IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received June 24, 2020, accepted June 29, 2020, date of publication July 2, 2020, date of current version July 15, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3006469

A Modified Grey Wolf Optimizer for Optimum Parameters of Multilayer Type-2 Asymmetric Fuzzy Controller

TIEN-LOC LE[1,2], **(Member, IEEE), TUAN-TU HUYNH**[2,3], **(Member, IEEE), AND SUNG KYUNG HONG**[1]

[1]Faculty of Mechanical and Aerospace Engineering, Sejong University, Seoul 05006, South Korea
[2]Department of Electrical, Electronics, and Mechanical Engineering, Lac Hong University, Bien Hoa 810000, Vietnam
[3]Department of Electrical Engineering, Yuan Ze University, Taoyuan City 320, Taiwan

Corresponding author: Sung Kyung Hong (skhong@sejong.ac.kr)

This work was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (2020R1A6A1A03038540) and by the MSIT(Ministry of Science and ICT), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2020-2018-0-01423) supervised by the IITP (Institute for Information & communications Technology Promotion).

ABSTRACT This study presents a modified algorithm of the grey wolf optimizer to solve the problem of learning rate selection in the multilayer type-2 asymmetric fuzzy controller (MT2AFC). The improvements of our modified optimizer are: the best position of the swarm is memorized, thus making the alpha wolves only update when a better position appears in the next iteration; search performance is enhanced by giving more freedom to update the grey wolf position. The proposed optimizer algorithm is then applied to optimize the suitable learning rates for the proposed controller. The multilayer type-2 asymmetric membership function is used in the fuzzy control network to enhance the learning ability and flexibility of the designed network architecture. The gradient descent method is used to adjust the parameters of the proposed MT2AFC controller online. The stability of the system is guaranteed using the Lyapunov approach. Besides, the self-evolving algorithm is used to construct the network structure autonomously. Ultimately, the numerical simulations of the chaotic synchronization systems are carried out to verify the effectiveness of our proposed method.

INDEX TERMS Grey wolf optimizer, interval type-2 fuzzy neural network, asymmetric membership function, chaotic synchronization systems.

I. INTRODUCTION

In the past decade, some studies have shown type-2 fuzzy logic systems (T2FLSs) have better uncertainty handling capacity than type-1 fuzzy logic systems (T1FLSs) [1]–[5]. The membership degrees in T1FLSs are crisp numbers, whereas the membership degrees in T2FLSs are fuzzy membership grades. By including the uncertainty terms in the membership function, the T2FLSs can better handle higher degrees of uncertainty than T1FLSs [6]. However, the T2FLSs come with more complex rules and computational cost. Therefore, the interval type-2 fuzzy logic systems (IT2FLSs) in [7] have been proposed to reduce the computation operations for T2FLSs. Some notable studies in the past decade on T1FLSs and IT2FLSs are given in [8]–[18]. In 2019, Roman *et al.* proposed a combined model-free adaptive control with fuzzy component [8]. In 2020, Mohammadzadeh and E. Kayacan presented a novel fractional-order type-2 fuzzy control method for online frequency regulation in ac microgrid [11]. In 2020, Moreno *et al.* introduced an interval type-2 fuzzy model with justifiable uncertainty [18]. Recently, many studies used symmetric membership functions in the IT2FLSs because their adaptive laws are easily designed. The studies in [19]–[21] presented type-2 asymmetric membership functions (T2AFM) for IT2FLSs, which is supposed to increase system accuracy. A T2AMF comprises two Gaussian upper membership functions and two Gaussian lower membership functions.

This paper proposes a multi-layer structure for T2AFM to enhance the learning ability and flexibility of the IT2FLSs. With this structure, the proposed network inputs will be better covered. The number of rules will be reduced compared with all membership functions into one layer. Moreover, splitting membership functions into different layers allows easy adding of new required membership functions or removal of unused membership functions,

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

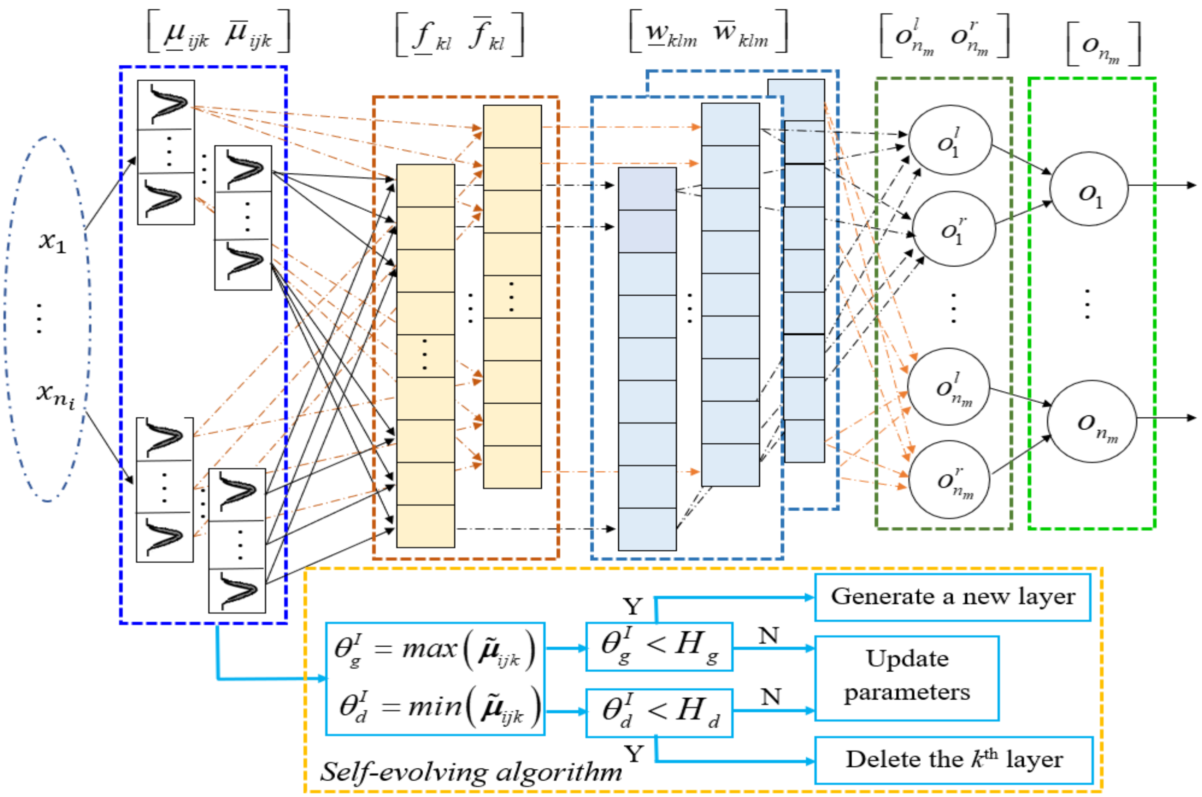
while still satisfying the number of member functions required for the network operation. The operation of generating and deleting member functions is called a self-evolving algorithm. The self-evolving and self-organizing algorithm has recently been used in many studies to construct the network structure autonomously [22]–[28]. In 2017, Chen and Liu proposed an intelligent tracking control of a PMLSM using a self-evolving probabilistic fuzzy neural network [22]. In 2018, Lin *et al.* introduced a self-evolving function-link interval type-2 fuzzy neural network for nonlinear system identification and control [25]. In addition, in 2018, Qiao *et al.* presented a self-organizing deep belief network for nonlinear system modeling [27].

The study in [29] shows, in the gradient descent based updating method, the learning rate for adaptive laws significantly affects system performance. If the chosen learning rate is large, the learning process may involve oscillating convergence or even divergence. Contrarily, if the learning rate is too small, the system is slow to converge or is even trapped in the local minimum. Many studies used the trial-and-error method to obtain the learning rates, but this method often takes a long time to obtain the most suitable values. To tackle this problem, the studies in [30], [31] proposed particle swarm optimization (PSO) to promote the parameter learning efficiency. However, the PSO algorithm also has some disadvantages, such as premature convergence and slow convergence in the iterative process. Therefore, this study designed a modified algorithm of a grey wolf optimizer (MGWO) to optimize the learning rates for the adaptive laws of the proposed MT2AFC controller. The grey wolf optimizer (GWO) was first introduced by Mirjalili in 2014, which is a type of meta-heuristic algorithm based on the grey wolf community hierarchy and hunting mechanisms [32]. Recently, some notable studies on different GWO variants have been presented to better respond to the complex search space of optimization problems [33]–[41]. In 2018, Abdo *et al.* presented a developed GWO for solving non-smooth optimal power flow problems [33]. In 2019, Khanum *et al.* proposed two new improved variants of GWO for unconstrained optimization [35]. In 2020, Gupta and Deep presented a memory-based GWO for global optimization tasks [40]. According to [41], the variants GWO can be listed as modified versions, hybridized versions, parallel versions, and multi-objective versions. In which, the modified versions of GWO can be categorized into four types such as updating mechanisms, new operators, encoding scheme of the individuals, population structure and hierarchy. This study improved the conventional GWO algorithm by adding a global memory variable to save the best position of the swarm and adding a multidimensional random coefficient in the swarm position update formula. These enhancements make the MGWO algorithm more flexible and converge more rapidly.

In the past decade, besides PSO and GWO, many other metaheuristic algorithms have been proposed, such as evolutionary algorithms (genetic algorithm [42], [43], differential evolution [44], [45], shuffled complex evolution [46], [47]), physics-based algorithms (wind driven optimization [48], [49], flower pollination algorithm [50], [51], gravitational search algorithm [52], [53]), swarm-based algorithms (artificial bee colony [54], [55], cat swarm optimization [56], [57], whale optimization algorithm [58], [59], grasshopper optimization [60], [61], moth-flame optimization [62], [63]), and human-based algorithms (harmony search [64], [65], human behavior-based optimization [66], [67]). However, in metaheuristic algorithms, balancing the exploration and exploitation operation is the biggest challenge [68]. The exploration operation expands the search space to find more promising regions, whereas the exploitation evaluates the solutions to obtain optimal solutions within the search space. Some metaheuristic algorithms have good exploration ability, but their convergence speed is low. In contrast, some metaheuristic algorithms have good exploitation ability with rapid convergence speed, but they are easily trapped in a local minimum [69]. Therefore, in recent years, developing a meta-heuristic algorithm with optimal exploration and exploitation operation has attracted the attention of many researchers.

Synchronization chaotic systems have attracted great interest because of their potential applications in various fields such as electronics, signal processing, secure communication, aerospace, and biological systems [70]–[77]. Synchronization of chaotic systems is the control of the slave system to mimic the behavior of the master system. It is a very challenging topic in nonlinear sciences because chaotic systems have nonlinear behavior and are often sensitive to the initial values [78]. In the past decade, notable synchronization methods of chaotic systems have been introduced such as the fuzzy adaptive controller, sliding mode controller, type-2 fuzzy controller, and the cerebellar model articulation controller [79]–[84]. However, most of the above methods have complicated implementation, and system performance can be further improved.

Based on the above discussion, we propose a new method to synchronize the chaotic systems, combining the advantages of interval type-2 fuzzy logic systems, asymmetric membership functions, the multi-layer structure, the modified grey wolf optimizer, and the self-evolving algorithm. The novelty of this study is the design of a multilayer structure for the type-2 asymmetric fuzzy controller, which uses the self-evolving algorithm to autonomously construct the network structure. Moreover, a new modified grey wolf optimizer is proposed for optimizing the learning rates of the parameter adaptive laws. Compared with existing research in [24]–[26], the proposed network has some advantages, such as the initial network structure being easy to design and rapidly obtaining suitable learning rates. The main contributions of the proposed modified grey wolf optimizer MT2AFC (MMT2AFC) synchronization system are (1) A new modified grey wolf optimizer is proposed for optimizing the learning rates for the proposed network; (2) A new multi-layer structure is proposed to enhance the learning ability and flexibility of the IT2FLSs; (3) A self-evolving algorithm is



$[\underline{\mu}_{ijk} \ \overline{\mu}_{ijk}]$
$[\underline{f}_{kl} \ \overline{f}_{kl}]$
$[\underline{w}_{klm} \ \overline{w}_{klm}]$
$[o^l_{n_m} \ o^r_{n_m}]$
$[o_{n_m}]$
x_1
x_{n_i}
o^l_1
o^r_1
$o^l_{n_m}$
$o^r_{n_m}$
o_1
o_{n_m}
$\theta^I_g = max(\tilde{\mu}_{ijk})$
$\theta^I_d = min(\tilde{\mu}_{ijk})$
$\theta^I_g < H_g$
$\theta^I_d < H_d$
Y
N
N
Y
Generate a new layer
Update parameters
Delete the k^{th} layer
Self-evolving algorithm

FIGURE 1. The architecture of the MMT2AFC control system.

used to autonomously construct the network structure; (4) An asymmetric membership function is used to enhance system accuracy; (5) The numerical results of the chaotic synchronization system are presented to verify the validity of the proposed method.

The remainder of this paper is organized as follows. Section II presents the details of the proposed MMT2AFC controller and its parameter learning. Section III provides the structure learning algorithm and the modified grey wolf optimizer. Section IV explains the simulation results for the chaotic synchronization systems. Finally, Section V offers conclusions.

II. STRUCTURE of MMT2AFC

A. MULTILAYER INTERVAL TYPE-2 FUZZY WITH ASYMMETRIC MEMBERSHIP FUNCTIONS

The l^{th} fuzzy inference rule for the MMT2AFC is presented as follows:

$$\begin{aligned} &Rule\ l:\ IF\ x_1\ is\ \tilde{\mu}_{1jk}\ and\ x_2\ is\ \tilde{\mu}_{2jk}, \ldots, and\ x_{n_i}\ is\ \tilde{\mu}_{n_ijk} \\ &\qquad Then\ \tilde{w}_{klm} = \left[\underline{w}_{klm} \quad \overline{w}_{klm}\right] \end{aligned} \tag{1}$$

where $\boldsymbol{X} = \left[x_1, x_2, \ldots, x_{n_i}\right]^T \epsilon \Re^{n_i}$ is the input vector; $i = 1, 2, \ldots, n_i$ is the number of the input; $j = 1, 2, \ldots, n_j$ is the number of blocks in each layer; $k = 1, 2, \ldots, n_k$ is the number of layers; $l = 1, 2, \ldots, n_l$ is the number of rulers; $\tilde{\mu}_{ijk}$ is the membership function for the j^{th} block of the k^{th} layer in the i^{th} input; $\tilde{w}_{klm}$ is presented for the consequent part of the fuzzy systems, which is used to the connect the m^{th} output with the l^{th} firing node in the k^{th} layer.

As shown in Fig. 1, the structure of the proposed MMT2AFC network is formed by five spaces. The final output of the proposed controller can be represented as follows:

$$\begin{aligned} &u^m_{MMT2AFC} \\ &= o_m = \frac{1}{2}\left(\overline{o}_m + \underline{o}_m\right) \\ &= \frac{1}{2}\left[q\sum_{k=1}^{n_k}\sum_{l=1}^{n_l}\left(\overline{f}_{kl}\overline{w}_{klm}\right) + (1-q)\sum_{k=1}^{n_k}\sum_{l=1}^{n_l}\left(\underline{f}_{kl}\underline{w}_{klm}\right)\right] \end{aligned} \tag{2}$$

where q is the weighting gain for adjusting the contribution of the lower and upper membership functions; $\overline{w}_{klm}$ and $\underline{w}_{klm}$ respectively are the upper and lower fuzzy connecting weight, which are expressed as

$$\begin{aligned} &\begin{bmatrix} \overline{\boldsymbol{w}}_1 \\ \overline{\boldsymbol{w}}_2 \\ \vdots \\ \overline{\boldsymbol{w}}_{n_m} \end{bmatrix} \\ &= \begin{bmatrix} \overline{w}_{111}\ldots\overline{w}_{1n_l1}; & \overline{w}_{211}\ldots\overline{w}_{2n_l1}; & \ldots & ;\overline{w}_{n_k11}\ldots\overline{w}_{n_kn_l1} \\ \overline{w}_{112}\ldots\overline{w}_{1n_l2}; & \overline{w}_{212}\ldots\overline{w}_{2n_l2}; & \ldots & ;\overline{w}_{n_k12}\ldots\overline{w}_{n_kn_l2} \\ \vdots & \vdots & \vdots & \vdots \\ \overline{w}_{11n_m}\ldots\overline{w}_{1n_ln_m}; & \overline{w}_{21n_m}\ldots\overline{w}_{2n_ln_m}; & \ldots & ;\overline{w}_{n_k1n_m}\ldots\overline{w}_{n_kn_ln_m} \end{bmatrix} \end{aligned} \tag{3}$$

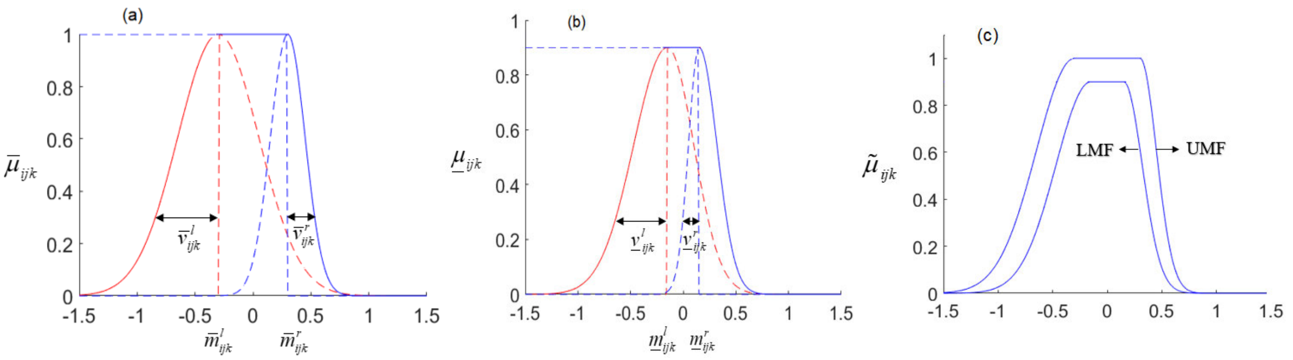


FIGURE 2. **The asymmetric membership function of the MMT2AFC control system. (a) The upper MF; (b) The lower MF; (c) The combined MF.**

$$\begin{bmatrix} \underline{\boldsymbol{w}}_1 \\ \underline{\boldsymbol{w}}_2 \\ \vdots \\ \underline{\boldsymbol{w}}_{n_m} \end{bmatrix} = \begin{bmatrix} \underline{w}_{111}\cdots\underline{w}_{1n_l1}; & \underline{w}_{211}\cdots\underline{w}_{2n_l1}; & \cdots & ;\underline{w}_{n_k11}\cdots\underline{w}_{n_kn_l1} \\ \underline{w}_{112}\cdots\underline{w}_{1n_l2}; & \underline{w}_{212}\cdots\underline{w}_{2n_l2}; & \cdots & ;\underline{w}_{n_k12}\cdots\underline{w}_{n_kn_l2} \\ \vdots & \vdots & \vdots & \vdots \\ \underline{w}_{11n_m}\cdots\underline{w}_{1n_ln_m}; & \underline{w}_{21n_m}\cdots\underline{w}_{2n_ln_m}; & \cdots & ;\underline{w}_{n_k1n_m}\cdots\underline{w}_{n_kn_ln_m} \end{bmatrix} \tag{4}$$

$\bar{f}_{kl}$ and $\underline{f}_{kl}$ respectively are the upper and lower fuzzy firing strength, which are defined as

$$\bar{f}_{kl} = \prod_{i=1}^{n_i} \bar{\mu}_{ijk} \text{ and } \underline{f}_{kl} = \prod_{i=1}^{n_i} \underline{\mu}_{ijk} \tag{5}$$

where $\bar{\mu}_{ijk}$ and $\underline{\mu}_{ijk}$ respectively are the upper and lower type-2 asymmetric membership function, which are illustrated in Fig. 2 and can be represented as follows:

$$\bar{\mu}_{ijk} = \begin{cases} \exp\left\{\dfrac{-\left(x_i - \bar{m}^l_{ijk}\right)^2}{2\left(\bar{v}^l_{ijk}\right)^2}\right\}, & x_i \le \bar{m}^l_{ijk} \\ 1, & \bar{m}^l_{ijk} \le x_i \le \bar{m}^r_{ijk} \\ \exp\left\{\dfrac{-\left(x_i - \bar{m}^r_{ijk}\right)^2}{2\left(\bar{v}^r_{ijk}\right)^2}\right\}, & \bar{m}^r_{ijk} \le x_i \end{cases} \tag{6}$$

$$\underline{\mu}_{ijk} = \begin{cases} \theta \exp\left\{\dfrac{-\left(x_i - \underline{m}^l_{ijk}\right)^2}{2\left(\underline{v}^l_{ijk}\right)^2}\right\}, & x_i \le \underline{m}^l_{ijk} \\ \theta, & \underline{m}^l_{ijk} \le x_i \le \underline{m}^r_{ijk} \\ \theta\exp\left\{\dfrac{-\left(x_i - \underline{m}^r_{ijk}\right)^2}{2\left(\underline{v}^r_{ijk}\right)^2}\right\}, & \underline{m}^r_{ijk} \le x_i \end{cases} \tag{7}$$

where $\bar{m}^l_{ijk}$, $\bar{m}^r_{ijk}$ and $\underline{m}^l_{ijk}$, $\underline{m}^r_{ijk}$ are the means of two upper and two lower Gaussian MFs, respectively; $\bar{v}^l_{ijk}$, $\bar{v}^r_{ijk}$ and $\underline{v}^l_{ijk}$, $\underline{v}^r_{ijk}$ are the variance of two upper and two lower Gaussian MFs, respectively;

To avoid unreasonable MFs, the following constraints should be added.

$$\begin{cases} \bar{m}^l_{ijk} \le \underline{m}^l_{ijk} \le \underline{m}^r_{ijk} \le \bar{m}^r_{ijk} \\ \underline{v}^l_{ijk} \le \bar{v}^l_{ijk} \le \underline{v}^r_{ijk} \le \bar{v}^r_{ijk} \\ 0.5 \le \theta \le 1 \end{cases} \tag{8}$$

Figure 3a illustrates the fuzzy rule creation mechanism for the conventional type-2 fuzzy system, in which all member functions are placed in one layer; therefore, the number of fuzzy rules is formed by the multiple of the number of member functions in the inputs. The structure of our proposed controller splits the membership functions into several layers, as illustrated in Fig. 3b. Therefore, the number of rules can be reduced significantly while still ensuring the full member functions information.

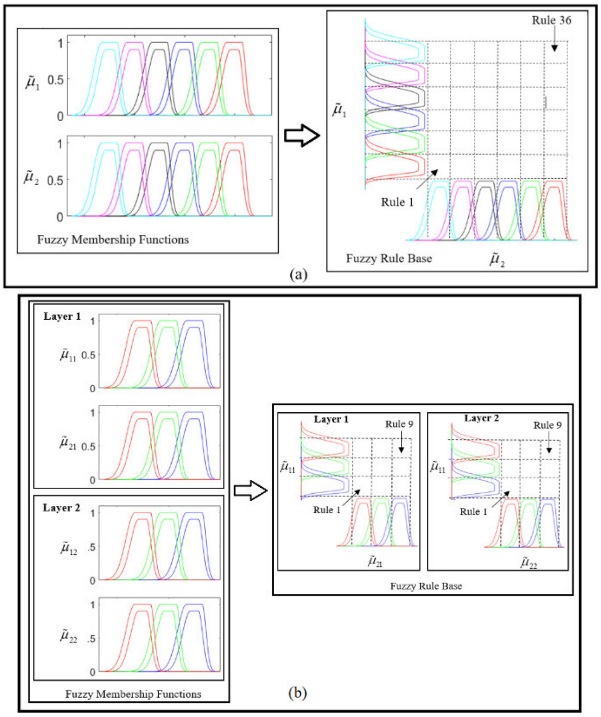


FIGURE 3. **The membership functions and rule base illustrative example. (a) The conventional type-2 fuzzy system. (b) The proposed MMT2AFC system with two layers.**

B. PARAMETER LEARNING

Considering the following uncertain nonlinear system:

$$\begin{aligned} x^{(n)}(t) &= A\left(\boldsymbol{x}(t)\right) + B\left(\boldsymbol{x}(t)\right)u(t) + d(t) \\ &= A_0\left(\boldsymbol{x}\left(t\right)\right) + \Delta A\left(\boldsymbol{x}\left(t\right)\right) \\ &\quad + \left[B_0\left(\boldsymbol{x}\left(t\right)\right) + \Delta B\left(\boldsymbol{x}\left(t\right)\right)\right]u\left(t\right) + d\left(t\right) \\ &= A_0\left(\boldsymbol{x}\left(t\right)\right) + B_0\left(\boldsymbol{x}\left(t\right)\right)u\left(t\right) + \xi\left(\boldsymbol{x}\left(t\right)\right) \end{aligned} \tag{9}$$

In which,

$$\xi\left(\boldsymbol{x}\left(t\right)\right) = \Delta f\left(\boldsymbol{x}\left(t\right)\right) + \Delta g\left(\boldsymbol{x}\left(t\right)\right)u\left(t\right) + d\left(t\right) \tag{10}$$

where $x(t)$ and $u(t)$ respectively are the system output and the control input; $A\left(\boldsymbol{x}\right), B\left(\boldsymbol{x}\right)$ and $A_0\left(\boldsymbol{x}\left(t\right)\right), B_0\left(\boldsymbol{x}\left(t\right)\right)$ respectively are the bounded nonlinear functions and its nominal parts; $\Delta A\left(\boldsymbol{x}\left(t\right)\right)$ and $\Delta B\left(\boldsymbol{x}\left(t\right)\right)$ respectively are the unknown uncertainties of $A\left(\boldsymbol{x}\right)$ and $B\left(\boldsymbol{x}\right)$; $d(t)$ and $\xi\left(\boldsymbol{x}\left(t\right)\right)$ respectively are the external disturbance and the lumped uncertainty term.

First, considering the system tracking error vector $\boldsymbol{e}\left(t\right) = \left[e\left(t\right), \dot{e}\left(t\right), \ldots, e^{(n-1)}\left(t\right)\right]^T$, where $e\left(t\right)$ is the tracking error between the desired signal $x_d\left(t\right)$ and the system output $x(t)$ as follows:

$$e\left(t\right) = x_d\left(t\right) - x\left(t\right) \tag{11}$$

From (9), if the lumped uncertainty term $\xi\left(\boldsymbol{x}\left(t\right)\right)$ and the nominal parts $A_0\left(\boldsymbol{x}\left(t\right)\right)$, $B_0\left(\boldsymbol{x}\left(t\right)\right)$ are obtained exactly, the ideal controller can be obtained as:

$$\begin{aligned} u^*\left(t\right) = B_0^{-1}\left(\boldsymbol{x}\left(\mathrm{t}\right)\right)\Big[x_d^{(n)}\left(t\right) - A_0\left(\boldsymbol{x}\left(t\right)\right) \\ - \xi\left(\boldsymbol{x}\left(t\right)\right) + \boldsymbol{G}^T\boldsymbol{e}\left(t\right)\Big] \end{aligned} \tag{12}$$

where $\boldsymbol{G}^T$ is the feedback gain vector.

Applying (12) into (9), the error dynamics is obtained as:

$$e^{(n)} + \boldsymbol{G}^T\boldsymbol{e} = 0 \tag{13}$$

If $\boldsymbol{G}$ is chosen to satisfy the Hurwitz stability for (13), then $\lim_{t\to\infty} e\left(t\right) = 0$. However, the uncertainty term $\xi\left(\boldsymbol{x}\left(t\right)\right)$ is unknown or perturbed, therefore, we approximate the ideal controller $u^*\left(t\right)$ by an optimal MMT2AFC controller as follows:

$$\begin{aligned} \boldsymbol{u}^*(t) = \boldsymbol{u}^*_{MMT2AFC}(\underline{w}^*, \overline{w}^*, \underline{m}^{l*}, \overline{m}^{l*}, \underline{m}^{r*}, \overline{m}^{r*}, \\ \underline{v}^{l*}, \overline{v}^{l*}, \underline{v}^{r*}, \overline{v}^{r*}, t) + \varsigma(t) \end{aligned} \tag{14}$$

where $\varsigma(t)$ is the approximation error; $\underline{w}^*, \overline{w}^*, m^{l*}, \overline{m}^{l*}, \underline{m}^{r*}, \overline{m}^{r*}, \underline{v}^{l*}, \overline{v}^{l*}, \underline{v}^{r*}, \overline{v}^{r*}$ are the optimal parameters for $\underline{w}, \overline{w}, \underline{m}^l, \overline{m}^l, \underline{m}^r, \overline{m}^r, \underline{v}^l, \overline{v}^l, \underline{v}^r, \overline{v}^r$.

Since the $\boldsymbol{u}^*_{MMT2AFC}$ cannot be obtained, an estimation controller, $\hat{\boldsymbol{u}}_{MMT2AFC}$, is employed to estimate it as follows:

$$\begin{aligned} \hat{\boldsymbol{u}}\left(t\right) = \hat{\boldsymbol{u}}_{MMT2AFC}(\underline{\hat{w}}, \hat{\overline{w}}, \underline{\hat{m}}^l, \hat{\overline{m}}^l, \underline{\hat{m}}^r, \hat{\overline{m}}^r, \underline{\hat{v}}^l, \hat{\overline{v}}^l, \underline{\hat{v}}^r, \hat{\overline{v}}^r, t) \\ + \hat{\boldsymbol{u}}_F\left(t\right) \end{aligned} \tag{15}$$

where $\hat{u}_F$ is the simple fuzzy compensator controller; $\underline{\hat{w}}, \hat{\overline{w}}, \underline{\hat{m}}^l, \hat{\overline{m}}^l, \underline{\hat{m}}^r, \hat{\overline{m}}^r, \underline{\hat{v}}^l, \hat{\overline{v}}^l, \underline{\hat{v}}^r, \hat{\overline{v}}^r$ are the estimation of $w^*, \overline{w}^*, \underline{m}^{l*}, \overline{m}^{l*}, \underline{m}^{r*}, \overline{m}^{r*}, \underline{v}^{l*}, \overline{v}^{l*}, \underline{v}^{r*}, \overline{v}^{r*}$.

Consider the Lyapunov function candidate

$$V_1\left(\boldsymbol{s}\left(t\right)\right) = \frac{1}{2}\boldsymbol{s}^2\left(t\right) \tag{16}$$

where $\boldsymbol{s}(t)$ is the high-order sliding surface, which is given as:

$$\boldsymbol{s}(t) = \boldsymbol{e}^{(n-1)} + g_1\boldsymbol{e}^{(n-2)}\ldots + g_n\int_0^t \boldsymbol{e}\left(\tau\right)d\tau \tag{17}$$

Then, take the derivative of (17), to obtain

$$\dot{\boldsymbol{s}}(t) = \boldsymbol{e}^{(n)} + \boldsymbol{G}^T\boldsymbol{e} \tag{18}$$

where $\boldsymbol{G} = \left[g_n, \ldots, g_2, g_1\right]^T$ is the positive gain.

Taking the derivative of (16) and using (9), (11), (18), obtains

$$\begin{aligned} \dot{V}_1(\boldsymbol{s}(t)) = \boldsymbol{s}(t)\dot{\boldsymbol{s}}(t) = \boldsymbol{s}(t)\left[\boldsymbol{e}^{(n)} + \boldsymbol{G}^T\boldsymbol{e}\right] \\ = s(t)\Big[x_d^{(n)}\left(t\right) - \left(A_0\left(\boldsymbol{x}\left(t\right)\right) + B_0\left(\boldsymbol{x}\left(t\right)\right)\right. \\ \times\left(\hat{u}_{MMT2AFC} + \hat{u}_F\left(t\right)\right) + \xi(\boldsymbol{x}(t))\Big) + \boldsymbol{G}^T\boldsymbol{e}\Big] \end{aligned} \tag{19}$$

Then, the adaptation laws for $\hat{w}, \hat{\overline{w}}, \underline{\hat{m}}^l, \hat{\overline{m}}^l, \underline{\hat{m}}^r, \hat{\overline{m}}^r, \underline{\hat{v}}^l, \hat{\overline{v}}^l, \underline{\hat{v}}^r, \hat{\overline{v}}^r$ are given as

$$\begin{aligned} &\hat{\overline{w}}_{klm}\left(t+1\right) \\ &= \hat{\overline{w}}_{klm}\left(t\right) - \hat{\eta}_w\frac{\partial\dot{V}(t)}{\partial\dot{\overline{w}}_{klm}} \\ &= \hat{\overline{w}}_{klm}\left(t\right) - \hat{\eta}_w\frac{\partial\dot{V}_1(t)}{\partial\hat{u}^m_{MMT\ 2\ AFC}}\frac{\partial\hat{u}^m_{MMT\ 2\ AFC}}{\partial\overline{o}_m}\frac{\partial\overline{o}_m}{\partial\hat{w}_{klm}} \\ &= \hat{\overline{w}}_{klm}\left(t\right) + \frac{1}{2}\hat{\eta}_w qB_0 s(t)\overline{f}_{kl} \end{aligned} \tag{20}$$

$$\begin{aligned} &\underline{\hat{w}}_{klm}(t+1) \\ &= \underline{\hat{w}}_{klm}(t) - \hat{\eta}_w\frac{\partial\dot{V}_1(t)}{\partial\underline{\hat{w}}_{klm}} \\ &= \underline{\hat{w}}_{klm}(t) - \hat{\eta}_w\frac{\partial\dot{V}_1(t)}{\partial\hat{u}^m_{MMT\ 2\ AFC}}\frac{\partial\hat{u}^m_{MMT\ 2\ AFC}}{\partial\underline{O}_m}\frac{\partial\underline{o}_m}{\underline{\hat{w}}_{klm}} \\ &= \underline{\hat{w}}_{klm}(t) + \frac{1}{2}(1-q)\hat{\eta}_w B_0 s(t)\underline{f}_{kl} \end{aligned} \tag{21}$$

$$\begin{aligned} &\underline{\hat{m}}^l_{ijk}\left(t+1\right) \\ &= \underline{\hat{m}}^l_{ijk}\left(t\right) - \hat{\eta}_m\frac{\partial\dot{V}_1(t)}{\partial\underline{\hat{m}}^l_{ijk}} \\ &= \underline{\hat{m}}^l_{ijk}(t) - \hat{\eta}_m\left(\frac{\partial\dot{V}_1(t)}{\partial\hat{u}^m_{MMT\ 2\ AFC}}\frac{\partial\hat{u}^m_{MMT\ 2\ AFC}}{\partial\underline{o}_m}\frac{\partial\underline{o}_m}{\partial\underline{f}_{kl}}\frac{\partial f_{kl}}{\partial\mu_{\underline{\mu}_{ijk}}}\frac{\partial\mu_{ljk}}{\partial\underline{\hat{m}}^l_{ljk}}\right) \\ &= \underline{\hat{m}}^l_{ijk}\left(t\right) + \frac{1}{2}\hat{\eta}_m B_0 s(t)(1-q)\underline{w}_{klm}\left(\frac{f_{kl}}{\underline{\mu}_{ijk}}\right)\frac{\partial\underline{\mu}_{ijk}}{\partial\underline{\hat{m}}^l_{ijk}} \end{aligned} \tag{22}$$

$$\begin{aligned} &\underline{\hat{m}}^r_{ijk}\left(t+1\right) \\ &= \underline{\hat{m}}^r_{ijk}\left(t\right) - \hat{\eta}_m\frac{\partial\dot{V}_1(t)}{\partial\underline{\hat{m}}^r_{ijk}} \\ &= \underline{\hat{m}}^r_{ijk}(t) - \hat{\eta}_m\left(\frac{\partial\dot{V}_1(t)}{\partial\hat{u}^m_{MMT\ 2\ AFC}}\frac{\partial\hat{u}^m_{MMT\ 2\ AFC}}{\partial\underline{o}_m}\frac{\partial\underline{o}_m}{\partial\underline{f}_{kl}}\frac{\partial\underline{f}_{kl}}{\partial\underline{\mu}_{\underline{y}k}}\frac{\partial\mu_{ijk}}{\hat{D}\underline{\hat{m}}^r_{ijk}}\right) \\ &= \underline{\hat{m}}^l_{ijk}\left(t\right) + \frac{1}{2}\hat{\eta}_m B_0 s(t)(1-q)\underline{w}_{k\,\ln}\left(\frac{\underline{f}_{kl}}{\underline{\mu}_{ijk}}\right)\frac{\partial\underline{\mu}_{ijk}}{\partial\underline{\hat{m}}^l_{ijk}} \end{aligned} \tag{23}$$

$$\begin{aligned}
&\hat{\underline{v}}^{l}_{ijk}\,(t+1)\\
&=\hat{\underline{v}}^{l}_{ijk}\,(t)-\hat{\eta}_v\frac{\partial\dot{V}_1(t)}{\partial\hat{\underline{v}}^{l}_{ijk}}\\
&=\hat{\underline{v}}^{l}_{ijk}(t)-\hat{\eta}_v\left(\frac{\partial\dot{V}_1(t)}{\partial\hat{u}^{m}_{MMT\ 2\ AFC}}\frac{\partial\hat{u}^{m}_{MMT\ 2\ AFC}}{\partial\underline{o}_m}\frac{\partial\underline{o}_m}{\partial\underline{f}_{kl}}\frac{\partial\underline{f}_{kl}}{\partial\underline{\mu}_{ijk}}\frac{\partial\underline{\mu}_{ijk}}{\partial\hat{\underline{v}}^{l}_{ijk}}\right)\\
&=\hat{\underline{v}}^{l}_{ijk}\,(t)+\frac{1}{2}\hat{\eta}_v B_0 s(t)(1-q)\underline{w}_{klm}\left(\frac{\underline{f}_{kl}}{\underline{\mu}_{ijk}}\right)\frac{\partial\underline{\mu}_{ijk}}{\partial\hat{\underline{v}}^{l}_{ijk}} \qquad (24)
\end{aligned}$$

$$\begin{aligned}
&\hat{\underline{v}}^{r}_{ijk}\,(t+1)\\
&=\hat{\underline{v}}^{r}_{ijk}\,(t)-\hat{\eta}_v\frac{\partial\dot{V}_1(t)}{\partial\hat{\underline{v}}^{r}_{ijk}}\\
&=\hat{\underline{v}}^{r}_{ijk}\,(t)-\hat{\eta}_v\left(\frac{\partial\dot{V}_1(t)}{\partial\hat{u}^{m}_{MMT\ 2\ AFC}}\frac{\partial\hat{u}^{m}_{MM\ 2\ AFC}}{\partial\underline{o}_m}\frac{\partial\underline{o}_m}{\partial\underline{f}_{kl}}\frac{\partial\underline{f}_{kl}}{\partial\underline{\mu}_{iik}}\frac{\partial\underline{\mu}_{ikk}}{\partial\hat{\underline{v}}^{r}_{ijk}}\right)\\
&=\hat{\underline{v}}^{r}_{ijk}\,(t)+\frac{1}{2}\eta\hat{\eta}_v B_0 s(t)(1-q)\underline{w}_{klm}\left(\frac{\underline{f}_{kl}}{\underline{\mu}_{ijk}}\right)\frac{\partial\underline{\mu}_{ijk}}{\partial\hat{\underline{v}}^{r}_{ijk}} \qquad (25)
\end{aligned}$$

$$\begin{aligned}
&\hat{\overline{m}}^{l}_{ijk}\,(t+1)\\
&=\hat{\overline{m}}^{l}_{ijk}\,(t)-\hat{\eta}_m\frac{\partial\dot{V}_1(t)}{\partial\hat{\overline{m}}^{l}_{ijk}}\\
&=\hat{\overline{m}}^{l}_{ijk}(t)-\hat{\eta}_m\left(\frac{\partial\dot{V}_1(t)}{\partial\hat{u}^{m}_{MMT\ 2\ AFC}}\frac{\partial\hat{u}^{m}_{MMT\ 2\ AFC}}{\partial\overline{o}_m}\frac{\partial\overline{o}_m}{\partial\overline{f}_{kl}}\frac{\partial\overline{f}_{kl}}{\partial\overline{\mu}_{ijk}}\frac{\partial\overline{\mu}_{ijk}}{\partial\hat{\overline{m}}^{l}_{ijk}}\right)\\
&=\hat{\overline{m}}^{l}_{ijk}\,(t)+\frac{1}{2}\hat{\eta}_m B_0 s(t)q\overline{w}_{klm}\left(\frac{\overline{f}_{kl}}{\overline{\mu}_{ijk}}\right)\frac{\partial\overline{\mu}_{ijk}}{\partial\hat{\overline{m}}^{l}_{ijk}} \qquad (26)
\end{aligned}$$

$$\begin{aligned}
&\hat{\overline{m}}^{r}_{ijk}\,(t+1)\\
&=\hat{\overline{m}}^{r}_{ijk}\,(t)-\hat{\eta}_m\frac{\partial\dot{V}_1(t)}{\partial\hat{\overline{m}}^{r}_{ijk}}\\
&=\hat{\overline{m}}^{r}_{ijk}\,(t)-\hat{\eta}_m\left(\frac{\partial\dot{V}_1(t)}{\partial\hat{u}^{m}_{MMTAFC}}\frac{\partial\hat{u}^{m}_{MMT\ 2\ AFC}}{\partial\overline{o}_m}\frac{\partial\overline{o}_m}{\partial\overline{f}_{kl}}\frac{\partial\overline{f}_{kl}}{\partial\overline{\mu}_{ijk}}\frac{\partial\overline{\mu}_{ijk}}{\partial\hat{\overline{m}}^{r}_{ijk}}\right)\\
&=\hat{\overline{m}}^{r}_{ijk}\,(t)+\frac{1}{2}\hat{\eta}_m B_0 s(t)q\overline{w}_{klm}\left(\frac{\overline{f}_{kl}}{\overline{\mu}_{ijk}}\right)\frac{\partial\overline{\mu}_{ijk}}{\partial\hat{\overline{m}}^{r}_{ijk}} \qquad (27)
\end{aligned}$$

$$\begin{aligned}
&\overline{v}^{l}_{ijk}\,(t+1)\\
&=\overline{v}^{l}_{ijk}\,(t)-\hat{\eta}_v\frac{\partial\dot{V}_1(t)}{\partial\overline{v}^{l}_{ijk}}\\
&=\overline{v}^{l}_{ijk}(t)-\hat{\eta}_v\left(\frac{\partial\dot{V}_1(t)}{\partial\hat{u}^{m}_{MMT\ 2\ AFC}}\frac{\partial\hat{u}^{m}_{MMT\ 2\ AFC}}{\partial\overline{o}_m}\frac{\partial\overline{o}_m}{\partial\overline{f}_{kl}}\frac{\partial\overline{f}_{kl}}{\partial\overline{\mu}_{ijk}}\frac{\partial\overline{\mu}_{ijk}}{\partial\overline{v}^{l}_{ijk}}\right)\\
&=\overline{v}^{l}_{ijk}\,(t)+\frac{1}{2}\hat{\eta}_v B_0 s(t)q\overline{w}_{klm}\left(\frac{\overline{f}_{kl}}{\overline{\mu}_{ijk}}\right)\frac{\partial\overline{\mu}_{ijk}}{\partial\overline{v}^{l}_{ijk}} \qquad (28)
\end{aligned}$$

$$\begin{aligned}
&\overline{v}^{r}_{ijk}\,(t+1)\\
&=\overline{v}^{r}_{ijk}\,(t)-\hat{\eta}_v\frac{\partial\dot{V}_1(t)}{\partial\overline{v}^{r}_{ijk}}\\
&=\overline{v}^{r}_{ijk}\,(t)-\hat{\eta}_v\left(\frac{\partial\dot{V}_1(t)}{\partial\hat{u}^{m}_{MM\ 2\ AFC}}\frac{\partial\hat{u}^{m}_{MMT\ 2\ AFC}}{\partial\overline{o}_m}\frac{\partial\overline{o}_m}{\partial\overline{f}_{kl}}\frac{\partial\overline{f}_{kl}}{\partial\overline{\mu}_{ijk}}\frac{\partial\overline{\mu}_{ijk}}{\partial\overline{v}^{r}_{ijk}}\right)\\
&=\overline{v}^{r}_{ijk}\,(t)+\frac{1}{2}\hat{\eta}_v B_0 s(t)q\overline{w}_{klm}\left(\frac{\overline{f}_{kl}}{\overline{\mu}_{ijk}}\right)\frac{\partial\overline{\mu}_{ijk}}{\partial\overline{v}^{r}_{ijk}} \qquad (29)
\end{aligned}$$

where $\hat{\eta}_w$, $\hat{\eta}_m$, $\hat{\eta}_v$ are the positive learning-rates.

The derivative terms in (22) – (29) can be obtained by considering the region of input as shown in Table 1 and Table 2.

By using the adaptation laws given in (20)-(29), the estimation parameters for $\hat{\boldsymbol{u}}_{MMT2AFC}$ can be obtained.

C. FUZZY COMPENSATOR CONTROLLER

The fuzzy compensator controller in (15) is designed to compensate for the discrepancy between the proposed MMT2AFC and the ideal controller. For rapid response to the tracking errors, these three simple rules are considered

$$\begin{aligned}
&R^1:\ \textit{If}\ s_i\ \textit{is}\ P,\ \textit{then}\ u_F^i\ \textit{is}\ UP\\
&R^2:\ \textit{If}\ s_i\ \textit{is}\ Z,\ \textit{then}\ u_F^i\ \textit{is}\ UZ\\
&R^3:\ \textit{If}\ s_i\ \textit{is}\ N,\ \textit{then}\ u_F^i\ \textit{is}\ UN \qquad (30)
\end{aligned}$$

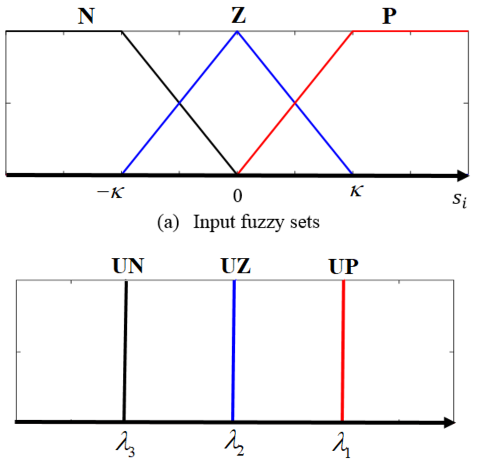
where N, P, Z, respectively, indicate the negative, positive, and zero triangular-typed input MFs for the fuzzy antecedent part; UN, UP, UZ, respectively, indicate the negative, positive, and zero output MFs for the consequent fuzzy part.

TABLE 1. The derivative terms for updating the lower parameters.

Region 1 $x_i \le \underline{m}^{l}_{ijk}$	**Region 2** $\underline{m}^{l}_{ijk} \le x_i \le \underline{m}^{r}_{ijk}$	**Region 3** $\underline{m}^{r}_{ijk} \le x_i$
$\frac{\partial\underline{\mu}_{ijk}}{\partial\hat{\underline{m}}^{l}_{ijk}}=\underline{\mu}_{ijk}\frac{\left(x_i-\underline{m}^{l}_{ijk}\right)}{\left(\underline{v}^{l}_{ijk}\right)^2};\ \frac{\partial\underline{\mu}_{ijk}}{\partial\hat{\underline{v}}^{r}_{ijk}}=0;$ $\frac{\partial\underline{\mu}_{ijk}}{\partial\hat{\underline{m}}^{r}_{ijk}}=0;\ \frac{\partial\underline{\mu}_{ijk}}{\partial\hat{\underline{v}}^{l}_{ijk}}=\underline{\mu}_{ijk}\frac{\left(x_i-\underline{m}^{l}_{ijk}\right)^2}{\left(\underline{v}^{l}_{ijk}\right)^3};$	$\frac{\partial\underline{\mu}_{ijk}}{\partial\hat{\underline{m}}^{l}_{ijk}}=0;\ \frac{\partial\underline{\mu}_{ijk}}{\partial\hat{\underline{v}}^{r}_{ijk}}=0;$ $\frac{\partial\underline{\mu}_{ijk}}{\partial\hat{\underline{m}}^{r}_{ijk}}=0;\ \frac{\partial\underline{\mu}_{ijk}}{\partial\hat{\underline{v}}^{l}_{ijk}}=0;$	$\frac{\partial\underline{\mu}_{ijk}}{\partial\hat{\underline{m}}^{l}_{ijk}}=0;\ \frac{\partial\underline{\mu}_{ijk}}{\partial\hat{\underline{v}}^{r}_{ijk}}=\underline{\mu}_{ijk}\frac{\left(I_i-\underline{m}^{r}_{ijk}\right)^2}{\left(\underline{v}^{r}_{ijk}\right)^3};$ $\frac{\partial\underline{\mu}_{ijk}}{\partial\hat{\underline{m}}^{r}_{ijk}}=\underline{\mu}_{ijk}\frac{\left(I_i-\underline{m}^{r}_{ijk}\right)}{\left(\underline{v}^{r}_{ijk}\right)^2};\ \frac{\partial\underline{\mu}_{ijk}}{\partial\hat{\underline{v}}^{l}_{ijk}}=0$

TABLE 2. The derivative terms for updating the upper parameters.

Region 1 $x_i \le \bar{m}^l_{ijk}$	Region 2 $\bar{m}^l_{ijk} \le x_i \le \bar{m}^r_{ijk}$	Region 3 $\bar{m}^r_{ijk} \le x_i$
$\frac{\partial \bar{\mu}_{ijk}}{\partial \bar{m}^l_{ijk}} = \bar{\mu}_{ijk} \frac{\left(x_i - \bar{m}^l_{ijk}\right)}{\left(\bar{v}^l_{ijk}\right)^2}; \frac{\partial \bar{\mu}_{ijk}}{\partial \bar{v}^r_{ijk}} = 0;$ $\frac{\bar{\mu}_{ijk}}{\partial \bar{m}^r_{ijk}} = 0; \frac{\partial \bar{\mu}_{ijk}}{\partial \bar{v}^l_{ijk}} = \bar{\mu}_{ijk} \frac{\left(x_i - \bar{m}^l_{ijk}\right)^2}{\left(\bar{v}^l_{ijk}\right)^3};$	$\frac{\partial \bar{\mu}_{ijk}}{\partial \bar{m}^l_{ijk}} = 0; \frac{\partial \bar{\mu}_{ijk}}{\partial \bar{v}^r_{ijk}} = 0;$ $\frac{\bar{\mu}_{ijk}}{\partial \bar{m}^r_{ijk}} = 0; \frac{\partial \bar{\mu}_{ijk}}{\partial \bar{v}^l_{ijk}} = 0;$	$\frac{\partial \bar{\mu}_{ijk}}{\partial \bar{m}^l_{ijk}} = 0; \frac{\partial \bar{\mu}_{ijk}}{\partial \bar{v}^r_{ijk}} = \bar{\mu}_{ijk} \frac{\left(x_i - \bar{m}^r_{ijk}\right)^2}{\left(\bar{v}^r_{ijk}\right)^3};$ $\frac{\bar{\mu}_{ijk}}{\partial \bar{m}^r_{ijk}} = \bar{\mu}_{ijk} \frac{\left(x_i - \bar{m}^r_{ijk}\right)}{\left(\bar{v}^r_{ijk}\right)^2}; \frac{\partial \bar{\mu}_{ijk}}{\partial \bar{v}^l_{ijk}} = 0$



(a) Input fuzzy sets

(b) Output fuzzy sets

FIGURE 4. The input and output membership functions.

The membership function for the fuzzy input sets and fuzzy output sets are given in Fig. 4. Using the center of gravity defuzzification method, the output of the fuzzy compensator controller is achieved as:

$$u_F^i = \frac{\sum_{a=1}^{3} \lambda_a^i \theta_a^i}{\sum_{a=1}^{3} \theta_a^i} = \lambda_1^i \theta_1^i + \lambda_2^i \theta_2^i + \lambda_3^i \theta_3^i \tag{31}$$

where $\boldsymbol{\lambda}^i = [\lambda_1^i, \lambda_2^i, \lambda_3^i]$ are the fuzzy weights; $\boldsymbol{\theta}^i = [\theta_1^i, \theta_2^i, \theta_3^i]$ are firing strengths. By choosing the MFs for the fuzzy antecedent part as Fig. 4a, the firing strengths, and fuzzy outputs can be given as

$$\begin{aligned}
&Case\,1 : (s_i \le -\kappa) \\
&\qquad \theta_1^i = 0; \theta_2^i = 0; \theta_3^i = 1; \\
&\qquad u_F^i = \lambda_3^i \theta_3^i \\
&Case\,2 : (-\kappa \le s_i \le 0) \\
&\qquad \theta_1^i = 0; \theta_2^i = (s_i + \kappa)/\kappa; \theta_3^i = 1 - \beta_2^i; \\
&\qquad u_F^i = \lambda_2^i \theta_2^i + \lambda_3^i \theta_3^i \\
&Case\,3 : (0 \le s_i \le \kappa) \\
&\qquad \theta_1^i = 1 - \theta_2^i; \theta_2^i = (\kappa - s_i)/\kappa; \theta_3^i = 0; \\
&\qquad u_F^i = \lambda_1^i \theta_1^i + \lambda_2^i \theta_2^i \\
&Case\,4 : (s_i > \kappa) \\
&\qquad \theta_1^i = 1; \theta_2^i = 0; \theta_3^i = 0; \\
&\qquad u_F^i = \lambda_1^i \theta_1^i
\end{aligned} \tag{32}$$

To minimize the computational budget, choosing λ_1^i, λ_2^i, λ_3^i as the singletons-typed MFs as Fig. 4b and let $\lambda_1^i = \hat{\lambda}_i$, $\lambda_2^i = 0$, $\lambda_3^i = -\hat{\lambda}_i$. Then, (31) can be rewritten as

$$u_F^i = \hat{\lambda}_i \left(\theta_1^i - \theta_3^i\right) \tag{33}$$

Rewriting (19) using (14), (15) and (33), obtains

$$\begin{aligned}
\dot{V}_1 &= \sum_{i=1}^{m} B_0^i \left[s_i(t)\varsigma_i(t) - s_i(t)\hat{\lambda}_i \left(\theta_1^i - \theta_3^i\right) \right] \\
&\le \sum_{i=1}^{m} B_0^i \left[|s_i(t)| \,.\, |\varsigma_i(t)| - s_i(t)\hat{\lambda}_i \left(\theta_1^i - \theta_3^i\right) \right] \\
&= \sum_{i=1}^{m} B_0^i \left[|s_i(t)| \,.\, |\varsigma_i(t)| - \hat{\lambda}_i \,|s_i(t)| \,.\, \left|\theta_1^i - \theta_3^i\right| \right] \\
&= -\sum_{i=1}^{m} B_0^i \left[|s_i(t)| \,.\, \left|\theta_1^i - \theta_3^i\right| \right] \left(\hat{\lambda}_i - \frac{|\varsigma_i(t)|}{\left|\theta_1^i - \theta_3^i\right|} \right)
\end{aligned} \tag{34}$$

where m is the dimension of vector $\boldsymbol{s}_i$. In (34), if there is an estimated value $\hat{\lambda}_i > \frac{|\varsigma_i(t)|}{|\theta_1^i - \theta_3^i|}$, then $\dot{V} \le 0$ can be satisfied. Define an optimal value λ_i^* to achieve minimum value of $\hat{\lambda}_i$ as:

$$\lambda_i^* = \frac{|\varsigma_i(t)|}{\left|\theta_1^i - \theta_3^i\right|} + \Xi_i \tag{35}$$

where Ξ_i is a positive constant.

Defined an estimation error vector $\tilde{\boldsymbol{\lambda}}_i = \left[\tilde{\lambda}_1, \ldots, \tilde{\lambda}_i, \ldots, \tilde{\lambda}_m\right]^T$, where $\tilde{\lambda}_i$ is given as:

$$\tilde{\lambda}_i = \lambda_i^* - \hat{\lambda}_i \tag{36}$$

Define the new Lyapunov function as:

$$V_2\left(\boldsymbol{s}\left(t\right)\right)=\frac{1}{2}\boldsymbol{s}^T\left(t\right)\boldsymbol{s}\left(t\right)+\frac{1}{2}\tilde{\boldsymbol{\lambda}}^T\tilde{\boldsymbol{\lambda}} \tag{37}$$

Taking the derivative of (37), and using (12), (14), (19) and (33), one obtains

$$\begin{aligned}
&\dot{V}_2\left(\boldsymbol{s}\left(t\right)\right)\\
&\quad=\boldsymbol{s}^T\left(t\right)\dot{\boldsymbol{s}}\left(t\right)+\tilde{\boldsymbol{\lambda}}^T\dot{\tilde{\boldsymbol{\lambda}}}\\
&\quad=\sum_{i=1}^{m}B_0^i\left[s_i(t)\varsigma_i(t)-\hat{\lambda}_i s_i(t)\left(\theta_1^i-\theta_3^i\right)+\tilde{\lambda}_i\dot{\tilde{\lambda}}_i\right]\\
&\quad\le\sum_{i=1}^{m}B_0^i\left[|s_i(t)|\cdot|\varsigma_i(t)|+\tilde{\lambda}_i s_i(t)\left(\theta_1^i-\theta_3^i\right)\right.\\
&\qquad\qquad\left.-\lambda_i^*\left|s_i(t)\right|\cdot\left|\theta_1^i-\theta_3^i\right|+\tilde{\lambda}_i\dot{\tilde{\lambda}}_i\right]\\
&\quad=\sum_{i=1}^{m}B_0^i\left[|s_i(t)|\cdot|\varsigma_i(t)|+\tilde{\lambda}_i\left[s_i(t)\left(\theta_1^i-\theta_3^i\right)+\dot{\tilde{\lambda}}_i\right]\right.\\
&\qquad\qquad\left.-\lambda_i^*\left|s_i(t)\right|\cdot\left|\theta_1^i-\theta_3^i\right|\right]
\end{aligned} \tag{38}$$

Choose the estimation laws as

$$\dot{\hat{\lambda}}_i=-\dot{\tilde{\lambda}}_i=s_i(t)\left(\theta_1^i-\theta_3^i\right) \tag{39}$$

Rewrite (38), using (35), one obtains

$$\begin{aligned}
&\dot{V}_2\left(\boldsymbol{s}\left(t\right)\right)\\
&\le\sum_{i=1}^{m}B_0^i\left[|s_i(t)|\cdot|\varsigma_i(t)|-|s_i(t)|\left|\theta_1^i-\theta_3^i\right|\left(\frac{|\varsigma_i(t)|}{\left|\theta_1^i-\theta_3^i\right|}+\Xi_i\right)\right]\\
&=\sum_{i=1}^{m}B_0^i\left[|s_i(t)|\cdot|\varsigma_i(t)|-|s_i(t)|\left(|\varsigma_i(t)|+\Xi_i\left|\theta_1^i-\theta_3^i\right|\right)\right]\\
&=-\sum_{i=1}^{m}B_0^i\Xi_i\left|s_i(t)\right|\left|\theta_1^i-\theta_3^i\right|\le 0
\end{aligned} \tag{40}$$

According to (40), $\dot{V}_2\left(\boldsymbol{s}\left(t\right)\right)$ is negative semidefinite, therefore the stability of the MMT2AFC system can be guaranteed.

III. STRUCTURE LEARNING ALGORITHM AND LEARNING RATE OPTIMIZER

A. STRUCTURE LEARNING FOR MMT2AFC

The design structure of the MMT2AFC significantly affects the system performances. Therefore, a self-evolving algorithm is proposed in this study to construct the network structure autonomously. The objective of the structure learning is to obtain a suitable number of layers of the proposed network online. The self-evolving method consists of the growing and pruning algorithm, which operate based on the contribution of each layer as follows:

1) GROWING ALGORITHM

First, determine the maximum membership grade in all layers for the i^{th} input as

$$\rho_{\max}^i=max\left[\mu_{i11},\ldots,\mu_{i1n_k},\ \mu_{i21},\ldots,\right.\\
\left.\mu_{i2n_k},\ldots,\mu_{in_j1},\ldots,\mu_{in_jn_k}\right] \tag{41}$$

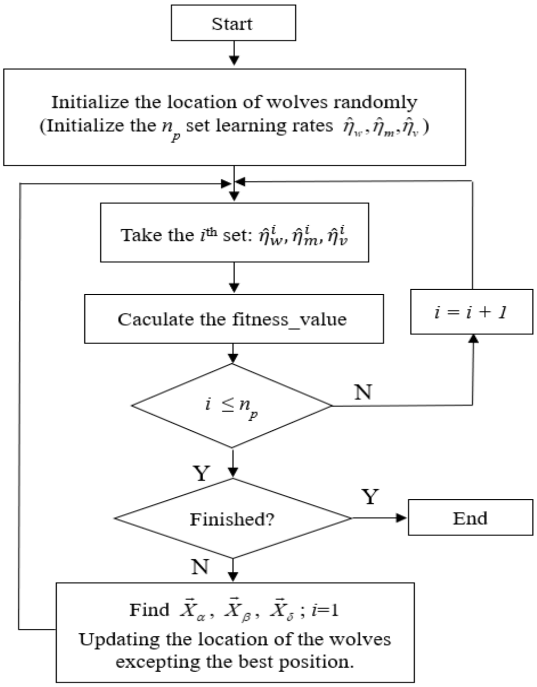

Start
Initialize the location of wolves randomly
(Initialize the n_p set learning rates $\hat{\eta}_w,\hat{\eta}_m,\hat{\eta}_v$)
Take the i^{th} set: $\hat{\eta}_w^i,\hat{\eta}_m^i,\hat{\eta}_v^i$
Caculate the fitness_value
$i=i+1$
$i\le n_p$
N
Y
Finished?
Y
End
N
Find $\vec{X}_\alpha$, $\vec{X}_\beta$, $\vec{X}_\delta$; i=1
Updating the location of the wolves
excepting the best position.

FIGURE 5. Flowchart of MGWO.

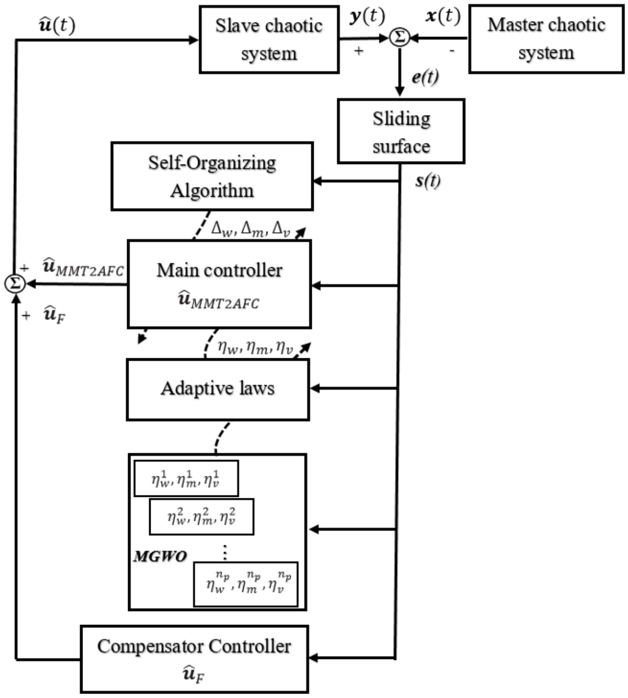


FIGURE 6. Block diagram of the MMT2AFC synchronizer chaotic systems.

where

$$\mu_{ijk}=\frac{\underline{\mu}_{ijk}+\overline{\mu}_{ijk}}{2} \tag{42}$$

Then, consider its contribution by comparing with a prescribed growing threshold, P_g, as

$$\rho_{\max}^i<P_g \tag{43}$$

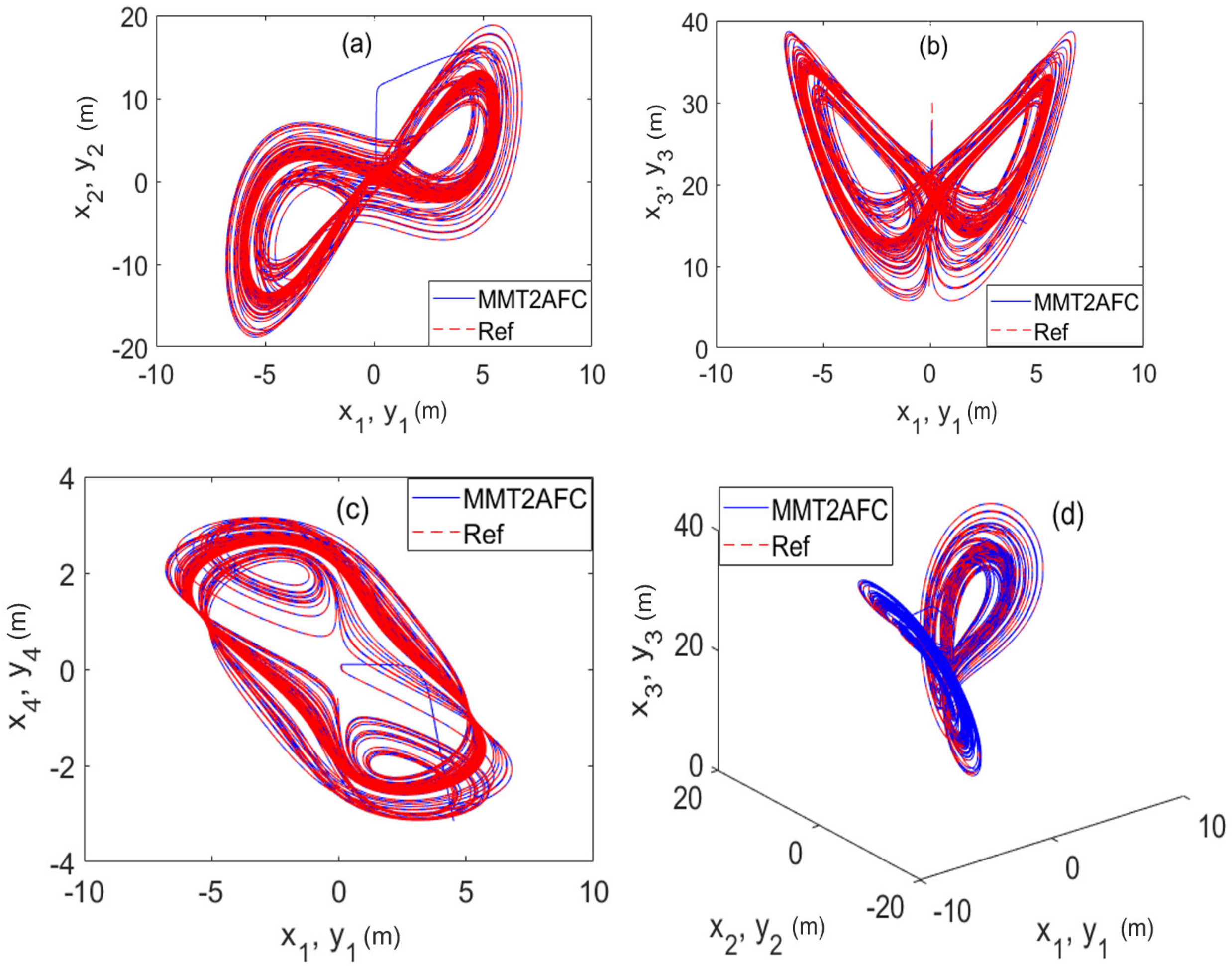


FIGURE 7. The dynamics of the 4-D Lorenz-Stenflo chaotic system in controlled state by using MMT2AFC controller (a) $x_1 - x_2$ orbits, (b) $x_1 - x_3$ orbits, (c) $x_1 - x_4$ orbits, (d) $x_1 - x_2 - x_3$ orbits.

If the condition in formula (43) is satisfied, it means none of the current layers contribute much to the fuzzy completeness norm, and a new layer should be added. The initial parameters for the upper and lower type-2 asymmetric membership function are set as follows.

$$\left[\overline{m}^l_{ijk}, \underline{m}^l_{ijk}, m^r_{ijk}, \overline{m}^r_{ijk}\right] = \left[(x_i - 2\kappa), (x_i - \kappa), (x_i + \kappa), x(I_i + 2\kappa)\right] \tag{44}$$

$$\left[v^l_{ijk}, \overline{v}^l_{ijk}, \underline{v}^r_{ijk}, \overline{v}^r_{ijk}\right] = \left[(v_{init} - 2\vartheta), (v_{init} - \vartheta), (v_{init} + \vartheta), (v_{init} + 2\vartheta)\right] \tag{45}$$

where v_{init}, κ and ϑ are the initial value of the variance, the mean uncertainty, and the variance uncertainty, respectively.

2) PRUNING ALGORITHM

First, determine the minimum membership grade in all layers for the i^{th} input as

$$\rho^i_{\min} = min\left[\mu_{i11}, \ldots, \mu_{i1n_k}, \mu_{i21}, \ldots, \mu_{i2n_k}, \ldots, \mu_{in_j1}, \ldots, \mu_{in_jn_k}\right] \tag{46}$$

Then, consider its contribution by comparing with a prescribed pruning threshold, P_d, as

$$\rho^i_{\min} < P_d \tag{47}$$

If the condition in formula (47) is satisfied, it means the minimum membership grade does not contribute much to the fuzzy completeness norm and it should be deleted.

Based on the proposed structure learning algorithm, the structure of the MMT2AFC network can adjust online to achieve a suitable number of layers.

B. THE MODIFIED GREY WOLF OPTIMIZER

This section applies the MGWO to optimize the suitable learning rates for the adaptive laws of the MT2AFC proposed controller. The proposed optimal algorithm in this work is a modification of the original GWO in [32]. The advantages of our proposed MGWO are having the best global position memorization in the swam; and increasing the movement freedom of the grey wolf members by adding the random terms of multidimensional motion, M and ψ_{n_d}, in the equation for updating the position of the swarm. Figure 5 shows

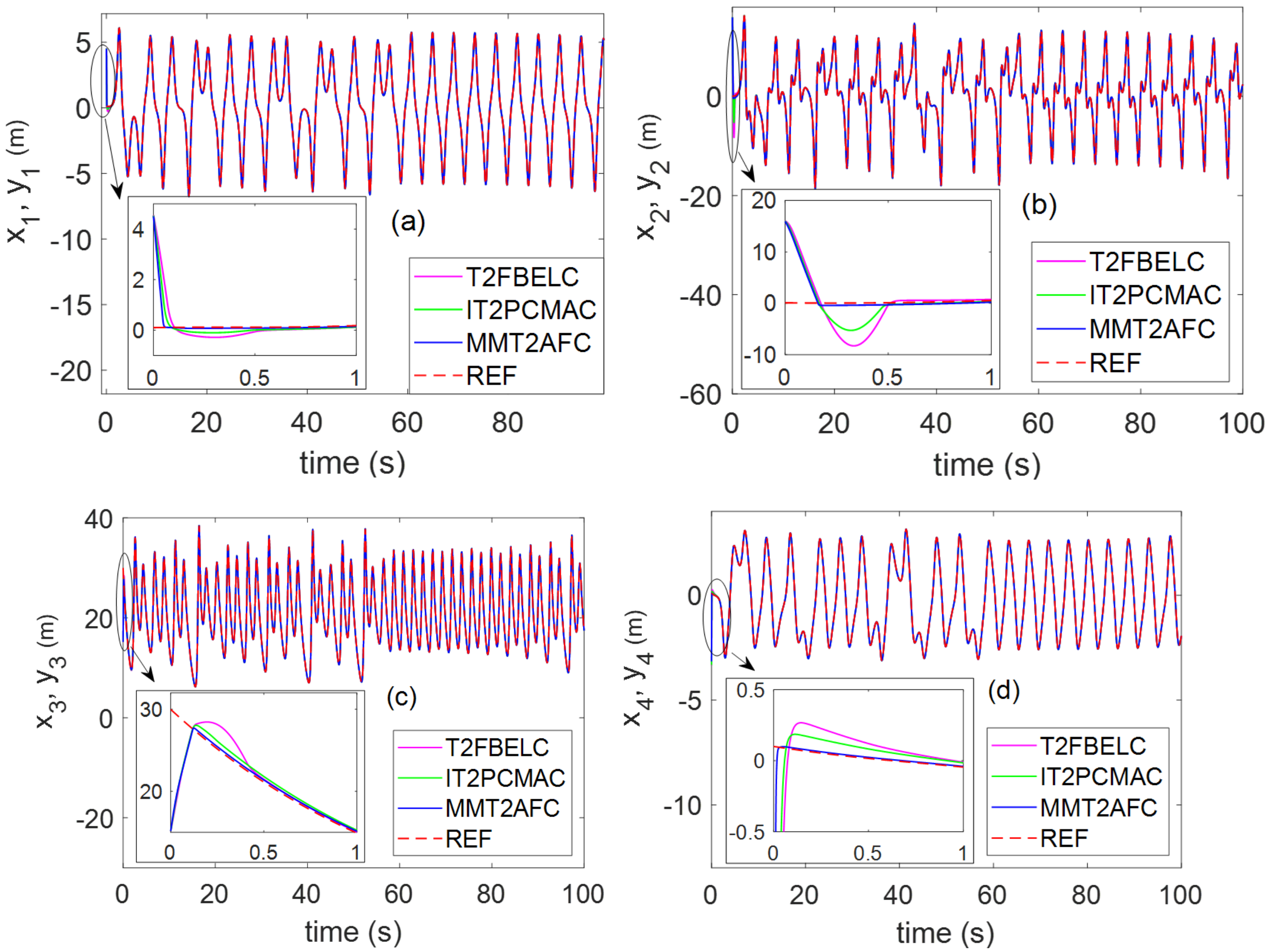


FIGURE 8. The time domain orbits of state variables. (a) x_1, y_1, (b) x_2, y_2, (c) x_3, y_3, and (d) x_4, y_4.

the flowchart of the MGWO algorithm. The equations for updating the grey wolf position, $\vec{X}$, are based on the best first three positions, $\vec{X}_\alpha, \vec{X}_\beta, \vec{X}_\delta$, in the swarm as follows

$$\vec{D}_\alpha = \left|\vec{C}_1 \cdot \vec{X}_\alpha - \vec{X}\right|; \quad \vec{D}_\beta = \left|\vec{C}_2 \cdot \vec{X}_\beta - \vec{X}\right|;$$
$$\vec{D}_\delta = \left|\vec{C}_3 \cdot \vec{X}_\delta - \vec{X}\right| \tag{48}$$

$$\vec{X}_1 = \left|\vec{X}_\alpha - \vec{A}_1 \cdot \vec{D}_\alpha\right|; \quad \vec{X}_2 = \left|\vec{X}_\beta - \vec{A}_2 \cdot \vec{D}_\beta\right|;$$
$$\vec{X}_3 = \left|\vec{X}_\delta - \vec{A}_3 \cdot \vec{D}_\delta\right| \tag{49}$$

$$\vec{X}(t+1) = \frac{\vec{X}_1 + \vec{X}_2 + \vec{X}_3}{3} + M\left[\psi_1; \quad \psi_2; \ldots; \psi_{n_d}\right] \tag{50}$$

where $\vec{D}_\alpha, \vec{D}_\beta, \vec{D}_\delta$, respectively, are the distance vectors between the $\vec{X}_\alpha, \vec{X}_\beta, \vec{X}_\delta$ and the $\vec{X}$; $\vec{A}_1, \vec{A}_2, \vec{A}_3$ and $\vec{C}_1, \vec{C}_2, \vec{C}_3$ are the coefficient vectors; $\vec{X}_1, \vec{X}_2, \vec{X}_3$ are the direction vectors for calculating the next position of a grey wolf; ψ_{n_d} is the random coefficient factor; n_d is the dimension of the search space; the variable for adjusting the movement freedom is denoted by M, which is given by

$$M = \left(satlin\left(\|e(t)\|^2\right)\right) \tag{51}$$

The coefficient vectors are expressed as:

$$\vec{A} = 2ar_1 - a; \quad \vec{C} = 2r_2 \tag{52}$$

where the value of a is decreased from 2 to 0 linearly with iterations; r_1, r_2 are the random numbers in [0, 1].

IV. ILLUSTRATIVE EXAMPLES

This section gives two examples to illustrate the effectiveness of our proposed method. The block diagram of the chaotic synchronization using the MMT2AFC controller is shown in Fig. 6. In which, based on the feedback tracking errors $\boldsymbol{e}(t)$, the proposed controller will generate the control signals $\boldsymbol{u}(t)$ to synchronize the slave system $\boldsymbol{y}(t)$ and the master system $\boldsymbol{x}(t)$. As shown in Eq. (15), the control signals $\boldsymbol{u}(t)$ combine the main controller, $u^m_{MMT2AFC}$, in Eq. (2) and the fuzzy compensator controller, u^i_F, in Eq. (31). The structure learning algorithm is given in section III (A) to obtain a suitable structure of the proposed network online, and the MGWO is given in section III (B) to optimal the learning rates of the parameter adaptive laws online. The initial parameters for the proposed controller are chosen as $v_{init} = 0.2$, $\kappa = 0.01$, $\vartheta = 0.01$, $P_g = 0.1$, $P_d = 0.01, n = 2$;

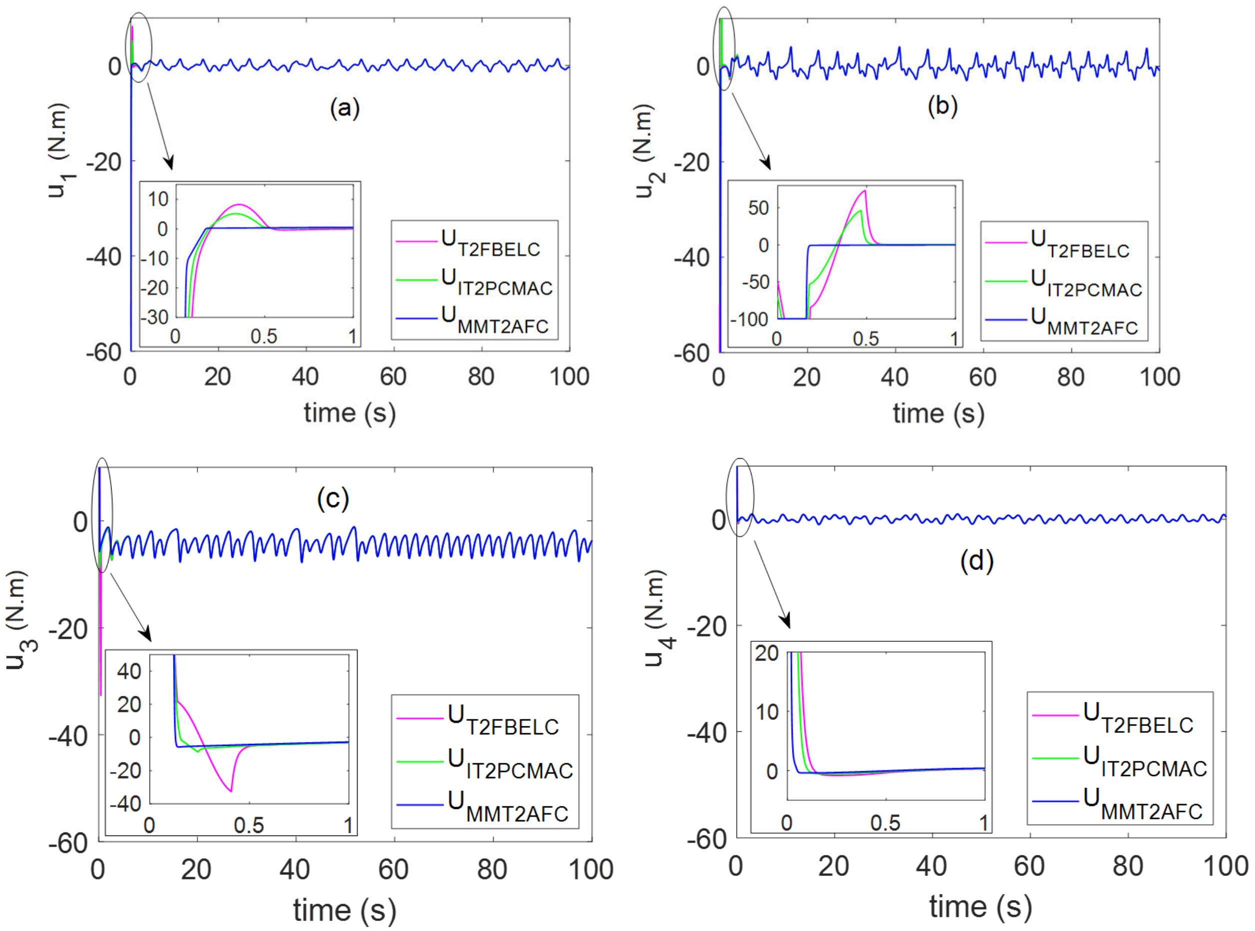


FIGURE 9. **The time domain of control signals. (a) u_1, (b) u_2, (c) u_3, and (d) u_4.**

the sampling time is 0.001 seconds; the minimum and maximum number of layers are respectively limited to one layer and seven layers.

Example 1: The synchronization of the four dimensions (4-D) chaotic system.

Consider the following 4D Lorenz–Stenflo chaotic system in [85].

The chaotic positions of the master system, $\boldsymbol{x}(t) = [x_1(t), x_2(t), x_3(t), x_4(t)]$, are given by:

$$\begin{aligned}\dot{x}_1(t) &= \alpha\,(x_2(t) - x_1(t)) + \gamma x_4\\ \dot{x}_2(t) &= \tau x_1(t) - x_1(t)x_3(t) - x_2(t)\\ \dot{x}_3(t) &= x_1(t)x_2(t) - \beta x_3(t)\\ \dot{x}_4(t) &= -x_1(t) - \alpha x_4 \end{aligned}\tag{53}$$

The chaotic positions of the slave system, $\boldsymbol{y}(t) = [y_1(t), y_2(t), y_3(t), y_4(t)]$, are given by:

$$\begin{aligned}\dot{y}_1(t) &= \alpha\,(y_2(t) - y_1(t)) + \gamma y_4 + d_1(t) + \Delta f(y_1) + u_1(t)\\ \dot{y}_2(t) &= \tau y_1(t) - y_1(t)y_3(t) - \lambda y_2(t) + d_2(t) + \Delta f(y_2) + u_2(t)\\ \dot{y}_3(t) &= y_1(t)y_2(t) - \beta y_3(t) + d_3(t) + \Delta f(y_3) + u_3(t)\\ \dot{y}_4(t) &= -y_1(t) - \alpha y_4 + d_4(t) + \Delta f(y_4) + u_4(t)\end{aligned}\tag{54}$$

where $\alpha,\ \tau,\ \beta,\ \gamma$ are the parameters for defining the chaotic attractor; $\boldsymbol{u}(t) = [u_1(t), u_2(t), u_3(t), u_4(t)]$ is the active control vector; $\boldsymbol{d}(t) = [d_1(t), d_2(t), d_3(t), d_4(t)]$ is the external disturbances vector; $\Delta \boldsymbol{f}(t) = [\Delta f(y_1), \Delta f(y_2), \Delta f(y_3), \Delta f(y_4)]$ is the vector uncertainties of the chaotic system.

The synchronization error vector, $\boldsymbol{e}(t) = [e_1(t), e_2(t), e_3(t), e_4(t)\ ]$, between the master and slave systems is defined as

$$\begin{aligned}e_1(t) &= y_1(t) - x_1(t)\\ e_2(t) &= y_2(t) - x_2(t)\\ e_3(t) &= y_3(t) - x_3(t)\\ e_4(t) &= y_4(t) - x_4(t)\end{aligned}\tag{55}$$

As in [74], we set $\alpha = 1$, $\beta = 0.7$, $\gamma = 1.5$ and $\tau = 26$. The initial conditions for the system are chosen as $[x_1, x_2, x_3, x_4] = [0.028, 0.02, 0.03, 0.048]^T$ and $[y_1, y_2, y_3, y_4] = [0.01, 0.037, 0.029, 0.008]^T$. The system uncertainties and the external disturbances are chosen as $[\Delta f(y_1), \Delta f(y_2), \Delta f(y_3), \Delta f(y_4)] = rd[0.2y_1, 0.2y_2, 0.2y_3,\ 0.2y_4]^T$ and $[d_1(t), d_2(t), d_3(t), d_4(t)] = [0.2\cos \pi t,\ 0.5\cos \pi t, 0.3\cos \pi t, 0.4\cos \pi t]^T$, respectively.

Figure 7 shows the dynamics state of the 4-D Lorenz-Stenflo chaotic system when the MMT2AFC controller is activated. In which, the dashed and dotted lines, respectively, correspond to the master system and slave system.

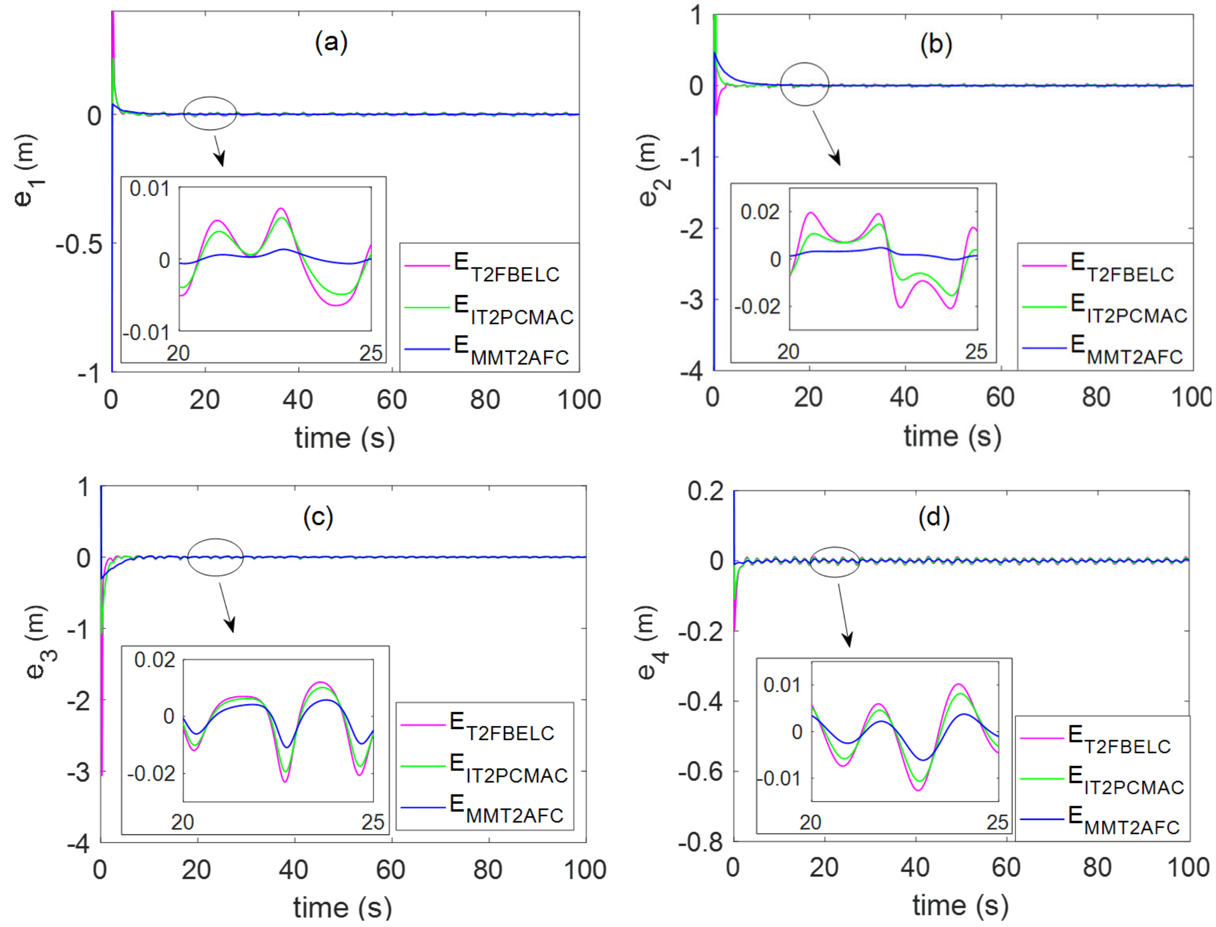


FIGURE 10. The time-domain of tracking errors. (a) e_1, (b) e_2, (c) e_3, and (d) e_4.

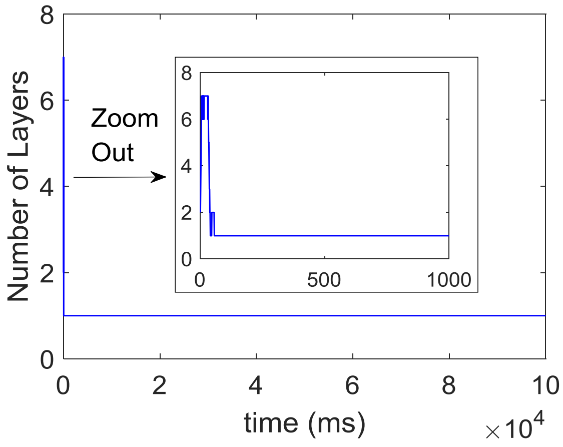


FIGURE 11. The number of layers using the self-evolving algorithm.

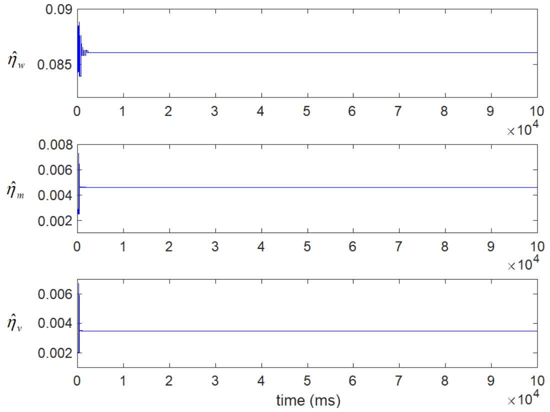


FIGURE 12. The alteration of the learning rates using the MGWO algorithm.

Figure 8 displays the time domain orbits of the state variables. The control signals and tracking errors are given in Fig. 9 and Fig. 10, respectively. The online adjustment in the number of layers using the self-evolving algorithm is shown in Fig. 11. The alteration in the learning rates using MGWO is shown in Fig. 12. These simulation results show the proposed MMT2AFC controller can control the slave system effectively to synchronize with the master system. The comparison results in the root mean square error (RMSE) among our proposed controller and other recent controllers are tabulated in Table 3.

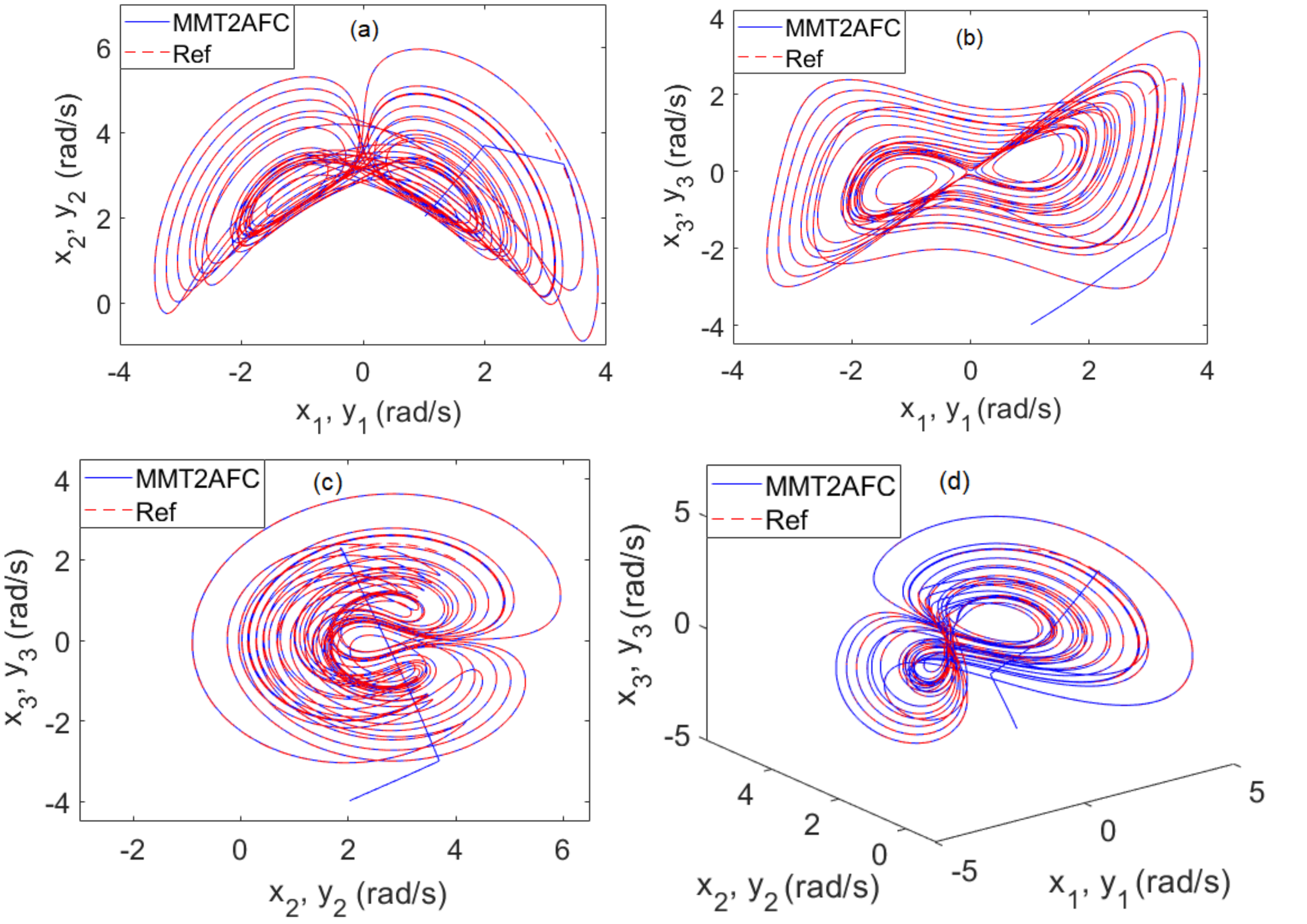


FIGURE 13. The dynamics of the chaotic satellite system in a controlled state by using MMT2AFC controller (a) $x_1 - x_2$ orbits, (b) $x_1 - x_3$ orbits, (c) $x_2 - x_3$ orbits, (d) $x_1 - x_2 - x_3$ orbits.

TABLE 3. Comparison results in RMSE synchronization of the chaotic system.

	Computation time (s)	Example 1	Example 2
T2FBELC	0.0192	0.9928	0.3619
IT2PCMAC	0.0168	0.8569	0.3045
MMT2AFC	0.0141	0.7493	0.2753

Example 2: Synchronization of the chaotic satellite system.

Consider the following chaotic satellite systems in [74].

The chaotic positions of the master system, $\boldsymbol{x}(t) = [x_1(t), x_2(t), x_3(t)]$, are given by:

$$\dot{x}_1(t) = \sigma_x x_2(t)x_3(t) - \frac{1.2}{I_x}x_1(t) + \frac{\sqrt{6}}{2I_x}x_3(t)$$
$$\dot{x}_2(t) == \sigma_y x_1(t)x_3(t) + \frac{0.35}{I_y}x_2(t)$$
$$\dot{x}_3(t) = \sigma_z x_1(t)x_2(t) - \frac{\sqrt{6}}{I_z}x_1(t) - \frac{0.4}{I_z}x_3(t) \quad (56)$$

The chaotic positions of the slave system, $\boldsymbol{y}(t) = [y_1(t), y_2(t), y_3(t)]$, are given by:

$$\begin{aligned}\dot{y}_1(t) &= \sigma_x y_2(t)y_3(t) - \frac{1.2}{I_x}y_1(t) + \frac{\sqrt{6}}{2I_x}y_3(t) \\ &\quad + d_1(t) + \Delta f(y_1) + u_1(t) \\ \dot{y}_2(t) &= \sigma_y y_1(t)y_3(t) + \frac{0.35}{I_y}y_2(t) + d_2(t) + \Delta f(y_2) + u_2(t) \\ \dot{y}_3(t) &= \sigma_z y_1(t)y_2(t) - \frac{\sqrt{6}}{I_z}y_1(t) - \frac{0.4}{I_z}y_3(t) \\ &\quad + d_3(t) + \Delta f(y_3) + u_3(t) \end{aligned} \quad (57)$$

where I_x, I_y, I_z are the principal moments of inertia; $\sigma_x = \frac{I_y - I_z}{I_x}$, $\sigma_y = \frac{I_z - I_x}{I_y}$, $\sigma_z = \frac{I_x - I_y}{I_z}$ are the chaotic coefficients; $\boldsymbol{u}(t) = [u_1(t), u_2(t), u_3(t)]$ is the active control vector; $\boldsymbol{d}(t) = [d_1(t), d_2(t), d_3(t)]$ is the external disturbances vector; $\Delta\boldsymbol{f}(t) = [\Delta f(y_1), \Delta f(y_2), \Delta f(y_3)]$ is the vector uncertainties of the chaotic system. The synchronization error vector, $\boldsymbol{e}(t) = [e_1(t), e_2(t), e_3(t)]$, between the master and slave

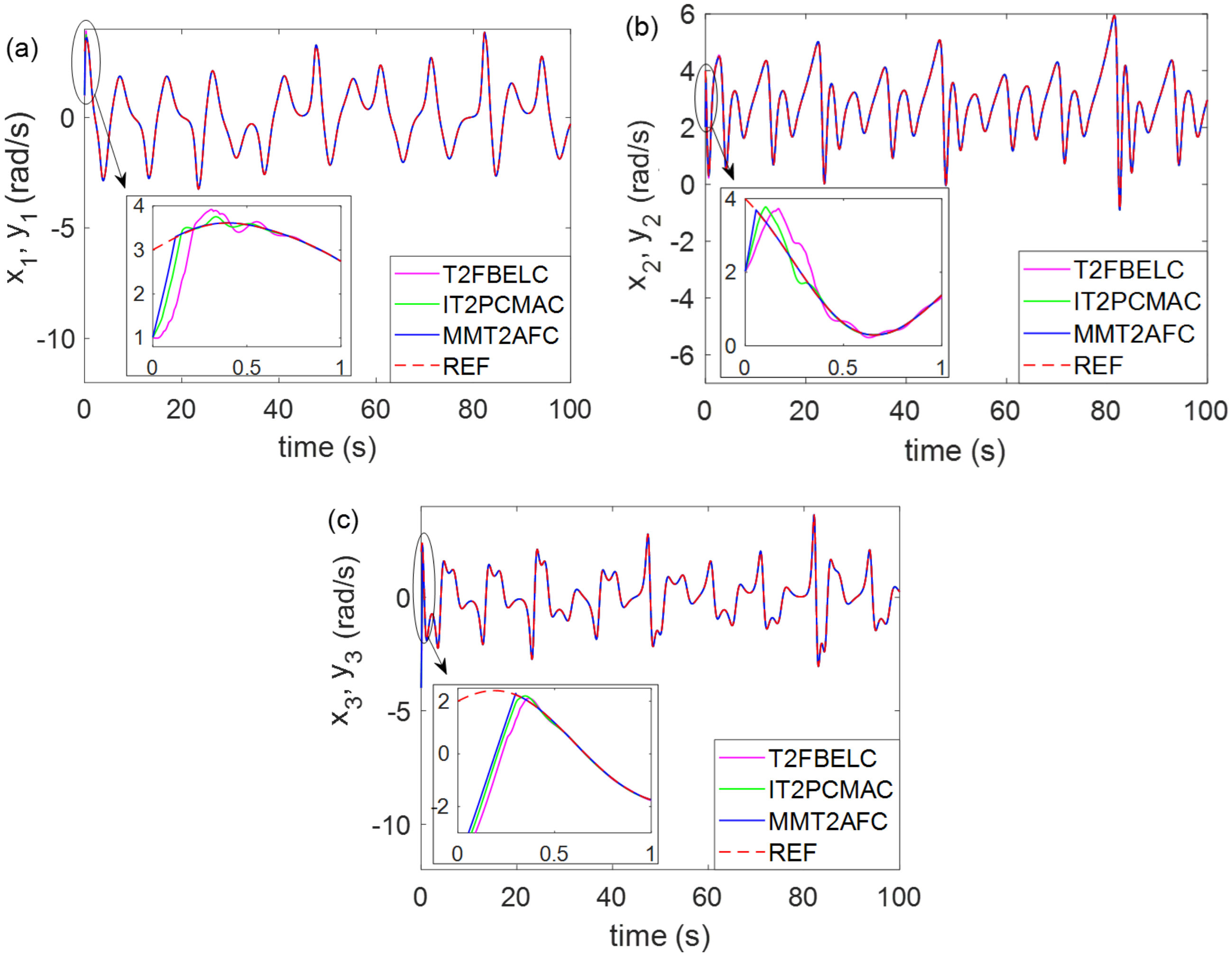


FIGURE 14. The time domain orbits of state variables. (a) x_1, y_1, (b) x_2, y_2, (c) x_3, y_3.

systems is defined as

$$\begin{aligned} e_1(t) &= y_1(t) - x_1(t) \\ e_2(t) &= y_2(t) - x_2(t) \\ e_3(t) &= y_3(t) - x_3(t) \end{aligned} \tag{58}$$

As in [74], we set $I_x = 3$, $I_y = 2$, $I_z = 1$. The initial conditions for the system are chosen as $\boldsymbol{x}(0) = [3, 4, 2]$ and $\boldsymbol{y}(0) = \lfloor 1, 2, -4 \rfloor$. The system uncertainties and external disturbances are chosen as $\Delta \boldsymbol{f}(t) = [0.8y_1, 0.8y_2, 0.8y_3]^T$ and $\boldsymbol{d}(t) = [\cos \pi t, 0.5 \cos t, 1.5 \cos 2t]^T$, respectively.

The dynamics state of the chaotic satellite system under the control of the MMT2AFC controller is shown in Fig. 13. In which, the dashed and dotted lines, respectively, correspond to the master system and slave system. Figure 14 shows the time domain orbits of the chaotic satellite state variables. Figures 15 and 16 respectively show the control signals and the tracking errors. The online adjustment in the number of layers using the self-evolving algorithm and the alteration of the learning rates using MGWO are shown in Figs. 17 and 18, respectively. These simulation results show the proposed MMT2AFC controller can control the slave satellite system effectively to synchronize with the master satellite system. Table 3 shows the comparison results in root mean square error (RMSE) among our proposed

TABLE 4. The statistical test results for example 1.

	T2FBELC	*MMT2AFC*	*IT2PCMAC*	*MMT2AFC*
Mean	0.9928	0.7493	0.8569	0.7493
Variance	9.43E-08	1.65E-07	1.21E-07	1.65E-07
df	18		18	
t_Stat	1510.83		635.75	
P_value	2.19E-47		1.28E-40	
CI	[0.2432	0.2438]	[0.1072	0.1080]

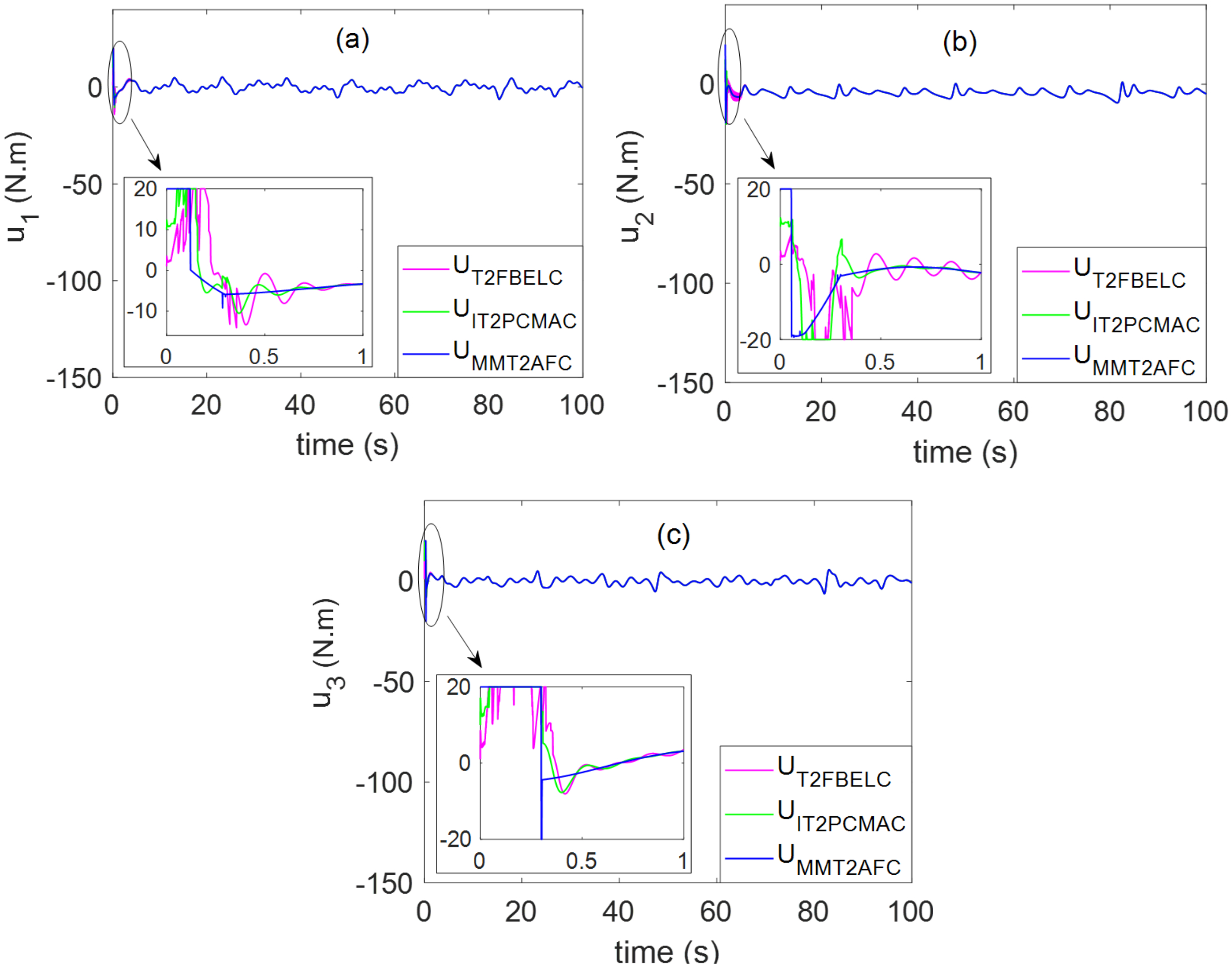


FIGURE 15. The time domain of control signals. (a) u_1, (b) u_2, (c) u_3.

TABLE 5. The statistical test results for example 2.

	T2FBELC	*MMT2AFC*	*IT2PCMAC*	*MMT2AFC*
Mean	0.3619	0.2753	0.3045	0.2753
Variance	3.65E-07	3.89E-07	4.44E-07	3.89E-07
df	18		18	
t_Stat	315.47		101.14	
P_value	3.83E-35		2.96E-26	
CI	[0.0860	0.0872]	[0.0286	0.0298]

df: degrees of freedom; ***t_Stat***: difference represented in standard-error units; ***P-value***: probability value; ***CI***: confidence interval.

controller and other recent controllers. Figures 8 and 14 show the MMT2AFC controller can rapidly achieve better control performance in synchronizing the chaotic systems, even when considering external disturbances and system uncertainties. The control signals in Figs 9 and 15 show the proposed controller can rapidly respond to the changes in the system, and it can control the system well from the beginning of the synchronization process. Figs. 10 and 16 show the proposed controller can make the slave chaotic system closely follow the master chaotic system faster than other controllers with the smallest RMSE tracking error. Figures 11 and 17 show, by using the self-evolving algorithm, at the beginning of the control process, the number of layers is rapidly generated to better cover the rapid changes in the tracking errors. After that, it can quickly adjust to the appropriate number of layers. Figures 12 and 18 show, by using the MGWO algorithm, the learning rates of the proposed controller can adjust to suitable values online. In both studied examples, even considering the system uncertainties and external disturbances, the proposed MMT2AFC achieves the fastest response, smallest RMSE tracking error, and highest synchronization performance. The *t-Test* statistical analysis for the different controllers is given in Tables 4, 5. In both examples, the statistical results show the *P_value* was lower than the alpha level

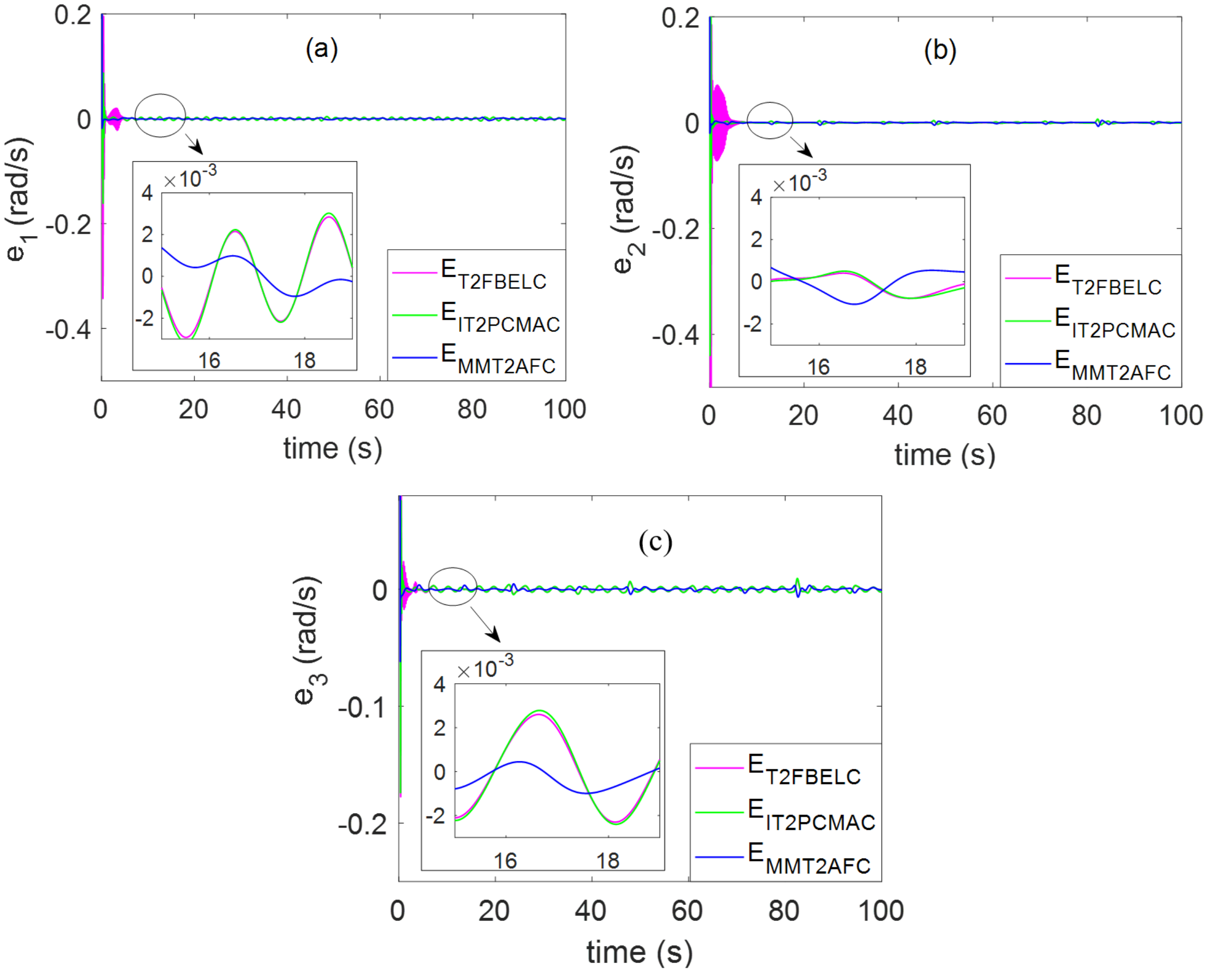


FIGURE 16. The time-domain of tracking errors. (a) e_1, (b) e_2, (c) e_3.

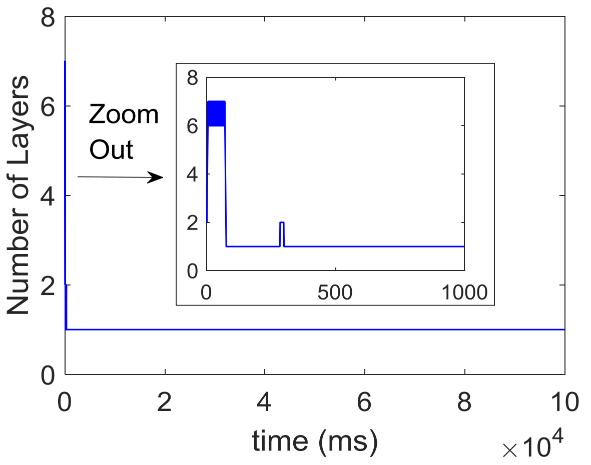


FIGURE 17. The number of layers using the self-evolving algorithm.

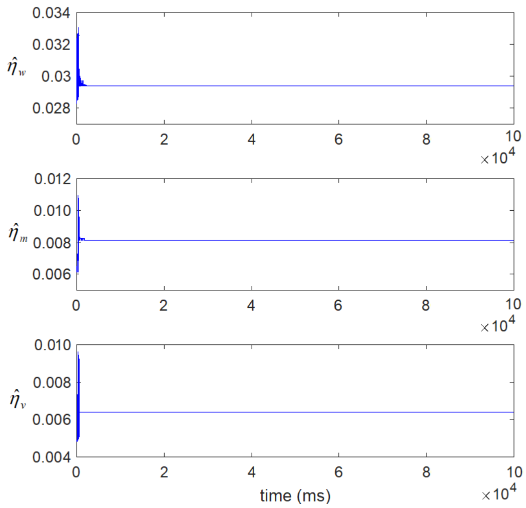


FIGURE 18. The alteration of the learning rates using the MGWO algorithm.

(α =0.05). Thus, we can conclude the RMSE results of our proposed controller had a statistically significant difference from other controllers. Therefore, the superiority of the proposed controller is illustrated.

V. CONCLUSION

This study developed a new MMT2AFC controller. The asymmetric membership function is applied to enhance system accuracy. The learning rate selection problems were considered for the proposed network using our modified grey wolf optimizer. Moreover, choosing a suitable network size for the designed controller was addressed by using the self-evolving algorithm. The adaptive laws were developed to update the controller's parameters online, and thus the tracking errors can be quickly converged and boundedness. The effectiveness of the proposed controller was verified by two examples of the synchronization of chaotic systems. Compared with some state-of-the-art methods, this method shows excellent performance for synchronizing the chaotic systems with the smallest RMSE tracking errors. Besides the application for synchronization problems, this method can be easily applied to a large number of nonlinear control systems. However, in the self-evolving algorithm, choosing thresholds for the growing and pruning rule significantly affect system performance. Applying the modern optimization method to estimate these thresholds will be our future work.

REFERENCES

[1] E. Kayacan and R. Maslim, "Type-2 fuzzy logic trajectory tracking control of quadrotor VTOL aircraft with elliptic membership functions," *IEEE/ASME Trans. Mechatronics*, vol. 22, no. 1, pp. 339–348, Feb. 2017.

[2] T.-L. Le, T.-T. Huynh, and C.-M. Lin, "Adaptive filter design for active noise cancellation using recurrent type-2 fuzzy brain emotional learning neural network," *Neural Comput. Appl.*, vol. 32, pp. 8725–8734, Jul. 2019.

[3] 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.

[4] O. Castillo, R. Martínez-Marroquín, P. Melin, F. Valdez, and J. Soria, "Comparative study of bio-inspired algorithms applied to the optimization of type-1 and type-2 fuzzy controllers for an autonomous mobile robot," *Inf. Sci.*, vol. 192, pp. 19–38, Jun. 2012.

[5] S.-K. Oh, H.-J. Jang, and W. Pedrycz, "A comparative experimental study of type-1/type-2 fuzzy cascade controller based on genetic algorithms and particle swarm optimization," *Expert Syst. Appl.*, vol. 38, no. 9, pp. 11217–11229, Sep. 2011.

[6] L. A. Zadeh, "The concept of a linguistic variable and its application to approximate reasoning—I," *Inf. Sci.*, vol. 8, no. 3, pp. 199–249, 1975.

[7] Q. Liang and J. M. Mendel, "Interval type-2 fuzzy logic systems: Theory and design," *IEEE Trans. Fuzzy Syst.*, vol. 8, no. 5, pp. 535–550, Oct. 2000.

[8] R.-C. Roman, R.-E. Precup, C.-A. Bojan-Dragos, and A.-I. Szedlak-Stinean, "Combined model-free adaptive control with fuzzy component by virtual reference feedback tuning for tower crane systems," *Procedia Comput. Sci.*, vol. 162, pp. 267–274, Jan. 2019.

[9] J. Z. Shi, "A fractional order general type-2 fuzzy PID controller design algorithm," *IEEE Access*, vol. 8, pp. 52151–52172, 2020.

[10] T.-L. Le, T.-T. Huynh, V.-Q. Nguyen, C.-M. Lin, and S.-K. Hong, "Chaotic synchronization using a self-evolving recurrent interval type-2 Petri cerebellar model articulation controller," *Mathematics*, vol. 8, no. 2, p. 219, Feb. 2020.

[11] A. Mohammadzadeh and E. Kayacan, "A novel fractional-order type-2 fuzzy control method for online frequency regulation in AC microgrid," *Eng. Appl. Artif. Intell.*, vol. 90, Apr. 2020, Art. no. 103483.

[12] J. Tang, L. Li, Z. Hu, and F. Liu, "Short-term traffic flow prediction considering spatio-temporal correlation: A hybrid model combing type-2 fuzzy C-means and artificial neural network," *IEEE Access*, vol. 7, pp. 101009–101018, 2019.

[13] V. Uslan, H. Seker, and R. John, "Overlapping clusters and support vector machines based interval type-2 fuzzy system for the prediction of peptide binding affinity," *IEEE Access*, vol. 7, pp. 49756–49764, 2019.

[14] G. Ruiz-Garcia, H. Hagras, H. Pomares, and I. R. Ruiz, "Toward a fuzzy logic system based on general forms of interval type-2 fuzzy sets," *IEEE Trans. Fuzzy Syst.*, vol. 27, no. 12, pp. 2381–2395, Dec. 2019.

[15] P. D'Alterio, J. M. Garibaldi, R. John, and A. Pourabdollah, "Constrained interval type-2 fuzzy sets," *IEEE Trans. Fuzzy Syst.*, early access, Jan. 31, 2020, doi: 10.1109/TFUZZ.2020.2970911.

[16] R. Sepúlveda, O. Montiel, O. Castillo, and P. Melin, "Embedding a high speed interval type-2 fuzzy controller for a real plant into an FPGA," *Appl. Soft Comput.*, vol. 12, no. 3, pp. 988–998, 2012.

[17] G. M. Méndez, P. N. M. Dorantes, and A. M. Santoyo, "Interval type-2 fuzzy logic systems optimized by central composite design to create a simplified fuzzy rule base in image processing for quality control application," *Int. J. Adv. Manuf. Technol.*, vol. 102, nos. 9–12, pp. 3757–3766, Jun. 2019.

[18] J. E. Moreno, M. A. Sanchez, O. Mendoza, A. Rodríguez-Díaz, O. Castillo, P. Melin, and J. R. Castro, "Design of an interval Type-2 fuzzy model with justifiable uncertainty," *Inf. Sci.*, vol. 513, pp. 206–221, Mar. 2020.

[19] C.-H. Lee, T.-W. Hu, C.-T. Lee, and Y.-C. Lee, "A recurrent interval type-2 fuzzy neural network with asymmetric membership functions for nonlinear system identification," in *Proc. IEEE Int. Conf. Fuzzy Syst. (IEEE World Congr. Comput. Intell.)*, Jun. 2008, pp. 1496–1502.

[20] H.-Y. Pan, C.-H. Lee, F.-K. Chang, and S.-K. Chang, "Construction of asymmetric type-2 fuzzy membership functions and application in time series prediction," in *Proc. Int. Conf. Mach. Learn. Cybern.*, vol. 4, 2007, pp. 2024–2030.

[21] T. Zhao, P. Li, and J. Cao, "Self-organising interval type-2 fuzzy neural network with asymmetric membership functions and its application," *Soft Comput.*, vol. 23, no. 16, pp. 7215–7228, Aug. 2019.

[22] S.-Y. Chen and T.-S. Liu, "Intelligent tracking control of a PMLSM using self-evolving probabilistic fuzzy neural network," *IET Electr. Power Appl.*, vol. 11, no. 6, pp. 1043–1054, Jul. 2017.

[23] D. Ge and X.-J. Zeng, "Learning evolving T-S fuzzy systems with both local and global accuracy—A local online optimization approach," *Appl. Soft Comput.*, vol. 68, pp. 795–810, Jul. 2018.

[24] T.-L. Le, C.-M. Lin, and T.-T. Huynh, "Self-evolving type-2 fuzzy brain emotional learning control design for chaotic systems using PSO," *Appl. Soft Comput.*, vol. 73, pp. 418–433, Dec. 2018.

[25] C.-M. Lin, T.-L. Le, and T.-T. Huynh, "Self-evolving function-link interval type-2 fuzzy neural network for nonlinear system identification and control," *Neurocomputing*, vol. 275, pp. 2239–2250, Jan. 2018.

[26] T.-L. Le, "Self-organizing recurrent interval type-2 Petri fuzzy design for time-varying delay systems," *IEEE Access*, vol. 7, pp. 10505–10514, 2019.

[27] J. Qiao, G. Wang, X. Li, and W. Li, "A self-organizing deep belief network for nonlinear system modeling," *Appl. Soft Comput.*, vol. 65, pp. 170–183, Apr. 2018.

[28] H.-G. Han, Z.-L. Lin, and J.-F. Qiao, "Modeling of nonlinear systems using the self-organizing fuzzy neural network with adaptive gradient algorithm," *Neurocomputing*, vol. 266, pp. 566–578, Nov. 2017.

[29] P. Kumar, R. Prasad, V. N. Mishra, D. K. Gupta, A. Choudhary, and P. K. Srivastava, "Artificial neural network with different learning parameters for crop classification using multispectral datasets," in *Proc. Int. Conf. Microw., Opt. Commun. Eng. (ICMOCE)*, Dec. 2015, pp. 204–207.

[30] T.-L. Le, T.-T. Huynh, and C.-M. Lin, "Self-evolving interval type-2 wavelet cerebellar model articulation control design for uncertain nonlinear systems using PSO," *Int. J. Fuzzy Syst.*, vol. 21, no. 8, pp. 2524–2541, Nov. 2019.

[31] C.-M. Lin and T.-L. Le, "PSO-self-organizing interval type-2 fuzzy neural network for antilock braking systems," *Int. J. Fuzzy Syst.*, vol. 19, no. 5, pp. 1362–1374, Oct. 2017.

[32] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," *Adv. Eng. Softw.*, vol. 69, pp. 46–61, Mar. 2014.

[33] M. Abdo, S. Kamel, M. Ebeed, J. Yu, and F. Jurado, "Solving non-smooth optimal power flow problems using a developed grey wolf optimizer," *Energies*, vol. 11, no. 7, p. 1692, Jun. 2018.

[34] Q. Tu, X. Chen, and X. Liu, "Multi-strategy ensemble grey wolf optimizer and its application to feature selection," *Appl. Soft Comput.*, vol. 76, pp. 16–30, Mar. 2019.

[35] R. A. Khanum, M. A. Jan, A. Aldegheishem, A. Mehmood, N. Alrajeh, and A. Khanan, "Two new improved variants of grey wolf optimizer for unconstrained optimization," *IEEE Access*, vol. 8, pp. 30805–30825, 2020.

[36] S. K. Goudos, T. V. Yioultsis, A. D. Boursianis, K. E. Psannis, and K. Siakavara, "Application of new hybrid Jaya grey wolf optimizer to antenna design for 5G communications systems," *IEEE Access*, vol. 7, pp. 71061–71071, 2019.

[37] M. Abdel-Basset, D. El-Shahat, I. El-henawy, V. H. C. de Albuquerque, and S. Mirjalili, "A new fusion of grey wolf optimizer algorithm with a two-phase mutation for feature selection," *Expert Syst. Appl.*, vol. 139, Jan. 2020, Art. no. 112824.

[38] Q. Al-Tashi, H. M. Rais, S. J. Abdulkadir, S. Mirjalili, and H. Alhussian, "A review of grey wolf optimizer-based feature selection methods for classification," in *Evolutionary Machine Learning Techniques*. Cham, Switzerland: Springer, 2020, pp. 273–286.

[39] M. H. Qais, H. M. Hasanien, and S. Alghuwainem, "A grey wolf optimizer for optimum parameters of multiple PI controllers of a grid-connected PMSG driven by variable speed wind turbine," *IEEE Access*, vol. 6, pp. 44120–44128, 2018.

[40] S. Gupta and K. Deep, "A memory-based grey wolf optimizer for global optimization tasks," *Appl. Soft Comput.*, vol. 93, Aug. 2020, Art. no. 106367.

[41] H. Faris, I. Aljarah, M. A. Al-Betar, and S. Mirjalili, "Grey wolf optimizer: A review of recent variants and applications," *Neural Comput. Appl.*, vol. 30, no. 2, pp. 413–435, Jul. 2018.

[42] N. Metawa, M. K. Hassan, and M. Elhoseny, "Genetic algorithm based model for optimizing bank lending decisions," *Expert Syst. Appl.*, vol. 80, pp. 75–82, Sep. 2017.

[43] A. A. Rahmani Hosseinabadi, J. Vahidi, B. Saemi, A. K. Sangaiah, and M. Elhoseny, "Extended genetic algorithm for solving open-shop scheduling problem," *Soft Comput.*, vol. 23, no. 13, pp. 5099–5116, Jul. 2019.

[44] X. Wu and A. Che, "A memetic differential evolution algorithm for energy-efficient parallel machine scheduling," *Omega*, vol. 82, pp. 155–165, Jan. 2019.

[45] A. W. Mohamed and A. K. Mohamed, "Adaptive guided differential evolution algorithm with novel mutation for numerical optimization," *Int. J. Mach. Learn. Cybern.*, vol. 10, no. 2, pp. 253–277, Feb. 2019.

[46] X. Gao, Y. Cui, J. Hu, G. Xu, Z. Wang, J. Qu, and H. Wang, "Parameter extraction of solar cell models using improved shuffled complex evolution algorithm," *Energy Convers. Manage.*, vol. 157, pp. 460–479, Feb. 2018.

[47] R. C. M. Gomes, M. A. Vitorino, M. B. de Rossiter Correa, D. A. Fernandes, and R. Wang, "Shuffled complex evolution on photovoltaic parameter extraction: A comparative analysis," *IEEE Trans. Sustain. Energy*, vol. 8, no. 2, pp. 805–815, Apr. 2017.

[48] Y. Zhou, Z. Bao, Q. Luo, and S. Zhang, "A complex-valued encoding wind driven optimization for the 0-1 knapsack problem," *Int. J. Speech Technol.*, vol. 46, no. 3, pp. 684–702, Apr. 2017.

[49] M. Lei, Y. Zhou, and Q. Luo, "Enhanced Metaheuristic optimization: Wind-driven flower pollination algorithm," *IEEE Access*, vol. 7, pp. 111439–111465, 2019.

[50] L. Shen, C. Fan, and X. Huang, "Multi-level image thresholding using modified flower pollination algorithm," *IEEE Access*, vol. 6, pp. 30508–30519, 2018.

[51] K. Li and Z. Tan, "An improved flower pollination optimizer algorithm for multilevel image thresholding," *IEEE Access*, vol. 7, pp. 165571–165582, 2019.

[52] Z. Lei, S. Gao, S. Gupta, J. Cheng, and G. Yang, "An aggregative learning gravitational search algorithm with self-adaptive gravitational constants," *Expert Syst. Appl.*, vol. 152, Aug. 2020, Art. no. 113396.

[53] M. Taradeh, M. Mafarja, A. A. Heidari, H. Faris, I. Aljarah, S. Mirjalili, and H. Fujita, "An evolutionary gravitational search-based feature selection," *Inf. Sci.*, vol. 497, pp. 219–239, Sep. 2019.

[54] S. Aslan and D. Karaboga, "A genetic artificial bee colony algorithm for signal reconstruction based big data optimization," *Appl. Soft Comput.*, vol. 88, Mar. 2020, Art. no. 106053.

[55] J.-Q. Li, M.-X. Song, L. Wang, P.-Y. Duan, Y.-Y. Han, H.-Y. Sang, and Q.-K. Pan, "Hybrid artificial bee colony algorithm for a parallel batching distributed flow-shop problem with deteriorating jobs," *IEEE Trans. Cybern.*, vol. 50, no. 6, pp. 2425–2439, Jun. 2020.

[56] D. Yan, H. Cao, Y. Yu, Y. Wang, and X. Yu, "Single-objective/multiobjective cat swarm optimization clustering analysis for data partition," *IEEE Trans. Autom. Sci. Eng.*, early access, Mar. 11, 2020, doi: 10.1109/TASE.2020.2969485.

[57] P. Bansal, S. Kumar, S. Pasrija, and S. Singh, "A hybrid grasshopper and new cat swarm optimization algorithm for feature selection and optimization of multi-layer perceptron," *Soft Comput.*, Mar. 2020, doi: 10.1007/s00500-020-04877-w.

[58] S. H. Hashemi Mehne and S. Mirjalili, "A direct method for solving calculus of variations problems using the whale optimization algorithm," *Evol. Intell.*, vol. 12, no. 4, pp. 677–688, Dec. 2019.

[59] W. Qiao, K. Huang, M. Azimi, and S. Han, "A novel hybrid prediction model for hourly gas consumption in supply side based on improved whale optimization algorithm and relevance vector machine," *IEEE Access*, vol. 7, pp. 88218–88230, 2019.

[60] M. A. Taher, S. Kamel, F. Jurado, and M. Ebeed, "Modified grasshopper optimization framework for optimal power flow solution," *Electr. Eng.*, vol. 101, no. 1, pp. 121–148, Apr. 2019.

[61] S. Arora and P. Anand, "Chaotic grasshopper optimization algorithm for global optimization," *Neural Comput. Appl.*, vol. 31, no. 8, pp. 4385–4405, Aug. 2019.

[62] M. A. Taher, S. Kamel, F. Jurado, and M. Ebeed, "An improved moth-flame optimization algorithm for solving optimal power flow problem," *Int. Trans. Electr. Energy Syst.*, vol. 29, no. 3, p. e2743, Mar. 2019.

[63] H. Jia, J. Ma, and W. Song, "Multilevel thresholding segmentation for color image using modified moth-flame optimization," *IEEE Access*, vol. 7, pp. 44097–44134, 2019.

[64] A. A. Al-Omoush, A. A. Alsewari, H. S. Alamri, and K. Z. Zamli, "Comprehensive review of the development of the harmony search algorithm and its applications," *IEEE Access*, vol. 7, pp. 14233–14245, 2019.

[65] A. Assad, K. Deep, N. Buckley, and A. K. Nagar, "Optimization of lycopene extraction from tomato processing waste skin using harmony search algorithm," in *Soft Computing for Problem Solving 2019*. Cham, Switzerland: Springer, 2020, pp. 141–154.

[66] R. Soto, B. Crawford, F. Gonzalez, E. Vega, C. Castro, and F. Paredes, "Solving the manufacturing cell design problem using human behavior-based algorithm supported by autonomous search," *IEEE Access*, vol. 7, pp. 132228–132239, 2019.

[67] Y. Wang, H. Liu, Z. Yu, and L. Tu, "An improved artificial neural network based on human-behaviour particle swarm optimization and cellular automata," *Expert Syst. Appl.*, vol. 140, Feb. 2020, Art. no. 112862.

[68] O. Olorunda and A. P. Engelbrecht, "Measuring exploration/exploitation in particle swarms using swarm diversity," in *Proc. IEEE Congr. Evol. Comput. (IEEE World Congr. Comput. Intell.)*, Jun. 2008, pp. 1128–1134.

[69] S. Barshandeh and M. Haghzadeh, "A new hybrid chaotic atom search optimization based on tree-seed algorithm and levy flight for solving optimization problems," *Eng. Comput.*, pp. 1–44, Feb. 2020, doi: 10.1007/s00366-020-00994-0.

[70] D. Biswas and T. Banerjee, "A simple chaotic and hyperchaotic time-delay system: Design and electronic circuit implementation," *Nonlinear Dyn.*, vol. 83, no. 4, pp. 2331–2347, Mar. 2016.

[71] C. Li, W. Joo-Chen Thio, J. C. Sprott, H. H.-C. Iu, and Y. Xu, "Constructing infinitely many attractors in a programmable chaotic circuit," *IEEE Access*, vol. 6, pp. 29003–29012, 2018.

[72] J. Petrzela and L. Polak, "Minimal realizations of autonomous chaotic oscillators based on trans-immittance filters," *IEEE Access*, vol. 7, pp. 17561–17577, 2019.

[73] H. Gan, S. Xiao, and Y. Zhao, "A novel secure data transmission scheme using chaotic compressed sensing," *IEEE Access*, vol. 6, pp. 4587–4598, 2018.

[74] A. Khan and S. Kumar, "T-S fuzzy modeling and predictive control and synchronization of chaotic satellite systems," *Int. J. Model. Simul.*, vol. 39, no. 3, pp. 203–213, Jul. 2019.

[75] Y. Scharf, "A chaotic outlook on biological systems," *Chaos, Solitons Fractals*, vol. 95, pp. 42–47, Feb. 2017.

[76] 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.

[77] S. Shao, M. Chen, and X. Yan, "Prescribed performance synchronization for uncertain chaotic systems with input saturation based on neural networks," *Neural Comput. Appl.*, vol. 29, no. 12, pp. 1349–1361, Jun. 2018.

[78] S. H. Hosseinnia, R. Ghaderi, A. Ranjbar N., M. Mahmoudian, and S. Momani, "Sliding mode synchronization of an uncertain fractional order chaotic system," *Comput. Math. Appl.*, vol. 59, no. 5, pp. 1637–1643, Mar. 2010.

[79] T.-L. Le, T.-T. Huynh, V.-Q. Nguyen, C.-M. Lin, and S.-K. Hong, "Chaotic synchronization using a self-evolving recurrent interval type-2 Petri cerebellar model articulation controller," *Mathematics*, vol. 8, no. 2, p. 219, Feb. 2020.

[80] Q. Zhou, F. Chao, and C.-M. Lin, "A Functional-link-based fuzzy brain emotional learning network for breast tumor classification and chaotic system synchronization," *Int. J. Fuzzy Syst.*, vol. 20, no. 2, pp. 349–365, Feb. 2018.

[81] L. Xu, H. Ma, and S. Xiao, "Exponential synchronization of chaotic Lur'e systems using an adaptive event-triggered mechanism," *IEEE Access*, vol. 6, pp. 61295–61304, 2018.

[82] A. Mohammadzadeh and S. Ghaemi, "Optimal synchronization of fractional-order chaotic systems subject to unknown fractional order, input nonlinearities and uncertain dynamic using type-2 fuzzy CMAC," *Nonlinear Dyn.*, vol. 88, no. 4, pp. 2993–3002, Jun. 2017.

[83] A. Boulkroune, A. Bouzeriba, and T. Bouden, "Fuzzy generalized projective synchronization of incommensurate fractional-order chaotic systems," *Neurocomputing*, vol. 173, pp. 606–614, Jan. 2016.

[84] C.-M. Lin and H.-Y. Li, "Self-organizing adaptive wavelet CMAC backstepping control system design for nonlinear chaotic systems," *Nonlinear Anal., Real World Appl.*, vol. 14, no. 1, pp. 206–223, Feb. 2013.

[85] U. E. Vincent, "Synchronization of identical and non-identical 4-D chaotic systems using active control," *Chaos, Solitons Fractals*, vol. 37, no. 4, pp. 1065–1075, Aug. 2008.

TIEN-LOC LE (Member, IEEE) was born in Vietnam, in 1985. He received the B.S. degree in electronics and telecommunication engineering from Lac Hong University, Vietnam, in 2009, the M.S. degree in electrical engineering from the Ho Chi Minh City University of Technology and Education, Vietnam, in 2012, and the Ph.D. degree in electrical engineering from Yuan Ze University, Taoyuan City, Taiwan, in January 2018. From February 2018 to December 2018, he was a Postdoctoral Research Fellow of the Department of Electrical Engineering, Yuan Ze University. He is currently a Postdoctoral Research Fellow of the Faculty of Mechanical and Aerospace Engineering, Sejong University, Seoul, South Korea. He is also a Lecturer with the Faculty of Mechatronics and Electronics, Lac Hong University. His research interests include intelligent control systems, fuzzy neural networks, type-2 fuzzy neural networks, and cerebellar model articulation controllers.

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 engineering from the Department of Electrical and Electronics Engineering, Ho Chi Minh City 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 City, Taiwan, in 2018. He is currently a Research Fellow of the Department of Electrical Engineering, Yuan Ze University. He is also a Lecturer with the Department of Electrical, Electronics, and Mechanical Engineering, Lac Hong University, Vietnam. His research interests include MCDM, fuzzy logic control, neural networks, cerebellar model articulation controllers, brain emotional learning-based intelligent controllers, deep learning, and intelligent control systems.

SUNG KYUNG HONG received the B.S. and M.S. degrees in mechanical engineering from Yonsei University, Seoul, South Korea, in 1987 and 1989, respectively, and the Ph.D. degree from Texas A&M University, College Station, in 1998. From 1989 to 2000, he was with the Unmanned Aerial Vehicle System Division and the Flight Dynamics and Control Laboratory of the Agency for Defense Development (ADD), South Korea. He is currently a Full Professor with the Department of Aerospace Engineering, Sejong University, South Korea. His research interests include fuzzy logic controls, inertial sensor applications, and flight control systems for unmanned vehicles.