Adaptive Impedance Control with Neural Networks and Variable Stiffness for Intelligent Robotic Arm Manipulation

Jingcai Liu
Artificial Intelligence Department, Jiangxi University of Technology, Nanchang 330098, China
E-mail: liujingcai093602@163.com

Keywords: adaptive, impedance control, robotic arm, intelligent control, application

Received: August 30, 2025

Affected by artificial intelligence and automation technology, robotic arms have become increasingly widespread. However, traditional impedance control algorithms have been unable to adapt robotic arms to increasingly complex environmental changes. Therefore, an adaptive impedance control algorithm based on variable stiffness, neural networks, and motion trajectories is developed. Then, it is used for intelligent control of robotic arms. This method employs an impedance relationship model and dynamically optimizes inertia, damping, and stiffness parameters through PD algorithm. The physical quantities such as joint angular acceleration and end displacement are input into the neural network, which outputs key stiffness and interference to estimate the unknown environment in real-time. An adaptive trajectory generation module is designed and combined with contact force feedback to achieve collaborative optimization. The experimental results showed that in terms of contact force with the environment, the error rate between the contact force controlled by this algorithm and the expected value was about 2%. In terms of joint position, the error between the selected joint position in the experiment and the expected value was kept within 0.5%. When running under the motion trajectory generated by the algorithm, the error of each parameter value was less than 1%. The adaptive impedance control algorithm can significantly improve the stability, accuracy, and safety of robotic arm control. The research results have potential applications in improving the environmental adaptability of robotic arms.

Povzetek: Prispevek predstavlja adaptivni algoritem impedance za robotske roke, ki združuje spremenljivo togost, nevronske mreže in prilagodljivo generiranje trajektorij. Metoda omogoča sprotno oceno okolja ter izboljša stabilnost, točnost in varnost interakcije v neznanih in dinamičnih okoljih.

1 Introduction

The continuous development of artificial intelligence and automation technology has driven the continuous growth of the robotic arm industry. Various robotic arms have been extensively applied in various fields. As the tasks that robotic arms need to perform become increasingly complex, their performance requirements are also becoming higher [1]. In the current tasks performed by robotic arms, most of them require interaction with the environment, which can generate contact forces with the environment for various reasons, thereby affecting the torque and robotic arm position. To ensure the smooth progress of the task, it is necessary to effectively control the position and force, making it adaptable to a certain extent. Impedance control algorithm is one of the main methods for implementing adaptive control. It can adjust the dynamic characteristics of the robotic arm, estimate environmental parameters, and achieve precise control of the robotic arm [2]. Therefore, scholars at home and abroad have conducted various studies on impedance control algorithms. To accurately control the foot position and legs of the quadruped robot, Pedro et al. used impedance control for each leg of the quadruped robot and generated reference trajectories using a sixth order Bezier curve. These trajectories were derived from the leg velocities in the planar kinematic model, which were used for body control. The test results demonstrated its effectiveness, enabling the robot to track the reference trajectory while exhibiting stable walking and running gaits [3]. Jing et al. designed an adaptive dual loop impedance control for controlling the interaction force between a dual arm manipulator and an object in an unknown environment. The proposed method satisfied three control objectives, including object trajectory tracking, internal force control between the object and the arm, and interaction force control between the object and the unknown environment. Finally, the stability verification was provided based on the Lyapunov method and the Routh criterion [4]. When a dual arm robot comes into direct contact with the external environment, it is crucial to control the interaction forces between the two. Therefore, Li et al. designed an impedance control based on particle swarm optimization to track the interaction forces and achieve compliant control of the robot end-effector. The system had a steady-state within 0.03 s, with overshoot convergence. Even under sudden force changes, the range of force fluctuations in steady-state reduced to around ±0.08 N [5]. Pengyu et al. proposed an equivalent stiffness impedance control to improve the accuracy of position tracking for hydraulic single leg robots. The

designed strategy achieved fast response speed and high tracking accuracy. In addition, the mechanical characteristics near the required location were approximate to traditional strategies [6]. Xuhui et al. proposed an adaptive impedance control method for

Table 1: Comparison of existing research work.

Author	Methods	Adoption of technology	Target systems	Performance indicators	Limitations
Pedro et al. [3]	Leg impedance control	Double loop impedance+Lyapunov stability	Quadruped robot	Trajectory Tracking Accuracy, Gait Stability	Depends on predefined trajectories, resulting in poor environmental adaptability
Jing et al. [4]	Adaptive dual-loop impedance control	Double loop impedance+Lyapunov stability	Dual-arm robot	Internal and External Force Tracking Accuracy	Complex controller structure, challenging real-time performance
Li et al. [5]	Particle swarm optimization impedance control	Impedance control+PSO optimization	Industrial robot	Overshoot, Convergence Time	Time-consuming optimization process, making online parameter adjustment difficult
Pengyu et al. [6]	Equivalent stiffness impedance control	Equivalent stiffness model	Hydraulic single-leg robot	Position Tracking Accuracy, Response Speed	Insufficient generalization for specific hydraulic systems
Xuhui et al. [7]	Kalman filter impedance control	Impedance control+Kalman filter	Blade polishing robot	Force Fluctuation, Surface Roughness	Filter performance relies on an accurate environmental model
Yanan et al. [8]	Variable impedance control	Variable impedance parameter	Legged robot	Foot Compliance	Limited to numerical simulation, lacking real-world validation

blade polishing based on Kalman filtering. This strategy analyzed the effects of stiffness and displacement changes on polishing force, and established a stiffness displacement coupling compensation model. The results showed that in the blade polishing experiment, compared with traditional methods, the roughness of this method was reduced to Ra0.32 μm or less, and the polishing force fluctuation range was within ± 1N, effectively improving the force control performance and blade surface quality [7]. The traditional impedance control parameters are fixed and subject to fixed-point constraints. To enhance the compliance of robot feet and achieve flexible interaction with the ground in complicated conditions, Yanan et al. built a variable impedance control. The effectiveness was verified through numerical simulation in MATLAB [8]. The existing research is shown in Table 1.

To sum up, although the existing impedance control has introduced modules such as online parameter adjustment, single neural network, or trajectory optimization, the existing fixed order impedance is difficult to give consideration to both rapid response and overshoot suppression, and the high-order or fractional order models lack real-time analytical means. In addition, the stiffness, damping, and trajectory of existing methods are adjusted through independent links connected in series, and the one-way transmission of information makes it difficult to obtain the overall optimal solution. In terms of collaboration, environment estimation, actuator saturation, and motion planning are separate. When stiffness jumps or force suddenly changes, the targets of each module conflict, and the system is prone to jitter and instability.

Therefore, this study aims to design a real-time adaptive impedance controller to break through the contradiction between real-time performance and stability of existing methods, build a collaborative closed-loop for environmental perception, dynamic adjustment, and trajectory optimization to solve the error accumulation, adapt to unstructured job requirements, and avoid system jitter. The controller adopts a 3-layer fully connected feedforward neural network, with the input layer containing physical quantities related to the motion and force of the robotic arm, the hidden layer using ReLU activation function, and the output layer using Tanh activation function to output environmental key parameters. The neural network training adopts Adam optimizer and mean square error loss function, and the variable stiffness parameters are fine tuned online through PD iterative algorithm. The simulation platform is built based on the dynamic parameters of the robotic arm and supports environment and disturbance simulation. The physical experiment platform includes the robotic arm, force sensors, and contact environment, and is equipped with data acquisition and control software. Thus, an adaptive impedance control algorithm is established, which integrates variable stiffness, neural networks, and motion trajectories.

The innovation of this study lies in proposing an integrated control architecture that integrates variable stiffness regulation, neural network online learning, and adaptive motion trajectory generation. Compared with traditional impedance control that relies solely on fixed parameters or a single compensation method, this architecture enhances its responsiveness to dynamic environments through online stiffness adjustment. By

combining neural networks for real-time estimation and compensation of unknown environmental parameters and system nonlinearity, the trajectory generation module further coordinates and optimizes position and contact force, thereby significantly improving the control accuracy and adaptability of the robotic arm in

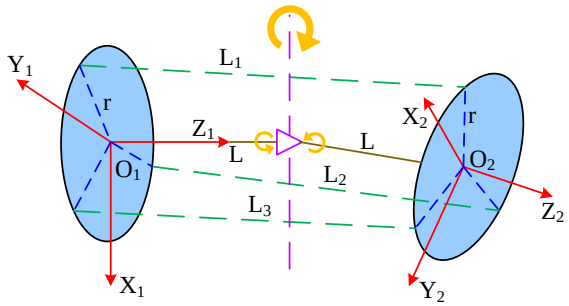


Figure 1: Kinematic model of a single joint of a robotic arm.

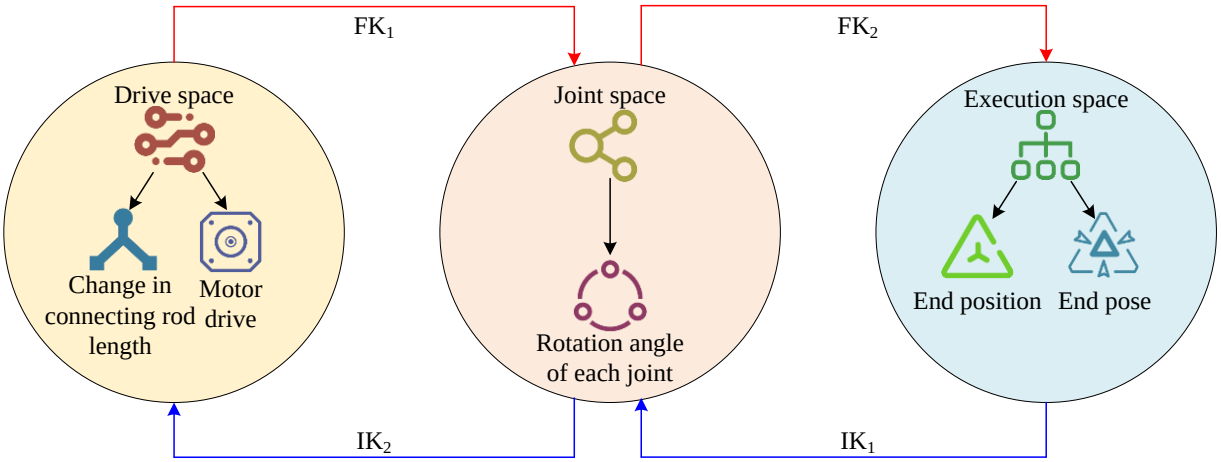


Figure 2: Overall structure and kinematic analysis of the robotic arm.

unstructured environments while maintaining system stability. The contribution of this research is to estimate the unknown environmental parameters and system nonlinearity in real-time according to the neural network, use the variable stiffness impedance controller to dynamically adjust the interactive dynamic characteristics, and take the trajectory generator to output the real-time motion path. The collaboration of the three provides an effective solution for achieving high-precision, high flexibility, and high intelligence interactive control of robotic arms in unknown and uncertain environments.

2 Intelligent control of robotic arm based on adaptive impedance control algorithm

2.1 Motion analysis of robotic arm

The analysis of the motion process of a robotic arm is the basis for implementing impedance control, which reveals the motion relationship between the coordinate systems of each connecting rod of the robotic arm. It is essential for intelligent control of the robotic arm [9]. A kinematic model of a single joint of the robotic arm is firstly established to analyze, as shown in Figure 1.

From Figure 1, the adjacent sleeves of the robotic arm are connected by universal joints. The platform where the coordinate system O_1 is located is a fixed platform, which is connected to the root rope. The platform where the coordinate system O_2 is located is a moving platform, close to the end-effector. The two are connected by three extendable links, and the three connection points are evenly distributed around the circumference of the platform. Based on the coordinates of the two ends of the connecting rod on the platform, the length of the connecting rod can be calculated, as shown in equation (1).

$$L_i = \sqrt{\left\| Q_{Ai} - Q_{Bi} \right\|} = f(\theta, r, L), \quad i = 1,2,3 \tag{1}$$

In equation (1), L_i is the length of each connecting rod. Q_{Ai} is the homogeneous coordinate of the connecting rod on the moving platform. Q_{Bi} is the homogeneous coordinate of the connecting rod on the fixed platform. θ is the rotation angle. r is the platform radius. L is half the platform distance. This indicates that as long as a certain joint rotation angle variable is given, the corresponding connecting rod length variable can be obtained [10]. A robotic arm generally has multiple joints, and the calculation method for each joint is the same as that for a single joint. Therefore, the changes of the connecting rod length at each joint can be obtained. The kinematic analysis of the entire robotic arm can be achieved, as shown in Figure 2.

The robotic arm controls the rotation angle of each joint universal joint by varying the connecting rod length, and the rotation angle ultimately determines the spatial position of each joint. The kinematic analysis of a robotic

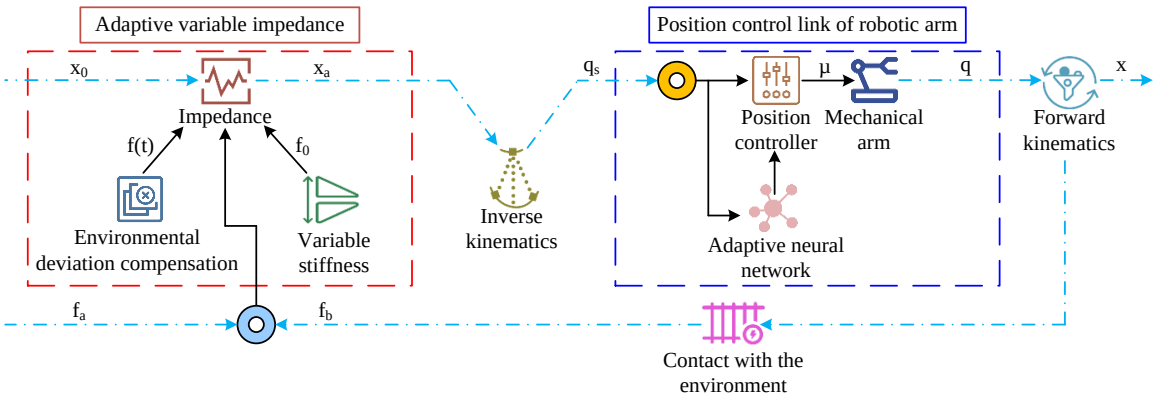


Figure 3: Adaptive impedance control of robotic arm.

arm actually analyzes the relationship between changes in connecting rod length, joint angles, and the spatial position of the end-effector [11]. From Figure 2, the length change of the connecting rod is a variable of the robotic arm drive part. The rotation angle is a variable of the joint part, and the end spatial position is a variable of the actuator part. There are two kinematic modes between these three parts. FK1 and FK2 represent forward kinematics, where the robotic arm calculates the specific spatial position of the end-effector using preset connecting rod lengths and joint angles. IK1 and IK2 represent inverse kinematics. With the known spatial position of the end-effector, the rotation angles of each joint are calculated through inverse kinematics, and the corresponding connecting rod lengths of each joint are obtained. Under the combined action of two kinematic modes, the robotic arm achieves motion and operation in space through motor drive [12].

2.2 Adaptive impedance control of robotic arm

The robotic arm often comes into contact with the environment during actual operation, which inevitably affects the rotation angles of each joint, leading to changes in the overall motion of the robotic arm. To achieve stability in the motion process, impedance control technology is adopted. However, the existing impedance control is difficult to meet the operational requirements of the robotic arm and needs to be improved. A adaptive impedance control method with varying stiffness is designed to address the transient performance of robotic arms that cannot be guaranteed due to parameter changes in impedance control, as shown in Figure 3.

From Figure 3, this method first calculates the impedance relationship between the robotic arm and the environment in a specified direction in space, as shown in equation (2).

$$m\left(v_a^{'}-v_b^{'}\right)+b\left(v_a-v_b\right)+k\left(x-x_0\right)=f_b-f_a=e_f \quad (2)$$

In equation (2), m, b, and k signify the inertia, damping, and stiffness coefficients of the robotic arm. x represents the position of the end-effector. v_a represents the speed of the end-effector. $v_a^{'}$ represents the acceleration at the end-effector. x_0 represents the trajectory of mechanical motion. v_b represents the robotic arm's speed, and $v_b^{'}$ represents the acceleration. f_b represents the contact force between the robotic arm and the environment. f_a represents the contact force threshold. e_f represents the contact force error value. To reduce environmental errors and improve control robustness, the PD algorithm is utilized for iteration, as shown in equation (3).

$$f(t)=f(t-\lambda)+\eta\left[f_a(t)-f_b(t)\right]+\frac{\varsigma}{\lambda}\left\{\left[f_a(t)-f_b(t)\right]-\left[f_a(t-\lambda)-f_b(t-\lambda)\right]\right\} \quad (3)$$

In equation (3), $f(t)$ represents the compensation amount. λ represents the sampling period. η and ς represent learning rates. After reducing environmental errors, adaptive impedance control is applied to optimize the contact force error between the environment and the robotic arm, reducing the impact, as shown in equation (4) [13].

$$w(t)=\frac{1}{2}\left(\mu[\frac{1}{b}(e_f-f(t)-f_0]^2+ce_f^{\ 2}\right) \quad (4)$$

In equation (4), $w(t)$ represents the optimized objective equation. μ represents the weight coefficient of the compensation bias term. c represents the weight coefficient of the initial error term. Among them, the first item $\mu[\frac{1}{b}(e_f-f(t)-f_0]^2$ can reflect the deviation between the compensated contact force $f(t)$ and the reference force f_0, while the second item $ce_f^{\ 2}$ represents the initial contact force error e_f. By optimizing the objective equation, it is possible to balance compensation accuracy and initial error suppression. To ensure the stability of the end-effector transient performance, it is also necessary to reduce the impact of end-effector position errors, as shown in equation (5).

$$u(t)=K_p\left[e(t)+\frac{1}{T}\int_0^t e(t)dt+T\frac{de(t)}{dt}\right] \quad (5)$$

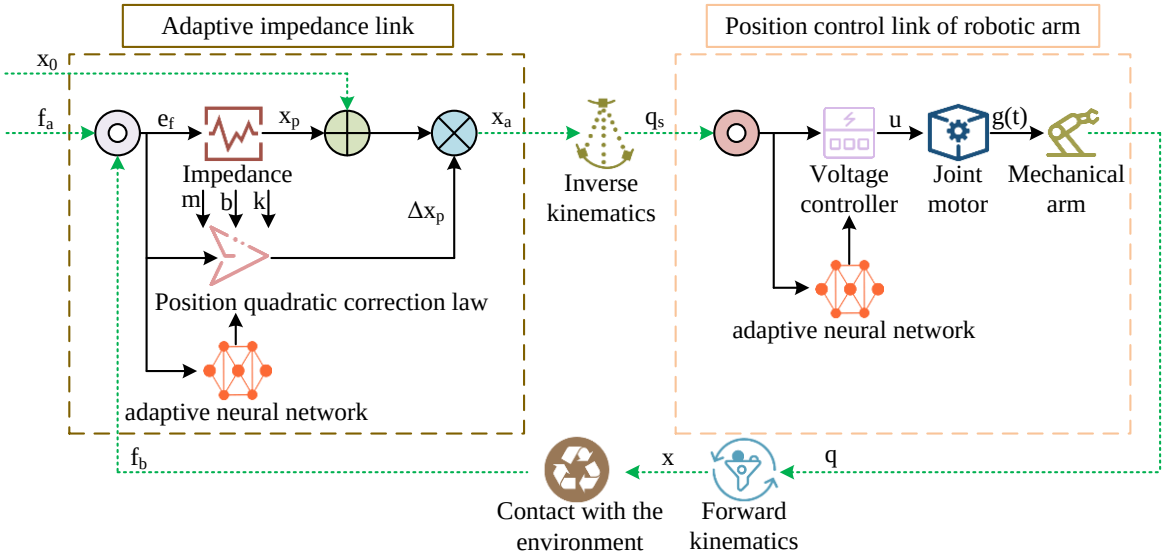


Figure 4: Impedance control of robotic arm based on adaptive neural network.

In equation (5), $u(t)$ represents the optimized value. K_p represents the scaling factor. $e(t)$ signifies the error. T signifies the motion time. To solve the poor adaptability and insufficient accuracy to the environment when using impedance control to adjust the position, the impedance control is combined with adaptive neural network to improve its performance in unknown environments, as shown in Figure 4.

From Figure 4, this method has made a quadratic correction to the impedance relationship between the end-effector and the environment represented in equation (2), as shown in equation (6).

$$e_f = f_b - f_a = k_e x_e - k_e\left(x_0+\Delta x_p+h(s)e_f\right)-f_a \quad (6)$$

In equation (6), k_e represents the environmental stiffness. x_e represents the environmental location. Δx_p represents the correction amount. $h(s)$ represents the impedance transfer function. By introducing environmental stiffness and correction variables, the initial impedance relationship is optimized twice to enable the dynamic response characteristics of the robotic arm to adapt to the physical characteristics of the environment in real-time, avoiding overshoot or response lag problems of traditional fixed impedance in software and hardware environments. Then, an adaptive neural network is used to estimate the unknown parts in the environment, as shown in equation (7) [14].

$$W_1^{*T}h(\theta_1)+\varepsilon_1=-\frac{f_d}{k_e}-\frac{b}{kk_e}x_{a2}-\frac{k+k_e}{kk_e}x_{a1}-\frac{m}{kk_e}\dot{\alpha}_{a1} \quad (7)$$

In equation (7), W_1^{*T} represents the weight vector of the unknown part in the environment. W_1^* represents the optimal weight value. $h(\theta_1)$ represents the activation function of the neural network. θ_1 represents the characteristic parameter. ε_1 represents the error value. x_{a1} and x_{a2} represent displacement variables. $\dot{\alpha}_{a1}$ represents joint angular acceleration. The estimated value obtained can be used to achieve the expected position in an unknown environment. To improve the impedance control effect, it is necessary to analyze the output characteristics of the joint motor that provides the robotic arm torque, as displayed in equation (8).

$$M(q)q_2+C(q,q_1)q_1+G(q)+f_n(q_1)+\tau_d=S(\tau) \quad (8)$$

In equation (8), M signifies the inertia matrix. q signifies the joint position. q_2 signifies the joint acceleration. C represents the centrifugal force. q_1 represents the joint velocity. G represents the gravity. f_n represents the frictional force. τ_d represents the external interference. τ represents the torque. S represents the saturation function. After obtaining the output power model of the robotic arm joint motor, an adaptive neural network is used to optimize it and reduce the instability caused by its saturated output, as shown in equation (9).

$$W_2^{*T}h(\theta_2)+\varepsilon_2=C\alpha_{b1}+G+M\dot{\alpha}_{b1}+f_n+\tau_d \quad (9)$$

In equation (9), W_2^* represents the optimal weight value. C represents a constant matrix. α_{b1} represents the position vector. M represents the quality matrix. $\dot{\alpha}_{b1}$ represents the velocity vector. The optimized joint motor output power model is shown in equation (10) [15].

$$\begin{cases} L\dot{i}+R\dot{i}+K_b\dot{q}=u \\ K_T i=\tau \end{cases} \quad (10)$$

In equation (10), L, R, K_b, and K_T represent diagonal matrices. $\dot{i}$ represents the current vector of the joint motor. u represents the input voltage of the joint motor. By clarifying the quantitative relationship between

electrical parameters and mechanical parameters, it ensures that the dynamic characteristics required for impedance control can be achieved through the actual driving of the motor, avoiding unstable control caused by the disconnection between algorithm instructions and hardware capabilities. Finally, as impedance control

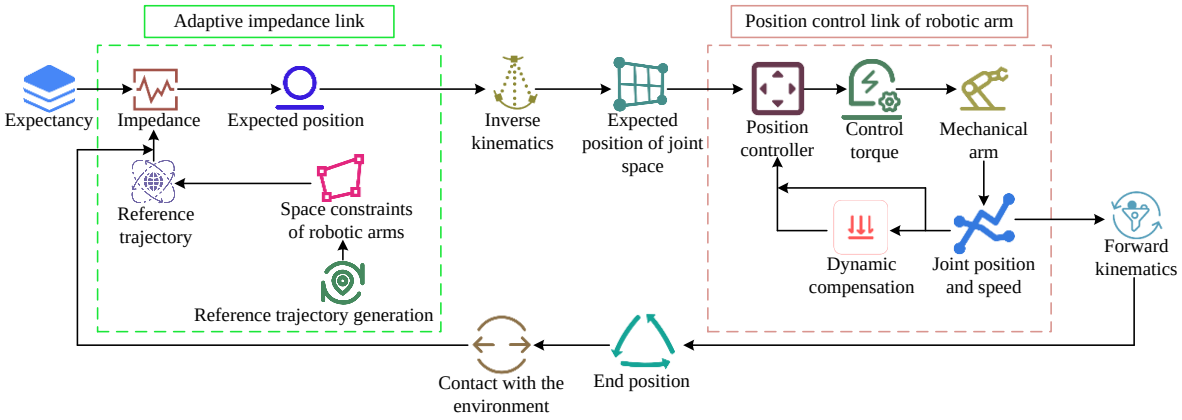


Figure 5: Impedance control for adaptive generation of motion trajectories.

requires the the motion trajectory, and the methods for the motion trajectory of the robotic arm vary in different environments, a impedance control method that can adaptively generate the motion trajectory is proposed, as displayed in Figure 5.

From Figure 5, this method first estimates the surrounding environmental information based on the end contact force, thereby determining a preliminary motion trajectory, as displayed in equation (11).

$$\begin{cases} \hat{f}_e = \hat{k}_e\left(\hat{x}_e - x_d\right) \\ x_r = \hat{x}_e - \dfrac{f_d}{\hat{k}_e} \end{cases} \tag{11}$$

In equation (11), f_d represents the estimated contact force. $\hat{k}_e$ represents the estimated contact stiffness. $\hat{x}_e$ represents the estimated contact location. x_d signifies the expected position. x_r signifies the motion trajectory. f_d signifies the expected contact force. The impedance control is used to perform secondary adjustments on the obtained motion trajectory to obtain the corrected end-effector of the robotic arm, which is then input into the position controller to improve impedance control of the contact force at the robotic arm end [16], as shown in equation (12).

$$\tau = C_0 x_2 + G_0 + M_0(\Phi + u) + M_0\dot{\alpha} - M_0 z_1 \tag{12}$$

In equation (12), τ signifies the robotic arm torque. C_0 signifies the power coefficient. x_2 represents the joint angle variable. Φ represents the position variable. u represents the input item. z_1 represents the error variable. To further reduce the error of the motion trajectory, except for improving the impedance control of its end contact force, it is also necessary to improve the impedance control of its position, as shown in equation (13).

$$\tau_\Delta = \Delta M(q)q_2 + \Delta C(q, q_1)q_1 + \Delta G(q) \tag{13}$$

In equation (13), ΔM, ΔC, and ΔG represent the position error coefficients. By compensating for the error value of the position, its impact on impedance control can be reduced [17]. This can enable the algorithm to generate adaptive motion trajectories that meet the accuracy requirements of contact force and position as much as possible.

2.3 Intelligent control application of robotic arm

In the designed adaptive impedance control method, variable stiffness impedance adjustment is the basic control layer, which compensates for initial environmental errors through PD algorithm, dynamically adjusts inertia, damping, and stiffness coefficients, and provides flexible dynamic characteristics for the robotic arm to adapt to environmental contact, solving the poor transient performance in traditional fixed parameter impedance control. As the core perception and optimization layer, neural networks can provide real-time environmental input for variable stiffness estimation through unknown environmental parameters, and provide stable actuator output for variable stiffness adjustment through motor parameter tuning, avoiding control deviation caused by external interference. Adaptive motion trajectory is the final execution layer, which generates preliminary trajectories based on environmental estimation information output by neural networks, and then uses variable stiffness impedance control to make secondary adjustments to the trajectories, ensuring that the robotic arm meets the control objectives of stable contact force and precise position in the position environment. The intelligent control of the robotic arm is shown in Figure 6.

From Figure 6, after receiving the target command, the robotic arm first needs to analyze its kinematic characteristics, then plan the motion trajectory based on the surrounding environmental information, and finally optimize the obtained motion trajectory through adaptive impedance control algorithm [18]. The adaptive impedance algorithm takes a PID controller to calculate the error, as shown in equation (14).

$$u(t) = Kp \cdot e(t) + Ki\int_0^t e(t)dt + Kd \cdot \dot{e}(t) \tag{14}$$

In equation (14), $u(t)$ represents the output value. Kp, Ki, and Kd signify proportional, integral, and differential coefficients. $\dot{e}(t)$ signifies the error change

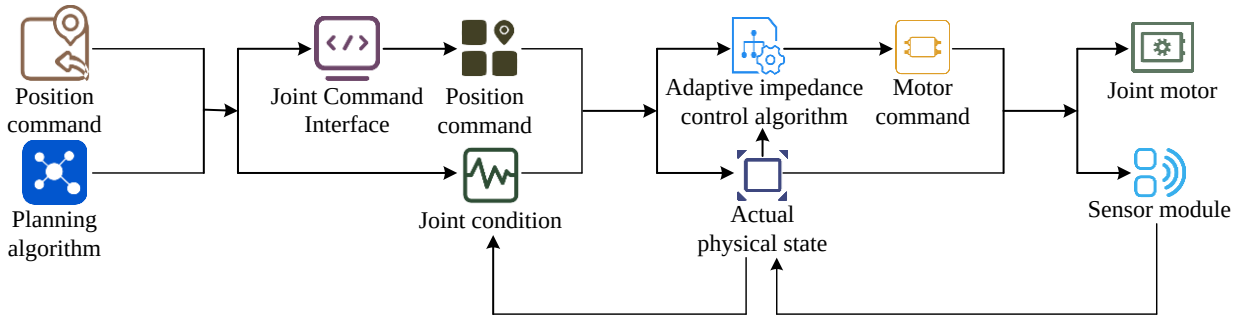


Figure 6: Intelligent control process of robotic arm.

Table 2: Pseudo code for adaptive impedance control.

```
Algorithm: Adaptive Impedance Control
# Init: W(NN weights), K0=500N/m, F_max=100N, e_F_prev=0
while Task running:
  F_act,q_act=Read_Sensors(); e_F=F_des-F_act; e_q=q_des-q_act  # Data
  if F_act>F_max: D=500Ns/m; Limit_Current(); continue  # Safety
  K_corr=NN_Forward([e_F,(e_F-e_F_prev)/T],W); W=Update_NN(W,K_corr)  # NN
  K=K0+K_corr; F_final=K*e_q+D*(q_dot_des-(q_act-q_act_prev)/T)  # Impedance
  τ=M*q_ddot_traj+F_final; Output_Voltage(τ/K_T)  # Motor Ctrl
  e_F_prev=e_F; q_act_prev=q_act  # Update States
```

rate. After obtaining the error value, it is optimized, as shown in equation (15).

$$e(t) = l_d(t) - l(t) \tag{15}$$

In equation (15), $l_d(t)$ signifies the expected position. $l(t)$ signifies the actual position [19]. After optimizing the motion path, it is input into the execution module. After the robotic arm starts running, real-time contact force and position signals of the robotic arm are collected through sensors and input back into the adaptive impedance controller. If the signal is abnormal, it will be adjusted again to achieve intelligent control [20]. The pseudo code for adaptive impedance control is shown in Table 2.

2.4 Neural network online learning mechanism

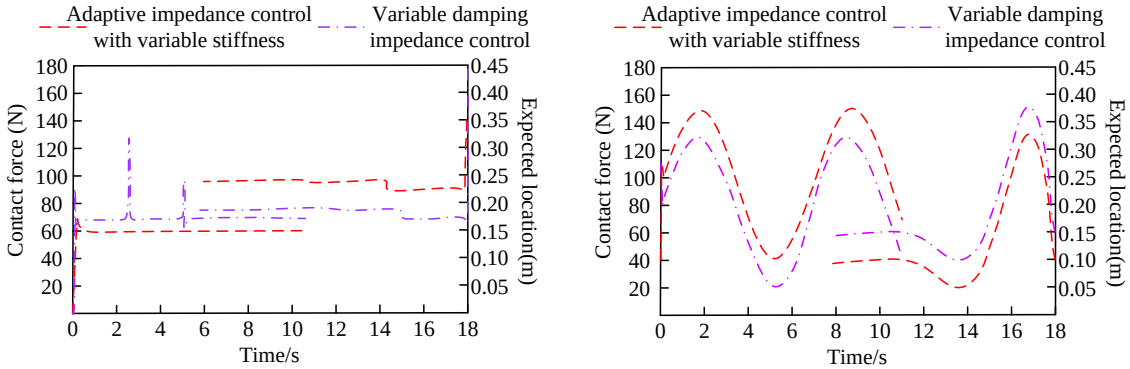
The online learning of neural networks takes the difference between the expected contact force and the actual contact force, as well as its first-order rate of change, as the two-dimensional input signal, with the target output being stiffness correction. The network uses stochastic gradient descent and completes forward and backward propagation once every 1 ms control cycle, with a fixed learning rate of 0.01. When the absolute value of the force error exceeds 2 N or the rate of change is greater than 5 N·s-1, the learning rate instantly increases to 0.05 and remains at 50 ms, while the gradient norm is clipped to within 2.0 to prevent weight divergence. The sliding window retains the last 500 samples, and the weight of historical samples decays exponentially by 0.995, allowing the network to continuously forget outdated information and quickly adapt to changing environments. A single iteration takes less than 0.2 ms and can run in real-time at a frequency of 1 kHz without the need for offline retraining to complete environment switching.

3 Control effects of adaptive impedance algorithm

3.1 Effects of adaptive impedance control with variable stiffness

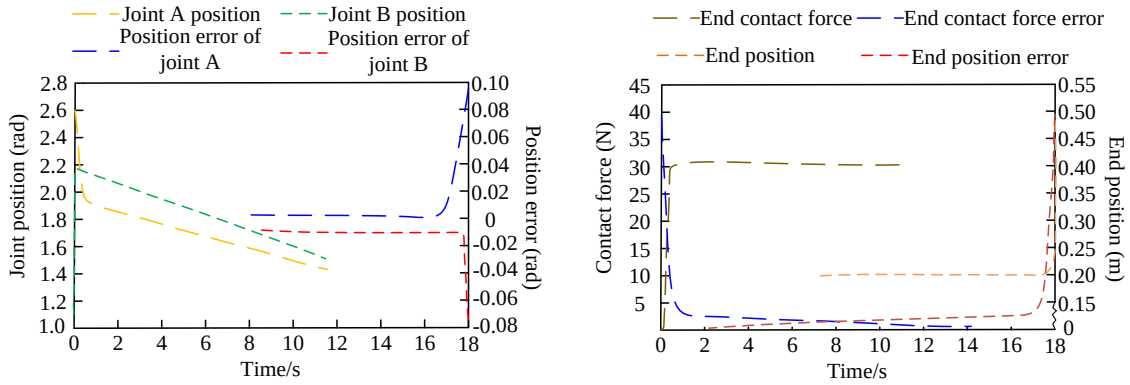
In this study, the UR5 CB3 six-degree-of-freedom manipulator is used, the base is fixed horizontally, and the ATI mini45 six-dimensional force sensor is installed on the end flange. The sampling frequency is 1 kHz, and the resolution is 0.06 n. The plane test is carried out on a 20 mm thick acrylic plate, and the surface test is successively covered with a 5 mm sponge and a 2 mm aluminum plate to form a stiffness jump environment. The room temperature is maintained at 23 ℃ and the humidity is 45% RH. The variable damping impact control is taken as the comparison algorithm, so that the damping adjustment range is 20-200ns/m, the inertia matrix is 2.5 times the identity matrix, and the stiffness matrix is 800 times the identity matrix. For the proposed variable stiffness adaptive impedance control method, the variable stiffness range is 200-1,200 NGM, the damping is adjusted synchronously according to the damping ratio of 0.707, and the inertia matrix is 2.5 times the identity matrix. The dimension of stiffness decision network is 2, the hidden layer is 1, containing 64 neurons, and the learning rate is 0.01. Online random gradient descent is used for training.

The dimension of trajectory prediction network is 3, the hidden layer is 1, including 32 neurons, the activation function is Tanh function, the maximum number of iterations is 200, and the learning rate is 0.001, which is optimized offline by Adam optimizer. All experiments are independently repeated 5 times.



（a）Comparison chart on the plane （b）Comparison chart on curved surface

Figure 7: Comparison results of contact force and position of robotic arm.



（a）Joint position analysis result chart （b）End contact force and position result diagram

Figure 8: Analysis results of contact force and position between joints and ends.

To demonstrate the effectiveness of variable stiffness adaptive control, a comparative analysis is conducted under the same environment and given parameters, as shown in Figure 7.

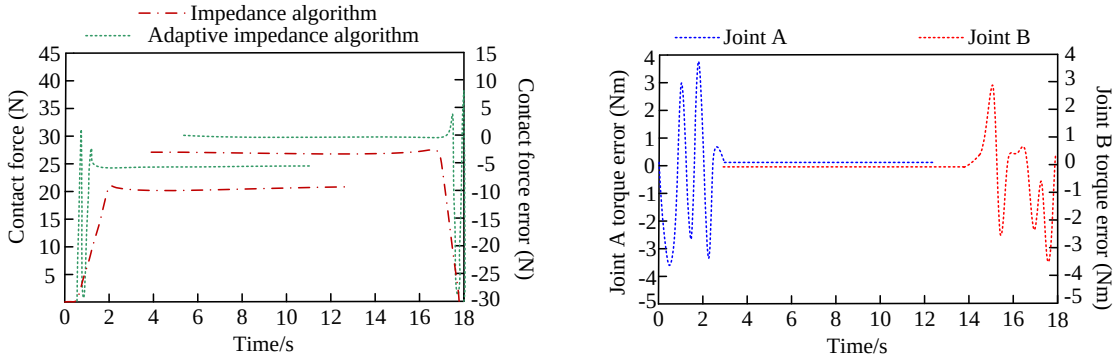
When conducting experiments on a plane, the expected contact force was set to 60 N and the motion trajectory was set to 0.23 m. From Figure 7 (a), when using the comparative algorithm for control, the contact force between the robotic arm and the environment after stabilization was around 70.3±2.5 N, which was about 10.3±1.3 N higher than that of the preset contact force value. The algorithm is prone to sudden force changes due to environmental factors at the moment of contact, resulting in poor stability. In contrast, the algorithm proposed in this article controls the steady-state contact force within ±2% of the set value, significantly reducing fluctuations with a 95% confidence interval of [59.1, 61.3] N, demonstrating better tracking performance and stability. The robotic arm operates according to predetermined instructions in the plane, thereby significantly improving operational accuracy and safety. From the position of the robotic arm during operation, the proposed algorithm also made the robotic arm more accurately preset the reference trajectory compared with the comparative algorithm, while the comparative algorithm had certain errors. When conducting experiments on curved surfaces, the study changed the parameter values of contact force and motion trajectory, as shown in Figure 7 (b). When using the designed method to control the contact force and operating environment position, the contact force and motion trajectory of the robotic arm were closer to the set parameter values, which was the same as in the planar experiment, improving the control accuracy and operational stability. To further test the control effect, two joints of the robotic arm are selected for analysis, as shown in Figure 8.

The standard value of the contact force in the experimental setup is 30N, and the motion trajectory of the end of the robotic arm is 0.21m. The robotic arm maintains a constant speed during operation. From Figure 8 (a), the positions of joint A and joint B differed significantly from the command value at the beginning of the robotic arm startup. Under the control of the proposed method, their positions were quickly adjusted to reach the command value and remained stable during operation. The joint position error after stabilization was small, only about 0.9%±0.1%, The 95% confidence intervals were [0.007, 0.009] rad and [0.008, 0.010] rad, respectively. This indicates that the algorithm has a fast response speed, which can quickly control and adjust joints with errors

based on input values to meet the needs of robotic arm operation. From Figure 8 (b), after the joint position error was eliminated, the contact force and position at the end of the robotic arm were correspondingly corrected to near the preset standard

Table 3: Performance comparison with frontier control methods.

Indicator		Steady-state contact force error (N)	Maximum trajectory deviation (mm)	Recovery time (s)	Single cycle CPU time (ms)	Controller tuning frequency
Method	Fractional order practical fixed time synchronous adaptive fuzzy control	2.9±0.3	5.1±0.4	0.85±0.05	3.1±0.2	5
	Output feedback projection lag synchronous control	2.4±0.2	4.3±0.3	0.72±0.04	2.8±0.1	4
	Robust Neural Adaptive Control	1.8±0.2	3.2±0.2	0.55±0.03	4.5±0.3	3
	Adaptive backstepping control	3.9±0.4	3.9±0.3	0.68±0.04	2.2±0.1	4
	Nonlinear Optimal Control of Induction Motor Driven Gas Compressor	1.9±0.2	3.5±0.3	0.59±0.03	2.6±0.1	3
	Single joint flexible arm DC motor backstepping control	2.1±0.2	3.7±0.3	0.65±0.04	2.4±0.1	3
	The proposed method	0.6±0.1	1.5±0.1	0.32±0.02	1.9±0.1	1



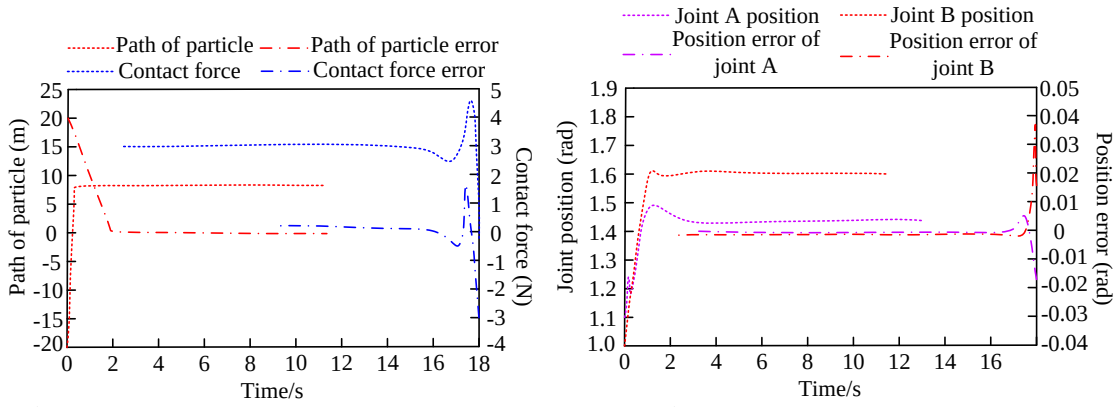
（a）Analysis of Contact Force and Its Error （b）Analysis of Joint Torque Range and Error

Figure 9: Analysis results of contact force and joint torque of robotic arm.

value in about 0.6s and remained stable overall during operation. After the contact force stabilized, there was an error of about 0.6±0.2N, with an error rate of about 2%±0.3%, the 95% confidence interval was [29.4,30.4] N. After the end-effector stabilized, there was an error of about 3±1mm, with an error rate of about 1.5%±0.1%, 95% confidence interval [0.2094, 0.2100] m. This proves the effectiveness of the control effect. To further validate the performance of the proposed adaptive impedance control algorithm based on variable stiffness, neural network, and motion trajectory, fractional order practical fixed time synchronous adaptive fuzzy control [11], output feedback projection lag synchronous control [12], robust neural adaptive control [13], adaptive backstepping control [14], nonlinear optimal control of induction motor-driven gas compressor [16], and single joint flexible arm DC motor backstepping control [17] are selected as comparative methods. The fractional order practical fixed time synchronous adaptive fuzzy control has an order of 0.95, 49 fuzzy rules, and a fixed time convergence parameter of 2 s. The high gain observer poles of the output feedback projection lag synchronous control are uniformly set at negative 100 rad·s-1, with a lag window of 0.1 seconds. The robust neural adaptive control RBF network adopts a width of 1.0, 81 nodes, and a robust gain of 10. The virtual control law gains of adaptive backstepping control are 5 and 20, respectively. The cost function weight matrix Q diag for nonlinear optimal control is 100, the R diag is 0.1, and the iterative solution step size is 1 ms. The virtual angular velocity gain for flexible arm DC motor backstepping control is set to 15, and the motor current gain is set to 50. All methods are run on the same UR5 platform. The steady-state contact force error, maximum trajectory deviation, recovery time, single cycle CPU time, and controller tuning frequency of different algorithms are compared, and the results are shown in Table 3.

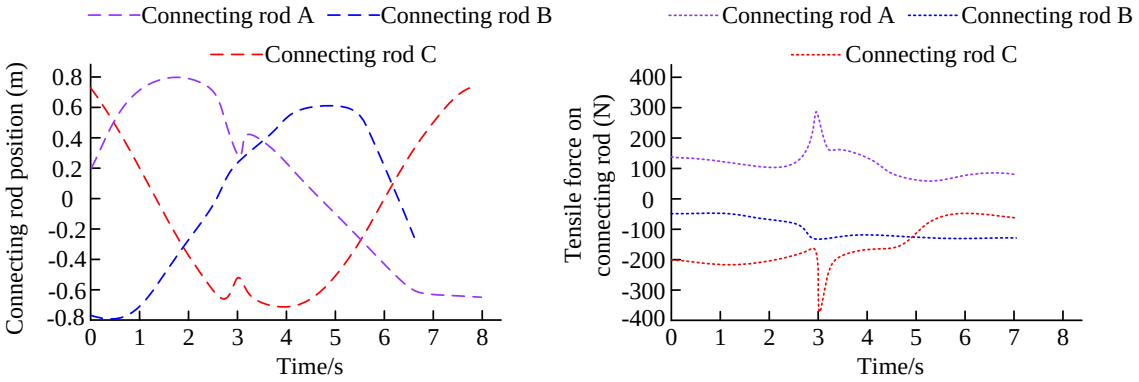
From Table 3, the steady-state contact force error of the proposed method was 0.6±0.1N, with a decrease of 66.7% compared to the sub-optimal robust neural adaptive control method. The maximum trajectory deviation was 1.5±0.1mm, which was 53.1% shorter than that of the sub-optimal robust neural adaptive control method. The recovery time was 0.32±0.02 s, which was 41.8% faster than that of the sub-optimal robust neural adaptive control method. The single cycle CPU time was 1.9±0.1ms, which was 20.8% less than that of the sub-optimal single joint flexible arm DC motor backstepping control. Only one time was needed to complete parameter tuning, which was significantly better than that of other schemes. The above results indicate that the proposed method can achieve higher accuracy, faster response, and lower computational

power requirements simultaneously in complex perturbation environments.



（a）Reference trajectory and contact force analysis diagram （a）Joint position analysis result chart

Figure 10: Analysis of the motion trajectory, contact force, and joint position of the robotic arm.



（a）Curve diagram of changes in connecting rod position （b）Curve of tensile force variation on connecting rod

Figure 11: Schematic diagram of connecting rod position and tension change analysis.

3.2 Effects of adaptive impedance control algorithm combined with neural networks

To verify the effectiveness of the designed control strategy, firstly, the control effects of contact force and torque on the robotic arm are analyzed, as displayed in Figure 9.

The expected contact force of the robotic arm in the experimental setup is 25 N, and the torque range of the selected joint is between -4Nm and 4Nm. From Figure 9 (a), the proposed adaptive impedance algorithm could bring the contact force into a stable state within about 1.0±0.1 s, and the convergence speed was better than that of traditional impedance algorithms. After the robotic arm stabilized, the contact force controlled by the adaptive impedance algorithm remained at around 25.1±0.3 N, the 95% confidence interval was [24.8, 25.4] N, which was the same as the set value, indicating that the neural network compensator effectively improved the accuracy of force control in step response tests. However, traditional control algorithms brought about a contact force error of about 5.0±0.5 N to the robotic arm, which cannot meet the operational accuracy requirements of the robotic arm. From Figure 9 (b), during the operation of the robotic arm, the output torque error of the selected joints was within the set torque range, without exceeded values. After the robotic arm stabilized, the output torque error of the joint was extremely small, about 2±0.2%, the 95% confidence interval was [-0.3, -0.1] N. This indicates that the adaptive impedance algorithm can achieve high-precision control, ensuring that it meets task parameters during operation and guaranteeing the safe and efficient execution of instructions issued by the robotic arm.

3.3 Effects of impedance control algorithm for adaptive generation of motion trajectory

To present the control effect of the impedance algorithm for adaptive motion trajectory generation, the motion trajectory, contact force with the environment, and the joint position are analyzed, as shown in Figure 10.

The expected contact force is set to 2.7 N, the expected position of joint A is 1.45 rad, and the expected position of joint B is 1.6 rad. From Figure 10 (a), the adaptive impedance algorithm initially generated a large error in the motion trajectory, but quickly adjusted it after about 0.5 s to ensure stable operation of the robotic arm. From the contact force between the robotic arm and the environment, after the system stabilized, the contact force was basically consistent with the expected value, and the

error was controlled within 1%±0.3%, with a 95% confidence interval of [2.67, 2.71] N, which enabled the

Table 4: Performance comparison of different control methods in different scenarios.

Scenario	Method	Contact force error rate (%)	Joint position error rate (%)	Stabilization time (s)
External force disturbance	Conventional impedance controller	11.8±0.5	3.2±0.2	1.8±0.1
	Adaptive controller based on particle swarm optimization	6.5±0.3	1.5±0.1	1.2±0.1
	Proposed algorithm	2.1±0.1	0.4±0.05	0.8±0.06
Unknown environmental stiffness	Conventional impedance controller	15.2±0.6	4.5±0.3	2.5±0.2
	Adaptive controller based on particle swarm optimization	7.3±0.4	2.1±0.2	1.6±0.1
	Proposed algorithm	2.3±0.2	0.5±0.03	0.9±0.07
Dynamic trajectory	Conventional impedance controller	9.6±0.4	2.8±0.2	1.5±0.1
	Adaptive controller based on particle swarm optimization	5.8±0.3	1.3±0.1	1.1±0.05
	Proposed algorithm	1.9±0.1	0.3±0.05	0.7±0.03

robotic arm to safely perform tasks. From Figure 10 (b), after the system stabilized, the positions of the selected joints A and B were extremely close to the expected values, with an error of only about 0.1%±0.02%, with 95% confidence intervals of [1.449, 1.451] rad and [1.598, 1.602] rad, respectively. The adaptive impedance control algorithm can adaptively generate safe, feasible, and high-precision motion trajectories based on the set parameters of the robotic arm and environmental information. This greatly optimizes the robotic arm operation in unknown environments and improves its ability to adapt to the environment.

3.4 Application analysis of intelligent control for robotic arms

To test the application effect of adaptive impedance algorithm in intelligent control of robotic arms, several connecting rods of the robotic arm are selected and their positions and tensile forces during operation are analyzed, as shown in Figure 11.

From Figure 11, the selected three connecting rods experienced a brief sudden change in tension at approximately 3 s, and the corresponding connecting rod positions also underwent significant changes to adapt to the changing force conditions. The 95% confidence intervals were [9.9, 10.3] N/[0.29, 0.31] m, [9.7, 10.1] N/[0.25, 0.27] m, and [9.8, 10.2] N/[0.27, 0.29] m. This indicates that the designed algorithm can enable the connecting rod of the robotic arm to have the ability to adapt to sudden external forces, ensuring that the robotic arm will not be disturbed by external forces. After the tension on the connecting rod returns to normal, the position of the connecting rod will quickly adjust back to the desired position controlled by the designed algorithm, and continue to move along the set reference trajectory, so that the robotic arm can operate normally. The designed algorithm can control and adjust the motion trajectory and contact force of the robotic arm environmental changes, enabling it to resist the influence of uncertain interference. This undoubtedly improves the stability, safety, and accuracy of the intelligent control system for robotic arms. To further verify the practical application effect of the proposed adaptive impedance control algorithm based on variable stiffness, neural network, and motion trajectory, the research tests it in external force disturbance scenarios, unknown environment stiffness scenarios, and dynamic trajectory scenarios. Among them, external disturbance refers to the random impact force of ±5N applied during the operation of the robotic arm. Unknown environmental stiffness refers to the stiffness of the environmental material switching from 2,000N/m to 500N/m. Dynamic trajectory scenario refers to the expected trajectory at the end switching from a straight line to a sine curve with a frequency of 0.5Hz. The contact force error rate, joint position error rate, and stabilization time of the proposed algorithm are calculated, respectively, and compared with the traditional impedance controller and the adaptive controller based on particle swarm optimization [5]. The results are shown in Table 4.

From Table 4, under external disturbance scenarios, the proposed algorithm had a contact force error rate of 2.1±0.1%, joint position error of 0.4±0.05%, and stabilization time of 0.8±0.06 s, all lower than those of the other two methods. This indicates that the proposed algorithm achieves fast disturbance suppression and maintains accuracy by estimating the disturbance amplitude in real-time through neural networks. In the unknown environmental stiffness, the contact force error rate, joint position error rate, and stabilization time of the proposed algorithm were 2.3±0.2%, 0.5±0.03%, and ±0.07 s, respectively. Compared with traditional impedance controllers, the contact force error rate and joint position error rate of the proposed method were reduced by 84.9% and 88.9%, respectively, and the stabilization time was shortened by 64.0%. Compared with the adaptive controller based on particle swarm optimization, the proposed method reduced the contact

force error rate and joint position error rate by 68.5% and 76.2%, respectively, and shortened the stabilization time by 43.8%. This indicates that the proposed algorithm can estimate the stiffness of the environment in real-time through neural networks, achieving rapid adaptation to unknown environments. In the dynamic trajectory scenario, the three indicators of the proposed algorithm were 1.9±0.1%, 0.3±0.05%, and 0.7±0.03s, respectively, all lower than those of comparison methods, indicating that it can adjust control parameters in real-time according to trajectory curvature and achieve high-precision tracking of dynamic trajectories. The above results demonstrate the adaptability and security of the proposed algorithm in complex work environments.

4 Conclusion

This study proposes a collaborative control method that integrates variable stiffness adjustment, neural network online compensation, and motion trajectory self-generation to address the insufficient adaptability of traditional impedance control in unstructured environments. The experimental results show that this method takes neural networks as intelligent perception units to estimate system uncertainty in real-time, variable stiffness controllers as fast response units to adjust interactive dynamics, and adaptive trajectory generators as precise execution units to achieve target tracking, significantly improving the interactive performance of the robotic arm in unknown or dynamic environments. Through the closed-loop collaboration of layered intelligent modules, this study effectively balances the system's response speed, control accuracy, and model dependence. This method helps to promote the transformation of robots from repetitive programming tools to intelligent partners with environmental perception and autonomous decision-making capabilities, providing a scalable technological path for building higher-level robot control systems. However, this study still has certain limitations. The adjustable parameter settings are limited in performance analysis, and there is a lack of comprehensive testing of various performance indicators of the robotic arm. Safety restrictions and potential risks in unknown contact environments are not fully considered. In addition, if the online learning process of neural networks lacks appropriate regularization mechanisms, it may over-fit to instantaneous disturbances, affecting its generalization ability and system stability. This study mainly focuses on contact force control, ignoring other potential influencing factors, and has not yet validated the computational efficiency of this method in multi joint systems, human-computer interaction, and real-time scenarios. Future work will extend single joint modules in parallel to multi-degree-of-freedom robotic arms, combining meta learning and model compression techniques to achieve rapid adaptation and lightweight deployment on low-cost hardware. Meanwhile, a security boundary protection mechanism is introduced, and regularization and online learning stabilization strategies are adopted to suppress over-fitting. Future research will further consider sensor delay and actuator saturation effects, verify zero sample generalization ability in high-frequency interference and shared workspaces, and promote the development of safe human-machine integration.

References

[1] Dongchen Liu, Junzheng Wang, and Dawei Shi. Impedance control for a Stewart-structure-based wheel-legged robotic system in wheel motion. International Journal of Robust and Nonlinear Control, 34(8):5346-5363, 2024. https://doi.org/10.1002/rnc.7267

[2] Peng Wang, and Renquan Dong. Research on fuzzy impedance control of upper-limb rehabilitation robot based on membership functions online optimization learning strategy. Industrial Robot: The International Journal of Robotics Research and Application, 51(1):58-72, 2024. https://doi.org/10.1108/ir-07-2023-0146

[3] Gabriel Duarte Gonçalves Pedro, Gabriel Bermudez, Vivian Suzano Medeiros, Hélio Jacinto da Cruz Neto, Luiz Guilherme Dias de Barros, Gustavo Pessin, Marcelo Becker, Gustavo Medeiros Freitas, and Thiago Boaventura. Quadruped robot control: An approach using body planar motion control, legs impedance control and bézier curves. Sensors (Basel, Switzerland), 24(12):3825-3825, 2024. https://doi.org/10.3390/s24123825

[4] Jing Xin, Loris Roveda, Jianfei Li, Y Wang, and Haibo Gao. An adaptive impedance control for dual-arm manipulators incorporated with the virtual decomposition control. Journal of Vibration and Control, 30(11-12):2647-2660, 2024. https://doi.org/10.1177/10775463231182462

[5]Li Li, Tong Huang, Chujia Pan, J. F. Pan, and Wenbin Su. Impedance control for force tracking of a dual-arm cooperative robot based on particle swarm optimization. Industrial Robot: The International Journal of Robotics Research and Application, 51(3):436-445, 2024. https://doi.org/10.1108/IR-10-2023-0247

[6]Pengyu Zhao, Anhuan Xie, Shiqiang Zhu, Lingkai Chen, Lingyu Kong, and Dan Zhang. A novel impedance control based on equivalent stiffness for hydraulic single-leg robot. International Journal of Control, Automation and Systems, 22(5):1636-1653, 2024. https://doi.org/10.1007/s12555-022-0264-8

[7]Xuhui Zhao, Jia Liu, Shengqiang Yang, Jingjing Zhang, Xufeng Lv, Long Cheng, and Xueqian Zhang. An adaptive impedance control method for blade polishing based on the Kalman filter. The International Journal of Advanced Manufacturing Technology, 132(3-4):1723-1739, 2024. https://doi.org/10.1007/s00170-024-13401-5

[8]Yanan Fan, Zhongcai Pei, and Zhiyong Tang. Variable impedance control for a single leg of a quadruped robot based on contact force estimation. International Journal of Control, Automation and Systems, 22(4):1360-1370, 2024. https://doi.org/10.1007/s12555-022-0601-y

[9] Yuexuan Xu, Xin Guo, Gaowei Zhang, Jian Li, Xingyu Huo, Bokai Xuan, Zhifeng Gu, and Hao Sun. A learning control strategy for robot-assisted bathing via impedance sliding mode technique with non-repetitive tasks. International Journal of Control, Automation and Systems, 22(3):946-962, 2024. https://doi.org/10.1007/s12555-022-0436-6

[10] Wenhao Zhang, Peng Song, Mingying Wu, Qiang Li, Xinmin Mo, and Pingxin Ji. Unknown system dynamics estimator-based impedance control for lower limb exoskeleton with enhanced performance. Control Theory and Technology, 22(1):56-68, 2024. https://doi.org/10.1007/s11768-023-00189-0

[11] A Boulkroune, F Zouari, and A Boubellouta. Adaptive fuzzy control for practical fixed-time synchronization of fractional-order chaotic systems. Journal of Vibration and Control, 2025:10775463251320258.

[12] Abdesselem Boulkroune, Sarah Hamel, Farouk Zouari, Abdelkrim Boukabou, and Asier Ibeas. Output-feedback controller based projective lag-synchronization of uncertain chaotic systems in the presence of input nonlinearities. Mathematical Problems in Engineering, 2017(1):8045803, 2017. https://doi.org/10.1155/2017/8045803

[13] Farouk Zouari, Kamel Ben Saad, and Mohamed Benrejeb. Robust neural adaptive control for a class of uncertain nonlinear complex dynamical multivariable systems. International Review on Modelling and Simulations, 5(5):2075-2103, 2012.

[14] Farouk Zouari, Kamel Ben Saad, and Mohamed Benrejeb. Adaptive backstepping control for a class of uncertain single input single output nonlinear systems. 10th International Multi-Conferences on Systems, Signals & Devices 2013 (SSD13). 2013:1-6, 2013. https://doi.org/10.1109/SSD.2013.6564134

[15] S K C Tulli. Artificial intelligence, machine learning and deep learning in advanced robotics, a review. International Journal of Acta Informatica, 3(1):35-58, 2024.

[16] Gerasimos Rigatos, Masoud Abbaszadeh, Bilal Sari, and Pierluigi Siano. Nonlinear optimal control for a gas compressor driven by an induction motor. Results in Control and Optimization, 11:100226, 2023. https://doi.org/10.2478/pead-2023-0014

[17] Yongming Li, Shaocheng Tong, and Tieshan Li. Adaptive backstepping control for a single-link flexible robot manipulator driven DC motor. 14(1):483-494, 2013. https://doi.org/10.1016/j.nonrwa.2012.07.010

[18]Yuichiro Taira, Shinichi Sagara, and Masahiro Oya. Impedance control based on error feedback for the manipulator of an underwater vehicle-manipulator system. Artificial Life and Robotics, 28 (4):830-849, 2023. https://doi.org/10.1007/s10015-023-00896-6

[19]Xujing Tian, Mengyao Lv, Jiazheng Sun, Hongzheng Zhao, Ziyuan Jiang, Jinshuo Han, Wei Gu, and Gang Cheng. An adaptive impedance control method for polishing system of an optical mirror processing robot. Robotica, 42(1):21-39, 2023. https://doi.org/10.1017/S0263574723001315

[20] Abdessamed Echikr, Ali Yachir, Chaker Abdelaziz Kerrache, Abdelkrim Kamel Oudjida, and Zakaria Sahraoui. Interoperable IoRT for healthcare: Securing intelligent systems with decentralized blockchain. Acta Informatica Pragensia, 13(2):168-192, 2024. http://aip.vse.cz/doi/10.18267/j.aip.233.html