Research Journal of Recent Sciences ISSN 2277-2502
Vol. **2(10),** 81-87, October **(2013)** *Res.J.Recent Sci.*

Tuning Fuzzy Control Rules via Genetic Algorithms: An Experimental Evaluation

Pitalúa Díaz N.[1], Lagunas Jiménez R.[2] and González Angelesa[3]
[1]Departamento de Ingeniería Industrial-Programa ingenieríamecatrónica, Universidad de Sonora, Unidad Regional Centro, Hermosillo, Sonora, MÉXICO
[2]Facultad De Ingeniería Universidad Autónoma De Campeche, Campeche, Campeche, MÉXICO
[3]Facultad de Ingeniería Campus Mexicali, Universidad Autónoma de baja California, Mexicali, Baja California, MÉXICO

Available online at: www.isca.in, www.isca.me
Received 22nd April 2013, revised 26th May 2013, accepted 28th June 2013

Abstract

In this article, a simple genetic algorithm is used to solve an optimization problem that minimizes an objective function, with the purpose of obtaining the rule base of a fuzzy controller. The plant under consideration is an experimental direct current servosystem. The variable to be controlled is the load angular position. The objective function used here is the criterion IAE (Integrated Absolute Error). The genetic algorithm operates with populations who respond to consequents of the rule base.

Keywords: Feedback systems, fuzzy control, genetic algorithms, linear systems.

Introduction

The fuzzy control incorporates expert knowledge in its design[1-3], however, the methods based on optimization can be used to tune fuzzy controllers. It is known that genetic algorithms are an important alternative for solving optimization problems[4]. Karr was one of the first researchers to use genetic algorithms to tune fuzzy systems. Genetic algorithms (GAs), are computational strategies which use operators based on the processes of natural evolution, who realize a process of heuristic search in a search space, where it is presupposed that the ideal solution for the optimization problem is there. The works of Zhang, Lagunass how that a fuzzy controller can be designed partly or in its totality, considering that it is possible to find an ideal controller, which can be seen as an optimization problem[5-7]. The methods of tuning fuzzy controllers, seen as an optimization problem, can be applied in the following cases: Tuning the membership functions or the rule base[8,9]. In this research, a simple genetic algorithm is used to obtain the rules of fuzzy inference system. One of the main purposes of our research is to compare the performance of a fuzzy controller, based on heuristics, with a controller designed with the optimization method. One of the advantages of using this method is that it is only needed evaluating a single objective. A sufficiently good solution of the optimization problem is found, applying genetic operators upon the individuals of the population, in feasible search space, and by means of certain numbers of generations. The achieved solution could possibly be the optimal solution. In sum, to optimize the inference rules of a fuzzy controller is to find out the best combination between the fuzzy input-output variables, for a certain range of the fuzzy logic controller operation (FLC).

He proposes an off-line approach for tuning the membership functions of a fuzzy system; nevertheless he did not include the rule base in the proposed procedure. From Karr's point of view, the rule base obtained empirically by human intelligence can cover a wider range of operation. In 2006, Zhang and Li designed a fuzzy controller using genetic algorithms, to control the temperature of a kiln. They used a genetic algorithm to obtain the rules of the fuzzy controller.

For effects of this investigation, a two dimension fuzzy controller is optimized in order to control the angular position of a direct current servomechanism. The performance of the controller designed is presented through the MATLAB®simulations, and also by the experimental results, achieved from a real plant.

Problem Statement

Consider the feedback control scheme shown in the figure 1, where: r denotes the reference input signal, u denotes the control signal, e denotes the error signal, y denotes the output signal, $G(s)$ denotes a Linear Time-Invariant (LTI) Single-Input Single-Output (SISO) plant or single variable system[10].

The control problem is then defined as follows:

Definition 1. Optimal Tracking Problem Control (OTPC). Taking the tracking control scheme shows in figure-1, find a fuzzy controller (FC), which minimizes the tracking error signal e for a specific reference r.

The model of the experimental servomotor was obtained from the data sheet of the manufacturer, who is presented in Theequation 1.

$$G(s)=\frac{65}{S(S+31)} \tag{1}$$

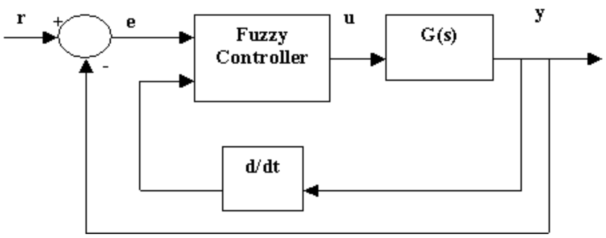

Figure-1
Tracking control scheme

The figure 2 shows a photograph of the direct current servo motor that was used, as an experimental plant to evaluate the fuzzy controller.

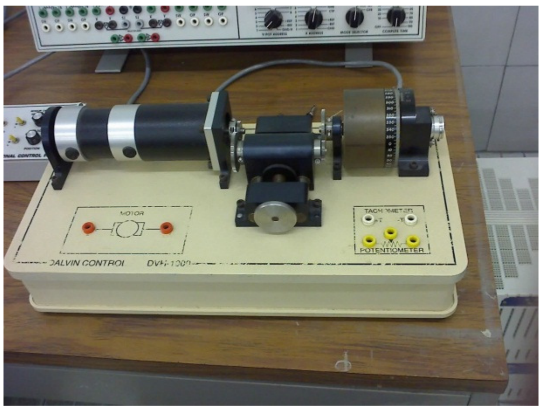

Figure-2
The direct current servosystem

Optimization Method

Nevertheless that the mathematical model of this process is known, it is not necessary to design the fuzzy controller. In some cases without high accuracy the designer can adopt flat fuzzy control, which in this case, the linguistic variables: error (e), and the error change (ec)are proposed. They take seven linguistic values, {Negative Big (NB), Negative Middle (NM), Negative Small (NS), Zero (ZO), Positive Small (PS), Positive Middle (PM), Positive Big (PB)}. The output variable u (the control value), which also have seven language values {Negative Big (NB), Negative Middle (NM), Negative Small (NS), Zero (ZO), Positive Small (PS), Positive Middle (PM), Positive Big (PB)}. The fuzzy controller was designed with the fuzzy sets presented in the figures -6, 7 and 8, based on the defined linguistic values.

The objective function proposed in the optimization problem is the criterion *IAE* (Integrated Absolute Error), which is defined by

$$J_1=\int_0^{\infty}\left|e^{(t)}\right|\,dt \qquad (2)$$

The objective function proposed in (2), which is minimized during the optimization process, considers a good response to setpoint changes, this objective function is considered among the most important points: the raise time, the settling time, the decay ratio, the overshoot and the steady-state error.

Consequence 1 (NM)	Consequence 2 (PB)	Consequence 49 (PB)
010	111	111

Figure-3
Codified chromosome

In this investigation only the rule base is obtained by means of the optimization method; the membership functions of the input-output variables are proposed based on the author experience. To obtain the rules, the genetic algorithm works with a population of individuals (chromosome), where each of them contains a particular fuzzy controller. The antecedents, which correspond to the input linguistic variables, are fixed in an array as a part of a MATLAB function. This function is used by the genetic algorithm during the evaluation process, and it receives as arguments, the consequents of a controller; the evaluation is made with each of the individuals of the population, in all the pre-established generations.

For simplicity, the binary code is adopted. The values {Negative Big (NB), Negative Middle (NM), Negative Small (NS), Zero (ZO), Positive Small (PS), Positive Middle (PM), Positive Big (PB)} of de output language variable are coded in order to be 001, 010, 011, 100, 101, 110, and 111. Table -1 shows a probable chromosome, which would be codified as: {100 110 111 111 101 101 101 101……001 001 100}, since it appears in the shaded positions of the table-1.

Table-1
Fuzzy control rule base

ec / e	**PB**	**PM**	**PS**	**ZO**	**NS**	**NM**	**NB**
PB	ZO	PS	ZO	NS	ZE	ZE	NM
PM	PM	PB	PM	ZO	PS	ZE	NS
PS	PB	PB	PB	ZO	PS	NS	NS
ZO	PB	PB	PS	ZO	NS	NS	NB
NS	PS	ZE	ZE	PS	NS	NS	NB
NM	PM	ZE	ZE	PS	NM	NB	NB
NB	PM	ZE	ZE	PN	NG	NG	ZO

Every chromosome has 149 binary numbers, which correspond to the 49 consequents of a particular fuzzy controller. The genetic algorithm operates with a population of individuals, in whom each one of these (chromosome), contains the consequents of the fuzzy rule base. The figure-3 shows an example of a possible codified chromosome.

During the evaluation of each of the individuals, the fitness value of each one of them, is obtained minimizing the criterion IAE (Objective function), this value is obtained due to a MATLAB simulation. The figure-4 presents the tuning procedure.

Genetic Process: The genetic process initiates with the creation of a population of individuals (created randomly), where every individual contains the 49 consequents of a fuzzy controller in particular. However so that the 49 consequents could be used in the MATLAB® function, to form the rules, is necessary to change the individual, codified as a binary chain, to entire numbers, in the values from 1 to 7, where: Negative Big (BN)=1, Negative Middle (NM)=2, Negative Small (NS)=3, Zero (ZO)=4, Positive Small (PS)=5, Positive Middle (PM)= 6 and Positive Big (PB)= 7. In the array presented by The Matrix 3, the 49 consequents are represented by c_n.

$$\begin{bmatrix} c_1 & c_2 & c_3 & c_4 & c_5 & c_6 & c_7 \\ c_8 & c_9 & c_{10} & c_{11} & c_{12} & c_{13} & c_{14} \\ c_{15} & c_{16} & c_{17} & c_{18} & c_{19} & c_{20} & c_{21} \\ c_{22} & c_{23} & c_{24} & c_{25} & c_{26} & c_{27} & c_{28} \\ c_{29} & c_{30} & c_{31} & c_{32} & c_{33} & c_{34} & c_{35} \\ c_{36} & c_{37} & c_{38} & c_{39} & c_{40} & c_{41} & c_{42} \\ c_{43} & c_{44} & c_{45} & c_{46} & c_{47} & c_{48} & c_{49} \end{bmatrix} \tag{3}$$

To form the rule base of a fuzzy controller, is necessary to know the values of the precedents, fuzzy operators in the fuzzy rules and rule base weights; in this case, the antecedents 1 and 2 and the other mentioned values, are fixed in a matrix inside the MATLAB® function, as can be seen in The Matrix 4.

$$\begin{bmatrix} a1_1a2_1c_1wo & a1_2a2_2c_2wo & \cdots & a1_7a2_7c_7wo \\ a1_8a2_8c_8wo & a1_9a2_9c_9wo & \cdots & a1_{14}a2_{14}c_{14}wo \\ a1_{15}a2_{15}c_{15}wo & a1_{16}a2_{16}c_{16}wo & \cdots & a1_{21}a2_{21}c_{21}wo \\ \vdots & \vdots & \ddots & \vdots \\ a1_{43}a2_{43}c_{43}wo & a1_{44}a2_{44}c_{44}wo & \cdots & a1_{49}a2_{49}c_{49}wo \end{bmatrix} \tag{4}$$

where: *a1* denotes the Antecedent 1, *a2*: denotes the Antecedent 2, *w* denotes the rules weights, *o* denotes the fuzzy operators in the fuzzy rules, $c_n$denotes the Consequence *n*.

The linguistic values of the antecedents correspond to the following entire values: Negative Big (BN)=1, Negative Middle (NM)=2, Negative Small (NS)=3, Zero (ZO)=4, Positive Small (PS)=5, Positive Middle (PM)= 6, Positive Big (PB)= 7.

After the Matrix 4 is completed, the MATLAB® function: *a=addrule (a,ruleList)*, generates the rules of the fuzzy controller[11,12].

Where rule List, correspond to The Matrix 4. Finally, in the same MATLAB® function are the membership function data and the input/output variables range. The output of this function is a MATLAB® FIS (*Fuzzy Inference System*) file, which corresponds to a structure where all the information of fuzzy inference of the system is included, which is used as a fuzzy controller into the feedback scheme of the SIMULINK library.

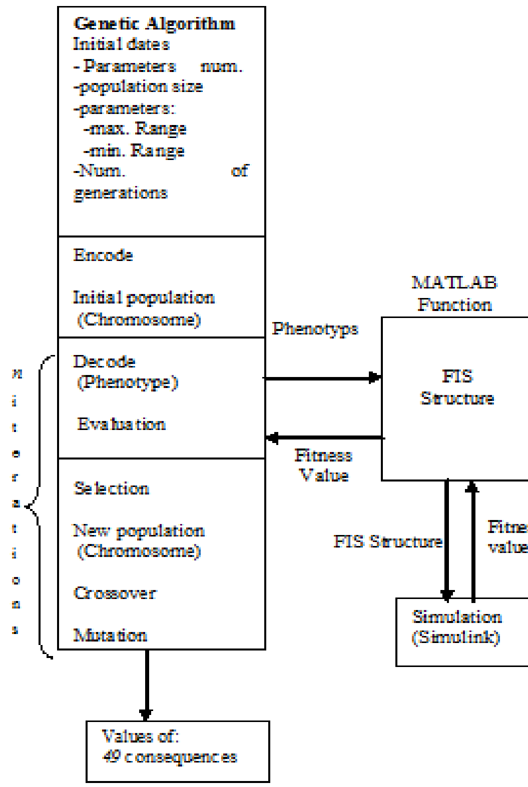

Figure-4
Tuningprocedure

The performance index is taken out from the feedback scheme and returned to the genetic algorithm for the continuity of the genetic process. For a better understanding, the finished process appears in figure -4. After applying the described procedure in this chapter, the obtained rule base is showed in table -3.

Fuzzy Controller

We use a general fuzzy controller based on Mamdani's fuzzy technique[13-15]. For position fuzzy control we use blocksets from the MATLAB-SIMULINK library, as we can see in the figure-5.

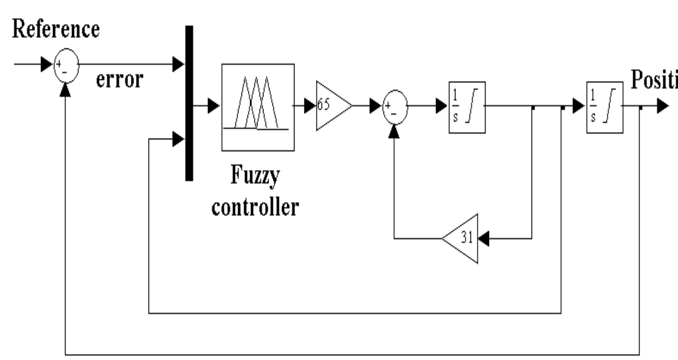

Figure-5
Position fuzzy control

Using the genetic algorithm from section 3, we have the following rule base:

Table-2
Genetic algorithmfollowing basic rules

Number	Fuzzy rules
1	If (Position-error is PG) and (Position-error-change is NG) then (control-output is PP)
2	If (Position-error is PP) and (Position-error-change is PM) then (control-output is PG)
3	If (Position-error is PG) and (Position-error-change is PM) then (control-output is PG)
4	If (Position-error is PG) and (Position-error-change is PG) then (control-output is PG)
5	If (Position-error is PG) and (Position-error-change is NP) then (control-output is PP)
6	If (Position-error is PG) and (Position-error-change is NM) then (control-output is NG)
7	If (Position-error is PG) and (Position-error-change is NG) then (control-output is PM)
8	If (Position-error is PM) and (Position-error-change is CE) then (control-output is PP)
9	If (Position-error is PM) and (Position-error-change is PP) then (control-output is PG)
10	If (Position-error is PM) and (Position-error-change is PM) then (control-output is NG)
11	If (Position-error is PM) and (Position-error-change is PG) then (control-output is PP)
12	If (Position-error is PM) and (Position-error-change is NP) then (control-output is PG)
13	If (Position-error is PM) and (Position-error-change is NM) then (control-output is PG)
14	If (Position-error is PM) and (Position-error-change is NG) then (control-output is PG)
15	If (Position-error is PP) and (Position-error-change is NG) then (control-output is PM)
16	If (Position-error is PP) and (Position-error-change is NM) then (control-output is PG)
17	If (Position-error is PP) and (Position-error-change is NP) then (control-output is NG)
18	If (Position-error is PP) and (Position-error-change is CE) then (control-output is PG)
19	If (Position-error is PP) and (Position-error-change is PP) then (control-output is PG)
20	If (Position-error is PP) and (Position-error-change is PM) then (control-output is PG)
21	If (Position-error is PP) and (Position-error-change is PG) then (control-output is NG)
22	If (Position-error is CE) and (Position-error-change is NG) then (control-output is PP)
23	If (Position-error is CE) and (Position-error-change is NM) then (control-output is PM)
24	If (Position-error is CE) and (Position-error-change is NP) then (control-output is PG)
25	If (Position-error is CE) and (Position-error-change is CE) then (control-output is NG)
26	If (Position-error is CE) and (Position-error-change is PP) then (control-output is NG)
27	If (Position-error is CE) and (Position-error-change is PM) then (control-output is NG)
28	If (Position-error is CE) and (Position-error-change is PG) then (control-output is CE)
29	If (Position-error is NP) and (Position-error-change is NG) then (control-output is CE)
30	If (Position-error is NP) and (Position-error-change is NM) then (control-output is NG)
31	If (Position-error is NP) and (Position-error-change is NP) then (control-output is NM)
32	If (Position-error is NP) and (Position-error-change is CE) then (control-output is NP)
33	If (Position-error is NP) and (Position-error-change is PP) then (control-output is NM)
34	If (Position-error is NP) and (Position-error-change is PM) then (control-output is CE)
35	If (Position-error is NP) and (Position-error-change is PG) then (control-output is NG)
36	If (Position-error is NM) and (Position-error-change is NG) then (control-output is NG)
37	If (Position-error is NM) and (Position-error-change is NM) then (control-output is NM)
38	If (Position-error is NM) and (Position-error-change is NP) then (control-output is NP)
39	If (Position-error is NM) and (Position-error-change is CE) then (control-output is NG)
40	If (Position-error is NM) and (Position-error-change is PP) then (control-output is NG)
41	If (Position-error is NM) and (Position-error-change is PM) then (control-output is NM)
42	If (Position-error is NM) and (Position-error-change is PG) then (control-output is CE)
43	If (Position-error is NG) and (Position-error-change is NG) then (control-output is CE)
44	If (Position-error is NG) and (Position-error-change is NM) then (control-output is CE)
45	If (Position-error is NG) and (Position-error-change is NP) then (control-output is CE)
46	If (Position-error is NG) and (Position-error-change is CE) then (control-output is CE)
47	If (Position-error is NG) and (Position-error-change is PP) then (control-output is CE)
48	If (Position-error is NG) and (Position-error-change is PM) then (control-output is CE)
49	If (Position-error is NG) and (Position-error-change is PG) then (control-output is NG)

The fuzzy inference is given by Mamdani's technique and the defuzzification interface is obtained by the centroid method[15,16].

The fuzzy controller is designed with the next fuzzy sets:

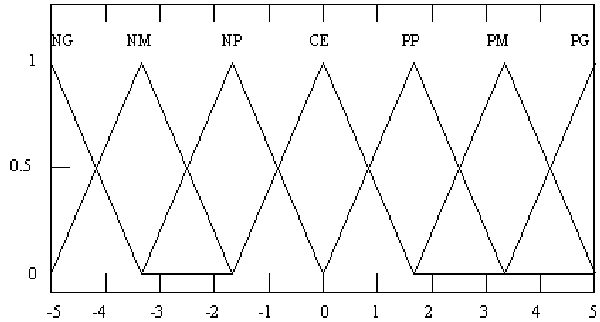

Figure-6
Input fuzzy sets: Position error and/or Position error Change

Notice that the membership functionsare the same for the above input sets, which were defined in section 3.

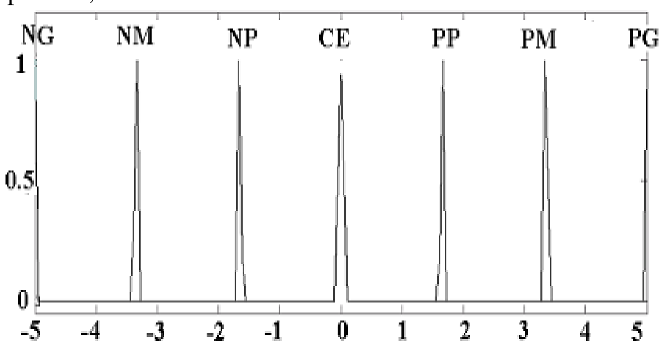

Figure-7
Output fuzzy set: Controloutput

Experimental Results

The main parameter values of the genetic algorithm were the following ones: i. Population size: 30, ii. Number of generation: 30, iii. Crossover probability: 0.80, iv. Mutation Probability: 0.09, v. Strategy selected: *Tournament,* vi. Elitism: Yes.

By means of the proposed method, we obtain a matrix which represents the fuzzy rules. This matrix is given by:

$$\begin{bmatrix} 5 & 7 & 7 & 7 & 5 & 4 & 6 \\ 5 & 7 & 4 & 5 & 7 & 7 & 7 \\ 6 & 7 & 4 & 7 & 7 & 7 & 4 \\ 5 & 6 & 7 & 4 & 4 & 4 & 1 \\ 1 & 4 & 2 & 3 & 2 & 1 & 4 \\ 4 & 2 & 3 & 4 & 4 & 2 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 4 \end{bmatrix} \quad (5)$$

The fuzzy controller was implemented in a MC9s12E128 microcontroller, because this microcontroller has specific functions such as fuzzification, fuzzy inference and defuzzification. The base of knowledge has the input and output membership functions. The rule evaluation was set in the microcontroller memory[17,18].

All the signals were obtained by a digital oscilloscope and were presented by a MATLAB software. Figure-8 shows the servomechanism output when a square signal is taken as the reference. This figure presents a small overshoot at the output signal rise with a state establishment of 150 msec. At the slope of the output signal there is not an overshoot and the state establishment is also 150 msec. The above results are generated because the fuzzy controller is a nonlinear system. Figure-9 shows a triangular signal and we can see a small steady state error.

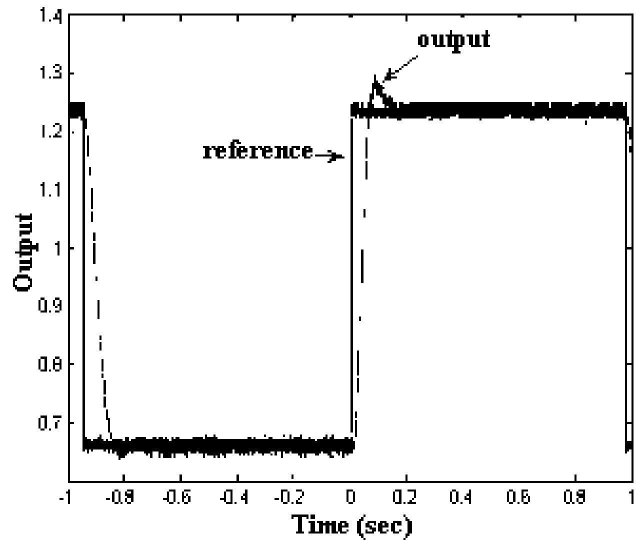

Figure-8
Temporary response of the servomechanism system to a square reference signal

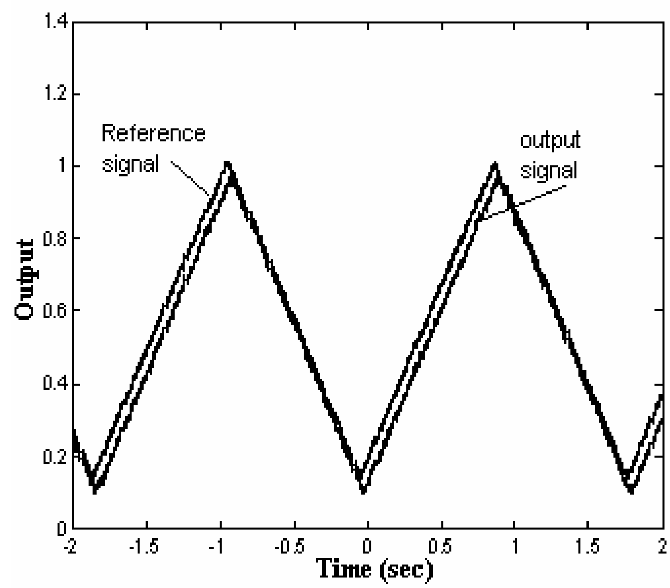

Figure-9
Temporary response of servomechanism system to a triangular reference signal

Figure -10 presents the output signal of the position control system, which corresponds to a sinoidal reference signal. This figure shows a small steady state when the reference signal changes from the maximum to the minimum value.

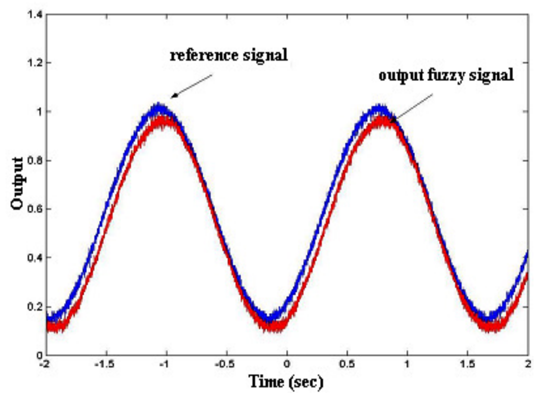

Figure-10
Temporary response of the servomechanism system to a sinoidal reference signal

Some rules were obtained by the optimization method and were modified in the experimental processby the performance improvement of the fuzzy controller. The above situation is justifiable because the fuzzy control was designed for a square reference signal. Then we present the final fuzzy rule base, given by table- 3.

Table-3
Final fuzzy rule base

ec / e	PG	PM	PP	CE	NP	NM	NG
PG	PG	PM	PP	PG	PG	PP	PM
PM	PG	PG	PG	PP	PM	PM	PM
PP	PM	PM	PG	PM	PM	PM	PM
CE	NM	NM	NP	CE	PP	PP	PP
NP	NG	NM	NP	NM	NP	NP	NP
NM	NP	NM	NP	NM	NP	NP	NP
NG	NG	CE	PG	NG	NP	NM	NG

Conclusion

The obtained results demonstrate that the optimization method to obtain de fuzzy rule base is satisfactory with a simple genetic algorithm.

The response of the real system has a better performance that the response of the simulation because the adjustment of the fuzzy rule base of the fuzzy controller was already done, when the fuzzy rules adjusted by the optimization method in the experimental tests process.

As a complement test, another reference signals (triangular and sinoidal) were used. Figures -9 and 10 show a good tracking of the reference signals with a small steady state error. The above results are justifiable because the fuzzy control was designed for a square reference signal.

The disadvantage of this optimization method for the design of the fuzzy rules is the simulation time. The authors propose to apply the optimization method presented in this paper to a complex process, where the extend simulation time of the tuning process is justifiable.

References

1. Lee C.C., Fuzzy Logic in Control Systems: Fuzzy Logic in Controller-Part 1, *Transactions on Systems, Man, and Cybernetics*, **20(1)**, 404-418 **(1990)**
2. Lee C.C., Fuzzy Logic in Control Systems: Fuzzy Logic in Controller-Part 2, *Transactions on Systems, Man, and Cybernetics*, **20(2),** 419-435 **(1990)**
3. ShahaboddinShamshirband and Ali Za'fari, Evaluation of the Performance of Intelligent Spray Networks Based On Fuzzy Logic, *Res.J.Recent Sci.,* **1(8),** 77-81**(2012)**
4. Sharifi M. and Shahriari B., Pareto Optimization of Vehicle Suspension Vibration for a Nonlinear Half-car Model Using a Multi-objective Genetic Algorithm, *Res. J. Recent Sci.,* **1(8),** 17-22 **(2012)**
5. Zhan J. Y. and Li Y., Application of Genetic Algorithm in Optimization of Fuzzy Control Rules, *Proceedings of the Sixth International Conference on Intelligent Systems Design and Applications, (ISDA06)***(2006)**
6. Karr C.L, Design of a Adaptive fuzzy logic controller using a genetic algorithm, *Proceedings of the Fourth International Conference on Genetic Algorithms*, Morgan Kaufmann, San Mateo CaliforniaUSA **(1991)**
7. Lagunas-Jiménez R. and Pitalúa-Díaz N., Tuning Fuzzy Control Rules via Genetic Algorithms, *Electronics, Robotics, and Automotive Mechanics Conference*, Cuernavaca, Morelos, México, september (CERMA) **(2007)**
8. Kuri Morales A., A Comprehensive Approach to Genetic Algorithms in Optimization and Learning Theory and applications, *Instituto Politecnico Nacional*, México,**1 (1999)**
9. Lagunas R., Fernández_Anaya G. and MartínezGarcía J.C., Experimental Evaluation of a Mixed H_2/H_∞-Based PID Using Genetic Algorithms, *Proceedings of the IASTED International Conference, Circuits, Signals and Systems*, May 19-21, Cancun México **(2003)**
10. Chen C., Linear System Theory and Design, Oxford University Press, Third Edition **(1999)**
11. *Fuzzy Logic Toolbox user's guide*, Mathworks Inc., USA**(2007)**
12. Bhattacharya Sourabh, Applications of DSTATCOM Using MATLAB/Simulation in Power System, *Res. J. Recent Sci.,* **1(ISC-2011),** 430-433 **(2012)**

13. Mamdani, E. H., Application of Fuzzy Algorithm for Control of Simple Dynamic Plant, *Proc. IEE.,***121(12)**, 1585-1588**(1974)**

14. Mamdani, E. H.and assilian, A Fuzzy Logic Controller for a Dynamic Plant, *Intl. J. Man Machine stud*., 1-13**(1975)**

15. Passino M. K., Yurkovich S., Fuzzy Control, Addison-Wesley, USA**(1998)**

16. PatilPallavi and Ingle Vikal, Obtaining a high Accurate Fault Classification of Power Transformer based on Dissolved Gas Analysis using ANFIS, *Res.J.Recent Sci.,* **1(2),** 97- 99**(2012)**

17. Pedrycz W., *Fuzzy control and Fuzzy systems*, John Wiley & Sons Inc. second edition, USA**(1996)**

18. CPU12 Reference Manual, MOTOROLA Inc. **(1997)**