Research Article
An Improved Adaptive Tracking Controller of Permanent Magnet Synchronous Motor

Tat-Bao-Thien Nguyen,[1] **Teh-Lu Liao,**[1] **Hang-Hong Kuo,**[1] **and Jun-Juh Yan**[2]

[1] *Department of Engineering Science, National Cheng Kung University, Tainan 701, Taiwan*
[2] *Department of Computer and Communication, Shu-Te University, Kaohsiung 824, Taiwan*

Correspondence should be addressed to Jun-Juh Yan; jjyan@mail.stu.edu.tw

This paper proposes a new adaptive fuzzy neural control to suppress chaos and also to achieve the speed tracking control in a permanent magnet synchronous motor (PMSM) drive system with unknown parameters and uncertainties. The control scheme consists of fuzzy neural and compensatory controllers. The fuzzy neural controller with online parameter tuning is used to estimate the unknown nonlinear models and construct linearization feedback control law, while the compensatory controller is employed to attenuate the estimation error effects of the fuzzy neural network and ensure the robustness of the controlled system. Moreover, due to improvement in controller design, the singularity problem is surely avoided. Finally, numerical simulations are carried out to demonstrate that the proposed control scheme can successfully remove chaotic oscillations and allow the speed to follow the desired trajectory in a chaotic PMSM despite the existence of unknown models and uncertainties.

1. Introduction

Permanent magnet synchronous motors have had a great deal of attention for industrial applications in the recent years [1–6]. Due to the compact size, high speed, high efficiency, high power density, and low inertia, PMSMs are widely used in many fields of industry. However, the control and the stabilization for PMSMs still have some challenges as their high nonlinearity and even chaotic behavior.

In the field of power electronics, chaos in motor drive systems occurs when parameters fall into a certain area as defined by Kuroe and Hayashi [7]. Then, the existence of chaos has been found in several types of motor driver systems [7–9]. Chaotic phenomenon in PMSMs is completely studied by Li et al. [10], and this study points out that the chaotic oscillations appear when the parameters lie in a certain area. Since the undesired chaotic oscillations can affect the stabilization of a motor drive system negatively, causing the drive system's collapse, it has been more and more critical to control and eliminate the chaos. Until now, despite various chaos control methods developed for PMSMs such as

feedback linearization [11], sliding mode control [12], quasi-sliding mode control [13, 14], adaptive backstepping control [15], and time-delay feedback control [16], there are some shortcomings which still exist. Most of the methods require the exact mathematical models to calculate the control laws. This leads to implemental difficulties in practical systems where mathematical models can be dynamic due to undesired uncertainties. Moreover, in the adaptive backstepping method, a single system parameter is considered as an unknown and constant parameter, which can be seen as a restriction. The time-delay feedback control encounters some problems; for example, the control object is not an equilibrium or unstable periodic orbit. Furthermore, determination of delay time for this controller is quite difficult.

In the past two decades, neural networks (NNs) and fuzzy logic (FL) have been widely used to model and control highly uncertain, nonlinear, and complex systems [17–22]. Incorporating the estimation abilities of NNs (or FL) into adaptive control method, direct and indirect adaptive control methods were developed [23]. In the indirect adaptive control method, the control input $u(t)$ usually appears in the form of

$u(t) = (1/\widehat{g}(\mathbf{x}, t))(v(t) - \widehat{f}(\mathbf{x}, t))$, where the estimations $\widehat{g}(\mathbf{x}, t)$ and $\widehat{f}(\mathbf{x}, t)$ are calculated by a FL system or NNs, and $v(t)$ is linearization input. It is well known that these estimations cannot be guaranteed to be bounded away from zero for all time t. In other words, $\widehat{g}(\mathbf{x}, t)$ may tend to be zero or be close to zero in some points in time. Such situations lead to very large control signals which may cause the controlled systems to lose their controllability or even damage the whole systems. This problem was known as the singularity problem which usually appears in the indirect adaptive control method based on fuzzy or neural estimation.

Fuzzy neural networks (FNNs) incorporate the advantages of fuzzy inference and neurolearning [24–28]. FNNs can simulate the merits of human knowledge representation and thinking of fuzzy theory and associate the learning ability and computational power of NNs. In harnessing these tools, numerous researchers have developed fuzzy systems and neural networks to control PMSMs. For example, a self-organizing fuzzy sliding mode controller is developed by Guo and Long [29], while a fuzzy model was built for chaotic dynamics of PMSM in fuzzy guaranteed cost control [30, 31]. Yu et al. proposed the adaptive fuzzy control method via backstepping [32]. Despite these control techniques' advances, some weaknesses still remain. It is difficult to choose the fuzzy gains scaling factors and parameters of membership functions in the self-organizing fuzzy sliding mode controller. Choosing the suitable rules for constructing chaotic dynamics of PMSM is significantly hard in the fuzzy guaranteed cost control, while in the adaptive fuzzy control method via backstepping, the error is impossible to converge to the origin.

Because the existing control methods for chaotic PMSM still have shortcomings, we develop a new controller which is able to operate under the effects of unknown system parameters and uncertainties. The developed controller uses a simple fuzzy neural network to online estimate the unknown dynamics and uncertainties and then constructs the linearization feedback control law. Due to the learning abilities inherited from fuzzy neural networks, the controller can work well with fully unknown dynamic model and uncertainties. Also, the compensatory controller is added to the controlled system to make the system robust and contribute to zero convergence of tracking error. As the proposed controller has the advantages in both linearization feedback control and fuzzy neural network, it can cancel the unknown nonlinear terms of the dynamic model and obtain the advanced tracking performance effectively. Moreover, with the improved design, the controller not only meets the control objective but also surely avoids the singularity problem even with the initial phase. In contrast, many previous works dealing with chaos control of the PMSM have the restriction in that controller design relies on the model of PMSM [11–16]. In other words, an exact mathematical model of PMSM is necessary to design the control laws. This also implies that these controllers cannot work or work imprecisely when the system parameters or the model of PMSM is not sufficiently known. On the other hand, some fuzzy control methods [29–32] can solve the control problem

with unknown model of PMSM, but some disadvantages still remain. The self-organizing fuzzy sliding mode control needs much knowledge of specialist to set up the fuzzy rules. The fuzzy guaranteed cost control may have the awful performance when the reference point is not the origin. In the adaptive fuzzy control method via backstepping, the tracking error is impossible to converge to zero. Therefore, in comparison with previous methods, the proposed control shows the improvements in controlling chaotic PMSM. With the proposed controller, chaotic oscillations in a PMSM are successfully suppressed and the motor speed is forced to follow the desired trajectory and the tracking error converges to zero asymptotically. Finally, the simulations are carried out to illustrate the effectiveness and robustness of the proposed controller.

The rest of this paper is organized as follows. The dynamics of a PMSM and the formulation of the chaos control problem are outlined in Section 2. The design of the adaptive controller based on a fuzzy neural network is described in Section 3. In Section 4, simulation results are given to confirm the validity of the proposed method. Finally, the conclusion is offered in Section 5.

2. Problem Formulation

2.1. Dynamic Model of Chaotic PMSM. The dynamic model of a PMSM with the smooth air gap can be described as follows [10]:

$$\frac{d\omega}{dt} = \sigma \left(i_q - \omega \right) + \widetilde{T}_L,$$
$$\frac{di_q}{dt} = -i_q - i_d\omega + \gamma\omega + \widetilde{u}_q, \tag{1}$$
$$\frac{di_d}{dt} = -i_d + i_q\omega + \widetilde{u}_d,$$

where ω, i_d, and i_q are state variables, which denote angle speed and d-q axis currents, respectively. The state ω can be directly measured, while states i_d and i_q can be calculated by using d-q transformation. σ and γ are system parameters. $\widetilde{T}_L$, $\widetilde{u}_d$, and $\widetilde{u}_q$ stand for the load torque and d-q axis voltages, respectively.

In system (1), when the external inputs are set to zero, namely, $\widetilde{T}_L = \widetilde{u}_d = \widetilde{u}_q = 0$, the system becomes an unforced system [10] as

$$\frac{d\omega}{dt} = \sigma \left(i_q - \omega \right),$$
$$\frac{di_q}{dt} = -i_q - i_d\omega + \gamma\omega, \tag{2}$$
$$\frac{di_d}{dt} = -i_d + i_q\omega.$$

The theories of bifurcation and chaos have been widely used to study the stability of PMSM drive systems in [10]. The study showed that a PMSM produces chaotic oscillations when system parameters σ and γ fall into a certain area. For
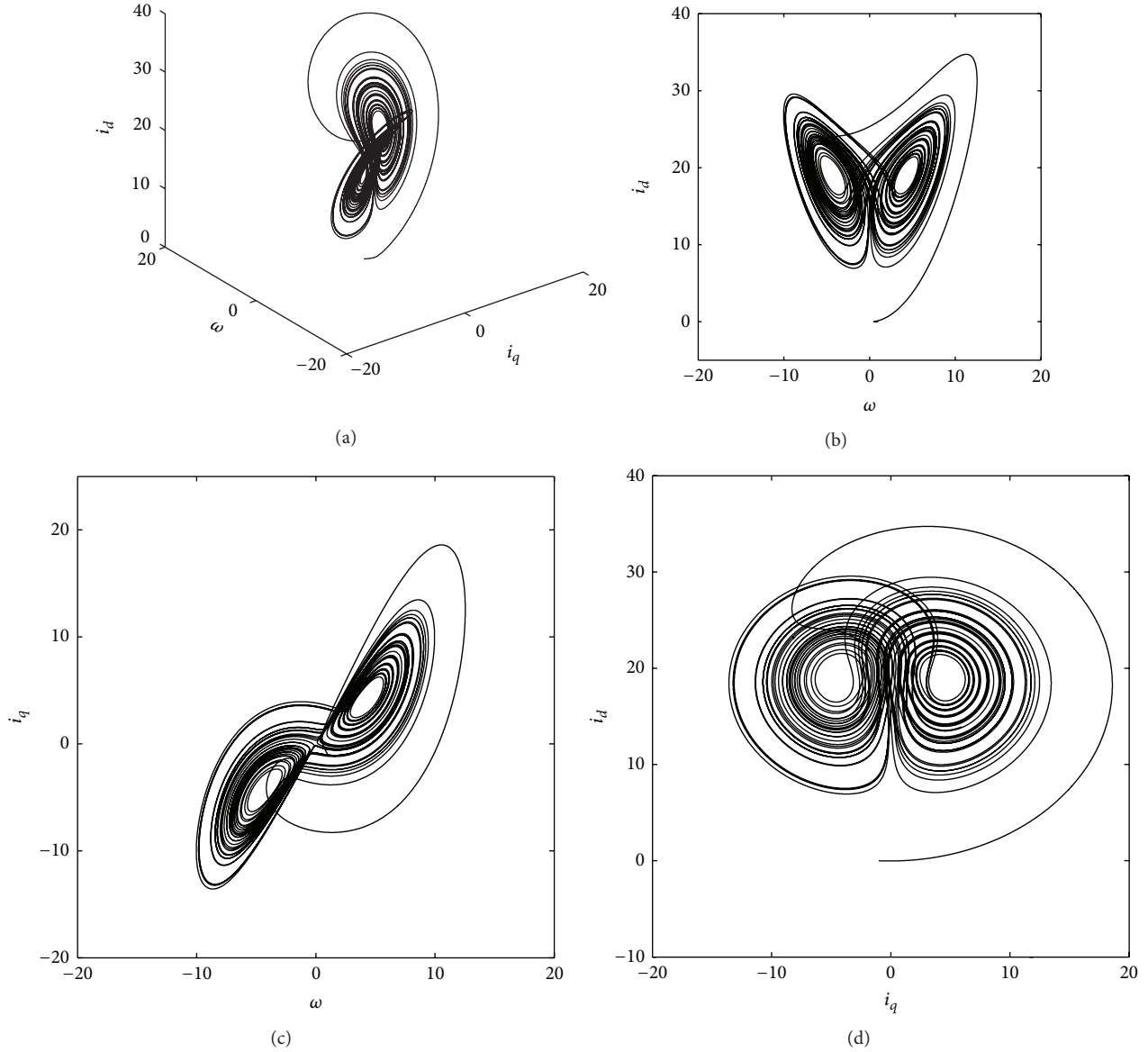
FIGURE 1: (a) Typical chaotic attractor in a PMSM. (b) Chaotic motion on ω-i_d plane. (c) Chaotic motion on ω-i_q plane. (d) Chaotic motion on i_q-i_d plane.

example, the system in (2) displays chaos when the system parameters are set as $\sigma = 5.45$ and $\gamma = 20$, and the initial states are given as $[\omega(0)\ \ i_q(0)\ \ i_d(0)]^T = [1\ \ -1\ \ 0]^T$. The typical chaotic attractor of a PMSM is exhibited in Figure 1 and the bifurcation diagrams of the quadrature current i_q versus the parameters, σ and γ, are presented in Figure 2, respectively.

2.2. Problem Formulation. Since the chaotic oscillations can destroy the stability of the PMSM drive system, we propose an adaptive controller, a PMSM, to suppress chaos and achieve the speed tracking control. Let us consider the PMSM drive system in (2). We add a control input u to the second differential equation as the manipulated variable, which is desirable for real applications. And for simplicity,

the following notations are introduced as $x_1 = \omega$, $x_2 = i_q$, and $x_3 = i_d$. In this manner, the system in (2) with uncertainties can be rewritten as follows:

$$\dot{x}_1 = \sigma(x_2 - x_1) + d_1,$$
$$\dot{x}_2 = -x_2 - x_1 x_3 + \gamma x_1 + d_2 + u, \qquad (3)$$
$$\dot{x}_3 = -x_3 + x_2 x_1 + d_3,$$

where $d_i(x, t) \in R$, $i = 1, 2, 3$, is uncertainty applied to the PMSM due to parameter perturbation and external uncertainties. σ and γ are unknown system parameters and are also located within the chaotic area.

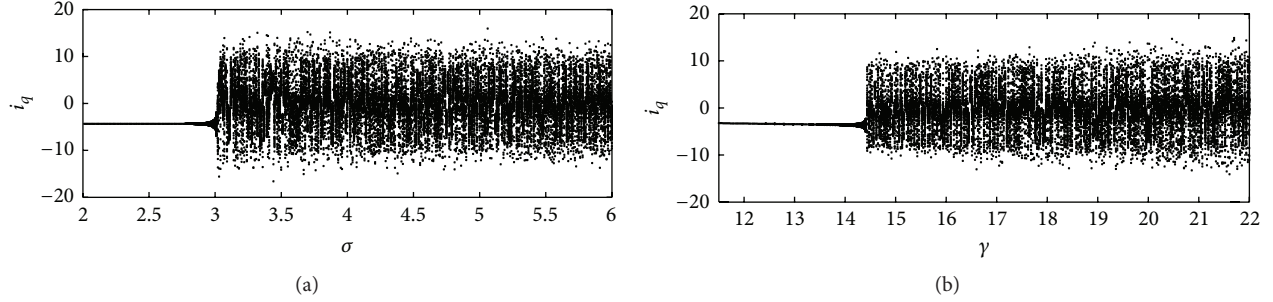*Assumption 1.* The uncertainty, $d_i(x, t) \in R$, $i = 1, 2, 3$, is bounded.

FIGURE 2: Bifurcation diagrams of i_q versus (a) σ with $\gamma = 20$ and (b) γ with $\sigma = 5.45$.

In order to force the speed x_1 of PMSM to follow the desired trajectory $y_d(t) \in R$, the system in (3) is expressed in the standard form of the single input, single output (SISO) system with output $y(t) = x_1$ as

$$\dot{x} = f(x) + g(x)u,$$
$$y = h(x),$$
(4)

where

$$x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}, \qquad f(x) = \begin{bmatrix} \sigma(x_2 - x_1) + d_1 \\ -x_2 - x_1 x_3 + \gamma x_1 + d_2 \\ -x_3 + x_2 x_1 + d_3 \end{bmatrix},$$
$$g(x) = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \qquad h(x) = x_1.$$
(5)

With the control signal u inserted into the system above, the SISO system in (4) has the relative degree $r = 2$. By using Lie derivatives, we take the transformation as

$$z_1 = y,$$
$$z_2 = L_f h(x) = \sigma(x_2 - x_1) + d_1,$$
(6)

which leads to

$$\dot{z}_1 = z_2,$$
$$\dot{z}_2 = a(x) + b(x)u,$$
$$y = z_1,$$
(7)

where

$$a(x) = L_f^2 h(x)$$
$$= \left(-\sigma + \frac{\partial d_1}{\partial x_1}\right)(\sigma x_2 - \sigma x_1 + d_1)$$
$$+ \left(\sigma + \frac{\partial d_1}{\partial x_2}\right)(-x_2 - x_1 x_3 + \gamma x_1 + d_2)$$
$$+ \frac{\partial d_1}{\partial x_3}(-x_3 + x_1 x_3 + d_3),$$
(8)

$$b(x) = L_g L_f h(x) = \sigma + \frac{\partial d_1}{\partial x_2}.$$

The control goal is to design a controller that can suppress chaos and allow the output $y(t) \in R$ to track the desired trajectory $y_d(t) \in R$. Based on linearization feedback control method [33], the ideal control law $u^*(x)$ is given to reach the control goal as

$$u^*(x) = \frac{1}{b(x)}(-a(x) + v(t)),$$
(9)

where $v(t) \in R$ is the linearization input and can be computed as

$$v(t) = \ddot{y}_d(t) + \bar{e}_s(t) + \eta e_s(t),$$
(10)

where η is a positive factor. $e_s(t)$ and $\bar{e}_s(t)$ can be calculated according to the following equations:

$$e_0(t) = y_d(t) - y(t),$$
(11)

$$e_s(t) = \dot{e}_0(t) + k e_0(t),$$
(12)

$$\bar{e}_s(t) = \dot{e}_s(t) - \ddot{e}_0(t) = k \dot{e}_0(t),$$
(13)

where $e_0(t)$ is the tracking error and k is chosen to ensure that $\Delta(s) = s + k$ is a Hurwitz polynomial.

In order to make (9) proper and use $b(x)$ to determine the property of the Lyapunov function candidate, the following assumptions are needed.

Assumption 2. $b(x)$ is lower bounded by a known positive constant $\underline{b}$; that is, $0 < \underline{b} \le b(x) < \infty$.

Assumption 3. Desired trajectory $y_d(t)$ is continuously differentiable and bounded up to the second-order. $\dot{y}_d(t)$ and $\ddot{y}_d(t)$ are measurable.

Substituting (9) into (7), one can get

$$\dot{z}_2 = \ddot{y}(t) = v(t) = \ddot{y}_d(t) + \bar{e}_s(t) + \eta e_s(t).$$
(14)

Using (14) and (11) leads to

$$\ddot{e}_0(t) + \bar{e}_s(t) + \eta e_s(t) = 0.$$
(15)

The error dynamics can be obtained by substituting (13) into (15) as

$$\dot{e}_s(t) + \eta e_s(t) = 0.$$
(16)

Since η is assumed to be positive, as mentioned in (10), and k satisfies the Hurwitz polynomial $\Delta(s) = s + k$, the equation in (16) expresses that both $e_s(t)$ and therefore $e_0(t)$ converge to zero exponentially. For this reason, the controlled system is stable and the perfect tracking is achieved.

Moreover, due to the relative degree $r = 2$ and the order of the system $n = 3$, the zero dynamics is considered. It is possible to find a function $h_3(x)$ such that $(\partial h_3(x)/\partial x)g(x) = 0$, and then we define the state $z_3 = h_3(x) = x_3$ to obtain the additional state equation as

$$\dot{z}_3 = \dot{x}_3 = -z_3 + \frac{1}{\sigma}z_1\left(z_2 + \sigma z_1 - d_1\right) + d_3. \qquad (17)$$

When $z_1 = z_2 = 0$, (17) can be rewritten as $\dot{z}_3 = -z_3 + d_3$. This equation expresses the stable zero dynamics of the system. Because the zero dynamics is stable, the system is a minimum phase system. Thus, the state variable z_3 is also stable when both state variables z_1 and z_2 are stable.

However, since σ, γ, and d_i, $i = 1, 2, 3$, are unknown, $a(x)$ and $b(x)$ cannot be determined exactly. This leads to the fact that the ideal control law in (9) cannot perform. In order to overcome this problem, we use fuzzy neural networks to estimate $a(x)$ and $b(x)$.

2.3. Description of Fuzzy Neural Networks. In this section, a fuzzy neural network (FNN), which is used to estimate unknown functions $a(x)$ and $b(x)$, is described. The FNN incorporates the advantages of a fuzzy logic system and a neural network; that is, the FNN possesses the learning ability of a neural network and the human thinking of a fuzzy logic system [25]. The basic structure of a fuzzy logic system consists of fuzzification, rulebase, fuzzy inference, and defuzzification. The fuzzification is the process of mapping inputs, state variables x_1, x_2, and x_3, to membership values in the input universes of discourse. The rulebase consists of nine antecedent-consequent linguistic rules (IF-THEN rules) in which the ith rule is described in the form of

$$\begin{aligned} &\text{IF } \left(x_1 \text{ is } A_1^i \text{ AND } x_2 \text{ is } A_2^i \text{ AND } x_3 \text{ is } A_3^i\right)\\ &\quad \text{THEN } \left(\widehat{a} \text{ is } B_a^i \text{ AND } \widehat{b} \text{ is } B_b^i\right), \end{aligned} \qquad (18)$$

where A_1^i, A_2^i, A_3^i, B_a^i, and B_b^i are fuzzy sets which are denoted by the membership functions $\mu_{A_1^i}$, $\mu_{A_2^i}$, $\mu_{A_3^i}$, $\mu_{B_a^i}$, and $\mu_{B_b^i}$, respectively. $\widehat{a}(x) \in R$ and $\widehat{b}(x) \in R$ are outputs of the fuzzy logic system, which stand for the estimations of $a(x)$ and $b(x)$, respectively. $\mu_{A_1^i}$, $\mu_{A_2^i}$, and $\mu_{A_3^i}$ use Gaussian functions to calculate their values, while $\mu_{B_a^i}$ and $\mu_{B_b^i}$ are fuzzy singletons. The fuzzy inference is the process of mapping membership values from the input windows, through the rulebase, to the output window. The fuzzy inference employs product inference for mapping. The defuzzification is the procedure of mapping from a set of inferred fuzzy signals contained within a fuzzy output window to a crisp signal. Using
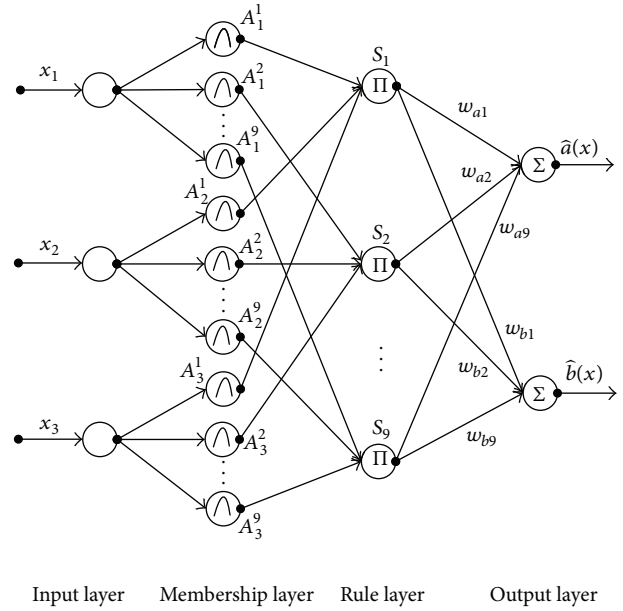

Figure 3: Structure of a fuzzy neural network.

the center-average defuzzification techniques, the outputs of a fuzzy logic system can be represented as

$$\begin{aligned} \widehat{a}(x) &= \frac{\sum_{i=1}^{9} w_{ai}\left(\prod_{j=1}^{3}\mu_{A_j^i}(x)\right)}{\sum_{i=1}^{9}\left(\prod_{j=1}^{3}\mu_{A_j^i}(x)\right)} = W_a^T S(x),\\[2mm] \widehat{b}(x) &= \frac{\sum_{i=1}^{9} w_{bi}\left(\prod_{j=1}^{3}\mu_{A_j^i}(x)\right)}{\sum_{i=1}^{9}\left(\prod_{j=1}^{3}\mu_{A_j^i}(x)\right)} = W_b^T S(x), \end{aligned} \qquad (19)$$

where $W_a^T = [w_{a1} \ w_{a2} \ \cdots \ w_{a9}]$ and $W_b^T = [w_{b1} \ w_{b2} \ \cdots \ w_{b9}]$ are weighting vectors adjusted according to adaptive laws. The parameters w_{ai} and w_{bi} with $i = 1, 2, \ldots, 9$ are the points where membership functions $\mu_{B_a^i}(w_{ai})$ and $\mu_{B_b^i}(w_{bi})$ achieve maximum values; that is, $\mu_{B_a^i}(w_{ai}) = \mu_{B_b^i}(w_{bi}) = 1$. $S^T(x) = [s_1(x) \ s_2(x) \ \cdots \ s_9(x)]$ is a fuzzy basic vector where each element $s_i(x)$, $i = 1, 2, \ldots, 9$, is defined as

$$s_i(x) = \frac{\prod_{j=1}^{3}\mu_{A_j^i}(x)}{\sum_{i=1}^{9}\left(\prod_{j=1}^{3}\mu_{A_j^i}(x)\right)}. \qquad (20)$$

The fuzzy logic system can be expressed by a neural network, which is known as a fuzzy neural network [25, 26]. As shown in Figure 3, the fuzzy neural network has four layers, including the input layer, membership layer, rule layer, and output layer. At the input layer, each node is an input representing a state variable. At the membership layer, the Gaussian functions are used as membership functions to calculate the membership values. At the rule layer, each node stands for an element $s_i(x)$, $i = 1, 2, \ldots, 9$, of the fuzzy basis vector $S(x)$ and performs a fuzzy rule. The links between the rule layer and output layer are fully connected by

the components of weighting vectors W_a and W_b. At the output layer, two outputs represent the value of $\hat{a}(x)$ and $\hat{b}(x)$.

3. Adaptive Fuzzy Neural Controller Design

Because $a(x)$ and $b(x)$, as described in (8), cannot be calculated explicitly, the ideal control law (9) is unable to be implemented. In order to overcome this impediment, a neural network, as shown in Figure 3, is proposed to estimate $a(x)$ and $b(x)$. Let $\hat{a}(x, t)$ and $\hat{b}(x, t)$ be the estimations of $a(x)$ and $b(x)$, respectively. Then, following the certainty equivalent approach, the fuzzy neural controller $u_{nn}(x)$ based on the ideal control law (9) can be obtained as

$$u_{nn}(x) = \frac{1}{\hat{b}(x, t)} \left(-\hat{a}(x, t) + v(t)\right). \tag{21}$$

However, the control law in (21) may face the singularity problem when $\hat{b}(x, t)$ closes to zero or even receives the zero value in some point in time initially, leading to possible large values for control signal $u_{nn}(x)$. In such situation, the closed-loop controlled system may lose controllability. To avoid this problem, we replace the control law in (21) with

$$u_{nn}(x) = \frac{\hat{b}(x, t)}{\hat{b}^2(x, t) + \varepsilon} \left(-\hat{a}(x, t) + v(t)\right), \tag{22}$$

where ε is a nonzero constant. The constant ε is introduced to guarantee that the term $\hat{b}^2(x, t) + \varepsilon$ is always nonzero, and therefore the singularity problem can be avoided. The estimations $\hat{a}(x, t)$ and $\hat{b}(x, t)$ are calculated by a fuzzy neural network as

$$\hat{a}(x, t) = W_a^T(t) S(x),$$
$$\hat{b}(x, t) = W_b^T(t) S(x), \tag{23}$$

where $W_a(t)$ and $W_b(t)$ are weighting vectors at the output layer of the neural network in Figure 3, while $S(x)$ is the fuzzy basic vector defined in (20). In the adaptive mechanism, $W_a(t)$ and $W_b(t)$ are online tuned so that $\hat{a}(x, t)$ and $\hat{b}(x, t)$ converge to $a(x)$ and $b(x)$, respectively, and reach their optimal values. The achieved optimal weighting vectors W_a^* and W_b^* are defined by

$$W_a^* = \arg\min_{W_a \in \Theta_a} \left\{ \sup_{x \in \Omega} \left| W_a^T(t) S(x) - a(x) \right| \right\},$$
$$W_b^* = \arg\min_{W_a \in \Theta_b} \left\{ \sup_{x \in \Omega} \left| W_b^T(t) S(x) - b(x) \right| \right\}, \tag{24}$$

where Θ_a and Θ_b are sets of acceptable values of vectors $W_a(t)$ and $W_b(t)$, respectively, and Ω is a compact set of state variable x. In this paper, we assume that the used fuzzy neural network does not violate the estimation property on the compact set Ω, and the compact set Ω is large enough so that state variables remain within Ω under the control action.

In adaptive mechanism, the adaptive laws for $W_a(t)$ and $W_b(t)$ are chosen as

$$\dot{W}_a(t) = -Q_a^{-1} S(x) e_s(t),$$
$$\dot{W}_b(t) = -Q_b^{-1} S(x) u_{nn}(x) e_s(t), \tag{25}$$

where Q_a and Q_b are positive-definite weighting matrices.

For the ideal situation, when $W_a(t)$ and $W_b(t)$, respectively, approach W_a^* and W_b^*, $\hat{a}(x, t)$ and $\hat{b}(x, t)$, respectively, approach $a(x)$ and $b(x)$. However, there exist the unavoidable estimation errors because $\hat{a}(x, t)$ and $\hat{b}(x, t)$ are estimated by a neural network which has a finite number of units in the hidden layer. Consequently, $\hat{a}(x, t)$ and $\hat{b}(x, t)$ cannot converge to $a(x)$ and $b(x)$ even when $W_a(t)$ and $W_b(t)$ converge to W_a^* and W_b^*, respectively. Let $\delta_a(x)$ and $\delta_b(x)$ be the estimation errors; then, the exact models of $a(x)$ and $b(x)$ can be expressed by

$$a(x) = W_a^* S(x) + \delta_a(x),$$
$$b(x) = W_b^* S(x) + \delta_b(x). \tag{26}$$

The differences between the estimation models and exact models can be described by

$$\hat{a}(x, t) - a(x) = \left(W_a(t) - W_a^*\right)^T S(x) - \delta_a(x)$$
$$= \widetilde{W}_a^T(t) S(x) - \delta_a(x),$$
$$\hat{b}(x, t) - b(x) = \left(W_b(t) - W_b^*\right)^T S(x) - \delta_b(x)$$
$$= \widetilde{W}_b^T(t) S(x) - \delta_b(x), \tag{27}$$

where $\widetilde{W}_a(t) = W_a(t) - W_a^*$ and $\widetilde{W}_b(t) = W_b(t) - W_b^*$ are parameter errors.

We suppose that the estimation errors of the neural network are bounded, and they can be expressed in the following assumption.

Assumption 4. The estimation errors are upper bounded by some known constants $\overline{\delta}_a > 0$ and $\overline{\delta}_b > 0$ over the compact set $\Omega \subset R^3$; that is,

$$\sup_{x \in \Omega} \left| \delta_a(x) \right| \le \overline{\delta}_a,$$
$$\sup_{x \in \Omega} \left| \delta_b(x) \right| \le \overline{\delta}_b. \tag{28}$$

The estimation errors are unavoidable, and sometimes they can break down the stability of the close-loop controlled system. In order to keep the system robustness, a compensatory controller $u_c(x)$ is used as an additional controller to compensate for the estimation errors. The compensatory controller $u_c(x)$ is designed as

$$u_c(x) = \frac{1}{\underline{b}} \left(\overline{\delta}_a + \overline{\delta}_b \left| u_{nn}(x) \right| + \left| u_r(x) \right| \right) \mathrm{sgn}\left(e_s(t)\right), \tag{29}$$

where $u_r(x) = (\varepsilon/(\hat{b}^2(x, t) + \varepsilon))(-\hat{a}(x, t) + v(t))$.
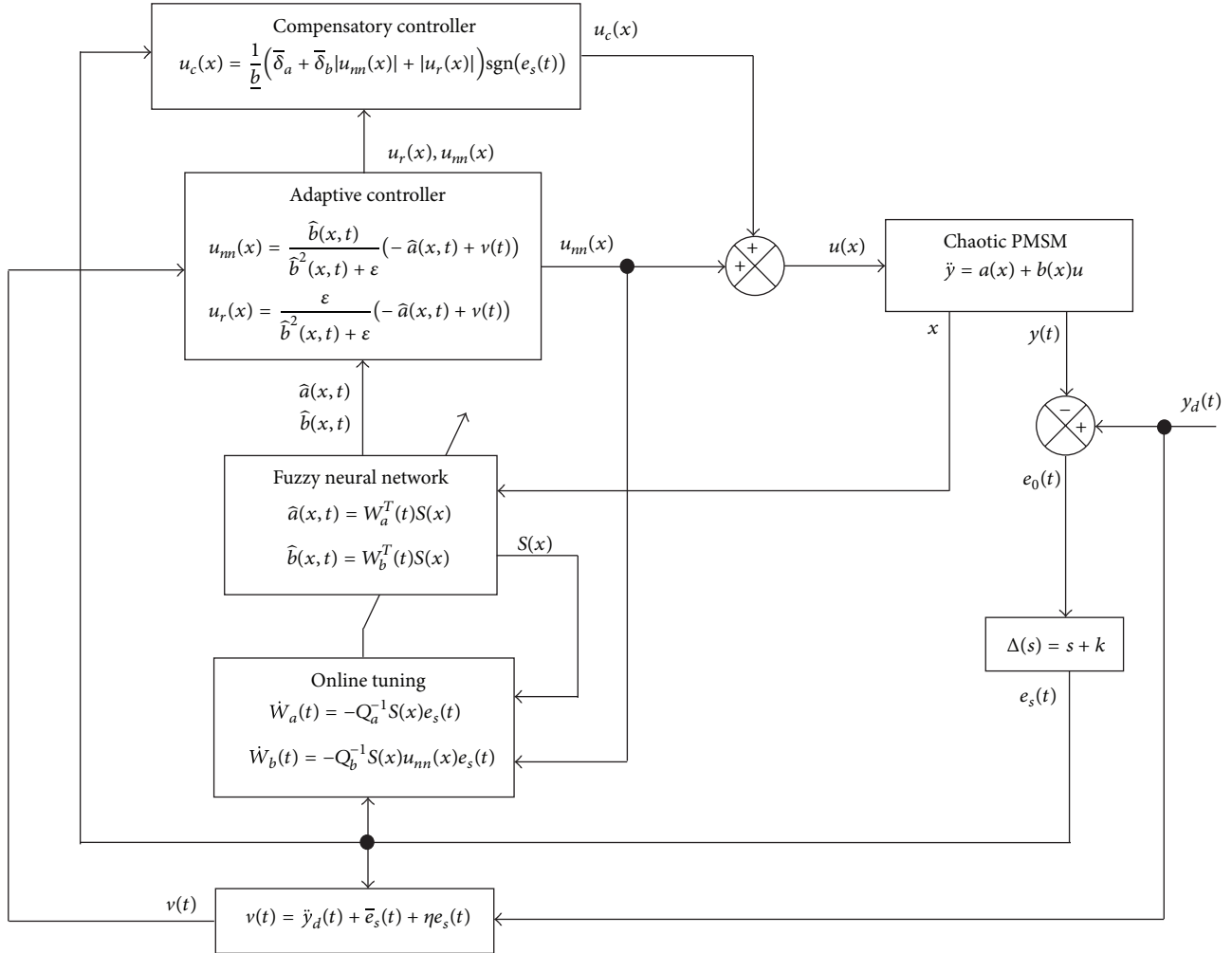
FIGURE 4: Overall scheme of the adaptive controller.

Therefore, the controller $u(x)$ has two control terms: the fuzzy neural controller $u_{nn}(x)$ and the compensatory controller $u_c(x)$. The overall scheme of the controller is illustrated in Figure 4 and the total control signal is given as

$$u(x) = u_{nn}(x) + u_c(x)$$

$$= u_{nn}(x) + \frac{1}{\underline{b}}\left(\overline{\delta}_a + \overline{\delta}_b\left|u_{nn}(x)\right| + \left|u_r(x)\right|\right)\mathrm{sgn}\left(e_s(t)\right)$$

$$= \frac{\widehat{b}(x,t)}{\widehat{b}^2(x,t) + \varepsilon}\left(-\widehat{a}(x,t) + v(t)\right) + \frac{\overline{\delta}_a}{\underline{b}}\,\mathrm{sgn}\left(e_s(t)\right)$$

$$+ \frac{\overline{\delta}_b}{\underline{b}}\left|\frac{\widehat{b}(x,t)}{\widehat{b}^2(x,t) + \varepsilon}\left(-\widehat{a}(x,t) + v(t)\right)\right|\mathrm{sgn}\left(e_s(t)\right)$$

$$+ \frac{1}{\underline{b}}\left|\frac{\varepsilon}{\widehat{b}^2(x,t) + \varepsilon}\left(-\widehat{a}(x,t) + v(t)\right)\right|\mathrm{sgn}\left(e_s(t)\right).$$

$$(30)$$

Theorem 5. *Consider the system in (3) and that the desired trajectory $y_d(t)$ satisfies Assumption 3. If Assumptions 1–4*

hold, then under the effect of controller (30) with the adaptive laws (25), chaos in the PMSM can be suppressed and its speed can asymptotically track the desired trajectory successfully.

Proof. Substituting (30) into (7) and using (22), we can obtain

$$\ddot{y}(t) = \dot{z}_2 = a(x) + b(x)\left(u_{nn}(x) + u_c(x)\right)$$

$$= a(x) + b(x)\,u_{nn}(x) + b(x)\,u_c(x)$$

$$= a(x) + \left(b(x) - \widehat{b}(x,t)\right)u_{nn}(x) + \widehat{b}(x,t)\,u_{nn}$$

$$+ b(x)\,u_c(x)$$

$$= a(x) + \left(b(x) - \widehat{b}(x,t)\right)u_{nn}(x)$$

$$\qquad\qquad\qquad\qquad\qquad\qquad (31)$$

$$+ \frac{\widehat{b}^2(x,t)}{\widehat{b}^2(x,t) + \varepsilon}\left(-\widehat{a}(x,t) + v(t)\right) + b(x)\,u_c(x)$$

$$= \left(a(x) - \widehat{a}(x,t)\right) + \left(b(x) - \widehat{b}(x,t)\right)u_{nn}(x)$$

$$+ v(t) - u_r(x) + b(x)\,u_c(x).$$

Taking the second-order derivative of (11) and then using (31) and (10), we can get

$$
\begin{aligned}
\ddot{e}_0(t) &= \ddot{y}_d(t) - \ddot{y}(t) \\
&= \ddot{y}_d(t) - v(t) + (\hat{a}(x,t) - a(x)) \\
&\quad + \left(\hat{b}(x,t) - b(x)\right) u_{nn}(x) + u_r(x) - b(x) u_c(x) \\
&= -\eta e_s(t) - \bar{e}_s(t) + (\hat{a}(x,t) - a(x)) \\
&\quad + \left(\hat{b}(x,t) - b(x)\right) u_{nn}(x) + u_r(x) - b(x) u_c(x).
\end{aligned}
\tag{32}
$$

From (12), (13), and (32), the error dynamics can be obtained as

$$
\begin{aligned}
\dot{e}_s(t) &= -\eta e_s(t) + (\hat{a}(x,t) - a(x)) + \left(\hat{b}(x,t) - b(x)\right) u_{nn}(x) \\
&\quad + u_r(x) - b(x) u_c(x).
\end{aligned}
\tag{33}
$$

In order to study the system stability, we consider the following Lyapunov function:

$$
V(x,t) = \frac{1}{2} e_s^2(t) + \frac{1}{2} \widetilde{W}_a^T(t) Q_a \widetilde{W}_a(t) + \frac{1}{2} \widetilde{W}_b^T(t) Q_b \widetilde{W}_b(t).
\tag{34}
$$

The time derivative of $V(x,t)$ with $\dot{\widetilde{W}}_a(t) = \dot{W}_a(t)$ and $\dot{\widetilde{W}}_b(t) = \dot{W}_b(t)$ can be computed as

$$
\begin{aligned}
\dot{V}(x,t) &= e_s(t) \dot{e}_s(t) + \frac{1}{2} \dot{\widetilde{W}}_a^T(t) Q_a \widetilde{W}_a(t) + \frac{1}{2} \widetilde{W}_a^T(t) Q_a \dot{\widetilde{W}}_a(t) \\
&\quad + \frac{1}{2} \dot{\widetilde{W}}_b^T(t) Q_b \widetilde{W}_b(t) + \frac{1}{2} \widetilde{W}_b^T(t) Q_b \dot{\widetilde{W}}_b(t) \\
&= e_s(t) \dot{e}_s(t) + \widetilde{W}_a^T(t) Q_a \dot{W}_a(t) + \widetilde{W}_b^T(t) Q_b \dot{W}_b(t).
\end{aligned}
\tag{35}
$$

Substituting (33) into (35), we can obtain

$$
\begin{aligned}
\dot{V}(x,t) &= e_s(t) \Big(-\eta e_s(t) + (\hat{a}(x,t) - a(x)) + \left(\hat{b}(x,t) - b(x)\right) \\
&\qquad\qquad \times u_{nn}(x) + u_r(x) - b(x) u_c(x) \Big) \\
&\quad + \widetilde{W}_a^T(t) Q_a \dot{W}_a(t) + \widetilde{W}_b^T(t) Q_b \dot{W}_b(t) \\
&= -\eta e_s^2(t) + e_s(t) (\hat{a}(x,t) - a(x)) \\
&\quad + e_s(t) \left(\hat{b}(x,t) - b(x)\right) u_{nn}(x) \\
&\quad + e_s(t) u_r(x) - e_s(t) b(x) u_c(x) \\
&\quad + \widetilde{W}_a^T(t) Q_a \dot{W}_a(t) + \widetilde{W}_b^T(t) Q_b \dot{W}_b(t).
\end{aligned}
\tag{36}
$$

Substituting (27) into (36) yields

$$
\begin{aligned}
\dot{V}(x,t) &= -\eta e_s^2(t) + e_s(t) (\hat{a}(x,t) - a(x)) \\
&\quad + e_s(t) \left(\hat{b}(x,t) - b(x)\right) u_{nn}(x) \\
&\quad + e_s(t) u_r(x) - e_s(t) b(x) u_c(x) \\
&\quad + \widetilde{W}_a^T(t) Q_a \dot{W}_a(t) + \widetilde{W}_b^T(t) Q_b \dot{W}_b(t) \\
&= -\eta e_s^2(t) + e_s(t) \left(\widetilde{W}_a^T(t) S(x) - \delta_a(x)\right) \\
&\quad + e_s(t) \left(\widetilde{W}_b^T(t) S(x) - \delta_b(x)\right) u_{nn}(x) \\
&\quad + \widetilde{W}_a^T(t) Q_a \dot{W}_a(t) + \widetilde{W}_b^T(t) Q_b \dot{W}_b(t) \\
&\quad + e_s(t) u_r(x) - e_s(t) b(x) u_c(x) \\
&= -\eta e_s^2(t) + \widetilde{W}_a^T(t) \left(e_s(t) S(x) + Q_a \dot{W}_a(t)\right) \\
&\quad + \widetilde{W}_b^T(t) \left(e_s(t) S(x) u_{nn}(x) + Q_b \dot{W}_b(t)\right) \\
&\quad - e_s(t) \delta_a(x) - e_s(t) \delta_b(x) u_{nn}(x) \\
&\quad + e_s(t) u_r(x) - e_s(t) b(x) u_c(x).
\end{aligned}
\tag{37}
$$

Using the adaptive laws in (25), (37) can be rewritten as

$$
\begin{aligned}
&\dot{V}(x,t) \\
&= -\eta e_s^2(t) - e_s(t) \delta_a(x) - e_s(t) \delta_b(x) u_{nn}(x) \\
&\quad + e_s(t) u_r(x) - e_s(t) b(x) u_c(x) \\
&\leq -\eta e_s^2(t) + |e_s(t)| \left(|\delta_a(x)| + |\delta_b(x)| |u_{nn}(x)| + |u_r(x)|\right) \\
&\quad - e_s(t) b(x) u_c(x) \\
&\leq -\eta e_s^2 + |e_s(t)| \left(\bar{\delta}_a + \bar{\delta}_b |u_{nn}(x)| + |u_r(x)|\right) \\
&\quad - e_s(t) b(x) u_c(x).
\end{aligned}
\tag{38}
$$

Substituting the compensatory controller in (29) into (38) and noting that $|e_s(t)| = \text{sgn}(e_s(t)) e_s(t)$, we can obtain

$$
\begin{aligned}
&\dot{V}(x,t) \\
&\leq -\eta e_s^2 + |e_s(t)| \left(\bar{\delta}_a + \bar{\delta}_b |u_{nn}(x)| + |u_r(x)|\right) \\
&\quad - \frac{b(x)}{\underline{b}} \left(\bar{\delta}_a + \bar{\delta}_b |u_{nn}(x)| + |u_r(x)|\right) \text{sgn}(e_s(t)) e_s(t) \\
&\leq -\eta e_s^2 - |e_s(t)| \left(\bar{\delta}_a + \bar{\delta}_b |u_{nn}(x)| + |u_r(x)|\right) \left(\frac{b(x)}{\underline{b}} - 1\right) \\
&\leq 0.
\end{aligned}
\tag{39}
$$

From (34) and (39), $V(x,t) > 0$ and $\dot{V}(x,t) \leq 0$ can be found, so the close-loop controlled system is stable. $e_s(t) \in L_\infty$, $\left\|\widetilde{W}_a(t)\right\| \in L_\infty$, and $\left\|\widetilde{W}_b(t)\right\| \in L_\infty$ are also determined.
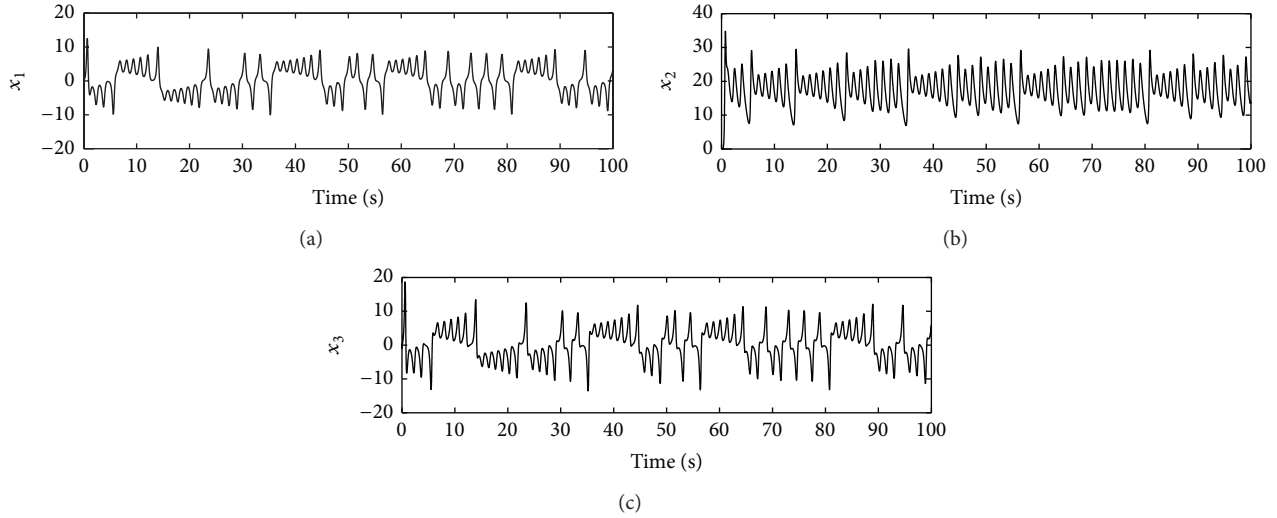
(a)

(b)

(c)

Figure 5: Chaotic oscillations of an uncontrolled PMSM.

Furthermore, using inequality in (39), we can get

$$\int_0^\infty \eta e_s^2(t)\, dt \le -\int_0^\infty \dot{V}(x,t)\, dt = V(x,0) - V(x,\infty) < \infty. \tag{40}$$

The above inequality, as in (40), implies that $e_s(t) \in L_2$, so $e_s(t) \in L_2 \cap L_\infty$ can be concluded. On the other hand, because $e_s(s) = \Delta(s)e_0(s)$, $e_0(t) \in L_\infty$, $\dot{e}_0(t) \in L_\infty$, and $\dot{e}_s(t) \in L_\infty$ can be obtained. Then, incorporating Barbalat's lemma [33] $\lim_{t\to\infty} e_s(t) = 0$ can be obtained, so $\lim_{t\to\infty} e_0(t) = 0$. Thus, the system stability is ensured and the advanced tracking performance is achieved. □

4. Numerical Simulations

Numerical results are given in this section to verify the proposed method. The system parameters and initial conditions are maintained as above; that is, $\sigma = 5.45$, $\gamma = 20$, and $[x_1(0)\ \ x_2(0)\ \ x_3(0)]^T = [1\ \ -1\ \ 0]^T$. First, the system without control action and uncertainties is considered. The simulation result points out that the state response falls into chaotic oscillations, as displayed in Figure 5.

Second, proposed controller is used to suppress chaos in the PMSM and track the desired speed under the effect of uncertainties. The uncertainties were chosen as $d_1 = 1 + \cos(t)$, $d_2 = -1$, and $d_3 = \sin(x_1)$ for simulation while the desired trajectory $y_d(t) = \sin(\pi t)$, which satisfies Assumption 3, is also chosen for this simulation. On the other hand, the control parameters are chosen as follows:

$$k = 40, \qquad \eta = 60, \qquad \underline{b} = 1, \qquad \varepsilon = 1, \qquad \overline{\delta}_a = \overline{\delta}_b = 0.1,$$

$$Q_a = 40 * \text{eye}(9), \qquad Q_b = 20 * \text{eye}(9). \tag{41}$$

The input states of the controller are normalized in a range of $[-1, 1]$ and nine membership functions, which relate to nine rules, are chosen as

$$\mu_{A_j^1}(x) = \exp\left[-\frac{(x+1)^2}{2*0.2^2}\right],$$

$$\mu_{A_j^2}(x) = \exp\left[-\frac{(x+0.75)^2}{2*0.2^2}\right],$$

$$\mu_{A_j^3}(x) = \exp\left[-\frac{(x+0.5)^2}{2*0.2^2}\right],$$

$$\mu_{A_j^4}(x) = \exp\left[-\frac{(x+0.25)^2}{2*0.2^2}\right],$$

$$\mu_{A_j^5}(x) = \exp\left[-\frac{(x+0)^2}{2*0.2^2}\right], \tag{42}$$

$$\mu_{A_j^6}(x) = \exp\left[-\frac{(x-0.25)^2}{2*0.2^2}\right],$$

$$\mu_{A_j^7}(x) = \exp\left[-\frac{(x-0.5)^2}{2*0.2^2}\right],$$

$$\mu_{A_j^8}(x) = \exp\left[-\frac{(x-0.75)^2}{2*0.2^2}\right],$$

$$\mu_{A_j^9}(x) = \exp\left[-\frac{(x-1)^2}{2*0.2^2}\right].$$

The results, as shown in Figures 6, 7, and 8, illustrate that the chaotic oscillations are removed and the speed of PMSM follows the desired trajectory perfectly while the tracking error converges to zero asymptotically.
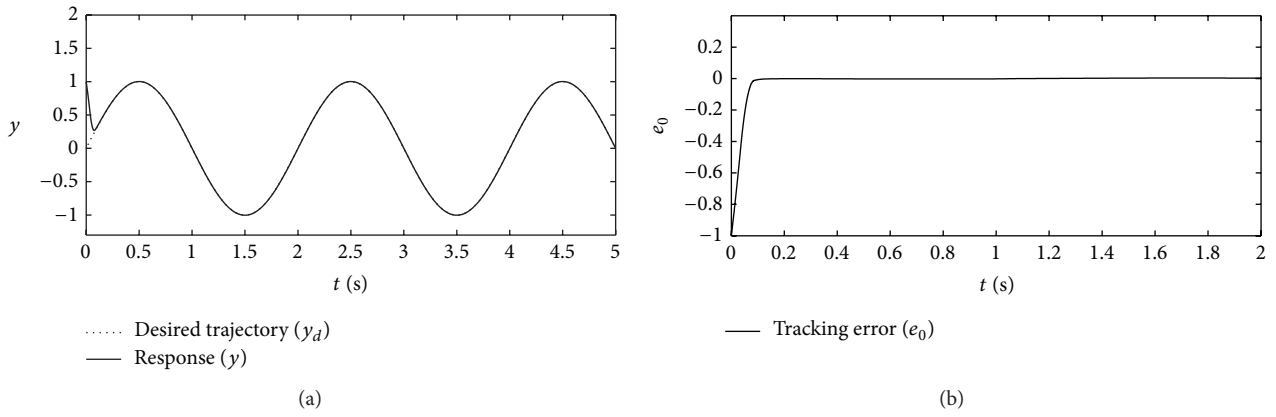
(a)

(b)

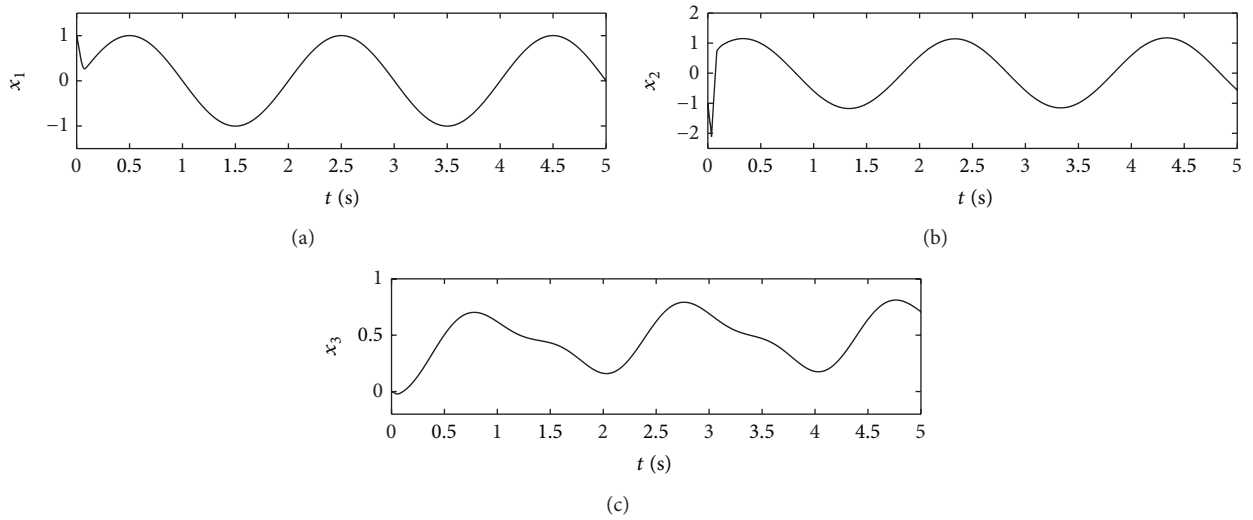FIGURE 6: Chaos suppression and speed tracking of the controlled PMSM.

(a)

(b)

(c)

FIGURE 7: State response of the controlled PMSM in x coordinate.
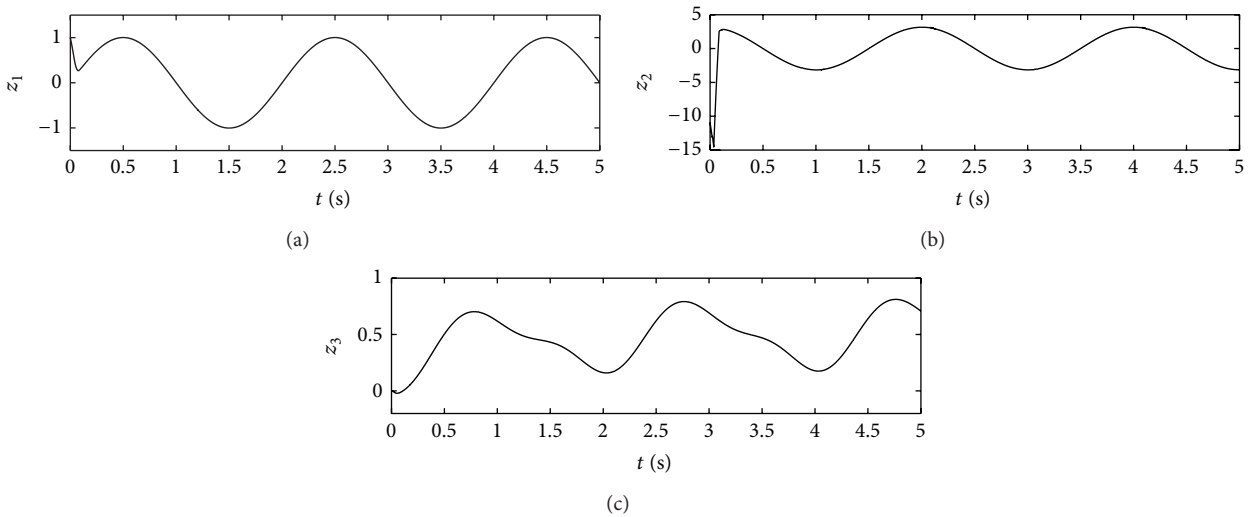

(a)

(b)

(c)

FIGURE 8: State response of the controlled PMSM in z coordinate.

5. Conclusion

In this paper, based on a fuzzy neural network, a new adaptive controller has been developed to suppress chaos and track the desired speed in a chaotic permanent magnet synchronous motor drive system. Derived from Lyapunov function, the stability of the system is ensured and the controller guarantees the perfect tracking performance where the tracking error converges to the origin even if uncertainties are applied to the system. The robustness and simple neural network structure can allow the controller to be feasibly applied to practical systems where the uncertainties are present. Simulation results are given to illustrate the effectiveness and robustness of the proposed method.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgment

References

[1] A. Bara, C. Rusu, and S. Dale, "DSP Application on PMSM drive control for robot axis," in *Proceedings of the 13th WSEAS International Conference on Systems (ICS '09)*, pp. 381–385, Rodos, Greece, July 2009.

[2] C.-H. Lin and C.-P. Lin, "The hybrid RFNN control for a PMSM drive electric scooter using rotor flux estimator," *Advances in Fuzzy Systems*, vol. 2012, Article ID 319828, 11 pages, 2012.

[3] Z. Lu, X. Sun, and J. Zhang, "Design and control of disc PMSM directly driven wheel for tramcar," *Advances in Mechanical Engineering*, vol. 2014, Article ID 747636, 8 pages, 2014.

[4] P. Pillay and R. Krishnan, "Control characteristics and speed controller design for a high performance permanent magnet synchronous motor drive," *IEEE Transactions on Power Electronics*, vol. 5, no. 2, pp. 151–159, 1989.

[5] C. M. Verrelli, "Synchronization of permanent magnet electric motors: new nonlinear advanced results," *Nonlinear Analysis: Real World Applications*, vol. 13, no. 1, pp. 395–409, 2012.

[6] Y. Yu, Z. Mi, Y. Xu, and T. Zhao, "Global multivariable control of permanent magnet synchronous motor for mechanical elastic energy storage system under multiclass nonharmonic external disturbances," *Abstract and Applied Analysis*, vol. 2014, Article ID 410282, 9 pages, 2014.

[7] Y. Kuroe and S. Hayashi, "Analysis of bifurcation in power electronic induction motor drive systems," in *Proceedings of the 20th Annual IEEE Power Electronics Specialists Conference (PESC '89)*, pp. 923–930, Milwankee, June 1989.

[8] J. H. Chen, K. T. Chau, and C. C. Chan, "Analysis of chaos in current-mode-controlled dc drive systems," *IEEE Transactions on Industrial Electronics*, vol. 47, no. 1, pp. 67–76, 2000.

[9] Z. Suto, I. Nagy, and E. Masada, "Avoiding chaotic processes in current control of AC drive," in *Proceedings of the 29th Annual IEEE Power Electronics Specialists Conference (PESC '98)*, vol. 1, pp. 255–261, Fukuoka, Japan, May 1998.

[10] Z. Li, J. B. Park, Y. H. Joo, B. Zhang, and G. Chen, "Bifurcations and chaos in a permanent-magnet synchronous motor," *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 49, no. 3, pp. 383–387, 2002.

[11] H. Ren and D. Liu, "Nonlinear feedback control of chaos in permanent magnet synchronous motor," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 53, no. 1, pp. 45–50, 2006.

[12] C. Ma, L. Wang, Z. Yin, J. Liu, and D. Chen, "Sliding mode control of chaos in the noise-perturbed permanent magnet synchronous motor with non-smooth air-gap," *Mining Science and Technology*, vol. 21, no. 6, pp. 835–838, 2011.

[13] C.-F. Huang, J.-S. Lin, T.-L. Liao, C.-Y. Chen, and J.-J. Yan, "Quasi-sliding mode control of chaos in permanent magnet synchronous motor," *Mathematical Problems in Engineering*, vol. 2011, Article ID 964240, 10 pages, 2011.

[14] C.-F. Huang, T.-L. Liao, C.-Y. Chen, and J.-J. Yan, "The design of quasi-sliding mode control for a permanent magnet synchronous motor with unmatched uncertainties," *Computers and Mathematics with Applications*, vol. 64, no. 5, pp. 1036–1043, 2012.

[15] X. Ge and J. Huang, "Chaos control of permanent magnet synchronous motor," in *Proceedings of the 8th International Conference on Electrical Machines and Systems (ICEMS '05)*, vol. 1, pp. 484–488, Nanjing, China, September 2005.

[16] Y. Gao and K. T. Chau, "Chaotification of permanent-magnet synchronous motor drives using time-delay feedback," in *Proceedings of the 28th Annual Conference of the IEEE Industrial Electronics Society (IECON '02)*, pp. 762–766, Sevilla, Spain, November 2002.

[17] F. Beaufays, Y. Abdel-Magid, and B. Widrow, "Application of neural networks to load-frequency control in power systems," *Neural Networks*, vol. 7, no. 1, pp. 183–194, 1994.

[18] N. G. Chalhoub and B. A. Bazzi, "Fuzzy logic control for an integrated system of a micro-manipulator with a single flexible beam," *Journal of Vibration and Control*, vol. 10, no. 5, pp. 755–776, 2004.

[19] X. P. Cheng and R. V. Patel, "Neural network based tracking control of a flexible macro-micro manipulator system," *Neural Networks*, vol. 16, no. 2, pp. 271–286, 2003.

[20] H.-W. Ge, F. Qian, Y.-C. Liang, W.-l. Du, and L. Wang, "Identification and control of nonlinear systems by a dissimilation particle swarm optimization-based Elman neural network," *Nonlinear Analysis: Real World Applications*, vol. 9, no. 4, pp. 1345–1360, 2008.

[21] W. Qin, Y. Yang, and J. Zhang, "Controlling the chaotic response to a prospective external signal using back-propagation neural networks," *Nonlinear Analysis: Real World Applications*, vol. 10, no. 5, pp. 2985–2989, 2009.

[22] H.-T. Yau, C.-C. Wang, C.-T. Hsieh, and C.-C. Cho, "Nonlinear analysis and control of the uncertain micro-electro-mechanical system by using a fuzzy sliding mode control design," *Computers and Mathematics with Applications*, vol. 61, no. 8, pp. 1912–1916, 2011.

[23] L. X. Wang, *A Course in Fuzzy Systems and Control*, Prentice Hall, 1997.

[24] P.-Y. Couzon and J. der Hagopian, "Neuro-fuzzy active control of rotor suspended on active magnetic bearing," *Journal of Vibration and Control*, vol. 13, no. 4, pp. 365–384, 2007.

[25] C.-T. Lin and C. S. G. Lee, *Neural Fuzzy Systems: A Neuro-Fuzzy Synergism to Intelligent Systems*, Prentice-Hall, Upper Saddle River, NJ, USA, 1996.

[26] O. Omidvar and D. L. Elliott, *Neural Systems for Control*, Academic Press, New York, NY, USA, 1997.

[27] Y.-C. Wang and C.-J. Chien, "An observer-based adaptive iterative learning control using filtered-FNN design for robotic systems," *Advances in Mechanical Engineering*, vol. 2014, Article ID 471418, 14 pages, 2014.

[28] X. Zhang and L. Mu, "Command filtered adaptive fuzzy neural network backstepping control for marine power system," *Mathematical Problems in Engineering*, vol. 2014, Article ID 461431, 6 pages, 2014.

[29] Y. Guo and H. Long, "Self organizing fuzzy sliding mode controller for the position control of a permanent magnet synchronous motor drive," *Ain Shams Engineering Journal*, vol. 2, no. 2, pp. 109–118, 2011.

[30] Y.-Y. Hou, "Controlling chaos in permanent magnet synchronous motor control system via fuzzy guaranteed cost controller," *Abstract and Applied Analysis*, vol. 2012, Article ID 650863, 10 pages, 2012.

[31] Y. Shen, Y. Zhou, and Z. Ji, "Fuzzy guaranteed cost control of permanent magnet synchronous motor chaos systems via fuzzy lyapunov function approach," in *Proceedings of the 2nd IEEE Conference on Industrial Electronics and Applications (ICIEA '07)*, pp. 2449–2454, Harbin, China, May 2007.

[32] J. Yu, B. Chen, H. Yu, and J. Gao, "Adaptive fuzzy tracking control for the chaotic permanent magnet synchronous motor drive system via backstepping," *Nonlinear Analysis: Real World Applications*, vol. 12, no. 1, pp. 671–681, 2011.

[33] J. J. E. Slotine and W. Li, *Applied Nonlinear Control*, Pearson Education Taiwan, New Taipei, Taiwan, 2005.