Design of A Fuzzy Logic Controller for A Plant of N-Order Based on Genetic Algorithms

MOHANAD ALATA*, MOHAMMAD MOLHIM** AND KHALED AL MASRI**

* Mechanical Eng. Dept. //King Saud University//Saudi Arabia
malata@ksu.edu.sa
** Mechanical Eng. Dept. //Jordan University of Science and Technology//Jordan

Abstract:- Design of fuzzy controllers has been always a job built on past experience and knowledge of fuzzy control and systems behavior. Unlike that fashion this research introduced a new methodology of designing fuzzy controllers using genetic algorithms. The design employs Sugeno type fuzzy controllers as the parameters can be manipulated using GA. Single and two inputs fuzzy controllers are used. This research presented a solution of first, second, and third order systems, using the absolute average error as a fitness function, the genetic algorithm manipulate all parameters of the fuzzy controller to find the optimum solution. The simulation model used Matlab GA toolbox for finding the optimal solution, the fitness function took a different shape than the usual form, the new shape is introduced using a short program that is capable of generating the whole system, then calculating its output, error and finally the average error which is used as a fitness value to finally design the fuzzy controller.

Keywords:- Controller design, Fuzzy control, Genetic algorithms, Sugeno systems.

1. Introduction

Since the initiation of the fuzzy logic by Lotfi Zadeh 1965, and until this date the world is witnessing one of its most remarkable revolutions. The invention of the fuzzy systems provided an alternative to traditional notions, by providing a set membership and logic that has its origins in ancient Greek philosophy. It was not too long after that, the scientists started to study different approaches to design a self-designing fuzzy system, which led to several successful methods by using hybrid system such as Adaptive Neuro-Fuzzy Inference System (ANFIS). ANFIS architecture represents both the Sugeno and Tsukamoto fuzzy models. The effectiveness of ANFIS with the hybrid learning is tested through simulation examples. This has motivated us to use Genetic algorithms to design the fuzzy system.

Genetic algorithms (GAs) are global, parallel, stochastic search methods, founded on Darwinian evolutionary principles. During the last decade GAs have been applied in a variety of areas, with varying degrees of success within each. A significant contribution has been made in control systems engineering. GAs exhibit considerable robustness in problem domains that is not conducive to formal, rigorous, classical analysis. They are not limited by typical control problem attributes such as ill-behaved objective functions, the existence of constraints, and variations in the nature of control variables.

The computational complexity of the GA has proved to be the chief impediment to real-time application of the technique. Hence, the majority of applications that use GAs are, by nature, off-line .

Commonly GAs have been used to optimize both structure and parameter values for both controllers and plant models. They have also been applied to fault diagnosis, stability analysis, robot path-planning and combinatorial problems.

2. Literature Review

Passino and Yurkovich, [1] have explained the "functional fuzzy systems" known as Takagi-Sugeno fuzzy systems, in this system the output is an arbitrary equation instead of linguistic membership function in the rule base. The output of the system is calculated by the sum of multiplying each output by its membership value and dividing on the total sum of the membership values. Castro [2] has studied the number of membership functions that are required to get a good fuzzy controller. Castro established an approach to find the number of labels to assure a fixed accuracy when these types of fuzzy logic controllers are used; his work helps on selecting the first prediction of the fuzzy controller to be modified by the proposed approach in this paper. Euntai Kim et al. [3] proposed a new approach of fuzzy modeling. The suggested fuzzy model can express a given unknown system with a few fuzzy rules as well as Takagi and Sugeno's model [4], because it has the same structure as that of Takagi and Sugeno's model. The suggested algorithm is composed of two steps: coarse tuning and fine tuning. FCRM is used for coarse tuning, and gradient descent algorithm is used for fine tuning. Bortolet and Palm [5] developed a Takagi-Sugeno fuzzy model and controller for multivariable nonlinear systems. The identified fuzzy model is used for control especially to compensate nonlinear effects, since fuzzy logic has been proved to be a suitable tool to model and control nonlinear systems [4]. Meng, Q.C. et al. [6] Studied the encoding techniques of Genetic Algorithms since GA encoding has significant influence on GA systems performance when solving problems with high complexity. A sufficient convergence condition on genetic encoding in Genetic Algorithms has been presented, such as Bias Code, Uniform Code, Tri-sector Code and Symmetric Code. Angelov [7] proposed a new approach for on-line design of fuzzy controllers of TS type; fuzzy rules are generated based on data collected during the process of control using newly introduced technique for on-line identification of TS type fuzzy models. Output of the plant under control and the respective control signal has been memorized and stored in on-line mode, and used to train in a non-iterative, recursive way the fuzzy controller. L.Gacôgne [8] has used the Genetic Algorithms to find a set of non dominated solutions in the sense of Pareto instead of a unique solution with a unique fitness function. Gacôgne First began with a small random population of points in the space of research and setting a maximal size, then he used a family of genetic operators in relation with each specific problem, and he made a control on that family to give reinforcement for the best of them. Magdalena and Monasterio [9] proposed a new way to apply GAs to fuzzy logic controllers, and applies it to a FLC designed to control the Synthesis of biped walk of a simulated. A new approach adapted to systems with a larger number of variables have been proposed and tested over a FLC controlling a complex problem the locomotion of a simulated six links biped robot. Lee and Takagi [10] proposed a method for automatically designing complete triangular fuzzy systems using a genetic algorithm and a penalty strategy to determine membership function shape and position, number of fuzzy rules, and consequent parameters. Experimental results demonstrated the practicality of the method comparably to a system produced by another method, in Lee and Takagi work they have used triangular and trapezoidal membership functions for the fuzzy controller, and experimental score function. Several papers have proposed automatic design methods. Much of the work has focused on tuning membership functions [11-13]. Takagi and Hayashi [14] used neural networks as a membership values generator and in [15] they treated fuzzy systems as networks and used back-propagation techniques to adjust membership functions. Alata and Demirli and Bulgak [16] investigated the influence of the shape, the distribution of the membership functions and the order of the functional consequent of Takagi-Sugeno controller on the interpolation function of the fuzzy system. Number of inputs, conjunction operator, the order of consequent, and complementary or noncomplementary triangular membership functions will determine the shape of the output.

In this work TS fuzzy system will be employed for control based on genetic algorithms trying to optimize the parameters of the fuzzy controller. N-Order systems and Gaussian distribution function will be considered in this study. Even that the parameters of the Gaussian distribution will increase the complexity of the design, it will steer us to a more general solution, also the proposed approach depends on the real performance of the system instead of experimental score function used in [10].

The proposed approach starts with designing a first order Sugeno type fuzzy controller for a first order system, which will be optimized using Genetic Algorithm. Then, this procedure will be generalized by designing fuzzy controllers for second and third order systems. The overall average error between the desired and the actual output is to be fed to a genetic algorithm as a fitness value for evaluating the system. Inputs of the genetic algorithm will be encoded then crossover and mutation will be carried out on the population to finally change the centers and variances of the Gaussian membership functions of the Sugeno fuzzy controller. The resulting controller will be capable of guiding the system to the desired characteristics with permissible value of error, and a MATLAB simulation will be built up to conclude the design procedure.

3. Problem Formulation

The basic tools used for this research will be discussed, equations of the Takagi Sugeno model, input and output relations of first order Sugeno model is investigated. In addition, Genetic Algorithms basics, methodology are presented and discussed. Furthermore, details for other required tools such as tools used to transfer between S-domain and steady states to interconnect between the fuzzy controller and the plant being controlled are also explained. linear mapping for Sugeno systems will be used as shown in the following equation:

$$b_i = g_i(\cdot) = a_{i,0} + a_{i,1}u_1 + \cdots + a_{i,n}\, u_n \qquad (1)$$

For defuzzification and obtaining the output the following equation will be used:

$$y = \frac{\sum_{i=1}^{R} b_i \mu_i}{\sum_{i=1}^{R} \mu_i} \qquad (2)$$

To calculate μ_i the standard equation that represents the Gaussian membership function will be used:

$$\mu(u) = exp\left(-\frac{1}{2}\left(\frac{u-c}{\sigma}\right)^2\right) \qquad (3)$$

Where *c and σ* represents the center, and the variance of the Gaussian membership function.

Combining equations 1, 2 and 3 for first order Sugeno system the output can be calculated as follows:

$$y = \frac{\sum_{i=1}^{R} \left(a_{i,0} + a_{i,1} \cdot u_1\right).\, exp\left(-\frac{1}{2}\left(\frac{u-c_i}{\sigma_i}\right)^2\right)}{\sum_{i=1}^{R} exp\left(-\frac{1}{2}\left(\frac{u-c_i}{\sigma_i}\right)^2\right)} \qquad (4)$$

After we have designed the fuzzy controller, its parameters should be manipulated by the genetic algorithm, this algorithm should be able to optimize centers, variances, and coefficients of Equation 1. Genetics toolbox of Matlab can solve such a problem. In this toolbox you can define the function to use as fitness function, then define other parameters, and start the Genetic algorithm to find the solution.

One of the most challenging problems is the design of the fitness function. In this work, a function has been designed which is capable of calculating the overall error and return it as fitness value.

GA block in Figure 1 will get the following input variables:

- Variances of the Gaussian membership functions.
- Centers of the Gaussian membership functions.
- Coefficients of Sugeno first order equation.

The output from this process is the average error, which is calculated by a loop that creates the fuzzy controller and calculates the error for each point and store it, after finishing the time interval, the average error is calculated and the resulting output will be the fitness value that is used for the GA.

4. Simulation Model

Controller design and its tuning using GA include three stages: Run-Evaluate-Modify. Figure 1 shows the simulation model that Starts by designing the fitness function, followed by the GA, and other tools used to achieve the stated objectives.

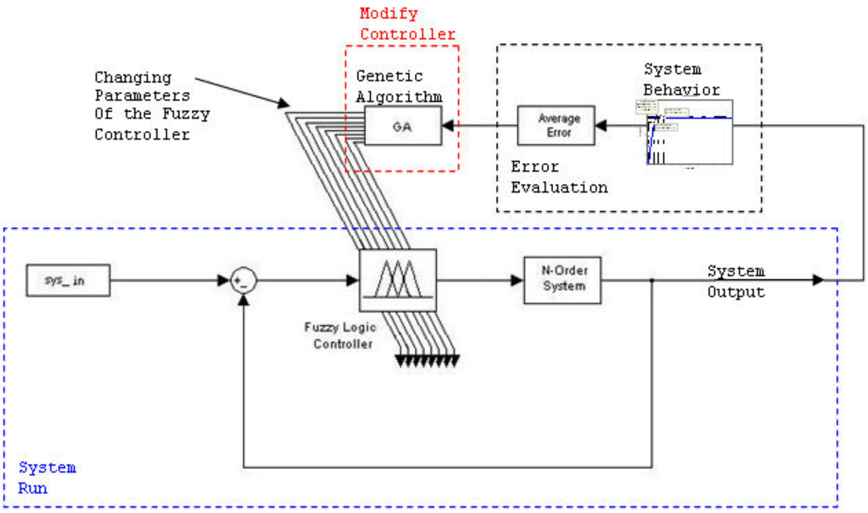

Fig. 1 Schematic diagram of the controller design and GA tuning process

4.1 Design of Fitness Function

Fitness function should return a value that evaluate the system and prepare it for the selection procedure. A computer program has been designed that will be executed on each time to evaluate the fitness of each child. Each time the program executed, it will do the following steps:

-System initialization.
-Creation of the reference system.
-Passing parameters and evaluating output.
-Calculation of new states.
-Error calculation or fitness value.

4.2 System Initialization

In order to start the simulation, this step is required for identifying the initial states of the system. Initially, inputs are set to zero. Also, initial error rate, and error sum are set to zero. The time step of the system is set to 0.1 second; this value provided smooth response, however the time for the system simulation is set to 10 seconds, this time is enough to evaluate the system performance, if the system proven a good characteristics on the 10 seconds time interval, it is tested again on 100 seconds time interval to make sure that the output does not deform on long time intervals.

4.3 Creation of Reference System

Reference system is required for comparing the resulted system with predefined behavior; variety of systems can be used here, but the following system showed a good output, the transfer function of the desired system behavior is given by equation:

$$T(S) = \frac{S^2 + 2S + 3}{S^3 + 4S^2 + 6S + 8}$$

It is worth to mention here that even if the system did not follow exactly the predefined behavior its performance could be evaluated as good performance, since we are not designing a path tracking system, but looking for best performance that could be achieved. Figure 2 shows the response of the desired system.

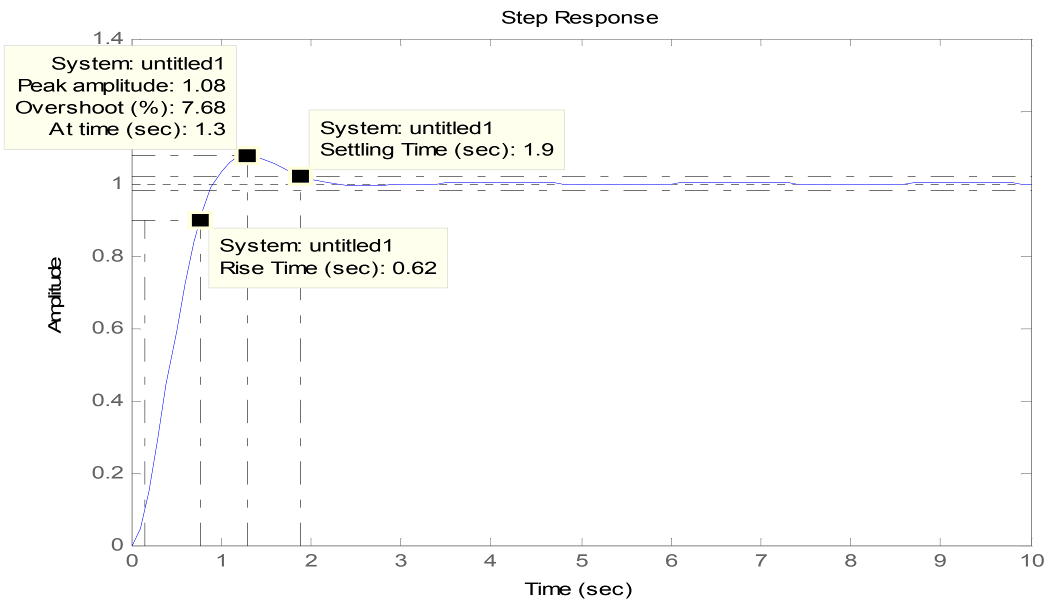

Fig. 2 the behavior of the desired system

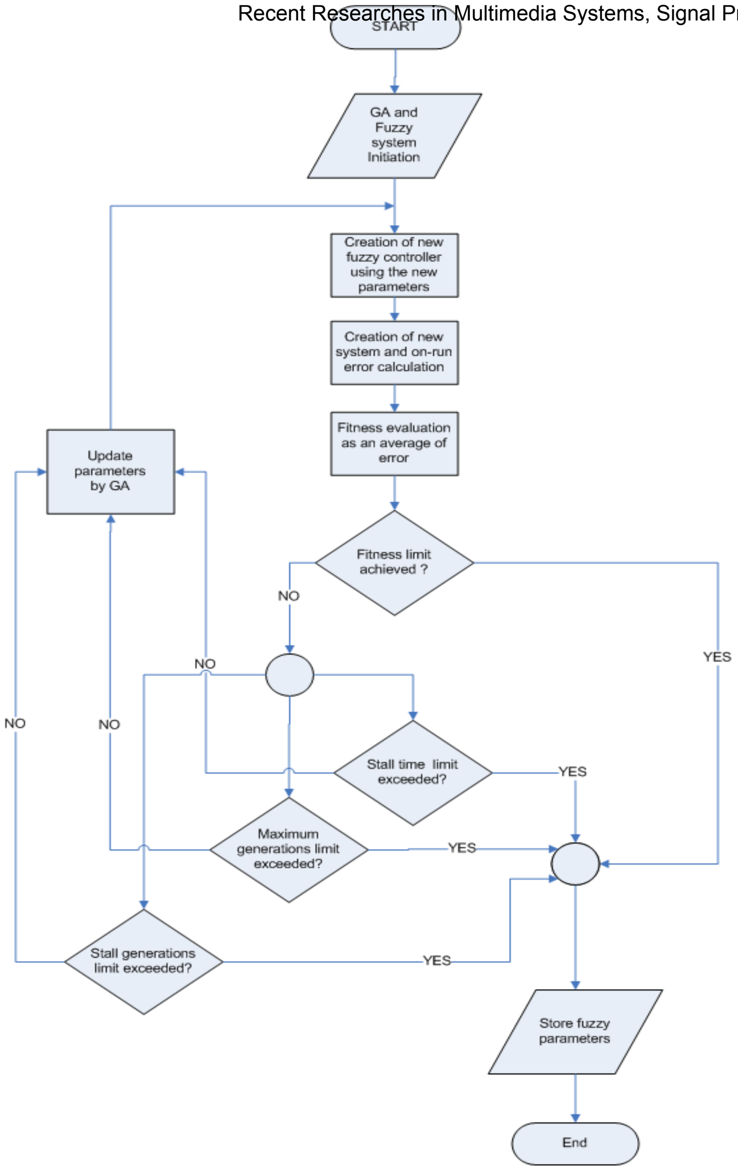

Fig. 3 Optimization process with GA

4.4 Passing parameters and Evaluating Output

After calculating the number of changing parameters in the fuzzy controller by the genetic algorithm, these parameters will be entered to the GA to be manipulated. The list below shows all the changing parameters by the GA:

- Gaussian membership function Variances (Sigma).
- Gaussian membership function centers.
- First order Sugeno equation coefficients in case of:
  - One input variable: $a_{i,0}$ *and* $a_{i,1}$ will be updated according to the following equation: $b_i = a_{i,0} + a_{i,1}.\ u_1$
  - In case of two input variables: $a_{i,0}$, $a_{i,1}$ *and* $a_{i,2}$ will be updated according to the following equation: $b_i = a_{i,0} + a_{i,1}.\ u_1 + a_{i,2}.\ u_2$

looking for the best behavior of the system, the error is calculated on every iteration and summed with the previous value; the final value of the error represents the fitness value that will be used by the genetic algorithm. The flowchart in Figure 3 explains the process in brief.

4.5 Results

4.5.1 First order system

For the first order system in equation below the step response and the unity feedback response are plotted in Figure 4.

$$T(S) = \frac{1}{2S+1}$$

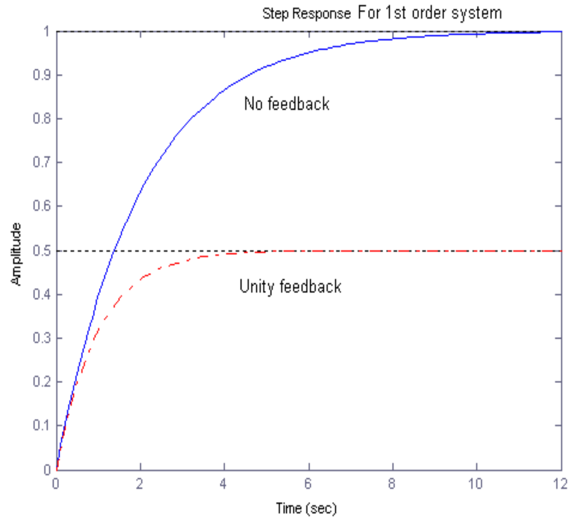

Fig. 4 The step response and the unity feedback response for the first order system

One input with three membership functions is used to get the results shown in Figure 5, the total average error is 1.8675 %.

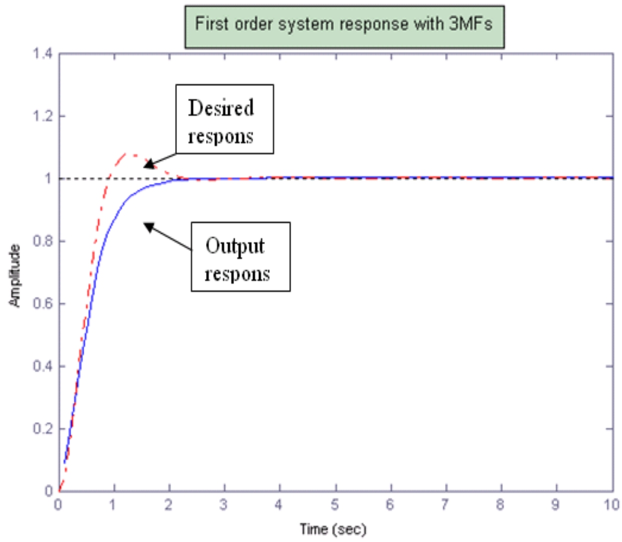

Fig. 5 The desired and the actual response for the first order system with 3 MFs

The resulting membership functions for the first order system with three membership functions is shown in Figure 6.

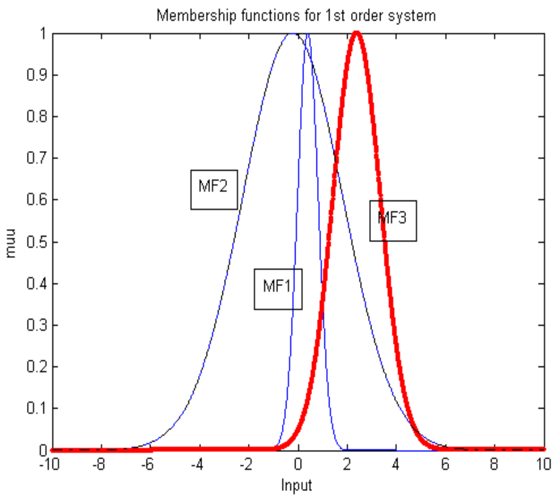

Fig. 6 The resulting membership functions for the first order system

4.5.2 Second order system: Mass-Spring-Damper

The schematic diagram of the Mass-Spring-Damper (MSD) system is shown in Figure 7.

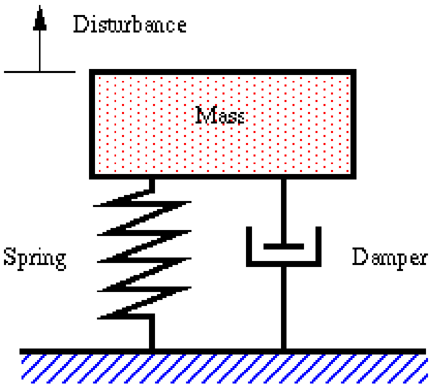

Fig. 7 Schematic diagram of the Mass-Spring-Damper

The transfer function of MSD system that we will design the controller for is:

$$T(S)=\frac{0.1}{S^2+0.05S+0.1}$$

In the previous example we have manipulated the system parameters for one input, in this example, another input variable will be created and considered in the solution, which is the rate of change of the error.

Since there are 2 inputs now with three membership functions for each the number of changing parameters raised dramatically. 39 parameters should be changed at the same time by the genetic algorithm. GA parameters remain the same as in the previous example. The rule base is as follows:

If Input1 is MF1 and Input2 is MF1 then output is Y1, $\mathbf{Y1= a_{11} u_1+b_{11} u_2+c_{11}}$

If Input1 is MF1 and Input2 is MF2 then output is Y2, $\mathbf{Y2= a_{12} u_1+b_{12} u_2+c_{12}}$

If Input1 is MF1 and Input2 is MF3 then output is Y3, $\mathbf{Y3= a_{13} u_1+b_{13} u_2+c_{13}}$.

If Input1 is MF3 and Input2 is MF3 then output is Y9, $\mathbf{Y9= a_{33} u_1+b_{33} u_2+c_{33}}$

With 15 children populations, the resulting response is plotted on the same figure with the desired output (Figure 8). The error is 0.65138 %. The membership functions for the error and error rate of change are displayed in Figures 9 and 10.

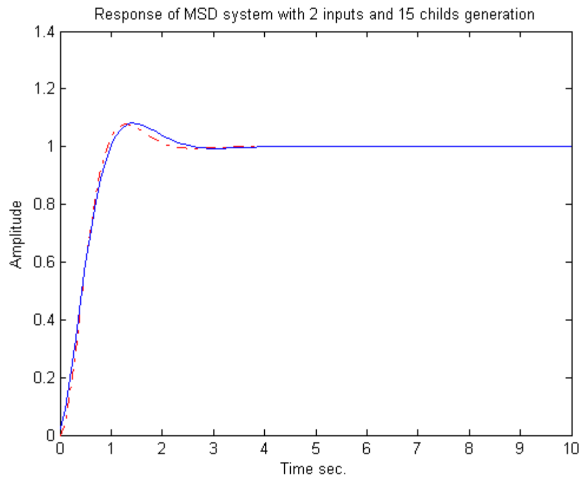

Fig. 8 The response of the MSD with 15 children population.

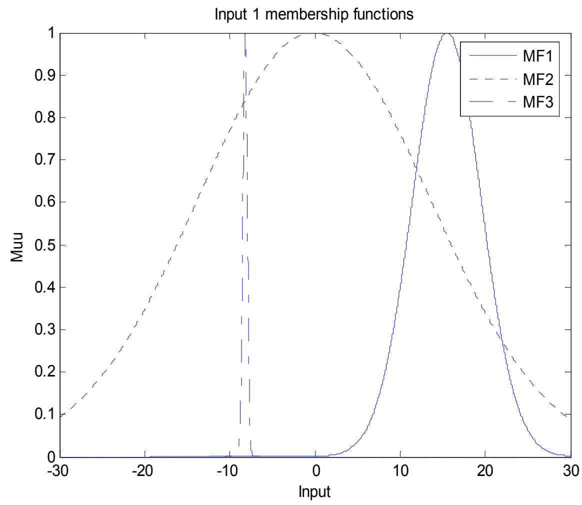

Fig. 9 The membership functions for the error for MSD system example

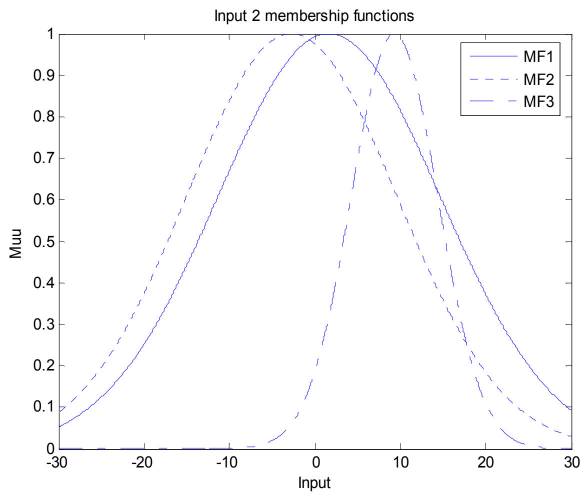

Fig. 10 The membership functions for the rate of change of error for MSD system example

4.5.3 Third Order Systems

The successive results of second order system encouraged us to examine the GA for third order systems. We have considered the following third order system.

$$T(S)=\frac{S^2+2S+3}{S^3+3S^2+4S+5}$$

ISBN: 978-960-474-283-7

The same GA configuration, and rule base used before for the second order system.
The total average error was 4.1996 %
Output response is plotted with desired response in Figure 11.

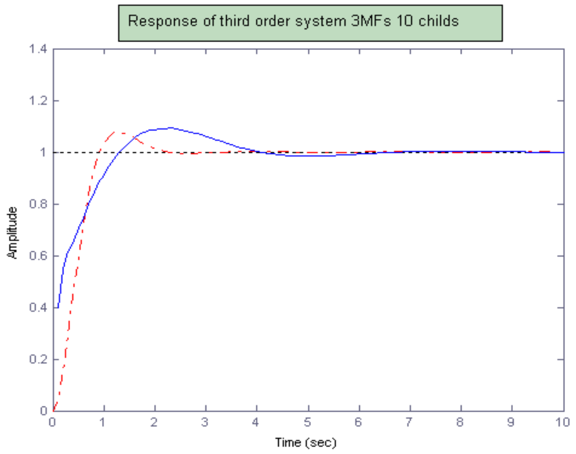

Fig. 11 The response of the third order system example

The membership functions for the error and error rate of change are displayed in Figures 12 and 13.

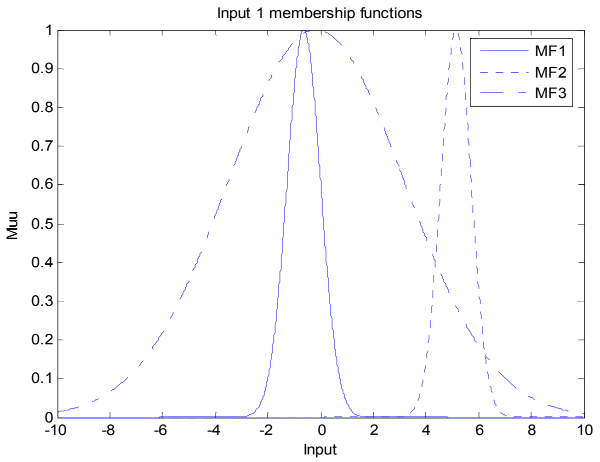

Fig. 12 The membership functions for the error for the third order system example

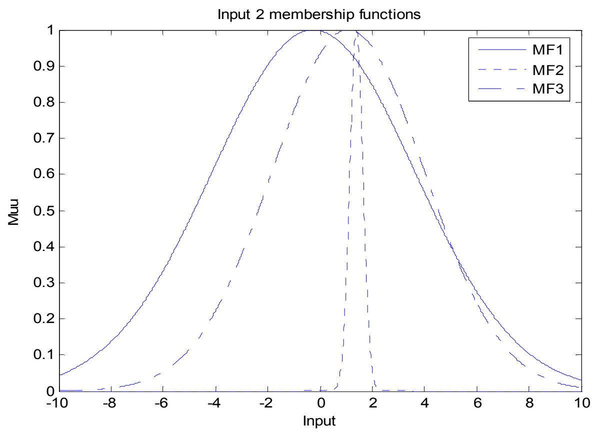

Fig. 13 The membership functions for the rate of change of error for the third order system example

5 Conclusions

In this work, updating all the parameters of the fuzzy controller using GA had led to a better response than that resulted from previously reported approaches in which parameters of only Sugeno consequents were updated. looking for the best performance of the system, the error is calculated on every iteration and summed with the previous value; the final value of the error represents the fitness value that has been used successfully by the genetic algorithm. The resulted controller is capable of guiding the system to the desired characteristics with permissible value of error. A Matlab simulation was built. Three examples were presented showing the capabilities of the proposed approach.

References

1. Kevin M. Passino, Stephen Yurkovich, "Fuzzy Control" Addison Wesley Longman Inc. 1998, 73-77:279-282.
2. J.L. Castr, "How many rules are necessary to get a "good" fuzzy controller for a control problem?" Proceedings of the Sixth IEEE International Conference. 1997, (2):749-754.
3. Euntai Kim, Minkee Park, Seunghwan Ji, Mignon Park. "A New Approach to Fuzzy Modeling," IEEE Transactions. 1997, 3(5):328-337.
4. T. Takagi and M. Sugeno. "Fuzzy identification of systems and its applications to modeling and control," IEEE Transaction System. 1985, (SMC)(15):116-132.
5. Bortolet, P., Palm, R. "Identification, modeling and control by means of_Takagi-Sugeno fuzzy systems," Proceedings of the Sixth IEEE International Conference. 1997, (1):515-520.
6. Meng, Q.C., Feng, T.J., Chen, Z., Zhou C.J., Bo. J.H. "Genetic algorithms encoding study and a sufficient convergence condition of Gas Systems," IEEE SMC '99 Conference Proceedings. 1999, (1):12-15, (1):649 - 652.
7. Plamen Angelov. "A fuzzy controller with evolving structure," Information Sciences. 2004, 161:21-35.
8. L.Gacôgne. "Research of Pareto Set By Genetic Algorithm, Application To Multicriteria Optimization Of Fuzzy Controller," Institut d'Informatique d'Entreprise (CNAM).
9. Magdalena, L., Monasterio F. "Evolutionary-based learning applied to fuzzy controllers," International Joint Conference of the Fourth IEEE International Conference. 1995, (3):20-24,(3):1111-1118.
10. Lee, M.A., Takagi, H. "Integrating design stage of fuzzy systems using Genetic algorithms," Second IEEE International Conference. 1993, (1) :612-617.
11. S. K. Sinha, R. N. Patel, R. Prasad " Application of GA and PSO Tuned Fuzzy Controller for AGC of Three Area Thermal-Thermal-Hydro Power System," International Journal of Computer Theory and Engineering. 2010, Vol. 2, No. 2, 1793-8201: 238-244.
12. Jang, R. "Fuzzy Controller Design without Domain Experts," IEEE Int. Conf. on Fuzzy Systems (FUZZ-IEEE'92).1992, 289-296.
13. Jang, R. "Self-Learning Fuzzy Controllers Based on Temporal Back Propagation," IEEE Trans. on Neural Networks.1992, 5(3):714-723.
14. Takagi, H., Hayashi I. "NN-driven Fuzzy Reasoning,' Int'l J. Approximate Reasoning (Special Issue ofIIZUKA'88). 1991, 3(5):191-212.
15. Nomura, H., Hayashi, I., Wakami N. "A self-tuning method of fuzzy control by descent method," 4th IFSA Congress.1991, 55-158.
16. Alata M., Demirli K., Bulgak A. "Interpolation Behavior of TS Fuzzy Controllers," International Symposium on Intelligent Control/Intelligent Systems and Semiotics Cambridge. 1999, 359-364.