Control in the loop for synchronization of nonlinear chaotic systems via adaptive intuitionistic neuro-fuzzy: a comparative study

Salah Helmy¹ · Mohamed Magdy² · Mohamed Hamdy¹

Received: 22 April 2021 / Accepted: 22 January 2022 / Published online: 24 February 2022
© The Author(s) 2022, corrected publication 2022

Abstract

This paper proposes control in the loop (CIL) for the synchronization between two nonlinear chaotic systems at the existence of uncertainties and disturbances using an adaptive intuitionistic neuro-fuzzy (AINF) control scheme. The chaotic systems have been subedited as one is the master and the other is the slave. They both have different initial conditions and parameters. The variation in the initial conditions leads to the butterfly effect, the concept that is well known in the chaos field and means that both systems diverge over time. Therefore, AINF control scheme has been proposed in this paper as a powerful scheme to get over this problem effectively. The main objective of using the AINF control scheme is that it collects the features of its contents. As the intuitionistic fuzzy gives the system flexibility and helps the controller collecting more information about the problem. The neural networks give the controller the ability to learn over time. The experimental results were obtained to verify the applicability and effectiveness of the proposed control scheme against external disturbance and model uncertainties with comparative study.

Keywords Chaotic systems · Synchronization · Adaptive intuitionistic neuro-fuzzy networks · Chua's oscillators

Introduction

Over the past few decades, chaotic systems have prompted considerable attention as it has applications in many fields. On account of their significant potential in the fields like secure communication [1], cryptosystems [2], neuro systems [3], chemical reactions [4], and information processing [5]. In biological systems, the control of chaos has a role as [6]. Besides being sensitive to initial conditions, chaotic systems have many complex features such as highly nonlinear systems with complex dynamics. Their sensitivity to initial conditions prompted Lorenz to call this phenomenon a butterfly effect. All these features make this field of research is challenging and attractive to the researchers. After that, the researchers paid attention to these types of systems, and other work has been raised on this topic such as a double-wing

chaotic system [7], double scroll attractor [8], Lure systems [9], and so on.

Due to all of these challenging features of the chaotic systems, they are difficult to be controlled. The most difficult is to make two chaotic systems have the same response. Depending on all of these points, chaos synchronization has been a very appealing issue over the past decades, besides it has many applications in many fields. The first chaos synchronization system was built by Pecora and Carroll in 1987 and they reviewed their work through [10]. Synchronization has a fundamental role in understanding a lot of physical phenomena in many fields. Chaos synchronization has an important role in model calibration [11] and secure communications [12]. Synchronization is important in many medical applications like some brain pathologies as Parkinson's disease [13], epilepsy [14], schizophrenia [15], and Alzheimer's disease [16]. It has a part in natural rhythms as heart beating [17]. Synchronization has been important in studying the nonlinear coupled networks of neurons in recent years [18–20].

Chaos synchronization attracted many researchers in the field of control, so different types of control schemes have been applied to synchronize chaotic systems such as adaptive

✉ Mohamed Hamdy
mohamed.hamdy@el-eng.menofia.edu.eg

¹ Department of Industrial Electronics and Control Engineering, Faculty of Electronic Engineering, Menoufia University, Menouf 32952, Egypt

² Department of Mechatronics Engineering, High Institute of Engineering and Technology, El-Mahalla El-Kobra 31951, Egypt

control [21, 22], active control [23], passive control [24], sliding mode control [25, 26], adaptive higher order PID control [27], and adaptive fuzzy control [28].

The controllers that use the expert and decision-making concepts have been hugely studied over the past decades such as fuzzy and neural systems. Because they could describe and deal with nonlinear systems, so they were more effective than classic control schemes to solve nonlinear problems, since Zadah proposed the fuzzy set (FS) theory which has been intensively studied especially in the field of control. The FS uses the human knowledge and linguistic variables to better analyze the system by defining a membership function $\mu(x)$. Therefore, many researchers have used FS-based controllers to control and synchronize chaotic systems such as [29, 30]. However, the assessment of the exact membership functions is troublesome due to inconsistencies associated with linguistic terminology, differences between experts, or data noise. Researchers and professionals also request for additional flexibility in the membership functions $\mu(x)$ nature, making it easier to mitigate the consequences of these uncertainties. This drives researchers to add another concept to describe the non-belongingness to the system, which has been named the non-membership function $\nu(x)$ [31]. Between the membership and non-membership functions, there is an uncertainty region described as $\pi(x) = 1 - \mu(x) - \nu(x)$ which has been named by a hesitancy margin. Atanassov named these proposed sets as intuitionistic fuzzy sets (IFS). After that, many studies have been published in this field. Intuitionistic fuzzy has been recently studied in various fields as decision-making [32] and image enhancement [33]; also, intuitionistic fuzzy has a recent role in control nonlinear systems [34] and for synchronization of chaotic systems [35].

The adaptive control technique is a method for designing automated control systems. It can be defined also as a way to adapt the parameters of the controller automatically to fix the undesired changes in the system as far as possible. This method necessitates the identification of the system, which can be done online or offline, as well as a controller adaption algorithm based on the recognition or approximation process model. There are many types of adaptive controllers that have been used to control nonlinear systems such as adaptive fuzzy control [36], adaptive finite-time tracking control [37], adaptive fault-tolerant control [38], neural network-based adaptive tracking control [39], and fuzzy H^∞ output feedback control [40]. Among these methods, neuro-fuzzy networks have demonstrated the potential to deal with nonlinear processes with high precision, reflecting the application of fuzzy systems to adaptive networks for the creation of fuzzy If–Then rules with acceptable membership functions as adaptive neuro-fuzzy (ANF) network [41]. In ANF network, the concepts of fuzzy and If–Then rules have been produced under the concept of neural systems as the fuzzy inference has been divided into layers presented in the form

of neural networks. As it could be noticed, the intuitionistic fuzzy is an extension of the fuzzy sets, and then, the integration between intuitionistic fuzzy and neural network is also possible. This means that a new concept of adaptive intuitionistic neuro-fuzzy (AINF) has been raised [42]. This can be happened by dividing the steps of the intuitionistic fuzzy into neural layers. Dividing the intuitionistic fuzzy control (IFC) method into neural layers, which gives it the ability to learn itself through any of the neural network learning methods based on backpropagation such as gradient descent and least-squares method.

Depending on the above, the contribution of this paper can be explained as follows:

- AINF control scheme has been proposed to control and synchronize two chaotic systems.
- The stability of the proposed controller has been guaranteed using the Lyapunov function.
- For practical validation, the proposed controller has been implemented using control in the loop (CIL) to verify the applicability and effectiveness of the proposed scheme against external disturbance and model uncertainties in practical implementation.
- A comparative study has been proposed depending on the CIL implementation between the proposed controller and the IFC.

The content of this paper has been arranged as follows. In the next section, we introduce the problem formulation. In the subsequent section, we discuss the basics and structure of the proposed controller. The experimental results for the two Chua oscillators with comparative study are presented in the penultimate section. The final section concludes this work and introduces our future research directions.

Problem formulation

Chaos synchronization is a phenomenon that happens when two or more chaotic systems must have the same output response or when one of the chaotic systems has to drive the others despite the difference in their initial conditions. This is difficult because of the difficult nature of the chaotic systems especially their sensitivity to initial conditions which is discussed in the literature. Then, in this problem, the master system imposes its dynamics on the slave system, so the slave system is synchronized using the proposed controller.

In this paper, two classes of chaotic systems one as a master and the other as a slave. The master system can be expressed as follows:

$$\dot{\underline{x}}(t) = A\underline{x}(t) + Bf(\underline{x}, t), \quad (1)$$

where $\underline{x} = [x_1, x_2, \dots, x_n] \in \mathcal{N}^n$ is the state vector of the master system, A, B are the constant matrices, and $f(\underline{x}, t)$ is the nonlinear function of the master system.

The slave system can be expressed as follows:

$$\dot{\underline{y}}(t) = A\underline{y}(t) + Bf(\underline{y}, t) + \Delta\underline{y}(t) + \underline{d}(t) + \underline{u}(t), \quad (2)$$

where $\underline{y} = [y_1, y_2, \dots, y_n] \in \mathcal{N}^n$ is the state vector of the slave system and $f(\underline{y}, t)$ is the nonlinear function of the slave system. $\Delta\underline{y}(t)$ is the uncertainty vector in the slave system, $\underline{d}(t)$ is the disturbance vector in the slave system, and $\underline{u}(t)$ is the control signal vector.

Then, the proposed controller has to synchronize the master and the slave systems depending on the error signal vector to converge it to zero.

Adaptive intuitionistic neuro-fuzzy control design

IFC strategy

FS are very popular, so the concept of fuzzy sets has been used by various control schemes such as Mamdani-based fuzzy controller (FC) and the proposed Mamdani-based IFC. IFC has an advantage over FC as it describes the degree of non-membership $v(x)$ besides the degree of membership $\mu(x)$ to the system which makes the controller better to define and deal with the vagueness and uncertainty in the system. For some real systems, it would be better to describe the non-belonging region to help the controller to get better performance. IFC is produced by [31] based on some main concepts as the following definitions show.

Definition 1 If there is an IF-set A in P , then it could be described as $A = \{ \langle p, \mu_A(p), v_A(p) \rangle | p \in P \}$ where $\mu_A(p) \in [0, 1]$, $v_A(p) \in [0, 1]$, $0 \leq \mu_A(p) + v_A(p) \leq 1 \forall p \in P$ where $\mu_A(p)$ is the membership function and $v_A(p)$ is the non-membership function.

Definition 2 Between the membership and non-membership function, there is a margin called the degree of hesitance defined as $\pi_A(p) = 1 - \mu_A(p) - v_A(p)$

Definition 3 The membership and non-membership fuzzy sets could have any type of popular fuzzy sets like triangular, trapezoidal, or Gaussian. The chosen type in this paper is Gaussian and is described for the membership and non-membership function as follows:

$$\mu_A(p) = e^{-\frac{(p-m_m)^2}{\sigma_m^2}}, \quad v_A(p) = 1 - e^{-\frac{(p-m_n)^2}{\sigma_n^2}}, \quad (3)$$

where the membership function mean is m_m and the standard deviation is σ_m . Also, the non-membership function mean is m_n and the standard deviation is σ_n .

Definition 4 In the IF inference system of Mamdani type, the i th IF (control) rules for membership function and non-membership function are

If $\mu_e(e_i)$ is $E_{i\mu}$ and $\mu_{\dot{e}}(\dot{e}_i)$ is $\dot{E}_{i\mu}$ Then $\mu(u_i)$ is $\min(E_{i\mu}, \dot{E}_{i\mu})$.

If $v_e(e_i)$ is E_{iv} and $v_{\dot{e}}(\dot{e}_i)$ is $\dot{E}_{iv}$ Then $v(u_i)$ is $\max(E_{iv}, \dot{E}_{iv})$.

The design of the proposed intuitionistic neuro-fuzzy (INF) network would consist of six layers, as shown in Fig. 1, where all the weights are considered to be 1.

INF network construction

Layer 1: In this layer, the input nodes are transmitting the crisp inputs to the next layer. The inputs of this layer named as r_i^1 , where i is the input number and 1 stands for layer 1. The net input of this layer $f_i^1 = r_i^1$, the activation function $a_i^1 = f_i^1$, and the output layer 1 is $o_i^1 = a_i^1$.

Layer 2: In this layer, the antecedents of the If–Then rules are determined by calculating the degree of membership and non-membership for each input. Then, the activation functions of this layer have been set as $a_{ij}^{2\mu}$ determine the membership function $\mu_j(r_i^2)$ of the inputs, where j is the membership function set number and i the input number and r_i^2 are the inputs for layer 2. The value of a_{ij}^{2v} determine the non-membership function $v_j(r_i^2)$ of the inputs. The net inputs $f_{ij}^{2\mu}$, f_{ij}^{2v} and activation functions $a_{ij}^{2\mu}$, a_{ij}^{2v} are as follows:

$$f_{ij}^{2\mu} = - \left(\frac{(r_i^2 - m_{ij}^\mu)^2}{\sigma_{ij}^{\mu^2}} \right), \quad f_{ij}^{2v} = - \left(\frac{(r_i^2 - m_{ij}^v)^2}{\sigma_{ij}^{v^2}} \right),$$

$$a_{ij}^{2\mu} = e^{f_{ij}^{2\mu}}, \quad a_{ij}^{2v} = 1 - e^{f_{ij}^{2v}}. \quad (4)$$

Then, the output of the neurons of this layer is also equal to the activation function of each neuron as $o_{ij}^{2\mu} = a_{ij}^{2\mu}$, $o_{ij}^{2v} = a_{ij}^{2v}$.

Layer 3: The consequents of each rule in the If–Then rules are calculated in this layer. The inputs of the layer are $r_i^{3\mu}$ and r_i^{3v} , where the net inputs $f_k^{3\mu}$, f_k^{3v} are calculated as follows:

$$f_k^{3\mu} = \min(r_1^{3\mu}, r_2^{3\mu}, \dots, r_n^{3\mu}),$$

$$f_k^{3v} = \max(r_1^{3v}, r_2^{3v}, \dots, r_n^{3v}) \quad (5)$$

as k is the number of the rule base fired, i takes the values from 1 to n , and n stands for the number of the inputs for

Fig. 1 The intuitionistic neuro-fuzzy network

layer 3. The output and the activation functions of this layer are equal to the net inputs.

Layer 4: In this layer, the operator max(min) is determined for every repeated consequent in the If–Then rules to find the final fuzzy output value for each output fuzzy sets. Let the input of this layer are $r_i^{4\mu}, r_i^{4v}$, and then, the net inputs of this layer are $f_p^{4\mu}, f_p^{4v}$ are calculated as follows:

$$\begin{aligned} f_p^{4\mu} &= \max(r_1^{4\mu}, r_2^{4\mu}, \dots, r_n^{4\mu}), \\ f_p^{4v} &= \min(r_1^{4v}, r_2^{4v}, \dots, r_n^{4v}), \end{aligned} \quad (6)$$

where p is the number of the output fuzzy sets. The output and the activation functions of this layer are equal to the net inputs.

Layer 5: In the literature, many defuzzification methods have been proposed by [43]. In this paper, the center of area

(COA) method has been used which is defined as

$$u = \frac{\sum_{i=1}^p [\mu_i(u) - v_i(u)] \bar{u}_i}{\sum_{i=1}^p [\mu_i(u) - v_i(u)]} \quad \text{for } \mu_i(u) \geq v_i(u), \quad (7)$$

where $\mu(u_i), v(u_i)$ are the membership and non-membership function of output, and $\bar{u}_i$ is the center of the output fuzzy sets and i is the output membership and non-membership functions number. The defuzzification method (7) would be done in two steps, the first in layer 5 and the second in layer 6. In layer 5, the inputs are $r_p^{5\mu} = \mu(u_i)$ and $r_p^{5v} = v(u_i)$. The net input of this layer is defined as

$$f_p^5 = r_p^{5\mu} - r_p^{5v}. \quad (8)$$

The output and the activation function of this layer are equal to the net input.

Layer 6: In this layer, there is one node to determine the crisp output. The input is the output of the last layer and is defined as $r_p^6 = f_p^5$. Then, the net input and the activation function in this layer as follows:

$$f^6 = \sum_{p=1}^p r_p^6 \bar{u}_i, \quad a^6 = \frac{f^6}{\sum_{p=1}^p r_p^6}. \quad (9)$$

The output of this layer is equal to the activation function. This control system would be adapted using the backpropagation gradient descent method as shown in the next subsection.

Learning of INF network using backpropagation

After establishing the whole INF network, then the network comes to the learning phase to adapt the parameters of membership and non-membership functions of the system inputs and output. In this paper, the learning phase has been done based on the idea of the backpropagation gradient descent method by minimizing the error function

$$E = \frac{1}{2}(y(t) - x(t))^2, \quad (10)$$

where $x(t)$ is the reference and $y(t)$ is the output of the system. For each control cycle, the inputs enter the input nodes at layer one in the forward pass. Then, the activity levels are computed for all the nodes in every layer until getting the output from the output layer. Then, from the output nodes, the value of $\frac{\partial E}{\partial q}$ is calculated for all hidden layers in a backward pass. Let m is the adaptable parameter in any node such as the center of output membership function, and then, the rule, Δm is proportional to $\frac{\partial E}{\partial q}$, is the learning rule which leads to

$$\underline{q}(t+1) = \underline{q}(t) + \eta \left(-\frac{\partial E}{\partial q} \right), \quad (11)$$

where the learning rate is η , and

$$\begin{aligned} \frac{\partial E}{\partial q} &= \frac{\partial E}{\partial (\text{net - output})} \frac{\partial (\text{net - output})}{\partial q} \\ &= \frac{\partial E}{\partial o} \frac{\partial o}{\partial q} = \frac{\partial E}{\partial o} \frac{\partial o}{\partial a} \frac{\partial a}{\partial f} \frac{\partial f}{\partial q}, \end{aligned} \quad (12)$$

where o is the output of the layer, q is updated parameter vector, f is the net input, and a is the net activation function.

In this paper, the chosen parameters to be adjusted are the center of the output $\bar{u}_i$ in (7) and the standard deviation of the input membership and non-membership functions in (4).

For computing the learning rule for those parameters, $\frac{\partial E}{\partial q}$ is computed layer by layer in the backward pass.

Learning rule in layer 6: using (11) and (12), the learning rule for the centers of the output $\bar{u}_i$ are calculated as follows:

$$\begin{aligned} \frac{\partial E}{\partial \bar{u}_i} &= \frac{\partial E}{\partial a^6} \frac{\partial a^6}{\partial f^6} \frac{\partial f^6}{\partial \bar{u}_i} = (y(t) - x(t)) \frac{r_p^6}{\sum_{p=1}^p r_p^6} \\ &= e(t) \frac{r_p^6}{\sum_{p=1}^p r_p^6}. \end{aligned} \quad (13)$$

Then, the centers of the output are updated by

$$\bar{u}_i(t+1) = \bar{u}_i(t) - \eta_1 e(t) \frac{r_p^6}{\sum_{p=1}^p r_p^6}. \quad (14)$$

The propagated error from this layer to layer 5 is that

$$\delta = e(t). \quad (15)$$

In layers 3, 4, and 5, there are no parameters to be updated. The propagated error δ has been passed only from one layer to the other until it reached layer 2, where there are parameters to be updated.

Learning rule in layer 2:

The parameters to be updated in this layer are the standard deviation of the membership and non-membership functions σ_{ij}^μ , σ_{ij}^v using the following learning rules:

$$\frac{\partial E}{\partial \sigma_{ij}^\mu} = \frac{\partial E}{\partial a^2} \frac{\partial a^2}{\partial f^2} \frac{\partial f^2}{\partial \sigma_{ij}^\mu} = e(t) e^{f_{ij}^{2\mu}} \left(\frac{2(r_i^2 - m_{ij}^\mu)^2}{(\sigma_{ij}^\mu)^3} \right) \quad (16)$$

$$\frac{\partial E}{\partial \sigma_{ij}^v} = \frac{\partial E}{\partial a^2} \frac{\partial a^2}{\partial f^2} \frac{\partial f^2}{\partial \sigma_{ij}^v} = e(t) (-e^{f_{ij}^{2v}}) \left(\frac{2(r_i^2 - m_{ij}^v)^2}{(\sigma_{ij}^v)^3} \right). \quad (17)$$

As $\frac{\partial E}{\partial a^2}$ is the propagated error δ from the previous layers. Then, the parameters are updated as follows:

$$\sigma_{ij}^\mu(t+1) = \sigma_{ij}^\mu(t) - \eta_2 e(t) e^{f_{ij}^{2\mu}} \left(\frac{2(r_i^2 - m_{ij}^\mu)^2}{(\sigma_{ij}^\mu)^3} \right) \quad (18)$$

$$\sigma_{ij}^v(t+1) = \sigma_{ij}^v(t) + \eta_3 e(t) e^{f_{ij}^{2v}} \left(\frac{2(r_i^2 - m_{ij}^v)^2}{(\sigma_{ij}^v)^3} \right). \quad (19)$$

The updating rates η_1 , η_2 , and η_3 are selected, so that the controller gives a good performance.

The steps of designing the proposed controller can be summarized as follows:

Step 1 The error and the derivative of error for the controlled states have been calculated and entered into the AINF network as its inputs.

Step 2 In the second layer of the network, the fuzzification process has proceeded.

Step 3 Through the network layers, the crisp output of the controller has been calculated to regulate the slave system. This is considered the forward path of the neural network.

Step 4 The backpropagation method for learning the neural network has been applied depending on the error and the updating rates through the updating laws.

Step 5 Steps 2, 3, and 4 have been repeated every cycle over the online control for adapting the chosen parameters for an effective synchronization.

Stability analysis

The stability analysis of the IFC system has been discussed in [35]. The main difference between IFC and AINF is that AINF has some parameters to be updated with the learning rates depending on the backpropagation method, which makes the stability of the AINF has been dependent on the learning rates. The learning rate has a significant impact on the convergence of the backpropagation that depends on the gradient descent method. During the training process, a large value of η may cause oscillation and instability, whereas a little value may cause the training to slow down. A Lyapunov function [44] can be used to find the best solution for each instance of time, and we may derive the following theorem.

Theorem 1 Consider there is a learning rate vector $\underline{\eta} = [\eta_1 \ \eta_2 \ \eta_3]^T$ for any parameter vector $\underline{q} = [\bar{u}_i(t) \ \sigma_{ij}^\mu(t) \ \sigma_{ij}^v(t)]^T$, as shown in (11), then the learning algorithm is convergent when $\underline{\eta}$ is under the following condition:

$$0 < \underline{\eta} < \frac{2}{\max_t \left\| \frac{\partial y(t)}{\partial \underline{q}} \right\|_2^2}. \quad (20)$$

Proof Let the candidate Lyapunov function

$$V(t) = \frac{1}{2}e^2(t) = \frac{1}{2}(y(t) - x(t))^2. \quad (21)$$

Then, the change of the candidate Lyapunov function is

$$\begin{aligned} \Delta V(t) &= V(t+1) - V(t) = \frac{1}{2} \left(e^2(t+1) - e^2(t) \right) \\ &= \frac{1}{2} \Delta e(t) (2e(t) + \Delta e(t)), \end{aligned} \quad (22)$$

where $\Delta e(t) = e(t+1) - e(t)$

Based on the total differential approximation formula, it's important to note that $e(t) = x(t) - y(t)$, and then

$$\Delta e(t) = \Delta \underline{q} \left[\frac{\partial e(t)}{\partial \underline{q}} \right]^T = \Delta \underline{q} \left[\frac{\partial y(t)}{\partial \underline{q}} \right]^T. \quad (23)$$

By taking $E(t) = \frac{1}{2}e(t)^2$, then

$$\Delta \underline{q} = -\underline{\eta} \frac{\partial E(t)}{\partial \underline{q}} = -\underline{\eta} e(t) \frac{\partial y(t)}{\partial \underline{q}}. \quad (24)$$

Therefore, the following equation can be obtained:

$$\begin{aligned} \Delta V(t) &\approx \frac{1}{2} \Delta \underline{q} \left[\frac{\partial y(t)}{\partial \underline{q}} \right]^T \left(2e(t) + \Delta \underline{q} \left[\frac{\partial y(t)}{\partial \underline{q}} \right]^T \right) \\ &= -\frac{1}{2} \underline{\eta} e(t) \frac{\partial y(t)}{\partial \underline{q}} \left[\frac{\partial y(t)}{\partial \underline{q}} \right]^T \\ &\quad \times \left(2e(t) - \underline{\eta} e(t) \frac{\partial y(t)}{\partial \underline{q}} \left[\frac{\partial y(t)}{\partial \underline{q}} \right]^T \right) \\ &= -\frac{1}{2} \underline{\eta} e^2(t) \left\| \frac{\partial y(t)}{\partial \underline{q}} \right\|_2^2 \left(2 - \underline{\eta} \left\| \frac{\partial y(t)}{\partial \underline{q}} \right\|_2^2 \right). \end{aligned} \quad (25)$$

Asymptotic stability is ensured by the Lyapunov stability theorem if $\Delta V(t) < 0$, so for convergence, the condition (20) can be concluded.

Results and discussion

Experimental setup

The performance of the AINF has been examined in this section through two uncertain chaotic systems using practical implementation depending on the CIL. Figure 2 shows the physical configuration of the CIL. As the controller has been implemented in an Arduino MEGA board. The other Arduino board is the interface between the controller board and the models in MATLAB. The master and the slave chaotic systems have been modeled using MATLAB. The MATLAB model sends a reference signal to the controller through an interface Arduino board. The controller computes the control signal and sends it back to the model through the interface board. The proposed controller is designed in the form of AINF like PID (AINF-PID), as shown in Fig. 3.

Experimental example

The proposed controller has been used to control and synchronize two of Chua's oscillators. Chua's oscillator is a

Fig. 2 **a** The synchronization responses using the CIL-AINF-PID. **b** The Arduino boards configuration

Fig. 3 Adaptive intuitionistic neuro-fuzzy control scheme

popular model that exhibits chaotic behavior. Therefore, a wide number of control techniques have been applied to synchronize two Chua oscillators such as adaptive control [41], active control [45], robust control [46], sliding mode control [47], and impulsive control [48].

Chua oscillator is a nonlinear dynamical system that gives chaotic behavior. This system was introduced by [49]. The dimensionless model of the piecewise linear Chua's oscillator has been heavily studied in many research as [35] and this model is defined as follows:

$$\begin{aligned}\dot{x}_1 &= \alpha_m(x_2 - x_1 - h(x_1)) \\ \dot{x}_2 &= x_1 - x_2 + x_3 \\ \dot{x}_3 &= -\beta_m x_2\end{aligned}\quad (26)$$

$$h(x_1) = -s_1 x_1 - \frac{1}{2}(s_0 - s_1)(|x_1 + 1| - |x_1 - 1|), \quad (27)$$

where α_m , β_m , s_0 , s_1 are the master system parameters and the suffix m stands for the master.

The piecewise linear Chua oscillator has been chosen as the master system. The slave system is chosen to be the Cubic

Chua oscillator. This system was firstly introduced by [50]. The nonlinear part of this system is cubic. Therefore, the slave system is given by

$$\begin{aligned}\dot{y}_1 &= \alpha_s \left(y_2 - \frac{1}{7}(2y_1^3 - y_1) \right) + u_1 \\ \dot{y}_2 &= y_1 - y_2 + y_3 \\ \dot{y}_3 &= -\beta_s y_2 + u_2,\end{aligned}\quad (28)$$

where α_s , β_s are the slave system parameters and the suffix s stands for the slave.

The controller has to synchronize the slave system with the master and minimizing the error. Therefore, the controller inputs are chosen as vectors of error and change of error as $[e(t) \ \dot{e}(t)]^T$ and the controller output is control signal vector $\underline{u}(t)$.

The vectors of error and the change of error are defined as

$$\underline{e}(t) = \underline{y}(t) - \underline{x}(t) \quad (29)$$

$$\underline{\dot{e}}(t) = \underline{\dot{y}}(t) - \underline{\dot{x}}(t). \quad (30)$$

Fig. 4 Membership functions of r_1^1 and r_2^1

Fig. 5 Non-membership functions of r_1^1 and r_2^1

The results of synchronization between the two chaotic systems based on AINF-PID would be represented in this section and compared with IFC-PID proposed in [35]. As mentioned in [35], the inputs have been chosen as $(e, \dot{e})$ in which $e = [e_1 \ 0 \ e_3]^T$ and $\dot{e} = [\dot{e}_1 \ 0 \ \dot{e}_3]^T$. Then, e_1 , $\dot{e}_1$, e_3 and $\dot{e}_3$ are considered to be layer 1 inputs r_1^1 , r_2^1 , r_3^1 and r_4^1 respectively.

The value of u_1 depends on the calculations based on the inputs e_1 , $\dot{e}_1$ and the value of u_3 depends on the calculations based on the inputs e_3 , $\dot{e}_3$. Because this paper would be compared with [35], the same basic structure of the controller would be used in this paper with the same controller gains and the same initials of the membership and non-membership functions. Therefore, the fuzzy sets for the inputs have been selected to be in the Gaussian formula as shown in (3). As [35], the output gains have been used in this paper as $K_1 = [3 \ 0.5]^T$, $K_2 = [3 \ 3]^T$. Also, the inputs fuzzified into five fuzzy sets, and the parameters of those fuzzy sets are chosen in [35], for the inputs r_1^1 and r_2^1 as $m_{i1}^\mu = m_{i1}^v = -0.3$ for negative big (NB) fuzzy set, $m_{i2}^\mu = m_{i2}^v = -0.15$ for negative small (NS), $m_{i3}^\mu = m_{i3}^v = 0$ for zero (Z), $m_{i4}^\mu = m_{i4}^v = 0.15$ for positive small (PS) and $m_{i5}^\mu = m_{i5}^v = 0.3$ for positive big (PB). The initial values of the fuzzy sets standard deviation for the r_1^1 and r_2^1 membership functions are $\sigma_{ij}^\mu = 0.15$ and for non-membership functions are $\sigma_{ij}^v = 0.4$. Those values of standard deviation are updated for each fuzzy set in every iteration according to (18) and (19), respectively. Figures 4 and 5 show the initial shapes of r_1^1 and r_2^1 membership and non-membership functions, respectively.

For the inputs r_3^1 and r_4^1 as $m_{i1}^\mu = m_{i1}^v = -1$ for (NB) fuzzy set, $m_{i2}^\mu = m_{i2}^v = -0.5$ for (NS), $m_{i3}^\mu = m_{i3}^v = 0$ for (Z), $m_{i4}^\mu = m_{i4}^v = 0.5$ for (PS) and $m_{i5}^\mu = m_{i5}^v = 1$ for (PB). The initial values of the fuzzy sets standard deviation

Fig. 6 Membership functions of r_3^1 and r_4^1

Fig. 7 Non-membership functions r_3^1 and r_4^1

for the r_3^1 and r_4^1 membership functions are $\sigma_{ij}^\mu = 0.5$ and for non-membership functions is $\sigma_{ij}^v = 1$. Those values of standard deviation are updated for each fuzzy set in every iteration according to (16) and (17), respectively. Figures 6 and 7 show the initial shapes of r_3^1 and r_4^1 membership and non-membership functions, respectively.

The controller output signals u_1 , u_3 , where u_1 has been used to control the first state and u_3 has been used to control the second state. They both have five intuitionistic fuzzy sets; the centers of those control signals have to be updated according to (14). The initial values of the centers for both control signals are the same, where $\bar{u}_{i1} = -3$, $\bar{u}_{i2} = -1.5$, $\bar{u}_{i3} = 0$, $\bar{u}_{i4} = 1.5$, $\bar{u}_{i5} = 3$, where i is the control signal number.

The comparison between the controllers would depend on the value of the summation of absolute error which is calculated as follows:

$$SAE = \sum_{t=0}^{t_f} |e(t)| dt, \quad (31)$$

where the final time is t_f . The parameters and the initial conditions of the systems have been chosen by [35], so the master's system parameters and the initial conditions are $\alpha_m = 9$, $\beta_m = 14.28$, $s_0 = 1.14$, $s_1 = 0.714$, $x_1(0) = 0$, $x_2(0) = 0$, and $x_3(0) = 0.6$, respectively. The slave's system parameters and the initial conditions are $\alpha_s = 10$, $\beta_s = 14.28$, $y_1(0) = 0.3$, $y_2(0) = 0$, and $y_3(0) = 0$, respectively.

Table 1 Updating rates of the AINF-PID controller

Controller	Parameters					
	η_{11}	η_{12}	η_{13}	η_{31}	η_{32}	η_{33}
AINF	0.201	0.001	0.0002	0.598	0	0

Table 2 Comparison between AINF-PID and IFC-PID based on SAE when they are operated at $t = 0$ s where no disturbance nor uncertainty

Controller	Index		
	SAE ₁	SAE ₂	SAE ₃
AINF-PID	13.2811	4.0475	26.7872
IFC-PID [35]	18.754	7.208	30.561

No disturbance nor uncertainty and $t = 0$ s

In this subsection, the AINF-PID has been used to control the system at $t = 0$ s. The updating rates have been chosen in Table 1. Then, based on these updating rates, the AINF-PID controller has been used to synchronize the master and the slave chaotic systems.

Table 2 shows a comparison between the proposed controller AINF-PID and IFC-PID based on SAE for the three states. Then, Fig. 8, shows the error states response for AINF-PID and IFC-PID. Figure 9 shows the response of the three states under the proposed controller.

Remark 1 The AINF-PID shows better performance than IFC-PID depending on the value of the performance index shown in Table 2. Figure 8 shows that AINF-PID has a faster error response than IFC-PID. Also, AINF-PID, and IFC-PID in this case at the same initial conditions and values are better than sliding mode control for the same problem proposed by [46].

Delayed control with no disturbance nor uncertainty

In this subsection, the updating rates have been chosen, as shown in Table 3. Then, based on these updating rates, the AINF-PID controller has been used to synchronize the master and the slave chaotic systems when it was activated at $t = 5$ s. Table 4 shows a comparison between the AINF-PID and IFC-PID based on SAE for the three states. After that, the error states of the proposed controller are shown in Fig. 10.

Remark 2 In this case, when the controller has been delayed for 5 s, then AINF-PID also shows a faster response than IFC-PID, as shown in Fig. 10. Table 4 also shows that AINF-PID has a better performance index than IFC-PID.

Delayed control with disturbance in the slave system

In this section, a disturbance vector has been added to the slave system to increase the difficulty of the controller's job. The disturbance vector has been as mentioned in [35] as $\underline{d} = [d_1 \ d_2 \ d_3]^T = [3\sin(10t) \ 0 \ 5]^T$. By adding this vector to the slave system (28), the system can be rewritten as

$$\begin{aligned}\dot{y}_1 &= \alpha_s \left(y_2 - \frac{1}{7} (2y_1^3 - y_1) \right) + d_1 + u_1 \\ \dot{y}_2 &= y_1 - y_2 + y_3 + d_2 \\ \dot{y}_3 &= -\beta_s y_2 + d_3 + u_2.\end{aligned}\quad (32)$$

The delay time has been changed in this sub-section to show that the efficiency of the AINF-PID is better than ordinary IFC-PID at any condition. The delay time has been chosen here as $t = 4$ s. Table 5 shows the comparison between AINF-PID and IFC-PID for this case when the controller is switched on after 4 s at the existence of disturbance vector $\underline{d}$. The updating rates have been changed and the new results of the updating rates are shown in Table 5. Based on Table 6, the AINF-PID shows better CIL indexes than IFC-PID. Then, Fig. 11 shows the error states response for AINF-PID and IFC-PID.

Remark 3 In this case, the challenge is raised where the controller has been delayed for 5 s in the presence of disturbance vector, and then, AINF-PID also shows better performance depending on the values of performance index shown in Table 6 and Fig. 11.

Delayed control with disturbance and uncertainty in the slave system

In this section, disturbance and uncertainty vectors have been added to the slave system. The controller is operated after a 4 s delay time. This case is the hardest for the controller not only because there are disturbances and uncertainties in the slave system but also the disturbance vector has been changed with time. The new disturbance vector is $\underline{d} = [3\sin(10t) \ 0 \ 2t]^T$, where the step disturbance in the former section has been changed to be a ramp. The difficulty is that ramp disturbances change their value with time. The uncertainty vector is $\Delta \underline{y}(t) = [2\Delta y_1 \ 3\Delta y_2 \ 3\Delta y_3]^T$. Therefore, the slave system after adding the uncertainty and

Fig. 8 Error states for the non-delayed system at no disturbance nor uncertainty:
a CIL-AINF-PID;
b CIL-IFC-PID

(a)

(b)

Fig. 9 State responses for the non-delayed system at no disturbance nor uncertainty:
 CIL-AINF-PID **a** state 1; **b** state 2; **c** state 3

(a)

(b)

(c)

Table 3 Updating rates of the AINF-PID controllers

Controller	Parameters					
	η_{11}	η_{12}	η_{13}	η_{31}	η_{32}	η_{33}
AINF-PID	0.4	0.8	0.057	0.007	0.004	0.005

Table 4 Comparison between AINF-PID and IFC-PID based on SAE when they are operated at $t = 5$ s where no disturbance nor uncertainty

Controller	Index		
	SAE ₁	SAE ₂	SAE ₃
AINF-PID	1.4231	6.6216	5.3350
IFC-PID [35]	13.2105	10.3483	5.2698

Fig. 10 Error states for 5 s delayed system at no disturbance nor uncertainty:
a CIL-AINF-PID;
b CIL-IFC-PID

(a) **(b)****Table 5** Updating rates of the controllers

Controller	Parameters					
	η_{11}	η_{12}	η_{13}	η_{31}	η_{32}	η_{33}
AINF-PID	0	0.8	0	0	0.207	0

Table 6 Comparison between AINF-PID and IFC-PID based on SAE when they are operated at $t = 4$ s with disturbance but without uncertainty

Controller	Index		
	SAE ₁	SAE ₂	SAE ₃
AINF-PID	12.3105	4.9887	7.4486
IFC-PID [35]	20.161	7.715	8.831

Fig. 11 Error states for 5 s delayed system at the existence of disturbance but no uncertainty: **a** CIL-AINF-PID; **b** CIL-IFC-PID

Table 7 Updating rates of the controllers

Controller	Parameters					
	η_{11}	η_{12}	η_{13}	η_{31}	η_{32}	η_{33}
AINF	0.002	0.04	0	0.001	0.645	0

disturbance vectors has been shown as follows:

$$\begin{aligned}\dot{y}_1 &= \alpha_s \left(y_2 - \frac{1}{7} (2y_1^3 - y_1) \right) + d_1 + 2\Delta y_1 + u_1 \\ \dot{y}_2 &= y_1 - y_2 + y_3 + d_2 + 3\Delta y_2 \\ \dot{y}_3 &= -\beta_s y_2 + d_3 + 3\Delta y_3 + u_2.\end{aligned}\quad (33)$$

The updating rates have been chosen, as shown in Table 7. Then, based on these updating rates, the AINF-PID controller has been used to synchronize the master and the slave chaotic systems after 5 s delay and with disturbance and uncertainty added to the slave system. Table 8 shows a comparison between the proposed AINF-PID controller and IFC-PID based on SAE for the three states. After that, the error states of the proposed controller are shown in Fig. 12.

Remark 4 This is the most difficult case for the controller, where the controller has been delayed for 5 s in the presence of disturbance and uncertainty vector, but even with these difficulties, the AINF-PID shows a faster response than IFC-PID, as shown in Fig. 10, with better performance index, as shown in Table 4.

The flexibility of the neurons gives the AINF-PID an advantage over IFC-PID, because there are some main parameters in the membership and non-membership of the

inputs and outputs adapting with the change of error over time to reach the best performance. This advantage has been proved by the last simulation and CIL results.

Conclusion

An AINF has been introduced as a solution for the synchronization of two chaotic systems. This proposed controller has been compared with IFC using two CIL experimental examples. The first example has four test cases. The second one has three test cases. The experimental results discussed the test cases to show the performance of the proposed AINF scheme associated by a comparison with IFC scheme. All these cases have been introduced to evaluate the efficiency and the robustness of the proposed controller when compared to a former effective IFC scheme. The performance has been indicated by SAE indices and some figures. The overall performance indices and figures for each case in both examples show that the proposed AINF scheme is very effective and better than the IFC, because there are some main parameters in the neural network that have been adapted using the proposed adaptation rates. In the future work, it is recommended to use the AINF scheme based on the Takagi–Sugeno rule

Table 8 Comparison between AINF-PID and IFC-PID based on SAE when they are operated at $t = 4$ s with disturbance and uncertainty

Controller	Index		
	SAE ₁	SAE ₂	SAE ₃
AINF	13.2702	34.2115	30.6148
IFC [35]	19.585	46.619	50.668

Fig. 12 Error states for 5 s delayed system at the existence of disturbance and uncertainty:
a CIL-AINF-PID;
b CIL-IFC-PID

base to control and synchronize some classes of chaotic systems such as a fractional-order chaotic system.

Funding Not applicable.

Declarations

Conflict of interest Not applicable.

Availability of data and materials Not applicable.

Code availability Not applicable.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

1. Wang DM, Wang LS, Guo YY et al (2019) Key space enhancement of optical chaos secure communication: chirped FBG feedback semiconductor laser. *Opt Express* 27:3065–3073
2. Murillo-Escobar MA, Meranza-Castillon MO, Lopez-Gutierrez RM et al (2019) Suggested integral analysis for chaos-based image cryptosystems. *Entropy* 21(8):815
3. Gao J, Fan J, Wu BW et al (2016) Entrainment of chaotic activities in brain and heart during MBSR mindfulness training. *Neurosci Lett* 616:218–223
4. Berezowski M (2020) Chaos predictability in a chemical reactor. *Int J Bifurc Chaos* 30(11):2050221
5. Justin M, Zdravkovic S, Hubert MB et al (2020) Chaotic vibration of microtubules and biological information processing. *Biosystems* 198:104230
6. Ali AM, Ramadhan SM, Tahir FR (2019) A novel 2D-grid of scroll chaotic attractor generated by CNN. *Symmetry* 11(1):99
7. Sambas A, Vaidyanathan S, Zhang S et al (2019) A new double-wing chaotic system with coexisting attractors and line equilibrium: bifurcation analysis and electronic circuit simulation. *IEEE Access* 7:115454–115462
8. Hong Q, Xie Q, Xiao P (2016) A novel approach for generating multi-direction multi-double-scroll attractors. *Nonlinear Dyn* 87:1015–1030
9. Zeng D, Zhang R, Liu Y et al (2017) Sampled-data synchronization of chaotic Lur'e systems via input-delay-dependent-free-matrix zero equality approach. *Appl Math Comput* 315:34–46
10. Pecora L, Carroll T (2015) Synchronization of chaotic systems. *Chaos* 25:097611

11. Pillai N, Schwartz SL, Ho T et al (2019) Estimating parameters of nonlinear dynamic systems in pharmacology using chaos synchronization and grid search. *J Pharmacokinet Pharmacodyn* 46:193–210
12. Vaseghi B, Pourmina MA, Mobayen S (2017) Finite-time chaos synchronization and its application in wireless sensor networks. *Trans Inst Meas Control* 40:3788–3799
13. Dash S, Abraham A, Luhach AK et al (2020) Hybrid chaotic firefly decision making model for Parkinson's disease diagnosis. *Int J Distrib Sens Netw* 16(1):1–18
14. Panahi S, Shirzadian T, Jalili M, Jafari S (2019) A new chaotic network model for epilepsy. *Appl Math Comput* 346:395–407
15. Bowyer SM, Gjini K, Zhu X et al (2015) Potential biomarkers of schizophrenia from MEG resting-state functional connectivity networks: preliminary data. *J Behav Brain Sci* 5(1):1–11
16. Babiloni C, Lizio R, Marzano N et al (2016) Brain neural synchronization and functional coupling in Alzheimer's disease as revealed by resting state EEG rhythms. *Int J Psychophysiol* 103:88–102
17. Kumar P, Parmananda P (2018) Control, synchronization, and enhanced reliability of aperiodic oscillations in the mercury beating heart system. *Chaos* 28:045105
18. Li C-H, Yang S-Y (2015) Eventual dissipativeness and synchronization of nonlinearly coupled dynamical network of Hindmarsh-Rose neurons. *Appl Math Model* 39(21):6631–6644
19. Malik S, Mir AJNN (2020) Synchronization of Hindmarsh Rose neurons. *Neural Netw* 123:372–380
20. Ge M et al (2019) Wave propagation and synchronization induced by chemical autapse in chain Hindmarsh-Rose neural network. *Appl Math Comput* 352:136–145
21. Aghababa MP, Haghighi AR, Roohi M (2015) Stabilisation of unknown fractional-order chaotic systems: an adaptive switching control strategy with application to power systems. *IET Gener Transm Distrib* 9:1883–1893
22. Yaghooti B, Siah Shadbad A, Safavi K et al (2019) Adaptive synchronization of uncertain fractional-order chaotic systems using sliding mode control techniques. *Proc Inst Mech Eng Part I J Syst Control Eng* 234:3–9
23. Radwan AG, Moaddy K, Salama KN et al (2014) Control and switching synchronization of fractional order chaotic systems using active control technique. *J Adv Res* 5:125–132
24. Lai B-C, He J-J (2018) Dynamic analysis, circuit implementation and passive control of a novel four-dimensional chaotic system with multiscroll attractor and multiple coexisting attractors. *Pramana* 90(3):33
25. Pai MC (2015) Chaotic sliding mode controllers for uncertain time-delay chaotic systems with input nonlinearity. *Appl Math Comput* 271:757–767
26. Xue Y, Zheng BC, Li T et al (2017) Robust adaptive state feedback sliding-mode control of memristor-based Chua's systems with input nonlinearity. *Appl Math Comput* 314:142–153
27. Zirkohi MM, Khorashadizadeh S (2018) Chaos synchronization using higher-order adaptive PID controller. *AEU-Int J Electron Commun* 94:157–167
28. Boulkroune A, Bouzeriba A, Hamel S et al (2015) Adaptive fuzzy control-based projective synchronization of uncertain nonaffine chaotic systems. *Complexity* 21:180–192
29. Li SY, Tam LM, Tsai SE et al (2016) Novel fuzzy modeling and synchronization of chaotic systems with multilinear terms by advanced ge-li fuzzy model. *IEEE Trans Cybern* 45:2228–2237
30. Lee CH, Chang FY, Lin CM (2014) An efficient interval type-2 fuzzy CMAC for chaos time-series prediction and synchronization. *IEEE Trans Cybern* 44:329–341
31. Atanassov K (1986) Intuitionistic fuzzy sets. *Fuzzy Sets Syst* 20:87–96
32. Hinduja A, Pandey M (2019) A distance-based method for computing priorities of intuitionistic fuzzy preference relation and its application in AHP. *Vision* 23:329–340
33. Deng H, Sun X, Liu M et al (2016) Image enhancement based on intuitionistic fuzzy sets theory. *IET Image Proc* 10:701–709
34. Hamdy M, Helmy S, Magdy M (2020) Design of adaptive intuitionistic fuzzy controller for synchronisation of uncertain chaotic systems. *CAAI Trans Intell Technol* 5(4):237–246
35. Hamdy M, Magdy M, Helmy S (2021) Control and synchronization for two Chua systems based on intuitionistic fuzzy control scheme: a comparative study. *Trans Inst Meas Control* 43(7):1650–1667
36. Zhou P, Zhang L, Zhang S, Alkhateeb A (2020) Observer-based adaptive fuzzy finite-time control design with prescribed performance for switched pure-feedback nonlinear systems. *IEEE Access* 9:69481–69491
37. Chang Y, Zhang S, Alotaibi N, Alkhateeb A (2020) Observer-based adaptive finite-time tracking control for a class of switched nonlinear systems with unmodeled dynamics. *IEEE Access* 8:204782–204790
38. Wang Y, Xu N, Liu Y, Zhao X (2021) Adaptive fault-tolerant control for switched nonlinear systems based on command filter technique. *Appl Math Comput* 392:125725
39. Wang Y, Niu B, Wang H, Alotaibi N, Abozinadah E (2021) Neural network-based adaptive tracking control for switched nonlinear systems with prescribed performance: an average dwell time switching approach. *Neurocomputing* 435:295–306
40. Wu B, Chang X, Zhao X (2020) Fuzzy H_∞ output feedback control for nonlinear NCSs with quantization and stochastic communication protocol. *IEEE Trans Fuzzy Syst* 29(9):2623–2634
41. Lin YC, Nguyen HLT (2020) Adaptive neuro-fuzzy predictor-based control for cooperative adaptive cruise control system. *IEEE Trans Intell Transp Syst* 21:1054–1063
42. Hajek P, Olej V (2017) Intuitionistic neuro-fuzzy network with evolutionary adaptation. *Evol Syst* 8:35–47
43. Angelov P (1995) Crispification: defuzzification over intuitionistic fuzzy sets. *Bull Stud Exch Fuzziness Appl* 64:51–55
44. Feng S, Chen CP (2019) Nonlinear system identification using a simplified fuzzy broad learning system: stability analysis and a comparative study. *Neurocomputing* 337:274–286
45. Mahmoud GM, Bountis T, Mahmoud EE (2007) Active control and global synchronization of the complex Chen and Lü systems. *Int J Bifurc Chaos* 17:4295–4308
46. Li Z, Zhao X (2016) New results on robust control for a class of uncertain systems and its applications to Chua's oscillator. *Nonlinear Dyn* 84:1929–1941
47. Mufti MR, Afzal H, Rehman FU et al (2018) Synchronization and antisynchronization between two non-identical Chua oscillators via sliding mode control. *IEEE Access* 6:45270–45280
48. Sun J, Zhang Y (2004) Impulsive control and synchronization of Chua's oscillators. *Math Comput Simul* 66:499–508
49. Chua L, Komuro M, Matsumoto T (1986) The double scroll family. *IEEE Trans Circuits Syst* 33:1072–1118
50. Huang A, Pivka L, Wu CW et al (1996) Chua's equation with cubic nonlinearity. *Int J Bifurc Chaos* 6:2175–2222

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.