

Research Article

Synchronization of Discrete-Time Chaotic Fuzzy Systems by means of Fuzzy Output Regulation Using Genetic Algorithm

Tonatiuh Hernández Cortés,¹ A. Verónica Curtidor López,¹ Jorge Rodríguez Valdez,¹ Jesús A. Meda Campaña,¹ Ricardo Tapia Herrera,¹ and José de Jesús Rubio²

¹*Instituto Politécnico Nacional, SEPI-ESIME Zacatenco, Avenida IPN S/N, 07738 México, DF, Mexico*

²*Instituto Politécnico Nacional, SEPI-ESIME Azcapotzalco, Avenida de las Granjas No. 682, 02250 Azcapotzalco, México, DF, Mexico*

Correspondence should be addressed to Tonatiuh Hernández Cortés; tona_hernand@yahoo.com.mx

Received 7 August 2015; Accepted 4 November 2015

Academic Editor: Rongwei Guo

Copyright © 2015 Tonatiuh Hernández Cortés et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The synchronization of chaotic systems, described by discrete-time T-S fuzzy models, is treated by means of fuzzy output regulation theory. The conditions for designing a discrete-time output regulator are given in this paper. Besides, when the system does not fulfill the conditions for exact tracking, a new regulator based on genetic algorithms is considered. The genetic algorithms are used to approximate the adequate membership functions, which allow the adequate combination of local regulators. As a result, the tracking error is significantly reduced. Both the Complete Synchronization and the Generalized Synchronization problem are studied. Some numerical examples are used to illustrate the effectiveness of the proposed approach.

1. Introduction

A special nonlinear dynamical phenomenon, known as chaos, emerged in mid-1960s and reached applicable technology in the late 1990s and was considered as one of the three monumental discoveries of the twentieth century. On the other hand, fuzzy logic, a set theory and then an infinite-valued logic, gets a wide applicability in many industrial, commercial, and technical fields, ranging from control, automation, and artificial intelligence, just to name a few. Fuzzy logic and chaos had been considered by many researches and engineers as fundamental concepts and theories and their broad applicability in technology as well. The interaction between fuzzy logic and chaos has been developed for the last 20 years leading to research topics as fuzzy modeling of chaotic systems using Takagi-Sugeno models, linguistic descriptions of chaotic systems, fuzzy control of chaos, synchronization, and a combination of fuzzy chaos for engineering applications [1, 2].

In the 1960s, Rechenberg [3] introduced “evolution strategies,” a method to optimize real-valued parameters for

devices such as airfoils. This idea was further developed by Schwefel in [4]. Genetic algorithms (GAs) were initially developed by Bremermann [5] in 1958 but popularized and developed by Holland in the 1960s. In contrast with evolution strategies and evolutionary programming, Holland’s idea was not to design algorithms to solve specific problems but rather to formally study the phenomenon of adaptation, as it occurs in nature, and develop ways in which the mechanisms of natural adaptation might be transferred into computer systems [6]. The genetic algorithm is presented as abstraction of biological evolution and theoretical framework for adaptation for moving from one population of “chromosomes” (e.g., strings of ones and zeros, or “bits”) to a new population by using a kind of “natural selection” together with the genetics-inspired operators of crossover, mutation, and inversion. Each chromosome consists of “genes” (e.g., bits); each gene is being an instance of a particular “allele” (e.g., 0 or 1). The operator selection chooses those chromosomes in the population that will be allowed to reproduce and those adjusted chromosomes produce more offspring than the less ones [7].

According to Fogel and Aderson [8], Bremermann was the first to implement real-coded genetic algorithms as well as providing a mathematic model of GA known as the one-max function. In contrast to genetic algorithms, Evolutionary Strategies were initially developed for the purpose of parameter optimization. The idea was to imitate the principles of organic evolution in experimental parameter optimization for applications such as pipe bending or PID control for a nonlinear system [9].

Synchronization of chaotic systems is one of the more exiting problems in control science and can be referred at least to Huygen's observations [10]; it is understood as one of the trajectories of two autonomous chaotic systems, starting from nearly initial conditions and converging to the other, and remains as $t \rightarrow \infty$; in [11] it was reported that some kind of chaotic systems possesses a self-synchronization property. However, not all chaotic systems can be decomposed in two separate responses subsystems and be able to synchronize the drive system. The ideas of these works have led to improvement in many fields, such as communications [12], encrypted systems, the complex information processing within the human brain, coupled biochemical reactors, and earthquake engineering [13].

Synchronization can be classified as follows: Complete Synchronization: it is when two identical chaos oscillators are mutually coupled and one drives to the other; Generalized Synchronization: it differs from the previous case by the fact that there are different chaos oscillators and the states of one are completely defined by the other; Phase Synchronization: it occurs when the coupled oscillators are not identical and have different amplitude that is still unsynchronized, while the phases of oscillators evolve to be synchronized [14]. It is worth mentioning that studies in synchronization of nonlinear systems have been reformulated based on the previous results from classical control theory such as [15–18].

In this paper, the fuzzy output regulation theory and Takagi-Sugeno (T-S) fuzzy models are used to solve the Complete and Generalized Synchronization by using linear local regulators. Isidori and Byrnes [19] showed that the output regulation established by Francis could be extended for a nonlinear sector as a general case, resulting in a set of nonlinear partial differential equations called Francis-Isidori-Byrnes (FIB). Unfortunately these equations in many cases are too difficult to solve in a practical manner. For this reason in [20] the approach based on the weighted summation of local linear regulators is presented and in [21] the new membership functions in the regulator are approximated by soft computing techniques.

So, the main contribution of the present work is to find a control law for synchronizing of chaotic systems described by discrete-time Takagi-Sugeno fuzzy models, first when the system fulfills the following: (1) the input matrix for all subsystems is the same and (2) the local regulators share the same zero error manifold $\pi(w_k)$. In this way, the results given in [20] are extended to the discrete-time domain. On the other hand, when the system *master-slave* does not fulfill the aforementioned conditions, new membership functions are computed in order to enhance the performance of the fuzzy regulator. Such proposed membership functions are

different from those given in the plant or exosystem and are tuned by using the GA. The tuning of the new membership functions, which is as generalized bell-shaped function, is given by optimization of the form parameter.

The rest of the paper is organized as follows. In Section 2 the discrete-time output regulation problem formulation is given with a brief review of the Takagi-Sugeno models and the discrete-time fuzzy regulation problem. In Section 3 the tuning of membership functions by means of GAs is thoroughly discussed. In Section 4 Complete and Generalized Synchronization with some examples are presented and finally, in Section 5, some conclusions are drawn.

2. The Discrete-Time Fuzzy Output Regulation Problem

Consider a nonlinear discrete-time system defined by

$$x_{k+1} = f(x_k, w_k, u_k), \quad (1)$$

$$y_k = c(x_k), \quad (2)$$

$$w_{k+1} = s(w_k), \quad (3)$$

$$y_{\text{ref},k} = q(w_k), \quad (4)$$

$$e_k = h(x_k, w_k), \quad (5)$$

where $x_k \in \mathbb{R}^n$ is the state vector of the plant, $w_k \in W \subset \mathbb{R}^s$ is the state vector of the exosystem, which generates the reference and/or the perturbation signals, and $u_k \in \mathbb{R}^m$ is the input signal. Equation (5) refers to difference between output system of the plant ($y_k \in \mathbb{R}^m$) and the reference signal ($y_{\text{ref},k} \in \mathbb{R}^m$); that is, $h(x_k, w_k) = y_k - y_{\text{ref},k} = c(x_k) - q(w_k)$ and take into account that $m \leq n$. Besides, it is assumed that $f(x_k, u_k, w_k)$, $h(x_k, w_k)$, and $s(w_k)$ are analytical functions and also that $f(0, 0, 0) = 0$, $s(0) = 0$, and $h(0, 0) = 0$ [22].

Clearly, by linearizing (1)–(5) around $x_k = 0$, one gets

$$\begin{aligned} x_{k+1} &= Ax_k + Bu_k + Pw_k, \\ y_k &= Cx_k, \\ w_{k+1} &= Sw_k, \\ y_{\text{ref},k} &= Qw_k, \\ e_k &= Cx_k - Qw_k. \end{aligned} \quad (6)$$

Thus, the Nonlinear Regulator Problem [19, 23] consists of finding a controller $u_k = \alpha(x_k, w_k)$, such that the closed-loop system $x_{k+1} = Ax_k + B\alpha(x_k, 0)$ has an asymptotically stable equilibrium point, and the solution of system (6) satisfies $\lim_{k \rightarrow \infty} e_k = 0$.

So, by defining $\pi(w_k)$ as the steady-state zero error manifold and $\gamma(w_k)$ as the steady-state input, the following theorem gives the conditions for the solution of nonlinear regulation problem.

Theorem 1. *Suppose that $w_{k+1} = s(w_k)$ is Poisson stable and there exists a gain K such that the matrix $A + BK$ is stable and*

there exist mappings $x_{ss}(t) = \pi(w_k)$ and $u_{ss} = \gamma(w_k)$ with $\pi(0) = 0$ and $\gamma(0) = 0$ satisfying

$$\begin{aligned}\pi(s(w_k)) &= f(\pi(w_k), w_k, \gamma(w_k)), \\ 0 &= h(\pi(w_k), w_k).\end{aligned}\quad (7)$$

Then the control signal for the nonlinear regulation is given by

$$u_k = K(x_k - \pi(w_k)) + \gamma(w_k). \quad (8)$$

Proof. See [22–24]. \square

The equation set (7) is known as Discrete-Time Francis-Isidori-Byrnes (DTFIB) equations and linear counterpart is obtained when the mappings $x_{ss,k} = \pi(w_k)$ and $u_{ss,k} = \gamma(w_k)$ transform into $x_{ss,k} = \Pi w_k$ and $u_{ss,k} = \Gamma w_k$, respectively. Thus, the problem is reduced to solve linear matrix equations [25] given by

$$\begin{aligned}\Pi S &= A\Pi + B\Gamma + P, \\ 0 &= C\Pi - Q.\end{aligned}\quad (9)$$

2.1. The Discrete-Time Output Fuzzy Regulation Problem. Takagi and Sugeno proposed a fuzzy model composed of a set of linear subsystems with IF-THEN rules capable of relating physical knowledge, linguistic characteristics, and properties of the system. Such a model successfully represents a nonlinear system at least in a predefined region of phase space [15]. The T-S model for the plant and exosystem can be described as follows [26]:

Plant Model

Rule i :

IF $z_{1,1,k}$ is $M_{1,1}^i$ and ... and $z_{1,p_1,k}$ is M_{1,p_1}^i ,

$$\text{THEN} \begin{cases} x_{k+1} = A_i x_k + B_i u_k + P_i w_k, \\ y_k = C_i x_k, \end{cases} \quad (10)$$

$$i = 1, 2, \dots, r_1,$$

where r_1 is the number of rules in the model of the plant and the sets $M_{1,j}^i$ are the fuzzy sets defined based on the previous dynamic knowledge of the system.

Exosystem Model

Rule i :

IF $z_{2,1,k}$ is $M_{2,1}^i$ and ... and $z_{2,p_2,k}$ is M_{2,p_2}^i ,

$$\text{THEN} \begin{cases} w_{k+1} = S_i w_k, \\ y_{\text{ref},k} = Q_i w_k, \end{cases} \quad i = 1, 2, \dots, r_2, \quad (11)$$

where r_2 is the number of rules in the model of the exosystem and $M_{2,j}^i$ are the fuzzy sets.

Then, the regulation problem defined by (1)–(5) can be represented through the T-S discrete-time fuzzy model; that is, [20]

$$\begin{aligned}x_{k+1} &= \sum_{i=1}^{r_1} h_{1,i}(z_{1,k}) \{A_i x_k + B_i u_k + P_i w_k\}, \\ w_{k+1} &= \sum_{i=1}^{r_2} h_{2,i}(z_{2,k}) S_i w_k, \\ e_k &= \sum_{i=1}^{r_1} h_{1,i}(z_{1,k}) C_i x_k - \sum_{i=1}^{r_2} h_{2,i}(z_{2,k}) Q_i w_k,\end{aligned}\quad (12)$$

where $x_k \in \mathbb{R}^n$ is the state vector of the plant, $w_k \in \mathbb{R}^s$ is the state vector of the exosystem, $u_k \in \mathbb{R}^m$ is the input signal, $e_k \in \mathbb{R}^m$, and $h_{*,i}(z)$ is the normalized weight of each rule, 1 for the plant and 2 for the exosystem, which depends on the membership function for the premise variable $z_{*,k}$ in $M_{*,j}^i$; that is,

$$\begin{aligned}\bar{w}_{*,i}(z_{*,k}) &= \prod_{j=1}^{p_*} M_{*,j}^i(z_{*,j,k}), \\ h_{*,i}(z_{*,k}) &= \frac{\bar{w}_{*,i}(z_{*,k})}{\sum_{i=1}^{r_*} \bar{w}_{*,i}(z_{*,k})},\end{aligned}\quad (13)$$

$$\sum_{i=1}^{r_*} h_{*,i}(z_{*,k}) = 1,$$

$$h_{*,i}(z_{*,k}) \geq 0$$

with $z_{*,k} = [z_{*,1,k} \ z_{*,2,k} \ \dots \ z_{*,p_*,k}]$ as a function of x_k and/or w_k , $i = 1, \dots, r_*$ and $j = 1, \dots, p_*$.

The *discrete-time fuzzy output regulation problem* consists of finding a controller $u_k = \alpha(x_k, w_k)$, such that the closed-loop system with no external signal

$$x_{k+1} = \sum_{i=1}^{r_1} h_{1,i}(z_{1,k}) \{A_i x_k + B_i \alpha(x_k, 0)\} \quad (14)$$

has an asymptotically stable equilibrium point.

The solution of system (12) satisfies

$$\lim_{k \rightarrow \infty} e_k = 0. \quad (15)$$

In order to achieve the synchronization of chaotic systems described by a T-S discrete-time fuzzy model it is necessary to fulfill (7) [27, 28]. Then

$$\begin{aligned}\pi(s(w_k)) &= \sum_{i=1}^{r_1} h_{1,i}(z_{1,k}) \{A_i \pi(w_k) + B_i \gamma(w_k) + P_i w_k\},\end{aligned}\quad (16)$$

$$0 = \sum_{i=1}^{r_1} h_{1,i}(z_{1,k}) C_i \pi(w_k) - \sum_{i=1}^{r_2} h_{2,i}(z_{2,k}) Q_i w_k, \quad (17)$$

where $\pi(w_k)$ is the zero error steady-state manifold which becomes invariant by the effect of the steady-state input $\gamma(w_k)$.

Assuming the mappings $\pi(w_k)$ and $\gamma(w_k)$ as

$$\tilde{\pi}(w_k) = \sum_{i=1}^{r_1} h_{1,i}(z_{1,k}) \sum_{j=1}^{r_2} h_{2,i}(z_{2,k}) \Pi_{i,j} w_k, \quad (18)$$

$$\tilde{\gamma}(w_k) = \sum_{i=1}^{r_1} h_{1,i}(z_{1,k}) \sum_{j=1}^{r_2} h_{2,i}(z_{2,k}) \Gamma_{i,j} w_k, \quad (19)$$

respectively, with $\Pi_{i,j}$ and $\Gamma_{i,j}$ as a solution of $r_1 \cdot r_2$ lineal local problems,

$$\Pi_{ij} S_j = A_i \Pi_{ij} + B_i \Gamma_{ij} + P_i, \quad (20)$$

$$0 = C_i \Pi_{i,j} - Q_i, \quad (21)$$

for all $i = 1, \dots, r_1$ and $j = 1, \dots, r_2$, the following control law can be obtained [20, 22, 23]:

$$u_k = \sum_{h=1}^{r_1} h_{1,i}(z_{1,k}) \cdot K_i \left[x_k - \sum_{i=1}^{r_1} h_{1,i}(z_{1,k}) \sum_{j=1}^{r_2} h_{2,j}(z_{2,k}) \Pi_{ij} w_k \right] + \sum_{i=1}^{r_1} h_{1,i}(z_{1,k}) \sum_{j=1}^{r_2} h_{2,j}(z_{2,k}) \Gamma_{ij} w_k. \quad (22)$$

However, by substitution of $\pi(w_k)$ and $\gamma(w_k)$ in (16) and (17) and considering

(1) the steady-state zero error manifold $\pi(w_k) = \Pi w_k$, that is, $\Pi_{ij} = \Pi$,

(2) the input matrices $B_i = B$ and/or $\Gamma_{ij} = \Gamma$,

for all $i = 1, \dots, r_1$ and $j = 1, \dots, r_2$, the following control signal u_k emerges:

$$u_k = \sum_{h=1}^{r_1} h_{1,i}(z_{1,k}) K_i [x_k - \Pi w_k] + \sum_{i=1}^{r_1} h_{1,i}(z_{1,k}) \sum_{j=1}^{r_2} h_{2,j}(z_{2,k}) \Gamma_{ij} w_k. \quad (23)$$

On the other hand, the existence of a fuzzy stabilizer of the form $u = \sum_{i=1}^{r_1} h_{1,i}(z_{1,k}) K_i x_k$, ensuring that the tracking error converges asymptotically to zero, can be obtained from the Parallel Distributed Compensator (PDC) [29, 30] or another stability analysis for T-S fuzzy models such as [31].

Remark 2. The control signal in (22) is given by the substitution of (18) and (19) in (16) and (17); the proposed controller provides the following advantages:

(1) All parameters included in the controller are known; this includes the membership functions of the plant and exosystem, which are well defined in the T-S fuzzy model.

On the other hand, Π_{ij} and Γ_{ij} come directly from solving of $r_1 \cdot r_2$ local linear problems equivalent to solving the Francis equations; such problems can be easily solved by using programs like Matlab or Mathematica.

(2) In the case when the $r_1 \cdot r_2$ local linear problems lead to $\Pi_{ij} \neq \Pi$, then at least a bounded error is ensured.

(3) It is clear that when $\Pi_{ij} = \Pi$, the term $\sum_{i=1}^{r_1} h_{1,i}(z_{1,k}) \sum_{j=1}^{r_2} h_{2,j}(z_{2,k}) \Pi_{ij} w_k$ changes to Πw_k leading to controller defined in (23).

(4) The following condition: the input matrices $B_i = B$ and/or $\Gamma_{ij} = \Gamma$ avoids the crossed terms in the solutions of (16) and (17), allowing the exact fuzzy output regulation.

(5) The proposed controller can be seen as a simple substitution of the aforementioned elements.

On the other hand, the following disadvantages can appear:

(1) If the condition is that the steady-state zero error manifold $\pi(w_k) \neq \Pi w_k$, that is, $\Pi_{ij} \neq \Pi$, then, it will be necessary to adjust the local regulator by means of new membership functions. Please refer to Section 3.

(2) As expected, the complexity of the controller increases according to the number of local subsystems.

The following theorem provides the conditions for the existence of the exact fuzzy output regulator for a discrete-time T-S fuzzy models.

Theorem 3. *The exact fuzzy output regulation with full information of systems defined as (12) is solvable if (a) there exists the same zero error steady-state manifold $\pi(w_k) = \Pi$; (b) there exist $u_k = \sum_{i=1}^{r_1} h_{1,i}(z_{1,k}) K_i x_k$ for the fuzzy system; (c) the exosystem $w_{k+1} = s(w_k)$ is Poisson stable, and the input matrices for all subsystems B_i are equal. Moreover, the Exact Output Fuzzy Regulation Problem is solvable by the controller*

$$u_k = \sum_{h=1}^{r_1} h_{1,i}(z_{1,k}) K_i [x_k - \Pi w_k] + \sum_{i=1}^{r_1} h_{1,i}(z_{1,k}) \sum_{j=1}^{r_2} h_{2,j}(z_{2,k}) \Gamma_{ij} w_k. \quad (24)$$

Proof. From the previous analysis, the existence of mappings $\pi(w_k) = \Pi w_k$ and $\gamma(w_k) = \sum_{i=1}^{r_1} h_{1,i}(z_{1,k}) \sum_{j=1}^{r_2} h_{2,j}(z_{2,k}) \Gamma_{ij} w_k$ is guaranteed when the input matrices for all subsystems B_i are equal, and the solution of $r_1 \cdot r_2$ lineal local problems is

$$\Pi_{ij} S_j = A_i \Pi_{ij} + B_i \Gamma_{ij} + P_i, \quad (25)$$

$$0 = C_i \Pi_{i,j} - Q_i,$$

leading to $\Pi_{ij} = \Pi$ for all $i = 1, \dots, r_1$ and $j = 1, \dots, r_2$.

On the other hand, the inclusion of condition (b) has been thoroughly discussed in [22, 23, 27, 28, 31–33], and it implies the existence of a fuzzy stabilizer.

Condition (c) ensures the nonexistence of crossed terms in the local Francis equations. Finally, condition (d) is introduced to avoid the fact that the reference signal converges to zero, which would turn the regulation problem into a simple stability problem. The rest of the proof follows directly from the previous analysis. \square

3. The Output Regulator by means of Local Regulators and Tuning of New Membership Functions

In this section, a discrete-time T-S fuzzy model is considered to solve the exact output regulation on the basis of linear local controllers. So, the main goal is to find a complete regulator based on the fuzzy summation of local regulators using adequate membership functions, such that the result given in Theorem 3 can be relaxed [34]. These membership functions are not necessarily the same included in the fuzzy plant, as is described in (19). Thus, the steady-state input $\gamma(w_k)$ can be defined as

$$\gamma(w_k) = \sum_{i=1}^{r_1} \mu_{1,i}(z_{1,k}) \sum_{j=1}^{r_2} \mu_{2,j}(z_{2,k}) \Gamma_{ij} w_k, \quad (26)$$

where $\mu_{1,i}(z_{1,k})$ and $\mu_{2,j}(z_{2,k})$ are new membership functions, such that the fuzzy output regulator obtained from local regulators provides the exact fuzzy output regulation. This approach requires the computation of the linear local controllers and the computation of the new membership functions. In this work, such functions are represented by the following expression:

$$\mu_{1,i}(x_k) = \frac{1}{1 + |(x_k - c_i)/a_i|^{2b_i}}, \quad \forall i = 1, \dots, r_1, \quad (27)$$

$$\mu_{2,j}(w_k) = \frac{1}{1 + |(w_k - c_j)/a_j|^{2b_j}}, \quad \forall j = 1, \dots, r_2. \quad (28)$$

$\mu_{1,i}$ and $\mu_{2,j}$ are well known as *generalized bell-shaped membership functions* and the parameters a_i , a_j , b_i , b_j , c_i , and c_j determine the form, center, and amplitude, respectively. Therefore, from (22), the input can be defined by

$$u_k = \sum_{i=1}^{r_1} h_{1,i}(x_k) k_i \{x_k - \pi(w_k)\} + \sum_{i=1}^{r_1} \mu_{1,i}(w_k) \sum_{j=1}^{r_2} \mu_{2,j}(w_k) \Gamma_{ij} w_k, \quad (29)$$

because $z_{*,k}$ is a function of x_k and in steady-state $x_k = \pi(w_k)$.

Then, for tuning the membership functions (27) and (28), the parameters a_i and a_j will be optimized by means of genetic algorithms, ensuring the correct interpolation between the local linear regulators. The foregoing can be summarized in the control scheme depicted in Figure 1.

To this end, the following algorithm should be carried out.

Algorithm 4. Main steps to take into account to solve the tuning membership functions problem are as follows [35]:

- (1) Start with a randomly generated population of n -bit chromosomes (candidate solutions to a problem). The traditional representation is binary as in Figure 2. A binary string is called "chromosome." Each position therein is called "gene" and the value in this position is named "allele."
- (2) Calculate the fitness $f(x)$ of each chromosome x in the population.
- (3) Repeat the following steps until n offspring have been created:
 - (i) Select a pair of parent chromosomes from the current population, the probability of selection being an increasing function of fitness. Selection is done "with replacement," meaning that the same chromosome can be selected more than once to become a parent.
 - (ii) With probability p_c (the "crossover probability" or "crossover rate"), crossover the pair at a randomly chosen point (chosen with uniform probability) to form two offspring. If no crossover takes place, form two offspring that are exact copies of their respective parents. (Note that here the crossover rate is defined to be the probability that two parents will cross over in a single point.)
 - (iii) Mutate the two offspring at each locus with probability p_m (the mutation probability or mutation rate), and place the resulting chromosomes in the new population.

- (4) Replace the current population with the new population.
- (5) Go to step (2).

This process ends when the fitness function reaches a value less than or equal to a predefined bound, or when the maximum number of iterations is reached. Each iteration of this process is called a generation.

In order to apply a genetic algorithm it requires the following five basic components:

- (i) A representation of the potential solutions to the problem.
- (ii) One way to create an initial population of possible solutions (typically a random process).
- (iii) An evaluation function to play the role of the environment, classifying the solutions in terms of its "fitness."
- (iv) Genetic operators that alter the composition of the chromosomes that will be produced for generations.
- (v) Values for the various parameters used by the genetic algorithm (population size, crossover probability, mutation probability, maximum number of generations, etc.).

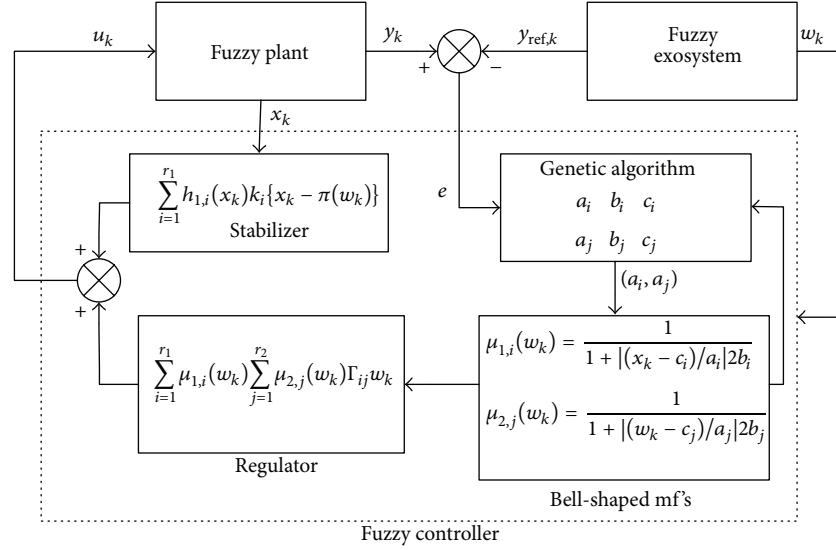


FIGURE 1: Control scheme for fuzzy output regulation and genetic algorithms.

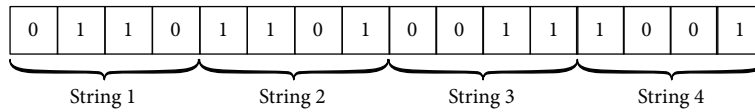


FIGURE 2: Binary string commonly used in genetic algorithms.

The aforementioned can be summarized by the flowchart depicted in Figure 3.

4. Synchronization of T-S Discrete-Time Fuzzy Systems

The stabilization and synchronization of chaotic systems are two of the most challenging and stimulating problems due to their capabilities of describing a great variety of very interesting phenomena in physics, biology, chemistry, and engineering, to name a few. In this section, the regulation theory is used to synchronize chaotic systems described by T-S discrete-time fuzzy models. Both the drive system and the response system are modeled by the same attractor (Rössler's equation) with the difference that response system can be influenced by an input signal. This type of synchronization is known as *Complete Synchronization* (CS) [36].

Considering two Rössler chaotic oscillators as $\dot{w} = f(w)$ (drive system) and $\dot{x} = f(x, w, u)$ (response system), the ordinary differential equations of these systems are

$$\begin{aligned}\dot{w}_1 &= -(w_2 + w_3), \\ \dot{w}_2 &= w_1 + aw_2, \\ \dot{w}_3 &= bw_1 - (c - w_1)w_3,\end{aligned}$$

Rössler attractor

Drive system,

$$\dot{x}_1 = -(x_2 + x_3),$$

$$\dot{x}_2 = x_1 + ax_2,$$

$$\dot{x}_3 = bx_1 - (c - x_1)x_3 + u,$$

Rössler attractor

Response system.

(30)

According to [15], these systems can be exactly represented by means of the following two-rule T-S fuzzy models when a, b and c are constants and $x_1 \in [c - d, c + d]$ with $d > 0$. Thus

$$\dot{x}(t) = \sum_{i=1}^2 h_{1,i}(x_1(t)) \{ \tilde{A}_i x(t) + \tilde{B}_i u(t) \},$$

$$\dot{w}(t) = \sum_{i=1}^2 h_{2,i}(w_1(t)) \tilde{S}_i w(t),$$

$$e(t) = \sum_{i=1}^2 h_{1,i}(x_1(t)) \tilde{C}_i x(t)$$

$$- \sum_{i=1}^2 h_{2,i}(w_1(t)) \tilde{Q}_i w(t),$$

(31)

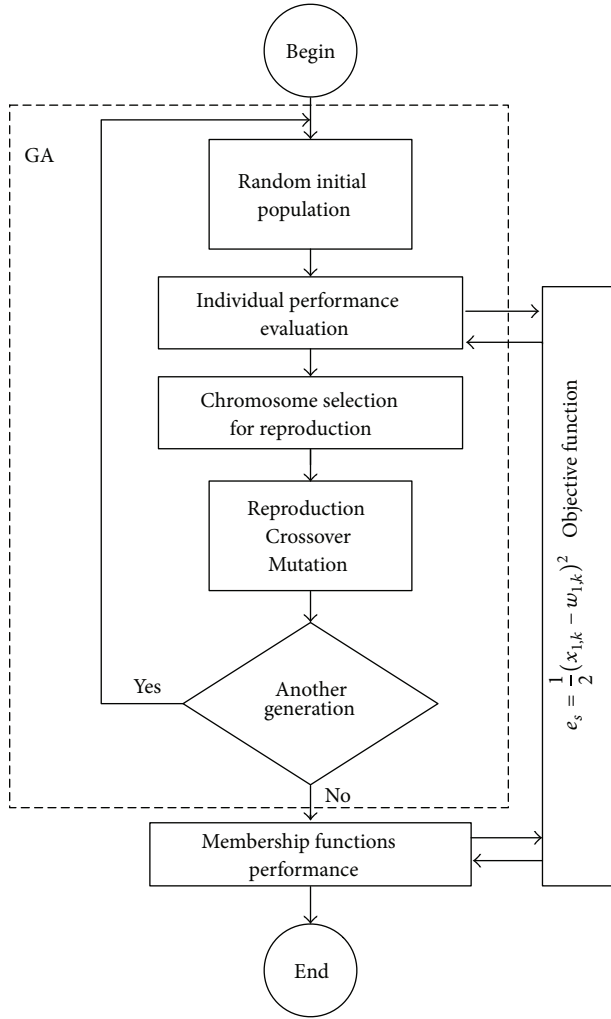


FIGURE 3: Flowchart of genetic algorithms.

where

$$\begin{aligned}
 \tilde{A}_1 &= \begin{bmatrix} 0 & -1 & -1 \\ 1 & a & 0 \\ b & 0 & -d \end{bmatrix}, \\
 \tilde{S}_1 &= \begin{bmatrix} 0 & -1 & -1 \\ 1 & a_w & 0 \\ b_w & 0 & -d_w \end{bmatrix}, \\
 \tilde{A}_2 &= \begin{bmatrix} 0 & -1 & -1 \\ 1 & a & 0 \\ b & 0 & d \end{bmatrix}, \\
 \tilde{S}_2 &= \begin{bmatrix} 0 & -1 & -1 \\ 1 & a_w & 0 \\ b_w & 0 & d_w \end{bmatrix}, \\
 \tilde{B}_1 &= \tilde{B}_2 = [0 \ 0 \ 1]^T, \\
 \tilde{C}_i &= \tilde{Q}_i = [1 \ 0 \ 0].
 \end{aligned} \tag{32}$$

Besides, the membership functions for such systems are

$$\begin{aligned}
 h_{1,1}(x_1) &= \frac{1}{2} \left(1 + \frac{c - x_1}{d} \right), \\
 h_{1,2}(x_1) &= \frac{1}{2} \left(1 - \frac{c - x_1}{d} \right), \\
 h_{2,1}(w_1) &= \frac{1}{2} \left(1 + \frac{c - w_1}{d} \right), \\
 h_{2,2}(w_1) &= \frac{1}{2} \left(1 - \frac{c - w_1}{d} \right).
 \end{aligned} \tag{33}$$

Now, the continuous-time T-S fuzzy model can be converted to the following discrete counterpart by using [1]

$$\begin{aligned}
 G_i &= \exp(A_i T_s) = I + A_i T_s + A_i^2 \frac{T_s^2}{2!} + \dots, \\
 H_i &= \int_0^{T_s} \exp(A_i \tau) B_i d\tau = (G_i - I) A_i^{-1} B_i.
 \end{aligned} \tag{34}$$

Therefore, the discrete-time T-S fuzzy model is given by

$$\begin{aligned}
 x_{k+1} &= \sum_{i=1}^2 h_{1,i}(x_{1,k}) \{A_i x_k + B_i u_k + P_i w_k\}, \\
 w_{k+1} &= \sum_{i=1}^2 h_{2,i}(w_{1,k}) S_i w_k, \\
 e_k &= \sum_{i=1}^2 h_{1,i}(x_{1,k}) C_i x_k - \sum_{i=1}^2 h_{2,i}(w_{1,k}) Q_i w_k,
 \end{aligned} \tag{35}$$

where

$$\begin{aligned}
 A_1 &= \begin{bmatrix} 1 & -T_s & -T_s \\ T_s & aT_s + 1 & 0 \\ bT_s & 0 & 1 - dT_s \end{bmatrix}, \\
 S_1 &= \begin{bmatrix} 1 & -T_s & -T_s \\ T_s & a_w T_s + 1 & 0 \\ b_w T_s & 0 & 1 - d_w T_s \end{bmatrix}, \\
 A_2 &= \begin{bmatrix} 1 & -T_s & -T_s \\ T_s & aT_s + 1 & 0 \\ bT_s & 0 & T_s d + 1 \end{bmatrix}, \\
 S_2 &= \begin{bmatrix} 1 & -T_s & -T_s \\ T_s & a_w T_s + 1 & 0 \\ b_w T_s & 0 & d_w T_s + 1 \end{bmatrix}, \\
 B_1 &= B_2 = [0 \ 0 \ T_s]^T, \\
 C_i &= Q_i = [1 \ 0 \ 0].
 \end{aligned} \tag{36}$$

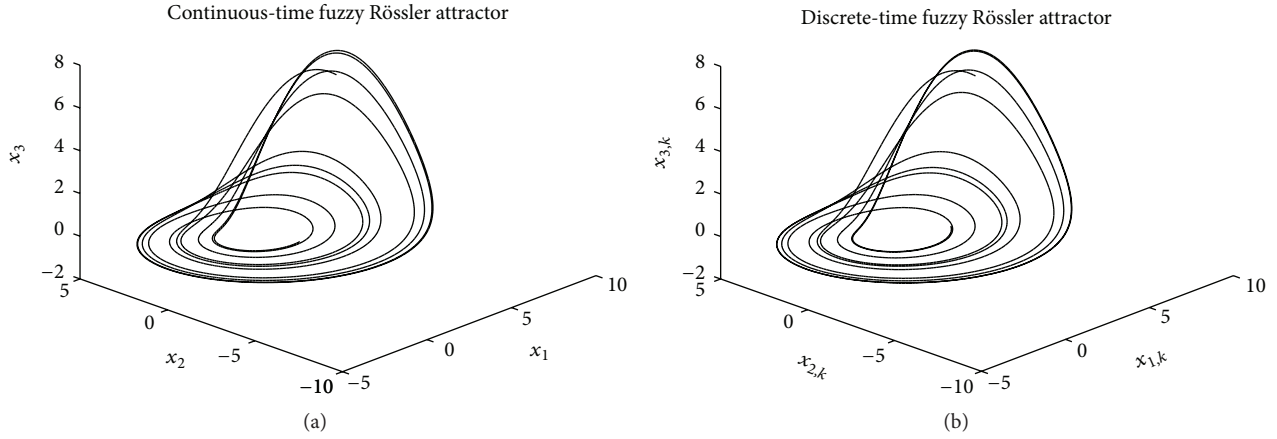


FIGURE 4: (a) Continuous T-S fuzzy model for a Rössler attractor and (b) discrete T-S fuzzy model for a Rössler attractor.

Besides, the membership functions for the T-S discrete-time fuzzy model are

$$\begin{aligned}
 h_{1,1}(x_{1,k}) &= \frac{1}{2} \left(1 + \frac{c - x_{1,k}}{d} \right), \\
 h_{1,2}(x_{1,k}) &= \frac{1}{2} \left(1 + \frac{c - x_{1,k}}{d} \right), \\
 h_{1,2}(w_{1,k}) &= \frac{1}{2} \left(1 + \frac{c_w - w_{1,k}}{d_w} \right), \\
 h_{2,2}(w_{1,k}) &= \frac{1}{2} \left(1 + \frac{c_w - w_{1,k}}{d_w} \right),
 \end{aligned} \tag{37}$$

with $a = 0.34$, $b = 0.4$, $c = 4.5$, $d = 10$, $a_w = 0.34$, $b_w = 0.4$, $c_w = 4.5$, $d_w = 10$, and T_s as the sampling time.

Figure 4(b) shows the trajectory of the discrete-time version of the continuous-time T-S fuzzy Rössler model, with $T_s = 0.00357$. It can be seen that the overall shape of the trajectory is similar to that in Figure 4(a).

Then, by using the approach derived in this work and from (20) and (17), the zero error steady-state manifold $\pi(w_k) = \Pi$ is

$$\Pi = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{38}$$

Γ_{ij} are

$$\begin{aligned}
 \Gamma_{1,1} &= [0 \ 0 \ 0], \\
 \Gamma_{1,2} &= [0 \ 0 \ 20], \\
 \Gamma_{2,1} &= [0 \ 0 \ -20], \\
 \Gamma_{2,2} &= [0 \ 0 \ 0].
 \end{aligned} \tag{39}$$

On the other hand, the fuzzy stabilizer for this system is computed by means of *Ackermann's formula*, and by locating the eigenvalues at

$$(1) \text{ subsystem } [0.9980+0.0062i \ 0.9980-0.0062i \ 0.9982],$$

$$(2) \text{ subsystem } [0.9980+0.0062i \ 0.9980-0.0062i \ 0.9982],$$

the following gains are obtained:

$$\begin{aligned}
 K_1 &= [3.1629 \ 0.9257 \ 8.0354], \\
 K_2 &= [3.1629 \ 0.9257 \ -11.9646].
 \end{aligned} \tag{40}$$

Remark 5. It is important to verify that the fuzzy feedback stabilizer is valid for the overall T-S fuzzy model, by checking that the eigenvalues of the interpolation regions are inside the unit circle also [31].

Then, by setting the initial conditions as $x_{1,k} = 5$, $x_{2,k} = 0$, $x_{3,k} = 6$, $w_{1,k} = 1$, $w_{2,k} = 0$, and $w_{1,k} = 0$ and by applying the controller (23), the results depicted in Figures 5 and 6 are obtained.

The tracking for the drive states and response states is drawn in Figure 7.

4.1. Generalized Synchronization. The discrete-time fuzzy synchronization problem is solvable when the conditions of Theorem 3 are fulfilled. However, when two chaotic systems are different, fulfilling these conditions is not so common. This is because the local regulators have different zero error steady-state manifolds, in general [20]. From the regulation point of view, the problem can be seen as finding, if it is possible, a transformation $\pi : w_k \rightarrow x_k$ regarding mapping the trajectories of the drive system into the ones of the response systems; that is, $x_k = \pi(w_k)$; this is known as Generalized Synchronization [37, 38] and satisfies

$$\lim_{k \rightarrow \infty} \|x_k(x(0)) - w_k(w(0))\| = 0, \tag{41}$$

with $x(0)$ and $w(0)$ as initial conditions.

Now, consider the chaotic drive system as $\dot{w} = f(w)$ and $\dot{x} = f(x, w, u)$ as the response system.

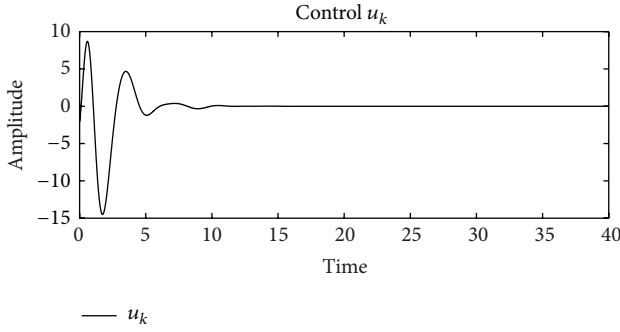


FIGURE 5: Control signal for the Complete Synchronization of two discrete-time T-S fuzzy Rössler systems.

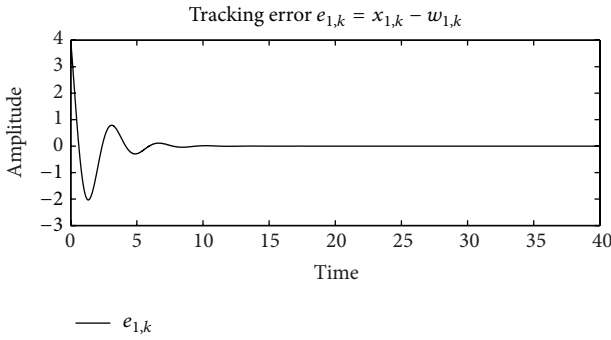


FIGURE 6: Tracking error signal for the Complete Synchronization of two discrete-time T-S fuzzy Rössler systems.

The equations for the aforementioned systems are described as follows:

$$\begin{aligned}
 \dot{w}_1 &= \alpha(w_2 - w_1), \\
 \dot{w}_2 &= rw_1 - w_2 - w_1w_3, \\
 \dot{w}_3 &= w_1w_2 - \beta w_3, \\
 \text{Lorenz Attractor} \\
 \text{Drive system} \\
 \dot{x}_1 &= a(x_2 - x_1), \\
 \dot{x}_2 &= (c - a)x_1 - x_1x_3 + cx_2 + u, \\
 \dot{x}_3 &= x_1x_2 - bx_3, \\
 \text{Chen's attractor} \\
 \text{Response system}
 \end{aligned} \tag{42}$$

As before these systems can be exactly represented by means of the following two-rule continuous-time T-S fuzzy models when $x_1 \in [X_{\min}, X_{\max}]$ and $w_1 \in [M_1, M_2]$. Therefore,

$$\begin{aligned}
 \dot{x}(t) &= \sum_{i=1}^2 h_{1,i}(x_1(t)) \{ \bar{A}_i x(t) + \bar{B}_i u(t) \}, \\
 \dot{w}(t) &= \sum_{i=1}^2 h_{2,i}(w_1(t)) \tilde{S}_i w(t),
 \end{aligned}$$

$$\begin{aligned}
 e(t) &= \sum_{i=1}^2 h_{1,i}(x_1(t)) \bar{C}_i x(t) \\
 &\quad - \sum_{i=1}^2 h_{2,i}(w_1(t)) \bar{Q}_i w(t),
 \end{aligned} \tag{43}$$

where

$$\begin{aligned}
 \bar{A}_1 &= \begin{bmatrix} -a & a & 0 \\ c - a & c & -X_{\min} \\ 0 & X_{\min} & -b \end{bmatrix}, \\
 \tilde{S}_1 &= \begin{bmatrix} -\alpha & \alpha & 0 \\ r & -1 & -M_1 \\ 0 & M_1 & -\beta \end{bmatrix}, \\
 \bar{A}_2 &= \begin{bmatrix} -a & a & 0 \\ c - a & c & -X_{\max} \\ 0 & X_{\max} & -b \end{bmatrix}, \\
 \tilde{S}_2 &= \begin{bmatrix} -\alpha & \alpha & 0 \\ r & -1 & -M_2 \\ 0 & M_2 & -\beta \end{bmatrix}, \\
 \bar{B}_1 = \bar{B}_2 &= [0 \ 1 \ 0]^T, \\
 \bar{C}_i = \bar{Q}_i &= [1 \ 0 \ 0].
 \end{aligned} \tag{44}$$

Then, by using (34), the discrete counterpart is obtained:

$$\begin{aligned}
 x_{k+1} &= \sum_{i=1}^2 h_{1,i}(x_{1,k}) \{ A_i x_k + B_i u_k + P_i w_k \}, \\
 w_{k+1} &= \sum_{i=1}^2 h_{2,i}(w_{1,k}) S_i w_k, \\
 e_k &= \sum_{i=1}^2 h_{1,i}(x_{1,k}) C_i x_k - \sum_{i=1}^2 h_{2,i}(w_{1,k}) Q_i w_k,
 \end{aligned} \tag{45}$$

where

$$\begin{aligned}
 A_1 &= \begin{bmatrix} 1 - aT_s & aT_s & 0 \\ -(a - c)T_s & cT_s + 1 & -X_{\min}T_s \\ 0 & T_s X_{\min} & 1 - bT_s \end{bmatrix}, \\
 S_1 &= \begin{bmatrix} 1 - a_w T_s & a_w T_s & 0 \\ c_w T_s & 1 - T_s & -M_1 T_s \\ 0 & M_1 T_s & 1 - b_w T_s \end{bmatrix}, \\
 A_2 &= \begin{bmatrix} 1 - aT_s & aT_s & 0 \\ -(a - c)T_s & cT_s + 1 & -X_{\max}T_s \\ 0 & T_s X_{\max} & 1 + bT_s \end{bmatrix}, \\
 S_2 &= \begin{bmatrix} 1 - a_w T_s & a_w T_s & 0 \\ c_w T_s & 1 - T_s & -M_2 T_s \\ 0 & M_2 T_s & 1 - b_w T_s \end{bmatrix},
 \end{aligned}$$

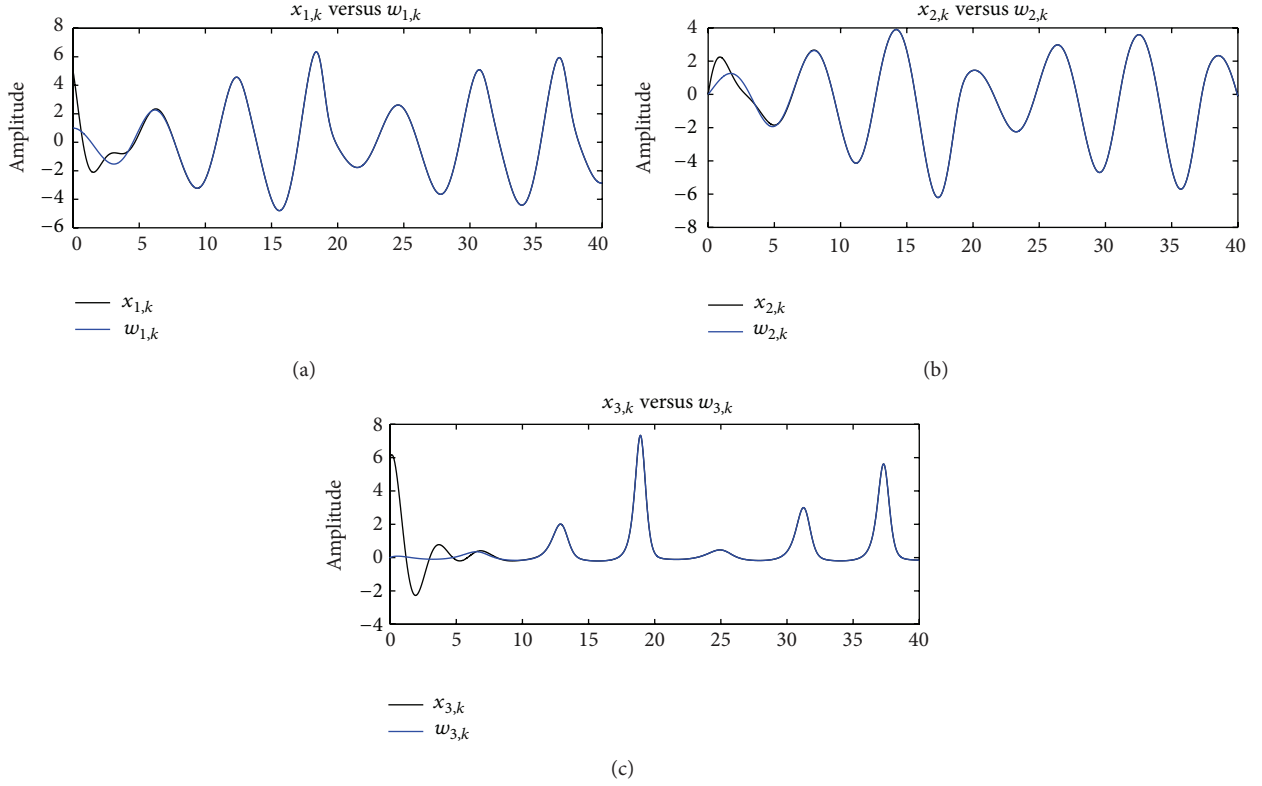


FIGURE 7: States of drive and response tracking for a Complete Synchronization.

$$\begin{aligned} B_1 = B_2 &= [0 \ T_s \ 0]^T, \\ C_i = Q_i &= [1 \ 0 \ 0], \end{aligned} \quad (46)$$

with $a = 35$, $b = 3$, $c = 28$, $a_w = 10$, $b_w = 8/3$, and $c_w = 28$ and T_s as the sampling time. Notice that, in this case, the Generalized Synchronization problem consists of the tracking of w_1 by x_1 . This can be inferred by the form of C_i and Q_i . The membership functions for this system are defined as follows:

Plant Membership Functions

$$\begin{aligned} h_{1,1}(x_{1,k}) &= \frac{-x_{1,k} + X_{\max}}{X_{\max} - X_{\min}}, \\ h_{1,2}(x_{1,k}) &= \frac{x_{1,k} - X_{\min}}{X_{\max} - X_{\min}}. \end{aligned} \quad (47)$$

Exosystem Membership Functions

$$\begin{aligned} h_{2,1}(w_{1,k}) &= \frac{-w_{1,k} + M_1}{M_2 - M_1}, \\ h_{2,2}(w_{1,k}) &= \frac{w_{1,k} - M_2}{M_2 - M_1}. \end{aligned} \quad (48)$$

They are depicted in Figure 8.

Figures 9(a) and 9(b) show the behavior of the two discrete-time T-S fuzzy models, with $x(0) = [1 \ 1 \ 1]^T$ and

$w(0) = [1 \ 0 \ 0]^T$. Then, by using the approach derived in this work and from (20) and (17), the zero error steady-state manifold for each subsystem is

$$\begin{aligned} \Pi_{1,1} &= \begin{bmatrix} 1 & 0 & 0 \\ 0.7143 & 0.2857 & 0 \\ 3.0041 & -0.0143 & 1.2861 \end{bmatrix}, \\ \Pi_{1,2} &= \begin{bmatrix} 1 & 0 & 0 \\ 0.7143 & 0.2857 & 0 \\ 3.0041 & -0.0143 & -1.2861 \end{bmatrix}, \\ \Pi_{2,1} &= \begin{bmatrix} 1 & 0 & 0 \\ 0.7143 & 0.2857 & 0 \\ -3.0041 & 0.0143 & -1.2861 \end{bmatrix}, \\ \Pi_{2,2} &= \begin{bmatrix} 1 & 0 & 0 \\ 0.7143 & 0.2857 & 0 \\ -3.0041 & 0.0143 & 1.2861 \end{bmatrix}. \end{aligned} \quad (49)$$

Γ_{ij} are

$$\begin{aligned} \Gamma_{1,1} &= [-102.2648 \ -0.7142 \ -30.0121], \\ \Gamma_{1,2} &= [-102.2648 \ -0.7142 \ 30.0121], \\ \Gamma_{2,1} &= [-102.2648 \ -0.7142 \ -30.0121], \\ \Gamma_{2,2} &= [-102.2648 \ -0.7142 \ 30.0121]. \end{aligned} \quad (50)$$

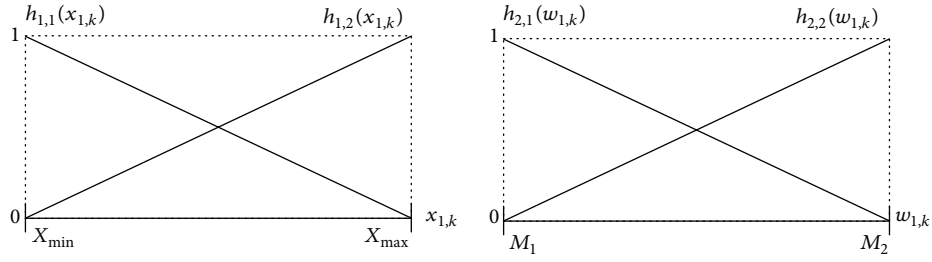


FIGURE 8: Membership functions for Generalized Synchronization.

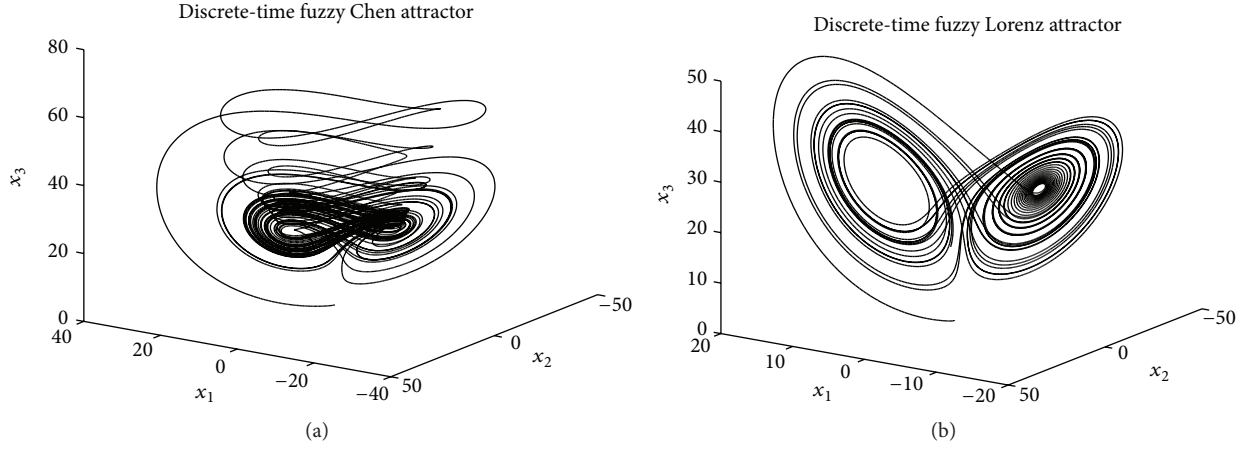


FIGURE 9: (a) Discrete-time T-S fuzzy model for a Chen attractor and (b) discrete T-S fuzzy model for a Lorenz attractor.

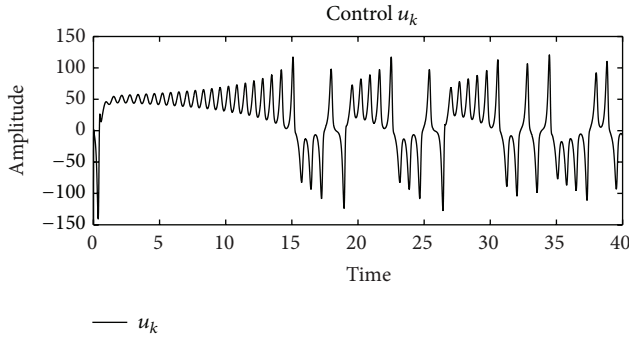


FIGURE 10: Control signal for the Generalized Synchronization of Chen-Lorenz discrete-time T-S fuzzy systems.

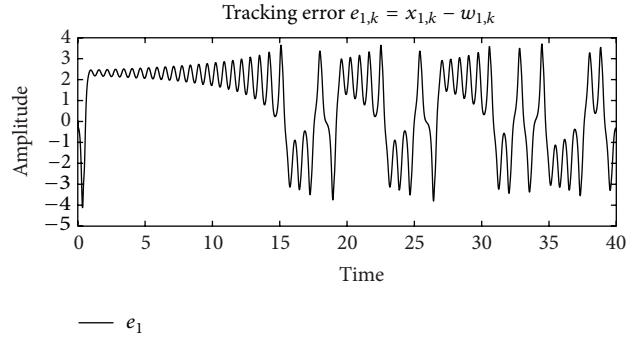


FIGURE 11: Tracking error $e_{1,k}$ signal for the Generalized Synchronization of Chen-Lorenz discrete-time T-S fuzzy systems.

On the other hand, the fuzzy stabilizer for this system is computed by means of *Ackermann's formula*, and by locating the eigenvalues at

- (1) subsystem $[0.9596 \ 0.9955 \ 0.82]$,
- (2) subsystem $[0.9596 \ 0.9955 \ 0.82]$,

the following gains are obtained:

$$\begin{aligned} K_1 &= [-14.3228 \ -214.9000 \ -19.6566], \\ K_2 &= [-14.3228 \ -214.9000 \ 19.6566]. \end{aligned} \tag{51}$$

Notice that the stability is ensured in the fuzzy interpolation region by these gains, for all $t \geq 0$.

As can be readily observed, conditions of Theorem 3 are not fulfilled because the zero error steady-state manifold is not the same for the local regulators. However, the tracking error can be bounded by using the controller defined in (22). Figures 10–12 depicted the behavior of the controller (22) with $x(0) = [1 \ 1 \ 1]^T$ and $w(0) = [1 \ 0 \ 0]^T$ as the initial conditions.

4.2. Generalized Synchronization by Using Genetic Algorithms. The main objective of integrating genetic algorithms is to obtain values of the parameters (a , b , and c) of the new membership functions of the regulator and reducing, in this

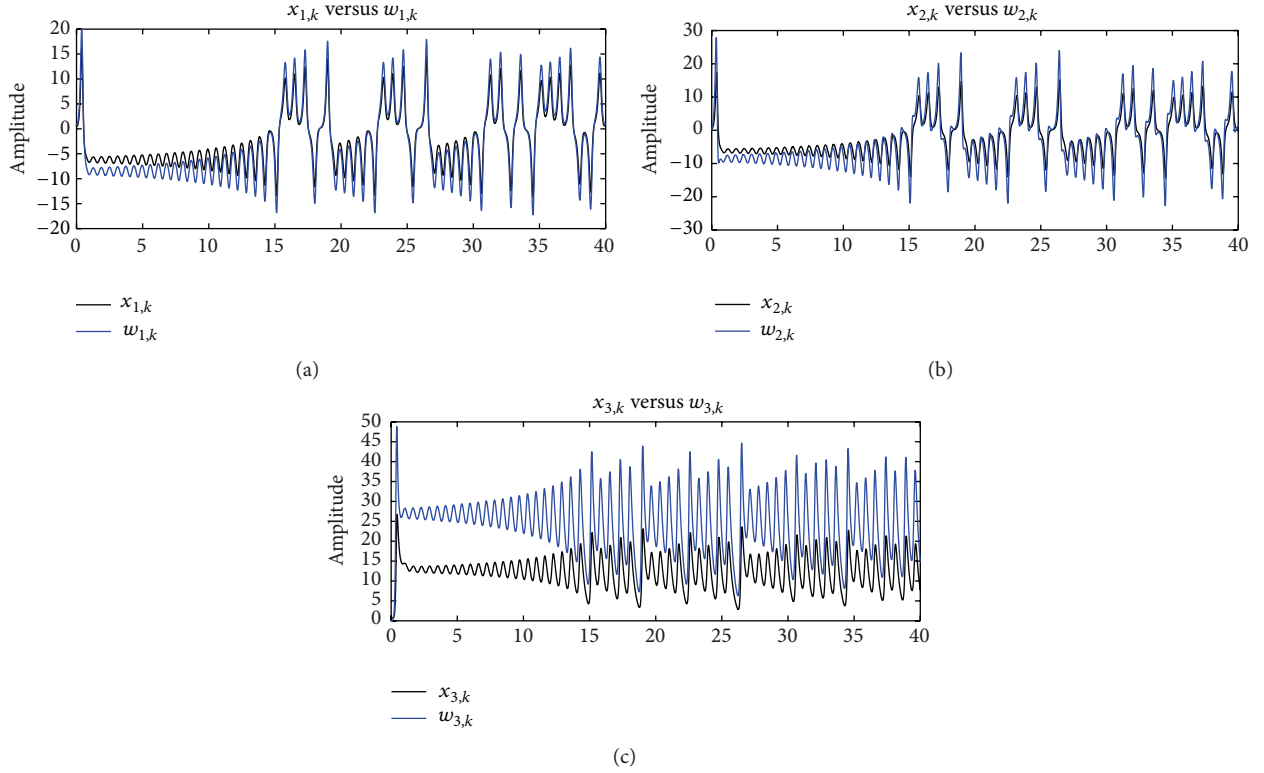


FIGURE 12: States of drive and response system for a Generalized Synchronization.

way, the tracking error. To this end, it is important to consider the following:

- (i) Consider the representation of the potential solutions of the problem; the common representation is binary.
- (ii) An initial population of possible solutions, in a random process, is selected (population size affects the efficiency and performance of GA). For this case, a population size equal to 20 chromosomes is chosen.
- (iii) A fitness function, which indicates how good or bad a certain solution is, is defined. In this case the fitness function is the mean square tracking error, given by the following expression:

$$e_s = \frac{1}{2} (x_{1,k} - w_{1,k})^2. \quad (52)$$

For the representation of possible solutions it is necessary to know the variables to optimize; for this case it has a_i and a_j parameters for $i = 1, 2$ and $j = 1, 2$ defined in (27) and (28); besides, b_i , b_j , c_i , and c_j will be fixed with a constant value; that is, $b_1 = 3$, $c_1 = 30$ for (27) considering $\mu_{1,2} = 1 - \mu_{1,1}$; $b_1 = 3$, $c_1 = 20$ for (28) also considering $\mu_{2,2} = 1 - \mu_{2,1}$. The abovementioned are represented in Figure 13.

It is also important to know the interval value in which the variable a will be operating; for this case, such interval is $x_{1,k} \in [-30 \ 30]$. Thus, the size of each variable, in bits,

and the chromosome length can be computed by using the following expression [39]:

$$S_{\text{var}} = (\text{int } \uparrow) \log_2 (\text{lim}_{\text{upper}} - \text{lim}_{\text{lower}}) 10^{\text{accurate}}, \quad (53)$$

where S_{var} is the size of each variable in bits, the term $(\text{int } \uparrow)$ is the decimal integer value, and $\text{lim}_{\text{upper}} = 30$ and $\text{lim}_{\text{lower}} = -30$ with an accuracy in 3. Therefore, the variable size is equal to 16 bits, but there are two variables, a_1 and a_2 ; thus the chromosomes length is 32 bits. Therefore, the chromosomes are represented in Figure 14.

The initial population is random binary. To assess each individual in the objective function is needed to decode, in this case, a real number. To this end, the following expression is used:

$$\text{Var} = \left[\text{Dec}_{\text{val}} \left(\frac{\text{lim}_{\text{upper}} - \text{lim}_{\text{lower}}}{2^{\text{length}(\text{bit})} - 1} \right) + \text{lim}_{\text{lower}} \right]. \quad (54)$$

Var is a decimal value (phenotype) for each binary chromosome (genotype), Dec_{val} is the decimal value for each binary string, and $\text{length}(\text{bit})$ is equal to 16 bits.

In order to compute the individual aptitude is necessary to introduce the value of each parameter in the bell-shaped membership functions μ_1 and μ_2 . The tracking error function (52) disregards the nonoptimal solution and allows the optimal performance by considering that $x_{1,k}$ tracks $w_{1,k}$. In Table 1 the individual aptitude is depicted.

TABLE 1: Individual aptitude.

Chromosome	Binary string	Decoded integer	Fitness function
1	01011010001000100010111010101101	$f(-7.691493, -16.518837)$	0.971109
2	1100101110111111110101000000111	$f(15.386557, 21.537530)$	6.840302
3	10100011000010001110111101010111	$f(7.116228, 22.616648)$	9.404465
4	01101110111100100011110001011000	$f(-3.463890, -13.742489)$	0.096779
5	0111110110010001000111010000000	$f(-0.247166, 2.945754)$	0.017387
6	0111110110010001000111010000000	$f(-2.282414, -24.456703)$	12.438373
7	01110100110000110000011110011001	$f(-13.451286, 16.387121)$	0.761323
8	00111101110001111101000010101100	$f(4.320043, -25.338247)$	13.686271
9	10010101010001000000001101000010	$f(19.709377, 4.057404)$	0.209690
10	10000010001011110001000101110111	$f(0.443946, -22.452399)$	9.159322
11	0011110010100011101110011100001	$f(-13.682979, 12.366613)$	0.000746
12	00001100011111011100101000101100	$f(-23.463279, 15.066789)$	0.000895
13	10010110000010001001010001111101	$f(4.475563, 4.162142)$	0.017576
14	1011001101011010100100101111010	$f(10.432136, -11.074846)$	0.001021
15	01010011101010101111101011001101	$f(-9.005478, 24.944686)$	13.030471
16	0010001101100110001100000001011	$f(-18.809583, -16.241123)$	0.399280
17	10010001100100001111110100111101	$f(3.567834, 25.439811)$	13.825474
18	1000101100101011010011101110001	$f(2.268925, -9.964782)$	0.003840
19	01100000011011101100000001001011	$f(-6.412421, 13.060105)$	0.039420
20	00001011110001010001010011000101	$f(-23.609277, -21.781125)$	5.130460

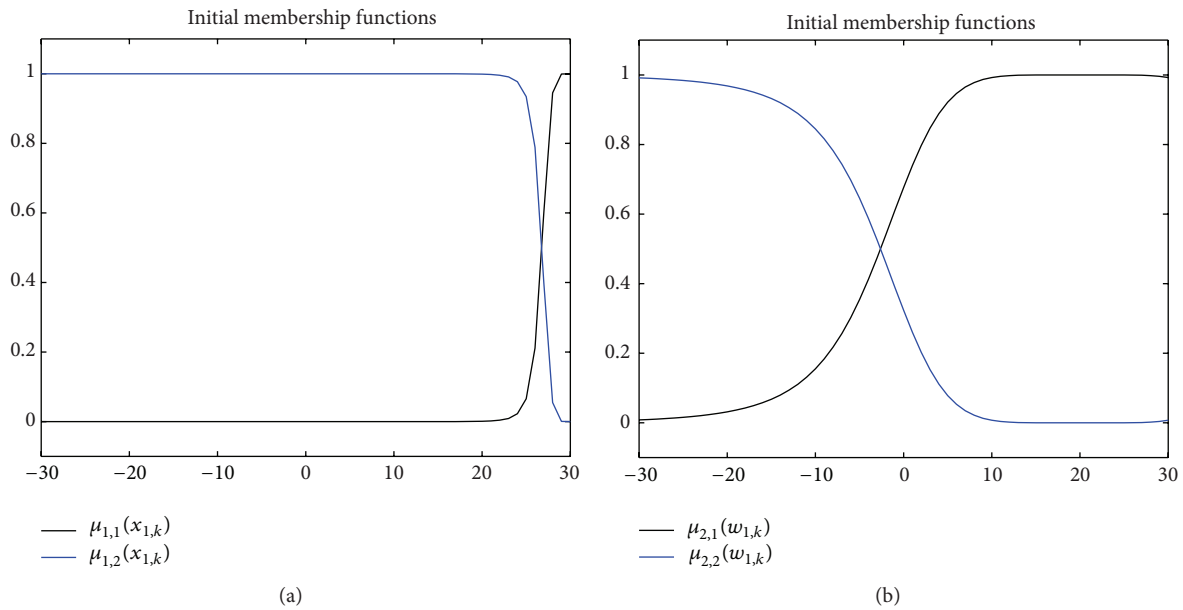


FIGURE 13: Initial membership functions for the proposed fuzzy regulator tuned by GAs.

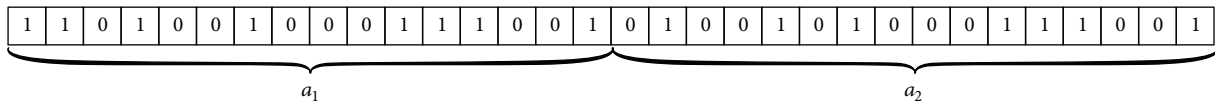


FIGURE 14: Individual binary representation.

Each fitness function value is converted into a set point or fitness value. To this end, the following expression is used:

$$\text{fitness}_{\text{val}} = F(\star) - F_{\min}, \quad (55)$$

where $F(\star)$ is the fitness function value and F_{\min} is the minimum fitness function value.

The two chromosomes with better fitness (elite chromosomes) can live and produce offspring in the next generation. There are many methods to select a pair of chromosomes; the most popular one is named *proportional selection method*:

- (a) Calculate the total fitness value

$$\text{fitness}_T = \sum_{i=1}^{20} \text{fitness}(i). \quad (56)$$

- (b) Compute the probability P_i for each chromosome

$$P_i = \frac{\text{fitness}(i)}{\text{fitness}_T}. \quad (57)$$

- (c) Calculate the cumulative probability for each chromosome

$$Q_k = \sum_{k=0}^i P_k. \quad (58)$$

The total fitness value fitness_T is equal to 86.03420.

The probabilities P_i and Q_k are shown in Table 2.

- (d) Generate a random number r in the range $[0, 1]$.
(e) If $Q_{i-1} < r \leq Q_i$, then select the chromosome to be the one of the parents.
(f) Repeat (d) and (e) to obtain the other parent.

Apply crossover operation on the selected pair, if they have been chosen for crossover (based on probability of crossover $p_c = 1.0$). The most applied crossover operation is single point crossover. Based on the probability of bit mutation $p_m = 0.01$, flip the correspondent bit if selected for mutation. At this point, the process of producing a pair of offspring from two selected parents is finished.

The elite chromosomes of the previous population are not subject to mutation.

In the following simulations the states $x_{1,k}$, control signal u_k , and tracking error $e_{1,k}$ depicted in Figures 16–18, respectively, are obtained after applying (29) and by replacing the new membership functions adjusted by GAs. The final membership functions after tuning the form parameter by means of GA are depicted in Figure 15. It can be readily observed that these new membership functions are different from the original ones (see Figure 13). Notice that, in this example, the tracking error (Figure 18) is less than the error obtained by using the approach discussed in Section 4. This is due to the new membership functions which allow reducing considerably the tracking error. This suggests that the approach presented in this work may be improved by tuning the parameters of center and amplitude in the new membership functions; however this study is not completed yet.

TABLE 2: Individual aptitude.

Chromosome	P_i	Q_k
1	0.011	0.011
2	0.080	0.091
3	0.109	0.200
4	0.001	0.201
5	0.000	0.201
6	0.145	0.346
7	0.009	0.355
8	0.159	0.514
9	0.002	0.516
10	0.106	0.623
11	0.000	0.623
12	0.000	0.623
13	0.000	0.623
14	0.000	0.623
15	0.151	0.775
16	0.005	0.779
17	0.161	0.940
18	0.000	0.940
19	0.000	0.940
20	0.060	1.000

4.3. Pseudofuzzy Generalized Synchronization by Using GAs.

In this section the Generalized Synchronization by using GAs and linear local regulator design will be addressed. However, the crossed terms within the local regulators are arbitrarily removed from the design process. Thus, the steady-state input $\gamma(w_k)$ can be defined as

$$\gamma(w_k) = \sum_{i=1}^{r_1} \mu_{1,i}(z_{1,k}) \Gamma_{ii} w_k, \quad (59)$$

where $\mu_{1,i}(z_{1,k})$ are new membership functions, such that the fuzzy output regulator obtained from local regulators provides the exact fuzzy output regulation. This approach requires the computation of the linear local controllers and the computation of the new membership functions; such functions are represented by (27) and (28). Therefore the final control system is defined by

$$u_k = \sum_{i=1}^{r_1} h_{1,i}(x_k) k_i \{x_k - \pi(w_k)\} + \sum_{i=1}^{r_1} \mu_{1,i}(w_k) \Gamma_{ii} w_k, \quad (60)$$

because $z_{*,k}$ is a function of x_k and in steady-state $x_k = \pi(w_k)$.

For this case the initial membership functions proposed are shown in Figure 19. As it can be seen, these membership functions are not properly fuzzy because they do not fulfill the convex sum; that is, $\sum_{i=1}^{r_1} \mu_i(w_k) \neq 0$. Even so, these functions will be adjusted by GAs in order to ensure the output regulation by using input (60). The following simulation provides a better behavior in the Lorenz-Chen discrete-time fuzzy system by using different membership function and tuning by GAs. Such results are depicted in Figures 20–22.

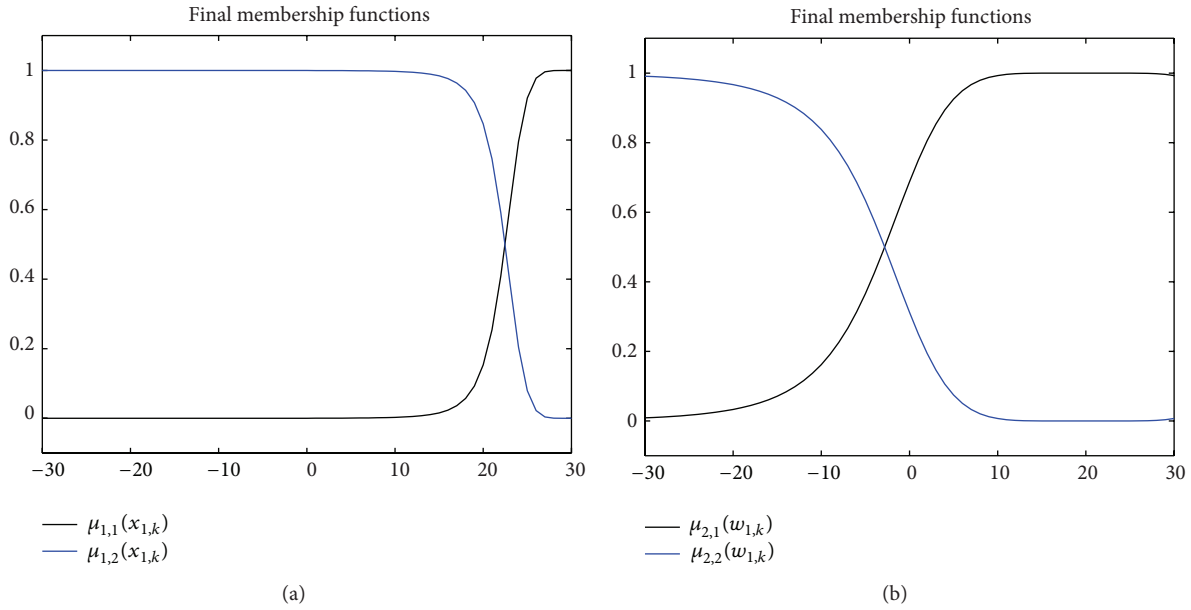


FIGURE 15: Final membership functions for the proposed fuzzy regulator tuned by GAs.

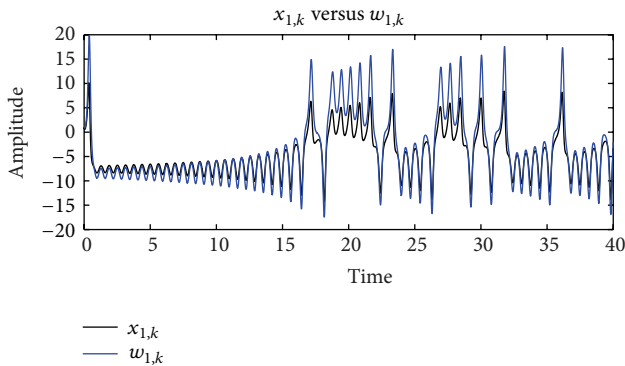


FIGURE 16: States $x_{1,k}$ and $w_{1,k}$ for the Generalized Synchronization of Chen-Lorenz discrete-time T-S fuzzy systems.

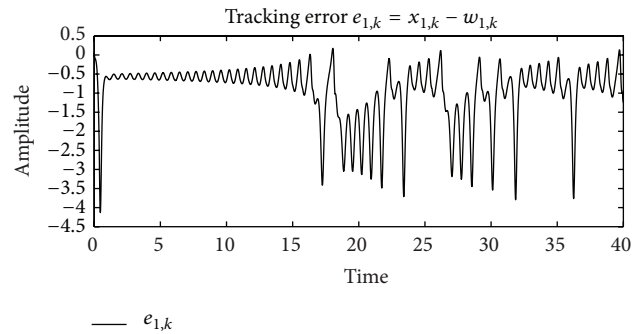


FIGURE 18: Tracking error $e_{1,k}$ signal for the Generalized Synchronization of Chen-Lorenz discrete-time T-S fuzzy systems.

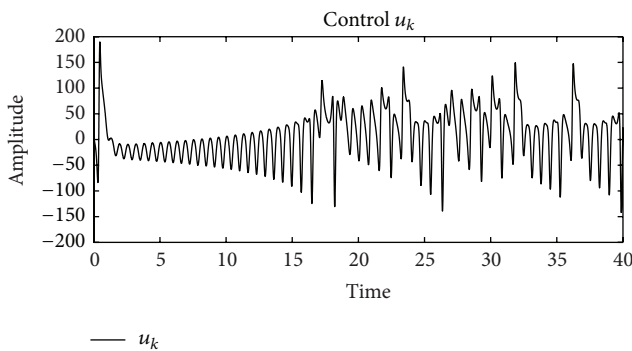


FIGURE 17: Control signal for the Generalized Synchronization of Chen-Lorenz discrete-time T-S fuzzy systems.

Notice that the final membership function depicted in Figure 23 is different from the initial proposed in Figure 19 and also notice that the sum of these new interpolation

functions is not equal to one. For that reason this approach is called Pseudofuzzy Generalized Synchronization.

5. Conclusions

A fuzzy output regulator for discrete-time systems, based on the combination of linear regulators combined by different membership functions, has been presented. Synchronization of discrete-time chaos attractors can be possible; by means of fuzzy output regulation, sufficient conditions for the controller are given. However, when the conditions can not be fulfilled, new membership functions in the regulator are included; these ones are optimized by genetic algorithms. The main advantage is that membership functions, which allow the proper combination of the local regulators, can be easily obtained by means of GAs. As a consequence, the presented result allows a very precise synchronization of chaotic systems described by T-S discrete-time fuzzy models on the basis of local regulators.

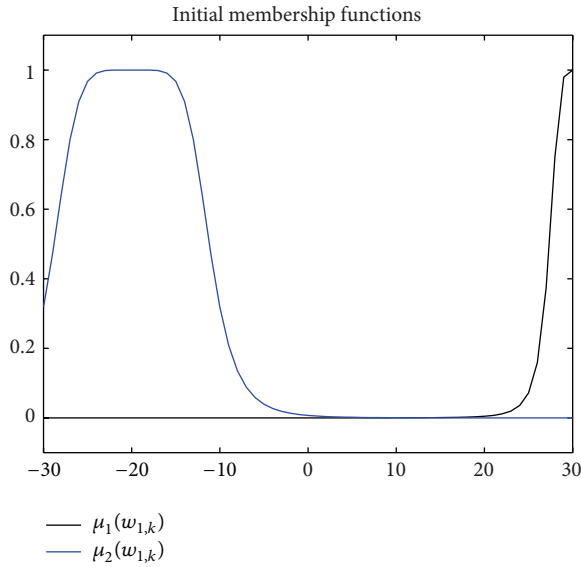


FIGURE 19: Initial membership functions for the pseudofuzzy regulator tuned by GAs.

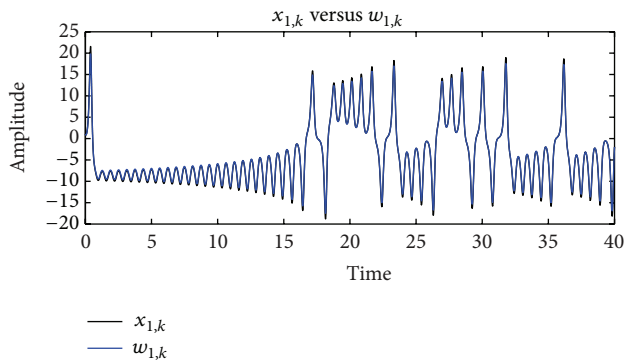


FIGURE 20: States $x_{1,k}$ and $w_{1,k}$ for the Generalized Synchronization of Chen-Lorenz discrete-time T-S fuzzy systems.

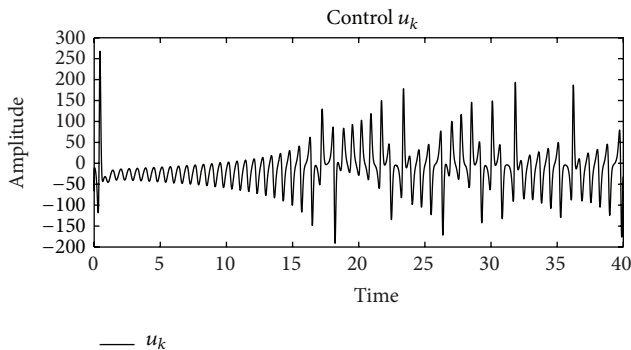


FIGURE 21: Control signal for the Generalized Synchronization of Chen-Lorenz discrete-time T-S fuzzy systems.

Complete and Generalized Synchronization are used to illustrate the applicability of the proposed approach. Besides, the method proposed in this work avoids the disadvantage of constructing an exact fuzzy regulator based on overall T-S fuzzy system which may result to be very large. Instead, the given approach offers a simple way to design the complete

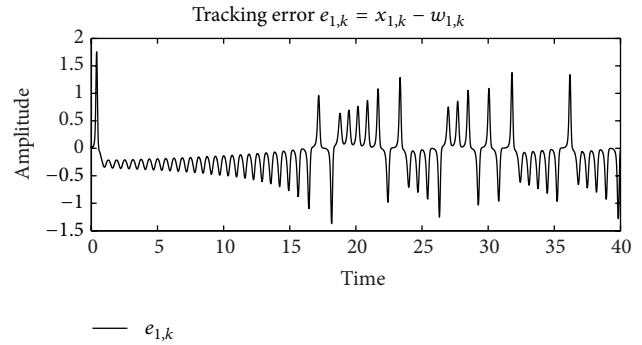


FIGURE 22: Tracking error $e_{1,k}$ for the Generalized Synchronization of Chen-Lorenz discrete-time T-S fuzzy systems.

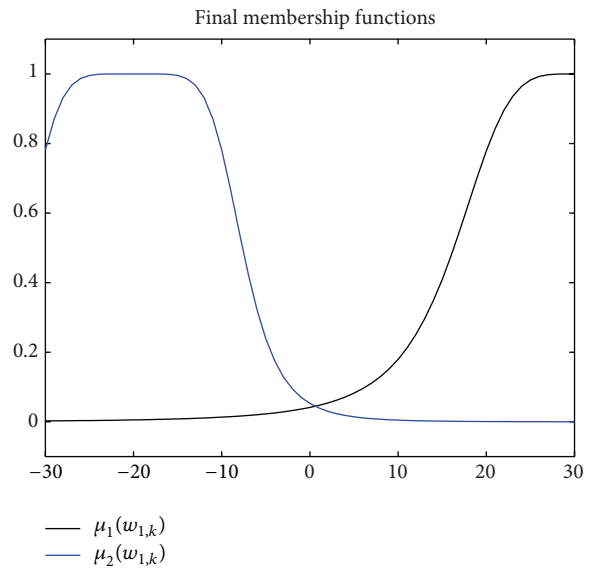


FIGURE 23: Final interpolation functions for the pseudofuzzy regulator tuned by GAs.

regulator based on local regulators but with membership functions optimized by soft computing.

Conflict of Interests

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

Acknowledgments

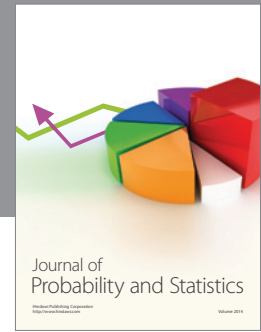
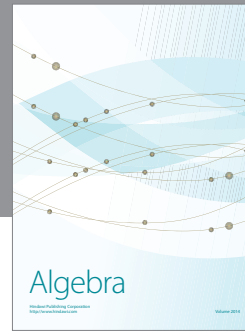
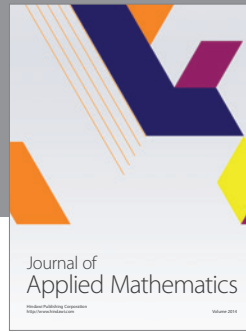
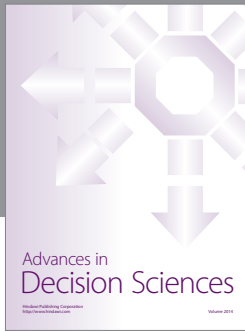
This work is partially supported by Consejo Nacional de Ciencia y Tecnología (CONACYT) through Scholarship SNI and by Instituto Politécnico Nacional (IPN) through research Projects 20150487 and 20150794 and Scholarships COFAA, EDI, and BEIFI.

References

[1] Z. Li and G. Chen, Eds., *Integration of Fuzzy Logic and Chaos Theory*, vol. 187, Springer, Berlin, Germany, 2006.

- [2] B. Wang, H. Cao, Y. Wang, and D. Zhu, "Linear matrix inequality based fuzzy synchronization for fractional order chaos," *Mathematical Problems in Engineering*, vol. 2015, Article ID 128580, 14 pages, 2015.
- [3] I. Rechenberg, *Optimization of technical systems after the principles of biological evolution [Ph.D. thesis]*, Fromman-Holzboog, Stuttgart, Germany, 1971.
- [4] H.-P. Schwefel, *Numerical Optimization of Computer Models*, John Wiley & Sons, New York, NY, USA, 1981.
- [5] H. Bremermann, *The Evolution of Intelligence: The Nervous System as a Model of Its Environment*, University of Washington, Department of Mathematics, 1958.
- [6] J. H. Holland, "Outline for a logical theory of adaptive systems," *Journal of the ACM*, vol. 9, no. 3, pp. 297–314, 1962.
- [7] J. H. Holland, *Adaptation in Natural and Artificial Systems*, MIT Press, Cambridge, Mass, USA, 1992.
- [8] D. Fogel and R. Anderson, "Revisiting Bremermann's genetic algorithm. I. Simultaneous mutation of all parameters," in *Proceedings of the Congress on Evolutionary Computation*, vol. 2, pp. 1204–1209, La Jolla, Calif, USA, 2000.
- [9] I. Rechenberg, "Cybernetic solution path of an experimental problem," in *Royal Aircraft Establishment Translation No. 1122*, B. F. Toms, Ed., Ministry of Aviation, Royal Aircraft Establishment, Farnborough Hants, UK, 1965.
- [10] C. Hugenii, "Horologium oscillatorium (parisiis, France, 1973)," Tech. Rep., Iowa State University Press, Ames, Iowa, 1986, (English translation: The pendulum clock).
- [11] L. M. Pecora and T. L. Carroll, "Synchronization in chaotic systems," *Physical Review Letters*, vol. 64, no. 8, pp. 821–824, 1990.
- [12] K. M. Cuomo and A. V. Oppenheim, "Circuit implementation of synchronized chaos with applications to communications," *Physical Review Letters*, vol. 71, no. 1, pp. 65–68, 1993.
- [13] M.-L. Hung and H.-T. Yau, "Circuit implementation and synchronization control of chaotic horizontal platform systems by wireless sensors," *Mathematical Problems in Engineering*, vol. 2013, Article ID 903584, 6 pages, 2013.
- [14] I. Zelinka, S. Celikovský, H. Richter, and G. Chen, *Evolutionary Algorithms and Chaotic Systems*, Studies in Computational Intelligence, Springer, Berlin, Germany, 2010.
- [15] K. Tanaka and H. O. Wang, *Fuzzy Control Systems Design and Analysis. A Linear Matrix Inequality Approach*, John Wiley & Sons, New York, NY, USA, 2001.
- [16] E. Solak, Ö. Morgül, and U. Ersoy, "Observer-based control of a class of chaotic systems," *Physics Letters A*, vol. 279, no. 1-2, pp. 47–55, 2001.
- [17] Ö. Morgül and E. Solak, "On the synchronization of chaos systems by using state observers," *International Journal of Bifurcation and Chaos*, vol. 7, no. 6, pp. 1307–1322, 1997.
- [18] P. Sangapate, "Adaptive control and synchronization of the shallow water model," *Mathematical Problems in Engineering*, vol. 2012, Article ID 529251, 9 pages, 2012.
- [19] A. Isidori and C. I. Byrnes, "Output regulation of nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 35, no. 2, pp. 131–140, 1990.
- [20] J. A. Meda-Campaña, B. Castillo-Toledo, and G. Chen, "Synchronization of chaotic systems from a fuzzy regulation approach," *Fuzzy Sets and Systems*, vol. 160, no. 19, pp. 2860–2875, 2009.
- [21] R. Tapia-Herrera, J. A. Meda-Campaña, S. Alcántara-Montes, T. Hernández-Cortés, and L. Salgado-Conrado, "Tuning of a TS fuzzy output regulator using the steepest descent approach and ANFIS," *Mathematical Problems in Engineering*, vol. 2013, Article ID 873430, 14 pages, 2013.
- [22] C. I. Byrnes, F. Delli Priscoli, and A. Isidori, *Output Regulation of Uncertain Nonlinear Systems*, Systems & Control: Foundations & Applications, Birkhäuser, Boston, Mass, USA, 1997.
- [23] A. Isidori, *Nonlinear Control Systems*, Springer, Berlin, Germany, 1995.
- [24] B. Castillo-Toledo and J. A. Meda-Campaña, "The fuzzy discrete-time robust regulation problem: an LMI approach," *IEEE Transactions on Fuzzy Systems*, vol. 12, no. 3, pp. 360–367, 2004.
- [25] B. A. Francis, "The linear multivariable regulator problem," *SIAM Journal on Control and Optimization*, vol. 15, no. 3, pp. 486–505, 1977.
- [26] L.-X. Wang, *A Course in Fuzzy Systems and Control*, Prentice Hall PTR, Upper Saddle River, NJ, USA, 1997.
- [27] B. Castillo-Toledo, J. A. Meda-Campaña, and A. Titli, "A fuzzy output regulator for takagi-sugeno fuzzy models," in *Proceedings of the IEEE International Symposium on Intelligent Control*, vol. 2, pp. 310–315, Houston, Tex, USA, December 2003.
- [28] J. A. Meda-Campaña and B. Castillo-Toledo, "On the output regulation for TS fuzzy models using sliding modes," in *Proceedings of the American Control Conference (ACC '05)*, pp. 4062–4067, Portland, Ore, USA, June 2005.
- [29] H. O. Wang, K. Tanaka, and M. Griffin, "Parallel distributed compensation of nonlinear systems by Takagi-Sugeno fuzzy model," in *Proceedings of the International Joint Conference of the 4th IEEE International Conference on Fuzzy Systems and the Second International Fuzzy Engineering Symposium*, vol. 2, pp. 531–538, Yokohama, Japan, March 1995.
- [30] K. Tanaka and M. Sugeno, "Stability analysis and design of fuzzy control systems," *Fuzzy Sets and Systems*, vol. 45, no. 2, pp. 135–156, 1992.
- [31] J. A. Meda-Campaña, J. Rodríguez-Valdez, T. Hernández-Cortés, R. Tapia-Herrera, and V. Nosov, "Analysis of the fuzzy controllability property and stabilization for a class of T-S fuzzy models," *IEEE Transactions on Fuzzy Systems*, vol. 23, no. 2, pp. 291–301, 2015.
- [32] I. Abdelmalek, N. Goléa, and M. Laid Hadjili, "A new fuzzy Lyapunov approach to non-quadratic stabilization of Takagi-Sugeno fuzzy models," *International Journal of Applied Mathematics and Computer Science*, vol. 17, no. 1, pp. 39–51, 2007.
- [33] B. Castillo-Toledo and J. A. Meda-Campaña, "The fuzzy discrete-time robust regulation problem: a LMI approach," in *Proceedings of the 41st IEEE Conference on Decision and Control*, vol. 2, pp. 2159–2164, IEEE, Las Vegas, Nev, USA, December 2002.
- [34] T. Hernández Cortés, J. Meda Campaña, L. Páramo Carranza, and J. Gómez Mancilla, "A simplified output regulator for a class of takagi-sugeno fuzzy models," *Mathematical Problems in Engineering*, vol. 2015, Article ID 148173, 18 pages, 2015.
- [35] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison Wesley Longman Publishing, Boston, Mass, USA, 1st edition, 1989.
- [36] K. Pyragas, "Weak and strong synchronization of chaos," *Physical Review E*, vol. 54, no. 5, pp. R4508–R4511, 1996.
- [37] L. Kocarev and U. Parlitz, "Generalized synchronization, predictability, and equivalence of unidirectionally coupled dynamical systems," *Physical Review Letters*, vol. 76, no. 11, pp. 1816–1819, 1996.

- [38] R. He and P. G. Vaidya, "Analysis and synthesis of synchronous periodic and chaotic systems," *Physical Review A*, vol. 46, no. 12, pp. 7387–7392, 1992.
- [39] A. H. Wright, "Genetic algorithms for real parameter optimization," in *Foundations of Genetic Algorithms*, pp. 205–218, Morgan Kaufmann, Boston, Mass, USA, 1991.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

