Intrinsic Plasticity-Based Neuroadptive Control With Both Weights and Excitability Tuning

Qing Chen, Anguo Zhang, and Yongduan Song, *Fellow, IEEE*

Abstract—**This brief presents an intrinsic plasticity (IP)-driven neural-network-based tracking control approach for a class of nonlinear uncertain systems. Inspired by the neural plasticity mechanism of individual neuron in nervous systems, a learning rule referred to as IP is employed for adjusting the radial basis functions (RBFs), resulting in a neural network (NN) with both weights and excitability tuning, based on which neuroadaptive tracking control algorithms for multiple-input–multiple-output (MIMO) uncertain systems are derived. Both theoretical analysis and numerical simulation confirm the effectiveness of the proposed method.**

Index Terms—**Intrinsic plasticity (IP), neuroadaptive control, nonlinear systems, radial basis function neural network (RBFNN).**

I. Introduction

Most practical systems not only involve nonlinearities but also contain modeling uncertainties, posing significant challenge for control design. Neural network (NN)-based methods have been widely adopted in solving various control problems associated with uncertain nonlinear systems [1]–[5]. For example, a NN-based approach is proposed for adaptive output feedback control of uncertain nonlinear systems in [2]. In [3], adaptive NN control for the robotic system with full-state constraints is designed. The works of [4] and [5] address two critical issues in NN control field, that is, how to fulfill the compact set precondition for NN approximation and how to use varying rather than a fixed NN structure to improve the functionality of NN control.

In particular, among different types of NNs, radial basis function neural networks (RBFNNs) have received considerable attention and have b

本文推导了一种基于RBF的IP规则，并将其应用于RBFNN，以提高传统RBFNN的性能，在此基础上提出了多输入多输出(MIMO)不确定系统的神经自适应跟踪控制方案

to the [6]–[9 functi The e two c unknown function in control system and the others are focused on designing the structure and adjusting the parameter to improve the performance of RBFNN. For instance, in [11], an incremental RBFNN that is constructed based on neuronal activity is proposed to solve the nonlinear system modeling issue. The authors in [12] propose to combine radial basis functions (RBFs) and cerebellar model arithmetic computer (CMAC) to prevent weight drift while avoiding the curse of dimensionality. In [13], RBFNN-based adaptive critic control of induction motors is proposed where not only the weights between hidden and output layers but also centers and widths of the RBF are adjusted by the projected learning rules.

However, these RBFNNs (or the modified versions) do not consider the practical biological characteristics. As the NN unit has been widely utilized in dealing with system nonlinearities and uncertainties encountered in control design, further improving its performance from the perspective of neurobiology becomes more and more interesting and practical [5], [14]. In this brief, the neural plasticity mechanism, which has been observed across brain areas of many species and regarded as a ubiquitous phenomenon in biological neurons, is incorporated into the traditional RBFNN. The idea of making use of IP to diversify or reshape the basis function in the NN unit is motivated by the fact that the intrinsic excitability of individual biological neurons can be adjusted to match the synaptic input by the activity of their voltage-gated channels [15], [16]. To better understand the intrinsic plasticity (IP) mechanisms, several IP models have been developed for different neuron-related applications. Stemmler and Koch [17] developed an IP model for a Hodgkin–Huxley model neuron. Inspired by neuroscience and information theory, Triesch [18] proposed an IP learning rule for the sigmoid transfer function. In [19] and [20], IP models for stochastically spiking neuron with refractoriness and IF neuron are presented, respectively. The effectiveness of related IP rules is also demonstrated l [22].

that a variety of computational models for IP rule have osed, there is still a lack of corresponding IP model that ed for RBF directly. Therefore, in this brief, a RBF-rule is derived and applied to RBFNN to improve the ce of traditional RBFNN, based on which a neuroadaptive tracking control scheme for multiple-input–multiple-output (MIMO) uncertain systems is developed. The main contributions of this brief are summarized as follows.

1) A new IP computational model for RBF is integrated to adjust the excitability of the NN unit.
2) The proposed IP is applied to the RBFNN in building the control scheme, which can adjust simultaneously the activation functions and weights online, providing an alternative approach for tracking controlling MIMO uncertain systems.
3) The proposed method is shown readily applicable to robotic systems, and numerical simulation confirms that the RBFNN with IP rule achieves better control performance compared with the traditional RBFNN without IP rule.

II. IP Rule-Based RBFNN and Its Approximating Capability

A. Radial Basis Function Neural Network

A traditional RBFNN includes an input layer, a hidden layer, and an output layer. Neurons at the hidden layer are activated by a RBF,

Manuscript received January 3, 2020; revised April 25, 2020; accepted July 9, 2020. Date of publication August 5, 2020; date of current version July 7, 2021. This work was supported in part by the National Natural Science Foundation of China under Grant 61803053, Grant 61833013, Grant 61860206008, and Grant 61773081; and in part by the Graduate Scientific Research and Innovation Foundation of Chongqing under Grant CYB19056 and Grant CYB19057. *(Corresponding author: Yongduan Song.)*

Qing Chen and Yongduan Song are with the State Key Laboratory of Power Transmission Equipment and System Security and New Technology, Chongqing University, Chongqing 400044, China, and also with the Chongqing Key Laboratory of Intelligent Autonomous Systems, School of Automation, Chongqing University, Chongqing 400044, China (e-mail: ydsong@cqu.edu.cn).

Anguo Zhang is with the College of Physics and Information Engineering, Fuzhou University, Fuzhou 350108, China, and also with the Research Institute of Ruijie, Ruijie Networks Company, Ltd., Fuzhou 350002, China.

Color versions of one or more of the figures in this article are available online at https://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TNNLS.2020.3011044

which contains a center vector c with the same dimension as the input vector z. The output of jth hidden neuron is

$$h_j(t) = \exp\left(-\frac{\|z(t) - c_j(t)\|^2}{2b_j^2}\right), \quad j = 1, 2, \ldots, J \tag{1}$$

where J denotes the number of hidden nodes and b_j denotes the width of jth hidden neuron. $\|z(t) - c_j(t)\|$ denotes the Euclidean distance between z and c_j. The output layer is a linear-weighted combination as follows:

$$\bar{y} = \sum_{j=1}^{J} w_j h_j(t), \quad j = 1, 2, \ldots, J \tag{2}$$

where $w = [w_1, w_2, \ldots, w_J]$ are the connection weights between the hidden layer and the output layer and $\bar{y}$ denotes the output of RBFNN.

B. IP Rule

As the IP rule is used for adjusting the intrinsic excitability of individual biological neurons, in this study, inspired from the work of [18], we first introduce an RBF with the following form and then implement an IP rule to shape the basis function:

$$\tilde{h}(z) = \exp\left(-\frac{p\|qz - c\|^2}{2b^2}\right) \tag{3}$$

where z is the input of RBF and c and b represent the center and width, respectively. Parameter p represents the width gain and parameter q represents the input gain. As exponential distributions have been observed in visual cortical neurons and are thought to allow the neuron to transmit the maximal mutual entropy of information following a given stationary level of metabolic cost [23], the goal of IP rule is to make the output distribution $f_{\tilde{h}}$ close to the desired exponential distribute $f_{\exp} = (1/\mu)\exp(-\tilde{h}/\mu)$, where μ denotes the mean value of $f_{\exp}$. Typically, the similarity between $f_{\exp}$ and $f_{\tilde{h}}$ can be measured by the Kullback–Leibler divergence (KL-divergence) [24]

$$\begin{aligned} D_{\mathrm{KL}}(f_{\tilde{h}} \parallel f_{\exp}) &= \int f_{\tilde{h}}(\tilde{h}) \log\left(\frac{f_{\tilde{h}}(\tilde{h})}{f_{\exp}(\tilde{h})}\right) d\tilde{h} \\ &= \int f_{\tilde{h}}(\tilde{h}) \log(f_{\tilde{h}}(\tilde{h})) d\tilde{h} \\ &\quad - \int f_{\tilde{h}}(\tilde{h}) \left(-\frac{\tilde{h}}{\mu} - \log\mu\right) d\tilde{h} \\ &= -H(\tilde{h}) + \frac{1}{\mu} E(\tilde{h}) + \log\mu \end{aligned}$$

where $H(\tilde{h})$ and $E(\tilde{h})$ are the entropy and expected values of $\tilde{h}$, respectively. Upon simple inspection, we see that D_{KL} is minimized if $H(\tilde{h})$ is maximized and $E(\tilde{h})$ is minimized ($\log\mu$ is fixed). To proceed, we take derivatives of D_{KL} with respect to p and q, respectively, and it is obtained that

$$\begin{aligned} \frac{\partial D_{KL}}{\partial p} &= -\frac{\partial H(\tilde{h})}{\partial p} + \frac{1}{\mu}\frac{\partial E(\tilde{h})}{\partial p} \\ &= E\left[-\frac{\partial}{\partial p}\log\left(\left|\frac{\partial \tilde{h}}{\partial z}\right|\right) + \frac{1}{\mu}\frac{\partial \tilde{h}}{\partial p}\right] \\ &= -\frac{1}{p} + E\left[\left(1 + \frac{\tilde{h}}{\mu}\right)\frac{1}{2b^2}\|qz - c\|^2 - \frac{1}{pq\|qz - c\|}\right] \\ \frac{\partial D_{KL}}{\partial q} &= -\frac{\partial H(\tilde{h})}{\partial q} + \frac{1}{\mu}\frac{\partial E(\tilde{h})}{\partial q} \\ &= E\left[-\frac{\partial}{\partial q}\log\left(\left|\frac{\partial \tilde{h}}{\partial z}\right|\right) + \frac{1}{\mu}\frac{\partial \tilde{h}}{\partial q}\right] \\ &= -E\left[\frac{z}{\|qz - c\|} - \frac{\tilde{y}}{\mu}\frac{p}{b^2}z^T\|qz - c\|\right] \end{aligned}$$

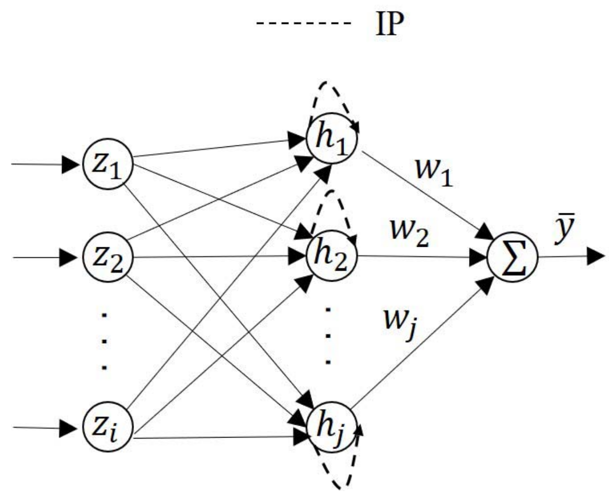

Fig. 1. Schematic of RBFNN with IP for reshaping basis function.

where

$$\begin{aligned} H(\tilde{h}) &= -\int f_{\tilde{h}}(\tilde{h}) \log(f_{\tilde{y}}(\tilde{h})) d\tilde{h} \\ &= E\left[\log\left(\left|\frac{\partial \tilde{h}}{\partial x}\right|\right) + \frac{1}{\mu}\frac{\partial \tilde{h}}{\partial p}\right] \\ \frac{\partial \tilde{h}}{\partial z} &= -\tilde{h}\frac{pq}{b^2}\|qz - c\| \\ \frac{\partial \tilde{h}}{\partial p} &= -\tilde{h}\frac{1}{b^2}\|qz - c\|^2 \\ \frac{\partial \tilde{h}}{\partial q} &= -\tilde{h}\frac{zp}{b^2}\|qz - c\|. \end{aligned}$$

The resulting IP rule to be used for updating the parameters p and q is as follows:

$$\begin{aligned} \Delta p &= \eta\left(\frac{1}{p} - \left(1 + \frac{\tilde{h}}{\mu}\right)\frac{1}{2b^2}\|qz - c\|^2 + \frac{1}{qp|I^T(qz - c)|}\right) \\ \Delta q &= \eta\left(\frac{I^T z}{|I^T(qz - c)|} - \frac{\tilde{h}}{\mu}\frac{p}{b^2}z^T(qz - c)\right) \end{aligned} \tag{4}$$

where I is a unit vector with the same dimension as that of the NN input vector z and η is the small learning rate. Note that the adjustment of p and q actually reshapes the basis function (or the excitability). The schematic of RBFNN with IP is shown in Fig. 1.

C. Universal Approximation Theory

According to the universal approximation theory, for any given real continuous function $f(z) : \mathbb{R}^n \to \mathbb{R}$ within a compact set Ω_z and constant $\epsilon > 0$, there exists an ideal RBFNN equipped with an appropriate network structure and a sufficiently large number of neurons such that

$$\sup_{z \in \Omega_z} |f(z) - \bar{y}(z)| \leq \epsilon. \tag{5}$$

The function $f(z)$ can then be expressed as

$$f(z) = w^{*T} h(z) + \epsilon^* \quad \forall z \in \Omega_z \tag{6}$$

where $|\epsilon^*| \leq \epsilon$, ϵ^* is the approximation error, $w^{*T} \in \mathbb{R}^{N \times 1}$ is the optimal output weight that minimizes the approximation error, and

N is the number of hidden neuron of RBFNN. Therefore

$$w^* = \arg \min_{w^{*T} \in \mathbb{R}^{N\times 1}} \left\{ \sup_{z\in\Omega_z} |f(z) - w^{*T} h(z)| \right\}. \quad (7)$$

III. Main Results

A. System Description and Problem Statement

Consider the following class of MIMO uncertain systems:

$$\begin{aligned} \dot{x}_i &= x_{i+1}, \quad i = 1, 2, \ldots, n-1 \\ \dot{x}_n &= F(x,t) + G(x,t)u_b \\ y &= x_1 \end{aligned} \quad (8)$$

where $x_i = [x_{i1}, \ldots, x_{im}]^T \in \mathbb{R}^m, i = 1, 2, \ldots, n$ and $x = [x_1, \ldots, x_n]^T \in \mathbb{R}^{mn}$ are the system state vector, $u_b \in \mathbb{R}^r$ is the input vector of the system, $G(x,t) \in \mathbb{R}^{m\times r}$ $(m < r)$ is the time-varying and uncertain nonsquare control gain, $F(x,t) \in \mathbb{R}^m$ denotes all the lumped uncertainties and external disturbances, and $y \in \mathbb{R}^m$ is the output vector of the system.

In this brief, the control objective is to design a neuroadaptive control scheme for system (8) such that the system state $x_1 = [x_{11}, \ldots, x_{1m}]^T$ closely follows the desired trajectory $x_d = [x_{d1}, \ldots, x_{dm}]^T$ and all the closed-loop signals in the system are bounded.

In order to achieve the abovementioned control objectives, the following assumptions are imposed.

Assumption 1: The desired trajectory $x_{di}, i = 1, \ldots, m$ is known bounded function of time with bounded known derivatives.

Assumption 2: The states of the system are measurable.

Assumption 3: There exists a known and bounded matrix $G_1(x) \in \mathbb{R}^{m\times r}$ with full row rank and an unknown matrix $G_2(x,t) \in \mathbb{R}^{r\times r}$ such that $G(x,t) = G_1(x)G_2(x,t)$ and $G^*(\cdot) = (G_2 + G_2^T)/2$ is positive definite.

Remark 1: The condition on the desired trajectory as imposed in Assumption 1 is commonly utilized in addressing trajectory tracking [25]. Assumption 2 is also necessary for tackling the tracking control problem for such type of systems. If only partial states are measurable, state observer is normally needed, which, however, is beyond the scope of this work. The condition imposed on $G^*(\cdot)$ guarantees that the system is controllable, which has been commonly used by most existing works in addressing MIMO systems [26].

B. Controller Design and Stability Analysis

Define tracking error $E = x_1 - x_d$, and for the later analysis convenience, the following filter error ξ is introduced:

$$\xi(t) = \lambda_1 E + \lambda_2 \dot{E} + \cdots + \lambda_{n-1} E^{(n-2)} + E^{(n-1)} \quad (9)$$

where λ_i, $i = 1, \ldots, n-1$ are some positive constants chosen by the user such that the polynomial $s^{(n-1)} + \lambda_{n-1} s^{(n-2)} + \cdots + \lambda_1$ is Hurwitz. Differentiating (9), we can obtain that

$$\begin{aligned} \dot{\xi}(t) &= \lambda_1 \dot{E} + \cdots + \lambda_{n-1} E^{(n-1)} + E^{(n)} \\ &= G(x,t)u_b + L(x,t) \end{aligned} \quad (10)$$

where $L(x,t) = \lambda_1 \dot{E} + \cdots + \lambda_{n-1} E^{(n-1)} + F(x,t) - x_d^{(n)}$. In this brief, $L(\cdot)$ is approximated by the proposed IP-based RBFNN. However, different from the most existing studies that approximate $L(\cdot)$ directly, in this study, the RBFNN is used to approximate the upper bound of $L(\cdot)$ to simplify the control structure and reduce the online computations, that is

$$\begin{aligned} \|L(\cdot)\| &= \omega_b^{*T} \phi(z_b) + \epsilon_b \\ &\le \|\omega_b^{*T}\| \|\phi(z_b)\| + \epsilon_{bm} \\ &\le \varpi_1 \Psi(z_b) \end{aligned} \quad (11)$$

where ω_b^* is the optimal constant weight, z_b is the input of RBFNN, $\phi(\cdot)$ is the RBF, ϵ_b is the approximation error, and $|\epsilon_b| \le \epsilon_{bm}$ with ϵ_{bm} being some unknown positive constant, $\varpi_1 = \max\{\|\omega_b^{*T}\|, \epsilon_{bm}\}$, and $\Psi(z_b) = \|\phi(z_b)\| + 1$. Then, the control scheme is proposed as follows:

$$u_b = -\frac{G_1^T}{\|G_1\|}(k_1 \xi + c_1 \hat{\varpi}_1 \Psi^2(z_b) \xi) \quad (12)$$

with

$$\dot{\hat{\varpi}}_1 = -\beta_1 \hat{\varpi}_1 + c_1 \|\xi\|^2 \Psi^2(z_b) \quad (13)$$

where $k_1 > 0$, $\beta_1 > 0$, and $c_1 > 0$ are constants chosen by the designer, $\hat{\varpi}_1$ is the estimated value of ϖ_1, and $\hat{\varpi}_1(0) \ge 0$ is the arbitrarily chosen initial value.

Theorem 1: Consider system (8) with uncertainties and external disturbances. Suppose that Assumptions 1–3 hold. If the control law (12) and the adaptive law (13) are applied, then the system output tracks the desired trajectory closely in which the tracking error is ensured to be uniformly ultimately bounded. Furthermore, all the closed-loop signals in the system are bounded.

Proof: Consider the following Lyapunov function candidate:

$$V = \frac{1}{2}\xi^T \xi + \frac{1}{2g_1}\tilde{\varpi}_1^2 \quad (14)$$

where $\tilde{\varpi}_1 = \varpi_1 - g_1 \hat{\varpi}_1$ and g_1 is positive constant will be defined later. Taking time derivative of (14) and inserting (10), it can be obtained that

$$\begin{aligned} \dot{V} &= \xi^T (G u_b + L) - \tilde{\varpi}_1 \dot{\hat{\varpi}}_1 \\ &\le \xi^T G u_b + \|\xi\| \|L\| - \tilde{\varpi}_1 \dot{\hat{\varpi}}_1. \end{aligned} \quad (15)$$

Using (11) and (12), $\dot{V}$ becomes

$$\begin{aligned} \dot{V} \le -k_1 \xi^T \frac{G_1 G_2 G_1^T}{\|G_1\|}\xi - c_1 \hat{\varpi}_1 \Psi^2(z_b) \xi^T \frac{G_1 G_2 G_1^T}{\|G_1\|}\xi \\ + \varpi_1 \Psi(z_b)\|\xi\| - \tilde{\varpi}_1 \dot{\hat{\varpi}}_1 \end{aligned} \quad (16)$$

where

$$\begin{aligned} -\xi^T \frac{G_1 G_2 G_1^T}{\|G_1\|}\xi = -\xi^T \frac{G_1 (G_2 + G_2^T) G_1^T}{2\|G_1\|}\xi \\ - \xi^T \frac{G_1 (G_2 - G_2^T) G_1^T}{2\|G_1\|}\xi. \end{aligned} \quad (17)$$

It can be verified that $(G_1(G_2 + G_2^T)G_1^T)/2$ is positive definite under Assumption 3, and therefore, there exists some unknown positive constant g_1 satisfying $0 < g_1 < (1)/(\|G_1\|)\min\{\text{eig}(G^*)\}$. Also, $G_1(G_2 - G_2^T)G_1^T$ is skew symmetric, and hence, $\xi^T(G_1(G_2 - G_2^T)G_1^T)\xi = 0$. Consequently, we have

$$-\xi^T \frac{G_1 G_2 G_1^T}{\|G_1\|}\xi \le -g_1 \|\xi\|^2. \quad (18)$$

Inserting (18) into (16), $\dot{V}$ becomes

$$\begin{aligned} \dot{V} \le -k_1 g_1 \|\xi\|^2 - c_1 g_1 \hat{\varpi}_1 \Psi^2(z_b)\|\xi\|^2 \\ + \varpi_1 \Psi(z_b)\|\xi\| - \tilde{\varpi}_1 \dot{\hat{\varpi}}_1. \end{aligned} \quad (19)$$

By using Young inequality, it follows that for any $c_1 > 0$

$$\Psi(z_b)\|\xi\| \le \frac{1}{4c_1} + c_1 \Psi^2(z_b)\|\xi\|^2. \quad (20)$$

Applying (13) and (20) to (19) yields

$$\begin{aligned} \dot{V} &\le -k_1 g_1 \|\xi\|^2 + \frac{\varpi_1}{4c_1} + \beta_1 \tilde{\varpi}_1 \hat{\varpi}_1 \\ &\le -k_1 g_1 \|\xi\|^2 - \frac{\beta_1}{2g_1}\tilde{\varpi}_1^2 + \frac{\varpi_1}{4c_1} + \frac{\beta_1}{2g_1}\varpi_1^2 \\ &\le -\varrho_1 V + \varrho_2 \end{aligned} \quad (21)$$

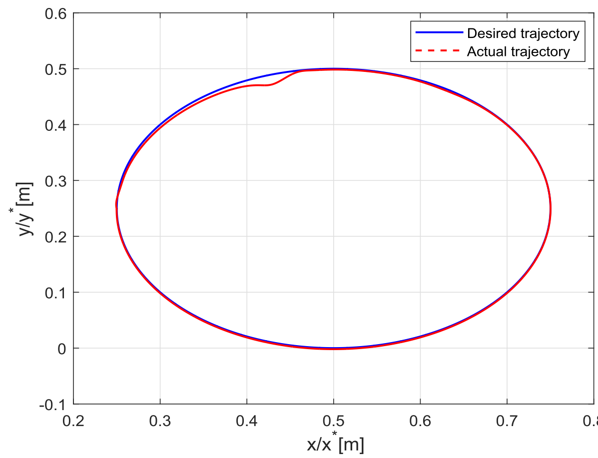

Fig. 2. Tracking processing.

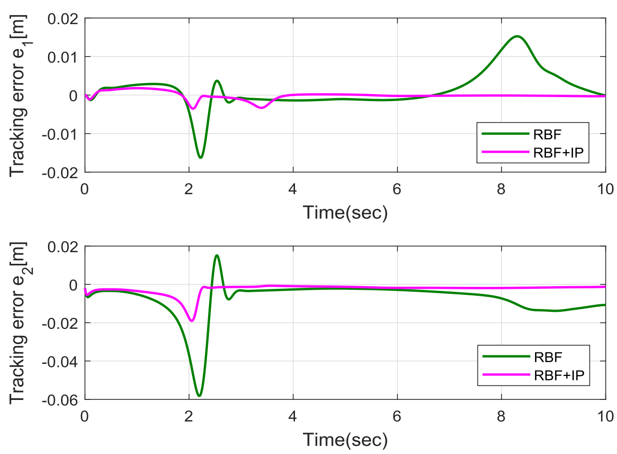

Fig. 3. Tracking errors comparison between the RBF and the RBF+IP.

where $\varrho_1 = \min\{2k_1 g_1, \beta_1\}$, $\varrho_2 = (\varpi_1)/(4c_1) + (\beta_1)/(2g_1)\varpi_1^2$. From (21), it is readily obtained that $V \in \ell_\infty$, which implies that $\xi \in \ell_\infty$ and $\tilde{\varpi}_1(\hat{\varpi}_1) \in \ell_\infty$. Then, in view of the relation between ξ and E, it holds that $E \in \ell_\infty$, and hence, $x \in \ell_\infty$ and $\Psi \in \ell_\infty$. From (12), it is seen that $u_b \in \ell_\infty$. Furthermore, from (21), it follows that $\dot{V} < 0$ as long as $\|\xi\|^2 > (\varpi_1)/(4c_1k_1g_1) + \beta_1/2k_1g_1^2\varpi_1^2$ and $\tilde{\varpi}_1^2 > (g_1\varpi_1)/(2c_1\beta_1) + \varpi_1^2$. Thus, ξ and $\tilde{\varpi}_1$ are uniformly ultimately bounded as $\|\xi\| < ((\varpi_1)/(4c_1k_1g_1) + \beta_1/2k_1g_1^2\varpi_1^2)^{1/2}$ and $|\tilde{\varpi}_1| < ((g_1\varpi)/(2c_1\beta_2) + \varpi_1^2)^{1/2}$. Thus, E and its derivatives up to $n-1$ are uniformly ultimately bounded. ■

Remark 2: Different from traditional RBFNN-based control that only involves weights tuning, here additionally, an IP-based mechanism is implemented for adjusting the RBF, resulting in an alternative NN control scheme with both weights and activation/excitability tuning. As the IP mechanism is used for adjusting basis function only, it has no impact on the control design process and stability analysis (convergence). Nevertheless, it might demand additional online computation, which, however, is bearable given the current computing resources.

IV. Application to Robotic Systems and Simulation Verification

Now, we examine the applicability of the proposed method to the real-world system—robotic system. Consider an n-joint rigid-link robotic manipulator with the following joint-space dynamics

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) + F(\dot{q}) = \tau$$

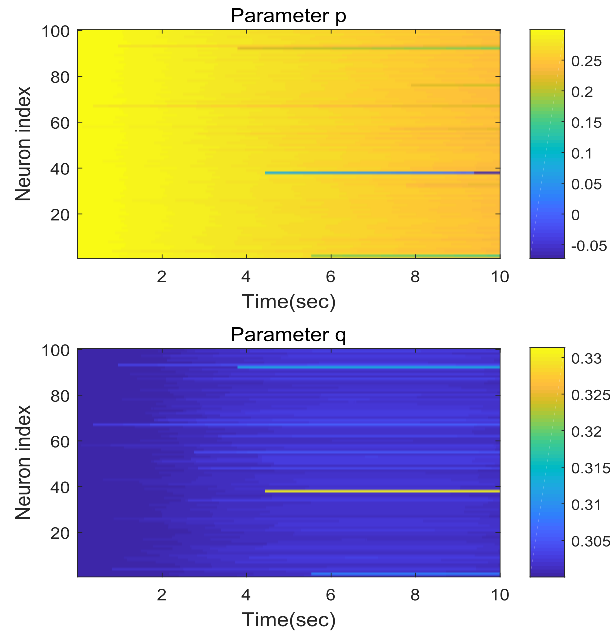

Fig. 4. Evolution of parameters p and q.

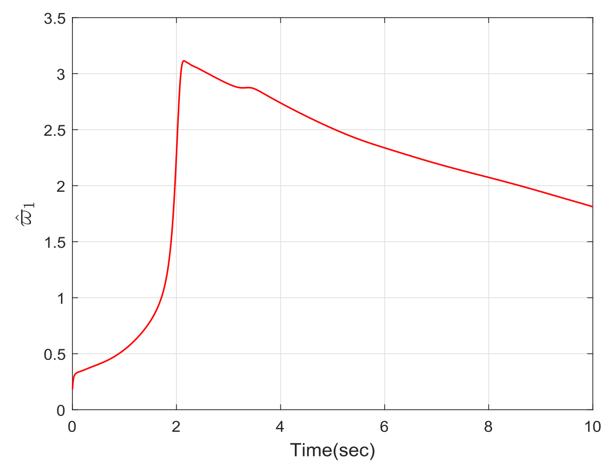

Fig. 5. Estimate parameter $\hat{\varpi}_1$ updating process.

where $M(q) \in \mathbb{R}^{n\times n}$ is the inertia matrix, $q \in \mathbb{R}^n$ is generalized joint coordinates, $C(q,\dot{q}) \in \mathbb{R}^{n\times n}$ is the matrix containing the centrifugal and Coriolis terms, $G(q) \in \mathbb{R}^n$ vector containing gravitational forces and torques, $\tau \in \mathbb{R}^n$ is joint torques, and $F(\dot{q}) \in \mathbb{R}^n$ denotes disturbances. When operating in 3-D space described by the Cartesian coordinates $X = [x, y, z]^T$, the tracking error is $E = X - X_d$ and the filtered error is $\xi = \dot{E} + \lambda_1 E$, where $X_d = [x^*, y^*, z^*]^T$ is the desired trajectory for the manipulator's end-effector to track and λ_1 is some positive constant. As $\dot{X} = J\dot{q}$, where $J \in \mathbb{R}^{3\times n}$ is the Jacobian matrix, it is straightforward to derive the following task-space error dynamics:

$$\begin{aligned}\dot{\xi} &= \ddot{E} + \lambda_1\dot{E}\\ &= \dot{J}\dot{q} + J\ddot{q} - \ddot{X}_d + \lambda_1(\dot{X} - \dot{X}_d)\\ &= JM^{-1}(q)\tau + L(\cdot)\end{aligned}$$

where $L(\cdot) = \dot{J}\dot{q} - JM^{-1}(q)(F(\dot{q}) + G(q) + C(q,\dot{q})\dot{q}) - \ddot{X}_d + \lambda_1(\dot{X} - \dot{X}_d)$ is the lumped uncertainty, $JM^{-1}(q) \in \mathbb{R}^{3\times n}$ is nonsquare with J being a known full row-rank, and $M^{-1}(q) \in \mathbb{R}^{n\times n}$. It can be verified that $(M^{-1} + (M^{-1})^T)/2$ is symmetric and positive definite. Thus, $J(M^{-1} + (M^{-1})^T)J^T/2$ is symmetric and positive definite,

TABLE I
PERFORMANCE STATISTICS COMPARISON

	$\int \lvert e_1(t)\rvert dt$	$\int \lvert e_2(t)\rvert dt$	$\int (e_1(t)-\bar{e}_1)^2 dt$	$\int (e_2(t)-\bar{e}_2)^2 dt$
RBF	32.3056	80.8112	0.2348	0.9534
RBF+IP	6.2592	24.5716	0.0096	0.0739

meaning that Assumption 3 holds. Therefore, the proposed control is readily applicable to motion control of robotic systems, and the corresponding control scheme becomes

$$\tau = -\frac{J^T}{\|J\|}(k_1\xi + c_1\hat{\varpi}_1\Psi^2\xi)$$
$$\dot{\hat{\varpi}}_1 = -\beta_1\hat{\varpi}_1 + c_1\|\xi\|^2\Psi^2(z_b).$$

To validate the effectiveness of the developed control scheme, we conduct simulation on making the end-effector of a three-joint robot track a circle in the 2-D plane. In the simulation, $k_1 = 30$, $c_1 = 1$, $\beta_1 = 0.1$, and $\lambda_1 = 15$. The desired trajectory $x^* = 0.5 - 0.25\cos(\omega t)$ and $y^* = 0.25 + 0.25\sin(\omega t)$ with $\omega = 0.2\pi$. The initial conditions are $q(0) = [-0.16, 1.22, 2.30]$ (rad), $\dot{q}(0) = [0, 0, 0]$ (rad/s), and $\hat{\varpi}_1(0) = 0$. The number of neuron used in RBFNN is 100. The center of RBFNN is set as 0, and the same width for each basic function is 2. The parameters used in IP rule are set as $\eta = 0.0001$, $\mu = 0.1$. The simulation results are shown in Figs. 2–5, where Fig. 2 shows the trajectory tracking in $x - y$ position, and Fig. 3 shows the tracking error. In addition, we examine the integrated absolute error $\int |e(t)|dt$ to measure the system tracking performance and the integrated square error $(\int |e(t) - \bar{e})^2 dt$ ($\bar{e}$ is the average error of the whole process) [27] to evaluate the smoothness of the tracking profile and the results are presented in Table I. From Fig. 3 and Table I, it is seen that the control scheme with IP-driven NN unit can achieve better control performance compared with the one without IP rule. Fig. 4 shows the evolution of parameters p and q for all of the hidden neurons (here, a total number of 100 neurons are used), and Fig. 5 shows the evolution process of the estimate parameter $\hat{\varpi}_1$.

V. CONCLUSION

In this study, neuroadaptive control schemes based on RBFNN with IP rule are proposed to address the tracking problem of a class of nonlinear systems. The control precision, compared with that obtained by traditional RBFNN control, is evidently improved with the proposed method due to the simultaneous online adjustment of both activation functions and neural weights. It is noted that the computation complexity here is slightly increased, but with the current computing technology, this seems not a crucial issue. Furthermore, as IP is not the unique neural plasticity mechanism during the learning activities, an interesting topic for further research is to investigate the influence of multineural plasticity on the performance of NN for controlling more general nonlinear systems.

REFERENCES

[1] K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators," *Neural Netw.*, vol. 2, no. 5, pp. 359–366, Jan. 1989.

[2] N. Hovakimyan, F. Nardi, A. Calise, and N. Kim, "Adaptive output feedback control of uncertain nonlinear systems using single-hidden-layer neural networks," *IEEE Trans. Neural Netw.*, vol. 13, no. 6, pp. 1420–1431, Nov. 2002.

[3] W. He, Y. Chen, and Z. Yin, "Adaptive neural network control of an uncertain robot with full-state constraints," *IEEE Trans. Cybern.*, vol. 46, no. 3, pp. 620–629, Mar. 2016.

[4] Z.-J. Jia and Y.-D. Song, "Barrier function-based neural adaptive control with locally weighted learning and finite neuron self-growing strategy," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 6, pp. 1439–1451, Jun. 2017.

[5] Y. D. Song, X. C. Huang, and Z. J. Jia, "Dealing with the issues crucially related to the functionality and reliability of NN-associated control for nonlinear uncertain systems," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 11, pp. 2614–2625, Nov. 2017.

[6] Y.-J. Liu, J. Li, S. Tong, and C. L. P. Chen, "Neural network control-based adaptive learning design for nonlinear systems with full-state constraints," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 7, pp. 1562–1571, Jul. 2016.

[7] K. Zhao, Y. Song, T. Ma, and L. He, "Prescribed performance control of uncertain Euler–Lagrange systems subject to full-state constraints," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 8, pp. 3478–3489, Aug. 2018.

[8] B. Ren, S. S. Ge, K. P. Tee, and T. H. Lee, "Adaptive neural control for output feedback nonlinear systems using a barrier Lyapunov function," *IEEE Trans. Neural Netw.*, vol. 21, no. 8, pp. 1339–1345, Aug. 2010.

[9] T. P. Zhang and S. S. Ge, "Adaptive neural control of MIMO nonlinear state time-varying delay systems with unknown dead-zones and gain signs," *Automatica*, vol. 43, no. 6, pp. 1021–1033, Jun. 2007.

[10] J. Park and I. W. Sandberg, "Universal approximation using radial-basis-function networks," *Neural Comput.*, vol. 3, no. 2, pp. 246–257, Jun. 1991.

[11] J. Qiao, X. Meng, and W. Li, "An incremental neuronal-activity-based RBF neural network for nonlinear system modeling," *Neurocomputing*, vol. 302, pp. 1–11, Aug. 2018.

[12] C. J. B. Macnab, "Using RBFs in a CMAC to prevent parameter drift in adaptive control," *Neurocomputing*, vol. 205, pp. 45–52, Sep. 2016.

[13] C.-K. Lin, "Radial basis function neural network-based adaptive critic control of induction motors," *Appl. Soft Comput.*, vol. 11, no. 3, pp. 3066–3074, Apr. 2011.

[14] Z.-J. Jia, Y.-D. Song, D.-Y. Li, and P. Li, "Tracking control of nonaffine systems using bio-inspired networks with auto-tuning activation functions and self-growing neurons," *Inf. Sci.*, vols. 388–389, pp. 191–208, May 2017.

[15] R. H. Cudmore and G. G. Turrigiano, "Long-term potentiation of intrinsic excitability in LV visual cortical neurons," *J. Neurophysiol.*, vol. 92, no. 1, pp. 341–348, Jul. 2004.

[16] G. Daoudal, "Long-term plasticity of intrinsic excitability: Learning rules and mechanisms," *Learn. Memory*, vol. 10, no. 6, pp. 456–465, Nov. 2003.

[17] M. Stemmler and C. Koch, "How voltage-dependent conductances can adapt to maximize the information encoded by neuronal firing rate," *Nature Neurosci.*, vol. 2, no. 6, pp. 521–527, Jun. 1999.

[18] J. Triesch, "A gradient rule for the plasticity of a neuron's intrinsic excitability," in *Proc. 15th Int. Conf. Artif. Neural Netw.*, Sep. 2005, pp. 65–70.

[19] C. Savin, P. Joshi, and J. Triesch, "Independent component analysis in spiking neurons," *PLoS Comput. Biol.*, vol. 6, no. 4, Apr. 2010, Art. no. e1000757.

[20] C. Li and Y. Li, "A spike-based model of neuronal intrinsic plasticity," *IEEE Trans. Auto. Mental Develop.*, vol. 5, no. 1, pp. 62–73, Mar. 2013.

[21] B. Schrauwen, M. Wardermann, D. Verstraeten, J. J. Steil, and D. Stroobandt, "Improving reservoirs using intrinsic plasticity," *Neurocomputing*, vol. 71, nos. 7–9, pp. 1159–1171, Mar. 2008.

[22] X. Wang, Y. Jin, and K. Hao, "Echo state networks regulated by local intrinsic plasticity rules for regression," *Neurocomputing*, vol. 351, pp. 111–122, Jul. 2019.

[23] R. Baddeley *et al.*, "Responses of neurons in primary and inferior temporal visual cortices to natural scenes," *Proc. Roy. Soc. London B, Biol. Sci.*, vol. 264, no. 1389, pp. 1775–1783, Dec. 1997.

[24] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. New York, NY, USA: Wiley, 2006.

[25] W. Wang and C. Wen, "Adaptive actuator failure compensation control of uncertain nonlinear systems with guaranteed transient performance," *Automatica*, vol. 46, no. 12, pp. 2082–2091, Dec. 2010.

[26] Y. Song, X. Huang, and C. Wen, "Tracking control for a class of unknown nonsquare MIMO nonaffine systems: A deep-rooted information based robust adaptive approach," *IEEE Trans. Autom. Control*, vol. 61, no. 10, pp. 3227–3233, Oct. 2016.

[27] Q. Chen, L. Shi, J. Na, X. Ren, and Y. Nan, "Adaptive echo state network control for a class of pure-feedback systems with input and output constraints," *Neurocomputing*, vol. 275, pp. 1370–1382, Jan. 2018.