2023 IEEE/ASME International Conference on
Advanced Intelligent Mechatronics (AIM)

Intelligent Servo Control Strategy for Robot Joints With Incremental Bayesian Fuzzy Broad Learning System

Guoyu Zuo, Jiyong Zhou, Daoxiong Gong, Gao Huang

***Abstract*—Intelligent servo control significantly reduces the need to adjust control parameters, and is therefore widely used in robot joint control. However, existing intelligent servo control strategies for robot joints have problems of computational redundancy, limited prediction accuracy, and insufficient generalization capability. To solve these problems, this paper proposes a servo control strategy for robot joints that is based on the incremental Bayesian fuzzy broad learning system (IBFBLS). Firstly, we construct an intelligent servo control strategy with broad learning system on the basis of fuzzy rules to achieve good self-learning and generalization abilities. Secondly, the learning parameters of the control strategy are optimized by Bayesian inference to achieve precise joint servo control. Finally, the convergence of the control strategy is enhanced by combining it with Lyapunov theory to constrain the learning parameters of the proposed control strategy. The feasibility and superiority of the proposed control strategy are verified by simulation to compare it with existing intelligent servo control methods. In addition, experiments are conducted using robot joint test bed. Both the simulation and experiments verify that the proposed servo control strategy outperforms other servo control methods with respect to tracking accuracy, stability, and convergence. The root-mean-square error in servo control of robot joints was 0.012%, which has been reduced by 55.56% compared to current state-of-art.**

***Index Terms*—Bayesian inference, fuzzy rules, incremental broad learning system, intelligent servo control, Lyapunov theory.**

I. Introduction

ROBOTS are widely used in various industries as intelligent devices, and their servo control have attracted substantial attention from researchers [1], [2]. The main aim of servo control is to improve the control strategy's precision, intelligence, and generalization capability [3], [4]. Servo control of robot joints is a key technology of robot motion and has a critical impact on the overall robot performance [5].

This paper was produced by the xxx.

Manuscript received XXXX. This work was supported in part by the National Natural Science Foundation of China under Grant 61873008 and 62103053,and National Key R & D Plan (2018YFB1307004)(Corresponding author: Gao Huang, Guoyu Zuo).

G. Y. Zuo, J. Y. Zhou and D. X. Gong are with the Faculty of Information Technology, Beijing University of Technology, Beijing 100124, China, and also with Beijing Key Laboratory of Computing Intelligence and Intelligent Systems, Beijing 100124, China. (e-mail: zuoguoyu@bjut.edu.cn, zhoujiyong2020@126.com, gongdx@bjut.edu.cn). G. Huang is with the Faculty of Information Technology, Beijing University of Technology, Beijing 100124, China, and also with Beijing Advanced Innovation Center for Intelligent Robots and Systems, Beijing Institute of Technology, Beijing 100081, China. (e-mail: huanggao@bjut.edu.cn).

Many researchers have designed high-performance strategies for such servo control [6], [7], such as adaptive PID, feedback linearization-based methods, and sliding-mode control (SMC) [8]–[10]. The increasing internal structural complexity of robot joints significantly increases the cost of accurately modeling mathematically. The above conventional strategies cannot be adapted to precise servo control of robot joints with unknown parameters. To solve this problem, Huang et al. proposed an integrated interval type-2 fuzzy logic approach to address the effects of modeling uncertainty and external disturbances on controller stability [11]. Li et al. proposed an adaptive fuzzy control scheme for a dual-arm robot. Their scheme includes an algorithm that uses an approximate Jacobian matrix to address uncertain kinematic control, and a decentralized fuzzy controller to accommodate motion uncertainty and unknown disturbances [12]. Fuzzy systems with good approximation capability can estimate the unknown dynamics of robot joints with any specific accuracy. Therefore, the fuzzy system-based control strategy can effectively solve the problems of motion uncertainty and unknown disturbances to improve the control accuracy of the strategy [13], [14]. However, these strategies lack self-learning capability and generalization ability, so they cannot satisfy the requirements of intelligent servo control of robot joints.

Neural networks with learning and generalization abilities show superior performance in approximately modeling of the dynamic of robot joints. They include many adaptive weight parameters to learn uncertainties and external disturbances, which improves the servo control accuracy of the robot joints [15], [16]. To further improve the servo control of robot joints, He et al. proposed a strategy based on an adaptive fuzzy neural network (FNN) to ensure control accuracy under unknown system dynamics and uncertainty disturbances [17]. However, the FNN-based servo control strategy requires a complex training process to determine the controller parameters, which consumes a large amount of computational resources. Moreover, it is necessary to retrain the controller when it does not satisfy the needs of the application [18]. In recent years, the broad learning system (BLS) has become widely used [19]–[22]. The BLS has a simple network structure and accurate approximation performance. Some researchers have attempted to apply it to servo control of robot joints [23]–[25]. Xu et al. proposed an intelligent servo control strategy based on BLS and derived the controller parameter constraints using Lyapunov theory to obtain with strong generalization performance and error convergence [26]. Fei et al. proposed

a fuzzy broad learning control strategy, consisting of a fuzzy broad controller and a robust controller, to improve approximation and learning capability [27]. However, this control strategy does not consider the effects of noise disturbances and unknown disturbances on servo control accuracy. In addition, it does not consider the important advantage of the BLS, namely incremental learning, and therefore the accuracy of this strategy is adversely affected.

To address these problems, we propose an intelligent servo control strategy for robot joints with strong generalization capability, low computational redundancy, and high control accuracy. The contributions of this paper are as follows:

1) A control strategy based on an incremental Bayesian fuzzy broad learning system (IBFBLS) is proposed for the first time and applied to robot joint servo control to improve generalization performance and control accuracy.
2) The proposed servo control strategy can handle noisy disturbances or unknown disturbances. In the proposed strategy, the posteriori probabilistic estimates of the output and output weights are calculated to efficiently perform incremental learning in a probabilistic manner, which improves the performance of the control strategy.
3) Lyapunov theory is combined with the IBFBLS to obtain constraints on the learning parameters of the proposed control strategy. This provides the control strategy with stable convergence capability.

II. Problem Statement and Preliminary Discussion

This section describes the principles of the robot servo control system and fuzzy broad learning system (FBLS).

A. Robot Joint Servo Control Strategy

An important part of joint servo control is joint position control, the control target is the joint can reach to the expected position. The desired position and desired velocity are known. The actual position and actual velocity are obtained using the end-of-joint sensors. In this paper, the desired position, desired velocity, actual position, and actual velocity are denoted by $d_{re}(t)$, $v_{re}(t)$, $d(t)$, and $v(t)$, respectively [28], [29]. The tracking error in the task space is expressed as:

$$\begin{cases} e_1(t) = d_{re}(t) - d(t) \\ e_2(t) = v_{re}(t) - v(t) \end{cases} \tag{1}$$

where $e_1(t)$ denotes the position tracking bias, $e_2(t)$ denotes the velocity tracking bias. The ultimate goal of the control system is to ensure that both position tracking bias $e_1(t)$ and velocity tracking bias $e_2(t)$ converge to 0. To enhance the approximation capability of the control strategy, this paper proposes an objective function to ensure that the velocity tracking bias $e_2(t)$ eventually converges to a value near 0.

$$\min_{v(t)} f = |v_{re}(t) - v(t)| \tag{2}$$

where f denotes the objective optimization function. Approximating of the desired velocity helps the position tracking bias $e_1(t)$ to converge to 0. Following this principle, the position tracking problem of servo control is regarded as a minimization problem.

B. Fuzzy Broad Learning System (FBLS)

FBLS is proposed in this paper, which combines the BLS with fuzzy rules. The key idea of FBLS is to replace the feature nodes of the BLS with fuzzy subsystems [28]. Suppose that the input data are denoted by $\mathbf{X}^{M\times N} = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x_N}]$, where, $\mathbf{x}_t = [x_{t1}, x_{t2}, \cdots, x_{tM}]$. M denotes the number of data in the vector $\mathbf{x}_t$. The fuzzy dataset can then be expressed as $z_{ti}^j = f_i^j(x_{t1}, x_{t2}, \cdots, x_{tM})$, where f_i^j denotes the coefficient of the jth fuzzy dataset, j denotes the jth fuzzy subsystem, and i denotes the ith fuzzy rule of the jth fuzzy subsystem. The jth fuzzy subsystem of FBLS is defined as:

$$z_{ti}^j = f_i^j(x_{t1}, x_{t2}, \cdots, x_{tM}) = \sum_{m=1}^{M} \upsilon_{im}^j x_{tm} \tag{3}$$

where υ_{im}^j denotes the coefficient of the mth data for $\mathbf{x}_t$.

Feature nodes generated by the Gaussian function are the most suitable for robot joint control [25]. Therefore, the Gaussian affiliation function of the proposed fuzzy dataset is defined in (4).

$$\mu_{im}^j = exp(-(\frac{x_{tm} - c_{im}^j}{\sigma_{im}^j})^{\ 2}) \tag{4}$$

where σ_{im}^j and c_{im}^j denote width and centroid of the Gaussian affiliation function, respectively.

The fire strength of the ith fuzzy rule of the jth fuzzy subsystem is defined as:

$$\tau_{ti}^j = \prod_{m=1}^{M} \mu_{im}^j (x_{tm}) \tag{5}$$

The weight of the ith fuzzy rule of the jth fuzzy subsystem is defined as:

$$\omega_{ti}^j = \frac{\tau_{ti}^j}{\sum_{i=1}^{I} \tau_{ti}^j} \tag{6}$$

where I denotes the number of fuzzy rules of the jth fuzzy subsystem and $i = 1, 2, \cdots, I$.

The output of the jth fuzzy system with input data $\mathbf{x}_t$ is defined as:

$$\mathbf{z}_t^j = \left[\omega_{t1}^j \upsilon_{t1}^j, \omega_{t2}^j \upsilon_{t2}^j, \cdots, \omega_{tI}^j \upsilon_{tI}^j\right] \tag{7}$$

The output of the jth fuzzy system with input data $\mathbf{X}^{M\times N}$ is defined as:

$$\mathbf{z}^j = \left[\mathbf{z}_1^j, \mathbf{z}_2^j, \cdots, \mathbf{z}_N^j\right] \tag{8}$$

The output of the l fuzzy subsystem is then defined as:

$$\mathbf{Z}^l = \left[\mathbf{z}^1, \mathbf{z}^2, \cdots, \mathbf{z}^l\right] \tag{9}$$

The enhancement network is based on the enhancement nodes generated by the feature nodes of fuzzy logic. The principle is expressed as follows:

$$\mathbf{h} = \psi\left(\alpha_{h_q} \cdot \mathbf{Z}^l + \beta_{h_q}\right) \tag{10}$$

where $\psi = 1/(1 + exp(\cdot)) - 1$ denotes the activation function of the enhancement nodes. α_{h_q} denotes the weight vector of the enhancement nodes. β_{h_q} denotes the offset parameters of the enhancement nodes. All these parameters are randomly

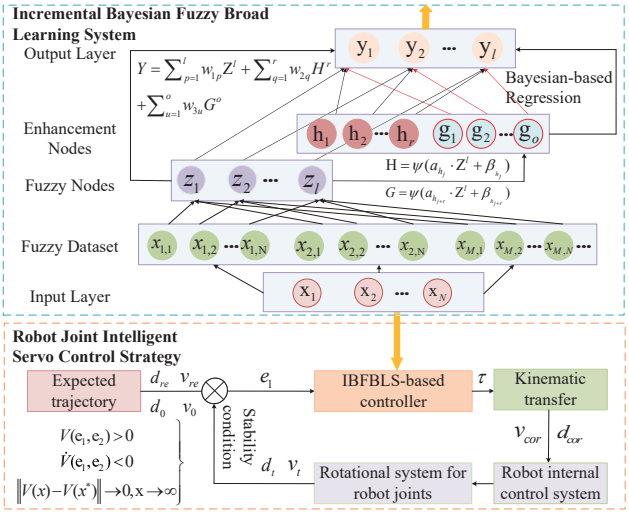

Fig. 1. Intelligent servo control principle based on IBFBLS. The top part of the figure shows the network model of IBFBLS. The model contains inputs, fuzzy sets, fuzzy nodes, augmented nodes, and outputs. The lower part of the figure shows the robot joint's control strategy.

generated. The enhancement nodes generated by the FBLS are then represented as:

$$\mathbf{H}^r = [\mathbf{h}_1, \mathbf{h}_2, \cdots, \mathbf{h}_r] \tag{11}$$

The FBLS control system generates nodes as shown in (12).

$$\mathbf{F}^{l+r} = [\mathbf{Z}^l, \mathbf{H}^r] \tag{12}$$

Finally, the FBLS can be described as

$$\mathbf{Y}(\mathbf{X}) = \sum_{k=1}^{l+r} w_k \mathbf{F}^{l+r} \tag{13}$$

where w_k denotes the output matrix weights, which is generated by the network training.

III. IBFBLS Servo Control Strategy

This section explains the servo control strategy based on the IBFBLS, which achieves learning ability and generalization of the servo control of robot joints, and it's principle is shown in Figure 1. It comprises three main modules: the control strategy based on the IFBLS, ehe Bayesian-based Regression module, and the stability analysis based on Lyapunov theory.

A. Servo Control Strategy Based on IFBLS

When designing a control strategy, it is necessary to construct a reasonable transfer function to obtain corrected velocity command $v_{cor}(t)$ and corrected position command $d_{cor}(t+1)$ [26]. This principle can be expressed as:

$$d_{cor}(t+1) = d(t) + \int_t^{t+\Delta t} v_{cor}(t)dt \tag{14}$$

where $d(t)$ denotes the actual position at moment t, Δt denotes the time interval, which is the time difference between t and $t+1$.

It is worth stating that velocity tracking error e_2 used in this paper is not the difference between actual velocity $v_{re}(t)$ and desired velocity $v(t)$, but is obtained by calculating the position tracking error e_1. The design of the advanced control strategy can be described abstractly as follows.

$$\tau = e_2 = f(e_1) \tag{15}$$

where the position tracking error e_1 is the input of the control strategy, and τ denotes the control strategy, which in servo control denotes the velocity correction command.

The IFBLS-based position servo control strategy can then be expressed as:

$$\begin{aligned}\tau = &\sum_{p=1}^{l} w_p \mathcal{F}(e_1) + \sum_{q=l}^{l+r} w_q \left[\psi\left(\alpha_{h_q} \cdot \mathbf{Z}^l + \beta_{h_q}\right) - \lambda_q \varepsilon_q\right] \\ &+ \sum_{u=l+r}^{l+r+o} w_u \left[\psi\left(\alpha_{h_u} \cdot \mathbf{Z}^l + \beta_{h_u}\right) - \lambda_u \varepsilon_u\right]\end{aligned} \tag{16}$$

where w_p, w_q, and w_u denote the output weights of fuzzy nodes, enhancement nodes, and incremental enhancement nodes, respectively. λ_q and λ_u denote the compression factors, ε_q and ε_u denote the residual errors. Introducting the compression factor and residual error effectively improves the convergence speed of the control strategy [22]. $\mathcal{F}(\cdot)$ denotes the fuzzy rule.

To design a controller that has excellent approximation capability, the model training process is treated as an optimization problem.

$$\begin{aligned}\min_{v(t)} \textstyle\sum_{m=1}^{M} || &\sum_{p=1}^{l} w_p \mathbf{Z}^l + \sum_{q=l}^{l+r} w_q (\mathbf{H}^r - \lambda_q \varepsilon_q) \\ &+ \sum_{u=l+r}^{l+r+o} w_u (G^o - \lambda_u \varepsilon_u) - \tau_m^{de} ||\end{aligned} \tag{17}$$

where $G^o = \psi\left(\alpha_{h_u} \cdot \mathbf{Z}^l + \beta_{h_u}\right)$ represents the new enhancement nodes, τ_m^{de} denotes the desired control output.

B. Bayesian-based Regression Algorithm

The regularization parameters of ridge regression are usually predetermined manually, which results in low predictive accuracy. Bayesian inference provides an estimate of the posterior probability of the output and the output weight. Therefore, Bayesian regression algorithms are more suitable for position tracking of robot joints than the ridge regression [30]. Inspired by this observation, Bayesian inference is combined with the BLS to design the regression algorithm of the IBFBLS.

The output of the IBFBLS can be expressed as

$$\Lambda = \mathbf{F}\mathbf{w} + \delta \tag{18}$$

where δ denotes a random noise matrix that represents the interference of the external environment. $\mathbf{F}$ denotes the nodes of the proposed control policy, $\mathbf{w}$ denotes the weight of the nodes. To facilitate calculation, this noise is defined as Gaussian noise with mean of 0 and variance of φ^{-1}.

The likelihood function of output of the control strategy can be expressed as:

$$p(\mathbf{w}|\rho) = \mathcal{N}(\mathbf{w}|0, \rho^{-1}\mathbf{E}) \tag{19}$$

According to Bayesian inference, the posterior distribution of weight $\mathbf{w}$ is:

$$p(\mathbf{w}|\mathbf{t},\mathbf{F},\varphi,\rho)=\frac{p(\Lambda|\mathbf{F},\mathbf{w})p(\mathbf{w})}{p(\Lambda|\mathbf{F})}=\mathcal{N}(\mathbf{w}|\eta,\vartheta) \tag{20}$$

where

$$\vartheta=\left(\rho\mathbf{E}+\varphi\mathbf{F}^{\mathbf{T}}\mathbf{F}\right)^{-1} \tag{21}$$

and

$$\eta=\varphi\vartheta\mathbf{F}^{\mathbf{T}}\mathbf{\Lambda} \tag{22}$$

According to (21) and (22), the marginal likelihood is obtained by integrating over $\mathbf{w}$.

$$\begin{aligned}p\left(\mathbf{\Lambda}|\mathbf{F},\varphi,\rho\right)=&\left(\frac{\rho}{2\pi}\right)^{\frac{N}{2}}\left(\frac{\varphi}{2\pi}\right)^{\frac{M}{2}}exp\left\{-\Omega\left(\eta\right)\right\}\\&\cdot\left(2\pi\right)^{\frac{M}{2}}\left|\vartheta^{-1}\right|^{-\frac{1}{2}}\end{aligned} \tag{23}$$

where $\Omega(\eta)=\frac{\varphi}{2}\left\|\Lambda-\mathbf{F}\eta\right\|+\frac{\rho}{2}$. The parameters ρ and φ are updated iteratively by Eq.(24) and Eq.(25).

$$\varphi=\frac{\sum_{a=1}^{M}\left(1-\varphi\eta_a\right)}{\eta^{\mathbf{T}}\eta} \tag{24}$$

$$\rho=\frac{N-\sum_{a=1}^{M}\left(1-\varphi\eta_a\right)}{\left\|\Lambda-\mathbf{F}\eta\right\|^2} \tag{25}$$

To ensure that the control strategy has a high predictive performance, the incremental Bayesian regression is designed in this paper. The output of the incremental control strategy is $\mathbf{F}'=[\mathbf{F}\mid\mathbf{G}]$. Bringing $\mathbf{F}'$ into (21) produces (26).

$$\hat{\vartheta}=\begin{pmatrix}\vartheta+\mathbf{QBQ}^{\mathbf{T}} & -\mathbf{QB}\\ -\mathbf{BQ}^{\mathbf{T}} & \mathbf{B}\end{pmatrix} \tag{26}$$

where, $\mathbf{Q}=\rho\vartheta\mathbf{F}^{\mathbf{T}}\mathbf{G}$ and $\mathbf{B}=\left(\varphi\mathbf{E}+\rho\mathbf{G}^{\mathbf{T}}\mathbf{G}-\rho^2\mathbf{G}^{\mathbf{T}}\mathbf{F}\vartheta\mathbf{F}\mathbf{G}\right)^{-1}$. Equation (26) is then brought into (22) to obtain

$$\hat{\eta}=\begin{pmatrix}\eta+\rho^2\vartheta\mathbf{F}^{\mathbf{T}}\mathbf{GBG}^{\mathbf{T}}\left(\mathbf{F}\eta-\mathbf{\Lambda}\right)\\ -\rho\mathbf{BG}^{\mathbf{T}}\left(\mathbf{F}\eta-\mathbf{\Lambda}\right)\end{pmatrix} \tag{27}$$

The output weights are updated through the above equation. The hyperparameters are also updated to perform the next iteration and update the output weights.

$$\hat{\varphi}=\frac{\hat{\xi}}{\hat{\eta}^{\mathbf{T}}\hat{\eta}} \tag{28}$$

$$\hat{\rho}=\frac{N-\hat{\xi}}{\left\|\Lambda-\mathbf{F}'\hat{\eta}\right\|^2} \tag{29}$$

where $\hat{\xi}=\mathbf{B}-\hat{\varphi}\left(trace\left(\vartheta+\mathbf{QBQ}^{\mathbf{T}}+\mathbf{B}\right)\right)$.

C. System Stability Analysis

The work in this paper is based on the assumption that the low-level control system is in an ideal state. It is assumed that the low-level control system is stable and reliable and does not affect the high-level system. Now Lyapunov theory is introduced to analyze and prove the stability of the proposed control strategy. Using the stability analysis principle of Lyapunov theory, the control strategy is designed as a global asymptotically stable system [31]. Therefore, the following constraints need to be satisfied:

$$\begin{cases}V\left(e_1,e_2\right)>0\\ \dot{V}\left(e_1,e_2\right)<0\\ \left\|V(x)-V(x^*)\right\|\to 0,x\longrightarrow\infty\end{cases} \tag{30}$$

To satisfy the requirements of (30) for system stability, the Lyapunov stability function is defined as

$$V=\frac{1}{2}{e_1}^{\mathbf{T}}e_1+\frac{1}{2}{e_2}^{\mathbf{T}}\mathbf{K}e_2 \tag{31}$$

The analysis shows that the first and third inequalities of (30) are satisfied. It is now necessary to determine whether the second inequality is satisfied.

$$\dot{V}={e_1}^{\mathbf{T}}\dot{e}_1+{e_2}^{\mathbf{T}}\mathbf{K}\dot{e}_2 \tag{32}$$

Equation (32) contains four variables e_1, $\dot{e}_1$, e_2 and $\dot{e}_2$. In the proposed control strategy, e_2 is derived from e_1, that is, $e_2=\dot{e}_1$. After analysis, e_1, $\dot{e}_1$, and e_2 are already defiend. Therefore, only $\dot{e}_2$ needs to be calculated.]

$$\begin{aligned}\dot{e}_2=&\sum_{p=1}^{l}\frac{d\left(\mathcal{F}\left(e_1\right)\right)}{de}w_p\dot{e}_1+\sum_{q=r}^{l+r}\frac{dw_q\psi(\Delta)}{d\Delta}w_q\alpha_{h_q}\\&\cdot\frac{d\left(\mathcal{F}\left(e_1\right)\right)}{de_1}+\sum_{u=l+r}^{l+r+o}\frac{dw_u\psi(\Xi)}{d\Xi}w_u\alpha_{h_u}\frac{d\left(\mathcal{F}\left(e_1\right)\right)}{de_1}\end{aligned} \tag{33}$$

where, $\triangle=\alpha_{h_q}\cdot\mathbf{Z}^l+\beta_{h_q}$ and $\Xi=\alpha_{h_u}\cdot\mathbf{Z}^l+\beta_{h_u}$. The derivative of the activation function is greater than 0.

The $\dot{V}$ can be written as:

$$\begin{aligned}\dot{V}=&e_1^{\mathbf{T}}[\sum_{p=1}^{l}w_p\mathcal{F}\left(e_1\right)+\sum_{q=l}^{l+r}w_q\left[\psi(\Delta)-\lambda_q\varepsilon_q\right]+\\&\sum_{u=l+r}^{l+r+o}w_u\left[\psi(\Xi)-\lambda_u\varepsilon_u\right]]+{e_2}^{\mathbf{T}}K[\sum_{p=1}^{l}\frac{d\left(\mathcal{F}\left(e_1\right)\right)}{de}w_p\dot{e}_1\\&+\sum_{q=l}^{l+r}\frac{d\psi(\Delta)}{d\Delta}w_q\alpha_{h_q}\frac{d\left(\mathcal{F}\left(e_1\right)\right)}{de_1}\\&+\sum_{u=l+r}^{l+r+o}\frac{d\psi(\Xi)}{d\Xi}w_u\alpha_{h_u}\frac{d\left(\mathcal{F}\left(e_1\right)\right)}{de_1}]\end{aligned} \tag{34}$$

For convenience, the first term of (34) is defined as $\mathbb{R}_\mathbf{1}$ and the second term as $\mathbb{R}_\mathbf{2}$ in this paper. The $\mathbb{R}_\mathbf{1}$ term can be simplified as follows.

$$\begin{aligned}\mathbb{R}_1 =& \underbrace{e_1^{\mathbf{T}}\sum_{p=1}^{l} w_p\mathcal{F}\left(e_1\right)}_{I} + \underbrace{e_1^{\mathbf{T}}\sum_{q=l}^{l+r} w_q\dot{\psi}(s_m)\alpha_{h_q}\cdot\mathbf{Z}^l}_{II} \\ &+ \underbrace{e_1^{\mathbf{T}}\sum_{u+l}^{u+l+o} w_u\dot{\psi}(s_m)\alpha_{h_q}\cdot\mathbf{Z}^l}_{III} + \underbrace{e_1^{\mathbf{T}}\sum_{q=l}^{l+r} w_q\beta_{h_q}}_{IV} \\ &+ \underbrace{e_1^{\mathbf{T}}\sum_{u=l+r}^{l+r+o} w_u\beta_{h_u}}_{V} + \underbrace{e_1^{\mathbf{T}}\sum_{q=1}^{r} w_q\lambda_q\varepsilon_q}_{VI} \\ &+ \underbrace{e_1^{\mathbf{T}}\sum_{u=1}^{o} w_u\lambda_u\varepsilon_u}_{VII}\end{aligned} \tag{35}$$

where $\psi(0)$ represents the original state. $\dot{\psi}(s_m) = \frac{\psi(\triangle)-\psi(0)}{(\triangle-0)}$ or $\dot{\psi}(s_m) = \frac{\psi(0)-\psi(\triangle)}{(0-\triangle)}$. $s_i \in (\triangle, 0)$ or $s_i \in (0, \triangle)$). In addition, $\psi(0) = 0$, $\dot{\psi}(0) = 0$.

Similarly, $\mathbb{R}_2$ can be simplified as follows:

$$\begin{aligned}\mathbb{R}_{\mathbf{2}} =& \underbrace{{e_2}^{\mathbf{T}} K\sum_{p=1}^{l}\frac{d\left(\mathcal{F}\left(e_1\right)\right)}{de}w_p\dot{e}_1}_{VIII} \\ &+ \underbrace{{e_2}^{\mathbf{T}}\sum_{q=l}^{l+r}\frac{d\psi(\Delta)}{d\Delta}w_q\alpha_{h_q}\frac{d\left(\mathcal{F}\left(e_1\right)\right)}{de_1}}_{IX} \\ &+ \underbrace{{e_2}^{\mathbf{T}}\sum_{u=l+r}^{l+r+o}\frac{d\psi(\Xi)}{d\Xi}w_q\alpha_{h_u}\frac{d\left(\mathcal{F}\left(e_1\right)\right)}{de_1}}_{X}\end{aligned} \tag{36}$$

To satisfy the constraints of (30), the sum of the terms from I to X needs to be negative at all times. The tracking bias between the terms is difficult to determine. Therefore, the general case is to limit the parameters to ensure that each term is negative. Careful analysis reveals that the derivatives and slopes are positive; that is, the parameters $\frac{d\psi(\Delta_i)}{d\Delta_i}$, $\frac{d(\mathbf{F}(e_1))}{de_1}$, $\mathbf{F}\left(e_1\right)$, $e_1^{\mathbf{T}}$, $\dot{e}_1$ and ${e_2}^{\mathbf{T}}$ are positive. The following conditions need to be satisfied.

$$\begin{cases} w_p < 0 \\ w_q\alpha_{h_q} < 0, w_u\alpha_{h_u} < 0 \\ w_q\beta_{h_q} < 0, w_u\beta_{h_u} < 0 \\ w_q\lambda_q\varepsilon_q < 0, w_3\lambda_u\varepsilon_u < 0 \end{cases} \tag{37}$$

The parameters of the control strategy need to satisfy the constraints of (37) to satisfy the stability requirements.

IV. Simulation

Signal tracking and position tracking simulations were conducted to verify the performance of the IBFBLS-based control strategy. In this section, the simulations are designed, and the results are compared with the results of existing representative learning-based control strategies.

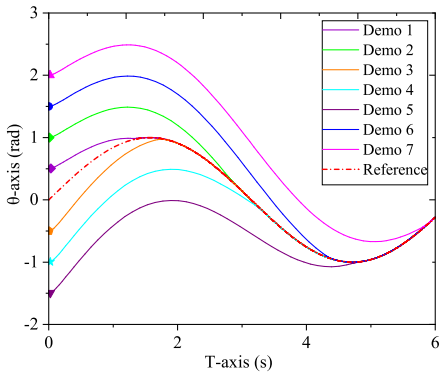

Fig. 2. Reference trajectory and demo trajectories. Seven demo trajectories was tracked performed from different starting points were selected.

A. Signal Tracking Simulation of Robot End-of-Joint Output

To obtain the demo data required for training the control strategy, this paper defined the sinusoidal trajectory with an amplitude of 1 and a frequency of 1 as the reference trajectory. Tracking was performed from different starting points using sliding-mode control in MATLAB, and seven well-converged demo trajectories were selected. The computation time of these tracks was 6s, and 2000 data points were saved. The reference trajectory and the several demo trajectories were shown in Fig. 2. The controller input was the position error of the demo data. The output of the controller was the correction velocity. The position error of the demo data was obtained by converging the tracking trajectory with the reference trajectory, and the velocity of the tracking trajectory was the target velocity.

The acquired demo data were divided into a training set and a test set. The training set included four of the convergent tracks, and the remaining three convergent tracks were included in the test set. The experiment was conducted on a server running Windows 10. The programming language was Python 3.7. The computing platform used for compilation and testing was PyTorch with CUDA v1.0. The hardware configuration included an Intel Core i9-10900X CPU at 3.70 GHz, an NVIDIA RTX 3090 Ti GPU, and 64 GB of memory. The proposed IBFBLS algorithm was compared with the SMC, BLS, FBLS, and IBLS algorithms. Five subsets based on data distribution are negligible bias (NB), low bias (LB), moderate bias (MB), high bias (HB) and very high bias (VB). The fuzzy rule can be expressed as: if NB then NB; if LB then LB; if MB then MB; if HB then HB; if VB then VB. The initial number of fuzzy nodes of FBLS and IBFBLS was 10. The initial number of enhancement nodes for the IBLS and IBFBLS was 10, the number for training epochs was 10, the step size was 100, and the expected accuracy was 0.90.

The results of the signal tracking simulation are visualized in Fig. 3. The BLS and FBLS maintained good tracking performance in the initial stage. In the middle and end stages, that is, $t = [3, 4]$ and $t = [5, 6]$, the correction velocity oscillated substantially, so the simulation results showed substantial deviation from the expected velocity and expected position.

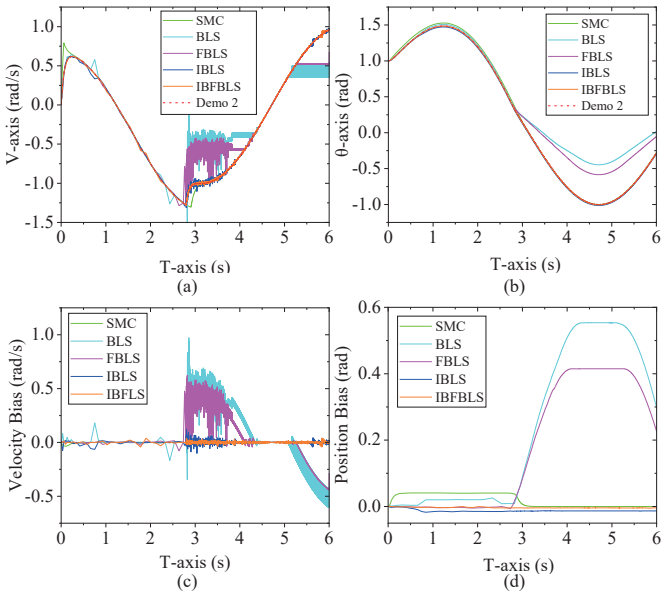

Fig. 3. Position tracking simulation results of demo trajectory 2 for different control methods: (a) X-axis velocity bias; (b) Y-axis velocity bias; (c) X-axis position bias; (d) Y-axis position bias.

The reason for this phenomenon may be that BLS and FBLS have no constraints. Both IBLS and IBFBLS could always converge to the reference trajectory. It is worth noting that IBLS is also an algorithm without constraints. However, it achieved better performance than non-incremental BLS. This aptly illustrates that the IBLS approximation performance has some advantages in trajectory tracking. However, the velocity of IBLS correction oscillated to some extent. This oscillation can be observed for the periods $t = [3, 4]$ and $t = [5.5, 6]$ in Fig. 3(a). The IBLS position always deviated from the reference position, as shown in Fig. 3(d). In contrast, the proposed servo control strategy performed almost no oscillation. The generated correction velocity was consistent with the reference velocity, and the generated position was consistent with the reference position. Overall, the signal tracking simulation results verify that the proposed control strategy has the best tracking performance. It effectively improves the control ability of the output signal, which in turn improves the precision of robot joint positioning.

B. Position Tracking Simulation of Robot Joint End-Effectors

This paper defined three different trajectories: l_1: $y = 0$, l_2: $y = x - 1$, and l_3: $(x-1)^2 + y^2 = 1$. As shown in the signal tracking simulations, the three trajectories were tracked from different starting points using sliding-mode control in MATLAB. The trajectories were run for 4s, and 2000 data points were saved. The most convergent tracking trajectory was selected for each of the three reference trajectories. The reference trajectories and convergent tracking trajectories are shown in Fig. 4.

The results are visualized in Fig. 5. For the trajectories shown in Fig. 5(a) and (b), all methods maintained good tracking performance. This maybe because the demo trajectories l_1 and l_2 are linear and the tracking is not difficult.

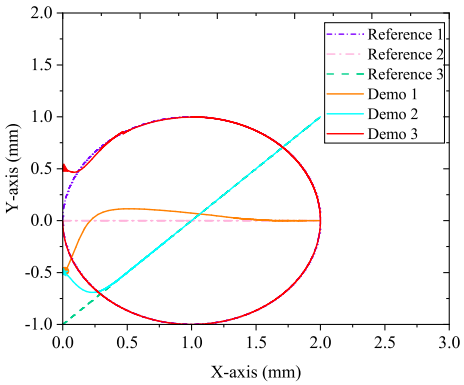

Fig. 4. Reference trajectory and demo trajectory. The dashed line represents the desired trajectory and the realized represents the demo trajectory

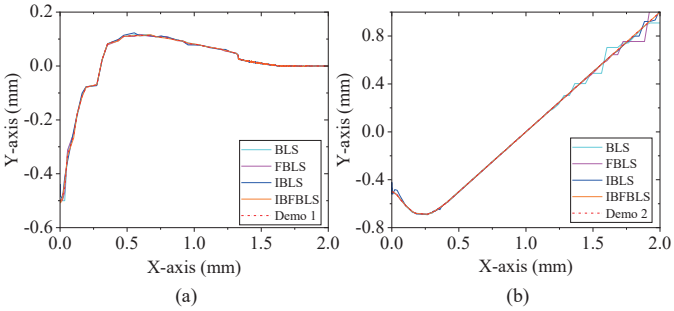

Fig. 5. Simulated position tracking simulation results for different control methods: (a) Demo trajectory 1; (b) Demo trajectory 2.

However, there was some degree of oscillation in the tracking performed by BLS and FBLS in Fig. 5(b). To show the tracking performance of each control strategy in more detail, the demo trajectory l_3 was taken as an example, and the velocity bias and position bias of the tracking trajectory and the demo trajectory were analyzed. The velocity biases of BLS and FBLS oscillated substantially, and their position biases did not converge to 0, as shown in Fig.6. The velocity biases of IBLS and IBFBLS oscillated only slightly, and their position biases could always converge to 0. However, IBLS was inferior to IBFBLS with respect to the convergence of the position bias. As shown in Fig. 6(d), the position bias of IBLS eventually failed to converge to 0, whereas the position bias of IBFBLS eventually converged to a value near 0. Accurate position tracking of the end effector is conducive to promoting intelligent servo control of robot joints, which is a significant advantage for autonomous robot control.

In this paper, we introduce the root-mean-square error (RMSE) [32] for quantitative evaluation of signal tracking and position tracking, as shown in Table I. The RMSE values of BLS and FBLS are significantly higher than those of IBLS and IBFBLS, which also verifies the advantages of the incremental control strategy at the quantitative comparison level. The data show that the RMSE value of IBFBLS is the smallest. The RMSE in signal tracking was 0.012%, which has been reduced by 55.56% compared to IFBLS. The RMSE

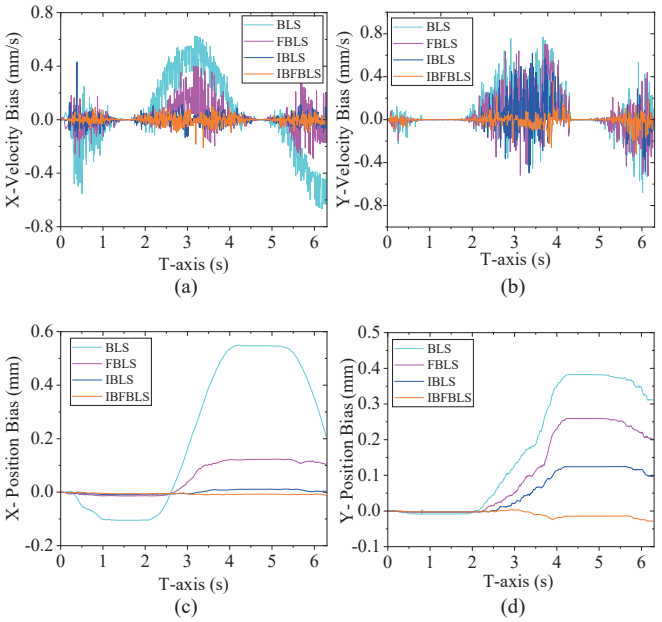

Fig. 6. Simulated velocity bias and position bias of demo trajectory 3 for different control models: (a) X-axis velocity bias; (b) Y-axis velocity bias; (c) X-axis position bias; (d) Y-axis position bias.

TABLE I
THE RMSE OF SIGNAL TRACKING AND POSITION TRACKING.(UNIT: %)

Control strategy	BLS	FBLS	IFBLS	IBFBLS
Signal tracking	0.926	0.722	0.027	0.012
Position tracking	1.278	0.559	0.246	0.045

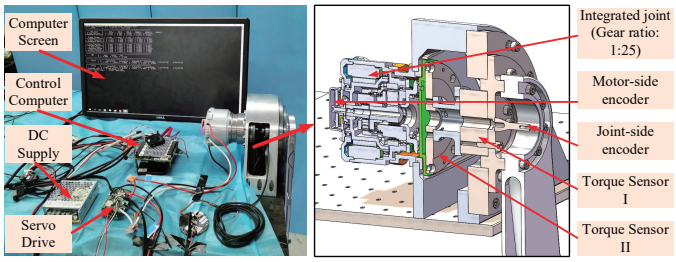

Fig. 7. Prototype of the rotary joint testbed. The left picture shows the hardware of the system, and the right picture shows the internal structure and sensor configuration of the joint platform.

in position tracking was 0.045%, which has been reduced by 81.7% compared to IFBLS.

V. EXPERIMENT AND DISCUSSION

The practicality and effectiveness of the proposed algorithm were verified with a rotary joint prototype, and the structure was shown in Fig. 7. The testbed could collect the torque, position, current, and other information synchronously in real time to simulate the robot's working environment and accurately tested the joint force control.

In this experiment, the controller trained as illustrated in Fig. 2 was used to control the rotating joint testbed. The high-level control commands were computed with the proposed method,

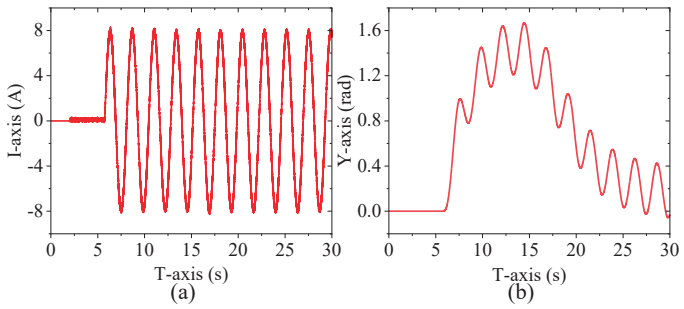

Fig. 8. Input and output of the rotary joint test platform: (a) input of the test platform; (b) output of the test platform.

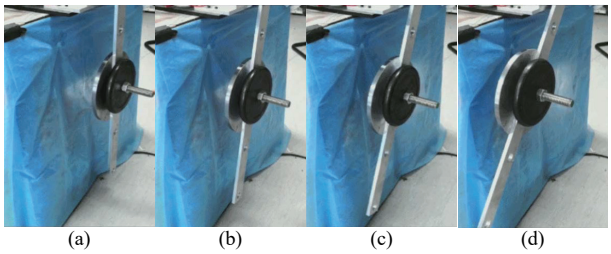

Fig. 9. Snapshots of control process of the robot joint test platform. Four different moments are shown as follows: (a) indicates the initial position; (b), (c), and (d) indicate the state at different positions.

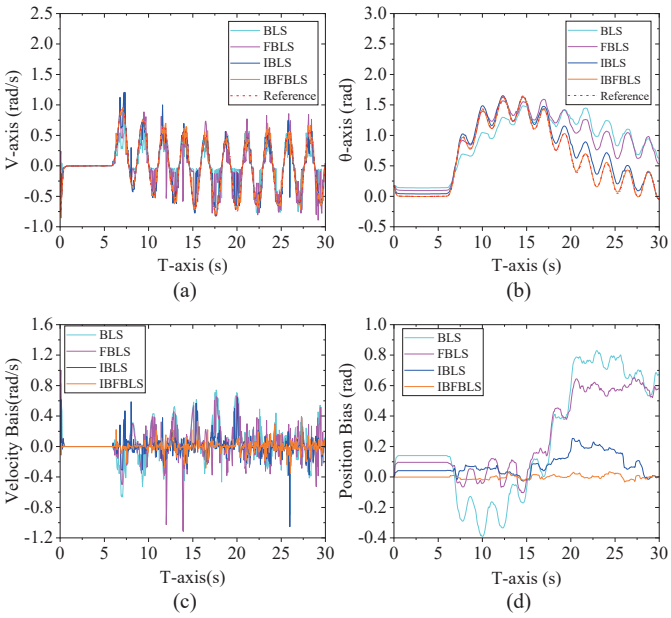

Fig. 10. Experimental results of the rotary joint testbed: (a) corrected velocity; (b) tracking trajectory; (c) velocity bias; (d) position bias.

whereas the low-level execution system was provided by the system of the testbed. An electrical signal was provided as an input to control the testbed, with a running time of 30s. The inputs and outputs of the testbed are shown in Fig. 8. The initial position was defined as $\theta = 10$, and the trained controller was used to perform the position tracking control task, shown in Fig. 9.

The experimental results are shown in Fig. 10 to measure

the effectiveness of the model more objectively. These results show that BLS and FBLS oscillated substantially during velocity tracking. This led to the deviation of the desired position from the actual position, as shown in Fig. 10(d). The tracking performance of IBLS controller was better than that of BLS and FBLS controllers. However, IBLS was inferior to IBFBLS with respect to the convergence of position bias. The performance of IBFBLS controller was generally the best. However, there was still some bias in the tracking: there was a small oscillation in the velocity bias, shown in Fig. 10(c), and the position bias finally failed to converge to 0, as shown in Fig. 10(d). The reason for this situation may be that physical experiments have more uncertainties than simulations. In particular, some degree of interference may be present in the low-level execution system of the joint.

VI. Conclusion

This paper proposed an intelligent servo control strategy for robot joints based on IBFBLS. This was combined with Lyapunov theory to design reasonable constraints for the control strategy. Simulations and experiments verified the proposed control strategy, which is superior to existing control strategies in most respects. However, the proposed strategy is based on the assumption that the low-level execution system is in the ideal state. In reality, the low-level execution system may have a substantial impact on the output. In future work, the internal structural parameters of the robot joint will be further explored and considered in our proposed end-to-end intelligent control system. Meanwhile, the algorithm performance will be further optimized in combination with constraint mechanism to mitigate oscillation and improve the joint control accuracy.

References

[1] G. G. Yang, C. Z. Chen, W. He, R. X. Cui, and Z. J. Li, "Robot Learning System Based on Adaptive Neural Control and Dynamic Movement Primitives," *IEEE Trans. Neural Netw. Learn Syst.*, vol. 30, no. 9, pp. 777-787, Mar. 2019.

[2] C. Yang, Y. Jiang, Z. Li, H. Wei, and C. Y. Su, "Neural control of bimanual robots with guaranteed global stability and motion precision," *IEEE Trans. Ind. Informat.*, vol. 13, no. 3, pp. 1162–1171, Jun. 2017.

[3] M. Le, P. Bazzoli, M. P. Sammons, G. R. Landers, and A. D. Douglas, "Modeling and Calibration of High-order Joint-dependent Kinematic Errors for Industrial Robots," *Robot. Comput. Integr. Manuf.*, no. 50, pp. 153-167, Apr. 2018.

[4] S. Xu, Y. Ou, J. Duan, X. Wu, W. Feng, and M. Liu, "Robot trajectory tracking control using learning from demonstration method," *Neurocomputing*, vol. 338, pp. 249–261, Apr. 2019.

[5] J. Y. Lu, B. Li, W. Q. Ge, C. Tan, B. B. Sun, "Analysis and experimental study on servo dynamic stiffness of electromagnetic linear actuator," *Mech. Syst. Signal Process.*, vol. 169, 18587, Apr. 2022.

[6] D. Y. Huang, C. G. Yang, Y. P. Pan, and L. Cheng, "Composite Learning Enhanced Neural Control for Robot Manipulator With Output Error Constraints," *IEEE Trans. Industr. Inform.*, vol. 17, no. 1, pp: 209-218, Jan. 2021.

[7] W. He, A. O. David, Z. Yin, and C. Sun, "Neural network control of a robotic manipulator with input deadzone and output constraint," *IEEE Trans. Syst. Man Cybern. Syst.*, vol. 46, no. 6, pp. 759–770, Jun. 2016.

[8] C. Zhao, L. Guo, "Control of Nonlinear Uncertain Systems by Extended PID," *IEEE Trans. Autom. Control.*, vol. 66, no. 8, pp. 3840-3847, Aug. 2021.

[9] M. B. Radac, R. E. Precup, E. M. Petriu, and S. Preitl, "Application of IFT and SPSA to servo system control," *IEEE Trans. Neural Netw.*, vol. 22, no. 12, pp. 2363–2375, Dec. 2011.

[10] D. A. Haghighi and S. Mobayen, "Design of an adaptive super-twisting decoupled terminal sliding mode control scheme for a class of fourth-order systems," *ISA Trans.*, vol. 75, pp. 216–225, Apr. 2018.

[11] J. Huang, M. Ri, D. R. Wu, and S. Ri, "Interval Type-2 Fuzzy Logic Modeling and Control of a Mobile Two-Wheeled Inverted Pendulum," *IEEE Trans. Fuzzy Syst.*, vol.26, no.4, pp. 2030-2038, Aug. 2018.

[12] C. Yang, Y. Jiang, J. Na, Z. Li, L. Cheng, and C. Y. Su, "Finite-time convergence adaptive fuzzy control for dual-arm robot with unknown kinematics and dynamics," *IEEE Trans. Fuzzy Syst.*, vol. 27, no. 3, pp. 574–588, Mar. 2019.

[13] W. Sun, S. F. Su, J. W. Xia, and V. T. Nguyen, "Adaptive fuzzy tracking control of flexible-joint robots with full-state constraints," *IEEE Trans. Syst. Man, Cybern. Syst.*, vol. 49, no. 11, pp. 2201–2209, Nov. 2019.

[14] H. Wang, W. Liu, J. Qiu, and P. X. Liu, "Adaptive fuzzy decentralized control for a class of strong interconnected nonlinear systems with unmodeled dynamics," *IEEE Trans. Fuzzy Syst.*, vol. 26, no. 2, pp. 836–846, Apr. 2018.

[15] A. Abdulla, M. Halwani, D. Swart, R. Muthusamy, F. Almaskari, and Y. Zweiri, "Neuromorphic Vision Based Control for the Precise Positioning of Robotic Drilling Systems," *Robot Comput. Integr. Manuf.*, vol. 79, 102419, Feb. 2023.

[16] D. Lei, S. S. Sun, M. D. Christie, W. X. Li, D. H. Ning, H. P. Du, S. W. Zhang, and W. H. Li, "Innovative Variable Stiffness and Variable Damping Magnetorheological Actuation System for Robotic Arm Positioning," *J. Intell. Mater. Syst. Struct.*, vol. 34, no. 2, pp.123-137, May. 2023.

[17] W. He and Y. Dong, "Adaptive fuzzy neural network control for a constrained robot using impedance learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 4, pp. 1174–1186, Apr. 2018.

[18] K. Y. Bai, X. M. Zhu, S. P. Wen, R. T. Zhang, and W. Y. Zhang, "Broad Learning Based Dynamic Fuzzy Inference System With Adaptive Structure and Interpretable Fuzzy Rules," *IEEE Trans. Fuzzy Syst.*, vol. 30, no. 8, pp. 3270-3283, Aug. 2022.

[19] C. Chen and Z. Liu, "Broad learning system: An effective and efficient incremental learning system without the need for deep architecture," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 1, pp. 10–24, Jan. 2018.

[20] Z. Liu, Z. Jin, and C. L. P. Chen, "Broad learning system: Feature extraction based on k-means clustering algorithm," *in Proc. Int. Conf. Inf.*, Dalian, China, pp. 683–687, Jul. 2017.

[21] X. R. Gong, T. Zhang, C. L. P. Chen, and Z. L. Liu, "Research Review for Broad Learning System: Algorithms, Theory, and Applications," *IEEE Trans. on Cybern.*, vol. 52, no. 9, pp. 8922-8950, Sep. 2022.

[22] W. D. Zou, Y. Q. Xia, and L. Dai. "Fuzzy Broad Learning System Based on Accelerating Amount," *IEEE Trans. Fuzzy Syst.*, vol. 30, no. 9, pp. 4017-4024, Sep. 2022.

[23] H. Huang, T. Zhang, C. Yang, and C. L. P . Chen, "Motor learning and generalization using broad learning adaptive neural control," *IEEE Trans. Ind. Electron.*, vol. 67, no. 10, pp. 8608–8617, Oct. 2020.

[24] S. Xu, Y. S. Ou, Z. Y. Wang, J. H. Duan, and H. Li, "Learning-Based Kinematic Control Using Position and Velocity Errors for Robot Trajectory Tracking," *IEEE Trans. Syst. Man Cybern. Syst.* vol. 52, no. 2, pp. 1100-1110, 2022.

[25] H. H. Huang, C. G. Yang, and C. L. P. Chen, "Optimal Robot-Environment Interaction Under Broad Fuzzy Neural Adaptive Control," *IEEE Trans. Cybern.*, vol. 51, no. 7, pp. 3824-3835, Jul. 2020.

[26] S. Xu, J. Liu, C. G. Yang, X. Y. Wu, and T. T. Xu, "A Learning-Based Stable Servo Control Strategy Using Broad Learning System Applied for Microrobotic Control," *IEEE Trans. Cybern.*, vol. 52, no. 12, pp. 13727 - 13737, Oct. 2021.

[27] C. F. Hsu, B. R. Chen, and B. F. Wu, "Fuzzy Broad Learning Adaptive Control for Voice Coil Motor Drivers," *Int. J. Fuzzy Syst.* vol. 24, no. 3, pp.1696-1707, Jan. 2022.

[28] S. M. Khansari-Zadeh and A. Billard, "Learning stable nonlinear dynamical systems with Gaussian mixture models," *IEEE Trans. Robot.*, vol. 27, no. 5, pp. 943–957, Oct. 2011.

[29] J. Hwangbo, J. Lee, A. Dosovitskiy, and et al., "Learning Agile and Dynamic Motor Skills for Legged Robots," *Sci. Robot.*, vol. 5, no. 47, Jan. 2019.

[30] Y. Liu, Y. F. Wang, L. Chen, J. Zhao, W. Wang, and Q. L. Liu. "Incremental Bayesian Broad Learning System and Its Industrial Application," *Artif. Intell Rev.* vol. 54, no. 5, pp. 3517-3537, Jun. 2021.

[31] Y. M. Li, S. C. Tong, "Adaptive Fuzzy Output Constrained Control Design for Multi-Input Multioutput Stochastic Nonstrict-Feedback Nonlinear Systems," *IEEE Trans. on Cybern.*, vol. 47, no. 12, pp. 4086-4095, Dec. 2017.

[32] T. Ning, Z. X. Ye, P. Yu, and F. L. Ni, "A Dual Fuzzy-Enhanced Neurodynamic Scheme for Model-Less Kinematic Control of Redundant and Hyperredundant Robots," *IEEE Trans. Fuzzy Syst.*, vol. 30, no. 10, pp. 4409-4422, Oct. 2022.