International Journal of Systems Signal Control and Engineering Application 9 (1): 1-10, 2016
ISSN: 1997-5422

Optimize and Control the Robot with Two Degrees of Freedom Using Scaling Coefficients Set Membership Functions Using Genetic Algorithm

[1]Mohahammad Javad Tajadini and [2]Majid Mohammadi
[1]Department of Electrical and Control, Kerman Science and Research Branch, Islamic Azad University, Sirjan, Iran
[2]Department of Computer Engineering, Shahid Bahonar University, Kerman, Iran

Abstract: The use of the optimization technologies for the two degree of freedom control for Robot manipulators is a new idea and there have been applied various methods for controlling and optimizing robots. The general theorem in such optimization methods is the determination of the decision variables amounts for maximizing or minimizing the objective function and this is a very tedious task when the number of membership functions is too many or the system dynamicity is very slow. In the present study, the optimized output membership functions have been identified through combining the genetic algorithm and fuzzy logic, based on the input membership functions for two degrees of freedom control robot manipulators. The method has been the use of genetic algorithm for finding the optimum parameters in the Sugeno Fuzzy Logic Method. The objective function in such a problem is in the form of a system of various objectives and goals of two degrees of freedom controls for robot manipulators. The main objective of the current study is to make use of a Genetic algorithm method to mechanize the design and reach to an optimum regulation of the membership functions and therefore the scientific considerations regarding the regulation and design through the use of scaling coefficients along with the fuzzy control for the two degrees of freedom controls for robot manipulators have been presented here.

Key words: Fuzzy control, Genetic algorithm, robot manipulator (arms), degree, robot

INTRODUCTION

The robotics arms system is currently being extensively used in the industry and there has been found a variety of usages for such systems application, among which one can refer to the use of robots in the automatic production lines, vehicles loading and unloading such as airplanes and ships, automatic painting, radioactive substances handling, deep-sea probes, space trips and military applications. Moreover, robotics arms or manipulators make use of completely nonlinear and complicated systems. Such systems do not possess a fixed and stable structure due to various reasons such as loading variations and/or friction between the arm joints and there is always present a sort of uncertainty in them. The uncertainty existing in the robotics arms can bring about instability in the system performance and make controlling job difficult. For the same reason, many various types of controllers have been designed for such systems. Among the controlling methods the fuzzy control method is very practical and efficient. Fuzzy logic which is based on fuzzy systems has been invented by Zadeh (1975) to handle the uncertainty and the imprecision existing in the knowledge and sciences. A fuzzy system makes use of fuzzy sets to describe the relationships mapped between the input and the output. Fuzziness of the complexes makes it possible for the output to be interpolated from among several theorems and a smooth and continuous output can be created from a combination of discontinuous theorems and rules. The evolutionary algorithms are basically considered as the searching algorithm-functions optimizations and they can be exerted on a wide range of the problems in which there is given a function but the decision variables optimizing the function are indefinite. Such problems include engineering optimizations, schedule planning, bioinformatics, developable hardware and even art.

MATERIALS AND METHODS

Robotics arms systems model: The n-degrees of freedom robotic arm dynamic equation can be expressed in the form of a quadratic non-linear equation by making use of Lagrange or Newton-Euler Method:

Corresponding Author: Mohahammad Javad Tajadini, Department of Electrical and Control, Kerman Science and Research Branch, Islamic Azad University, Sirjan, Iran

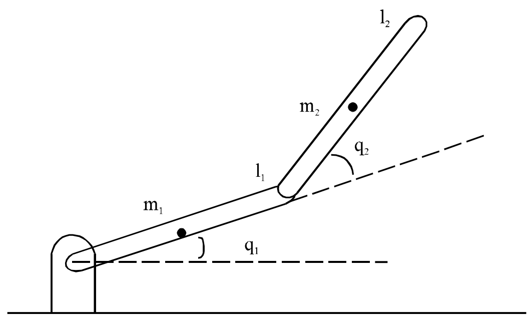

Fig. 1: The two-joint robot arm

$$\tau = M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) = W(q, \dot{q}, \ddot{q})\phi \quad (1)$$

Where:

$W(q, \dot{q}, \ddot{q})$	=	A definite function of the robot dynamics
ϕ	=	A vector including the uncertain parameters for the robot dynamics
$q = [q_1, ..., q_n]^T$	=	The robot joints positions vector
$\dot{q} = [\dot{q}_1, ..., \dot{q}_n]^T$, $\ddot{q} = [\ddot{q}_1, ..., \ddot{q}_n]^T$	=	The velocity and acceleration vectors, respectively
$M(q)$	=	A (n×n) inertial matrix
$C(q, \dot{q})$	=	The (n×n) Coriolis and centrifugal forces
$G(q)$	=	The n×1 gravity vector
$M(q)$	=	A symmetrical, positive and definite matrix

The two-joint robot arms have been illustrated in Fig. 1. As it is observed in the figure q_1 is the position of the first joint to the horizontal line and q_2 is the position of the second joint to the first one. The two-joint robot's dynamic equation is in the following form and it has been proposed in many of the articles and references:

$$\begin{aligned}\tau_1 &= m_2 l_2^2(\ddot{q}_1 + \ddot{q}_2) + m_2 l_1 l_2 c_2(2\ddot{q}_1 + \ddot{q}_2) + (m_1 + m_2) l_1^2 \ddot{q}_1 \\ &- m_2 l_1 l_2 s_2 \dot{q}_2^2 - 2m_2 l_1 l_2 s_2 \dot{q}_1 \dot{q}_2 + m_2 l_2 g c_{12} + (m_1 + m_2) l_1 g c_1 \\ \tau_2 &= m_2 l_1 l_2 c_2 \ddot{q}_1 + m_2 l_1 l_2 s_2 \dot{q}_1^2 + m_2 l_1 g c_{12} + m_2 l_2^2(\ddot{q}_1 + \ddot{q}_2)\end{aligned} \quad (2)$$

The above equations are of a general format according to the relation for which $M(q)$, $(q, \dot{q})$ and $G(q)$ matrices can be written as below:

$$\begin{aligned} M(q) &= \begin{bmatrix} (m_1 + m_2)l_1^2 + m_2 l_2^2 + 2m_2 l_1 l_2 c_2 & m_2 l_2^2 + m_2 l_1 l_2 c_2 \\ m_2 l_2^2 + m_2 l_1 l_2 c_2 & m_2 l_2^2 \end{bmatrix} \\ C(q, \dot{q}) &= \begin{bmatrix} -m_2 l_1 l_1 \dot{q}_2 s_2 & -m_2 l_1 l_1 \dot{q}_2 s_2 \\ m_2 l_1 l_1 \dot{q}_2 s_2 & 0 \end{bmatrix} \\ G(q) &= \begin{bmatrix} (m_1 + m_2) g l_1 c_1 + m_2 g l_2 c_{12} \\ m_2 g l_2 c_{12} \end{bmatrix} \end{aligned} \quad (3)$$

where in Eq. 3:

$$c_1 = \cos(q_1),\ c_2 = \cos(q_2),\ s_1 = \sin(q_1),$$
$$s_2 = \sin(q_2),\ c_{12} = \cos(q_1 + q_2)$$

And m_1 and m_2 are the arms masses and l_1 and l_2 are the lengths of the arms and g is the earth gravitational acceleration. Also, if the m_3 can be considered as the movable load acting on the second arm and it can be also regarded as concentrated on the ending point of the second arm, then the system model can be introduced as the relation with the following matrices:

$$M(q) = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix} \quad (4)$$

$$C(q, \dot{q}) = \begin{bmatrix} C_{11} & C_{12} \\ C_{21} & 0 \end{bmatrix} \quad (5)$$

$$G(q) = \begin{bmatrix} G_1 \\ G_2 \end{bmatrix}$$

Where:

$$\begin{aligned} M_{11} &= (m_1 + m_2 + m_3)l_1^2 + (m_2 + m_3)l_2^2 + 2(m_2 + m_3)l_1 l_2 c_2 \\ M_{12} &= (m_2 + m_3)l_2^2 + (m_2 + m_3)l_1 l_2 c_2 \\ M_{12} &= M_{21} \\ M_{22} &= (m_2 + m_3)l_2^2 \\ C_{11} &= -2(m_2 + m_3)l_1 l_2 \dot{q}_2 s_2 \\ C_{12} &= -(m_2 + m_3)l_1 l_2 \dot{q}_2 s_2 \\ C_{21} &= (m_2 + m_3)l_1 l_2 \dot{q}_1 s_2 \\ G_1 &= (m_1 + m_2 + m_3) g l_1 c_1 + (m_2 + m_3) g l_2 c_{12} \\ G_2 &= (m_2 + m_3) g l_2 c_{12} \end{aligned} \quad (6)$$

Fuzzy control: To overcome the need to describe sciences with certainty, the fuzzy sets were introduced by Zadeh (1965) which allowed for the possibility to make use of the logical approximate forms in such a manner that the terms "right" and "wrong" included part of reality. Ten years later, Zadeh generalized such fuzzy sets and systems to the known fuzzy systems entitled type 2 fuzzy systems to model the uncertainties which can be manifested in designing the type 1 fuzzy systems (Lughofer, 2011; Mendel, 2001).

During the past decade, fuzzy logic has been successfully applied in various fields such as regression, system modeling, pattern classifications and so forth. Among such usages, the area of control can be regarded as one of the most relevant one.

FLCs (Fuzzy Logic Controllers) have three main components such as fuzzification, fuzzy inferential engine

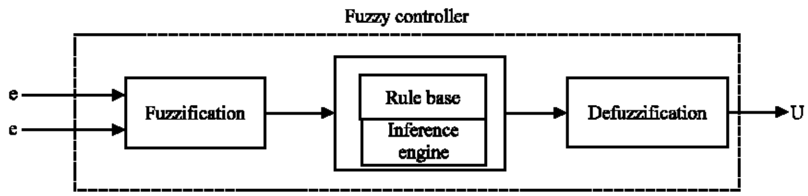

Fig. 2: The structure of a fuzzy logic controller

and defuzzification. The FLC block diagram has been illustrated in Fig. 2. The first block is defined in the fuzzification section of the fuzzy system for the input and output variables. In most of the cases, the output error that is to say the differential between the process output and the reference signal and the variations or its derivatives form the inputs to the fuzzy system. The base is indeed the heart of the fuzzy system and it contains the rules and the criteria by the use of which the controllers can be capable of actualizing the objectives. Usually, such rules and regulations are expressed in the "if, …, then" format and they act as the mapping between the input fuzzy variables to the output fuzzy variables. The inference engine can be regarded as the brain of a fuzzy controller which has the ability to imitate the human decision making methods based on fuzzy concepts. This part has to be made based on the databases, fuzzy inputs and outputs. In defuzzification, the actual value of the output is firstly obtained. There are various methods for defuzzification the most common and most widely used one of which is the gravity center.

Through the use of the fuzzy logic definitions we make practical use of a two-input-one-output fuzzy system to control each of the robotic arms. The system inputs are error that is the extent to which the actual angle deviates from the optimum and the error derivative, respectively. The output, as was expected is the amount of the torque inputted to the joint. To simplify the fuzzy system, the zero-order Sugeno fuzzy system has been applied. This system is similar to Mamdani-type fuzzy inference system to a great extent except that the membership functions are replaced by the single values in the output part.

The membership functions which have been considered for the input variables to both of the fuzzy controllers are of the Gaussian type according to the following figures and both have been normalized in [-1, 1] space (Fig. 3 and 4).

The membership functions have also been considered on the output dimension in the form of single values uniformly in [-1, 1] space according to Fig. 5.

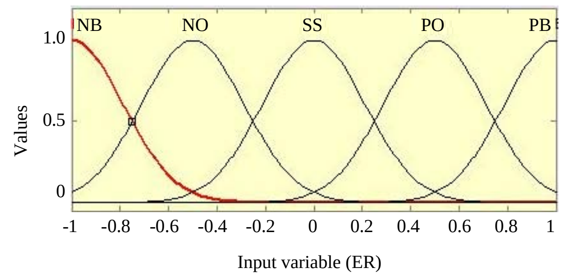

Fig. 3: Error membership functions

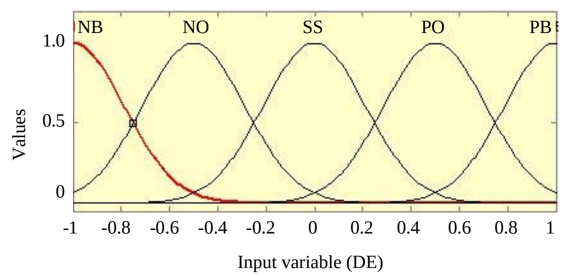

Fig. 4: Error derivatives membership functions

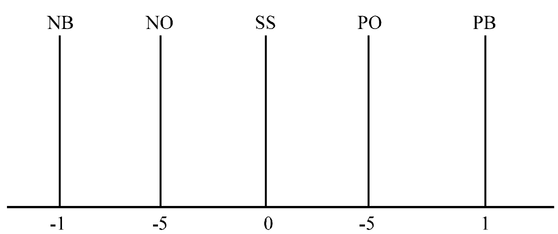

Fig. 5: The membership functions for the output variable

The fuzzy system inputs are firstly fuzzified according to the membership functions which have been taken into consideration for them and then the inputs are approximately deduced according to the collection of the rules and regulations which have been considered for the fuzzy system and the fuzzy system produces the intended output.

Since, the controller has two approaches and each of them has five membership functions, so one perfect rules database is comprised of 25 rules for such a system. The

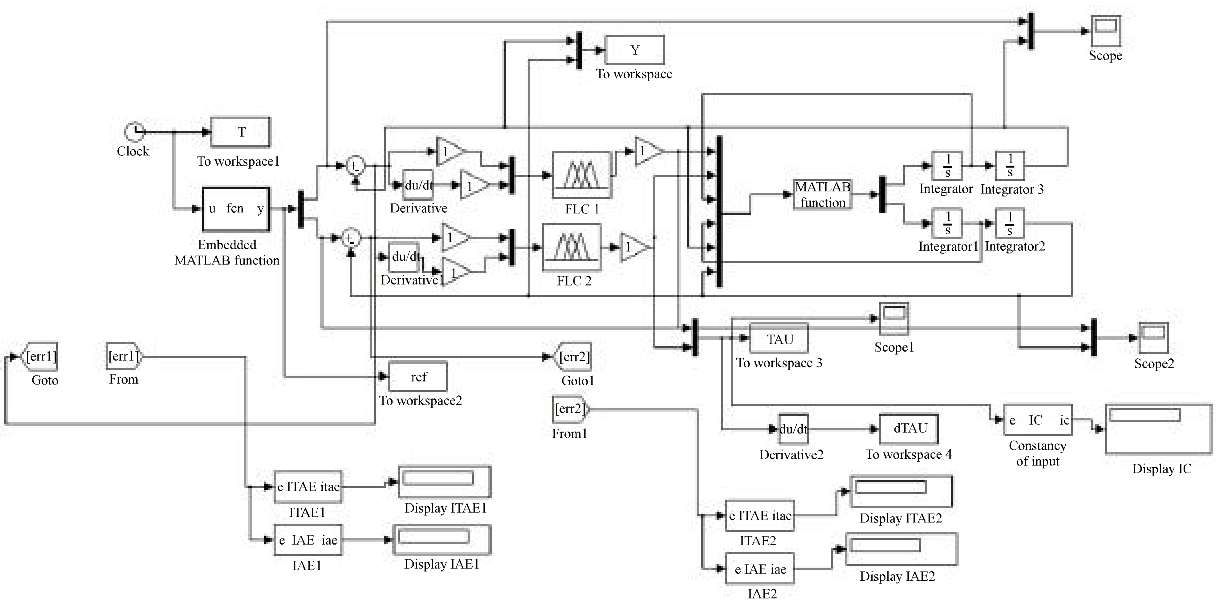

Fig. 6: A schematic view of the controller system in MATLAB's simulink

Table 1: Fuzzy rules database

	$\dot{e}$				
e	NB	NO	SS	PO	PB
NB	NB	NB	NO	NO	SS
NO	NB	NO	NO	SS	PO
SS	NO	NO	SS	PO	PO
PO	NO	SS	PO	PO	PB
PB	SS	PO	PO	PB	PB

rules database which has been considered here is a standard rules database which has been taken into consideration in the majority of the articles as the controller which is visible in Table 1.

A general schematic view of the controller system which has been executed in Simulink of the MATLAB Software can be observed shown in Fig. 6.

The important theme of the schema is the use of the scaling factors and they cause the variables e and $\dot{e}$ which have been regarded as the inputs to the fuzzy system to be spaced well in the [-1, 1] range. Another point of value here is the use of MATLAB FCN which takes advantage of an m-file named "dyn.m" and that the section 5.3 robotic equations have been implemented in it.

GA optimization algorithms: Genetic algorithms are a known set of EAs. These algorithms are currently being widely used in a way that sometimes genetic algorithms expressions and the evolutionary calculations are being used interchangeably (for instance in the present article). Such algorithms work with a fixed population of the individuals (chromosomes) and each of these chromosomes provide for a likely solution to the assigned problem. In each reproduction, the individuals are codified and then a fitness score is specified for each of them based on which one(s) solves the problem better than the others. The individuals with appropriate selection likelihood to individuals' goodness of fitness ratio can be selected for the next generation production. After the selection process was ended, a blending takes place between the pairs of the selected individuals. The two individuals' strands get combined, through the process of which a new individual is produced which is characterized by the traits inherited from two different fairly successful individuals. The next operation in the line is mutation; stochastic selection of the bits from chromosomes. This event usually takes place with a relatively low likelihood. Mutation makes it sure that the likelihood with which an existing section of the response space can be searched is never zero.

An introduction to the GFSs: As it was mentioned previously, designing a FS includes two stages: the selection of the system structure that is the rules and regulations in the rule base and defining the correct parameters (or the database) which means the antecedents, consequents, language variables and so forth. Therefore, manual designing and regulation of a FS is a very difficult task, especially when the number of the MF parameters increase. The automatic identification of the fuzzy system or structure parameters can be regarded as a search or optimization process (Cordon, 2011).

Corresponding to the great number of the parameters which should be specified in a FS the size of the search space is relatively very huge. To the same reason, we are

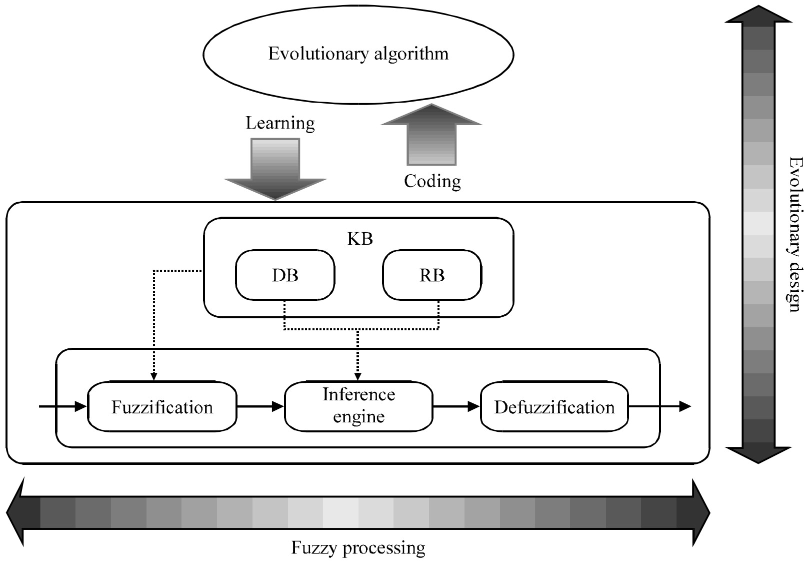

Fig. 7: Genetic fuzzy system architecture

in need of powerful search methods capable of managing the spaces with huge dimensions. At the present time, the evolutionary algorithms have been recognized as the general effective and competent search techniques for this set of the problems. The survey of the histories indicates that the pertinent and outstanding evolutionary fuzzy system types include genetic learning or the regulation of numerous components of the fuzzy rule-based systems (which are usually named Genetic Fuzzy Systems or GFS).

Genetic Algorithms (GAs) have been used with various complexity levels, from the membership function regulation to the production of fuzzy theorem which includes both regulation and training (Cordon *et al*., 2004; Bonarini, 1996).

GFS architecture has been shown in Fig. 7. FS genetic design includes knowledge-base parameters encoding in an appropriate genetic format. At the time that the knowledge base is optimized the fuzzy process can start working as a standard fuzzy system through calculating the outputs.

Fuzzy genetic system can be divided into two general approaches in a broad sense: genetic regulation processes and genetic learning processes (Cordon *et al*., 2004; Bonarini, 1996).

Genetic regulation processes have been defined as the previous FRBS (Fuzzy Rule-Based System) performance optimization procedure and they are used through regulating the DB (membership functions and/or scaling factors) parameters. Therefore, the main objective of the optimization procedure is the regulation of DB parameters and the rule base is left unchanged during the regulation process. The genetic learning processes are influenced by automatic production or optimization of a set of fuzzy rules which develop the input and output states interrelationships.

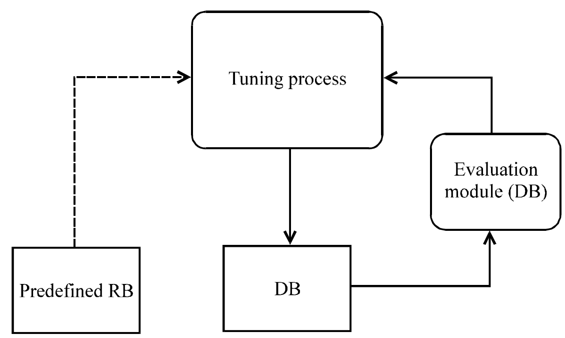

Fig. 8: Database regulation and tuning

Genetic regulation: Scaling and fuzzy membership functions regulation is a major task in the process of designing the FRBS. The scaling factors and the membership functions are adjusted by the GA based on fitness function which quantitatively specifies the design criteria. As it was stated previously, regulation processes assume a predefined DB and their objective would be finding a set of optimized parameters for the membership and/or scaling functions (Fig. 8).

In the primary task which was fulfilled regarding the genetic regulation (Karr, 1991) took the linguistic FRBs into consideration. The DB definition is encoded in the chromosomes which includes a series of fuzzy systems input and output parameters.

Scaling functions regulation: The scaling function which is operated on the FRBS input and output variables normalizes the space in which the fuzzy membership functions have been defined. It is possible to parametrize the scaling functions and then they can be adjusted to the GAs. Such parameters are adjusted in a manner that the scaled space can be better corresponded to the variable limits. Usually, the scaling functions which have been used in the history can be divided into linear sets which have been given in the article proposed by Ng and Li (1994) and non-linear sets which have been offered in an article authored by Magdalena and Monasterio-Huelin, (1997).

Membership functions regulation: The fuzzy systems can be defined by the use of local or general semantics. For the local semantics, each of the fuzzy systems in each of the theorems has its own specific membership function. For the general semantics in the article proposed by Chien *et al.* (2002), the definition of the membership function is shared between the theorems and they are iteratively used. The general semantics make the explicit isolation of the membership functions from the rule base possible and this enables the learning algorithms to target one or both of them.

The simplest method for making use of GA to improve fuzzy system is to start with an existing system and optimization or adjusting the shape and the position of the membership functions. Regulation or tuning is one of the preliminary applications considered for GA in fuzzy systems and this is due to the simplicity and the high number of the previously existing fuzzy systems which can be regulated through following this procedure.

When regulating the membership functions, an individual defines the entire DB as its own chromosome which the relevant parametrized membership functions can be encoded through the use of linguistic phrases in each fuzzy section of the FRBS. The most common membership functions shapes (in GFRBS) are triangular, trapezoidal or Gaussian functions ones. The number of the parameters per membership function usually ranges from 1-4 and each of the parameters is encoded in a binary or actual format (Liska and Melsheimer, 1994).

A specimen of the creation of the chromosomes (the actual encoded strands) for a fuzzy system with three throughputs has been illustrated in Fig. 9. Every input has two Membership Functions (MF) which is related to it and both are Gaussians. The Gaussian membership functions can be described with two parameters: a median amount and a width amount. Gaussian membership functions are usually used for the reason that they are continuous, derivable and simple to describe. Considering the fact that

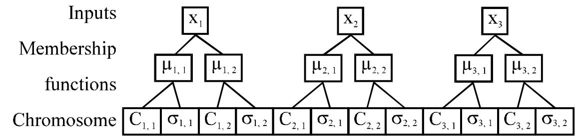

Fig. 9: Gaussian MF encoding

the rule base is left unchanged and there is a rule for every combination of the membership functions a chromosome as simple as such a vector is one of the MF parameters.

Rule base genetic learning: The main problem which is needed to be solved for FRBS learning includes finding a proper description which is capable of collecting the problem specifications and offering potentially appropriateness responses for it. Classically, three genetic learning approaches correspond to the machine learning system grounds and are therefore applied.

Pittsburgh approach: In this approach, every chromosome describes a general RB and the evolution takes place through the genetic operators which are exerted on the fuzzy rules sets level. The fitness function evaluates the FRBS precession which has been completely encoded in a chromosome. In the study proposed by thrift (1991) the first major step in Pittsburg approach is the RBs learning. This method works through taking advantage of a complete decision table which characterizes a certain case in crisp relations and it is defined in contrastive fuzzy systems with input and output variables. A chromosome of each of the decision table is obtained from each of the lines and through the codification of each fuzzy system output in the form of a number which incorporates a label "null". Therefore, GA applies an integer encoding method.

Michigan approach: In this approach the chromosomes are regarded as single fuzzy rules and the RB is described by the entire population. The set of the fuzzy rules are adjusted during the course of time through the use of some of the genetic operators exerted in every level of the single rules. This evolution is conducted via a credit allocation system which evaluates each of the single fuzzy rules correspondence. In the study performed by Valenzuela, the first GFS for RBs learning has been offered based on Michigan approach.

Iterative Rule Learning approach (IRL): In IRL, the same as Michigan approach, every chromosome in a population describes a unique single fuzzy rule but it is regarded as the best individual for the final part of RB. Therefore, in this approach, EA provides a partial solution to the learning problem and it stands in contrast to the two previously mentioned approaches. This approach has to be run iteratively to obtain a complete RB. This is carried out in this way that it includes an iterative plan based on obtaining the best current fuzzy rule for each of the systems which gets this rule participated in the final RB and it punishes RB before repeating the process. This procedure finishes when the RB is no longer capable of properly describing the system.

RB genetic learning considers a collection of predefined membership functions which are located in DB and the rules are being referred to through the use of linguistic labels (Fig. 10). And this is only implemented on the general semantics (descriptive semantics) in such a manner that in local semantics (approximate FRBS) rule adjustment is regarded as equal to the membership functions improvement. Three aforementioned learning methods can be considered for RBs training.

The Michigan approach presented in references (Bonarini, 1996; Ishibuchi *et al*., 1999; Subbu *et al*., 1998), the Pittsburg approach introduced in the reference articles (Thrift, 1991; Hoffmann and Pfister, 2013; Pham and Karaboga, 1991) and the iterative rule learning approach in the reference articles (Gonzblez and Perez, 1999; Cordon *et al*., 2001) regarding RB can be described via a relational matrix in reference (Thrift, 1991), a decision table (Pham and Karaboga, 1991) or a list of the rules (Hoffmann and Pfister, 2013; Pham and Karaboga, 1991; Gonzblez and Perez, 1999).

Knowledge base genetic learning: When using a rule base with local semantics, the membership systems and rules are inseparable and they should be trained altogether. Alternatively, general semantics enables the membership function to be trained independently from the rules and vice versa or they can be trained together. There are two ways for both of the membership functions and rules learning; sequential or simultaneous.

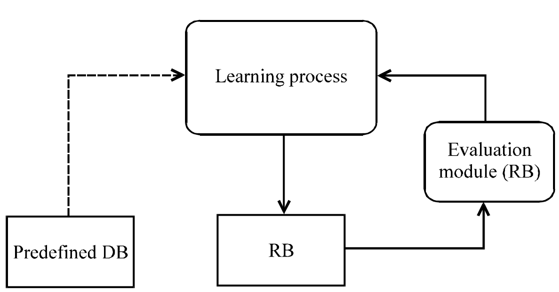

Fig. 10: Rule base learning

In the sequential learning approach, firstly the scale factors should be found by the GAs and the membership function and rule base are supposed to be fixed. When the GA succeeded in finding the best scaling factors coefficients values, such parameters become fixed and the membership functions parameters are processed by GA. Finally, through the use of scale coefficients and membership functions a new rule base is found out by GA. Since, the various components of the fuzzy systems (RB and DB) are not independent from one another such approached may lead to a generally suboptimum performance. In fact, through changing a parameter of a fuzzy system some of the other parameters change as well. In simultaneous approaches, both of the DB and RB of the fuzzy systems in one chromosome are encoded to obtain a better performance.

For instance, Homaifar and McCormick (1995) suggested the use of GAs for a perfect KB learning for controllers problems which describes both of the membership functions and RBs for the purpose of addressing their correlation (KB learning). They considered a simple GA for Pittsburgh approach accompanied by a numeral encryption for the rules consequences and numeral encoding for the membership function domain (5 different domain values) in an identical chromosome. This approach is regarded as a reference for the classic Pittsburgh approach to KB genetic learning.

RESULTS AND DISCUSSION

In the following study, we deal with the controller optimization results by the use of the genetic algorithm. In the current article we made use of genetic algorithm for the optimization of the centers and the extensions of the membership functions on the input and the fuzzy rule consequent values and scaling coefficients. To optimize the controllers there can be made use of various scales. In the current study, however, we made use of Mean Square Error (MSE) of the system as the objective function.

As it was mentioned previously, since our fuzzy system has five membership functions on each of the input dimensions (and therefore we have 20 free parameters to be optimized) and five consequences on the output dimensions (5 parameters), so we have 25 free parameters for each of the fuzzy systems to be described and quantified. Consequently, it can be stated that the chromosomes are comprised of 56 dimensions in genetic algorithm. So, we have chosen to set the iteration stages number equal to 60 and the number of the population has been selected as equal to 15 individuals. In the study to come we deal with the controllers' optimization via the genetic algorithm by taking advantage of scaling coefficients and Baun Genetic algorithm and also we have made use of genetic algorithms, the results of which have been presented in figures.

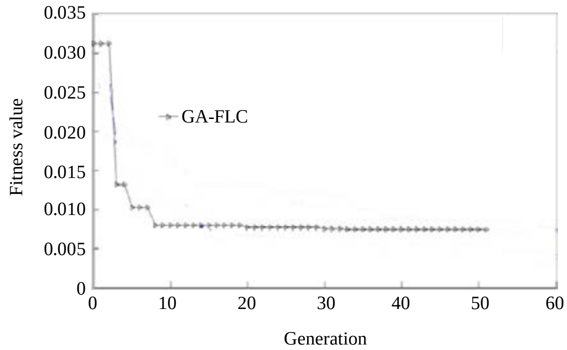

Fig. 11: The three algorithms objective function diagram

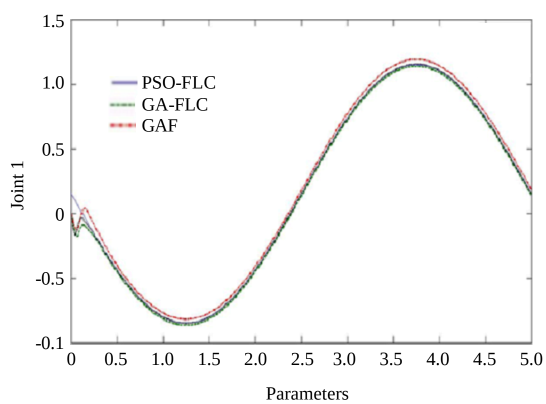

Fig. 12: The comparison of the reference detection for the three controllers (first joint)

In the present study, we made use of the presented algorithm to optimize the centers and the extensions of the membership functions on the input dimensions and the fuzzy rule consequences values and scaling factors. To optimize the controllers we can make use of various scales. In the present study, we have made use of mean square error as the objective function.

Below the objective function diagram has been given based on the number of the algorithm iteration stages for each of the algorithms for comparison purposes (Fig. 11-13).

As it is evident from the Fig. 11, we have made use of scaling coefficients in GFS and it indicates better final optimization values in comparison to the ordinary genetic algorithms. Another considerable theme in this figure is that the genetic algorithm firstly follows a more accelerated reductive procedure and it does not indicate an acceptable procedure after the tenth iteration in figuring out the better responses. But, the GFS controllers indicate a better performance and efficiency respective to the GA-FLC controllers.

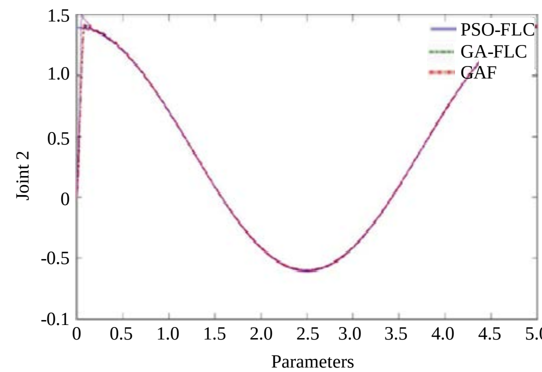

Fig. 13: The comparison of the reference detection for the three controllers (second joint)

Table 2: The control scales comparisons for the first joint

GFS	GA-FLC	FLC	Joint1
0.215	0.405	0.853	ITAE
0.112	0.176	0.392	IAE

Table 3: The control scales comparisons for the second joint

GA-FLC	PSO-FLC	FLC	Joint 2
0.045	0.063	1.806	ITAE
0.066	0.076	2.681	IAE

In the end, to precisely compare the controllers' performances we have made use of ITAE1 and IAE2 which are scales for detection errors through taking advantage of MATLAB instructions and they have been reported in Table 2 and 3. The reported values in Table 2 and 3 indicate that the GFS controllers outperform the other controllers regarding ITAE and IAR scales.

CONCLUSION

In the present study firstly we dealt with the survey of the fuzzy systems and in a separate chapter we handled the discussions regarding such systems and their advantages and disadvantages were also presented. The most important advantages regarding the fuzzy controllers are:

- The fuzzy logic is very tangible and understandable intuitively. Since, it imitates the human controlling strategy and its basics can be easily understood for the non-specialists
- It is flexible and changes can be made with ease
- It is durable against imprecision and uncertainties
- It can be made by the use of the specialists' latest experiences and knowledge (regarding the processes)
- Fuzzy controllers are based on the natural human language and therefore they can be easily understood and they have the capacity to be applied in parallel to the traditional control systems

- It allows you to introduce your own system by making use of simple expressive rules and so there is no need for the complicated mathematical formulas
- Using the membership function and rules, the continuous functions can be approximated by any arbitrary approximation degree
- Regarding the nonlinear systems, it is more superior to the nonlinear controllers since it is closer to the system's real model
- It is less expensive in respect to the traditional controllers. Since, fuzzy controller can be easily understood, the method learning time is very short that means that the "software costs" become smaller

But, according to the advantages cited above the fuzzy systems suffer from one thing and that is the need for a specialized expert in fuzzy system designing for every specific field of application. As it was stated previously, fuzzy controllers tuning and regulation including the membership functions for controlling certain systems is very time-consuming and labor-intensive task. To solve such a problem various methods have been created. One such method is the use of evolutionary algorithms in tuning and optimizing the fuzzy controllers and this was thoroughly discussed in section four which contains firstly a general overview of the discussion and then it specifically deals with the genetic algorithm details. Afterwards, it was dealt with the introduction of the most popular types of the fuzzy systems via the evolutionary algorithms that is the fuzzy genetic algorithms and the various methods of mixing the fuzzy systems with Genetic algorithms were surveyed. In the end, the two degrees of freedom controls for robot manipulators' equations were implemented on MATLAB and then the tuning and optimization of the fuzzy controllers for the robotic arm system was handled by the use of Genetic algorithm.

The results obtained in the present study by taking advantage of the fuzzy genetic algorithm are acceptable and the optimized fuzzy controllers have well accomplished the joint reference point detection. In the meantime, it was shown that the optimized controllers by the genetic algorithms provides for a better set of results. Finally, the three controllers' efficiency were compared from the perspective of the two popular and common control scales and the results were tabulated in the above tables in the present study.

REFERENCES

Bonarini, A., 1996. Evolutionary Learning of Fuzzy Rules: Competition and Cooperation. In: Fuzzy Modelling. Witold, P. (Ed.). Springer US, USA., pp: 265-283.

Chien, B.C., J.Y. Lin and T.P. Hong, 2002. Learning discriminant functions with fuzzy attributes for classification using genetic programming. Expert Syst. Appl., 23: 31-37.

Cordon, O., 2011. A historical review of evolutionary learning methods for Mamdani-type fuzzy rule-based systems: Designing interpretable genetic fuzzy systems. Int. J. Approximate Reasoning, 52: 894-913.

Cordon, O., F. Gomide, F. Herrera, F. Hoffmann and L. Magdalena, 2004. Ten years of genetic fuzzy systems: Current framework and new trends. Fuzzy Sets Syst., 141: 5-31.

Cordon, O., F. Herrera and P. Villar, 2001. Generating the knowledge base of a fuzzy rule-based system by the genetic learning of the data base. IEEE. Trans. Fuzzy Syst., 9: 667-674.

Gonzblez, A. and R. Perez, 1999. SLAVE: A genetic learning system based on an iterative approach. IEEE. Trans. Fuzzy Syst., 7: 176-191.

Hoffmann, F. and G. Pfister, 2013. Evolutionary design of a fuzzy knowledge base for a mobile robot. Intl. J. Approx. Reason., 17: 447-469.

Homaifar, A. and E. McCormick, 1995. Simultaneous design of membership functions and rule sets for fuzzy controllers using genetic algorithms. IEEE. Trans. Fuzzy Syst., 3: 129-139.

Ishibuchi, H., T. Nakashima and T. Murata, 1999. Performance evaluation of fuzzy classifier systems for multidimensional pattern classification problems. IEEE Trans. Syst. Man Cybern. B Cybern., 19: 601-618.

Karr, C., 1991. Genetic algorithms for fuzzy controllers. J. AI Expert, 6: 26-33.

Liska, J. and S.S. Melsheimer, 1994. Complete design of fuzzy logic systems using genetic algorithms in fuzzy systems, 1994. Proceedings of the Third IEEE Conference on IEEE World Congress on Computational Intelligence, June 26-29, 1994, IEEE, Orlando, FL., pp: 1377-1382.

Lughofer, E., 2011. Evolving Fuzzy Systems-Methodologies, Advanced Concepts and Applications. Springer, Berlin, Heidelberg, ISBN: 978-3-642-18086-6.

Magdalena, L. and F. Monasterio-Huelin, 1997. A fuzzy logic controller with learning through the evolution of its knowledge base. Intl. J. Approx. Reason., 16: 335-358.

Mendel, J.M., 2001. Uncertain Rule-Based Fuzzy Logic Systems: Introduction and New Directions. Prentice Hall, Upper Saddle River, New Jersey, USA., pp: 265-272.

Ng, K.C. and Y. Li, 1994. Design of sophisticated fuzzy logic controllers using genetic algorithms in fuzzy systems 1994. Proceedings of the Third IEEE Conference on IEEE World Congress on Computational Intelligence, June 26-29, 1994, IEEE, Orlando, FL., pp: 1708-1712.

Pham, D.T. and D. Karaboga, 1991. Optimum design of fuzzy logic controllers using genetic algorithms. J. Sys. Eng., 1: 114-118.

Subbu, R., A.C. Sanderson and P.P. Bonissone, 1998. Fuzzy logic controlled genetic algorithms versus tuned genetic algorithms: An agile manufacturing application. Proceedings of the IEEE International Symposium and Intelligent Control (ISIC) on Computational Intelligence in Robotics and Automation (CIRA), Intelligent Systems and Semiotics (ISAS), September 14-15, 1998, IEEE, Gaithersburg, Maryland, ISBN: 0-7803-4423-5, pp: 434-440.

Thrift, P.R., 1991. Fuzzy logic synthesis with genetic algorithms. Proceedings of the Fourth International Conference on Genetic Algorithms ICGA, July 13-16, 1991, Philip thrift, Dallax, Taxas, ISBN:1-55860-208-9, pp: 509-513.

Zadeh, L.A., 1965. Fuzzy sets. Inform. Control, 8: 338-353.

Zadeh, L.A., 1975. The concept of a linguistic variable and its application to approximate reasoning-I. Inf. Sci., 8: 199-249.