Control of Dynamical Systems: An Intelligent Approach

Soukkou Ammar, Abdelhafid Khellaf, Salah Leulmi, and Mourad Grimes

Abstract: In this paper, we introduce a fuzzy nonlinear feedback approach to the control of a class of chaotic dynamical systems. The fuzzy Parallel Distributed Compensation with Reduced Rule Base approach (PDC_RRB) is proposed. The design procedure is conceptually simple and considered to a nonlinear optimal and robust control problem due to the nonlinear nature of the Takagi-Sugeno (TS) fuzzy system. Simulation results are provided to show the effectiveness of the proposed methodology.

Keywords: Chaotic systems, genetic learning, nonlinear optimal controller.

1. INTRODUCTION

Chaos is a special feature of parametric nonlinear dynamical systems. It is usually difficult to accurately predict its future behaviour [1]. Recently, significant attention has been focused on developing techniques for the control of chaotic dynamical systems [2]. Generally, there are two ways to control chaos. Feedback control methods are used to control chaos by stabilizing a desired unstable periodic solution, which is embedded in a chaotic attractor, and non-feedback control methods [3] suppress chaotic behavior by applying weak periodic perturbation to some control parameters or variables. Although several control methods have been extensively applied to control regular and chaotic behavior of smooth nonlinear dynamical systems, non-smooth systems can also be controlled.

While chaos has become one of the most focusing research topics in the literature, we have witnessed rapidly growing interest in making the control systems more intelligent. Among intelligent control approaches, fuzzy control [2,4-6], neural networks modelling and control [1,7,8] and genetic optimization [9] have enjoyed remarkable success in various applications.

The fuzzy control designed is carried out based on

Soukkou Ammar and Mourad Grimes are with the Department of Electronics, University of Jijel, BP. 98, Ouled Aissa, Jijel 18000, Algeria (e-mails: {soukkou_a, grimes_morad}@yahoo.fr).

Abdelhafid Khellaf is with the Department of Electronics, University of Ferhat Abbas-Setif 19000, Algeria Algeria (e-mail: ah_khellaf@yahoo.fr).

Salah Leulmi is with the Department of Electrotechnics, University of Skikda 21000, Algeria (e-mail: leulmi_salah @yahoo.fr).

the fuzzy model via a so called Parallel Distributed Compensation (PDC) strategy [10,11]. The idea is that for each local linear model, a linear feedback control is designed.

The resulting overall controller (which is nonlinear in general) is a fuzzy blending of each individual linear controller. To designing an optimal controller, an efficient optimization technique should be used. In particular, evolutionary computation has received considerable attention in recent years [12]. Genetic Algorithms (GAs) [13] are optimization routines that operate in a similar manner to natural genetic selection. GAs have been proposed as a learning method that allows automatic generation of optimal parameters for fuzzy controllers based on an objective criterion. The concept of chaos being radically different from statistical randomness is introduced in this paper to solve the problem of maintaining the population diversity of GA in the learning of PDC_RRB. Concerning the performance of fuzzy control systems, the optimality and robustness have quite often been considered as the important issues. Specifically, on the optimality issue for fuzzy control systems [14]. In this paper, we propose a fuzzy nonlinear feedback approach to the control of a class of chaotic systems. Our aim is to alter the dynamics of the given chaotic system appropriately by using the control input to obtain a desirable behaviour, i.e. to drive the system from chaos to periodic behaviour. Another problem we address is to force a chaotic system to track a reference trajectory. Furthermore, we consider the Duffing chaotic system, the Van der Pol oscillator and Chua's circuit as illustrative examples, and their numerical simulation results show that, in spite of system uncertainties, the proposed PDC_RRB controller can be successfully applied for tracking a periodic orbit. The remaining part of this paper is organized as follows. The design and learning algorithm of the proposed system is described in Section 2 and 3, respectively. Some simulation results

to illustrate the effectiveness of the proposed control system structure are displayed in Section 4. Finally, Section 5 concludes the paper.

2. DESIGN PROCESS

2.1. Problem statement

The problem to be addressed is that of achieving the optimal set-point control of a process driven by a nonlinear controller. Consider a nonlinear system described by

$$
\begin{cases}
\dot{x}(t) = g\big(x(t), u(t), d_{ist}(t)\big) \\
y(t) = Cx(t) \\
\dot{d}_{ist}(t) = s\big(d_{ist}(t)\big) \\
x(o) = x_0, \qquad t \in \big[0, t_f\big],
\end{cases}
\tag{1}
$$

where $x \in \Re^n$, $u \in \Re^m$ and $y \in \Re^q$ are the state vector, the control vector and the output vector, respectively. $d_{ist} \in R^q$ represents a bounded external disturbance, x_0 is the initial state vector, $g : C^\infty(\Re^n \times \Re^m) \to \Re^n, g(0,0) = 0$, and $C \in \Re^{q \times n}$ is a constant matrix. In general, the objective of the control is to find the optimal law u^* such that

$$
\lim_{t \to \infty} \big| y(t) - y^d(t) \big| \le \varepsilon,
\tag{2}
$$

where ε is a suitably chosen constant. That is the faster y tracks the reference model y^d, and the performance index given by the user is minimized (the better the controller will perform [15,16]). The diagram of plant control loop is plotted in Fig. 1 that contains four blocks:
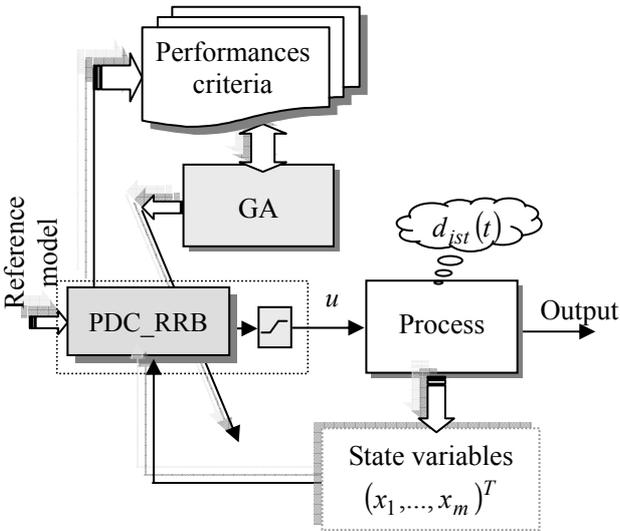

Fig. 1. Control and optimization structure.

i) Optimization / tuning block characterized by GA.
ii) Structural block representing the PDC_RRB.
iii) Decisions block defining the performances criteria.
iv) System block to be controlled.

The interaction between these four blocks is summarized by
1. Generating initial population of chromosomes (each chromosome represents a optimal fuzzy knowledge base).
2. Projecting of each chromosome on the structure of PDC_RRB.
3. For all chromosomes and all $\big(x^T, y^d\big) \in \Omega$,

   • Evaluate fitness &
   • Classify the chromosomes according to their fitness.

Steps 2 and 3 are repeated until a maximum number of generations is carried out. After the evolution process, the final generation of population consists of highly fit strings that provide optimal or near optimal solutions.

2.2. Proposed optimal controller

The history of the Parallel Distributed Compensation (PDC) started with a model-based design process proposed by Tanaka and Sugeno [17]. The PDC offers a scheme to design a fuzzy controller from the TS fuzzy model. Compared with the widely used PI, PD and PID controllers that require tuning only two or three parameters, the TS controller using PDC is extremely far removed from ease-of-use [10]. To overcome this disadvantage, a new control scheme called Parallel Distributed Compensation with Reduced Rule Base (PDC_RRB), which can significantly reduce the number of parameters in PDC, is proposed in this work. The ith rule of the proposed PDC_RRB is as follows:

$$
\underbrace{R^{(i)}}_{i=1,\dots,M} : \text{IF } \tilde{x}_1 \text{ is } A_1^{(i)} \dots \text{ AND } \tilde{x}_m \text{ is } A_m^{(i)} \text{ THEN}
$$
$$
u^{(i)} = -K^{(i)} \cdot \tilde{x} \ \underline{with} \ \big\{ \tilde{c}_F^{(i)} \big\},
\tag{3}
$$

where $A_1^{(i)} \sim A_m^{(i)}$ are linguistic values of the fuzzy variables to express the universe of discourse of the fuzzy set in the antecedent. $K^{(i)}, \big(i = 1,\dots,M\big)$ represent the feedback gain vector (resp. gain matrix in the MIMO case). $\tilde{x}^T = \big[\tilde{x}_1,\dots,\tilde{x}_m\big]$ is a vector input (system states) and m is a number of input variables. $\tilde{x}_i = x_i - x_i^d$ and x^d is a state reference trajectory. $\big\{ \tilde{c}_F^{(i)} \big\}$ represents the certainty factor of the ith rule [18,19]. The latter can take only two values; either 0

or 1 ($\left\{ \tilde{c}_F^{(i)} \right\} = 1/0$, characterizes enabled/disabled rule). $u^{(i)}$ the ith rule controller at time t and <u>with</u> is the operator modelling the certainty factor of a rule [20]. The overall state feedback fuzzy controller is represented by

$$u^* = \frac{-\sum_{i=1}^{i=M} w^{(i)}(\tilde{x}) \cdot \tilde{c}_F^{(i)} \cdot K^{(i)} \cdot \tilde{x}}{\sum_{i=1}^{i=M} w^{(i)}(\tilde{x})}$$

$$= -\sum_{i=1}^{M} h^{(i)}(\tilde{x}) \cdot \tilde{c}_F^{(i)} \cdot K^{(i)} \cdot \tilde{x}, \qquad (4)$$

where,

$$\underbrace{w^{(i)}(\tilde{x})}_{i=1,\dots,M} = \prod_{j=1}^{j=m} \mu_{ij}(\tilde{x}), \quad \begin{cases} \sum_{i=1}^{i=M} w^{(i)}(\tilde{x}) \neq 0 \\ w^{(i)}(\tilde{x}) \geq 0, \end{cases} \qquad (5)$$

$$h^{(i)}(\tilde{x}) = \frac{w^{(i)}(\tilde{x})}{\sum_{i=1}^{i=M} w^{(i)}(\tilde{x})}, \quad \begin{cases} \sum_{i=1}^{i=M} h^{(i)}(\tilde{x}) = 1 \\ h^{(i)}(\tilde{x}) \geq 0. \end{cases} \qquad (6)$$

$\mu_{ij}(\tilde{x})$ is the grade of membership of $\tilde{x}$ in fuzzy set $A_j^{(i)}$ and $h^{(i)}(\tilde{x})$ denotes the normalized weight of each fuzzy rule. From the expression (4), the optimal control law will be represented by

$$u^* = u^{opt} = -\underline{\Psi}^T(\tilde{x}) \cdot \tilde{x}, \qquad (7)$$

where $\underline{\Psi}^T(\tilde{x})$ is the nonlinear feedback gain vector (resp. gain matrix in the MIMO case) due to the nonlinear nature of the Takagi-Sugeno fuzzy system. The problem considered in this paper is to find the optimal PDC_RRB controller, i.e., the optimal nonlinear gain $\Psi^T(\tilde{x})$ based on the following objectives:
(1) Minimize the absolute error between the output signals and reference models.
(2) Reduce the quantity of the energy of the control signals applied to the system.
(3) Minimize the number of fuzzy rules.

Those objectives are selected with respect to the imposed constraints (fuzzy constraints membership function and system control constraints). The discrete step values of u^* equispaced over process operation time are considered as optimization variables.

One of the most important issues in fuzzy systems is how to reduce the rule base and their corresponding computation requirements. Generally, the number of fuzzy rules grows exponentially with the number of input variables. Specifically, a single output fuzzy system with m input variables and n fuzzy sets defined for each input variable requires n^m number of fuzzy rules. As it is well known, the curse-of-dimensionality is an unsolved problem in the fields of fuzzy and neuro-fuzzy systems [21]. To construct fuzzy systems using as less as possible fuzzy rules with guaranteed desired performances is a meaningful problem, which has attracted much attention for a long time in the fuzzy community [18,20,22,23]. Wan *et al.* [24] introduced a computational geometry approach to determine the minimum number of rules required in building a fuzzy model to achieve a given approximation accuracy. In [25,26], the authors present a systematic procedure of fuzzy control system design that consists of fuzzy model construction, rule reduction, and robust compensation for nonlinear systems using a generalized form of TS fuzzy systems. Conditions to reduce the number of rules have been represented in term of LMIs. Lee [19] presented two phases optimization of fuzzy controller with weighted rule based on evolutionary programming and the principle of maximum entropy.

The use of hierarchical structure in designing a fuzzy system has been reported in [21,27,28]. The hierarchical topology of fuzzy system permits to reduce the size of rule base to some extend. The number of fuzzy rules employed in a hierarchical fuzzy system is proportional to the number of input variables.

3. GENETIC LEARNING

Genetic learning (search) strategies are population based probabilistic optimization techniques mimicking Darwin's idea of natural selection [13]. Some of the advantages of a GA include that it [29]:
- Optimizes with continuous or discrete variables.
- Doesn't require derivative information.
- Simultaneously searches from a wide sampling of the cost surface.
- Deals with a large number of variables.
- Is well suited for parallel computers.
- Optimizes variables with extremely complex cost surfaces (they can jump out of a local minimum).
- Provides a list of optimum variables, not just a single solution.
- May encode the variables so that the optimization is done with the encoded variables, and works with numerically generated data, experimental data, or analytical functions.

The main purpose of introducing the GA to the design of a fuzzy controller is not only to use the robust and global benefits of GA but also to develop a systematic design approach of the fuzzy controller.

The concrete steps realizing the genetic optimization of PDC_RRB are summarized as follows:
1. Create some chromosomes randomly.

2. Evaluation of fitness value: Calculate the fitness value of each chromosome in the population:

$$F_{it} = \left(1 + J\right)^{-1}, \tag{8}$$

where J is the optimisation index (equation (15)).

3. Application of genetic operators: Selection, crossover and mutation. The actual process of mutation depends on the coding form. In binary coding part of chromosome, mutation only performs 1-bit flip, i.e., the bit value changes from '0' to '1' or from '1' to '0'. If the mutation takes place in the real coded part, we develop a chaotic mutation operator inspired of the works presented in [30-32].

4. Elitism: To put a limitation to the genetic divergence, one of the elitism strategies has been introduced. The latter, based on the technic of the steady-state selection, permits to construct a new more effective generation than the previous (the best member in last generation (*gen*-1*)* will be substituted into the worst member in the actual generation).

If the stop criterion is satisfied, return the chromosome with the best fitness, i.e., the optimal PDC_RRB. Otherwise, go to step 2 and proceed with the next generation. In the Fig. 2, L_{XY1} & L_{XY2} (real coding) are the widths of the universes of discourses of the fuzzy subsets of the input variables x_1 and x_2, respectively. $K_{x1,2}$ represent the scaling factors of the input variables (real coding).

The partitions are symmetric about the membership function *ZE*. This approach simplifies the computation while typically giving robust and satisfactory results. It also simplifies the optimisation testing of the GA. We assume that MFs are strictly monotone decreasing (or increasing) and continuous functions with respect to x_i, while x_k^L is a maximal left tolerance limit to b_k and x_k^R is a maximal right tolerance limit to b_k.
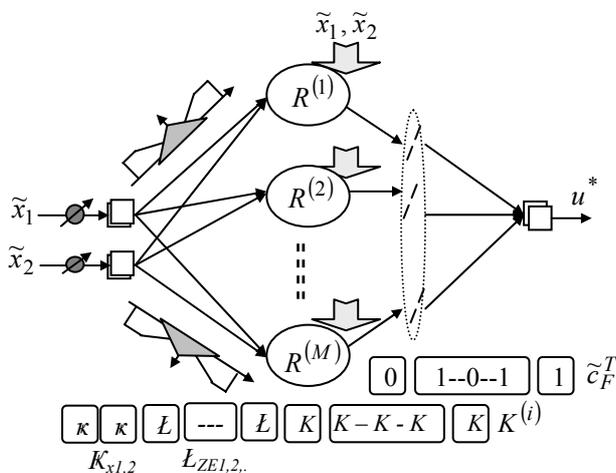
$$\underbrace{\mu_{A_i^j}^{\overbrace{ZE}}(x_i)}_{\text{Fuzzy Equal}} = \begin{cases} 0 & \text{if } x_i < b_i - x_i^L \\[2mm] \dfrac{x_i - \left(b_i - x_i^L\right)}{x_i^L} & \text{if } b_i - x_i^L \le x_i \le b_i \\[2mm] 1 & \text{if } x_i = b_i \\[2mm] 1 - \dfrac{x_i - b_i}{x_i^R} & \text{if } b_i \le x_i \le b_i + x_i^R \\[2mm] 0 & \text{if } x_i > b_i + x_i^R \end{cases} \tag{9}$$

3.1. Chaotic mutation

Chaos theory indicates that gene mutation can be regarded as action of chaos dynamic. So chaos is used to improve mutation. Operator of mutation for real coded GA is always random mutation. Generally nonlinear [33-35] or Gaussian distribution [31]. The logistic map display many of the generic features of chaotic dynamical systems such as the transition from regular to random behaviour, the presence of period-doubling bifurcation cascades, the influence of attractor and the underlying characteristics of a fractal. The chaos model used is the logistic mapping [36]:

$$x_{i+1} = A \cdot x_i \cdot \left(1 - x_i\right) \tag{10}$$

in which $A \in [0,4]$ is a control parameter (chaos attractor), $i = 0,1,2,....$, and x is chaotic variable. If $A \in [3.56,4]$, then the above system enters into a chaos state and chaotic variable x_i is produced. The chaotic variable is in the interval $[0,1]$, and is extreme sensitive to initial conditions. Fig. 3 illustrates the bifurcation for logistic mapping. For each value of A the initial value of x was taken as 0.3 and the ordinate is x_{i+1} for $i = 1,2,...,100$. The bifurcation occurs from $A \geq 3.0$.

The performance of GA mainly depends on the genetic operations applied to the chromosome,

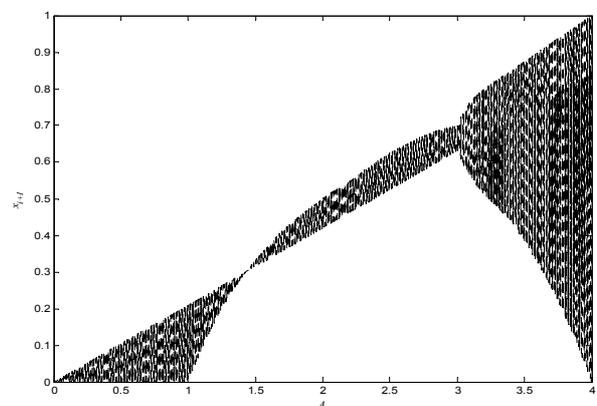Fig. 2. Chromosome structure and corresponding rule base.

Fig. 3. Bifurcation for logistic mapping.

mainly: the selection, crossover and mutation. The mutation operator plays an important role in maintaining the population diversity over the space of interest through the GA. There are many merits of real coded GA. There exists a number of mutation operators defined in the literature, which of course, depends on the representation used: binary or real coding. The most used mutation operators classified by the representation type are :

• Mutation operators for bit string.
• Mutation operators for real-coded string. Generally three types are used in the literature: standard mutation, uniform and non uniform mutation. Given the vector (chromosome):

$$Ch_v^{(gen)} = [V_1, \cdots, V_K, \cdots, V_m], \qquad (11)$$

where $V_i|_{i=1,\ldots,m}$ is the ith gene of the vector Ch_v in generation *gen*.

For binary string, the usual mutation operator just performs a bit-flipping in one of the positions of the individual [13]. The *standard mutation* operator of real coding chromosomes consists in the addition of a genes to be mutated a random values [37] as follows

$$\tilde{V}_K = V_K + N(0, \sigma_K), \qquad (12)$$

where $N(0, \sigma_K)$ is random number drawn from a Gaussian distribution with zero mean and standard deviation σ_K. In [38], the author applied this operator with an adaptive covariance. In the *uniform mutation* (boundary mutation) for real coding-type, a random selected element V_K is replaced by $\tilde{V}_K$, which is a random number in the range $\left[V_K^{\min}, V_K^{\max} \right]$.

The *non-uniform mutation* [39] modifies its disruptive effect depending on the stage of evolutionary process. The effect of this mutation operator is to perform big exploratory steps in the beginning of the evolutionary process and small steps at the end.

The precision of real coded GA is generally much better than that of binary coded GA [32]. In real coded GA, generally, the arithmetic (resp. standard) crossover operation and the uniform (resp. non-uniform) mutation are designed to preserve the constraint.

The chaotic mutation is introduced for maintaining the population diversity during the evolution process of the genetic algorithm. In this paper, the chaotic mutation operator is defined as follows

$$\tilde{V}_K = V_K^{\min} + Mut_{(V_K)} \left(V_K^{\max} - V_K^{\min} \right), \qquad (13)$$

where

$$Mut_{(V_K)} = A \cdot V_K \cdot (1 - V_K), \qquad (14)$$

where $A = 3.88$ is a control parameter.

4. SIMULATION

This section aims at illustrating some capabilities of the proposed PDC_RRB controller to suppress chaos in nonlinear chaotic systems. We use the Duffing oscillator, the extended Van der Pol oscillator and the Chua's circuit to illustrate the efficiency of our feedback strategy.

The optimisation of the PDC_RRB is to find the '*best*' structure and the parameters, i.e., an optimal fuzzy knowledge base, which can be represented as an extremum problem of optimisation index. As indicated in the following formula:

Minimize J, with

$$
\begin{aligned}
J = & \sum_{k=1}^{max\_t} \underbrace{C^{(1)}(k) \cdot \left(\tilde{x}^T(k) \cdot Q \cdot \tilde{x}(k) \right)}_{J_{\text{Error\_States}}} \\
& + \sum_{k=0}^{max\_t-1} C^{(2)}(k) \cdot \underbrace{\left(u^T(k) \cdot R \cdot u(k) \right)}_{J_{\text{Energy}}} \\
& + \underbrace{C^{(3)}(k) \cdot \sum_{i=1}^{Max\_Rl} \left(\tilde{c}_F^{(i)} \right)}_{J_{\text{Rule\_Base}}}
\end{aligned}
\qquad (15)
$$

$$
\text{Subject to} \underbrace{\begin{cases} \sum_i \left(\tilde{c}_F^{(i)} \right) \geq 1 \\ \sum_{i=1}^{i=M} w^{(i)}(\tilde{x}) \neq 0 \quad (\text{in } (4)) \end{cases}}_{\text{Optimization index}}
$$

where $C^{(i)}(k)\Big|_{i=1,2,3}^{k=1,\ldots,max\_t}$ are a time varying weighting factors, which can be manipulated to get the best response according to problem definition. $Q = R = I^{2 \times 2}$ (matrix identity), *max_t* is the maximum of time and *Max_Rl* is the maximal number of rules. A constraints are violated if $\sum_i \left(\tilde{c}_F^{(i)} \right) = 0$

and/or $\sum_{i=1}^{i=M} w^{(i)}(\tilde{x}) = 0$ in (4), respectively. IF one of the optimization constraints is violated, THEN reject the corresponding chromosome and regenerate another chromosome. This process is repeated until all optimization constraints satisfied.

The controller thus developed for this application is initially characterized by: 5 fuzzy subsets for the first input and 5 fuzzy subsets for the second input. This

Table. 1. Specifications of the GA.

Parameters	Value/Type
Population Size	10
Max_Gen	500/100
Representation	Mixed binary-real
Initialization	Random
Scaling factors $K_{x1,2}$	$[K_{SF1}, K_{SF2}]$
Feedback gains $K^{(i)}$	$[-K_{Fg1}, K_{Fg1}]$
universe of discourses $L_{ZE1,2}$	[-1, 1]
Crossover operator	Dual-point with probability $P_c = 0.8$
Mutation operator	Mixed, uniform and chaotic with probability $P_m = 0.02$

gives 25 rules (*Max_Rl* = 25 in (15)). Specifications of the GA mechanisms are listed in Table 1.

Example 1: Consider the Duffing chaotic system whose dynamics is as following [40,41]

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = -0.1 \cdot x_2 - x_1^3 + 12 \cdot \cos(t) + u + d_{ist}, \end{cases} \quad (16)$$

where u is the control input, d_{ist} is a bounded external disturbance given by

$$d_{ist}(t) = 0.001 \cdot \sin(2t) \exp(-0.1t). \quad (17)$$

If $u = 0$ and $d_{ist} = 0$, then the system is chaotic system. The phase plane is shown in Fig. 4 for initial condition $x_1(0) = x_2(0) = 2$.

Using the PDC_RRB to control Duffing chaotic
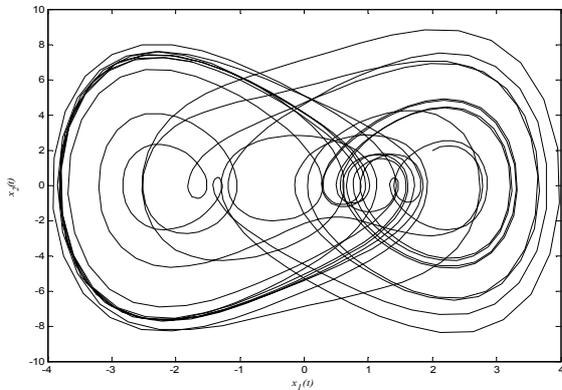
Fig. 4. Phase plane without controller $u = 0$ and disturbance $d_{ist} = 0$.
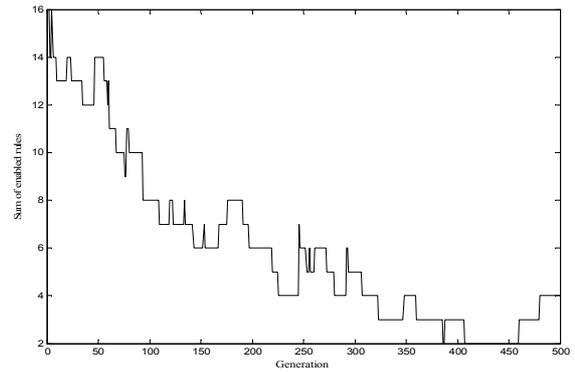
Fig. 5. Number of enabled rules.
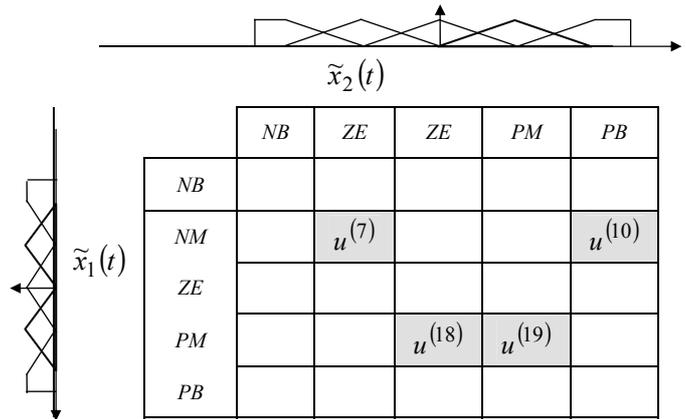
Fig. 6. The rule base of PDC_RRB.

system in order to force the states x_1 and x_2 to track the given bounded reference signals:

$$(x_1^d, x_2^d) = (\sin(t), \cos(t)). \quad (18)$$

In the phase plane, this reference trajectory is the unit circle: $(x_1^d)^2 + (x_2^d)^2 = 1$. For simulation, $C^{(i)}(k)\big|_{i=1,2,3}^{k=1,\dots,max\_t} = \{0.1, \ 0.001, \ 0.1\}$ and $max\_t = 1000$ in (15), respectively. $[K_{SF1}, K_{SF2}] = [5, 20]$ and $[-K_{Fg1}, K_{Fg1}] = [-1000, 1000]$ in Table 1. The enabled rule numbers during the evolution of GA is represented in Fig. 5.
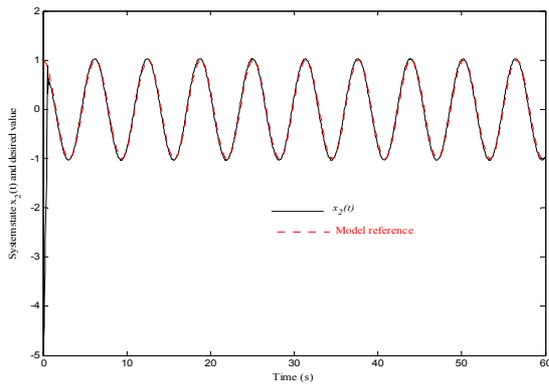
Fig. 6 represents the rule base obtained at the end of GA execution.

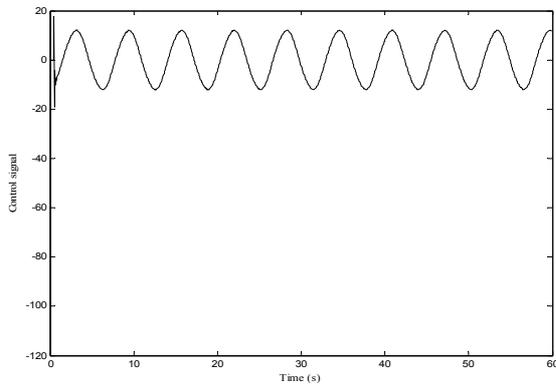$$u^{(i)}(k)\Big|_{i=7,10}^{i=18,19} = -K^{(i)} \cdot (x - x^d) \ \&$$

$$\begin{cases} K^{(7)} = \begin{bmatrix} -254.57 & 460.09 \end{bmatrix} \\ K^{(10)} = \begin{bmatrix} 94.99 & 363.41 \end{bmatrix} \\ K^{(18)} = \begin{bmatrix} -229.65 & 660.16 \end{bmatrix} \\ K^{(19)} = \begin{bmatrix} 831.46 & 297.48 \end{bmatrix} \end{cases} \quad (19)$$
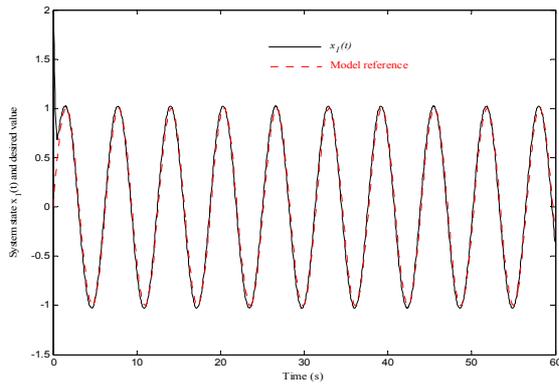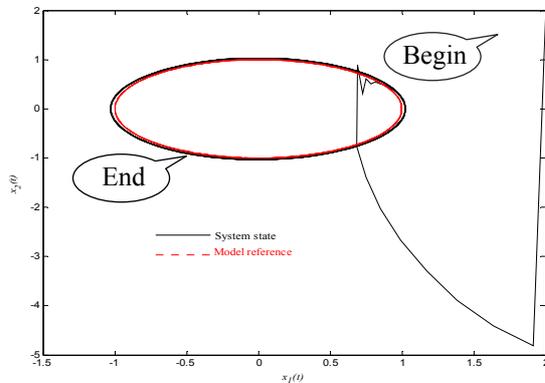
(a) The state x_1 and its desired value.

(b) The state x_2 and its desired value.

(c) Control signal.

(d) Phase plane.

Fig. 7. System states using PDC_RRB.

The simulation results shown in Figs. 7(a), (b), (c) and (d) illustrate that the controller achieve best control performance. From these Figures, we can see that x_1 and x_2 can track x_1^d and x_2^d quickly.

Example 2: The chaotic system considered in this example is the extended Van der Pol oscillator with an additive controller described by the following representation [3]:

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = 0.4(1-x_1)\cdot x_2 - 0.46\cdot x_1 - x_1^3 \\ \quad\quad - 0.1\cdot x_1^5 + f_0\cos(0.86\cdot t) + u, \end{cases} \quad (20)$$

where u is the control signal needed to be chosen. f_0 =4.5 create chaotic behaviour in the dynamical system (20) if no control is applied [42]. Fig. 8 illustrates an example of the trajectory Van de Pol system. The initial conditions were located at the origin. As can be seen, the solutions are chaotic.

The control objective is to have the system states (x_1, x_2) follows a given reference trajectories (x_1^d, x_2^d). Thus, the tracking errors must be as small as possible and the closed-loop system must be globally stable and robust, i.e., all its parameters are uniformly bounded and the effect of the external disturbances is attenuated to a prescribed level.

In this example, we consider again the control of chaotic system (in (20)). Here we use the PDC_RRB to control the extended Van der Pol chaotic oscillator system state $x = (x_1, x_2)^T$ to track the periodic reference trajectory

$$(x_1^d(t), x_2^d(t)) = (\cos(\omega\cdot t), -\omega\cdot\sin(\omega\cdot t)). \quad (21)$$

For simulation, $C^{(l)}(k)\Big|_{i=1,2,3}^{k=1,\dots,max\_t}$ are equal to $\{0.01,\ 0.001,\ 0.01\}$ and $max\_t = 1400$ in (15), respectively. $[K_{SF1}, K_{SF2}] = [5,\ 20]$ and $[-K_{Fg1},$
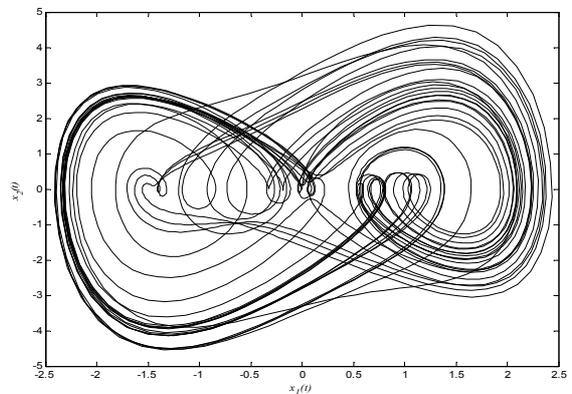

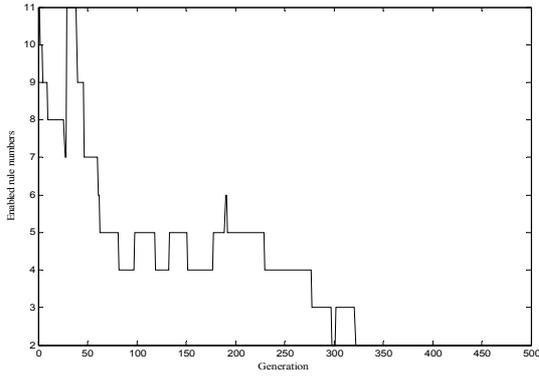Fig. 8. A plot example of the trajectory Van der Pol system.

Fig. 9. Number of enabled rules.

$K_{Fg1}] = [-1000, 1000]$ in Table 1. In this example, we choose the initial system state $x(0) = (0,0)^T$. The controller was applied at $t = 60$ seconds. The enabled rule numbers during the evolution of GA is represented in Fig. 9. Only two rules remain at the end of GA execution, as indicated the expression (22).

$$R^{(17)}: \quad \text{IF } \tilde{x}_1 \text{ is } PM \text{ and } \tilde{x}_2 \text{ is } ZE \text{ THEN}$$
$$u^{(17)} = [263.50 \quad 74.53] \cdot (x - x^d)$$
$$R^{(19)}: \quad \text{IF } \tilde{x}_1 \text{ is } PM \text{ and } \tilde{x}_2 \text{ is } PM \text{ THEN}$$
$$u^{(19)} = [733.97 \quad 209.77] \cdot (x - x^d) \tag{22}$$

Figs. 10(a), (b), (c), (d) and (e) present the simulation results by applying the PDC_RRB controller (7) and (22) to the extended Van der Pol oscillator (20) for tracking the desired periodic reference trajectory (x_1^d, x_2^d). From Figs. 10(a) and (b), it is clearly appears that when $u = 0$ the periodic reference trajectory and the system trajectories diverge. However, as soon as the PDC_RRB controller is started, the controlled trajectories of the chaotic Van der Pol oscillator tend to the periodic reference trajectory and the tracking control problem is achieved as shown by Figs. 10(c) and (d). In order to add evidence of the effectiveness and efficiency of the proposed PDC_RRB controller, Fig. 10(e)
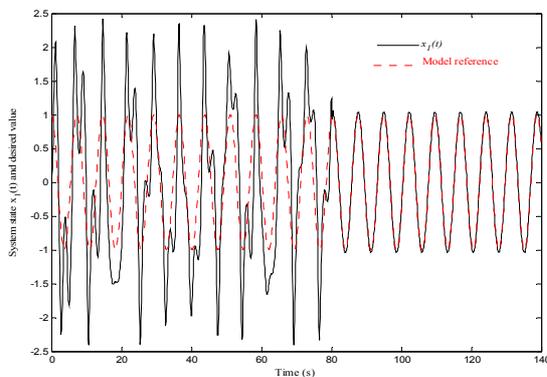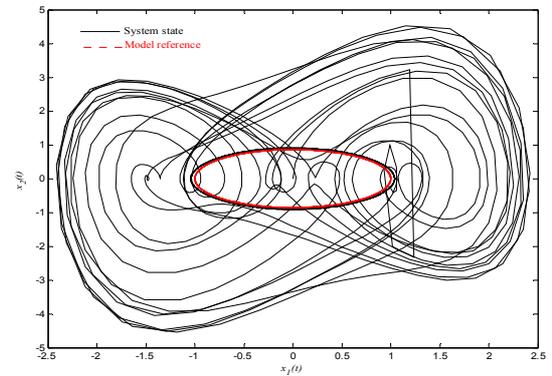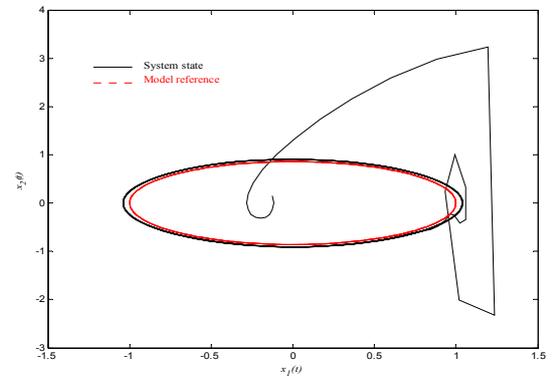

(a) The state x_1 and its desired value.
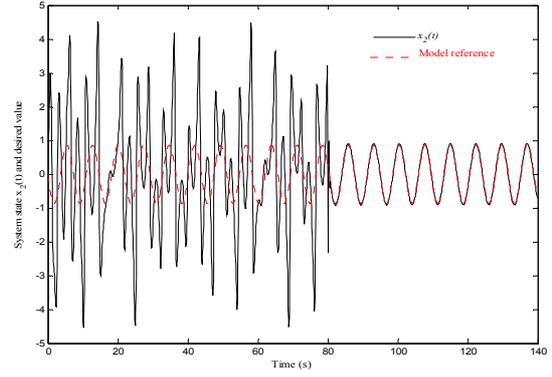
(b) The state x_2 and its desired value.
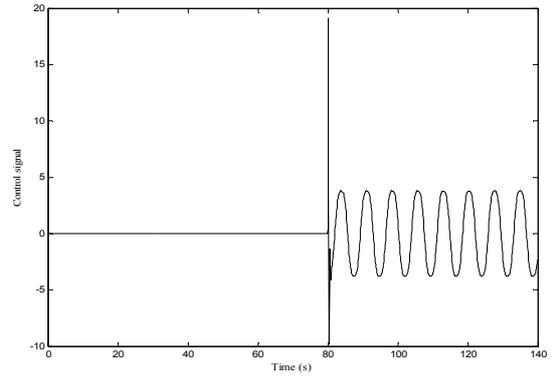
(c) Control signal.

(d) Phase plane.

(e) Phase plane.

Fig. 10. System states using PDC_RRB.

presents x_1 and x_2 in phase space under control actions when the transient oscillatory period has been eliminated. One can see that the attractor changes its dynamical structure in such a way that the canonical plane (x_1, x_2) has a periodic structure.

Example 3: The chaotic *Chua's circuit*, as shown in Fig. 11, is a simple electronic system, which consists of one inductor (L), two capacitor (C_1, C_2), one linear resistor (R) and one piecewise linear or nonlinear resistor (g). It has been shown to possess very rich nonlinear dynamics such as bifurcation and chaos [43]. The dynamic equations of Chua's circuit in dimensionless form is as follows [3]

$$\begin{cases} \dot{x}_1 = \gamma_1 \left(x_2 - x_1 - g(x_1) \right) + u \\ \dot{x}_2 = x_1 - x_2 + x_3 \\ \dot{x}_3 = -\gamma_2 x_2, \end{cases} \quad (23)$$

where u is the control input to be chosen and

$$g(x_1) = \begin{cases} b \cdot x_1 + a - b, & x_1 > 1 \\ a \cdot x_1, & |x_1| > 1 \\ b \cdot x_1 - a + b, & x_1 < -1, \end{cases} \quad (24)$$

where $x_1 = v_{C_1}/E$, $x_2 = v_{C_2}/E$, $x_3 = i_L/ER$, and E is a constant voltage. $\gamma_1 = C_2/C_1$, $\gamma_2 = C_2 R^2 / L$. $a = RG_a$ and $b = RG_b$. In the model equation, the state variables x_1 and x_2 represent the voltage across the two capacitors, and variable x_3 is the current through the inductor. Typical values of the system parameters $\gamma_1 = 9$, $\gamma_2 = 100/7$, $a = -8/7$ and $b = -5/7$ create chaotic behaviour in the dynamical systems (23) when $u = 0$ as indicate clearly Figs. 12(a), (b) and (c). We have selected the initial conditions as $(x_1(0), x_2(0), x_3(0)) = (1.0, 0.0, 0.0)$.

In this case, we consider again the control of Chua's system (23). Here we use the PDC_RRB to control the Chua's system state $x = (x_1, x_2, x_3)^T$ at the origin. A hierarchical PDC_RRB is adopted in this case (Fig. 13). This structure provides flexible architecture for modeling nonlinear systems and reduces the size of
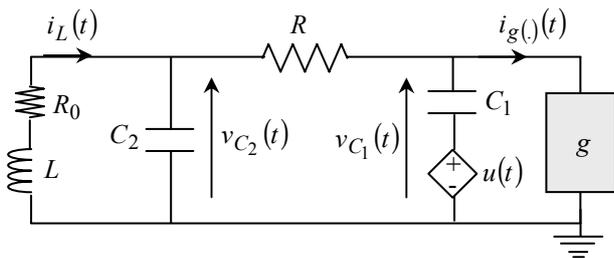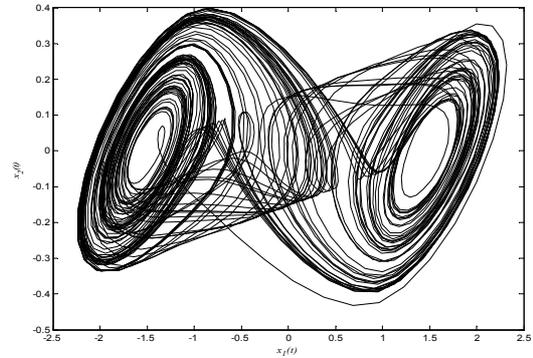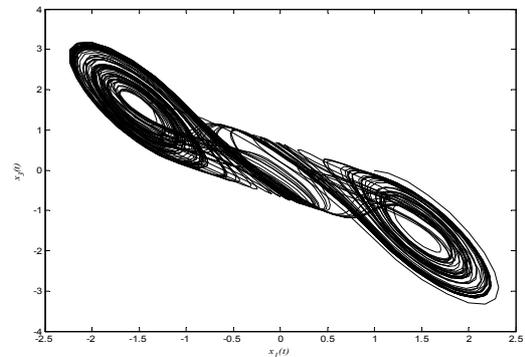
rule base to some extend. The problems in designing of hierarchical structure include [21]:
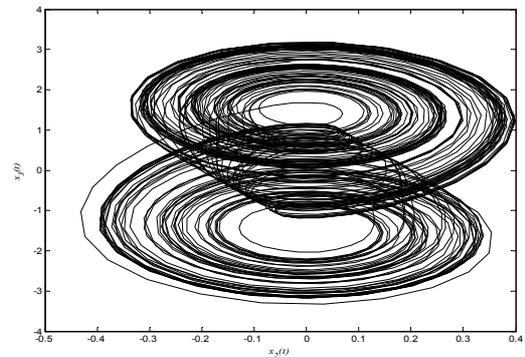- Select a proper hierarchical structure.
- Select the inputs for each Ts-fuzzy sub-model.
- Determine the initial configuration for each Ts-fuzzy sub-model.

(a)

(b)

(c)

Fig. 12. Phase plane of the uncontrolled Chua's oscillator.
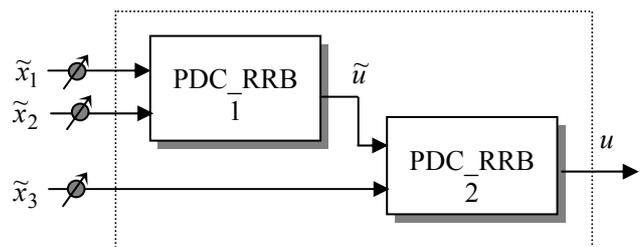

Fig. 11. Diagram of Chua's circuit.

Fig. 13. Hierarchical structure of the controller.

- Optimize the fuzzy knowledge base by using the genetic learning.

In Fig. 13, $\tilde{x}_i = x_i - x_i^d$, $(i = 1, 2, 3)$ and

$$\tilde{u} = -\underline{\Psi}_{11}^{T}(\tilde{x}_1, \tilde{x}_2) \cdot \tilde{x}_1 - \underline{\Psi}_{12}^{T}(\tilde{x}_1, \tilde{x}_2) \cdot \tilde{x}_2. \qquad (25)$$

The controller output is given by

$$\begin{aligned} u &= -\underline{\Psi}_{21}^{T}(\tilde{u}, \tilde{x}_3) \cdot \tilde{u} - \underline{\Psi}_{22}^{T}(\tilde{u}, \tilde{x}_3) \cdot \tilde{x}_3 \\ &= \Phi_1(\cdot) \cdot \tilde{x}_1 + \Phi_2(\cdot) \cdot \tilde{x}_2 + \Phi_3(\cdot) \cdot \tilde{x}_3, \end{aligned} \qquad (26)$$

$$\begin{cases} \Phi_1(\cdot) = \underline{\Psi}_{21}^{T}(\tilde{u}, \tilde{x}_3) \cdot \underline{\Psi}_{11}^{T}(\tilde{x}_1, \tilde{x}_2) \\ \Phi_2(\cdot) = \underline{\Psi}_{21}^{T}(\tilde{u}, \tilde{x}_3) \cdot \underline{\Psi}_{12}^{T}(\tilde{x}_1, \tilde{x}_2) \\ \Phi_1(\cdot) = -\underline{\Psi}_{22}^{T}(\tilde{u}, \tilde{x}_3). \end{cases} \qquad (27)$$

For simulation, $C^{(i)}(k)\Big|_{i=1,2,3}^{k=1,...,max\_t}$ and max_t in expression (15) are equal to $\{0.01, \quad 0.0001, \quad 0.001\}$ and 600, respectively. $[K_{SFi}]\Big|_{i=1}^{i=3} \in [5, 15]$, $[-K_{Fg\tilde{u}}, K_{Fg\tilde{u}}] \in [-1.0, 1.0]$ and $[-K_{Fgu}, K_{Fgu}] \in [-100.0, 100.0]$ in Table 1. The controller was applied at $t = 100$ seconds. The enabled rule numbers during the evolution of GA is represented in Fig. 14.

Only two rules remain at the end of GA execution. The rules base of PDC_RRB$_1$ and PDC_RRB$_2$ obtained in the last generation are given by

$R^{(13)}$: IF $\tilde{x}_1$ is ZE and $\tilde{x}_2$ is ZE THEN

$$\tilde{u}^{(13)} = [-0.59 \quad 0.36] \cdot (x - x^d),$$

$R^{(14)}$: IF $\tilde{x}_1$ is ZE and $\tilde{x}_2$ is PM THEN

$$\tilde{u}^{(14)} = [0.45 \quad -0.52] \cdot (x - x^d),$$

$R^{(18)}$: IF $\tilde{x}_1$ is PM and $\tilde{x}_2$ is ZE THEN   (28)

$$\tilde{u}^{(18)} = [0.74 \quad 0.57] \cdot (x - x^d),$$

$R^{(19)}$: IF $\tilde{x}_1$ is PM and $\tilde{x}_2$ is PM THEN
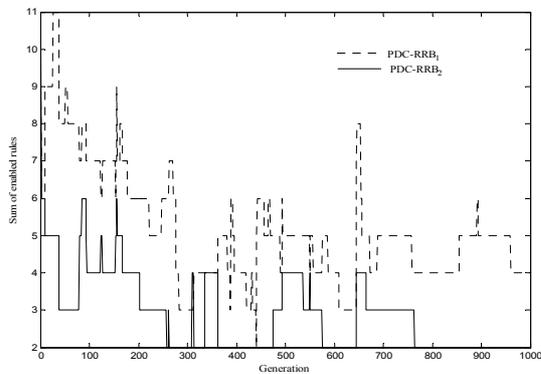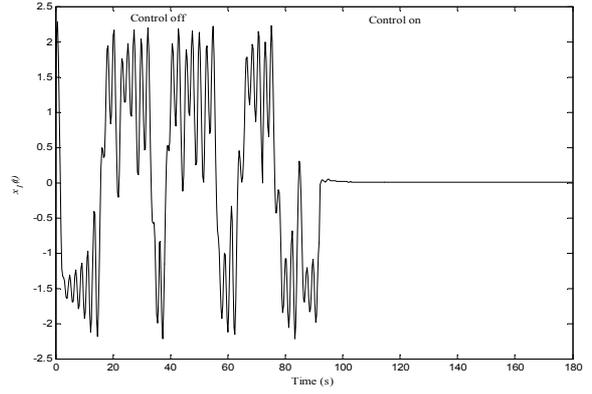
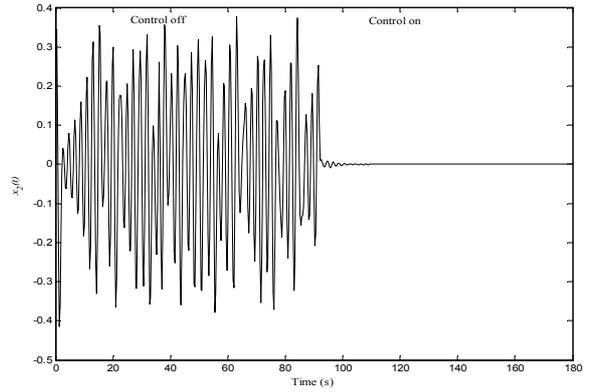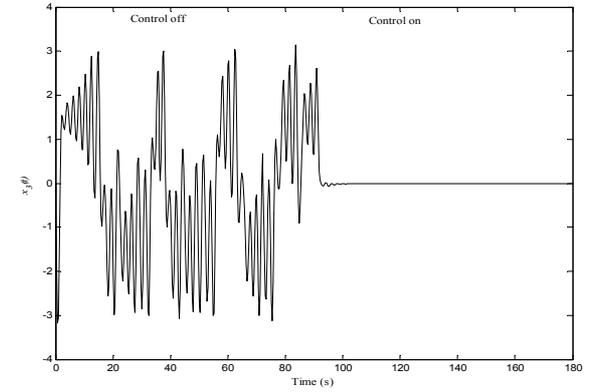$$\tilde{u}^{(19)} = [0.18 \quad 0.34] \cdot (x - x^d),$$

Fig. 14. Number of enabled rules.
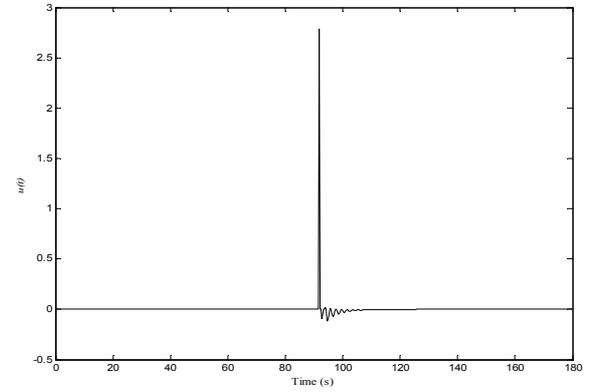

(a) The state x_1.

(b) The state x_2.

(c) The state x_3.

(d) Control signal.

Fig. 15. System states using PDC_RRB.

$R^{(13)}$: IF $\tilde{u}$ is ZE and $\tilde{x}_3$ is *ZE* THEN

$$u^{(13)} = [\text{-}26.65 \quad \text{-}97.10] \cdot (x - x^d),$$

$R^{(25)}$: IF $\tilde{u}$ is *PB* and $\tilde{x}_3$ is *PB* THEN

$$u^{(19)} = [\text{-}95.72 \quad 6.28] \cdot (x - x^d). \tag{29}$$

Figs. 15(a), (b), and (c) show the individual evolution of the states x_1, x_2 and x_3, respectively. The states of the Chua's circuit have been regulated effectively and efficiency to the equilibrium point $(x_1^d, x_2^d, x_3^d) = (0.0, 0.0, 0.0)$ and the control objective is attained. The corresponding input signal of the system is depicted by Fig. 15(d).

The real-coded GA is robust, accurate and efficient because the floating point representation is conceptually closest to the real design space and moreover, the sting length is reduced to the number of design variables.

The chaotic mutation operator is used to solve the problem of maintaining the population diversity of GA in the learning process. The nonlinear optimal controller designed (via fuzzy topology and the genetic optimization) in this paper is very simple and contains a minimal number of rules.

5. CONCLUSION

This paper contributes a new alternative for the synthesis the fuzzy optimal controller with reduced rule base. The genetic learning algorithm with chaotic mutation is proposed for constructing a robust nonlinear optimal controller. The developed controller design techniques have been applied to the control of chaotic systems. The chaotic mutation is introduced for maintaining the population diversity during the evolution process of the genetic algorithm. Simulation results are provided to show the effectiveness of the proposed methodology. Based on the simulation results, the following main conclusions can be stated about the proposed PDC_RRB:

- The design procedure is conceptually simple and computationally efficient.
- Exploit the fine abilities and advantages of the fuzzy logic, chaotic phenomenon and genetic algorithms.

The advantages of the proposed designing methodologies are that it reduces the number of rules and the computational time maintaining almost the same level of desired performances.

REFERENCES

[1] Y. Ishikawa, Y. Masukake, and Y. Ishida, "Control of chaotic dynamical systems using RBF networks," *International Journal of Applied Science, Engineering and Technology*, vol. 4, no. 1, pp. 175-178, 2007.

[2] H. O. Wang, K. Tanaka, and T. Ikeda, "Fuzzy modelling and control of chaotic systems," *Proc. of IEEE International Symposium on Circuits and Systems*, 12-15 May 1996, Atlanta, GA, USA, vol. 3, pp. 209-212, 1996.

[3] S. Bowong and A. Temgoua Kagou, "Adaptive control for linearizable chaotic systems," *Journal of Vibration and Control*, vol. 12, no. 2, pp. 119-137, 2006.

[4] K. Tanaka, T. Ikeda, and H. O. Wang, "A unified approach to controlling chaos via an LMI-based fuzzy control system design," *IEEE Trans. on Circuits and Systems-I: Fundamental Theory and Applications*, vol. 45, no. 10, pp. 1021-1040, October 1998.

[5] Y.-H. Joo, L.-S. Shieh, and G. Chen, "Hybrid state-space fuzzy model-based controller with dual-rate sampling for digital control of chaotic systems," *IEEE Trans. on Fuzzy Systems*, vol. 7, no. 4, pp. 394-408, August 1999.

[6] J. Wang, X. Ye, and W. Si, "Robust ISS-satisfying fuzzy control of chaotic systems," *Proc. of the Sixth International Conference on Intelligent Systems Design and Applications*, 2006.

[7] W. Tan, Y. Wang, S. Zhou, and Z. Liu, "Robust adaptive dynamical neural control for uncertain chaotic system," *Proc. of the 5th World Congress on Intelligent Control and Automation*, Hangzhou, P. R. China, pp. 1300-1303, June 15-19, 2004.

[8] C.-F. Hsu, C.-M. Lin, and T.-T. Lee, "Wavelet adaptive backstepping control for a class of nonlinear systems," *IEEE Trans. on Neural Networks*, vol. 17, no. 8, pp. 1175-1183, September 2006.

[9] C.-T. Lin and C.-P. Jou, "Controlling chaos by GA-based reinforcement learning neural network," *IEEE Trans. on Neural Networks*, vol. 10, no. 4, pp. 846-859, July 1999.

[10] M. J. Er and D. H. Lin, "A new approach for stabilizing nonlinear systems with time delays," *International Journal of Intelligent Systems*, vol. 17, pp. 289-302, 2002.

[11] X. Zhang, Z. Cheng, and Q. Liu, "A fuzzy logic approach to optimal control of nonlinear time-delay systems," *Proc. of the 5th World Congress on Intelligent Control and Automation*, Hangzhou, P. R. China, pp. 902-906, June 15-19, 2004.

[12] K.-B. Sim, K.-S. Byun, and D.-W. Lee, "Design a fuzzy controller using schema coevolutionary algorithm," *IEEE Trans. on Fuzzy Systems*, vol. 12, no. 4, pp. 565-568, August 2004.

[13] D. E. Goldberg, *Algorithmes Génétiques: Exploration, Optimisation et Apprentissage*

Automatique, Addison-Wesley, 1994.

[14] Y. Park, M.-J. Tahk, and H. Bang, "Design and analysis of optimal controller for fuzzy systems with input constraint," *IEEE Trans. on Fuzzy Systems*, vol. 12, no. 6, pp. 766-779, December 2004.

[15] B.-D. Liu, C.-Y. Chen and J.-Y. Tsao, "Design of adaptive fuzzy logic controller based on Linguistic-Hedge concepts and genetic algorithms," *IEEE Trans. on Systems, Man, and Cybernetics-Part B: Cybernetics*, vol. 31, no. 1, pp. 32-53, February 2001.

[16] G.-Y. Tang, Y.-D. Zhao, and B.-L. Zhang, "Optimal output tracking control for nonlinear systems via successive approximation approach," *Nonlinear Analysis*, vol. 66, pp. 1365-1377, 2007.

[17] 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.

[18] K. Belarbi, F. Titel, W. Bourebia, and K. Benmahammed, "Design of Mamdani fuzzy logic controllers with rule base minimisation using genetic algorithm," *Engineering Applications of Artificial Intelligence*, vol. 18, no. 7, pp. 875-880, October 2005.

[19] C.-H. Lee, M. Yuchi and J.-H. Kim, "Two phase optimization of fuzzy controller by evolutionary programming," *The Congress on Evolutionary Computation*, vol. 3, pp. 1949-1956, 8-12 Dec. 2003.

[20] R. Alcaléa, J. Alcaléa-Fdez, J. Casillas, O. Cordéon, and F. Herrera, "Hybrid learning models to get the interpretability–accuracy trade-off in fuzzy modelling," *Soft Computing*, vol. 10, pp. 717-734, 2006.

[21] Y. Chen, J. Dong, and B. Yang, "Automatic design of hierarchical TS-FS model using ant programming and PSO algorithm," *Proc. of the Eleventh International Conference on Artificial Intelligence: Methodology, Systems, and Applications*, Varna, Bulgaria, vol. 3192, pp. 285-294, September 2-4, 2004.

[22] H. Wang, S. Kwong, Y. Jin, W. Wei, and K. F. Man, "Multi-objective hierarchical genetic algorithm for interpretable fuzzy rule-based knowledge extraction," *Fuzzy Sets and Systems*, vol. 149, pp. 149-186, 2005.

[23] N. Xiong and L. Litz, "Reduction of fuzzy control rules by means of premise learning-method and case study," *Fuzzy Sets and Systems*, vol. 132, pp. 217-231, 2002.

[24] F. Wan, H. Shang, L.-X. Wang, and Y.-X. Sun, "How to determine the minimum number of fuzzy rules to achieve given accuracy: A computational geometric approach to SISO case," *Fuzzy Sets and Systems*, vol. 150, no. 2,

pp. 199-209, March 2004.

[25] K. Tanaka, T. Taniguchi, and H. O. Wang, "Generalized Takagi-Sugeno fuzzy systems: Rule reduction and robust control," *Proc. of the Ninth IEEE International Conference on Fuzzy Systems*, vol. 2, pp. 688-693, May 7-10, 2000.

[26] T. Taniguchi, K. Tanaka, H. Ohtake, and H. O. Wang, "Model construction, rule reduction, and robust compensation for generalized form of Takagi–Sugeno fuzzy systems," *IEEE Trans. on Fuzzy Systems*, vol. 9, no. 4, pp. 525-538, August 2001.

[27] M. G. Joo and J. S. Lee, "A class of hierarchical fuzzy systems with constraints on the fuzzy rules," *IEEE Trans. on Fuzzy Systems*, vol. 13, no. 2, pp. 194-203, April 2005.

[28] G. V. S. Raju and J. Zhou, "Adaptive hierarchical fuzzy controller," *IEEE Trans. on Systems, Man, and Cybernetics*, vol. 23, no. 4, pp. 973-980, July/August 1993.

[29] R. L. Haupt and S. E. Haupt, *Practical Genetic Algorithms*, Second Edition, John Wiley & Sons, In., Publication, 2004.

[30] Q. Lü, G. Shen, and R. Yu, "A chaotic approach to maintain the population diversity of genetic algorithm in network training," *Computational Biology and Chemistry*, vol. 27, pp. 363-371, 2003.

[31] Y. Yong, S. Wanxing, and W. Sunam, "Study of chaos genetic algorithms and its application in neural networks," *Proc. of IEEE Region 10 Conference on Computers, Communications, Control and Power Engineering*, vol. 1, pp. 232-235, 2002.

[32] X. F. Yan, D. Z. Chen, and S. X. Hu, "Chaos-genetic algorithms for optimizing the operating conditions based on RBF-PLS model," *Computer and Chemical Engineering*, vol. 27, pp. 1393-1404, 2003.

[33] S. K. Sharma and G. W. Irwin, "Fuzzy coding of genetic algorithms," *IEEE Trans. on Evolutionary Computation*, vol. 7, no. 4, pp. 344-355, August 2003.

[34] K. L. Lo and M. O. Sadegh, "Systematic method for the design of a full–scale fuzzy PID controller for SVC to control power system stability," *IEE Proceedings Generation Transmission and Distribution*, vol. 150, no. 3, pp. 297-304, May 2003.

[35] Z.-H. Cui, J.-C. Zeng, and Y.-B. Xu, "Dynamic circle nonlinear genetic algorithm," *Proc. of the Second International Conference on Machine Learning and Cybernetics*, pp. 1836-1840, 2-5 November 2003.

[36] R. May, "Simple mathematical models with very complicated dynamics," *Nature*, vol. 261, pp. 459-467, 1976.

[37] D. B. Fogel, *Evolutionary Computation: Towards New Philosophy of Machine Intelligence*, IEEE Press, Piscataway, N.J., 1995.

[38] L. Chambers, *The Practical Handbook of Genetic Algorithms: Applications*, Second Edition, Chapman & Hall/CRC, 2001.

[39] Z. Michalewics, *Genetic Algorithms + Data Structures = Evolution Programs*, 3rd Edition, Springer Verlag, 1999.

[40] W. Jiang, Q.-G. Dong, and D. Bin, "Observer-based robust adaptive variable universe fuzzy control for chaotic system," *Chaos, Solitons and Fractals*, vol. 23, pp. 1013-1032, 2005.

[41] L.-X. Wang, "Stable adaptive fuzzy control of nonlinear systems," *IEEE Trans. on Fuzzy Systems*, vol. 1, no. 2, pp. 146-155, May 1993.

[42] F. Moukam, S. Bowong, C. Tchawoua, and E. Kaptouom, "Chaos control and synchronization of a Φ^6 Van der Pol oscillator," *Physics Letters*, A 322, pp. 305-323, 2004.

[43] Y.-H. Joo, L.-S. Shieh, and G. Chen, "Hybrid state-space fuzzy model-based controller with dual-rate sampling for digital control of chaotic systems," *IEEE Trans. on Fuzzy Systems*, vol. 7, no. 4, pp. 394-408, August 1999.

Abdelhafid Khellaf was borne in Algeria. He received the DEA degree in Informatics and Random Systems in 1975 and the 3rd Cycle Doctorate in 1979 in Electronics both from Rennes University, France. He received state Doctor in Electronics from the University of Constantine, Algeria in 1990. Currently, he is a Professor at the University of Setif, Algeria. His research interest is concerned with the automation system design and biotechnological process.

Salah Leulmi received the Ms. of Eng. from RPI in EPS Eng. & a PhD in EE from ISU, USA. He is the author of more than 30 publications in journals & proceedings. He was the Head of the University of Skikda, Algeria. Currently, he is a Professor & President of the scientific counsel of the Engineering Faculty at the same University. He is also a referee of four Algerian Journals & a referee of one overseas society 'WSEAS' for proceedings and journals.

Mourad Grimes was born in Algeria on Mars 31, 1979. He received the Ing. degree and Master degree in electronic from Jijel University. He is currently working on a Ph.D. Since 2008, he has been an Assistant Professor at the Electronics department, University of Jijel, Algeria. His current research interests include nondestructive testing and numerical techniques.

Soukkou Ammar was born in Djimla-Jijel, Algeria. He received his Diploma in Engineering from the Electronics department, University of Sétif, Algeria in 1996, the Diplôme d'Etudes Approfondies in Automatic Control Electronics department, University of Sétif in 1997, the Magister degree from the Laboratoire de l'intelligence artificielle et de l'Automatique from the Electronics department, University of Sétif in 1999. Since 2000 to 2005, he held different positions involved in industrial field and education. He is currently working towards a Ph.D. in Electrical Engineering in the Sétif University. Since 2005, he has been an Assistant Professor at the Electronics Department, University of Jijel, Algeria. His current interests include optimal and robust control, nonlinear systems, fuzzy logic, neural networks and evolutionary algorithms.