

Theorem 2. [33] Suppose that there exists a Lyapunov function $V(x)$ with the constants $\alpha, \beta, r_1, r_2, \eta_0$ satisfying $\alpha > 0, \beta > 0, 0 < \eta_0 < \infty, 0 < r_1 < 1$ and $r_2 > 1$. If $\dot{V}(x) \leq -\alpha V(x)^{r_1} - \beta V(x)^{r_2} + \eta_0$ holds, then at a fixed time, $V(x)$ is able to converge to a neighborhood of the equilibrium point, which can be denoted as:

$$\left\{ \lim_{t \rightarrow T} x \mid V(x) \leq \min \left\{ \left(\frac{\eta_0}{\alpha(1-\theta)} \right)^{r_1}, \left(\frac{\eta_0}{\beta(1-\theta)} \right)^{r_2} \right\} \right\} \tag{50}$$

where θ is a constant and satisfies $0 < \theta < 1$. The time required for the system variables to converge to this neighborhood satisfies the following inequality:

$$T \leq \frac{1}{\alpha\theta} \frac{1}{1-r_1} + \frac{1}{\beta\theta} \frac{1}{r_2-1} \tag{51}$$

Proof. By substituting Equation (29) into Equation (28) and based on the comparison principle in Lemma 1, the following is obtained:

$$\begin{aligned}
\dot{V}_3 &\leq \sum_{i=1}^3 [-k_{i1} \ln^{\gamma_1}(\cdot)_i - k_{i2} \ln^{\gamma_2}(\cdot)_i] - \sum_{j=1}^2 \left[\frac{\sigma_{wj}}{2\eta_{wj}} \tilde{W}_j^T \tilde{W}_j + \frac{\sigma_{\theta j}}{2\eta_{\theta j}} \tilde{\theta}_j^T \tilde{\theta}_j \right] \\
&\quad + \sum_{j=1}^2 \left[\frac{\sigma_{wj}}{2\eta_{wj}} \mathbf{W}_j^{*T} \mathbf{W}_j^* + \frac{\sigma_{\theta j}}{2\eta_{\theta j}} \theta_j^{*T} \theta_j^* \right] + \frac{1}{2} \|\varepsilon\|^2 \\
&\leq -\rho V_3 + \zeta
\end{aligned} \tag{52}$$

where

$$\rho = \min\{k_{11}, k_{12}, k_{21}, k_{22}, k_{31}, k_{32}, \sigma_{w1}, \sigma_{w2}, \sigma_{\theta1}, \sigma_{\theta2}\} \quad (53)$$

$$\zeta = \sum_{j=1}^2 \left[\frac{\sigma_{wj}}{2\eta_{wj}} \mathbf{W}_j^* \mathbf{W}_j^* + \frac{\sigma_{\theta j}}{2\eta_{\theta j}} \boldsymbol{\vartheta}_j^* \boldsymbol{\vartheta}_j^* \right] + \frac{1}{2} \|\boldsymbol{\varepsilon}\|^2 \quad (54)$$

There are integrals on both sides of Equation (52):

$$0 \leq V_3 \leq \zeta/\rho + [V_3(0) - \zeta/\rho]e^{-\rho t} \quad (55)$$

When t tends to infinity, there is $0 \leq V_3 \leq \zeta/\rho$. From the fact that V_3 is bounded it follows that the tracking error of the system is always bounded within the time-varying boundary $(-k_{bi}(t), k_{bi}(t))$.

Since V_3 is bounded, the error signals z_3 , $\tilde{\mathbf{W}}_j$ and $\tilde{\boldsymbol{\vartheta}}_j$ are all bounded, i.e., there exist positive constants $\bar{W}_j$ and $\bar{\vartheta}_j$ satisfying $\tilde{\mathbf{W}}_j \leq \bar{W}_j$ and $\tilde{\boldsymbol{\vartheta}}_j \leq \bar{\boldsymbol{\vartheta}}_j$. This leads to the following inequality holding:

$$\begin{aligned} & -\left(\frac{\sigma_{wj}}{2\eta_{wj}} \tilde{\mathbf{W}}_j^T \tilde{\mathbf{W}}_j\right)^{\gamma_1} - \left(\frac{\sigma_{\theta j}}{2\eta_{\theta j}} \tilde{\boldsymbol{\vartheta}}_j^T \tilde{\boldsymbol{\vartheta}}_j\right)^{\gamma_1} + \left(\frac{\sigma_{wj}}{2\eta_{wj}} \mathbf{W}_j^* \mathbf{W}_j^*\right)^{\gamma_1} + \left(\frac{\sigma_{\theta j}}{2\eta_{\theta j}} \boldsymbol{\vartheta}_j^* \boldsymbol{\vartheta}_j^*\right)^{\gamma_1} \geq 0 \\ & -\left(\frac{\sigma_{wj}}{2\eta_{wj}} \tilde{\mathbf{W}}_j^T \tilde{\mathbf{W}}_j\right)^{\gamma_2} - \left(\frac{\sigma_{\theta j}}{2\eta_{\theta j}} \tilde{\boldsymbol{\vartheta}}_j^T \tilde{\boldsymbol{\vartheta}}_j\right)^{\gamma_2} + \left(\frac{\sigma_{wj}}{2\eta_{wj}} \mathbf{W}_j^* \mathbf{W}_j^*\right)^{\gamma_2} + \left(\frac{\sigma_{\theta j}}{2\eta_{\theta j}} \boldsymbol{\vartheta}_j^* \boldsymbol{\vartheta}_j^*\right)^{\gamma_2} \geq 0 \end{aligned} \quad (56)$$

Substituting Equation (56) into Equation (52) yields:

$$\begin{aligned} \dot{V}_3 \leq & \sum_{i=1}^3 [-k_{i1} \ln^{\gamma_1}(\cdot)_i - k_{i2} \ln^{\gamma_2}(\cdot)_i] + \sum_{j=1}^2 \left[\frac{\sigma_{wj}}{2\eta_{wj}} \mathbf{W}_j^* \mathbf{W}_j^* + \frac{\sigma_{\theta j}}{2\eta_{\theta j}} \boldsymbol{\vartheta}_j^* \boldsymbol{\vartheta}_j^* \right] - \\ & \sum_{j=1}^2 \left[\left(\frac{\sigma_{wj}}{2\eta_{wj}} \tilde{\mathbf{W}}_j^T \tilde{\mathbf{W}}_j \right)^{\gamma_1} + \left(\frac{\sigma_{\theta j}}{2\eta_{\theta j}} \tilde{\boldsymbol{\vartheta}}_j^T \tilde{\boldsymbol{\vartheta}}_j \right)^{\gamma_1} \right] - \sum_{j=1}^2 \left[\left(\frac{\sigma_{wj}}{2\eta_{wj}} \tilde{\mathbf{W}}_j^T \tilde{\mathbf{W}}_j \right)^{\gamma_2} + \left(\frac{\sigma_{\theta j}}{2\eta_{\theta j}} \tilde{\boldsymbol{\vartheta}}_j^T \tilde{\boldsymbol{\vartheta}}_j \right)^{\gamma_2} \right] + \\ & \sum_{j=1}^2 \left[\left(\frac{\sigma_{wj}}{2\eta_{wj}} \mathbf{W}_j^* \mathbf{W}_j^* \right)^{\gamma_1} + \left(\frac{\sigma_{\theta j}}{2\eta_{\theta j}} \boldsymbol{\vartheta}_j^* \boldsymbol{\vartheta}_j^* \right)^{\gamma_1} \right] + \sum_{j=1}^2 \left[\left(\frac{\sigma_{wj}}{2\eta_{wj}} \mathbf{W}_j^* \mathbf{W}_j^* \right)^{\gamma_2} + \left(\frac{\sigma_{\theta j}}{2\eta_{\theta j}} \boldsymbol{\vartheta}_j^* \boldsymbol{\vartheta}_j^* \right)^{\gamma_2} \right] + \\ & \frac{1}{2} \|\boldsymbol{\varepsilon}\|^2 \end{aligned} \quad (57)$$

This can be obtained using Lemma 2:

$$\dot{V}_3 \leq -\alpha V_3^{\gamma_1} - \beta V_3^{\gamma_2} + \eta_0 \quad (58)$$

where α , β and η_0 can be expressed, respectively, as:

$$\alpha = \min\{k_{i1}, \sigma_{w1}^{\gamma_1}, \sigma_{\theta1}^{\gamma_1}\} \quad (59)$$

$$\beta = \min\{(2n)^{1-\gamma_2} k_{i2}, (2n)^{1-\gamma_2} \sigma_{w2}^{\gamma_2}, (2n)^{1-\gamma_2} \sigma_{\theta2}^{\gamma_2}\} \quad (60)$$

$$\begin{aligned} \eta_0 = & \sum_{j=1}^2 \left[\left(\frac{\sigma_{wj}}{2\eta_{wj}} \mathbf{W}_j^* \mathbf{W}_j^* \right)^{\gamma_1} + \left(\frac{\sigma_{\theta j}}{2\eta_{\theta j}} \boldsymbol{\vartheta}_j^* \boldsymbol{\vartheta}_j^* \right)^{\gamma_1} + \left(\frac{\sigma_{wj}}{2\eta_{wj}} \mathbf{W}_j^* \mathbf{W}_j^* \right)^{\gamma_2} \right] + \\ & \sum_{j=1}^2 \left[\left(\frac{\sigma_{\theta j}}{2\eta_{\theta j}} \boldsymbol{\vartheta}_j^* \boldsymbol{\vartheta}_j^* \right)^{\gamma_2} + \frac{\sigma_{wj}}{2\eta_{wj}} \mathbf{W}_j^* \mathbf{W}_j^* + \frac{\sigma_{\theta j}}{2\eta_{\theta j}} \boldsymbol{\vartheta}_j^* \boldsymbol{\vartheta}_j^* \right] + \frac{1}{2} \|\boldsymbol{\varepsilon}\|^2 \end{aligned} \quad (61)$$

Using Lemma 2, it can be obtained that the controller proposed in this section can converge in a neighborhood near the zero point in a fixed time, where the neighborhood expression is $\left\{ \lim_{t \rightarrow T} \mathbf{x} | V(\mathbf{x}) \leq \min\{(\eta_0/k_{i1}(1-\theta))^{\gamma_1}, (\eta_0/k_{i2}(1-\theta))^{\gamma_2}\} \right\}$ and the convergence time expression is $T = 1/k_{i1}\theta(1-\gamma_1) + 1/k_{i2}\theta(\gamma_2-1)$. $\square$

4.2. Suboptimality Analysis of Disturbance Compensator

Next, the optimality and suboptimality of the compensation signal are analyzed:

Define the value function under any strategy as:

$$V_{i+1}(\mathbf{x}(k), F_D(k)) = r(k) + \lambda V_i(\mathbf{x}(k+1), F_D(k+1)) \quad (62)$$

According to Equation (40), $Q_{i+1}(\mathbf{x}(k), F_D(k))$ is the result of minimizing the value function, while $V_{i+1}(\mathbf{x}(k), F_D(k))$ is the result of the value function of an arbitrary action, so there is:

$$Q_{i+1}(\mathbf{x}(k), F_D(k)) \leq V_{i+1}(\mathbf{x}(k), F_D(k)) \quad (63)$$

When $Q_0 = V_0 = 0$, this holds for both $i = 0, 1, 2 \dots$

According to Equation (62) there is:

$$\begin{aligned} V_{i+1}(\mathbf{x}(k), F_D(k)) - V_i(\mathbf{x}(k), F_D(k)) &= \lambda V_i(\mathbf{x}(k+1), F_D(k+1)) - \lambda V_{i-1}(\mathbf{x}(k+1), F_D(k+1)) \\ &= \lambda^2 V_{i-1}(\mathbf{x}(k+2), F_D(k+2)) - V_{i-2}(\mathbf{x}(k+2), F_D(k+2)) \\ &= \lambda^i V_1(\mathbf{x}(k+i), F_D(k+i)) - V_0(\mathbf{x}(k+i), F_D(k+i)) \\ &= \lambda^i V_1(\mathbf{x}(k+i), F_D(k+i)) \end{aligned} \quad (64)$$

Therefore, the value function is expressed iteratively as:

$$\begin{aligned} V_{i+1}(\mathbf{x}(k), F_D(k)) &= \lambda^i V_1(\mathbf{x}(k+i), F_D(k+i)) + V_i(\mathbf{x}(k), F_D(k)) \\ &= \lambda^i V_1(\mathbf{x}(k+i), F_D(k+i)) + \lambda^{i-1} V_1(\mathbf{x}(k+i-1), F_D(k+i-1)) \\ &\quad + V_{i-1}(\mathbf{x}(k), F_D(k)) \\ &= \lambda^i V_1(\mathbf{x}(k+i), F_D(k+i)) + \lambda^{i-1} V_1(\mathbf{x}(k+i-1), F_D(k+i-1)) \\ &\quad + \dots + \lambda V_1(\mathbf{x}(k+1), F_D(k+1)) + V_1(\mathbf{x}(k), F_D(k)) \\ &= \sum_{j=0}^i \lambda^j V_1(\mathbf{x}(k+j), F_D(k+j)) \end{aligned} \quad (65)$$

According to Equation (62) there is:

$$V_{i+1}(\mathbf{x}(k), F_D(k)) \leq \sum_{j=0}^{\infty} \lambda^j r(\mathbf{x}(k+j), F_D(k+j)) \quad (66)$$

Due to the positive characterization of the stage cost function $\delta(\mathbf{x}(k+i), F_D(k+i))$, it is assumed that $0 \leq \delta(\mathbf{x}(k+i), F_D(k+i)) \leq C$, there exists an upper bound $\mathbb{C}$.

$$\sum_{j=0}^{\infty} \lambda^j \delta(\mathbf{x}(k+j), F_D(k+j)) \leq \sum_{j=0}^{\infty} \lambda^j C \leq \sum_{j=0}^{\infty} \lambda^j \mathbb{C} \quad (67)$$

There is through Equation (63):

$$Q_{i+1}(\mathbf{x}(k), F_D(k)) \leq V_{i+1}(\mathbf{x}(k), F_D(k)) \leq \mathbb{C} \quad (68)$$

Next, the optimality of the value function and control policy will be proven.

Since $Q_0 = V_0 = 0$, it follows that when $i = 0$, there is $Q_0(\mathbf{x}(k), F_D(k)) \leq Q_1(\mathbf{x}(k), F_D(k)) = \delta(k)$, there is:

$$Q_1(\mathbf{x}(k), F_D(k)) - V_0 = \delta(k) \geq 0 \quad (69)$$

Thus, by mathematical induction, there is:

$$V_{i-1}(\mathbf{x}(k), F_D(k)) \leq Q_i(\mathbf{x}(k), F_D(k)) \quad (70)$$

Further, there is:

$$\begin{aligned} & Q_{i+1}(x(k), F_D(k)) - V_i(x(k), F_D(k)) \\ &= \lambda [Q_i(x(k), F_D(k)) - V_{i-1}(x(k), F_D(k))] \geq 0 \end{aligned} \quad (71)$$

According to inequality (63), there is:

$$Q_i(x(k), F_D(k)) \leq V_i(x(k), F_D(k)) \leq Q_{i+1}(x(k), F_D(k)) \quad (72)$$

For any control strategy $F_D(k)$, there is the following inequality:

$$Q_i(x(k), F_D(k)) \leq \delta(k) + \lambda Q_{i-1}(x(k+1), F_D(k+1)) \quad (73)$$

According to Equation (72),

$$Q_i(x(k), F_D(k)) \leq \delta(k) + \lambda Q_\infty(x(k+1), F_D(k+1)) \quad (74)$$

It applies to $i \rightarrow \infty$:

$$Q_\infty(x(k), F_D(k)) \leq \delta(k) + \lambda Q_\infty(x(k+1), F_D(k+1)) \quad (75)$$

When there is an arbitrary control strategy $\pi(k+1)$, there is:

$$Q_\infty(x(k), F_D(k)) \leq \delta(k) + \lambda \min_{\pi(k+1)} Q_\infty(x(k+1), F_D(k+1)) \quad (76)$$

Also, according to Equations (40) and (72),

$$Q_\infty(x(k), F_D(k)) \geq \delta(k) + \lambda Q_\infty(x(k+1), F_D(k+1)) \quad (77)$$

Obviously, there is:

$$Q_\infty(x(k), F_D(k)) = \delta(k) + \lambda \min_{\pi(k+1)} Q_\infty(x(k+1), F_D(k+1)) = \delta(k) + \lambda Q_\infty(x(k+1), F_D(k+1)) \quad (78)$$

The optimal control strategy can be solved by the following equation:

$$F_{D\infty}(k) = \operatorname{argmin} Q_\infty(x(k), F_D(k)) \quad (79)$$

Therefore, according to Equations (33) and (34), there is:

$$Q_\infty(x(k), F_D(k)) = J^*(x(k), F_D(k)) \quad F_{D\infty}(k) = F_D^*(k) \quad (80)$$

However, due to the inherent error in neural networks, the optimal convergence properties of the value function and control policy only hold under ideal conditions. Therefore, considering the approximation errors in Equations (41) and (46),

$$\hat{F}_{Di}(k) = \operatorname{argmin} \hat{Q}_i(x(k), F_D(k)) + \varepsilon_i(x(k)) \quad (81)$$

$$\hat{Q}_{i+1}(x(k), \hat{F}_D(k)) = \delta(k) + \lambda \hat{Q}_i(x(k+1), \hat{F}_D(k+1)) + \delta_i(x(k), \hat{F}_D(k)) \quad (82)$$

where $\varepsilon_i(x(k))$ and $\delta_i(x(k), \hat{F}_D(k))$ are the estimation errors and $\varepsilon_i(0) = \delta_i(0, 0) = 0$.

Next, a new convergence criterion will be established at each iteration, taking into account the error, i.e., the value iterates converge to a neighborhood of the optimal value function, and the iterative value function is:

$$Q_{t,i}(x(k), F_D(k)) = \delta(k) + \lambda \min_{\pi(k+1)} \hat{Q}_{i-1}(x(k+1), F_D(k+1)) \quad (83)$$

where $\hat{Q}_0(x(k), F_D(k)) = \hat{Q}_{t,0}(x(k), F_D(k)) = 0$ and there exists a constant $1 \leq \omega < \infty$ such that the following inequality holds:

$$\hat{Q}_i(x(k), F_D(k)) \leq \omega Q_{t,i}(x(k), F_D(k)) \quad (84)$$

Defining the constant $0 < \nu < \infty$, the following inequality holds:

$$J^*(x(k+1), F_D(k+1)) \leq \omega \delta(k) \quad (85)$$

According to Equation (84), when $i = 0$,

$$\hat{Q}_0(x(k), F_D(k)) \leq \omega J^*(x(k), F_D(k)) \quad (86)$$

When $i = 1$,

$$\begin{aligned} Q_{t,1}(x(k), F_D(k)) &= \delta(k) + \lambda \min_{\pi(k+1)} \hat{Q}_0(x(k+1), F_D(k+1)) \\ &\leq \delta(k) + \lambda \omega J^*(x(k+1), F_D(k+1)) \\ &\leq (1 + \nu(\omega - 1)/(\omega + 1))\delta(k) + \\ &\quad \lambda(\omega - (\omega - 1)/(\omega + 1))J^*(x(k+1), F_D(k+1)) \\ &= (1 + \nu(\omega - 1)/(\omega + 1))J^*(x(k+1), F_D(k+1)) \end{aligned} \quad (87)$$

According to Equations (83) and (85),

$$\lambda J^*(x(k+1), F_D(k+1)) \leq J^*(x(k+1), F_D(k+1)) \leq \nu \delta(k) \quad (88)$$

Therefore, the following expression holds:

$$\hat{Q}_1(x(k), F_D(k)) \leq \omega(1 + \nu(\omega - 1)/(\omega + 1))J^*(x(k), F_D(k)) \quad (89)$$

Define the following equation:

$$\beta = \sum_{j=1}^{i-1} (\nu^{j-1} \omega^{j-1} (\omega - 1)) / (\nu + 1)^j \quad \omega = (\nu^{j-1} \omega^{j-1} (\omega - 1)) / (\nu + 1)^i \quad (90)$$

There is:

$$\begin{aligned} Q_{t,i}(x(k), \pi(k)) &= \delta(k) + \lambda \min_{\pi(k+1)} \hat{Q}_{i-1}(x(k+1), F_D(k+1)) \\ &\leq \delta(k) + \lambda \omega(1 + \nu\beta)J^*(x(k+1), F_D(k+1)) \\ &\leq (1 + \omega\beta + \omega\omega)\delta(k) + \lambda(\omega(1 + \nu\beta) - \beta - \omega)J^*(x(k+1), F_D(k+1)) \\ &= (1 + \sum_{j=1}^i \nu^j \omega^{j-1} (\omega - 1) / (\nu + 1)^j)J^*(x(k+1), F_D(k+1)) \end{aligned} \quad (91)$$

According to Equation (84), by mathematical induction,

$$\hat{Q}_i(x(k), F_D(k)) \leq (1 + \sum_{j=0}^i \nu^j \omega^{j-1} (\omega - 1) / (\nu + 1)^j)J^*(x(k+1), F_D(k+1)) \quad (92)$$

texture of upper computer-controller-CAN communication-lower computer-sensor, and the real-time data acquisition and control of the system is based on dSPACE (DS4302 and DS2004 Germany) and MATLAB (2015b) with a sampling period of 10 ms. The sampling period is 10 ms.

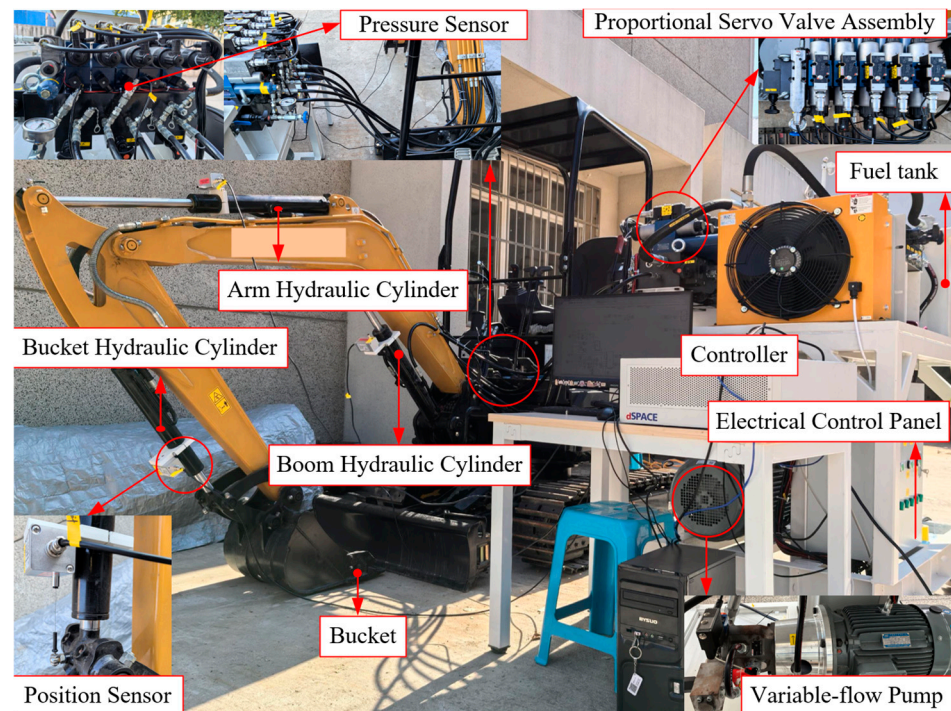


Figure 5. Hydraulic manipulator experimental platform.

As the controller-related parameters need the real-time angle of each joint, the control algorithm joins the hydraulic cylinder and the joint angle conversion equation. The length of the boom cylinder with the angle change equation is:

$$l_1 = \sqrt{L_{AE}^2 + L_{AF}^2 - 2L_{AE}L_{AF} \cos(\angle EAN + \angle BAF + \theta_1)} \quad (97)$$

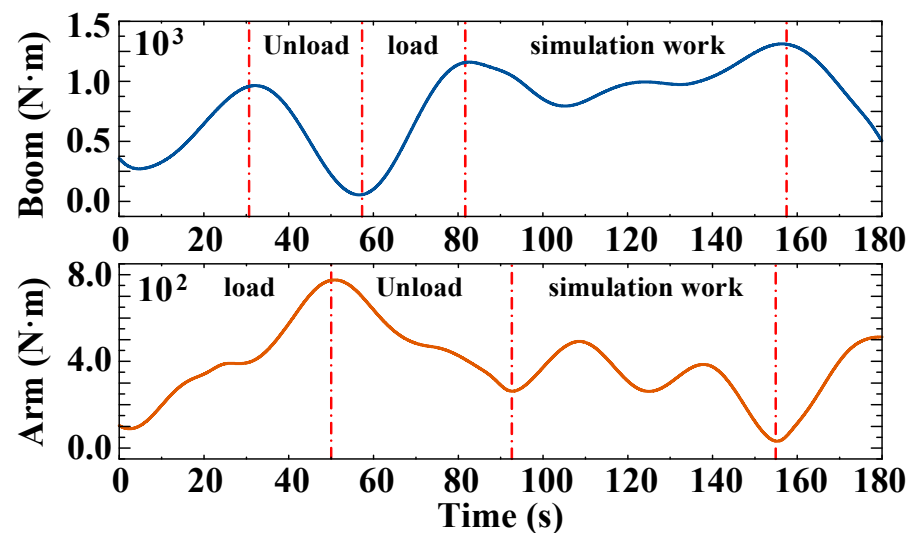
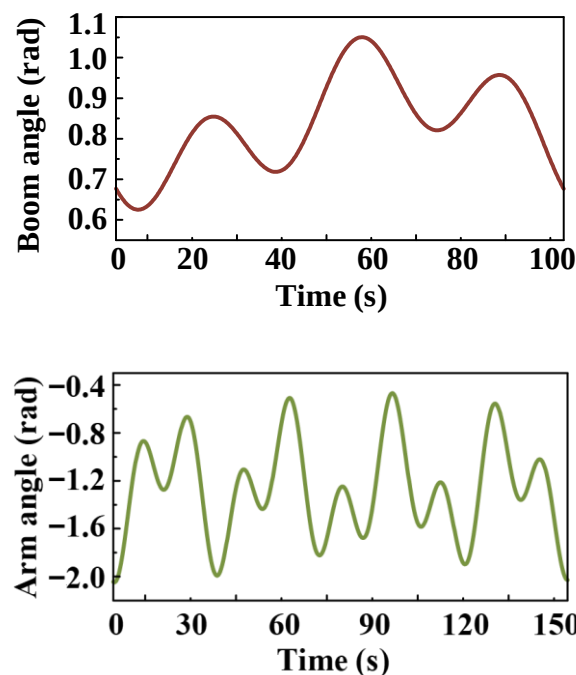
The length of the arm cylinder with the angle change equation is:

$$\begin{cases} \angle GBH = \arccos\left(\frac{L_{BG}^2 + L_{BH}^2 - l_2^2}{2L_{BG}L_{BH}}\right) \\ \theta_2 = \pi - \angle ABG - \angle GBH - \angle HBC \\ l_2 = \sqrt{L_{BG}^2 + L_{BH}^2 - 2L_{BG}L_{BH} \cos(\pi - \angle ABG - \angle HBC - \theta_2)} \end{cases} \quad (98)$$

Since the neural network does not converge at the initial stage, it is necessary to train the disturbance compensator offline and deploy it online to the controller. This enables fine-tuning of network weights to estimate and compensate for external disturbances. The relevant structural parameters of the manipulator are shown in Table 1. By constructing the manipulator model using Equation (6) and introducing irregular external disturbances as depicted in Figure 6, the network weights are ensured to learn real disturbances of the same magnitude. Given the excitation trajectory shown in Figure 7, the disturbance compensator is trained using the performance-specific controller designed in this paper for tracking. When the tracking error $Z_1 \leq E_A$, the training is stopped and the trained actor and critic neural network weights are deployed in dSPACE. Taking the boom controller training as an example, the training process is shown in Figure 8, consisting of 30 iterations. Figure 8a represents the first iteration, while Figure 8b shows the last iteration.

Table 1. Some parameters of the hydraulic manipulator.

Parameter	Value	Parameter	Value	Parameter	Value	Parameter	Value	Parameter	Value
A_{11}	$3.12 \times 10^{-3} \text{ m}^2$	A_{12}	$2.4 \times 10^{-3} \text{ m}^2$	A_{21}	$2.38 \times 10^{-3} \text{ m}^2$	A_{22}	$1.7 \times 10^{-3} \text{ m}^2$	g	9.8 m/s^2
P_r	0 Pa	I_1	$28 \text{ kg}\cdot\text{m}^2$	I_2	$10.2 \text{ kg}\cdot\text{m}^2$	I_3	$1.95 \text{ kg}\cdot\text{m}^2$	m_1	86.6 kg
m_2	64 kg	m_3	43 kg	L_1	1.806 m	L_2	1.151 m	L_3	0.535 m
β_e	1×10^9	k_{v1}	3.3×10^{-8}	k_{v2}	3.3×10^{-8}				

**Figure 6.** External disturbance.**Figure 7.** Excitation trajectory.

In this subsection, the superiority of the proposed control algorithm is verified through comparison and ablation experiments, firstly, the PI controller is compared with the prescribed performance controller based on neural network compensation to verify the effectiveness of the prescribed performance function and the neural network in the controller, and secondly, the prescribed performance controller is compared with the addition of the trained disturbance compensator for ablation experiments to verify the effectiveness of the disturbance compensator, and the relevant parameters are shown below:

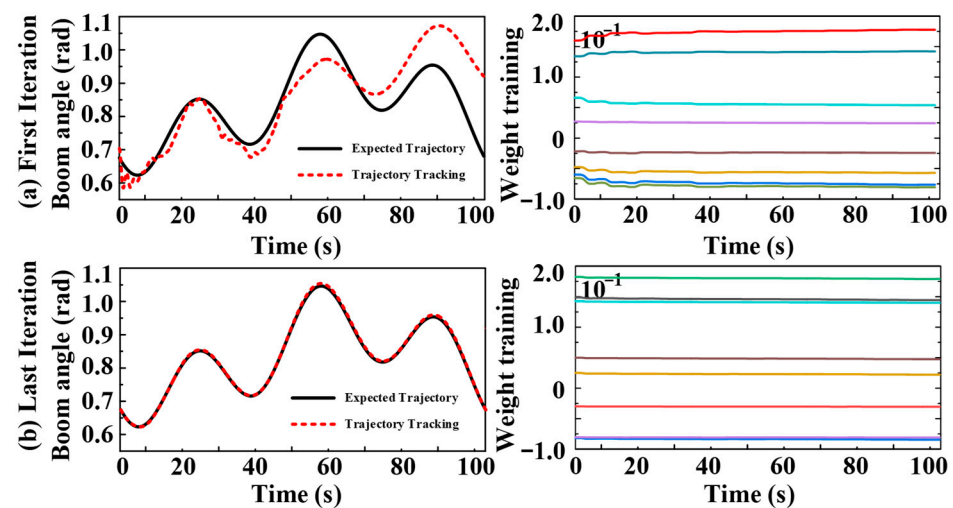


Figure 8. The training process of arm disturbance compensator.

C1: This is a typical Proportional Integral (PI) controller with control gain $k_p = \text{diag}\{55, 85\}$, $k_i = \text{diag}\{35, 75\}$.

C2: This is a prescribed performance controller based on neural network compensation. The constraint boundaries of the moving arm and the boom are the same, and the constraint boundaries are set as: $k_{a1} = 0.8e^{-0.7t} + 0.8$, $k_{a2} = 1.3e^{-0.8t} + 1.6$, $k_{a3} = 1.8e^{-0.4t} + 2.1$. Select controller parameters through a combination of simulation and experimentation: $k_{11}^* = k_{21}^* = k_{31}^* = \text{diag}\{16, 16\}$, $k_{12}^* = k_{22}^* = k_{32}^* = \{24, 24\}$, $\gamma_1 = 1.2$, $\gamma_2 = 3$. the number of nodes in the hidden layer of the RBF neural network is set to 32, the initial value of the neural network weights $W_{1i0} = W_{2i0} = 0.2$, $i = 1, 2, \dots, 32$, the initial value of the approximation error $\vartheta_{10} = \vartheta_{20} = 0$, $\eta_{w1} = \eta_{w2} = 1$, $\eta_{\vartheta1} = \eta_{\vartheta2} = 210$, $\sigma_{\vartheta1} = \sigma_{\vartheta2} = 0.16$, center value interval is $[-1.5, 1.5]$. The relevant structural parameters of the manipulator are shown in Table 1.

C3: This is the prescribed performance controller with the addition of the disturbance compensator, and its training parameters are: $h_c = h_a = 0.01$, $\lambda = \text{diag}[0.9, 0.9, 0.9]$. Its base control parameters are the same as those in C2 to ensure fairness in comparison.

The experiment is used to simulate the normal operation of the hydraulic manipulator arm by giving the time-varying sinusoidal signals of the boom and arm, while 20 kg sandbags are added to the tip of the bucket teeth to simulate the load during normal operation. The signals are as follows:

$$q_d = [10 \sin(0.2t + 8) + 25, 25 \sin(0.2t + 10.5) - 59.8]^T \quad (99)$$

Figure 9 shows the coverage effect of desired trajectories and tracking trajectories under three controllers for the boom and arm. Figure 10 depicts the angular tracking error of the boom and arm. Figure 11 illustrates the compensation signal for the boom and arm generated by the disturbance compensator. Figure 12 presents the unmodeled error estimation compensation value for the boom and arm derived from the neural network. Based on experimental results, it is evident that for continuously varying sinusoidal velocity signals, the tracking performance of C3 and C2 significantly outperforms that of C1. This improvement stems from the fact that C2 incorporates a predefined performance function to constrain system errors while ensuring stability, thereby mitigating the manipulator's inherent joint strong coupling characteristics and hydraulic system parameter uncertainties to a certain extent. Concurrently, the RBF neural network estimates and compensates for internal system uncertainties, thereby mitigating the manipulator's inherent joint strong coupling characteristics and hydraulic, which the PI controller cannot resolve.

Consequently, C2 achieves significantly higher control accuracy than C1. Furthermore, as shown in the error results of Figure 10, C3 demonstrates markedly superior control accuracy compared to C2. This is because C3 incorporates an additional disturbance compensator on top of C2, enabling compensation for disturbances encountered during the hydraulic manipulator's operation. In this experiment, a 20 kg sandbag was introduced to simulate external disturbances. However, due to the inherent robustness of the specified performance controller, this robustness effectively suppresses disturbances during the motion of the heavy-duty hydraulic manipulator. Due to its substantial mass and the time-varying characteristics of the inertia matrix, Coriolis torque matrix, and gravity matrix, the improvement in control accuracy achieved by C3 over C2 is less pronounced than the improvement C2 achieves over C1.

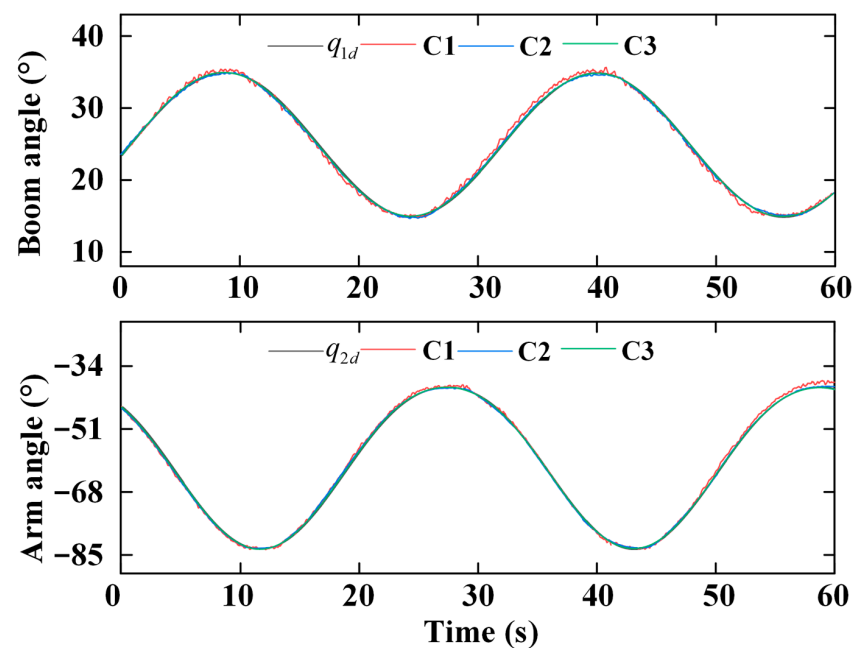


Figure 9. Angle tracking of boom/arm under different controllers.

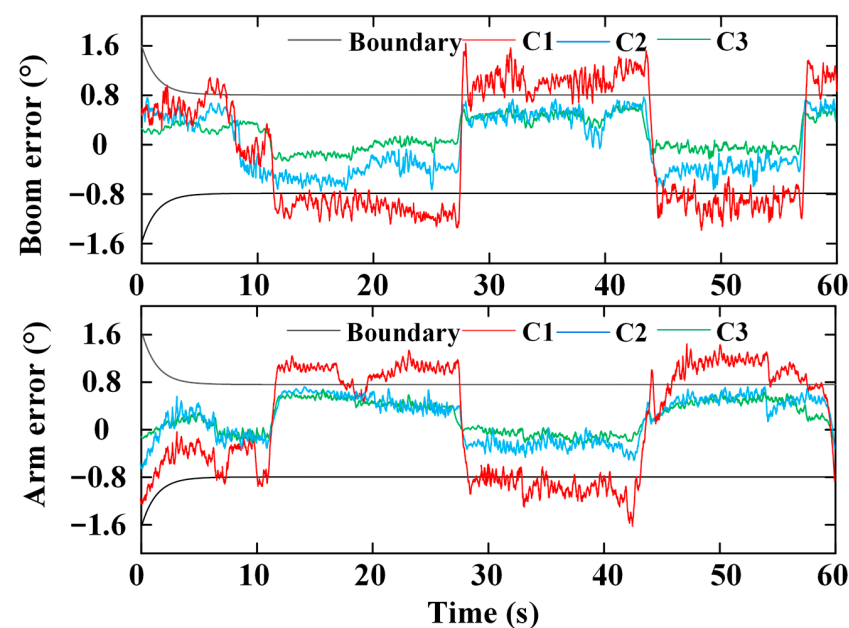


Figure 10. Angle tracking error of boom/arm under different controllers.

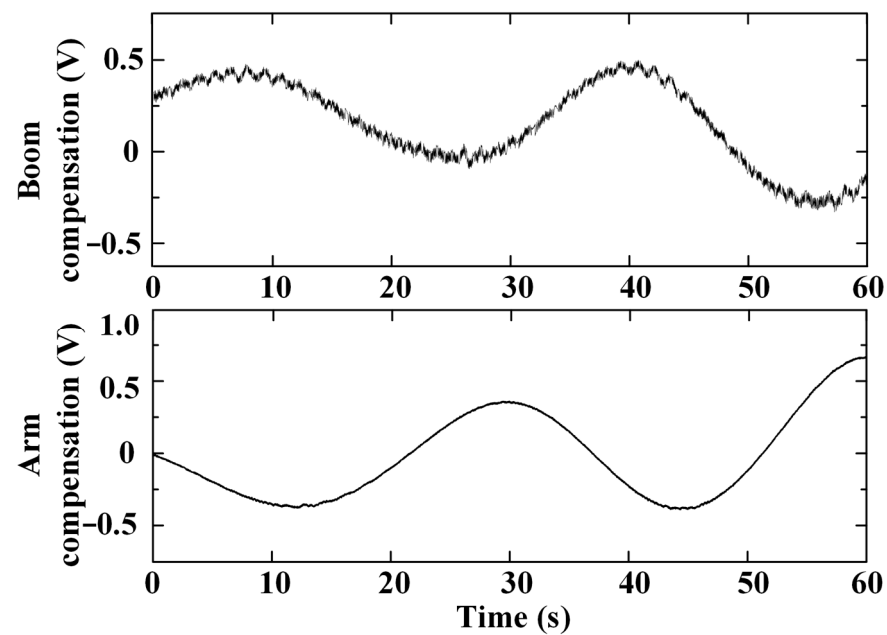


Figure 11. Disturbance compensator compensation signal.

To further analyze the performance of each controller, this paper employs widely used evaluation metrics: the maximum absolute value, mean, and standard deviation of tracking error as assessment criteria [34], expressed as:

$$M_e = \max_{i=1,\dots,N} \{|z_1(i)|\} \quad \mu = \frac{1}{N} \sum_{i=1}^N |z_1(i)| \quad \sigma = \sqrt{\frac{1}{N} \sum_{i=1}^N [|z_1(i) - \mu|]^2} \quad (100)$$

where M_e is the maximum absolute value error; μ is the mean error; and σ is the standard deviation.

The results are shown in Table 2. Based on the quantified data, it is evident that due to the performance controller imposing boundary constraints while the neural network compensates for unmodeled internal system errors, coupled with the control architecture's inherent robustness, suppressing certain disturbances, C2 achieved a 61.3% and 61.8% reduction in average tracking error compared to C1 for the boom and arm, respectively. The standard deviation decreased by 59.7% and 60.8%, respectively. Furthermore, since C3 incorporates a disturbance compensator on top of C2, its control accuracy improvement surpasses that of C2. Specifically, C3 reduced the mean tracking error for the boom and arm by 65% and 64.7% compared to C1. The standard deviation decreased by 63% and 62.8%, respectively. To more clearly demonstrate the effectiveness of the disturbance compensator, comparing the control accuracy between C2 and C3 shows that C3 reduced the average tracking error compared to C2 by 9.6% and 7.5% for the boom and arm, respectively. The standard deviation decreased by 8.2% and 5.2%, respectively, fully validating the effectiveness of the disturbance compensator.

Table 2. Control performance index of each controller.

Joint	Controller	M_e	μ	σ
Boom	C1	1.6104	0.6107	0.5701
	C2	0.7913	0.2364	0.2296
	C3	0.7026	0.2136	0.2108
Arm	C1	1.6073	0.6046	0.5658
	C2	0.7924	0.2309	0.2217
	C3	0.7043	0.2136	0.2102

Through the aforementioned experiments, the high-precision motion control algorithm for hydraulic manipulators proposed in this paper demonstrates that by defining a performance function, errors can be stabilized within a preset range. Concurrently, the RBF neural network compensates for internal system uncertainties, achieving a significant improvement in precision compared to PID control algorithms. Furthermore, a disturbance compensator based on the Actor–Critic reinforcement learning architecture was introduced. Ablation experiments confirmed the effectiveness of this disturbance compensator.

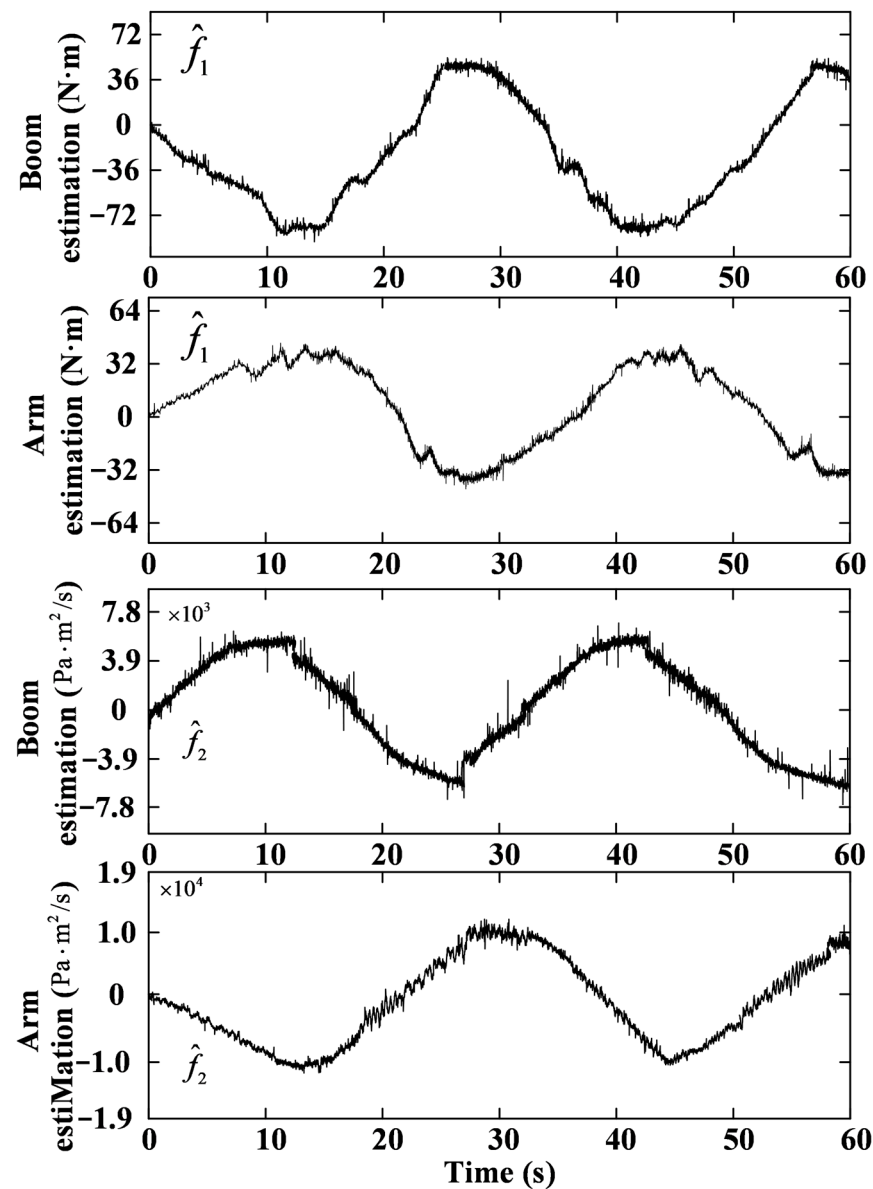


Figure 12. Neural network estimation.

6. Conclusions

This paper addresses the challenge of achieving high-precision motion control for hydraulic manipulators with strong electromechanical-hydraulic coupling. A high-precision motion control algorithm with a dual compensation mechanism and specified performance is designed, significantly enhancing the system's control accuracy. The following key conclusions are drawn:

- (1) Within a performance-based control architecture based on the inverse method, a neural network is introduced to estimate and compensate for internal system uncertainties:

unmodeled mechanical and hydraulic errors. The network update rate is obtained through a global stability proof, significantly enhancing system control accuracy.

- (2) An Actor–Critic reinforcement learning architecture is employed to design a disturbance compensator for external system uncertainties: external disturbances. The network’s online update rate was obtained using local stability, further enhancing the system’s motion control precision.
- (3) Through a hydraulic manipulator experimental platform, the proposed control algorithm demonstrated a 60–65% improvement in control accuracy compared to the PID algorithm. Ablation experiments confirmed that the disturbance compensator designed using the reinforcement learning Actor–Critic architecture further enhances the hydraulic manipulator’s control accuracy by 7–10%, validating its effectiveness.

Furthermore, due to the network’s local updates and proven optimality, the disturbance compensator is not only applicable to the proposed algorithm but can also be extended to adaptive robust control algorithms to further enhance system motion control accuracy.

Author Contributions: Methodology, X.Q.; Software, Y.L.; Validation, Y.L.; Investigation, Y.L.; Resources, Y.L.; Writing—original draft, Y.L.; Writing—review & editing, X.Q.; Supervision, X.Q. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Conflicts of Interest: Author Yuhe Li, employed by the Tianjin Research Institute of Construction Machinery Co., Ltd., Tianjin 300409, declares no financial or non-financial conflicts of interest related to the submitted work.

References

1. Phan, V.D.; Ahn, K.K. Fault-Tolerant Control for an Electro-Hydraulic Servo System with Sensor Fault Compensation and Disturbance Rejection. *Nonlinear Dyn.* **2023**, *111*, 10131–10146. [\[CrossRef\]](#)
2. Islam, R.U.; Iqbal, J.; Manzoor, S.; Khalid, A.; Khan, S. An Autonomous Image-Guided Robotic System Simulating Industrial Applications. In Proceedings of the 2012 7th International Conference on System of Systems Engineering (SoSE), Genova, Italy, 16–19 July 2012; pp. 344–349.
3. Xiang, Y.; Li, R.; Brach, C.; Liu, X.; Geimer, M. A Novel Algorithm for Hydrostatic-Mechanical Mobile Machines with a Dual-Clutch Transmission. *Energies* **2022**, *15*, 2095. [\[CrossRef\]](#)
4. Feng, H.; Yin, C.; Ma, W.; Yu, H.; Cao, D. Parameters Identification and Trajectory Control for a Hydraulic System. *ISA Trans.* **2019**, *92*, 228–240. [\[CrossRef\]](#) [\[PubMed\]](#)
5. Lv, L.; Chen, Z.; Yao, B. High Precision and High Efficiency Control of Pump and Valves Combined Hydraulic System. In Proceedings of the 2018 IEEE 15th International Workshop on Advanced Motion Control (AMC), Tokyo, Japan, 9–11 March 2018; pp. 391–396.
6. Li, C.; Ding, R.; Cheng, M.; Chen, Z.; Yao, B. Accurate Motion Control of an Independent Metering Actuator with Adaptive Robust Compensation of Uncertainties in Pressure Dynamics. *IEEE/ASME Trans. Mechatron.* **2024**, *29*, 3877–3889. [\[CrossRef\]](#)
7. Sha, Y.; Wang, Q.; He, X.; Zhu, X.; Yang, F.; Du, M. Trajectory Tracking Control of Hydraulic Manipulator Based on Fuzzy Compensation. *Inf. Control* **2021**, *50*, 184–194.
8. Khan, O.; Pervaiz, M.; Ahmad, E.; Iqbal, J. On the Derivation of Novel Model and Sophisticated Control of Flexible Joint Manipulator. *Rev. Roum. Sci. Techn.-Électrotechn. Énerg.* **2017**, *62*, 103–108.
9. Zhu, Y.; Qiao, J.; Guo, L. Adaptive Sliding Mode Disturbance Observer-Based Composite Control with Prescribed Performance of Space Manipulators for Target Capturing. *IEEE Trans. Ind. Electron.* **2019**, *66*, 1973–1983. [\[CrossRef\]](#)
10. Sun, Y.; Wan, Y.; Ma, H.; Liang, X. Compensation Control of Hydraulic Manipulator under Pressure Shock Disturbance. *Nonlinear Dyn.* **2023**, *111*, 11153–11169. [\[CrossRef\]](#)

11. Wang, Q.; Shen, Y.; Wang, J.; Su, F.; Feng, C.; Li, X. Free-Shape Contour Control for Excavators Based on Cross-Coupling and Double Error Pre-Compensation. *Autom. Constr.* **2024**, *160*, 105336. [\[CrossRef\]](#)
12. Li, C.; Lyu, L.; Helian, B.; Chen, Z.; Yao, B. Precision Motion Control of an Independent Metering Hydraulic System With Nonlinear Flow Modeling and Compensation. *IEEE Trans. Ind. Electron.* **2022**, *69*, 7088–7098. [\[CrossRef\]](#)
13. Guo, X.; He, X.; Wang, H.; Liu, H.; Sun, X. Model Feedforward Compensation Active Disturbance Rejection Control for Heavy Hydraulic Manipulator. *J. S. China Univ. Technol.* **2024**, *52*, 59–67.
14. Yao, J.; Deng, W. Active Disturbance Rejection Adaptive Control of Hydraulic Servo Systems. *IEEE Trans. Ind. Electron.* **2017**, *64*, 8023–8032. [\[CrossRef\]](#)
15. Zhang, X.; Shi, G. Dual Extended State Observer-Based Adaptive Dynamic Surface Control for a Hydraulic Manipulator with Actuator Dynamics. *Mech. Mach. Theory* **2022**, *169*, 104647. [\[CrossRef\]](#)
16. Shi, D.; Zhang, J.; Sun, Z.; Shen, G.; Xia, Y. Composite Trajectory Tracking Control for Robot Manipulator with Active Disturbance Rejection. *Control Eng. Pract.* **2021**, *106*, 104670. [\[CrossRef\]](#)
17. Liang, X.; Yao, Z.; Deng, W.; Yao, J. Adaptive Neural Network Finite-Time Tracking Control for Uncertain Hydraulic Manipulators. *IEEE/ASME Trans. Mechatron.* **2025**, *30*, 645–656. [\[CrossRef\]](#)
18. Zhao, Z.; Feng, K.; Wang, X.; Yang, C.; Li, X.; Hong, K.-S. Adaptive NN Control for a Flexible Manipulator with Input Backlash and Output Constraint. *IEEE Trans. Syst. Man Cybern. Syst.* **2024**, *54*, 7472–7481. [\[CrossRef\]](#)
19. Rouvinen, A.; Handroos, H. Deflection Compensation of a Flexible Hydraulic Manipulator Utilizing Neural Networks. *Mechatronics* **1997**, *7*, 355–368. [\[CrossRef\]](#)
20. Tran, D.; Truong, H.; Ahn, K.K. Adaptive Nonsingular Fast Terminal Sliding Mode Control of Robotic Manipulator Based Neural Network Approach. *Int. J. Precis. Eng. Manuf.* **2021**, *22*, 417–429. [\[CrossRef\]](#)
21. Guo, Q.; Zhang, Y.; Celler, B.G.; Su, S.W. Neural Adaptive Backstepping Control of a Robotic Manipulator with Prescribed Performance Constraint. *IEEE Trans. Neural Netw. Learn. Syst.* **2019**, *30*, 3572–3583. [\[CrossRef\]](#)
22. Fan, S.; Wang, S.; Wang, X.; Wang, Q.; Liu, D. Preset Adaptive Finite-Time Control for Electro-Hydrostatic Actuator in Wide Temperature Range. *J. Beijing Univ. Aeronaut. Astronaut.* **2025**, 1–18. [\[CrossRef\]](#)
23. Deng, W.; Zhou, H.; Zhou, J.; Yao, J. Neural Network-Based Adaptive Asymptotic Prescribed Performance Tracking Control of Hydraulic Manipulators. *IEEE Trans. Syst. Man Cybern. Syst.* **2023**, *53*, 285–295. [\[CrossRef\]](#)
24. Wang, S.; Yu, H.; Yu, J.; Na, J.; Ren, X. Neural-Network-Based Adaptive Funnel Control for Servo Mechanisms with Unknown Dead-Zone. *IEEE Trans. Cybern.* **2020**, *50*, 1383–1394. [\[CrossRef\]](#) [\[PubMed\]](#)
25. Yang, X.; Deng, W.; Yao, J. Neural Adaptive Dynamic Surface Asymptotic Tracking Control of Hydraulic Manipulators with Guaranteed Transient Performance. *IEEE Trans. Neural Netw. Learn. Syst.* **2023**, *34*, 7339–7449. [\[CrossRef\]](#) [\[PubMed\]](#)
26. Liang, X.; Yao, J. Adaptive Neural Network Force Tracking Control for Hydraulic Manipulator with Output Constraints. *Control Theory Appl.* **2025**, *42*, 138–148.
27. Wang, C.; Wang, J.; Ye, J.; Guo, Q.; Li, T. A Reinforcement Learning-Based Optimized Backstepping Control Approach for Uncertain Electro-Hydraulic Systems. *Mech. Syst. Signal Process.* **2025**, *237*, 112928. [\[CrossRef\]](#)
28. Yao, Z.; Xu, F.; Jiang, G.; Yao, J. Data-Driven Control of Hydraulic Manipulators by Reinforcement Learning. *IEEE/ASME Trans. Mechatron.* **2023**, *28*, 2673–2684. [\[CrossRef\]](#)
29. Yao, Z.; Liang, X.; Jiang, G.; Yao, J. Model-Based Reinforcement Learning Control of Electrohydraulic Position Servo Systems. *IEEE/ASME Trans. Mechatron.* **2023**, *28*, 1446–1455.
30. Bu, F.; Yao, B. Nonlinear Adaptive Robust Control of Hydraulic Actuators Regulated by Proportional Directional Control Valves with Deadband and Nonlinear Flow Gains. In Proceedings of the 2000 American Control Conference, Chicago, IL, USA, 28–30 June 2000; pp. 4129–4133.
31. Wang, X.; Shao, H. Theory of RBF Neural Network and Its Application in Control. *Inf. Control* **1997**, *4*, 32–44.
32. Xu, K.; Ai, C.; Chen, G.; Chen, J.; Kong, X. Modelling and Motion Control of Hydraulic Manipulator Based on Deep Learning and Reinforcement Learning. *Neurocomputing* **2026**, *669*, 132371. [\[CrossRef\]](#)
33. Gao, J.; Fu, Z.; Zhang, S. Adaptive Fixed-Time Attitude Tracking Control for Rigid Spacecraft with Actuator Faults. *IEEE Trans. Ind. Electron.* **2019**, *66*, 7141–7149. [\[CrossRef\]](#)
34. Chen, J.; Guo, Y.; Kong, X.; Xu, K.; Ai, C. Trajectory Planning and High-Precision Motion Control of Excavators Based on Independent Metering Hydraulic Configuration. *IEEE Trans. Syst. Man Cybern. Syst.* **2025**, *55*, 8689–8700. [\[CrossRef\]](#)

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.