IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received April 19, 2021, accepted May 7, 2021, date of publication May 12, 2021, date of current version May 25, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3079446

A New Self-Organizing Double Function-Link Brain Emotional Learning Controller for MIMO Nonlinear Systems Using Sliding Surface

CHIH-MIN LIN[1], (Fellow, IEEE), HIEP-BINH NGUYEN[1], AND TUAN-TU HUYNH[1,2], (Member, IEEE)

[1]Department of Electrical Engineering, Yuan Ze University, Taoyuan 320, Taiwan
[2]Faculty of Mechatronics and Electronics, Lac Hong University, Bien Hoa 810000, Vietnam

Corresponding authors: Chih-Min Lin (cml@saturn.yzu.edu.tw) and Tuan-Tu Huynh (huynhtuantu@saturn.yzu.edu.tw)

This work was supported in part by the Ministry of Science and Technology of Republic of China under Grant MOST 109-2811-E-155-504-MY3.

ABSTRACT This paper aims to propose a new type of neural network which is the self-organizing double function-link brain emotional learning controller (SO-DFL-BELC) for multiple input multiple output (MIMO) nonlinear systems. The proposed controller is a newly designed neural network containing the key mechanism of a typical brain emotional learning controller (BELC), which is a mathematical model that approximates the judgmental and emotional activity of a brain, in which it is combined with some additional functions and methods. Firstly, a double function-link (DFL) network is applied to expand the function for a BELC to improve the accuracy of the system weights. Secondly, the self-organizing mechanism is utilized to increase or decrease the number of neurons that possibly supports the main controller to adapt to the sharp change of the input and to reduce the computation time significantly. Thirdly, the learning rules of the SO-DFL-BELC are developed based on the gradient descent algorithm and sliding surface. Finally, all parameters of the system can be optimized. The proposed SO-DFL-BELC is applied to control two different MIMO nonlinear systems that are a 4D chaotic system and a four-tank system. The simulation results show the favorable control performance of the proposed control algorithm.

INDEX TERMS Brain emotional learning controller, function-link, self-organizing mechanism, sliding surface, 4D chaotic system, four-tank system.

I. INTRODUCTION

For dealing with system uncertainty, many disturbance observers-based control approaches have also been widely researched such as extended-state-observer-based output feedback backstepping control of hydraulic actuators with valve dynamics compensation [1], extended-state-observer-based adaptive control of electrohydraulic servomechanisms without velocity measurement [2], and an output feedback approach for time-varying input delay compensation for nonlinear systems with additive disturbance [3]. In recent years, in order to deal with complex computational problems, there is the increasing use of methods that are inspired by the structure of the human brain. Brain emotional learning controller (BELC) or brain emotional learning-based intelligent controller (BELBIC) is one of those methods. The idea is based on the research of Moren, Balkenius, and LeDoux that shows some parts of the brain produce emotion signals in the amygdala cortex and the orbitofrontal cortex [4]–[8]. From all of those studies, the structure of a BELC was built and utilized to control various systems such as doubly fed induction generator systems [9], omni-directional three-wheel robots [10], interior permanent-magnet synchronous motor drives [11], encoderless synchronous reluctance motor drives [12], etc. In this paper, the structure of BELC is modified using some efficient methods outlined below to improve its efficiency.

For neural networks, using a function-link network is an effective method for function approximation in order to get quick convergence speed and reduce the system error [13], and it is widely included in different types of neural networks and to control a great variety of systems [14], [15]. This paper proposes the design to apply function-link networks to a BELC. Since the BELC is a special type of neural network that has two independent networks, there is a requirement

The associate editor coordinating the review of this manuscript and approving it for publication was Juntao Fei.

of two separate function-link systems [16]. It is called as a double function-link (DFL) network.

Beside the DFL network, the novel structure of the proposed controller includes the self-organizing (SO) mechanism, which is also a potential technique to improve the quality of neural networks [17], [18]. The aim of the SO mechanism is to evaluate the important index of each neuron, thus the ability of structure of BELC can be learned when the old unimportant neurons are deleted or the new neurons are generated. By this way, noticeable computation time is saved and also the controller can adapt to the large variation of inputs.

For the original BELC, only weights are updated overtime. However, the proposed SO-DFL-BELC has a new point that is the updating rules for all means, variances, and weights which are built based on sliding surface [19] and the gradient descent algorithm [20]. Hence, the controller can learn more efficiently and completely.

In order to illustrate the effectiveness of the SO-DFL-BELC, it is used to control a 4D chaotic system. The first discovery of a chaotic system was in 1963, with the pioneer work of Lorenz, that it led to the famous chaos theory [21]. So far, the chaotic systems have attracted a great deal of attention from numerous scientists because of their application to various fields such as image encryption schemes [22], [23], communication systems [24], [25] or chaotic circuits [26], [27]. Therefore, this study uses the 4D chaotic system to test the effectiveness of the SO-DFL-BELC.

Beside the 4D chaotic system, the SO-DFL-BELC is also designed to control a four-tank system [28], [29] The four tank system is widely used to evaluate the control system design because it includes several interesting properties such as it is the nonlinear systems and its subsystems can affect each other [30].

In conclusion, this paper proposes a new type of neural network, which is SO-DFL-BELC, for MIMO system. The structure of the SO-DFL-BELC is the combination of the BELC with the SO mechanism and the DFL network. The updating rules of the SO-DFL-BELC are built based on sliding surface for all means, variances, and weights. The SO-DFL-BELC is utilized to control a 4D chaotic system and a four-tank system to illustrate its effectiveness. Comparisons between the SO-DFL-BELC and some different controllers show that the SO-DFL-BELC achieves better performance and smaller root mean square error (RMSE) than other controllers.

II. PROBLEM FORMULATION

A class of MIMO nonlinear systems is described by the following equation:

$$\begin{cases} x_1^{(n)} = f_1 + g_1 u_1 + \beta_1 \\ x_2^{(n)} = f_2 + g_2 u_2 + \beta_2 \\ \qquad \cdots \\ x_\gamma^{(n)} = f_\gamma + g_\gamma u_\gamma + \beta_\gamma \\ \qquad \cdots \\ x_N^{(n)} = f_N + g_N u_N + \beta_n \end{cases} \tag{1}$$

where N is the number of the system outputs and also the number of the control signals; $x_1, x_2, \ldots, x_N$ and $u_1, u_2, \ldots, u_N$ denote the system outputs and the control signals, respectively; $f_1, f_2, \ldots, f_N$ and $g_1, g_2, \ldots, g_N$ are the smooth nonlinear functions which are exactly known; $\beta_1, \beta_2, \ldots, \beta_N$ are the lumped unknown uncertainties.

The γ-th ideal controller can be designed as:

$$\begin{aligned} u_\gamma^* = g_\gamma^{-1} \Big[x_\gamma^{*(n)} - f_\gamma - \beta_\gamma + \alpha_{\gamma 1} \tilde{x}_\gamma^{(n-1)} + \alpha_{\gamma 2} \tilde{x}_\gamma^{(n-2)} \\ + \ldots + \alpha_{\gamma n} \tilde{x}_\gamma \Big] \end{aligned} \tag{2}$$

$$\begin{aligned} x_\gamma^{*(n)} + \alpha_{\gamma 1} \tilde{x}_\gamma^{(n-1)} + \alpha_{\gamma 2} \tilde{x}_\gamma^{(n-2)} + \ldots \\ + \alpha_{\gamma n} \tilde{x}_\gamma = f_\gamma + g_\gamma u_\gamma^* + \beta_\gamma \end{aligned} \tag{3}$$

where $\alpha_{\gamma 1}, \alpha_{\gamma 2}, \ldots, \alpha_{\gamma n}$ are the gains that correspond to the coefficients of a Hurwitz polynomial; x_γ^* is the γ-th ideal system output; $\tilde{x}_\gamma$ is the error between the γ-th ideal system output x_γ^* and the γ-th actual system output x_γ:

$$\tilde{x}_\gamma = x_\gamma^* - x_\gamma \tag{4}$$

The error between the γ-th ideal controller u_γ^* and the γ-th actual controller u_γ is defined as:

$$\tilde{u}_\gamma = u_\gamma^* - u_\gamma \tag{5}$$

Define the γ-th sliding surface as:

$$s_\gamma = \tilde{x}_\gamma^{(n)} + \alpha_{\gamma 1} \tilde{x}_\gamma^{(n-1)} + \ldots + \alpha_{\gamma n} \tilde{x}_\gamma \tag{6}$$

Subtracting the γ-th equation of (1) from (3), it is obtained as:

$$\tilde{x}_\gamma^{(n)} + \alpha_{\gamma 1} \tilde{x}_\gamma^{(n-1)} + \ldots + \alpha_{\gamma n} \tilde{x}_\gamma = g_\gamma \left(u_\gamma^* - u_\gamma \right) \tag{7}$$

$$s_\gamma = g_\gamma \tilde{u}_\gamma \tag{8}$$

If the γ-th ideal controller is achieved $u_\gamma = u_\gamma^*$, or $\tilde{u}_\gamma = 0$, (8) is equivalent to

$$s_\gamma = 0 \tag{9}$$

It implies that:

$$\lim_{t \to \infty} \tilde{x}_\gamma = 0 \tag{10}$$

However, it is difficult to achieve the ideal controller u_γ^* because it is depended on the lumped uncertainty β_γ that cannot be measured or calculated. Thus, the proposed SO-DFL-BELC is designed to control the MIMO nonlinear systems in (1), which is expressed in detail in the following sections.

III. DESIGN THE SELF-ORGANIZING DOUBLE FUNCTION-LINK BRAIN EMOTIONAL LEARNING CONTROLLER

As mentioned above, the structure of the SO-DFL-BELC is actually the combination of the BELC with the DFL network and the SO mechanism. The architecture of the SO-DFL-BELC is present as follows:

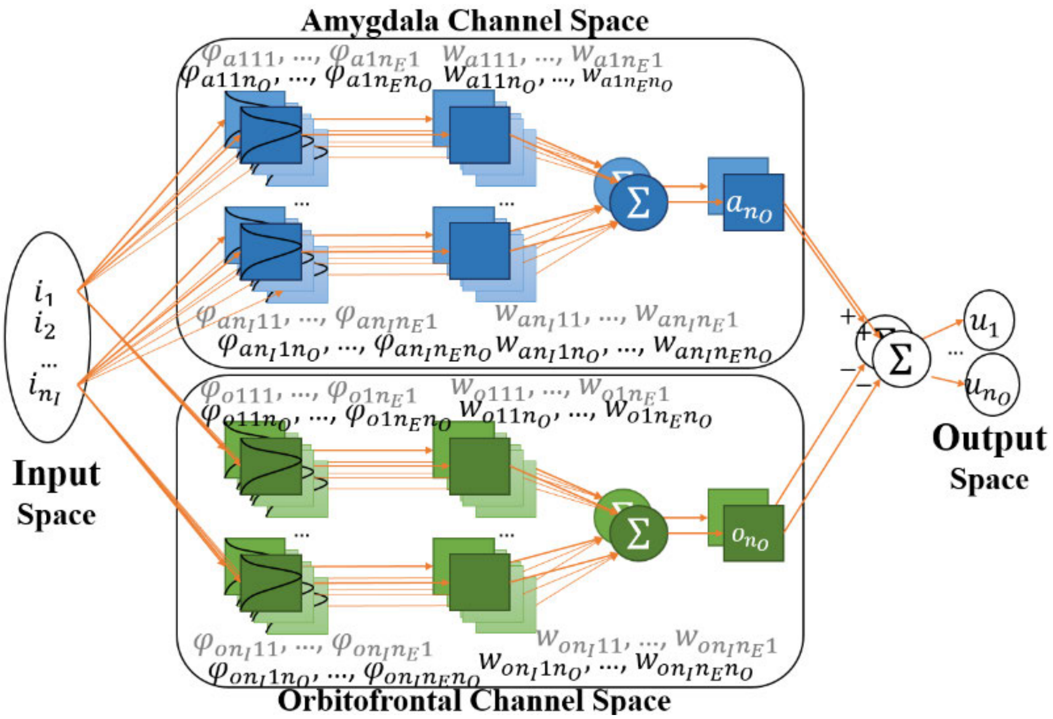
Amygdala Channel Space
φa111, ..., φa1nE1 wa111, ..., wa1nE1
φa11nO, ..., φa1nEnO wa11nO, ..., wa1nEnO
Σ
anO
φanI11, ..., φanInE1 wanI11, ..., wanInE1
φanI1nO, ..., φanInEnO wanI1nO, ..., wanInEnO
i1
i2
...
inI
Input
Space
+
−
Σ
u1
...
unO
Output
Space
φo111, ..., φo1nE1 wo111, ..., wo1nE1
φo11nO, ..., φo1nEnO wo11nO, ..., wo1nEnO
Σ
onO
φonI11, ..., φonInE1 wonI11, ..., wonInE1
φonI1nO, ..., φonInEnO wonI1nO, ..., wonInEnO
Orbitofrontal Channel Space

FIGURE 1. The architecture of the BELC.

A. THE BRAIN EMOTIONAL LEARNING CONTROLLER

The original BELC includes four main parts, which are the input space, the amygdala channel (AC) space, the orbitofrontal channel (OC) space, and the output space [10], [11], [16]:

➢ The input space:

The input of a BELC network is a continuous multi-dimensional signal. A given input signal is presented as (11), where N_I is the input dimension. The output of input space is sent directly to the AC and OC in a synchronous way.

$$\boldsymbol{i} = \left[i_1, i_2, \ldots, i_{N_I}\right]^T \tag{11}$$

➢ The amygdala channel space:

The AC is built based on the amygdala, which is the emotional part of the brain. In that space, firstly, the outputs of input space are mapped using Gaussian membership function for smooth transition as (12). Then, the output of the AC is the sum of all products of each output of the Gaussian function above with an amygdala weight as (13).

$$\varphi_{aij\gamma} = exp\left[-\frac{\left(i_i - m_{aij\gamma}\right)^2}{v_{aij\gamma}^2}\right] \tag{12}$$

$$a_\gamma = \sum_{i=1}^{N_I}\sum_{j=1}^{N_E} w_{aij\gamma}\varphi_{aij\gamma} \tag{13}$$

for $i = 1, 2, \ldots, N_I, j = 1, 2, \ldots, N_E$, and $\gamma = 1, 2, \ldots, N_o$, where N_E is the number of the neurons for each input, N_O is the number of the outputs of the controller, $m_{aij\gamma}, v_{aij\gamma}, w_{aij\gamma}$ are the mean parameter, the variance parameter and the weight parameter of AC, respectively.

➢ Orbitofrontal channel space:

Unlike the AC, the idea of OC is based on cognitive process to make a decision. However, the OC structure is nearly the same with the AC structure. It also uses the Gaussian membership function for smooth transition as (14) and calculates the output as (15):

$$\varphi_{oij\gamma} = exp\left[-\frac{\left(i_i - m_{oij\gamma}\right)^2}{v_{oij\gamma}^2}\right] \tag{14}$$

$$o_\gamma = \sum_{i=1}^{N_I}\sum_{j=1}^{N_E} w_{oij\gamma}\varphi_{oij\gamma} \tag{15}$$

for $i = 1, 2, \ldots, N_I, j = 1, 2, \ldots, N_E$, and $\gamma = 1, 2, \ldots, N_o$. $m_{oij\gamma}, v_{oij\gamma}, w_{oij\gamma}$ are the mean parameter, the variance parameter and the weight parameter of OC, respectively.

➢ The output space:

From the output of AC and OC, the outputs of the controller are formulated as:

$$u_\gamma = a_\gamma - o_\gamma \tag{16}$$

for $\gamma = 1, 2, \ldots, N_o$.

B. THE DOUBLE FUNCTION-LINK NETWORK

The DFL network is used to expand the functions for a BELC to improve the accuracy of the function approximation [14]–[17]. Fig. 2 illustrates how to

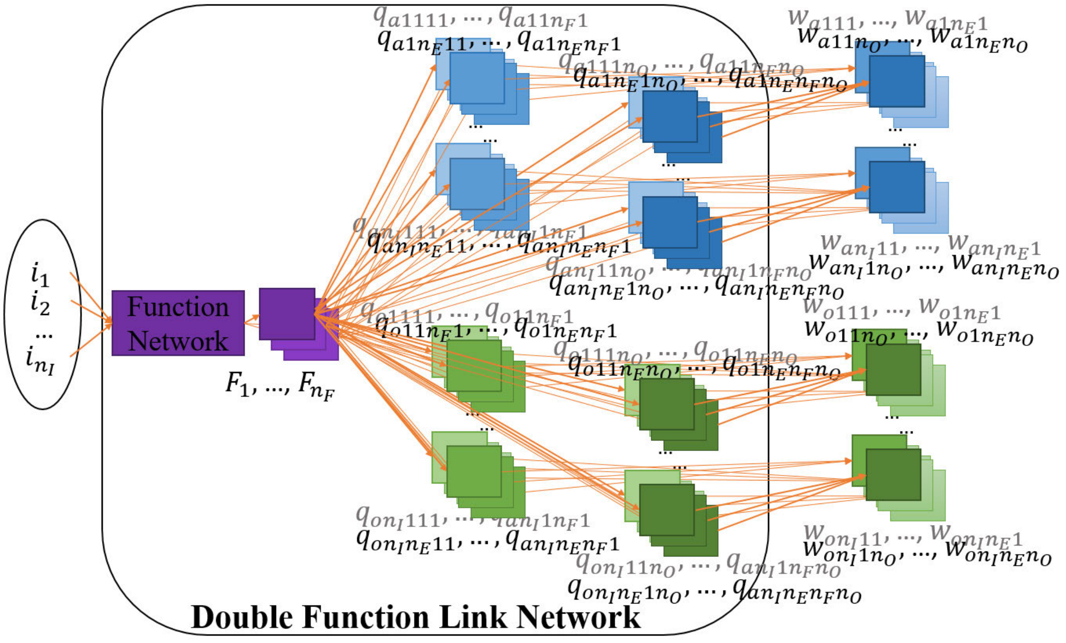
i_1
i_2
…
i_{n_I}
Function Network
$F_1, \ldots, F_{n_F}$
$q_{a1111}, \ldots, q_{a11n_F1}$
$q_{a1n_E11}, \ldots, q_{a1n_En_F1}$
$q_{a111n_O}, \ldots, q_{a11n_Fn_O}$
$q_{a1n_E1n_O}, \ldots, q_{a1n_En_Fn_O}$
$w_{a111}, \ldots, w_{a1n_E1}$
$w_{a11n_O}, \ldots, w_{a1n_En_O}$
$q_{an_I111}, \ldots, q_{an_I1n_F1}$
$q_{an_In_E11}, \ldots, q_{an_In_En_F1}$
$q_{an_I11n_O}, \ldots, q_{an_I1n_Fn_O}$
$q_{an_In_E1n_O}, \ldots, q_{an_In_En_Fn_O}$
$w_{an_I11}, \ldots, w_{an_In_E1}$
$w_{an_I1n_O}, \ldots, w_{an_In_En_O}$
$q_{o1111}, \ldots, q_{o11n_F1}$
$q_{o11n_F1}, \ldots, q_{o1n_En_F1}$
$q_{o111n_O}, \ldots, q_{o11n_Fn_O}$
$q_{o11n_Fn_O}, \ldots, q_{o1n_En_Fn_O}$
$w_{o111}, \ldots, w_{o1n_E1}$
$w_{o11n_O}, \ldots, w_{o1n_En_O}$
$q_{on_I111}, \ldots, q_{an_I1n_F1}$
$q_{on_In_E11}, \ldots, q_{an_In_En_F1}$
$q_{on_I11n_O}, \ldots, q_{an_I1n_Fn_O}$
$q_{on_In_E1n_O}, \ldots, q_{an_In_En_Fn_O}$
$w_{on_I11}, \ldots, w_{on_In_E1}$
$w_{on_I1n_O}, \ldots, w_{on_In_En_O}$
Double Function Link Network

FIGURE 2. The DFL network using for the BELC.

apply the double function-link network to the BELC. It is clear that the weights of the BELC are depended on the functions of the inputs. The functions can be chosen by the sine and cosine functions as $\boldsymbol{F} = \left[1, i_1, \sin(i_1), \cos(i_1), i_2, \sin(i_2), \cos(i_2), \ldots, F_{N_F}\right]$ [15]. Since the BELC has two different kinds of weights, which are the ones of the AC and the ones of the OC, hence there are the requirement of two function links. It is called as the double function link network. The outputs of the DFL network, which are also the weights of the BELC, are computed as:

$$w_{aij\gamma} = \sum_{k=1}^{N_F} q_{aijk\gamma} F_k \tag{17}$$

$$w_{oij\gamma} = \sum_{k=1}^{N_F} q_{oijk\gamma} F_k \tag{18}$$

for $i = 1, 2, \ldots, N_I, j = 1, 2, \ldots, N_E$, and $\gamma = 1, 2, \ldots, N_o$, where N_F is the number of expanded functions, $q_{aijk\gamma}$ and $q_{oijk\gamma}$ are the connecting weights of the DFL network for AC and OC, respectively.

C. THE SELF-ORGANIZING MECHANISM

In addition to the DFL network, the SO mechanism is also an effective method for neural networks to achieve a favorable performance. The key idea of the SO mechanism is to receive the information from all of the neurons overtime, and then it will evaluate when all of the neurons are not effective enough, so a new neuron should be generated, or when one of the old neurons are unnecessary, hence it should be deleted.

1) NEURON DECREASING PROCESS

Figure 3 presents the neuron decreasing process of the SO mechanism, which aims to delete the unnecessary neurons.

Step 1: Each neuron $\varphi_{aij\gamma}$ or $\varphi_{oij\gamma}$ is marked with an index $I_{aij\gamma}$ or $I_{oij\gamma}$ evaluating the importance of that neuron.

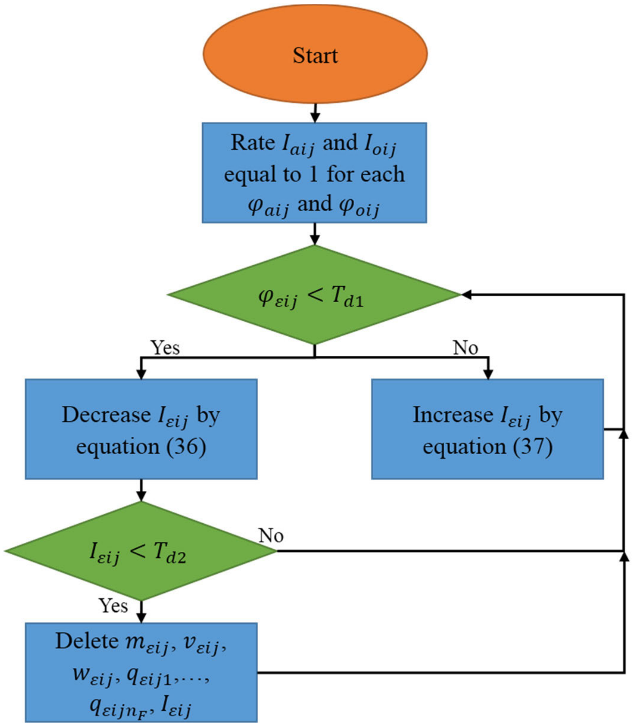
Start
Rate I_{aij} and I_{oij} equal to 1 for each φ_{aij} and φ_{oij}
$\varphi_{\varepsilon ij} < T_{d1}$
Yes
No
Decrease $I_{\varepsilon ij}$ by equation (36)
Increase $I_{\varepsilon ij}$ by equation (37)
$I_{\varepsilon ij} < T_{d2}$
No
Yes
Delete $m_{\varepsilon ij}$, $v_{\varepsilon ij}$, $w_{\varepsilon ij}$, $q_{\varepsilon ij1}, \ldots,$ $q_{\varepsilon ijn_F}$, $I_{\varepsilon ij}$

FIGURE 3. The neuron decreasing process.

Step 2: At each time t, that index is updated based on the value of $\varphi_{\varepsilon ij\gamma}$. It is clear that if the value of $\varphi_{\varepsilon ij\gamma}$ is small and less than a threshold T_{d1}, it has little meaning, thus the index of it $I_{\varepsilon ij\gamma}$ should be decreased. The smaller the value of $\varphi_{\varepsilon ij\gamma}$ is, the faster its index is reduced to zero, and vice versa. Otherwise, if the value of $\varphi_{\varepsilon ij\gamma}$ is large enough and bigger than that threshold T_{d1}, which means that neuron has a great significance, the index belonging to it will be increased to one. Similarly, the bigger the value of $\varphi_{\varepsilon ij\gamma}$ is, the more quickly its index raises to one, and vice versa.

Step 3: Also at each time t, if the index $I_{\varepsilon ij\gamma}$ becomes lower than a threshold T_{d2}, it means that $I_{\varepsilon ij\gamma}$ describing the importance of the node $\varphi_{\varepsilon ij\gamma}$ is decreased again and again for a long time, so the neuron and its related weight will be clear immediately.

The updating rule of index $I_{\varepsilon ij\gamma}$ is presented as:

$$I_{\varepsilon ij\gamma}\,(\mathrm{t}+1) = I_{\varepsilon ij\gamma}\,(t)\,exp\left(-\frac{1-\varphi_{\varepsilon ij\gamma}}{\tau_1}\right) \quad \text{if } \varphi_{\varepsilon ij\gamma} < T_{d1} \tag{19}$$

$$I_{\varepsilon ij\gamma}\,(\mathrm{t}+1) = I_{\varepsilon ij\gamma}\,(t)\left\{2-exp\left[-\frac{\varphi_{\varepsilon ij\gamma}(1-I_{\varepsilon ij\gamma}\,(t))}{\tau_2}\right]\right\} \quad \text{if } \varphi_{\varepsilon ij\gamma} \geq T_{d1} \tag{20}$$

for $i = 1, 2, \ldots, N_I, j = 1, 2, \ldots, N_E$, and $\gamma = 1, 2, \ldots, N_o$, where ε is a for AC or o for OC.

2) NEURON INCREASING PROCESS

Beside the neuron decreasing process, there is also the increasing process to generate new neurons. This process works based on the calculation of all of the neurons to check the maximum value of them. If that maximum value is smaller than a threshold T_i, it indicates that all of the current neurons are not sensitive to the current input data, in other words, the current neurons are all ineffective. Therefore, a new neuron corresponding to the input data should be generated at that time. By that way, the neuron network can be adapted to the input data, though the input data probably has a sharp change overtime. The process of neuron increasing is illustrated in Fig. 4 and the rules to add a new neuron of are shown as (21)-(25):

$$m_{\gamma\varepsilon i} = [m_{\gamma\varepsilon i}; i_i] \tag{21}$$
$$v_{\gamma\varepsilon i} = [v_{\gamma\varepsilon i}; \bar{v}_{\gamma\varepsilon i}] \tag{22}$$
$$\varphi_{\gamma\varepsilon i} = [\varphi_{\gamma\varepsilon i}; 1] \tag{23}$$
$$q_{\gamma\varepsilon i} = [q_{\gamma\varepsilon i}; A] \tag{24}$$
$$I_{\gamma\varepsilon i} = [I_{\gamma\varepsilon i}; 1] \tag{25}$$

for $i = 1, 2, \ldots, N_I$ and $\gamma = 1, 2, \ldots, N_o$, where ε is a for AC or o for OC, $[O; o]$ means to add a new element o into O if O is a vector or to add a new column into O if O is a matrix, $m_{\varepsilon i\gamma} = \left[m_{\varepsilon i1\gamma}, m_{\varepsilon i2\gamma}, \ldots, m_{\varepsilon iN_E\gamma}\right]$, $v_{\varepsilon i\gamma} = \left[v_{\varepsilon i1\gamma}, v_{\varepsilon i2\gamma}, \ldots, v_{\varepsilon iN_E\gamma}\right]$, $\varphi_{\varepsilon i\gamma} = \left[\varphi_{\varepsilon i1\gamma}, \varphi_{\varepsilon i2\gamma}, \ldots, \varphi_{\varepsilon iN_E\gamma}\right]$,

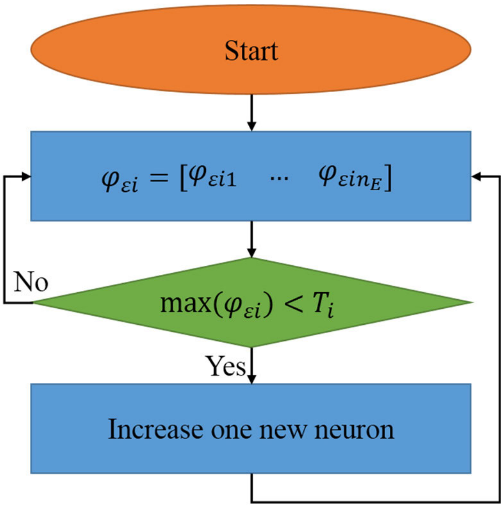
Start
$\varphi_{\varepsilon i} = [\varphi_{\varepsilon i1} \;\cdots\; \varphi_{\varepsilon in_E}]$
$\max(\varphi_{\varepsilon i}) < T_i$
No
Yes
Increase one new neuron

FIGURE 4. Neuron increasing process.

$I_{\varepsilon i\gamma} = \left[I_{\varepsilon i1\gamma}, I_{\varepsilon i2\gamma}, \ldots, I_{\varepsilon iN_E\gamma}\right]$,

$$q_{\varepsilon i\gamma} = \begin{bmatrix} q_{\varepsilon i11\gamma} & q_{\varepsilon i12\gamma} & \cdots & q_{\varepsilon i1N_F\gamma} \\ q_{\varepsilon i21\gamma} & q_{\varepsilon i22\gamma} & \cdots & q_{\varepsilon i2N_F\gamma} \\ \vdots & \vdots & \ddots & \vdots \\ q_{\varepsilon iN_E1\gamma} & q_{\varepsilon iN_E2\gamma} & \cdots & q_{\varepsilon iN_EN_F\gamma} \end{bmatrix},$$

$\bar{v}_{\varepsilon i\gamma}$ is the average value of $v_{\varepsilon i\gamma}$, $A = [\underbrace{0\;0\;\ldots\;0}_{N_F \text{ number } 0}]$.

D. UPDATING RULES OF THE SO-DFL-BELC

1) UPDATING RULES FOR THE WEIGHTS

The updating rules of the weights of the SO-DFL-BELC are determined as:

$$d_{q\gamma} = c_{q\gamma}u_\gamma + b_{q\gamma}s_\gamma \tag{26}$$
$$\Delta q_{aijk\gamma} = \alpha_a F_k \varphi_{aij\gamma} max(0, d_{q\gamma} - a_\gamma) \tag{27}$$
$$\Delta q_{oijk\gamma} = \alpha_o F_{k\gamma} \varphi_{oij\gamma}(u_\gamma - d_{q\gamma}) \tag{28}$$

for $i = 1, 2, \ldots, N_I, j = 1, 2, \ldots, N_E, k = 1, 2, \ldots, N_F$, and $\gamma = 1, 2, \ldots, N_o$, where $d_{q\gamma}$ is called as the reward function, reward signal or objective function [31]–[33], $b_{q\gamma}$ and $c_{q\gamma}$ are the positive gains, α_a and α_o are the learning rates of the weight parameters of AC and OC, respectively.

Normally, the reward function $d_{q\gamma}$ is chosen as the sum of the control signal u_γ and a function of the error of the system $\tilde{x}_\gamma$ [31], [32]. Since the sliding surface s_γ is also a function of $\tilde{x}_\gamma$ as (8), this paper proposes a novel reward function $d_{q\gamma}$ as (26).

2) UPDATING RULES FOR THE MEANS AND THE VARIANCES

The updating rules of the means and the variances of the SO-DFL-BELC are built based on the gradient descent algorithm. The cost function E_γ is chosen as:

$$E_\gamma = \frac{1}{2}s_\gamma^2 \tag{29}$$

for $\gamma = 1, 2, \ldots, N_o$.

There are two reasons why the cost function E_γ is chosen as (29). The first reason is that the aim of the gradient descent is to minimize the cost function E_γ. It means that the cost function E_γ here will be decreased to approximately zero, that makes the sliding surface s_γ come to nearly zero, too. It implies $\tilde{x}_\gamma = 0$ as presented above. The second reason is that it is easier to compute the derivative of the cost function E_γ with respect to the means and variances of the SO-DFL-BELC than when choosing the cost function as $E_\gamma = \frac{1}{2}\tilde{x}_\gamma^2$. It is based on the dynamic equation (8) that it is possible to calculate the derivative of s_γ with respect to $\tilde{u}_\gamma$. Also, the derivative of E_γ with respect to s_γ, the derivative of $\tilde{u}_\gamma$ with respect to u_γ, and the derivative of u_γ with respect to the means and variances of the SO-DFL-BELC can be calculated. Hence, the updating rules are given as:

$$\Delta m_{aij\gamma} = \eta_{m_a} g_\gamma s_\gamma w_{\gamma aij} \varphi_{\gamma aij} \frac{\left(i_{\gamma i} - m_{\gamma aij}\right)}{v_{\gamma aij}^2} \tag{30}$$

$$\Delta m_{\gamma oij} = -\eta_{m_o} g_\gamma s_\gamma w_{\gamma oij} \varphi_{\gamma oij} \frac{\left(i_{\gamma i} - m_{\gamma oij}\right)}{v_{\gamma oij}^2} \tag{31}$$

$$\Delta v_{\gamma aij} = \eta_{v_a} g_\gamma s_\gamma w_{\gamma aij} \varphi_{\gamma aij} \frac{\left(i_{\gamma i} - m_{\gamma aij}\right)^2}{v_{\gamma aij}^3} \tag{32}$$

$$\Delta v_{\gamma oij} = -\eta_{v_o} g_\gamma s_\gamma w_{\gamma oij} \varphi_{\gamma oij} \frac{\left(i_{\gamma i} - m_{\gamma oij}\right)^2}{v_{\gamma oij}^3} \tag{33}$$

for $i = 1, 2, \ldots, N_I, j = 1, 2, \ldots, N_E$, and $\gamma = 1, 2, \ldots, N_o$, where η_{m_a}, η_{m_o}, η_{v_a}, η_{v_o} are the learning rates of the means and the variances of the AC and the OC, respectively.

E. STRUCTURE OF THE PROPOSED CONTROLLER

Fig. 5 illustrates the structure of the SO-DFL-BELC.

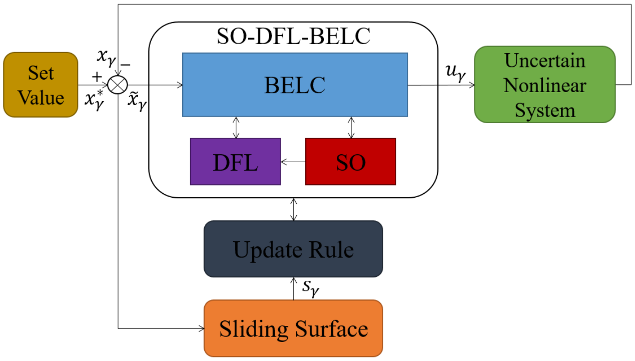

FIGURE 5. **Structure for online learning of the SO-DFL-BELC.**

F. STABILITY ANALYSIS

A Lyapunov function is given as

$$V(t) = \frac{1}{2} s_\gamma^2 \tag{34}$$

The change of the Lyapunov function is determined by

$$\begin{aligned} \Delta V(t) &= V(t+1) - V(t) = \frac{1}{2}[\left(s_\gamma^2(t+1)\right) - s_\gamma^2(t)] \\ &= \Delta s_\gamma(t) \cdot [\frac{1}{2}\Delta s_\gamma(t) + s_\gamma(t)] \end{aligned} \tag{35}$$

The error difference is calculated as

$$s_\gamma(t+1) = s_\gamma(t) + \Delta s_\gamma(t) \cong s_\gamma(t) + \left[\frac{\partial s_\gamma(t)}{\partial z}\right] \Delta z \tag{36}$$

where $\Delta s_\gamma(t)$ is the change of sliding surface and Δz is the change of z; z can be $q_{aijk\gamma}, q_{oijk\gamma}, m_{\gamma\varepsilon i}, or\ v_{\gamma\varepsilon i}.$

Note that $\Delta z(t+1) = z(t) + z(t) = z(t) - \eta_z \left[\frac{\partial E_\gamma}{\partial u_\gamma}\right] T_z(t)$ and $T_z(t) = \frac{\partial u_\gamma}{\partial z}$

Thus,

$$\Delta z(t) = -\eta_z \frac{\partial E_\gamma}{\partial z} = \eta_z s_\gamma(t) \frac{\partial u_\gamma}{\partial z} = \eta_z s_\gamma(t) T_z(t) \tag{37}$$

In addition,

$$\frac{\delta s_\gamma}{\delta z} = \frac{\delta s_\gamma}{\delta u_\gamma} \frac{\delta u_\gamma}{\delta z} = -g_\gamma T_z(t) \tag{38}$$

Substituting (37) and (38) into (35), gives

$$\begin{aligned} \Delta V(t) &= \Delta s_\gamma(t) \left[\frac{1}{2}\Delta s_\gamma(t) + s_\gamma(t)\right] \\ &= \left[\frac{\partial s_\gamma(t)}{\partial z}\right] \Delta z \left[\frac{1}{2}\left[\frac{\partial s_\gamma(t)}{\partial z}\right] \Delta z + s_\gamma(t)\right] \\ &= -g_\gamma T_z(t)\, \eta_z s_\gamma(t)\, T_z(t) \\ &\quad \times \left[-\frac{1}{2} g_\gamma T_z(t)\, \eta_z s_\gamma(t)\, T_z(t) + s_\gamma(t)\right] \end{aligned} \tag{39}$$

Therefore,

$$\Delta V(t) = \frac{1}{2}\eta_z \left(g_\gamma\right)^2 s_\gamma^2 \left\|T_z(t)\right\|^2 \left[\eta_z \left\|T_z(t)\right\|^2 - 2\right] \tag{40}$$

If η_z is selected as

$$0 < \eta_z < \frac{2}{\left\|T_z(t)\right\|^2} \tag{41}$$

Then $\Delta V(t) < 0$. It implies that the control system is stable.

Note that the optimal learning rates that can achieve the fastest convergence correspond to

$$\eta_z^* = \frac{1}{\left\|T_z(t)\right\|^2} \tag{42}$$

This comes from taking the derivative of (40) and let it equals to zero.

IV. USING SO-DFL-BELC TO CONTROL A 4D CHAOTIC SYSTEM

A. DESCRIPTION OF A 4D CHAOTIC SYSTEM

Consider the 4D Lorenz-Stenflo chaotic system used in [34] as follows:

- The master system:

$$\dot{x}_m = \zeta\left(y_m - x_m\right) + \gamma w_m \tag{43}$$

$$\dot{y}_m = \tau x_m - x_m z_m - y_m \tag{44}$$

$$\dot{z}_m = x_m y_m - \delta z_m \tag{45}$$

$$\dot{w}_m = -x_m - \lambda w_m \tag{46}$$

- The slave system:

$$\dot{x}_s = \zeta\left(y_s - x_s\right) + \gamma w_s + u_x + \beta_x \tag{47}$$

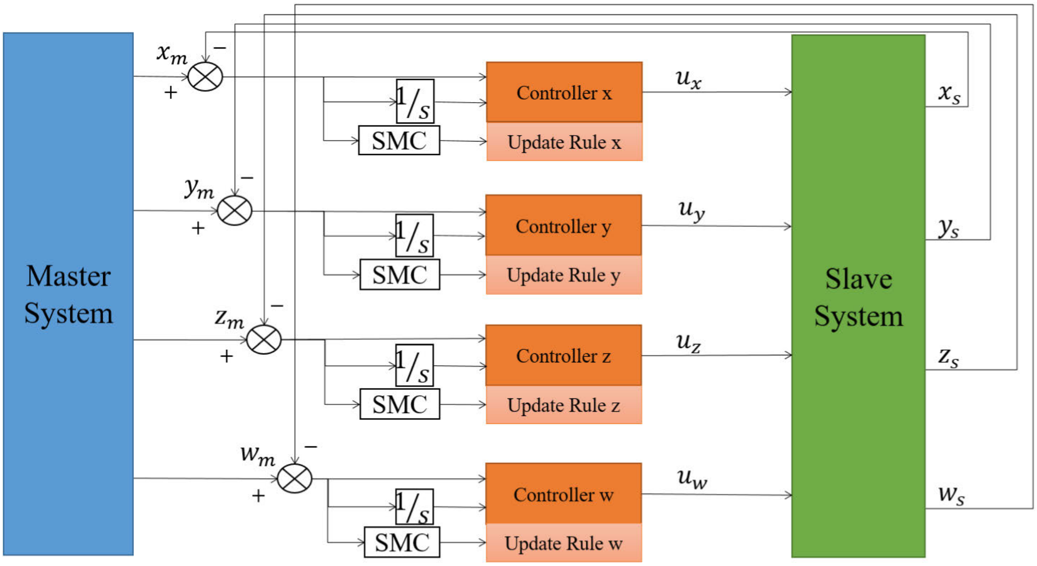

FIGURE 6. The block diagram of the 4D chaotic system.

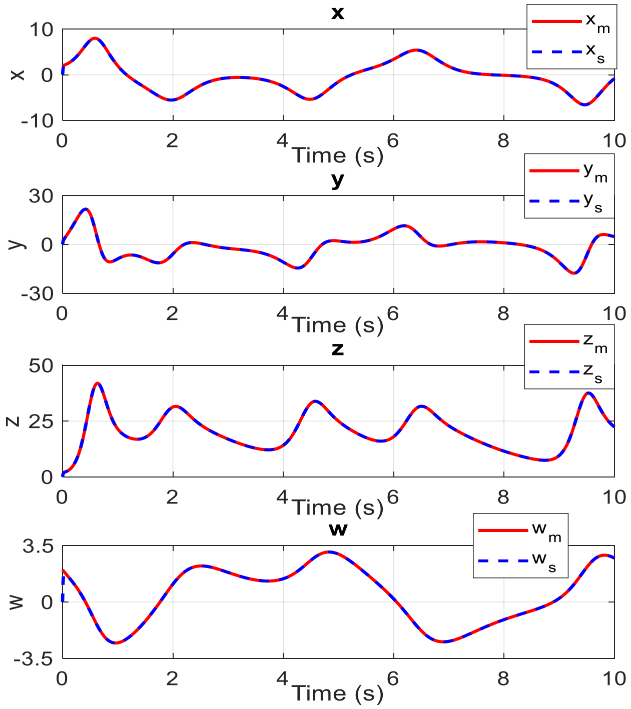

FIGURE 7. The trajectory and synchronization results.

$$\dot{y}_s = \tau x_s - x_s z_s - y_s + u_y + \beta_y \tag{48}$$

$$\dot{z}_s = x_s y_s - \delta z_s + u_z + \beta_z \tag{49}$$

$$\dot{w}_s = -x_s - \lambda w_s + u_w + \beta_w \tag{50}$$

where $\zeta, \tau, \lambda, \delta, \gamma$ are the gains, u_x, u_y, u_z, u_w are the control signals, $\beta_x, \beta_y, \beta_z, \beta_w$ are the sum of noises and disturbances that:

$$\beta_x = r_x \left[\zeta \left(y_s - x_s\right) + \gamma w_s + u_x\right] \tag{51}$$

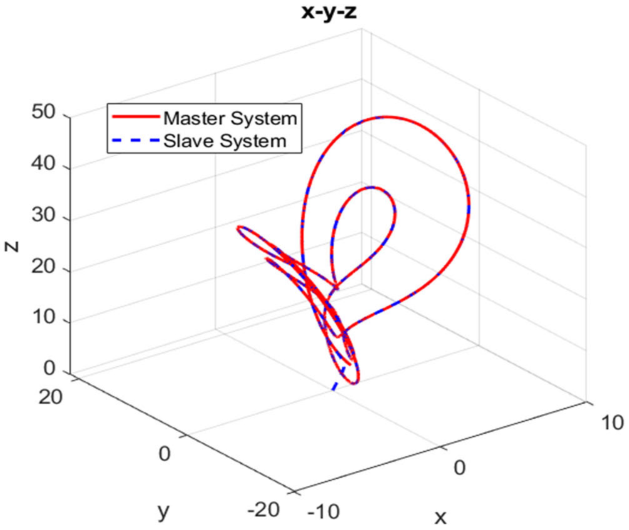

FIGURE 8. Projection of the synchronization in the 3D x-y-z space.

$$\beta_y = r_y \left[\tau x_s - x_s z_s - y_s + u_y\right] \tag{52}$$

$$\beta_z = r_z \left[x_s y_s - \delta z_s + u_z\right] \tag{53}$$

$$\beta_w = r_w \left[-x_s - \lambda w_s + u_w\right] \tag{54}$$

where r_x, r_y, r_z, r_w are the noise ratios which obey the Gaussian distribution with the mean is zero and the variance is 0.05.

The purpose of the controller is to make the outputs of the slave system x_s, y_s, z_s, w_s follow to the output of the master system x_m, y_m, z_m, w_m closely.

B. CONTROLLER DESIGN

The block diagram of the 4D chaotic system is shown in Fig. 6.

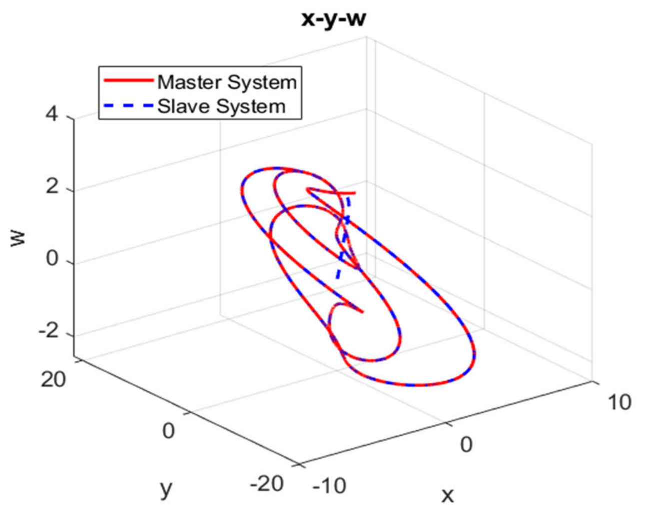

FIGURE 9. Projection of the synchronization in the 3D x-y-w space

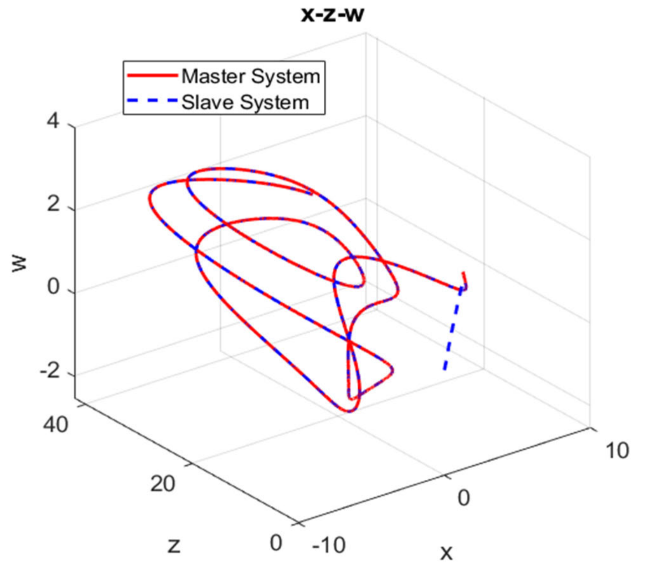

FIGURE 10. Projection of the synchronization in the 3D x-z-w space.

The parameters of the system are given as follow: The value of gains are $\zeta = 1, \tau = 26, \lambda = 1, \delta = 0.7$, and $\gamma = 1.5$. The initial positions of the master and the slave system are $x_m = 2, y_m = 2, z_m = 2, w_m = 2$ and $x_s = 0, y_s = 0, y_s = 0, y_s = 0$, respectively.

The parameters of the controller are initially chosen as follows: The learning rates of weights η_m and variances η_v are both chosen as 10^{-5}. The learning rates of weights α_a and α_o are both selected as 5×10^{-9}. The reward function parameters b_q and c_q are both chosen as 1. The parameters of the SO mechanism are set as $T_i = 0.5, T_{d1} = 0.01, T_{d2} = 0.1, \tau_1 = 100, \tau_2 = 10$. The parameters of sliding surface are determined as $\alpha_{\gamma 1} = 100$ for $\gamma = 1, 2, 3$, and 4. Firstly, all of the parameters are selected by trial and error to guarantee the desired control performance, then, they are adjusted online using the updating rules via (26)-(28), (30)-(33), and the learning rates can achieve the optimal values by (42). Also, in this example, the initial neurons for four controllers do not need to be designed in advance, because it can self-organize to a suitable value. In this study, the initial neurons are chosen as 38 for each controller u_x, u_y, u_z, u_w (see Fig. 17).

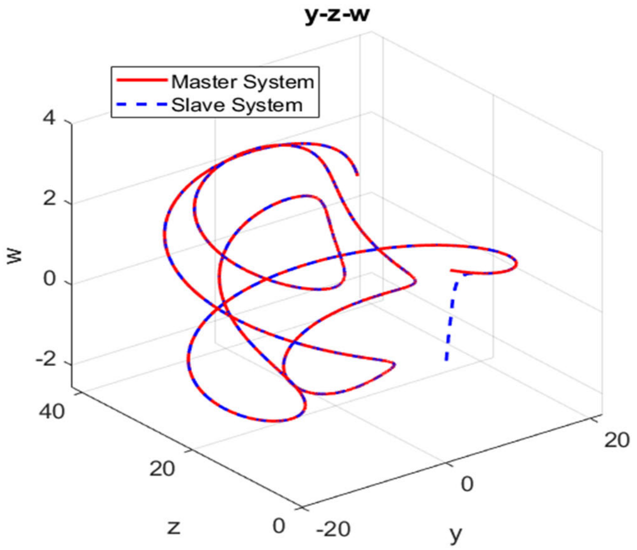

FIGURE 11. Projection of the synchronization results in the 3D y-z-w space.

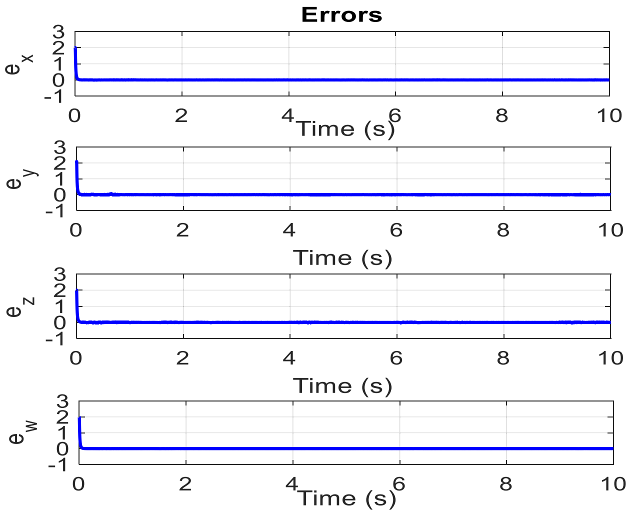

FIGURE 12. The errors of the system.

C. SIMULATION RESULTS

The simulation results about using SO-DFL-BELC to control the system are described as Figs. 7-17 while the comparison between different models are shown as Figs. 18-21 and Table 1.

From Figs. 18-21 and Table 1, the different controllers are compared when using the same noise ratio data as Fig. 16. Note that "BELC" implies using the conventional BELC [10], [11], "BELC updating m and v" is the

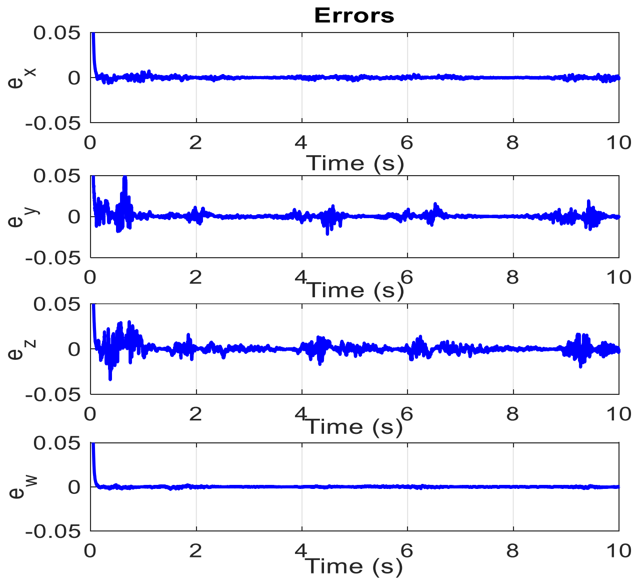

FIGURE 13. The enlarged errors of the system.

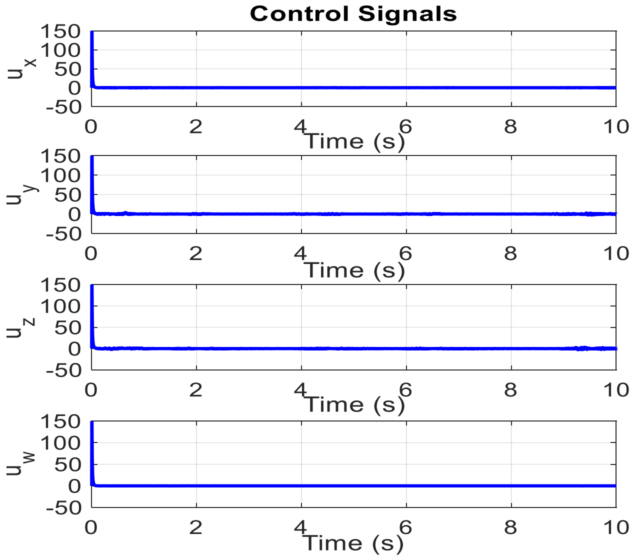

FIGURE 14. The control signals.

TABLE 1. Comparision in root mean square error for different controllers.

Model	RMSE
Conventional BELC	2.2575
BELC updating $\boldsymbol{m}$ and $\boldsymbol{v}$	1.4475
BELC using sliding surface	1.0586
DFL-BELC	0.0333
SO-DFL-BELC	**0.0324**

conventional BELC but the means and variances are updated overtime based on the gradient descent algorithm with $E_\gamma = \frac{1}{2}\tilde{x}_\gamma^2$ and $\frac{\partial E_\gamma}{\partial u_\gamma} = -\left(\tilde{x}+\dot{\tilde{x}}\right)$ as [35], [36], "BELC using sliding surface" means the BELC with all means, variances, and weights are updated overtime with the updating rules using sliding surface as mentioned above. "DFL-BELC" is "BELC using sliding surface" with the DFL network. "SO-DFL-BELC" is "DFL-BELC" with the SO mechanism, which is

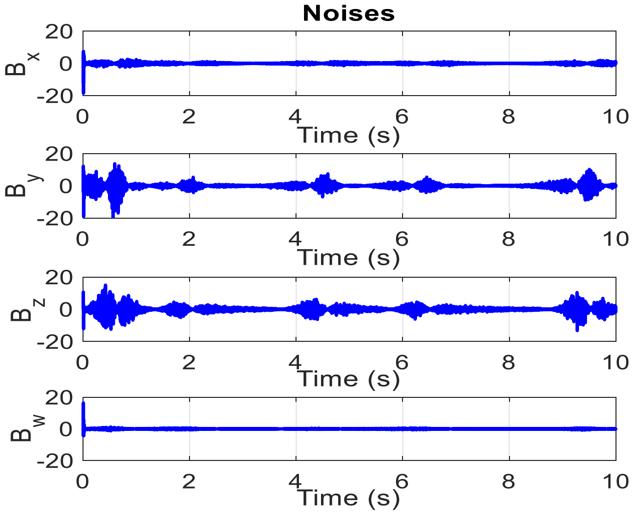

FIGURE 15. The noises of the system.

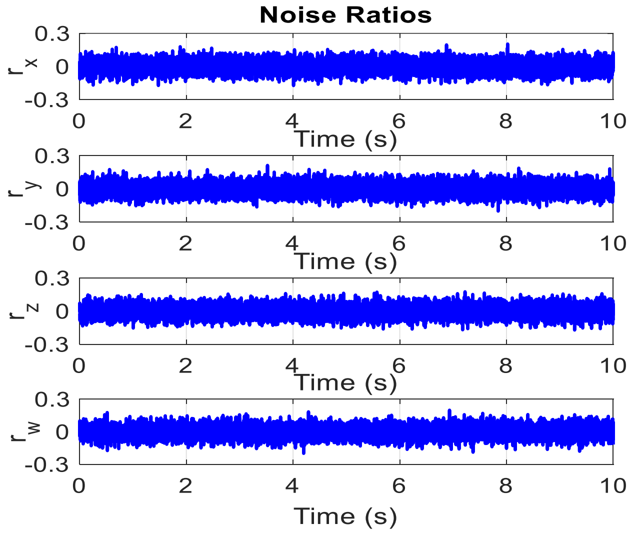

FIGURE 16. The noise ratios of the system.

also the controller introduced in section III. The root mean square error (RMSE) shown in Table 1 is calculated as the average value of the RMSE of x, y, z, and w.

D. COMMENTS OF THE SIMULATION RESULTS

Figures 7-13 present how good the controller is with the rise time is less than $0.05s$, and there is no overshoot. From Fig.13, it is clear that most of the time, the range of errors of the system is from -0.025 to 0.025 only. Figs. 15 and 16 illustrate the effect of noises to the system. It is clear that the noises will randomly increase or decrease the state of the system in the range from -20% to 20%. Fig. 17 describes how the SO mechanism affects the number of neurons for each output. Figs. 18-21 illustrate the comparison between the different controllers. It is clear that the tracking trajectories, the errors, and the control signals for the proposed SO-DFL-BELC

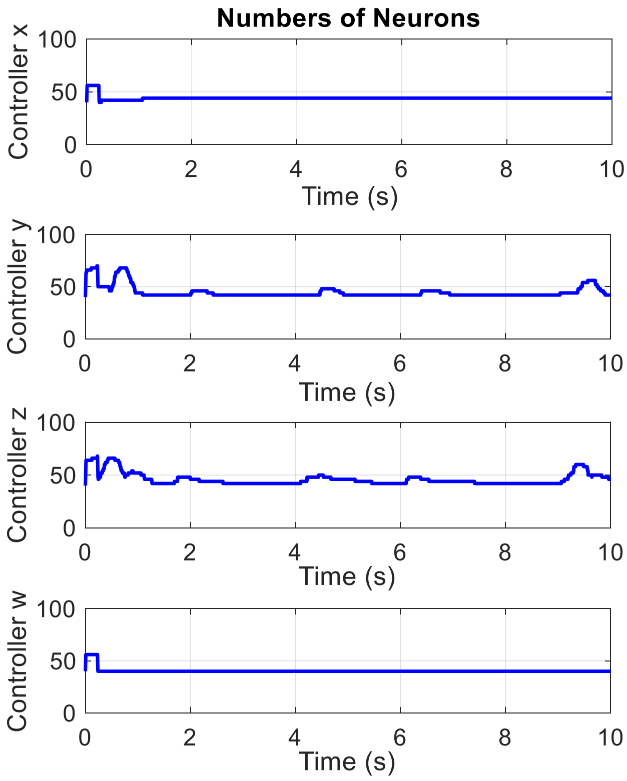

FIGURE 17. The neuron adjustment using the self-organizing mechanism.

achieve better performance than that for the conventional BELC [7], [8], the BELC updating m and v, the BELC using sliding surface, and the DFL-BELC. The RMSE result is also improved for the methods using the updating means and variances, the sliding surface for the updating rules, the DFL network, and the SO mechanism. Especially, when using the DFL network and the SO mechanism, the RMSE result is improved significantly.

V. USING SO-DFL-BELC TO CONTROL A FOUR-TANK SYSTEM

A. DESCRIPTION OF FOUR-TANK SYSTEM

Consider the four-tank system is given in [30] as follows:

$$\dot{h}_1 = -\frac{a_1}{S_1}\sqrt{2g_m h_1} + \frac{a_3}{S_1}\sqrt{2g_m h_3} + \frac{\gamma_a}{S_1}q_a + \beta_1 \quad (55)$$

$$\dot{h}_2 = -\frac{a_2}{S_2}\sqrt{2g_m h_2} + \frac{a_4}{S_2}\sqrt{2g_m h_4} + \frac{\gamma_b}{S_2}q_b + \beta_2 \quad (56)$$

$$\dot{h}_3 = -\frac{a_3}{S_3}\sqrt{2g_m h_3} + \frac{1-\gamma_b}{S_3}q_b + \beta_3 \quad (57)$$

$$\dot{h}_4 = -\frac{a_4}{S_4}\sqrt{2g_m h_4} + \frac{1-\gamma_a}{S_4}q_a + \beta_4 \quad (58)$$

where g_m is the acceleration due to gravity, h_1, h_2, h_3, h_4 are the water levels of the four tanks, a_1, a_2, a_3, a_4 represent the areas of inner diameter of pipes for tanks, S_1, S_2, S_3, S_4 describes the areas of tanks, $\beta_1, \beta_2, \beta_3, \beta_4$ mean the total noises and disturbances which affect to tank 1, 2, 3 and 4, respectively:

$$\beta_1 = r_1\left[-\frac{a_1}{S_1}\sqrt{2g_m h_1} + \frac{a_3}{S_1}\sqrt{2g_m h_3} + \frac{\gamma_a}{S_1}q_a\right] \quad (59)$$

$$\beta_2 = r_2\left[-\frac{a_2}{S_2}\sqrt{2g_m h_2} + \frac{a_4}{S_2}\sqrt{2g_m h_4} + \frac{\gamma_b}{S_2}q_b\right] \quad (60)$$

$$\beta_3 = r_3\left[-\frac{a_3}{S_3}\sqrt{2g_m h_3} + \frac{1-\gamma_b}{S_3}q_b\right] \quad (61)$$

$$\beta_4 = r_4\left[-\frac{a_4}{S_4}\sqrt{2g_m h_4} + \frac{1-\gamma_a}{S_4}q_a\right] \quad (62)$$

where r_1, r_2, r_3, r_4 are the noise ratios which obey the Gaussian distribution with the mean is zero and the variance is 0.05. Again, it means that the noises here can affect the system randomly.

The purpose of the controller is to control the levels of the tanks. However, it is clear that only the volume of water following to a tank can be controlled while the volume of water discharging from a tank is uncontrollable. It means that

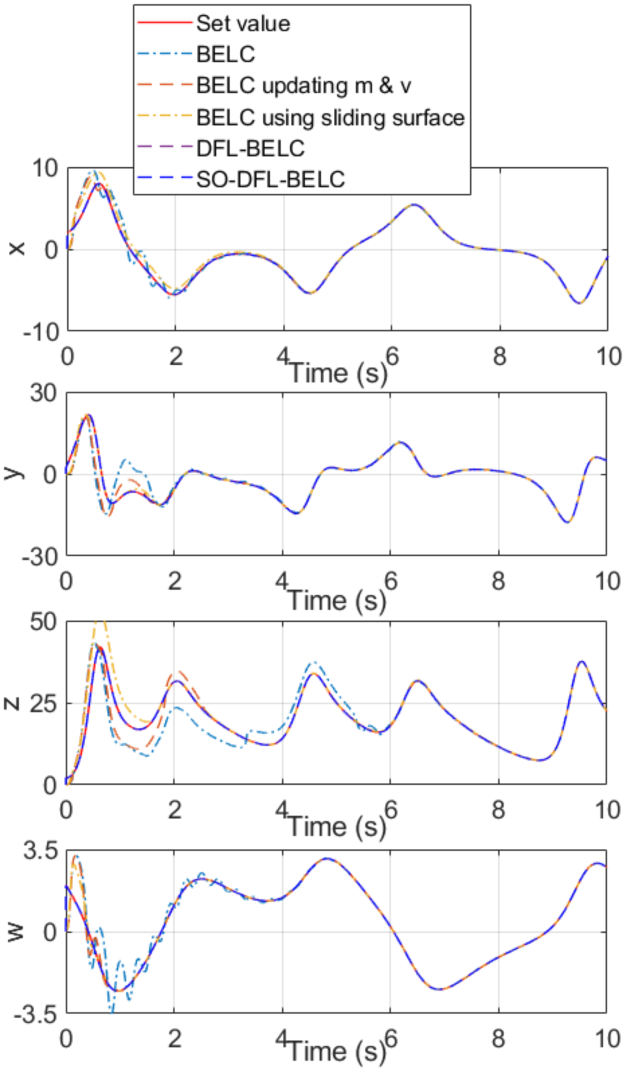

FIGURE 18. The comparison in the trajectory results.

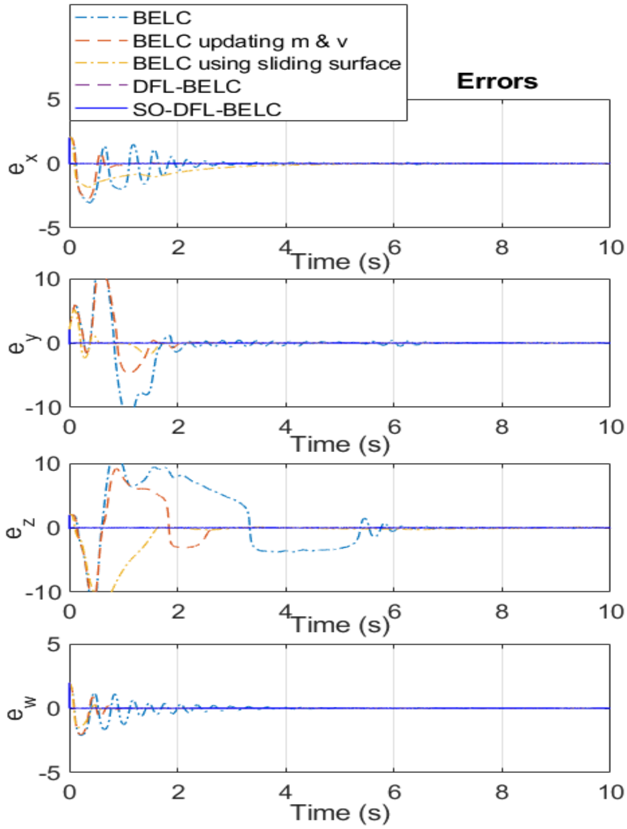

FIGURE 19. The comparison for the errors.

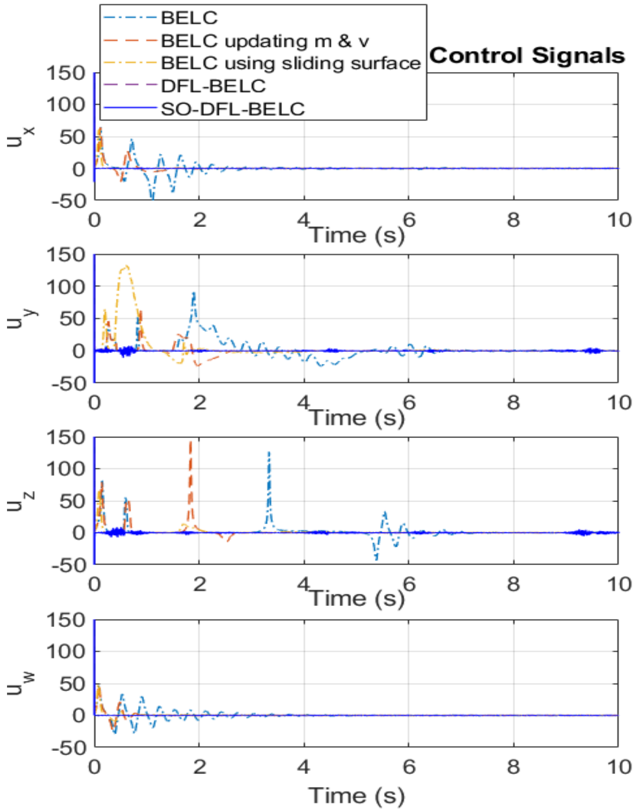

FIGURE 20. The comparison for the control signals.

the purpose of the controller is to increase or remain the level of the tank. If the level of the tank is set to be decreased, the only way to implement it is to wait for a volume of water to be discharged out of the tank.

B. DESIGN THE CONTROLLER

Define the errors of the water levels of the four tanks as:

$$\tilde{h}_1 = h_1^* - h_1 \tag{63}$$
$$\tilde{h}_2 = h_2^* - h_2 \tag{64}$$
$$\tilde{h}_3 = h_3^* - h_3 \tag{65}$$
$$\tilde{h}_4 = h_4^* - h_4 \tag{66}$$

where $\tilde{h}_1, \tilde{h}_2, \tilde{h}_3$ and $\tilde{h}_4$ are the errors of the water levels of tanks 1, 2, 3, and 4, respectively; h_1^*, h_2^*, h_3^*, and h_4^* are the set values of the water levels of tanks 1, 2, 3, and 4, respectively. Adding (55) to (58) yields (67), subtracting (58) from (55) yields (68):

$$\dot{h}_1 + \dot{h}_4 = f_1 + g_1 q_a \tag{67}$$
$$\dot{h}_1 - \dot{h}_4 = f_2 + g_2 \gamma_a \tag{68}$$

with f_1, g_1, f_2, g_2 as:

$$f_1 = -\frac{a_1}{S_1}\sqrt{2g_m h_1} + \frac{a_3}{S_1}\sqrt{2g_m h_3} - \frac{a_4}{S_4}\sqrt{2g_m h_4} + \beta_1 + \beta_4 \tag{69}$$

$$g_1 = \frac{\gamma_a}{S_1} + \frac{1-\gamma_a}{S_4} \tag{70}$$

$$f_2 = -\frac{a_1}{S_1}\sqrt{2g_m h_1} + \frac{a_3}{S_1}\sqrt{2g_m h_3} + \frac{a_4}{S_4}\sqrt{2g_m h_4} - \frac{q_a}{S_4} + \beta_1 - \beta_4 \tag{71}$$

$$g_2 = \frac{q_a}{S_1} + \frac{q_a}{S_4} \tag{72}$$

From (67), it is clear that the control signal q_a can control the sum of the h_1 and h_4, and it can let the sum of the errors of h_1 and h_4 come to zero. Similarly, (68) shows that the control signal γ_a probably makes the difference between the errors of h_1 and h_4 reach to zero:

$$\lim_{t\to\infty}\left(\tilde{h}_1 + \tilde{h}_4\right) = 0 \tag{73}$$
$$\lim_{t\to\infty}\left(\tilde{h}_1 - \tilde{h}_4\right) = 0 \tag{74}$$

Both (73) and (74) only happen when the errors of h_1 and h_4 approach to zero:

$$\lim_{t\to\infty}\tilde{h}_1 = \lim_{t\to\infty}\tilde{h}_4 = 0 \tag{75}$$

It is implemented with the same way for h_2, h_3, q_b, γ_b. Thus, the sliding mode surfaces for updating rules of the SO-DFL-BELC, which outputs the control signal q_a, q_b, γ_a, γ_b, are defined as:

$$s_{q_a} = \left(\tilde{h}_1 + \tilde{h}_4\right) + \alpha_1\left(\dot{\tilde{h}}_1 + \dot{\tilde{h}}_4\right) \tag{76}$$

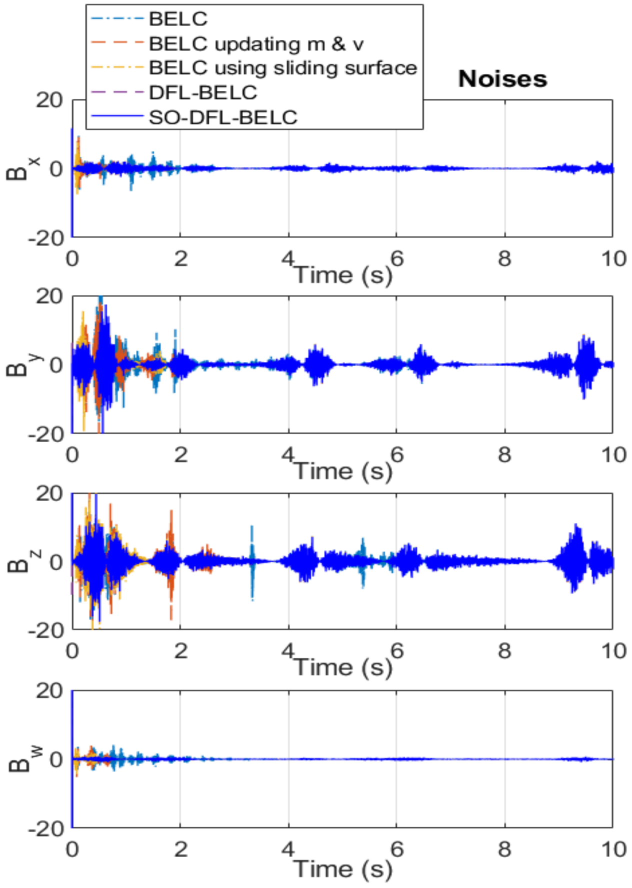

FIGURE 21. The comparison for the noises.

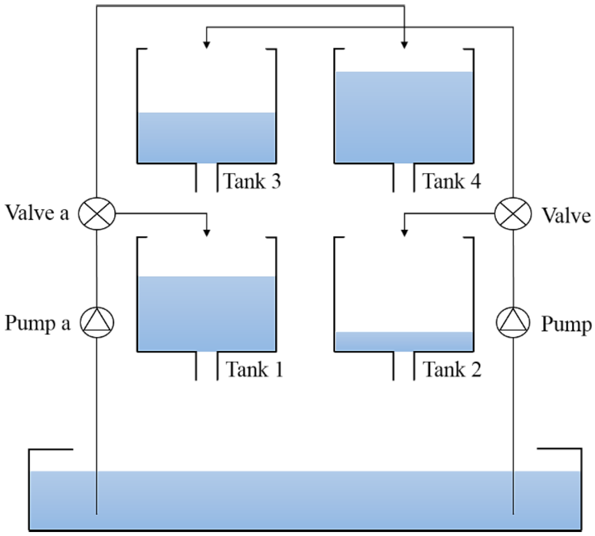

FIGURE 22. The four-tank system.

$$s_{q_b} = \left(\tilde{h}_2 + \tilde{h}_3\right) + \alpha_2\left(\dot{\tilde{h}}_2 + \dot{\tilde{h}}_3\right) \tag{77}$$

$$s_{\gamma_a} = \left(\tilde{h}_1 - \tilde{h}_4\right) + \alpha_3\left(\dot{\tilde{h}}_1 - \dot{\tilde{h}}_4\right) \tag{78}$$

$$s_{\gamma_b} = \left(\tilde{h}_2 - \tilde{h}_3\right) + \alpha_4\left(\dot{\tilde{h}}_2 - \dot{\tilde{h}}_3\right) \tag{79}$$

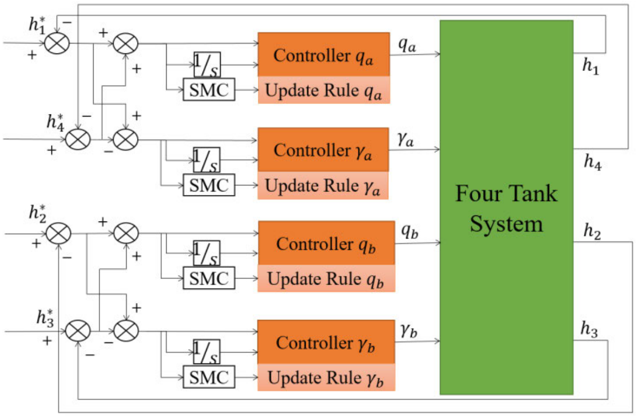

FIGURE 23. The block diagram of the four-tank system.

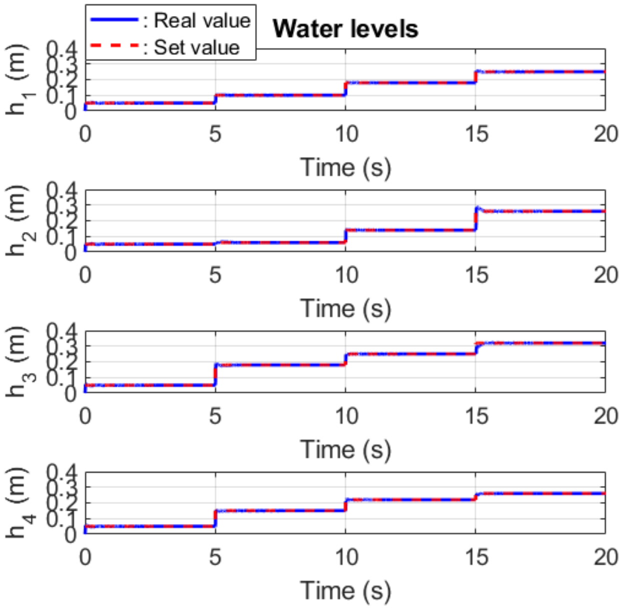

FIGURE 24. The water levels of the four tanks.

The block diagram of the proposed control system is illustrated as the Fig. 23.

The parameters of the benchmark plant are set as follow: the gravitational constant g_m is 9.81; all the initial heights of four tank h_1, h_2, h_3, h_4 are set as 0 m; the areas of inner diameter of pipe for tanks are $a_1 = 5.2\times10^{-3}$ m^2, $a_2 = 6.0\times10^{-3}$ m^2, $a_3 = 3.7\times10^{-3}$ m^2, $a_4 = 3.5\times10^{-3}$ m^2; the areas of tanks are $S_1 = 7.0\times10^{-2}$ m^2, $S_2 = 7.0\times10^{-2}$ m^2, $S_3 = 5.8\times10^{-2}$ m^2, $S_4 = 5.8\times10^{-2}$ m^2; the minimum of inlet flows q_a, q_b is 0 m^3/s; the minimum and maximum of the ratios of three-way valves γ_a, γ_b are 0 and 1, respectively.

Define the control signals which control q_a, q_b, γ_a and γ_b as the first, the second, the third and the fourth control signals, respectively. The parameters of the controller are initially chosen as follows: The learning rates of weights $\eta_{m1}, \eta_{m2}, \eta_{m3}$, and η_{m4} are chosen as 0.2, 0.2, 0.08 and 0.08, respectively. All the learning rates of variances $\eta_{v1}, \eta_{v2}, \eta_{v3}$,

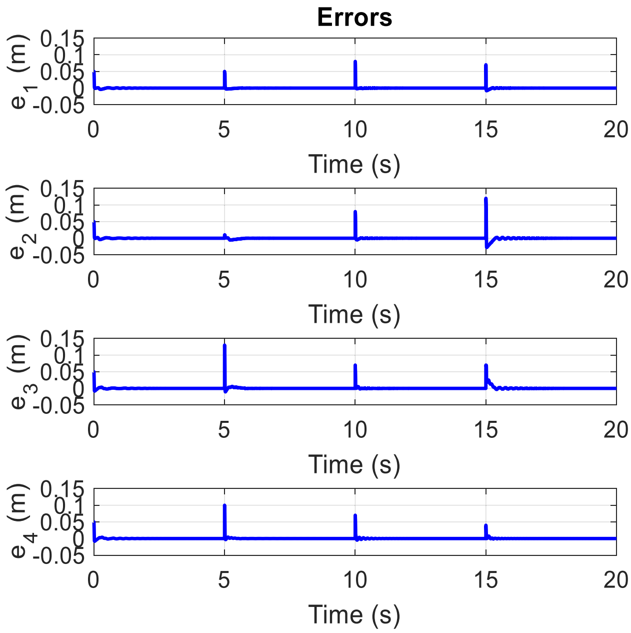

FIGURE 25. The errors of the water levels.

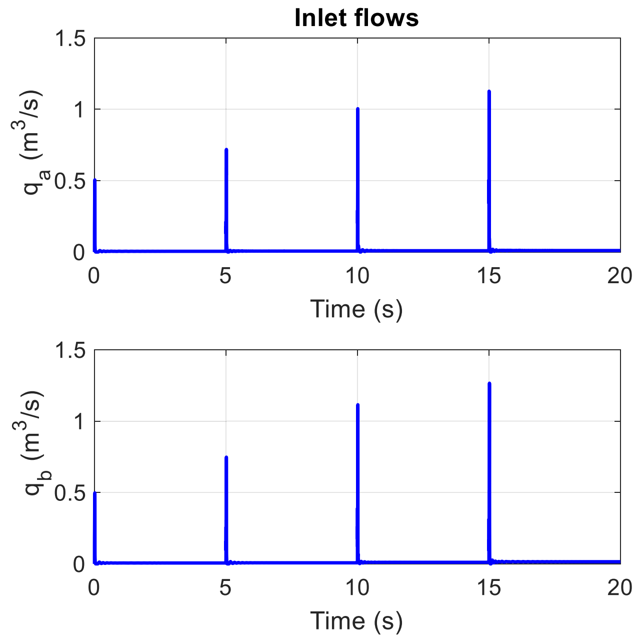

FIGURE 26. The inlet flows of the three-way valves.

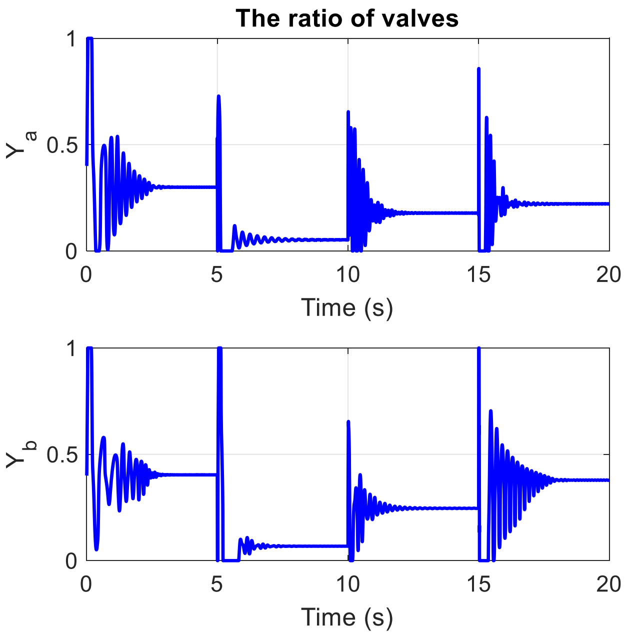

FIGURE 27. The ratios of the three-way valves.

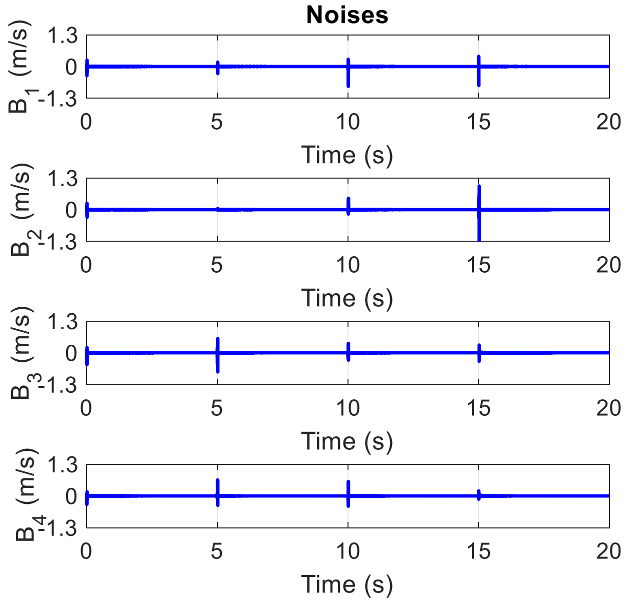

FIGURE 28. The noises of the system.

and η_{v4} are selected as 8×10^{-4}. All the weights of the first and second control signals have the learning rates $\alpha_{a1}, \alpha_{o1}, \alpha_{a2}$, and α_{o2} of 4×10^{-5} while all those figures for the third and forth control signals $\alpha_{a3}, \alpha_{o3}, \alpha_{a4}$, and α_{o4} are 1.5×10^{-4}. All the reward function parameters $b_{q1}, b_{q2}, b_{q3}, b_{q4}$ and $c_{q1}, c_{q2}, c_{q3}, c_{q4}$ are chosen as 1. The thresholds of the SO mechanism are set as $T_i = 0.3, T_{d1} = 0.001$ and $T_{d2} = 0.1$. All the parameters of sliding surface $\alpha_1, \alpha_2, \alpha_3$ and α_4 are determined as 200. All of the parameters are chosen by trial-and-error to ensure the desired control performance, and they are tuned online using the adaptive laws via (26)-(28), (30)-(33), and the learning rates can achieve the optimal values by (42). Moreover, the initial neurons for four controllers do not need to be designed in advance, because it can self-organize to a suitable value. In this example, the initial neurons are chosen as 60 for each controller $q_a, \gamma_a, q_b, \gamma_b$. It can be seen that when water levels change, the total neurons for each controller also increase to remain the control performance (see Fig. 30).

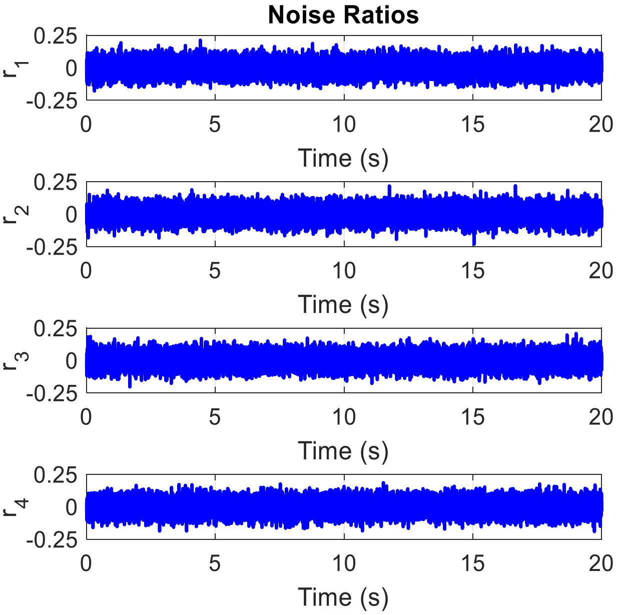

FIGURE 29. The noise ratios.

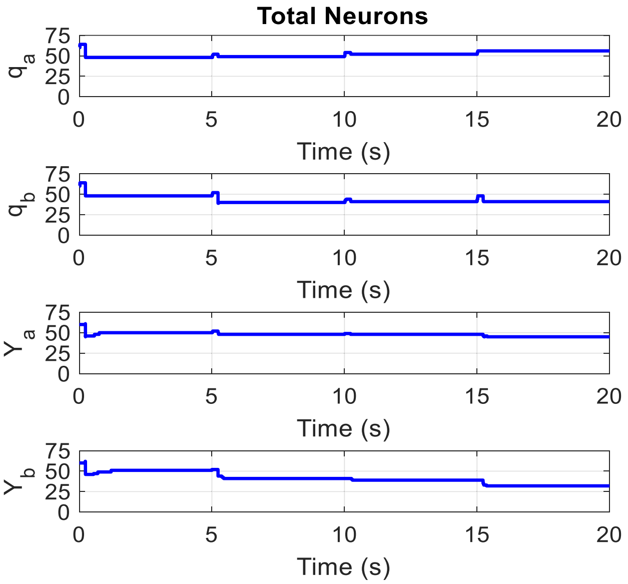

FIGURE 30. The neuron adjustment using the self-organizing mechanism.

C. SIMULATION RESULTS

The simulation results when using SO-DFL-BELC are described as Figs. 24-30. The comparison in RMSE of different controllers is presented in Figs. 31-35 and Table 2:

In Figs. 31-35, "BELC", "BELC updating m and v", "BELC using sliding surface", "DFL-BELC", SO-DFL-BELC" have the same meaning as mentioned in the first example. The different controllers are compared when using the same noise ratio data in Fig. 29. The RMSE shown in Table 2 is calculated as the average value of the RMSE of the four tanks.

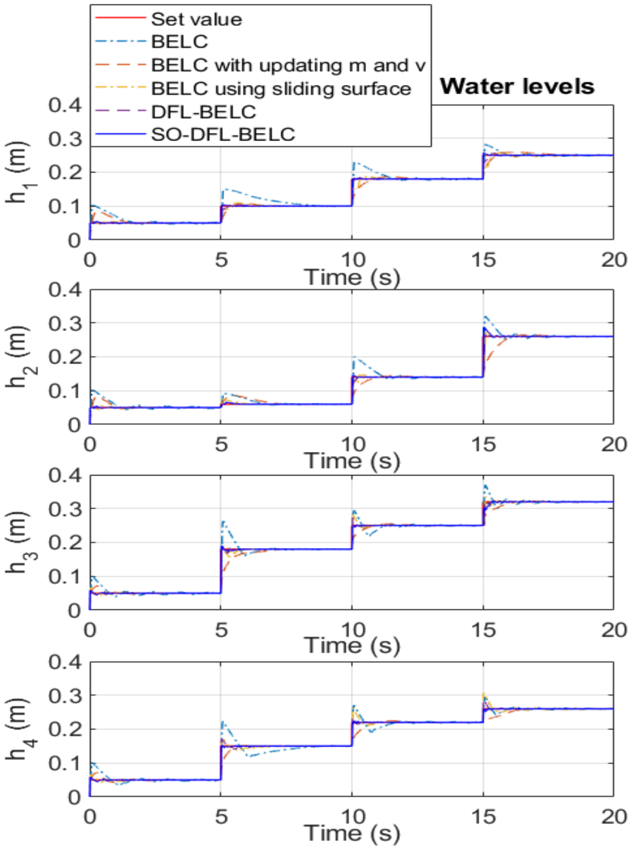

FIGURE 31. The comparison for the water levels.

TABLE 2. Comparision in root mean square error for different controllers.

Model	RMSE (m)
Conventional BELC	0.0151
BELC with updating $\boldsymbol{m}$ and $\boldsymbol{v}$	0.0116
BELC using sliding surface	0.0070
DFL-BELC	0.0049
SO-DFL-BELC	**0.0042**

D. COMMENTS OF THE SIMULATION RESULTS

From Figs. 20–26, it is seen that the SO-DFL-BELC is an effective controller for the four-tank system. For all cases of water levels, the rise time is less than $0.05s$, except the fourth rise of tank 2 and tank 3. It seems that at that time, the learning rates of the control signal q_b controlling the sum of the errors of tank 2 and tank 3 is slightly higher than these parameters of the control signal γ_b, which controls the difference between two errors. Figs. 24 and 25 illustrate the effect of noises to the system. It is clear that the noises randomly increase or decrease the state of the system in the range from -20% to 20%. The way how the SO mechanism has effects on the main controller is described clearly by Fig.26. The SO mechanism in this section is set up with the threshold for increasing, which is $T_i = 0.3$, is significantly

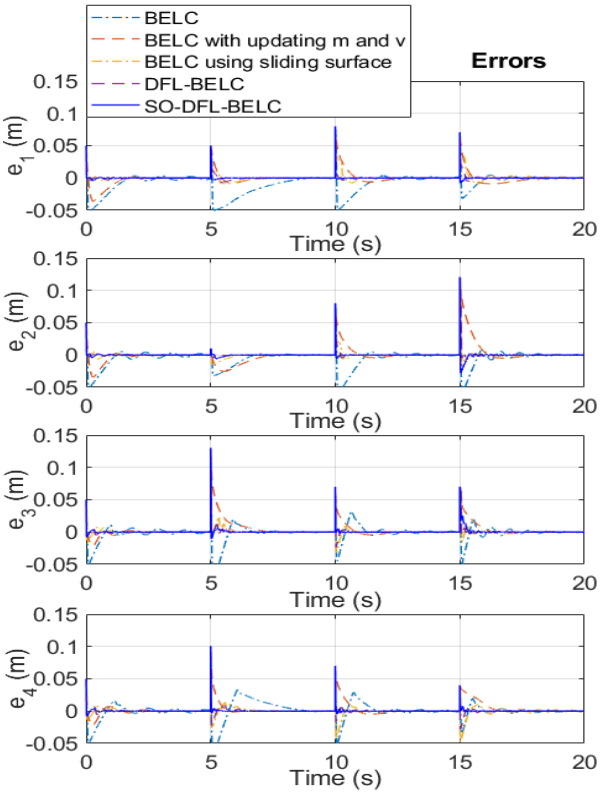

FIGURE 32. The comparison for the errors.

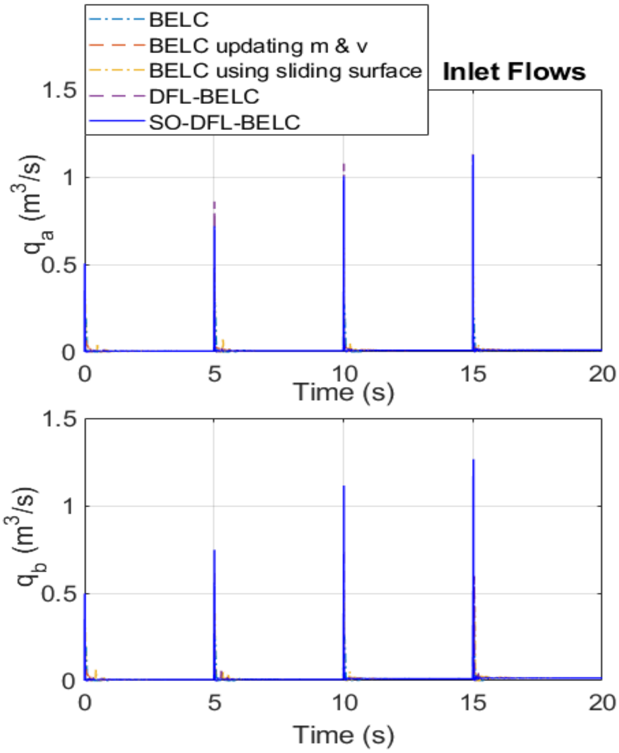

FIGURE 33. The comparison for the inlet flows.

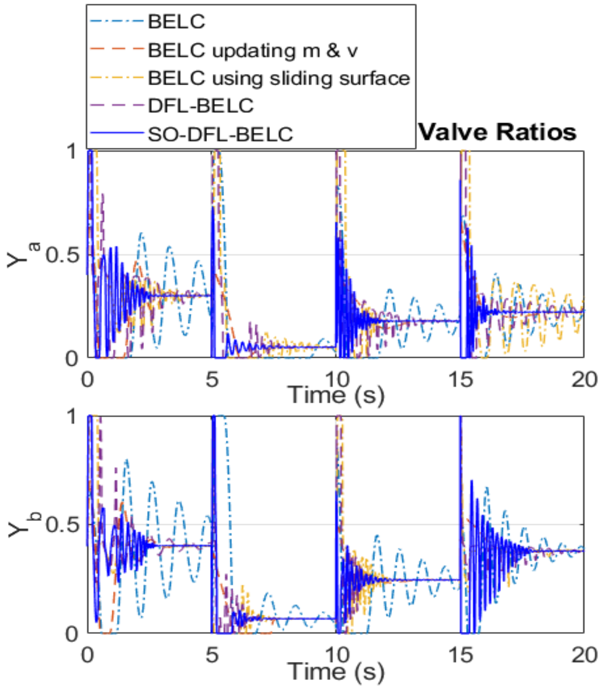

FIGURE 34. The comparison for the valve ratios.

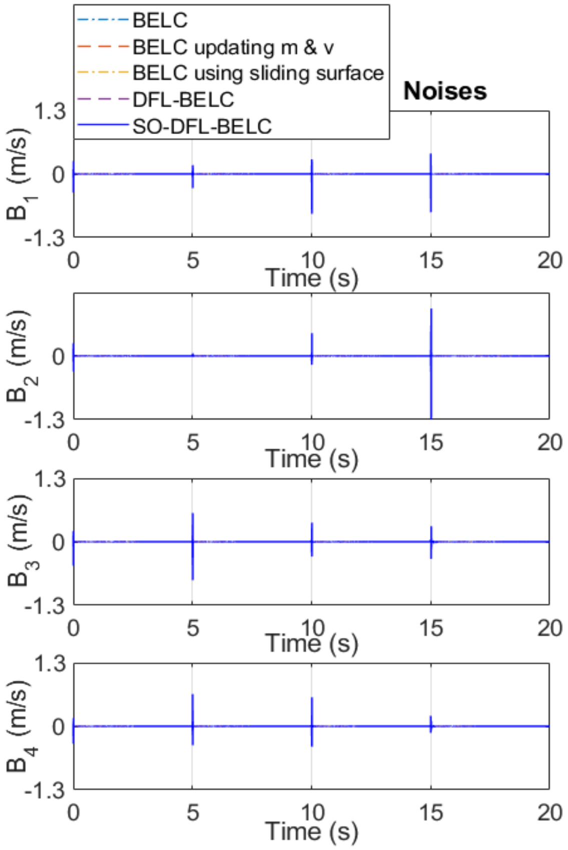

FIGURE 35. The comparison for the noises.

higher than the threshold for decreasing, that is $T_d = 0.001$. The SO mechanism can increase the computation time but it reduces the RMSE of the system. From Figs. 31-35 and Table 2, it can be observed that the RMSE is significantly improved when using the methods of updating means and variances overtime, the sliding surface for the updating rules, the DFL network, and the SO mechanism. In summary, the proposed SO-DFL-BELC attains better control performance and smaller RMSE than the others.

VI. CONCLUSION

This paper has successfully developed the BELC by combining it with the DFL network and the SO mechanism to create a novel neural network named as the SO-DFL-BELC. In addition, new updating rules based on gradient descent algorithm and sliding mode control are introduced to enhance the quality of the controller. Moreover, the proposed controller is utilized to control the 4D chaotic system and the four-tank system to demonstrate its effectiveness. Our future studies will focus on using the proposed control system for a practical model.

REFERENCES

[1] W. Deng, J. Yao, Y. Wang, X. Yang, and J. Chen, "Output feedback backstepping control of hydraulic actuators with valve dynamics compensation," *Mech. Syst. Signal Process.*, vol. 158, Sep. 2021, Art. no. 107769.

[2] W. Deng and J. Yao, "Extended-state-observer-based adaptive control of electrohydraulic servomechanisms without velocity measurement," *IEEE/ASME Trans. Mechatronics*, vol. 25, no. 3, pp. 1151–1161, Jun. 2020.

[3] W. Deng, J. Yao, and D. Ma, "Time-varying input delay compensation for nonlinear systems with additive disturbance: An output feedback approach," *Int. J. Robust Nonlinear Control*, vol. 28, no. 1, pp. 31–52, Jan. 2018.

[4] J. E. LeDoux, *The Amygdala: Neurobiological Aspects of Emotion*. New York, NY, USA: Wiley-Liss, 1992, pp. 339–351.

[5] J. E. LeDoux, "In search of an emotional system in the brain: Leaping from fear to emotion and consciousness," in *The Cognitive Neurosciences*. Cambridge, MA, USA: MIT Press, 1995, pp. 1049–1061.

[6] C. Balkenius and J. Moren, *A Computational Model of Emotional Learning in the Amygdala*. Cambridge, MA, USA: MIT Press, 2000.

[7] C. Balkenius and J. Morén, "Emotional learning: A computational model of the amygdala," *Cybern. Syst.*, vol. 32, no. 6, pp. 611–636, Sep. 2001.

[8] T.-T. Huynh, C.-M. Lin, T.-L. Le, N. P. Nguyen, S.-K. Hong, and F. Chao, "Wavelet interval type-2 fuzzy quad-function-link brain emotional control algorithm for the synchronization of 3D nonlinear chaotic systems," *Int. J. Fuzzy Syst.*, vol. 22, no. 8, pp. 2546–2564, Nov. 2020.

[9] M. Valikhani and C. Sourkounis, "A brain emotional learning-based intelligent controller (BELBIC) for DFIG system," in *Proc. Int. Symp. Power Electron., Electr. Drives, Autom. Motion*, Jun. 2014, pp. 713–718.

[10] M. A. Sharbafi, C. Lucas, and R. Daneshvar, "Motion control of omni-directional three-wheel robots by brain-emotional-learning-based intelligent controller," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 40, no. 6, pp. 630–638, Nov. 2010.

[11] M. A. Rahman, R. M. Milasi, C. Lucas, B. N. Araabi, and T. S. Radwan, "Implementation of emotional controller for interior permanent-magnet synchronous motor drive," *IEEE Trans. Ind. Appl.*, vol. 44, no. 5, pp. 1466–1476, Sep. 2008.

[12] H. A. Zarchi, E. Daryabeigi, G. R. A. Markadeh, and J. Soltani, "Emotional controller (BELBIC) based DTC for encoderless synchronous reluctance motor drives," in *Proc. 2nd Power Electron., Drive Syst. Technol. Conf.*, Feb. 2011, pp. 478–483.

[13] J. C. Patra and R. N. Pal, "A functional link artificial neural network for adaptive channel equalization," *Signal Process.*, vol. 43, no. 2, pp. 181–195, May 1995.

[14] C.-M. Lin, Y.-L. Liu, and H.-Y. Li, "SoPC-based function-link cerebellar model articulation control system design for magnetic ball levitation systems," *IEEE Trans. Ind. Electron.*, vol. 61, no. 8, pp. 4265–4273, Aug. 2014.

[15] C. M. Lin, T. T. Huynh, and C. C. Tsai, "A TOPSIS multi-criteria decision method-based function-link fuzzy CMAC control system design for uncertain nonlinear systems," in *Proc. IEEE Int. Conf. Fuzzy Syst. (FUZZ-IEEE)*, Jul. 2018, pp. 1–7.

[16] T.-T. Huynh and C.-M. Lin, "Wavelet dual function-link fuzzy brain emotional learning system design for system identification and trajectory tracking of nonlinear systems," in *Proc. IEEE Int. Conf. Syst., Man Cybern. (SMC)*, Oct. 2019, pp. 1653–1657.

[17] C.-M. Lin and H.-Y. Li, "Intelligent hybrid control system design for antilock braking systems using self-organizing function-link fuzzy cerebellar model articulation controller," *IEEE Trans. Fuzzy Syst.*, vol. 21, no. 6, pp. 1044–1055, Dec. 2013.

[18] T.-T. Huynh, C.-M. Lin, T.-L. Le, H.-Y. Cho, T.-T.-T. Pham, N.-Q.-K. Le, and F. Chao, "A new self-organizing fuzzy cerebellar model articulation controller for uncertain nonlinear systems using overlapped Gaussian membership functions," *IEEE Trans. Ind. Electron.*, vol. 67, no. 11, pp. 9671–9682, Nov. 2020.

[19] A. S. I. Zinober, *Deterministic Control of Uncertain Systems*. London, U.K.: Peter Peregrinus Press, 1990.

[20] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

[21] E. N. Lorenz, "Deterministic nonperiodic flow," *J. Atmos. Sci.*, vol. 20, no. 2, pp. 130–141, Mar. 1963.

[22] L. O. Tresor and M. Sumbwanyambe, "A selective image encryption scheme based on 2D DWT, henon map and 4D Qi hyper-chaos," *IEEE Access*, vol. 7, pp. 103463–103472, 2019.

[23] M. Samiullah, W. Aslam, H. Nazir, M. I. Lali, B. Shahzad, M. R. Mufti, and H. Afzal, "An image encryption scheme based on DNA computing and multiple chaotic systems," *IEEE Access*, vol. 8, pp. 25650–25663, 2020.

[24] T. Wang, D. Wang, and K. Wu, "Chaotic adaptive synchronization control and application in chaotic secure communication for industrial Internet of Things," *IEEE Access*, vol. 6, pp. 8584–8590, 2018.

[25] S. Sedaghatnejad and M. Farhang, "Detectability of chaotic direct-sequence spread-spectrum signals," *IEEE Wireless Commun. Lett.*, vol. 4, no. 6, pp. 589–592, Dec. 2015.

[26] I. Ahmad and B. Srisuchinwong, "A simple two-transistor 4D chaotic oscillator and its synchronization via active control," in *Proc. IEEE 26th Int. Symp. Ind. Electron. (ISIE)*, Jun. 2017, pp. 1249–1254.

[27] T.-T. Huynh, C.-M. Lin, T.-T.-T. Pham, H.-Y. Cho, and T.-L. Le, "A modified function-link fuzzy cerebellar model articulation controller using a PI-type learning algorithm for nonlinear system synchronization and control," *Chaos, Solitons Fractals*, vol. 118, pp. 65–82, Jan. 2019.

[28] K. H. Johansson, "The quadruple-tank process: A multivariable laboratory process with an adjustable zero," *IEEE Trans. Control Syst. Technol.*, vol. 8, no. 3, pp. 456–465, May 2000.

[29] I. Alvarado, D. Limon, D. de la Peña, J. M. Maestre, M. A. Ridao, H. Scheu, W. Marquardt, R. R. Negenborn, B. De Schutter, F. Valencia, and J. Espinosa, "A comparative analysis of distributed MPC techniques applied to the HD-MPC four-tank benchmark," *J. Process Control*, vol. 21, no. 5, pp. 800–815, Jun. 2011.

[30] X. Zhou, C. Li, T. Huang, and M. Xiao, "Fast gradient-based distributed optimisation approach for model predictive control and application in four-tank benchmark," *IET Control Theory Appl.*, vol. 9, no. 10, pp. 1579–1586, Jun. 2015.

[31] A. A. El-Gawad, A. N. Elden, M. E. Bahgat, and A. M. A. Ghany, "BELBIC load frequency controller design for a hydro-thermal power system," in *Proc. 21st Int. Middle East Power Syst. Conf. (MEPCON)*, Dec. 2019, pp. 809–814.

[32] A. M. Yazdani, A. Mahmoudi, M. A. Movahed, P. Ghanooni, S. Mahmoudzadeh, and S. Buyamin, "Intelligent speed control of hybrid stepper motor considering model uncertainty using brain emotional learning," *Can. J. Elect. Comput. Eng.*, vol. 41, no. 2, pp. 95–104, Aug. 2018.

[33] M. M. Zirkohi, "Stability analysis of brain emotional intelligent controller with application to electrically driven robot manipulators," *IET Sci., Meas. Technol.*, vol. 14, no. 2, pp. 182–187, Mar. 2020.

[34] T.-L. Le, C.-M. Lin, and T.-T. Huynh, "Interval type-2 Petri CMAC design for 4D chaotic system," in *Proc. Int. Conf. Syst. Sci. Eng. (ICSSE)*, Jul. 2019, pp. 420–424.

[35] C.-M. Lin and H.-Y. Li, "Dynamic Petri fuzzy cerebellar model articulation controller design for a magnetic levitation system and a two-axis linear piezoelectric ceramic motor drive system," *IEEE Trans. Control Syst. Technol.*, vol. 23, no. 2, pp. 693–699, Mar. 2015.

[36] F. J. Lin, C. H. Lin, and C. M. Hong, "Robust control of linear synchronous motor servodrive using disturbance observer and recurrent neural network compensator," *IEE Proc. Electr. Power Appl.*, vol. 147, no. 4, pp. 263–272, Jul. 2000.

CHIH-MIN LIN (Fellow, IEEE) was born in Changhua, Taiwan, in 1959. He received the B.S. and M.S. degrees from the Department of Control Engineering, National Chiao Tung University, Hsinchu, Taiwan, in 1981 and 1983, respectively, and the Ph.D. degree from the Institute of Electronics Engineering, National Chiao Tung University, in 1986. From 1997 to 1998, he was the Honor Research Fellow with The University of Auckland, New Zealand. He is currently a Chair Professor and the Vice President of Yuan Ze University, Taoyuan, Taiwan. He has published more than 200 journal articles and 160 conference papers. His research interests include fuzzy neural networks, cerebellar model articulation controller, intelligent control systems, adaptive signal processing, and classification problem. He also serves as an Associate Editor for IEEE Transactions on Cybernetics and IEEE Transactions on Fuzzy Systems.

HIEP-BINH NGUYEN was born in Hanoi, Vietnam, in 1995. He received the B.S. degree in control and automation from the Department of Electrical Engineering, Hanoi University of Science and Technology, Vietnam, in 2018. He is currently pursuing the M.S. degree in electrical engineering with Yuan Ze University, Taoyuan, Taiwan. His research interests include sliding mode control, backstepping control, neural networks, cerebellar model articulation controller, brain emotional learning-based intelligent controller, and deep learning.

TUAN-TU HUYNH (Member, IEEE) was born in Ho Chi Minh City, Vietnam, in 1982. He received the B.S. degree in electrical and electronics from the Department of Electrical and Electronics Engineering, Ho Chi Minh University of Technology and Education, Vietnam, in 2005, the M.S. degree in automation from the Ho Chi Minh City University of Transport, Vietnam, in 2010, and the Ph.D. degree in electrical engineering from Yuan Ze University, Taoyuan, Taiwan, in 2018. He is currently a Research Fellow with the Department of Electrical Engineering, Yuan Ze University, Chungli, Taiwan. He is also a Lecturer with the Faculty of Mechatronics and Electronics, Lac Hong University, Vietnam. His research interests include MCDM, fuzzy logic control, neural networks, cerebellar model articulation controller, brain emotional learning-based intelligent controller, deep learning, and intelligent control systems.