International Journal of Control

ISSN: (Print) (Online) Journal homepage: https://www.tandfonline.com/loi/tcon20

DDPG-based controlling algorithm for upper limb prosthetic shoulder joint

Yuting Guo, Baojiang Li, Sotirios Spanogiannopoulos, Haiyan Wang & Jibo Bai

To cite this article: Yuting Guo, Baojiang Li, Sotirios Spanogiannopoulos, Haiyan Wang & Jibo Bai (2023): DDPG-based controlling algorithm for upper limb prosthetic shoulder joint, International Journal of Control, DOI: 10.1080/00207179.2023.2201644

To link to this article: https://doi.org/10.1080/00207179.2023.2201644

Accepted author version posted online: 14 Apr 2023.

Publisher: Taylor & Francis & Informa UK Limited, trading as Taylor & Francis Group
Journal: *International Journal of Control*
DOI: 10.1080/00207179.2023.2201644

Equation Chapter 1 Section 1DDPG-based controlling algorithm for upper limb prosthetic shoulder joint

Yuting Guo[a,b], Baojiang Li[a,b]*, Sotirios Spanogiannopoulos[c], Haiyan Wang[a,b] and Jibo Bai[a,b]

[a]The School of Electrical Engineering, Shanghai DianJi University, Shanghai, China.
[b]Intelligent Decision and Control Technology Institute, Shanghai Dianji University, Shanghai, China
[c]Electronic Engineering with specialization, University of Kent (UK), Canterbury, England

*Corresponding author(s). E-mail(s): libj@sdju.edu.cn. 206001010403@st.sdju.edu.cn., sotiris.spanogianopoulos@gmail.com. wanghaiyan@sdju.edu.cn. 206001010303@st.sdju.edu.cn.

Abstract:

The development of intelligent prostheses has effectively improved the life of amputees. However, the current prosthetics mainly focus on restoring the basic mobility of amputees, without considering the use habits of the wearer and the diversity of arm movements, which makes the unknown interference and complex control requirements in daily life an obstacle to the use of prosthetics. To solve this problem, this paper proposes a combination method of adaptive control algorithms of bionic arm shoulder joint based on DDPG to realize intelligent control of the shoulder joint of upper limb prosthesis. Based on using adaptive control to reduce the interference of external variables, the accuracy of the joint module system is improved through reinforcement learning. The results show that the controller has a good effect on improving the dynamic performance of the mechanical system and can be widely used in bionic mechanical control.

Keywords: Reinforcement learning, Bionic arm, Adaptive control, Joint module

1. Introduction

The absence of arms often makes it difficult for upper limb amputees to take care of themselves in life and work. In recent years, the progress of technology has aroused great interest in intelligent prostheses and has made great progress in structural design and control methods(Precup et al, 2020), but for patients with self-shoulder amputation, which the bone joint has been lost humerus or part of the scapula, the function of the entire arm has been lost. Some prostheses are not suitable for such amputees (Yang et al, 2021), which has prompted research into bionic prostheses to replace the entire upper extremity.

The shoulder carries the movement of the whole arm. The load capacity and control accuracy of this position determines whether the bionic arm can operate normally. In this context, the joint module has the characteristics of high torque and small volume, providing a new idea for the modular design of new bionic machinery (Lee et al, 2018). Taking the joint module as the power source of the bionic arm can solve the problems of low flexibility and large weight of the bionic arm (Vincitorio et al, 2020). The joint module driven by a permanent magnet brushless DC (BLDC) motor can better fit with the human body and become the power source of the bionic arm because of its lower noise and excitation loss, higher output torque (Kumar et al, 2016). Compared with traditional synchronous motors, BLDC has the advantages of a simple structure, reliable operation, and high efficiency. but BLDC is also a complex object with multiple variables, strong coupling, and nonlinear and variable parameters. In order to complete the matching between the joint module and human joint, and obtain better control performance to maximize the restoration of limb function, it is necessary to design the joint module controller reasonably.

Another function of the arm joint is to move the hand flexibly to the target position in continuous motion. In fact, the human arm is often accompanied by a rapid change of

action and the sudden change of load in daily life. When reflected on the joint module, it shows the rapid change in speed and the sudden change of load torque (Kim et al, 2012; Lee et al, 2018). But the complex structure, insufficient system modeling, disturbance of torque change, friction coefficient, and other system parameters (Zhen et al, 2021) will affect the application of the joint module. For these problems, the traditional controller(Preitl et al, 2006; Carlucho et al, 2020) only be applied to the linear control system with low requirements for control performance. For the upper limb prosthetic, which is a controlled system with real-time change of controlled parameters or uncertain initial quantity, the traditional control algorithm often has large inaccuracy and often causes overshoot and oscillation, which cannot meet the requirements of control performance. The control with learning ability can simulate the actions and mechanisms of organisms, and better restore the functions that can be performed by the missing body parts.

Since the development of machine learning, reinforcement learning (RL), as a branch of machine learning, has achieved vigorous development. Unlike supervised learning and unsupervised learning(Bengio et al, 2013; Schmidhuber, 2015), reinforcement learning emphasizes that agents get the maximum reward in the process of interaction with the environment, to continuously improve the strategy until it reaches the optimal (Gheibi et al, 2020). Reinforcement learning is between supervised learning and unsupervised learning. It is called approximate dynamic programming or neuro dynamic programming (Su et al, 2018). Its essence is that agents collect data in the process of interacting with the environment and learn from the data to obtain the optimal strategy. At present, reinforcement learning has been widely used in the field of robot control. Xie et al. (2019) used deep reinforcement learning to jointly train the gait of the biped robot Kasi. Kasi gained the ability to walk in different scenes in a short time, including up and down steps,

jumping on the ground, walking, etc. Through deep reinforcement learning, the Kasi robot can also maintain walking balance by adjusting the size and frequency of its steps in the face of various unexpected events.

The adaptive control algorithm can modify its characteristics to adapt to the changes in the dynamic characteristics of the object and disturbance, which is the appropriate choice of the artificial limb controller. Chen et al. (2022) constructed an adaptive backstepping control scheme to improve the dynamic tracking performance of human-robot training mode in the presence of recognition error. Based on adaptive control, once the movement relationship between the environment of the reinforcement learning network and human motion is established, we can establish humanoid motion planning, arm power output, etc. At the same time, we can adjust the influence of the fuzzy processing of adaptive control, such as the reduction of the control accuracy of the system, through the reward and punishment mechanism. In our research, we proposed a basic adaptive feedback control based on the Lyapunov method (Spyros G. et al, 2014), then the Deep Deterministic Policy Gradient (DDPG) algorithm in reinforcement learning is introduced to design the control algorithm of the joint module, which can update the control strategy according to load disturbance, effectively improve the anti-interference ability of the system and reduce speed signal tracking error.

Based on the kinematic and dynamic analysis of the bionic arm, the main contributions of our work can be summarized as follows: 1)The mathematical model of joint modules under unknown disturbance is further studied in this paper. 2)A DDPG-based composition approach to the adaptive control (RLAC) algorithm is presented by collecting and judging the parameters of the joint module system. 3) The effectiveness of the proposed algorithm is analyzed through an evaluation experiment, and the uncertainty of human shoulder joint movement caused by external interference is fully considered.

Experiments show that the proposed algorithm can achieve excellent track tracking and is robust for variable loads in practice.

This article is organized as follows. Section 2 introduces related works in machine learning and intelligent control. In Section 3, it describes the shoulder joint module system structure and dynamic model. In Section 4, The stability analysis for RLAC algorithm is introduced in detail. Section 5 presents numerical Simulation and experimental results analysis. conclusions are given in Section 6.

2. Related Work

In the past few decades, the research of modern dynamic control algorithms has made great progress (Rigatos et al, 2022), Saadaoui et al. (2017) adopted sensorless speed control based on sliding mode observer and estimated the PMSM rotor position and speed according to the back electromotive force voltage. Yin et al. (2019) optimized the integrated position and velocity loop of PMSM by sliding mode control. Zhou et al. (2019) designed a drive system for a series of manipulators based on orthogonal fuzzy PID control. Wang et al. (2021) present a nonlinear optimal finite-time tracking controller based on a state-dependent equation for the multi-motor driving system. Fang et al. (2021) proposed a robust tracking control for a magnetic wheel mobile robot based on adaptive dynamic programming. Lu et al. (2020) realized the humanoid motion of the robot arm by mapping the joint angle of the robot arm to the corresponding joint angle of the human arm. The joint angle is calculated by an inverse kinematics algorithm based on elbow constraint. The T/P method can make the robot arm effectively obtain the humanoid motion path. However, when the robot arm needs to move to a new target point, the problem that the robot arm cannot independently generate the humanoid motion path will arise, and a new T/P operation is required. This method lacks flexibility and is suitable for limited scenarios. Chen et al. (2022) designed a variable admittance controller to

reduce the real-time interaction torque of human exoskeletons. At the same time, the extended state observer with backstepping iteration is used to compensate for the unmeasured system state, model uncertainty, and the unmodeled dynamics of the lower limb exoskeleton.

Since the development of artificial intelligence, researchers have been trying to combine motion control with neural networks to improve performance, and used BP neural network (BPNN) or convolutional neural network (CNN) to achieve better control effect (Yang et al, 2019; Khan et al, 2020). EI-Sousy et al. (2018) proposed a nonlinear robust optimal control scheme for an uncertain two-axis motion control system based on adaptive dynamic programming and neural network. Liu et al. (2022) studied the real-time cooperative control of multiple robots in a distributed scene based on a dynamic neural network. Su et al. (2019) used a new depth convolution neural network structure to reconstruct the relationship between the pose and rotation angle of the robot arm. The anthropomorphic motion of the redundant manipulator is realized by rotating motion. Zamfirache et al. (2022) proposed a new control method based on reinforcement learning, using Policy Iteration and a meta-heuristic Grey Wolf Optimizer algorithm to train the neural network. Cho et al. (2012) projected teaching data into potential space and the gaussian mixture model. With the increase of data obtained, the gaussian mixture model is gradually optimized. With this method, the robot arm can more accurately reproduce the human arm movement.

Reinforcement learning obtains rewards through the environment and guides the system's behavior, providing another way to solve problems for the perception and decision of a complex environment. In recent years, more and more researchers have participated in the research progress of deep reinforcement learning. Not only put forward many improvement strategies but also began to apply the research results of deep

reinforcement learning to practical engineering applications(Han et al, 2019; Chai et al, 2020). Zhang et al. (2019) used the reinforcement learning method to realize robot vehicle navigation path smoothing and tracking control. Chen et al. (2018) proposed a speed servo system control strategy based on a Reinforcement learning algorithm, which effectively overcomes the inertia mutation and torque disturbance of the DC motor. Song et al. (2021) study the deep reinforcement learning speed control strategy for the PMSM servo system. Liu et al. (201) studied the internal reward function of human-computer cooperative safety interaction of industrial robots based on deep reinforcement learning. However, most of the existing research has studied humanoid motion in an obstacle-free environment and has not involved the influence of joint control on humanoid motion. Based on the depth analysis of the dynamic performance of the joint module of the upper limb prosthesis, this paper introduces the depth deterministic strategy gradient algorithm, focusing on the impact of the speed and load changes caused by the environment on the joints, which is more suitable for the control application of the upper limb prosthesis.

3. System description

Although the characteristics of reinforcement learning do not require a given model in advance, to obtain accurate motion control and reduce the difficulty of training, it is still necessary to extract the physical parameters of the joint module as accurately as possible. In order to achieve low cost and popularization, we describe the mathematical modeling of joint modules as comprehensively as possible and consider the abnormal vibration of joints caused by sudden load changes.

3.1. Shoulder system

We place the joint module on the shoulder to replace the pitching motion of the human arm. The shoulder is the connection between the human arm and the trunk, the load and

speed change of the whole arm is finally gathered here. Simply, the action of the lower arm of the shoulder joint can regard as a connecting rod with changing torque and speed. Therefore, the load capacity and control accuracy of the joint module directly determine the dynamic performance of the bionic arm. The position of the joint module is shown in Figure 1.

3.2. Mathematical model of joint module

The joint module consisted of a Driver, brushless DC motor, brake, harmonic reducer, etc. (Figure 1). In the dynamic analysis, the action of the lower limb of the shoulder is simplified as the moment of inertia with the constant change of centroid. The mathematical model of the joint module is as follows. Firstly, the expression of the BLDC is:

$$\begin{bmatrix} u_a \\ u_b \\ u_c \end{bmatrix} = \begin{bmatrix} R_s & 0 & 0 \\ 0 & R_s & 0 \\ 0 & 0 & R_s \end{bmatrix} \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} + \begin{bmatrix} L & 0 & 0 \\ 0 & L & 0 \\ 0 & 0 & L \end{bmatrix} \begin{bmatrix} \dot{i}_a \\ \dot{i}_b \\ \dot{i}_c \end{bmatrix} + \begin{bmatrix} e_a \\ e_b \\ e_c \end{bmatrix} \tag{1}$$

$$i_a + i_b + i_c = 0 \tag{2}$$

where u_a, u_b, u_c are phase voltages of the three-phase winding, respectively. i_a, i_b, i_c are phase currents of the three-phase winding, respectively. e_a, e_b, e_c are respective back electromotive forces of the three-phase winding, respectively. R_s is the phase resistance of the BLDC. L_o is the equivalent inductance of the BLDC.

$$L_o = L_S - M \tag{3}$$

where L_S is the self-inductance of each phase winding, and M is the mutual inductance between each winding.

When the two-stage three-phase BLDC is running, only two phases of the three-phase winding are conducted. we can get the electromagnetic torque of the BLDC as:

$$\tau = \frac{e_a i_a + e_b i_b + e_c i_c}{\omega_r} \tag{4}$$

ω_r is the angular velocity of the BLDC. So, Eq. (4) can be written as:

$$\tau = \frac{2 e_a i_a}{\omega_r} \tag{5}$$

Since the phase voltage is related to the number of winding turns, there are:

$$e_a = k_e n \tag{6}$$

where k_e is a constant, and the relationship between the angular velocity of the BLDC and the number of coil turns is:

$$\omega_r = \frac{2\pi n}{60} \tag{7}$$

Therefore, Eq. (5) can be rewritten as:

$$\tau = \frac{2 k_e n i_a}{\omega_r} = \frac{60}{\pi} k_e i_a = k i_a \tag{8}$$

k is a constant. Then the torque of the BLDC is proportional to the phase current.

According to the effects of the harmonic reducer:

$$T_L = \lambda \eta T_{lp} \tag{9}$$

where T_L is the torque of the whole bionic arm system acting on the shoulder joint. to the joint module, λ is the reduction ratio, and η is the transmission efficiency. The dynamic model of the joint module is:

$$J \dot{\omega}_r + B \omega_r + \frac{T_L}{\lambda \eta} + T_f = \tau \tag{10}$$

where J is the moment of inertia of the rotor of the BLDC, B is the viscous friction coefficient of the BLDC, and T_f is the friction torque in the harmonic reducers.

In order to facilitate adaptive control, the a-b-c three-phase must be converted to the d-q reference frame. According to Kirchhoff laws, there has $U_a + U_b + U_c = 0$, the composite space vector U_t can be expressed as:

$$U_t = U_a + U_b e^{j\frac{2\pi}{3}} + U_c e^{j\frac{4\pi}{3}} = \frac{3}{2} V_m e^{j\theta} \tag{11}$$

The V_m is maximum phase voltage, θ is the angle of the rotor. According to the Clark transform, the constant amplitude conversion is written as:

$$\begin{bmatrix} u_\alpha \\ u_\beta \end{bmatrix} = k \begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix} \begin{bmatrix} u_a \\ u_b \\ u_c \end{bmatrix} \tag{12}$$

here $k = \frac{2}{3}$. In order to facilitate control and calculation, Park transform is calculated as:

$$\begin{bmatrix} u_d \\ u_q \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} u_\alpha \\ u_\beta \end{bmatrix} \tag{13}$$

the voltage expression can be obtained as:

$$\begin{cases} u_q = R_s i_q + L_s \dot{i}_q + L_s \omega_r i_d + \phi \omega_r \\ u_d = R_s i_d + L_s \dot{i}_d - L_s \omega_r i_q \end{cases} \tag{14}$$

u_q and u_d are the phase voltages of the reference coordinate system, i_q and i_d are phase currents, ϕ are the flux linkages produced by the permanent magnets of the BLDC.

According to the field-oriented control (Rezaei et al, 2017; Zhao et al, 2019), $i_d = 0$, Eq. (14) is rewritten as:

$$\dot{i}_q = -\frac{R_s}{L_s} i_q - \frac{\phi}{L_s} \omega_r + \frac{1}{L_s} u_q \tag{15}$$

The friction torque is nonlinear and has a great impact on the dynamic performance of the joint module system. Scholars used model friction forces to calculate the influence of different friction forces on mechanical systems and proposed different friction models to

eliminate the influence of friction (Pennestrì et al, 2016). Here we combine it with the load torque. Define:

$$T_o = \frac{T_L}{\lambda \eta} + T_f \tag{16}$$

In the d-q reference frame, the relationships between the angular velocity of the joint module and the phase current are:

$$\dot{\omega}_r = \frac{3p^2}{2J}\phi i_q - \frac{B}{J}\omega_r - \frac{p}{J}T_o \tag{17}$$

where, p is the number of pole pairs of the BLDC.

Combining Eq. (15) and Eq. (17), the state space of the joint module system is obtained as:

$$\begin{bmatrix} \dot{\omega}_r \\ \dot{i}_q \end{bmatrix} = \begin{bmatrix} -\frac{B}{J}\omega_r + \frac{3p^2\phi}{2J} i_q \\ -\frac{\phi}{L_s}\omega_r - \frac{R_s}{L_s} i_q \end{bmatrix} + \begin{bmatrix} -\frac{p}{J} \\ 0 \end{bmatrix} T_o + \begin{bmatrix} 0 \\ \frac{1}{L_s} \end{bmatrix} u_q \tag{18}$$

4. Controller design

4.1. Adaptive control

In order to detect variables and adapt to changes in work and life, adaptive control is designed. Firstly, the tracking error of angular velocity is defined as:

$$e_1 = \omega_{rd} - \omega_r \tag{19}$$

Hence, $\dot{e}_1$ is calculated as:

$$\dot{e}_1 = \dot{\omega}_{rd} - \dot{\omega} = \dot{\omega}_{rd} - \frac{3p^2\phi}{2J} i_q + \frac{B}{J}\omega_r + \frac{p}{J}T_o \tag{20}$$

where i_q is the controlled quantity. Currently, consider T_o as the known quantity. In order to obtain stable feedback, take

$$i_q = \frac{1}{\mu}\left(\dot{\omega}_{rd} + \frac{B}{J}\omega_r + \frac{p}{J}T_o + k_1 e_1 \right) \tag{21}$$

Here $\mu = \frac{3p^2\phi}{2J}$, $\dot{e}_1 = -k_1 e_1$. When the arm moves, T_o and T_f change and become unknown, the estimated value $\hat{T}_L$ is used to replace T_L to maintain the desired performance. Eq. (16) is redefined as:

$$\hat{T}_o = \frac{\hat{T}_L}{\lambda\eta} + T_f \tag{22}$$

Eq. (21) become:

$$\left(i_q\right)_{rd} = \frac{1}{\mu}\left(\dot{\omega}_{rd} + \frac{B}{J}\omega_r + \frac{p}{J}\hat{T}_o + k_1 e_1 \right) \tag{23}$$

Redefine the error signal containing the estimated variable as:

$$e_2 = \left(i_q\right)_{rd} - i_q = \frac{1}{\mu}\left(k_1 e_1 + \dot{\omega}_{rd} + \frac{B}{J}\omega_r + \frac{p}{J}\hat{T}_o \right) - i_q \tag{24}$$

Hence, Eq. (20) can be written as:

$$\dot{e}_1 = -k_1 e_1 + \mu e_2 - \frac{p}{J}\tilde{T}_o \tag{25}$$

where $\tilde{T}_o = \hat{T}_o - T_o$ is the estimation error. Through the adjustable gain adaptation rate, the dynamic equation $\dot{e}_2$ is:

$$\dot{e}_2 = \frac{d\left(i_q\right)_{rd}}{dt} - \frac{di_q}{dt} = -\frac{1}{\mu}k_1^2 e_1 + k_1 e_2 - \frac{1}{L_s}u_q - \frac{k_1 p}{\mu J}\tilde{T}_o + \lambda \tag{26}$$

and

$$\lambda = \frac{1}{\mu}\omega_{rd} + \left(\frac{\phi}{L_s} - \frac{B^2}{\mu J^2} \right)\omega_r + \left(\frac{B}{J} + \frac{R_s}{L_s} \right) i_q + \left(\frac{p}{\mu J} - \frac{Bp}{\mu J^2} \right)\hat{T}_o$$

To keep the expected performance of the joint module under the influence of unknown resistance, an adaptive feedback controller is proposed for the system Eq. (18).

Theorem 1: Considering the estimated variables Eq. (19) and Eq. (24), there exists $k_1, k_2 > 0$ and $k_T > 0$ such that the following controller stabilizes the joint module system.

$$\begin{cases} u_q = L_s \left[\left(\mu - \dfrac{k_1^2}{\mu} \right) e_1 + \left(k_1 + k_2 \right) e_2 + \lambda \right] \\ \tilde{T}_o = k_T \left[e_1 + \dfrac{k_1}{\mu} e_2 \right] \end{cases} \tag{27}$$

Therefore, the closed-loop joint module system constituted of Eq. (21) and Eq. (27) is shown in Figure 2.

Proof: Define a Lyapunov function as:

$$V_e = \frac{1}{2} \left[e_1^2 + e_2^2 + \frac{1}{k_T J} \tilde{T}_o^{\,2} \right] \tag{28}$$

The V_e is positive definite. Then take:

$$\dot{V}_e = e_1 \dot{e}_1 + e_2 \dot{e}_2 + \frac{p}{k_T J} \tilde{T}_o \dot{\tilde{T}}_o \tag{29}$$

From Eq. (25) and Eq. (26), we can get:

$$\dot{V}_e = -k_1 e_1^2 - k_2 e_2^2 + \frac{p}{J} \tilde{T}_o \left[-e_1 - \frac{k_1}{\mu} e_2 + \frac{1}{k_T} \tilde{T}_o \right]$$

$$+ e_2 \left[\left(\mu - \frac{k_1^2}{\mu} \right) e_1 + \left(k_1 + k_2 \right) e_2 + \lambda - \frac{u_q}{L_s} \right] \tag{30}$$

Substituting Eq. (21) and Eq. (27), it is obtained that:

$$\dot{V}_e = -k_1 e_1^2 - k_2 e_2^2 < 0 \tag{31}$$

Therefore, the Lyapunov stability condition is fully satisfied. It can be concluded that the state variables of the system would converge to zero in finite time.

4.2. DDPG algorithm

The DDPG algorithm follows the Actor-Critic architecture, in which the actor network learns the parameterized strategy through the policy gradient algorithm, and the critic network learns the value function to evaluate the state action obtained from the algorithm. A multi-layer fully connected layer is used to build an action network and evaluation network which is more suitable for continuous action space. DDPG absorbs the advantages of a single-step update of strategy gradient in Actor-Critic while retaining the skills of Q value estimation in DQN, which can achieve more effective learning while performing actions in the continuous time domain.

The main function of the actor network is to input the status s_t into the policy network μ to output an action a_t. For continuous actions, the activation function of the output layer generally uses the Tanh function or Sigmoid function. The policy network is mainly responsible for generating actions, and its gradient can be calculated as follows:

$$\nabla_{\theta^\mu}\mu = \sum_i \nabla_{\theta^\mu}\mu\left(s\middle|\theta^\mu\right)\middle|s=s_i \nabla_a Q\left(s,a\middle|\theta^Q\right)\middle|s=s_i, a=\mu\left(s_i\right) \tag{31}$$

The critic network is based on the value function to fit the state action-value function. In state s_t, the actor network executes the action selected by the agent according to the action policy $a_t = \mu\left(s_t\right)$. The expected state-action value obtained by the agent is expressed as:

$$Q\left(s_t, \mu\left(s_t\right)\middle|\theta^Q\right) = E\left[r\left(s_t, \mu\left(s_t\right)\right) + \gamma Q\left(s_{t+1}, \mu\left(s_{t+1}\right)\middle|\theta^Q\right)\right] \tag{32}$$

The update process of the value function network is as follows. First enter state s_t and action a_t, The critic network outputs $Q\left(s_t, a_t\right)$, which is part of the input loss function module. At the same time, s_{t+1} is input to the target strategy network and is input to the target value function network together with the expected action a' expected to be

performed in the next step. $Q'(s_{t+1}, a')$ is output as another part of the loss function module. The network structure diagram is shown in Figure 3.

4.3. RLAC algorithm

Aiming at the requirements of adaptive and self-learning ability of control algorithm for joint module system, an adaptive controller based on DDPG is proposed in this paper. The controller learns the system model according to the input and output, the actor network realizes the optimal expression of the strategy, and the critic network realizes the optimal approximation of the value function.

By analysing the input and output data of the system, the controller trains the simulation agent with batch data, and finally obtains the agent that can make decisions according to the changes of the environment. The agent adjusts the parameters of the adaptive controller in real time, and finally improves the accuracy of the joint module system.

The reinforcement agent is composed of policy initialization error $e_1(t)$, $e_2(t)$, error integral $\int e_1(t)$, $\int e_2(t)$ and feedback $\omega_r(t)$, and the state vector s_t is used to represent the state characteristics of the reinforcement agent system at the current time.

$$s_t = \left[e_1(t), e_2(t), \int e_1(t), \int e_2(t), \omega_r(t) \right]^T \tag{33}$$

In the initial state, an enhanced agent action output a_t' is mapped according to the current actor online strategy μ and the random process noise. After the execution of the controlled object, the reward value r_t and next moment state s_{t+1} will constitute the next enhanced agent action.

$$s_t = \left[e_1(t), e_2(t), \int e_1(t), \int e_2(t), \omega_r(t) \right]^T \tag{34}$$

The actor-network stores this state transition (s_t, a_t, r_t, s_{t+1}) process in memory M. Random sampled in the memory M as a small round training data of the online network.

After the system completes the N-step sampling, the target network Q^- and μ^- are obtained to calculate the critical target network value as follows:

$$y_i = r_i + \gamma Q^-\left(s_{t+1}, \mu\left(\left(s_{i+1}\middle|\theta^{\mu^-}\right)\theta^{e^-}\right)\right) \tag{35}$$

The discount factor γ value range as $0 < \gamma \le 1$, then the critical network is updated by minimizing the loss L.

$$L = \frac{1}{N}\sum_i\left(y_i - Q\left(s_i, a_i\middle|\theta^Q\right)\right)^2 \tag{36}$$

The estimated value of the critical output state and the minimum loss L is an important basis for judging the decision-making degree of the actor network. actor network is constantly updated according to the loss gradient $\nabla_{\theta\mu}J$. After several iterations of self-learning and self-tuning, a suitable reinforcement learning agent is obtained.

$$\nabla_{\theta\mu}J \approx \frac{1}{N}\sum_i \nabla_a Q\left(s, a\middle|\theta^Q\right)\middle|_{s=s, a=\mu(s_1)} \nabla_{\theta\mu}\mu\left(s\middle|\theta^\mu\right)\middle|_{s_t} \tag{37}$$

Considering the influence of system error and feedback value range on system control performance, the reward function is defined as:

$$r_t = \delta_1 r_1(t) + \delta_2 r_2(t) \tag{38}$$

δ_1, δ_2 are the reward coefficients limiting the error value range, and $r_1(t)$, $r_2(t)$ are the error value range and feedback value range respectively, which are defined as:

$$r_1(t) = -c_1 e_1(t)^2 - c_2 e_2(t)^2 \tag{39}$$

$$r_2(t)\begin{cases} 0 & \omega_{r_1} \le \omega_r \le \omega_{r_2} \\ -1 & else \end{cases} \tag{40}$$

c_1, c_2 are reward parameter, which is set according to the importance of error value, ω_{r_1}, ω_{r_2} are the upper and lower limits of the speed. The DDPG-based composition

approach to the adaptive control algorithm is shown in Figure 4. The upper part of the dotted line is a parameter regulator based on reinforcement learning, which is composed of a reinforcement learning agent, and the lower part of the dotted line is composed of a controller and controlled object as the interactive object of the agent environment.

5. Simulation and experiment

In order to verify the effectiveness of the proposed algorithm and avoid potential risks to the human body, this study uses the method of evaluation experiments to analyze the control performance under different control algorithms, and then more accurately evaluate the adaptability of module joint. We choose different reference signals to simulate and experiment under the action of different load changes and compare them with the PID control. The selection of system parameters is shown in Table 1.

5.1. Parameters Selection

Reasonable parameters can balance the control effect and control cost, which play an important part to improve the control performance.

1) Selection of value k*:* The parameter k will determine the convergence speed of the control and the range of the convergence. The larger value k selected the faster the convergence rate is, but the control cost will also increase. That is to say, in actual applications, the response time of the system and the steady-state error should be weighed.

2) Selection of value i_d : In theory, the change of the internal flux linkage of the motor will affect the i_d , and then affect u_d . But this slight change is often ignored in the manufacturing and actual application of the brushless DC motor. So $i_d = 0$ was taken in the simulation.

3) Selection of reinforcement learning parameters: the parameters of reinforcement learning mainly act on the reward function, the definition of reward and punishment mechanism is different in environments, and the reward and punishment function involved will also change; The error value coefficient under the same definition also needs to be changed according to the importance of the weight.

5.2. Simulations

Generally, the training of reinforcement learning is uncertain. In order to reduce the unknown influence of experimental training on the equipment and test the convergence and effectiveness of the controller, we designed the simulation of the joint module system under the change of speed and load in MATLAB/Simulink. Compare with the waveform change with reinforcement learning adaptive controller and PID controller, we analyzed the performance of RLAC.

In order to make the simulation of the joint module closer to the cooperative application in real life, we use a sinusoidal signal and step signal as reference speed and use the step function to simulate the sudden change of load in cooperation. The parameters of the two controllers are shown in Table 2.

In the simulation environment, for the reinforcement learning agent, the model simulation time $T_f = 0.2s$ and the sampling time $T_s = 0.001s$. During training, set the maximum number of Episodes to 2000. When the trained Episode gets the maximum reward value, the obtained optimal RL Agent will be saved and applied to the simulation and experiment.

Figure 5 shows the start-up operation of the joint module under the initial load. The joint module can start quickly and reach the rated speed and run more smoothly than the PID control. The torque output under this condition is shown in Figure 6. It can be seen that RLAC is more stable than the torque output under PID control.

In order to test the anti-interference ability of the joint module system, the simulation speed is initially set at 300 m/s, and when t = 0.5 s, the load disturbance of 5 N/m is added. The speed comparison of fuzzy PID control and reinforcement learning adaptive control is shown in Figure 7. The torque output adapted to torque variation under this condition is shown in Figure 8. It can be seen that expect for stabilizing the speed, it also plays a certain role in balancing the internal vibration of BLDC by square wave drive.

The normal operation of the bionic arm often changes with the speed and load torque at the same time. In order to test the following performance of the system, the simulation speed input is a sinusoidal signal, and the speed following simulation results controlled by RLAC is shown in Figure 9. It can be seen from the velocity waveform that the adaptive control after reinforcement learning has excellent following performance.

5.3. Experimental verification

We can see that the design of the RLAC algorithm does not involve some complex sensors. To test the implementation ability of the designed controller, we improve the generated code of MATLAB and build the experimental platform combined with the simulation results and the parameters of the actual equipment. In order to make the experiment more suitable for combination with the human body. We keep the reinforcement learning controller running on the PC side, directly transmit the control signal to the driver through serial communication, set the DC voltage source to 30V, stm32f429 arm controller, communicate through CAN-bus, and the brake motor provides variable load torque. The experimental device is shown in Figure 10.

The experiment mainly tests the control effect of the controller on the speed change and load change of the joint module. Figure 11 showed the comparison of experimental results of the proposed controller and PID controller under step response. Figure 12 showed the comparison between the proposed RLAC algorithm and the sinusoidal signal

tracking experimental results of PID control under a given load. The control performance of the RLAC algorithm is better than that of PID control. In addition, when the joint velocity change rate of the two controllers is the largest, the maximum tracking error will appear. It may be caused by the backlash of the harmonic reducer, Coulomb friction, and sensor error.

One of the main uncertainties in the motion of a bionic robot arm is the sudden change of motion speed. The change of the external environment often needs the movement of the arm to adjust, which indicates that the joint module often needs to face a sudden change of speed, which will inevitably lead to an uncertain change in the parameters such as friction or dynamics of the system. Therefore, the closed-loop performance is verified by large positive and negative changes. The experimental comparison results are shown in Figure 13.

Under the requirements of continuous forward and reverse output changes, RLAC control has a smaller amplitude and faster convergence than PID control, which further proves the performance of the proposed RLAC algorithm. In addition, in the operation of the joint module, the changes in system parameters such as friction coefficient caused by the holding brake of the joint module and harmonic reducer will also have a nonlinear impact on the system output, but it still can be seen that the proposed RLAC algorithm has better robustness to the centralized uncertainty caused by the external environment.

6. Conclusion

In order to optimize the performance index of the joint module, this paper designed a DDPG-based composition approach to the adaptive control algorithm to solve the intelligent control problem of shoulder joint upper limb prosthesis. Considering the changes in load torque and moment of inertia caused by human motion, the depth deterministic strategy gradient algorithm is integrated into the adaptive control. The

motion state of the joint module is judged through the network, which improves the control accuracy and anti-interference ability of the joint module. Both simulation and experimental results show that the proposed RLAC algorithm can enhance the robustness of the joint module to load disturbances and improve the tracking performance of the speed control system. In practical application, the parameters of the control algorithm can be adjusted according to the habits of prosthetic subjects, which can greatly reduce the discomfort of amputees and realize the two-way coordination between human and intelligent prosthetics. This provides a novel method and idea for subsequent scientific research and industrial engineering applications.

Disclosure statement The authors declare that they have no conflicts of interest.

References

Bengio, Yoshua, Aaron Courville, and Pascal Vincent. "Representation learning: A review and new perspectives." IEEE transactions on pattern analysis and machine intelligence 35.8 (2013): 1798-1828.

Carlucho, Ignacio, Mariano De Paula, and Gerardo G. Acosta. "An adaptive deep reinforcement learning approach for MIMO PID control of mobile robots." ISA transactions 102 (2020): 280-294.

Chai, Jiazheng, and Mitsuhiro Hayashibe. "Motor synergy development in high-performing deep reinforcement learning algorithms." IEEE Robotics and Automation Letters 5.2 (2020): 1271-1278.

Chen, Pengzhan, et al. "Control strategy of speed servo systems based on deep reinforcement learning." Algorithms 11.5 (2018): 65.

Chen, Zhenlei, et al. "Gait prediction and variable admittance control for lower limb exoskeleton with measurement delay and extended-state-observer." IEEE Transactions on Neural Networks and Learning Systems (2022).

Chen, Zhenlei, et al. "Model identification and adaptive control of lower limb exoskeleton based on neighborhood field optimization." Mechatronics 81 (2022): 102699.

Cho, Sumin, and Sungho Jo. "Incremental online learning of robot behaviors from selected multiple kinesthetic teaching trials." IEEE Transactions on Systems, Man, and Cybernetics: Systems 43.3 (2012): 730-740.

El-Sousy, Fayez FM, and Khaled A. Abuhasel. "Nonlinear robust optimal control via adaptive dynamic programming of permanent-magnet linear synchronous motor drive for uncertain two-axis motion control system." 2018 IEEE Industry Applications Society Annual Meeting (IAS). IEEE, 2018.

Fang, Hongwei, et al. "Robust tracking control for magnetic wheeled mobile robots using adaptive dynamic programming." ISA transactions 128 (2022): 123-132.

Gheibi, Amir, et al. "Designing of robust adaptive passivity-based controller based on reinforcement learning for nonlinear port-Hamiltonian model with disturbance." International Journal of Control 93.8 (2020): 1754-1764.

Han, Xiaoning, et al. "Active object detection with multistep action prediction using deep q-network." IEEE Transactions on Industrial Informatics 15.6 (2019): 3723-3731.

Khan, Ameer Hamza, et al. "Tracking control of redundant mobile manipulator: An RNN based metaheuristic approach." Neurocomputing 400 (2020): 272-284.

Kim, Hwi-Su, et al. "Safe joint module for safe robot arm based on passive and active compliance method." Mechatronics 22.7 (2012): 1023-1030.

Kumar, Rajan, and Bhim Singh. "BLDC motor-driven solar PV array-fed water pumping system employing zeta converter." IEEE Transactions on Industry Applications 52.3 (2016): 2315-2322.

Lee, Won-Bum, et al. "Safe robot joint brake based on an elastic latch module." Mechatronics 56 (2018): 67-72.

Liu, Mei, Jiazheng Zhang, and Mingsheng Shang. "Real-time cooperative kinematic control for multiple robots in distributed scenarios with dynamic neural networks." Neurocomputing 491 (2022): 621-632.

Liu, Quan, et al. "Deep reinforcement learning-based safe interaction for industrial human-robot collaboration using intrinsic reward function." Advanced Engineering Informatics 49 (2021): 101360.

Lu, J. W., Q. J. Zhang, and H. L. Zhao. "Design and human-like motion research of service robot for the elderly." Chinese Journal of Engineering Design 27.2 (2020): 269-278.

Pennestrì, Ettore, et al. "Review and comparison of dry friction force models." Nonlinear dynamics 83.4 (2016): 1785-1801.

Precup, Radu-Emil, et al. "Evolving fuzzy models for prosthetic hand myoelectric-based control." IEEE Transactions on Instrumentation and Measurement 69.7 (2020): 4625-4636.

Preitl, Zsuzsa, et al. "Use of multi-parametric quadratic programming in fuzzy control systems." Acta Polytechnica Hungarica 3.3 (2006): 29-43.

Rezaei, Hamed, and Mohammad Javad Khosrowjerdi. "A polytopic LPV approach to active fault tolerant control system design for three-phase induction motors." International Journal of Control 90.10 (2017): 2297-2315.

Rigatos, G., K. Busawon, and M. Abbaszadeh. "A nonlinear optimal control approach for the truck and N-trailer robotic system." IFAC Journal of Systems and Control 20 (2022): 100191.

Saadaoui, Oussama, et al. "A sliding-mode observer for high-performance sensorless control of PMSM with initial rotor position detection." International Journal of Control 90.2 (2017): 377-392.

Schmidhuber, Jürgen. "Deep learning in neural networks: An overview." Neural networks 61 (2015): 85-117.

Song, Zhe, et al. "Deep reinforcement learning for permanent magnet synchronous motor speed control systems." Neural Computing and Applications 33.10 (2021): 5409-5418.

Spyros G. Tzafestas, 5-Mobile Robot Control I: The Lyapunov-Based Method, in: Introduction to Mobile Robot Control, Elsevier, 2014, pp. 137-183.

Su, Hang, et al. "Deep neural network approach in human-like redundancy optimization for anthropomorphic manipulators." IEEE Access 7 (2019): 124207-124216.

Su, Hang, et al. "Online human-like redundancy optimization for tele-operated anthropomorphic manipulators." International Journal of Advanced Robotic Systems 15.6 (2018): 1729881418814695.

Vincitorio, Francesca, et al. "Targeted muscle reinnervation and osseointegration for pain relief and prosthetic arm control in a woman with bilateral proximal upper limb amputation." World Neurosurgery 143 (2020): 365-373.

Wang, Minlin, et al. "SDRE based optimal finite-time tracking control of a multi-motor driving system." International Journal of Control 94.9 (2021): 2551-2563.

Xie, Zhaoming, et al. "Iterative reinforcement learning based design of dynamic locomotion skills for cassie." arXiv preprint arXiv:1903.09537 (2019).

Yang, Aolei, et al. "Humanoid motion planning of robotic arm based on human arm action feature and reinforcement learning." Mechatronics 78 (2021): 102630.

Yang, Xuhui, et al. "Predictive control modeling of ADS's MEBT using BPNN to reduce the impact of noise on the control system." Annals of Nuclear Energy 132 (2019): 576-583.

Yin, Zhonggang, et al. "Integrated position and speed loops under sliding-mode control optimized by differential evolution algorithm for PMSM drives." IEEE Transactions on Power Electronics 34.9 (2019): 8994-9005.

Zamfirache, Iuliu Alexandru, et al. "Policy iteration reinforcement learning-based control using a grey wolf optimizer algorithm." Information Sciences 585 (2022): 162-175.

Zhang, Wenyu, et al. "Double-DQN based path smoothing and tracking control method for robotic vehicle navigation." Computers and Electronics in Agriculture 166 (2019): 104985.

Zhao, Ximei, and Dongxue Fu. "Adaptive neural network nonsingular fast terminal sliding mode control for permanent magnet linear synchronous motor." IEEE Access 7 (2019): 180361-180372.

Zhen, Sheng Chao, et al. "A new PD based robust control method for the robot joint module." Mechanical Systems and Signal Processing 161 (2021): 107958.

Zhou, Haibo, et al. "Design and analysis of a drive system for a series manipulator based on orthogonal-fuzzy PID control." Electronics 8.9 (2019): 1051.

Table 1 The variables and parameters of the joint module system

Definition	Notation	Value	Unit
Moment of inertia of joint module system	J	1.6×10^{-3}	$kg\cdot m^2$
Equivalent inductance	L_S	83.97	mH
Phase resistance	R_S	0.687	Ω
Coefficient of viscous friction	B	8×10^{-3}	$Nm/rad/sec$
Rotor flux	ϕ	0.48	Wb
Number of pole pairs	p	4	–
The transmission ratio of harmonic reducer	λ	100	–
Transmission efficiency of harmonic reducer	η	0.95	–

Table 2 The parameters of control algorithms

Control algorithm	Control parameters

RLAC	$k_1 = k_2 = 100\ k_T = 4$
PID	$k_p = 50\ \ k_i = 3\ \ k_d = 10$

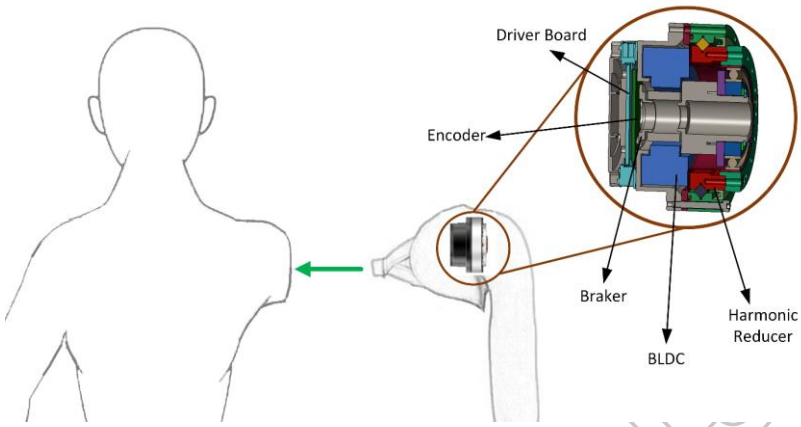

Figure 1. Joint module for upper limb prosthesis

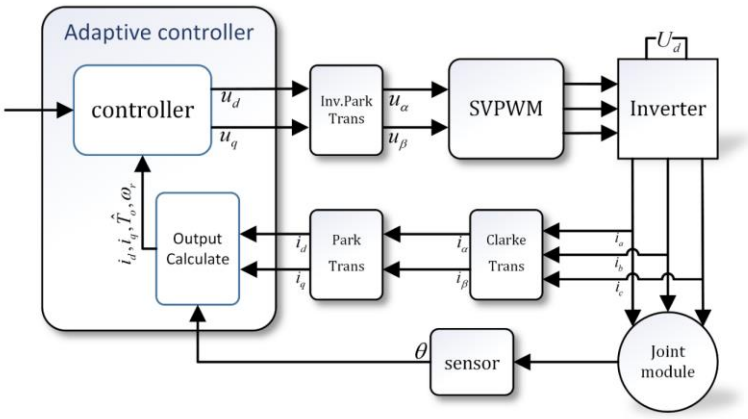

Figure 2. Structure of adaptive controller

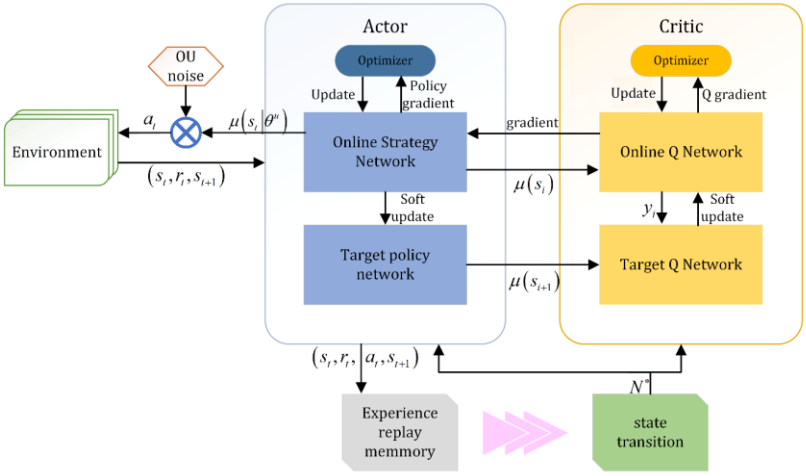

Figure 3. Network structure of reinforcement learning strategy

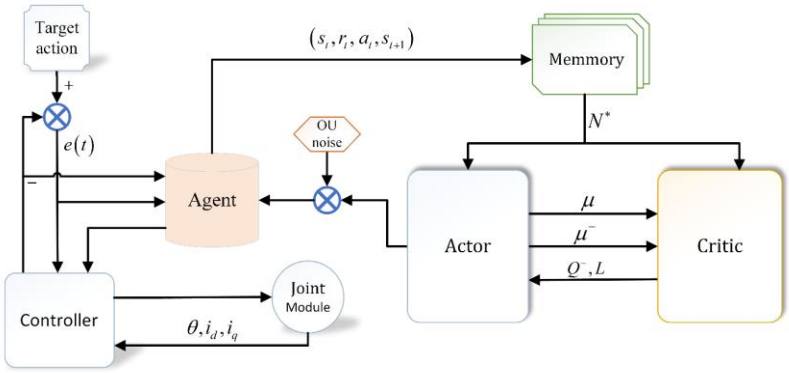

Figure 4. Structure of RLAC

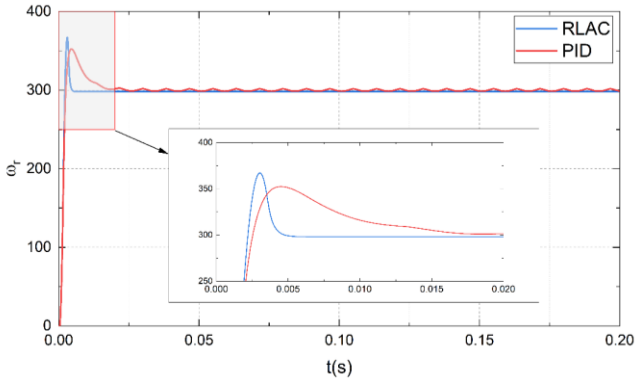

Figure 5. Speed comparison between RLAC and PID (speed step)

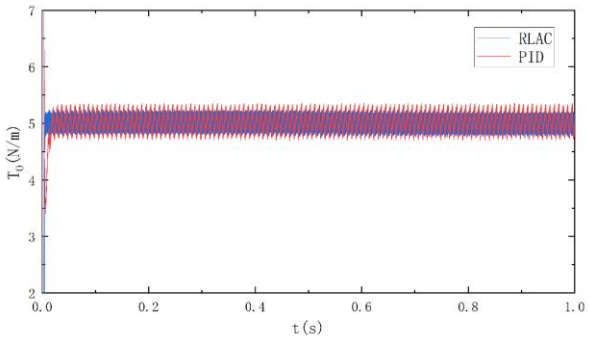

Figure 6. Torque comparison between RLAC and PID (speed step)

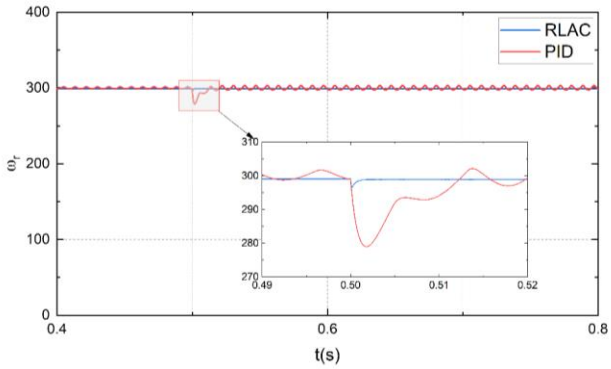

Figure 7. Speed comparison between RLAC and PID (torque step)

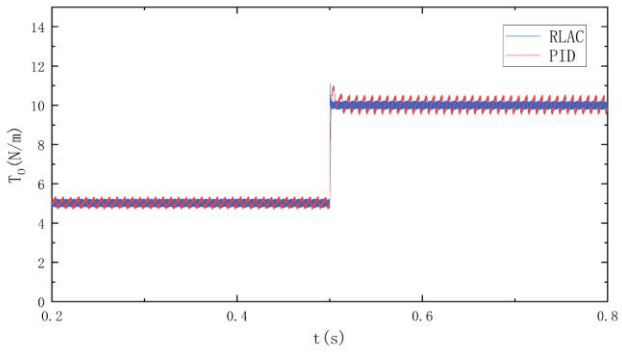

Figure 8. Torque comparison between RLAC and PID (torque step)

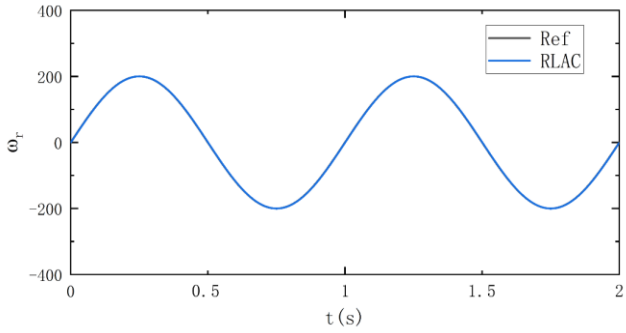

Figure 9. Performance of RLAC speed following

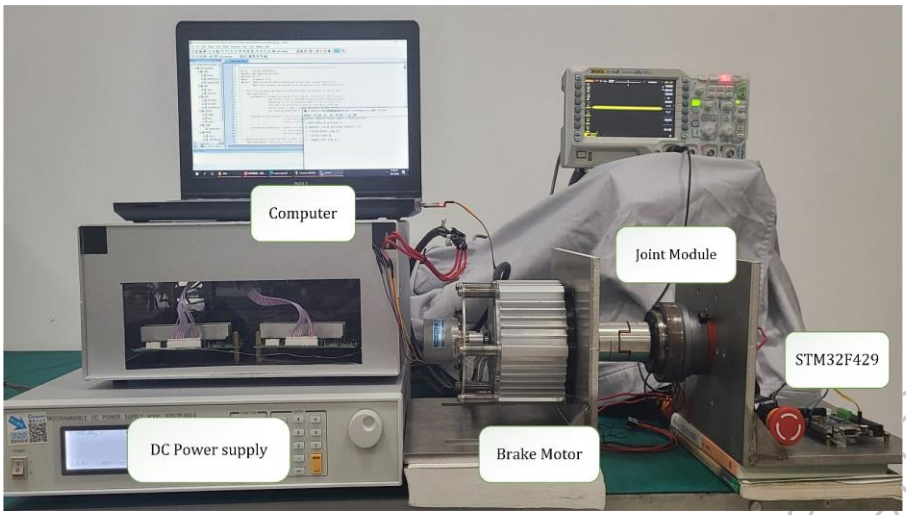

Figure 10. Experimental devices

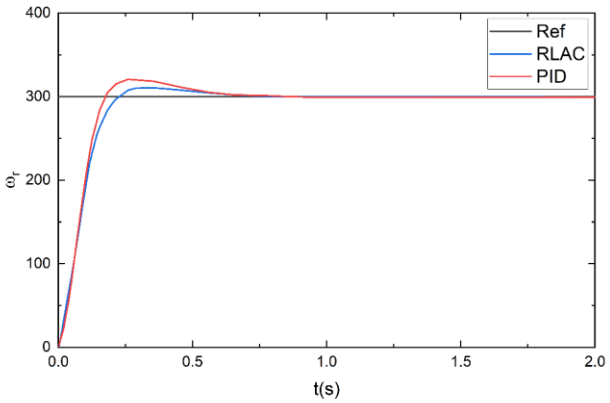

Figure 11. Experimental comparison of RLAC and PID (step response)

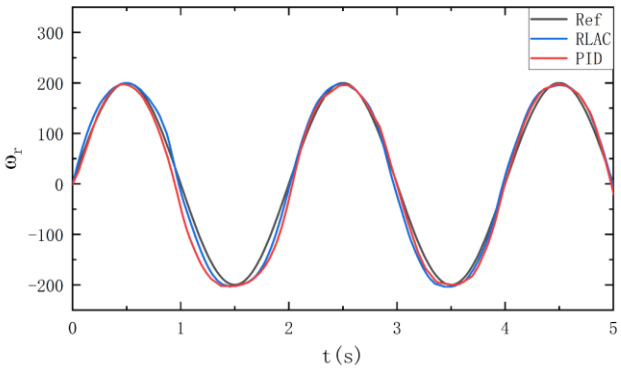

Figure 12. Experimental comparison of RLAC and PID (sinusoidal tracking)

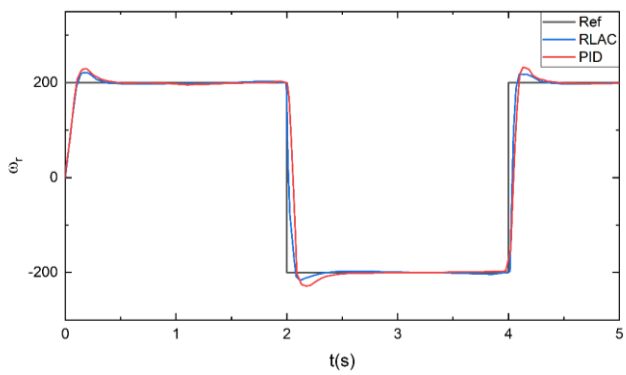

Figure 13. Experimental comparison of RLAC and PID (positive conversion)