Fuzzy Model and Particle Swarm Optimization for Nonlinear Identification of a Chua's Oscillator

Ernesto Araujo, Member, IEEE and Leandro dos Santos Coelho

Abstract—The identification of a nonlinear system with chaotic behavior denoted Chua's oscillator by using fuzzy models intertwined with particle swarm optimization (PSO) method is presented. This hybrid approach is applied to experimental data generated by an inductorless Chua's circuit that consists of an electronic chaotic oscillator. Chua's circuit has been used as a test platform by various scientific and engineering communities related to the study of chaos due its great potential in technological applications, for instance, telecommunications, cryptography and physics. Fuzzy set theory has been evolved as a powerful modeling tool that can cope with uncertainties and nonlinearities in modeling and identification procedures. The identification of an optimized fuzzy model Takagi-Sugeno (T-S) fuzzy model involves two primary tasks: parameter tuning and structure optimization. The premise part of production rules is optimized here by using the particle swarm optimization method. In turn, least mean squares technique is applied to the consequent part of a T-S fuzzy model. Results indicate PSO method and Least Mean Square technique succeeded in constructing a T-S fuzzy model when dealing with chaotic dynamics obtained through experimental data supplied by inductorless Chua's electronic circuit.

I. INTRODUCTION

One important area of study in the field of nonlinear dynamics that interests from mathematicians, engineers to scientists of diverse disciplines is nonlinear model identification. Among the distinct approaches for nonlinear system identification, an alternative is to use theoretically sound nonlinear functions and to develop identification schemes for these model descriptions. Volterra series, artificial neural networks and nonlinear ARMAX (Auto Regressive Moving Average with eXogenous inputs) models are examples of this methodology. If these approaches present, for one side, the advantage of obtaining global models of the underlying system, for the other side, the main disadvantage is the high computational cost [1].

Another option that has received a great deal of attention and have been employed in many applications in nonlinear system identification is fuzzy modeling [2]–[5]. A central feature of these mathematical representations of nonlinear dynamic identification is that they are based on fuzzy sets instead of classical numbers – i.e., singleton sets for information coding. One sort of fuzzy model is Takagi-Sugeno (T-S) approach which exhibits both high nonlinearity and simple structure [6], [7]. Moreover, it is capable of approximating a complex system using fewer fuzzy rules when compared to Mamdani-type fuzzy models [8]. The identification problem in T-S modeling consists mainly of structure identification and parameter identification. The structure identification is related to find out both the premise and the consequent of the production rules, respectively, by determining the premise space partition and extracting the number of rules as well as determining the structure of the output elements (equations). Finally, the parameter-learning task consists of computing the system parameters (membership functions) so that a functional index based on the output errors is minimized.

Additionally to these methods, over the past few years, an increasing number of hybrid intelligent methodologies have been proposed in literature [9]–[11], in which two or more computational intelligence technologies (evolutionary algorithms, swarm intelligence approaches, fuzzy systems, or artificial neural networks) have been integrated to leverage the advantages of individual approaches. By combining smoothness and embedded empirical qualitative knowledge with adaptability and general learning ability, these hybrid systems improve an overall algorithm performance [11].

This paper addresses the hybrid Particle Swarm Optimization (PSO) and fuzzy approach used in a synergetic manner. Tuning a fuzzy T-S model by using PSO approach is used with great success in the field of computational intelligence [12]–[18]. PSO is a kind of evolutionary algorithm based on a population of individuals and motivated by the simulation of social behavior instead of the survival of the fittest individual [19]. Similar to the other population-based evolutionary algorithms, it is initialized with a population of random solutions. However, unlike the most of the evolutionary algorithms, each potential solution (individual) in PSO is also associated with a randomized velocity, and the potential solutions (particles) are, then, flown through the problem space.

The structure identification of the premise and the consequent part of production rules of T-S fuzzy system are carried out by distinct methods. The T-S fuzzy system design employs PSO approach for figuring out the premise part meanwhile least mean squares is used for the calculus of consequent part of production rules of a T-S fuzzy system for nonlinear identification.

Chua's oscillator is used as a source of information in order to explore the effectiveness of PSO approach in constructing an adequate T-S fuzzy model when dealing with nonlinear identification. The Chua's oscillators have been widely used in chaotic secure communication systems,
chaotic spread-spectrum communications, and some other fields. It is one of the simplest electronic circuits that is capable of producing chaotic behavior. It can exhibit a vast array of behaviors including an assortment of steady-states, bifurcations and routes to chaos [20].

II. OPTIMAL TAKAGI-SUGENO FUZZY SYSTEMS

A. Takagi-Sugeno Fuzzy Systems

T-S models have recently become a powerful practical engineering tool for modeling and control of complex systems. The TS model representation often provides efficient and computationally attractive solutions to a wide range of modeling problems introducing a powerful multiple model structure that is capable to approximate nonlinear dynamics, multiple operating modes and significant parameter and structure variations. The structure optimization procedure aims to find the optimal structure of the local models, the relevant premise variables and a suitable partition of the premise space.

The essential idea of T-S fuzzy model is the partitioning of the input space into fuzzy areas and the approximation of each area through a linear model in such a way that a global nonlinear model is computed. It is characterized as a set of IF-THEN rules where the consequent part are linear sub-models describing the dynamical behavior of distinct operational conditions meanwhile the antecedent part is in charge of interpolating these sub-systems. The <IF> statements> define the premise part that is featured as linguistic terms while the <THEN functions> constitute the consequent part of the fuzzy system characterized, but not limited, as linear polynomial terms. The global model is, then, obtained by the interpolation between these various local models. This model can be used to approximate a highly nonlinear function through simple structure using a small number of rules [22].

One feasible representation for T-S models is given next:

\[
R^{(j)}: \text{IF } z_i \text{ IS } A_{j}^{(i)} \text{ AND } \ldots \text{ AND } z_m \text{ IS } A_{j}^{(m)} \text{ THEN } g_j = w_0^j + w_1^j u_1^j + \ldots + w_p^j u_p^j.
\] (1)

The IF statements define the premise part while the THEN functions constitute the consequent part of the fuzzy system; \( z = [z_1, \ldots, z_m]^T \), such as \( i = 1, \ldots, m \), is the input vector of the premise \( p \), and \( A_{j}^{(i)} \) are labels of fuzzy sets. The parameters \( w = [w_0^j, \ldots, w_p^j]^T \) represents the input vector to the consequent part of \( R^{(j)} \) that comprising \( q_j \) terms; \( g_j = g_j(w^j) \) denotes the \( j \)-th rule output which is a linear polynomial of the consequent input terms \( w^j \), and \( w = [w_1^j, \ldots, w_p^j]^T \) are the polynomial coefficients that form the consequent parameter set. Each linguistic label \( A_{j}^{(i)} \) is associated with a Gaussian membership function, \( \mu_{A_{j}^{(i)}}(z_i) \), described by 2 where \( m_{ij} \) and \( \sigma_{ij} \) are, respectively, the centers/core and the spreads/support are, respectively, the mean value and the standard deviation of the Gaussian membership function:

\[
\mu_{A_{j}^{(i)}}(z_i) = \exp \left[ -\frac{1}{2} \frac{(z_i - m_{ij})^2}{\sigma_{ij}^2} \right].
\] (2)

The union of all these parameters formulates the set of premise parameters. The firing strength of rule \( R^{(j)} \) represents its excitation level and it is given by:

\[
\mu_j(z) = \mu_{A_{j}^{(1)}}(z_1) \mu_{A_{j}^{(2)}}(z_2) \ldots \mu_{A_{j}^{(m)}}(z_m).
\] (3)

The fuzzy sets pertaining to a rule form a fuzzy region (cluster) within the premise space, \( A_{j}^{(1)} \times \ldots \times A_{j}^{(m)} \), with a membership distribution described in (3). The membership function of the fuzzy output is inferred by taking the weighted average of the local outputs \( g_j(w^j) \) that is given by

\[
y = \sum_{j=1}^{M} v_j(z) g_j(w^j),
\] (4)

where \( M \) denotes the number of rules and \( v_j(z) \) is the normalized firing strength of \( R^{(j)} \), defined as:

\[
v_j(z) = \frac{\mu_j(z)}{\sum_{j=1}^{M} \mu_j(z)}.
\] (5)

In this paper, the structure identification of T-S system is computed based on PSO for premise part optimization while the consequent part optimization is determined by batch least mean squares method (pseudo-inversion method).

B. Particle Swarm Optimization (PSO) for T-S Fuzzy Modeling

Particle Swarm Optimization (PSO) originally developed by Kennedy and Eberhart in 1995 is a population-based swarm algorithm [21], [22]. Similarly to genetic algorithms, an evolutionary algorithm approach, PSO is an optimization tool based in a population where the position of each particle (member) is a potential solution to an analyzed problem. Each particle in PSO has associated a randomized velocity that moves through the problem space. One advantage of PSO over genetic algorithms is that this method does not have operators, i.e., crossover and mutation. Moreover, PSO does not implement the survival of the fittest individuals; instead, it implements the simulation of social behavior.

The proposal of PSO algorithm was put forward by several scientists who developed computational simulations of the movement of organisms such as flocks of birds and schools of fish. Such simulations were heavily based on manipulating the distances between individuals, i.e., the synchrony of the behavior of the swarm was seen as an effort to keep an optimal distance between them. The fundamental point of developing PSO is a hypothesis in which the exchange of information among creatures of the same species insists some sort of evolutionary advantage. It is assumed that individuals of a swarm may benefit from the prior discoveries and experiences of all the members of a group when foraging.

Each particle in PSO keeps track of its coordinates in the problem space, which are associated with the best solution (fitness) it has achieved so far. This value is called pbest. Another “best” value that is tracked by the global version of the particle swarm optimizer is the overall best value. Its location, gbest, is obtained indeed by any particle in the
population. The past best position and the entire best overall position of the group are employed to minimize (maximize) the solution. The PSO concept consists of, at each time step, changing the velocity (acceleration) of each particle flying toward its pbest and gbest locations (global version of PSO). Acceleration is weighted by random terms, with separate random numbers being generated for acceleration toward pbest and gbest locations, respectively. The procedure for implementing the global version of PSO is given by the following steps:

(i) Initialize a population (array) of particles with random positions and velocities in the n-dimensional problem space using uniform probability distribution function.

(ii) For each particle, evaluate its fitness value.

(iii) Compare each particles fitness with the particles pbest. If current value is better than pbest, then set pbest value equal to the current value and the pbest location equal to the current location in n-dimensional space.

(iv) Compare the fitness with the populations overall previous best. If current value is better than gbest, then reset gbest to the current particles array index and value.

(v) Change the velocity, \(v_i\), and position of the particle, \(x_i\), respectively according to eq. 6 and 7:

\[
\begin{align*}
v_i(t + 1) &= \omega \cdot v_i(t) + \ldots + c_1 \cdot a_i(t) \cdot (p_i(t) - x_i(t)) + \ldots \\
&+ c_2 \cdot U_d(t) \cdot (p_2(t) - x_i(t)) \\
x_i(t + 1) &= x_i(t) + \Delta t \cdot v_i(t + 1)
\end{align*}
\]

(vi) Return to step (ii) until a stop criterion is met, usually a sufficiently good fitness or a maximum number of iterations (generations).

In this approach \(x_i = [x_{i1}, \ldots, x_{in}]^T\) stands for the position and \(v_i = [v_{i1}, \ldots, v_{in}]^T\) for the velocity of the \(i\)-th particle, and \(p_i = [p_{i1}, \ldots, p_{in}]^T\) represents the best previous position of the \(i\)-th particle (the position giving the best fitness value). The first part in (6) is the momentum part of the particle. The second part is the cognition one, which represents the independent thinking of the particle itself. Equation (7) represents the position update, according to its previous position and speed, considering \(\Delta t = 1\).

The inertia weight, \(\omega\), represents the degree of the momentum of the particles. The use of variable \(\omega\), inertia weight as proposed by Shi and Eberhart [21], is responsible for dynamically adjust the velocity of the particles. This parameter is accountable for balancing between local and global search, consequently, needing less or more iterations for the algorithm to converge. A small value of inertia weight implies in a local search; a high one leads to a global search, yet with a high computational cost. However, linear decreasing inertia function may also be used if it is interested in reduce the influence of past velocities during the optimization process.

The index, \(g\), represents the index of the best particle among all the particles in the group. Variables \(a_i(t)\) and \(U_i(t)\) are two random numbers generated using uniform probability distribution functions in the range \([0, 1]\).

Positive constants \(c_1\) and \(c_2\) are called cognitive and social components, respectively. These are the acceleration constants responsible for varying the particle speed toward pbest and gbest. In this paper, the constriction coefficient method is used in PSO based on approach of Clerc and Kennedy [22]. In doing so, the velocity equation is updated according to:

\[
\begin{align*}
v_i(t + 1) &= K \cdot [v_i(t) + \ldots + c_1 \cdot a_i(t) \cdot (p_i(t) - x_i(t)) + \ldots \\
&+ c_2 \cdot U_d(t) \cdot (p_2(t) - x_i(t))].
\end{align*}
\]

by using constriction coefficient, \(K\):

\[
K = \frac{2}{2 - \varphi - \sqrt{\varphi^2 - 4\varphi}}
\]

with \(\varphi = c_1 + c_2, \varphi > 4\). Usually, \(\varphi\) is set to 4.1 (\(c_1 = c_2 = 2.05\)) and, then, the constriction coefficient \(K\) is 0.729.

III. NONLINEAR IDENTIFICATION OF CHUA’S OSCILLATOR

A. Chua’s Oscillator

Since mid-1980s, Chua’s circuit [29], [30], which is the most famous chaos circuit, has been studied by many researchers because it is simple and easy to build, and can produce many interesting phenomena, such as chaos, bifurcations, and attractors. From the early 1990s, applications of Chua’s circuit have been proposed in several engineering applications and many variations of the circuit have also appeared [26]-[28], [31]-[33]. In this work, an inductorless Chua’s circuit [25] is evaluated using a T-S fuzzy model in nonlinear identification procedure. Due to its physical nature it has been studied extensively and accepted as a paradigm for analysis of the chaotic behavior and chaotic signal both theoretically and through experiments or even computer simulations. In particular, Chua’s oscillator have received increasing attention from various scientific and engineering communities due to its great potential in technological applications [23]-[28].

This oscillator uses Chua’s diode, which is a piecewise linear element that supplies static nonlinearity, and can be realized as depicted in Fig 1. Here, Chua’s diode is built up as a network of linear passive elements in which the nonlinear characteristic are realized using operational amplifiers [26].

![Fig. 1. Chua’s oscillator realization through passive elements](image-url)
The conception of this inductorless Chua’s circuit structure was based in [24],[25] in which different values for the components of the electric circuit result in attractors with different geometry.

B. Identification of T-S fuzzy model

Identification of dynamic systems can be performed with a series-parallel or parallel model. Series-parallel structure is the type of mathematical model adopted for identification of Chua’s oscillator when using the hybrid PSO-T-S modeling approach.

The estimated T-S fuzzy model output based on PSO, \( \hat{y}(k) \), used for computing the minimum square error when compared with the actual output, \( y(k) \) was computed by using one-step ahead forecasting. Denote \( n_y \), \( n_u \), and \( n_\theta \) as the time maximum lags of the model output, control input, and noise, respectively. Depending on the time-lagged inputs that are used for the T-S fuzzy model, different configurations of models can be used. In this work, a NAR (Nonlinear Auto Regressive) model was adopted, given by

\[
\hat{y}(k) = f_{TS}(u(k-1),\ldots,u(k-n_u),\ldots,y(k-1),\ldots,y(k-n_y),\theta)
\]

where the unknown nonlinear function \( f_{TS} \) is the TS fuzzy model of the system, \( k \) represents the \( k \)th instant of time, and \( \theta \) is the estimated vector of parameters for the model.

One of the most important tasks in building an efficient forecasting model based in T-S fuzzy model is the selection of the relevant input variables. The input selection problem can be stated as follows: among a large set of potential input candidates, choose those variables that highly affect the model output. Unfortunately, there is no systematic procedure, currently available, which can be followed in all circumstances. In this work, input selection is heuristically performed. The inputs of T-S fuzzy system are process output and control input signals of reduced order with \( n_y = 2 \), \( n_u = 0 \), and \( n_\theta = 0 \). In this work, the vectors of input for the TS fuzzy system are \([y(k-2); y(k-1)]\) and the model output is \( \hat{y}(k) \).

The first part of Chua’s oscillator experimental data (voltage on capacitor C1) shown in main results through Figure 1 was employed to elicit the fuzzy model through PSO. In this case, 1000 samples in training (estimation) phase of T-S system design using PSO, and other 1160 in validation (test or generalization) phase of T-S.

Although, PSO allows to extract the number of rules and to determine the premise and consequent elements, here this method is applied to obtain membership functions and thus to determine the premise space partition.

Setting up this parameter as 2, 3 or 4 membership functions, PSO needs to deal with a vector of particles positions and velocity whose elements are 9, 12 or 15 centers and 2, 3 or 4 spreads of a Gaussian function, respectively, core and support of membership functions. In this case, the spread of Gaussian membership function adopted for each input of vectors \([y(k-1); y(k-2)]\) of T-S fuzzy model is the same.

The system identification by T-S fuzzy model is appropriate if a suitable performance index is available according to the necessities of users. Among a population of potential solution to a problem, every particle of PSO has a fitness value for expressing appropriate optimization result. The function representing this quality measure employs the position of all particles, \( x_i \), which is calculated after each iteration.

The performance criterion (fitness function) chosen for evaluate the relationship between the real output and the estimate output during the optimization process (maximization problem) was the Pearson multiple correlation coefficient index \( R^2 \) of training phase of TS fuzzy model conducted by \( R^2 \) as given by:

\[
R^2_{\text{training}} = 1 - \frac{\sum_{k=1}^{N_u} [y(k) - \hat{y}(k)]^2}{\sum_{k=1}^{N_u} [y(k) - \bar{y}]^2}
\]

where \( N_u \) is the total number of samples evaluated in estimation phase, and \( \hat{y}(k) \) is the system real output. When \( R(\cdot)^2 \) is close to unit, a sufficient accurate model for the measured data of the system is found. A \( R^2 \) between 0.9 and 1.0 is suitable for applications in identification and model-based control. In this context, the performance evaluation of validation phase of optimized T-S fuzzy system is realized by

\[
R^2_{\text{validation}} = 1 - \frac{\sum_{k=1}^{N_v} [y(k) - \hat{y}(k)]^2}{\sum_{k=1}^{N_v} [y(k) - \bar{y}]^2}
\]

where \( N_v \) is the total number of samples evaluated in validation phase. The main parameters deeply related to the success of PSO for tuning the premise part of T-S fuzzy model are: (i) the number of particles (size of population), (ii) the initial position and velocity of particles, (iii) the cognitive and social components \( (c_1 \) and \( c_2 \)), (iv) the form of inertia factor updating, and (v) stopping criterion, \( t_{\text{max}} \) (\( t_{\text{max}} = 30 \) iterations). One of advantage of this technique is that the initial population of particles is randomly generated through a uniform probability distribution function. The sufficient number of particles for this application was setup as 5. The main parameters of PSO are shown in Table I.

I. MAIN RESULTS

In order to illustrate the effectiveness of the T-S fuzzy model several simulations were performed. All the programs were run on a 3.8 GHz Pentium IV processor with 2 GB of RAM. In each case studied, 30 independent runs were

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Selection</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of particles</td>
<td>5</td>
</tr>
<tr>
<td>Number of iterations (( t_{\text{max}} ))</td>
<td>30</td>
</tr>
<tr>
<td>Inertia weight setup</td>
<td>( k^{**} = 0.729 )</td>
</tr>
<tr>
<td>Cognitive component</td>
<td>( c_1 = 2.05 )</td>
</tr>
<tr>
<td>Social component</td>
<td>( c_2 = 2.05 )</td>
</tr>
<tr>
<td>* generations = stopping criterion, ( ** ) constriction factor</td>
<td></td>
</tr>
</tbody>
</table>
made for each of the optimization methods involving 5 different initial trial solutions (number of particles) for each optimization method.

All tested PSO-TS fuzzy models achieved a good approximation for experimental data in training and validation phases. Continuous and dashed lines represent measured and simulated outputs in results presented in Figures (2). Experimental results had shown that the integrating PSO and T-S fuzzy system presented successful results due to precision in one-step ahead predicting nonlinear dynamics.

Results (best of 30 independent runs with 30 generations) obtained for the PSO of T-S fuzzy system design in training
and validation phases are presented in Tables II and III. The best mean fitness value ($R^2_{\text{training}}$) was obtained by T-S with 4 membership functions. The results using T-S with 4 production rules present small standard deviation and also the best $R^2_{\text{training}}$ minimum value as it is presented in Table II. However, the computational cost (mean CPU time of each run) of this design is superior to the T-S fuzzy system with 2 and 3 membership functions.

V. CONCLUSION AND FUTURE RESEARCH

Fuzzy models represent attractive platforms to model nonlinear chaotic dynamics since fuzzy approach can appropriately (under very well defined conditions) incorporate efficient tuning algorithms. In this work, a PSO approach is evaluated when working in a sinergetic manner with T-S fuzzy system design.

This hybrid approach accomplished its goal of finding out nonlinear models upon experimental data generated by an inductorless Chua’s circuit. In doing so, this approach is able to be used in studies of the chaos in real world applications. The computational efficiency and the accuracy of the proposed methodology integrating T-S fuzzy model and PSO make it very well suited for applications in the design of nonlinear identification models for a wide class of the complex systems. It is worth mentioning that the elicited fuzzy model with only two membership functions determining the premise space partition demonstrated its effectiveness in emulating the time response for the Chua’s oscillator. Nevertheless, as expected, additional number of membership functions as well as more production rules give better results but a high computational cost.

Future work includes mechanisms from computational intelligence field in order to compare the proposed approach with other learning and optimization mechanisms to nonlinear identification through the use of statistical analysis.

REFERENCES


