FUZZY SYSTEM IDENTIFICATION FOR Ni-Cd BATTERY CHARGER USING GENETIC ALGORITHM

Ankita Anand
Student of E.C.E
Doaba Institute of Engineering &Technology, Kharar Mohali, Punjab, India

Kamaljeet Kaur
Seth Jai Parkash Mukand Lal Institute of Engg & Technology. Radaur Distt Yamuna Nagar, Haryana, India

Parminder Malhotra
Doaba Institute of Engineering &Technology, Kharar
Mohali, Punjab, India

Abstract

The identification of model is one of the key issues in the field of system modeling. The identification of model using input-output data consists of two parts: structure identification which involves the determination of antecedent and consequent variables and in parameter estimation step, antecedents' membership fictions and rule consequents are determined. This paper presents the identification of fuzzy system for rapid Nickel-Cadmium (Ni-Cad) batteries charger by applying Genetic algorithm on the input-output training data. First of all a model for the system is designed. For the design, Input output variables are divided into fuzzy regions and shape of membership functions are decided and then Parameter estimation of antecedent part of model is performed, in which best values for model parameter are identified using Genetic Algorithm.GA is global search algorithm based on mechanics of natural selection, which combines survival of fittest. It operates on population of solution by applying survival of fittest to find better solutions from all given population. Instead of operating directly on the parameters of model, GA works with random population of solution (random population of string) representing parameters of model. In this paper, we have successfully applied GA for optimization and Simulation results shows that MSE of system designed is 0.17and with increase in number of iterations, system performance gets improved.

Keywords
Genetic algorithm, System identification.

1. Introduction

Fuzzy modeling is the task of identifying the parameters of fuzzy inference system so as to achieve desired behavior .The identification is one of the key issues in field of system modeling. Developing mathematical models of real systems is a central topic in many disciplines of engineering and science. Models can be used for simulations, analysis of the system's behavior, better understanding of the mechanisms in the system, design of new processes. Precise mathematical models sometimes may not give satisfactory results in describing behavior of complex systems, In such cases fuzzy models are used to express the uncertainty of system, thus fuzzy model identification is important.

The ability to deal with both linguistic and numerical information in systematic and efficient manner is one of the important advantages of fuzzy models [1]. In beginning concept of fuzzy set and fuzzy logic encountered criticism, however large number of successful industrial fuzzy logic applications generated an increased interest in fuzzy logic.

The identification includes following issues [2]: 1) Selecting the type of fuzzy model, input output variables for fuzzy model. 2) Choosing the structure of membership function and determining the number of fuzzy rules. 3) Identifying the parameters of antecedent and consequent membership functions. 4) Defining Performance criteria for evaluating the performance. The problem of identification can be grouped into 3 categories:

Structure identification: In this case some appropriate input variables from all possible input variables are selected and then membership function for each input variable is determined.

Parameter identification: involves identifying optimal parameters of antecedent membership function and consequent part of rules. There are virtually many different approaches to modeling a system using the fuzzy set and fuzzy system theories, but only the classical least squares optimization method and the general Genetic Algorithm (GA) optimization technique are generally used.

Model validation: involves testing the model based on some performance criterion.

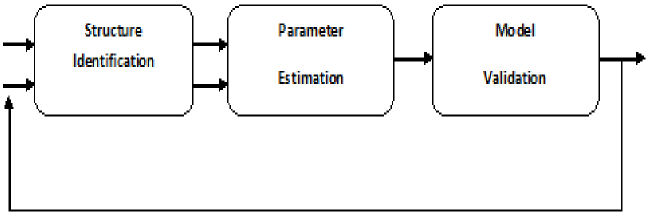

Fig 1: Fuzzy Model Identification Process

The principles of fuzzy modeling were first introduced by Zadeh in 1965 when he gave the concept of grade of membership and led to the birth of fuzzy logic technology [1]. He gave the concept, definition of fuzzy sets and membership function. During late 1990's data driven methods were developed to identify structure and parameters based on data. Wang and Mendel [3] in 1992 proposed an approach to combine fuzzy rule bases generated from numerical data and linguistic fuzzy rules. Pomaris et al. [6] provided structure identification in complete rule base fuzzy systems**.** In 1993 Sugeno and Yasukawa's qualitative modeling (SY) method [2] has gained much attention in the fuzzy research field mainly due to its advantage of building fuzzy rule bases automatically from sample input-out data.

In order to maintain this type of structure identification, many researchers employ clustering techniques. As explained in[2] by using fuzzy clustering to identify a system we do not have to consider the fuzzy partition of the input space, since it is obtained as a direct result of fuzzy clustering. Thus, the main advantage of fuzzy clustering is that the structure identification is only related to the number of rules. Different clustering techniques are discussed in [7,8,9]. Based on the concept of qualitative modeling, A combined approach for discrete-time fuzzy model identification is proposed [4]. By this approach, the identification is performed in two stages. First, the linguistic approach is utilized to obtain an approximate fuzzy relation from the sampled on fuzzy input-output data. This approximate fuzzy relation is then used as the initial estimate for the second stage in which a more accurate fuzzy relation is determined.

These are the techniques that make use of human knowledge and deductive processes. One new approach called artificial neural networks, on the other hand, realizes learning capabilities by imitating the functioning of biological neural systems on a simplified level. Vishwas Puttige [10] provides real-time system identification of an unmanned aerial vehicle (UAV) multi input and multi output system based on multiple neural networks.

A general identification approach for multi-input/single-output fuzzy systems is then presented by Chen-Wei Xu [5], which includes structure identification, parameter estimation, and the associated self-learning algorithm.

However some methods typically fail to properly handle system identification when the structure and parameters are completely unknown. With regard to this problem, a new method for identifying system was developed, thus the problem of system identification is converted into a problem of combinatorial optimization. Then we use the genetic algorithm to solve this problem. They were invented in 1975 by John Holland of the University of Michigan. After David Goldberg gave a basic framework of GAs in his popular book Genetic Algorithms in Search, Optimization & Machine Learning [11], they have received considerable and increasing interest.

GA is search and optimization technique inspired by two biological principles, the process of "natural selection" and the mechanics of "natural genetics". Tom V Mathew provide introduction, along with background and basic principles of genetic algorithm. An overview of application of genetic algorithm in control engineering problems is given by Wang et al. [12] Several other applications of GA like Image enhancement, in real world problems and finger print registration etc are provided in references [13, 14, 15] .A recent application of GA is to improve automated music composition [16]. Several other techniques like Particle Swarm optimization which is member of the broad category of swarm intelligence techniques based on metaphor of social interaction [17], Ant Colony Optimization(ACO) [18] also based on group of swarm intelligence technique, other technique Biogeography based optimization (BBO)[19] are suggested for fuzzy model identification.

1.1 Empirical Requirement of Fuzzy Modeling or Fuzzy Identification

1) Adequate processing of imprecise information:

When Parameters and associated data are not properly known, in that case Precise and conventional mathematical models cannot be used. For such cases, where some kind of uncertainty is there, Conventional models and stochastic approach is the best solution. Number of such approaches are available .Fuzzy logic is one of them used. It is a traditional way of dealing with uncertainty.

2) Transparent modeling and identification:

Fuzzy modeling is the task of identifying parameters of system designed using fuzzy logic, so as to achieve desired behavior. Many real-world systems are inherently nonlinear and cannot be represented by linear models used in conventional system identification. From the input-output view, fuzzy systems are flexible mathematical functions which can approximate other functions or just data (measurements) with a desired accuracy.

3) Incomplete or vague knowledge about systems.

Conventional system theory relies on crisp mathematical models of systems, For some systems such models can be defined, However there are certain systems for which adequate and precise models cannot be defined. A significant portion of information about these systems is available as the knowledge of human experts, process operators and designers. This knowledge may be to vague and uncertain to be expressed by mathematical functions. It is, however, often possible to describe the functioning of systems by means of natural language, in the form of if-then rules. Fuzzy rule-based systems can be used as knowledge-based models constructed by using knowledge of experts in the given field of interest.

2. Genetic Algorithm

A genetic algorithm (GA) is an optimization technique based on evolutionary idea of natural selection and genetics inspired by Darwin's theory of evolution "Survival of fittest" The basic principles and the mechanisms of GAs were first laid down by John Holland (Holland, 1962; Holland, 1975). He used the idea of the mechanisms of natural adaptation into computer systems and developed the genetic algorithm which was a major innovation. Holland in his book "Adaptation in Natural and Artificial Systems" in 1975, described the framework for adaptation of GA. In GA, a potential solution to a problem is an individual and can be represented by a set of parameters which are regarded as the genes of a chromosome.

A positive value, known as fitness value, is used to find out that how good an individual is as compared to other solution in the population. Higher the fitness values of an individual, higher its chances of survival and reproduction. During the optimization process, the algorithm alters the chromosomes, the of the population members using genetic operators. They can be divided into two main categories, namely mutation and crossover operators. Crossover is to recombine the selected parents to generate new sample points in search space. Mutation operator is applied after crossover, by which one or more bits of a parent chromosome are inverted randomly to ensure genetic diversity within the population. It is expected that with the help of this process a better chromosome will create a large number of offspring which in the end has higher chances of survival. This cycle is repeated until a desired termination condition is reached.

Genetic Algorithms are different from most of the traditional optimization methods. GA's work with population of solutions instead of single solution .Since there is more than one string which is processed simultaneously and used to operate one string in the population; it is very likely that expected solution is a global solution and also probabilistic rules are used to guide the search [20].

2.1 A Working Principle

First of all, the optimization problem to be solved is defined clearly and the possible candidate solutions are usually represented in bit string form. An appropriate fitness function is defined for the selection process. Then the optimum solution is approached by using the following steps that describe working of GA in brief.

Main Steps:

Step 1: Start with a randomly generated initial population $P(0)$ of candidate solutions.

Step 2: Calculate the fitness $f(p)$ for each individual p in the current population $P(t)$ and save these fitness values to decide about the "fittest" individuals.
The objective function, function to be optimized provides mechanism for evaluating each string. In general, a fitness function F(i) is first derived from the objective function and used in successive genetic operations. It is normally used to transform the objective function value into a measure of relative fitness.
Mathematically,

$$F(x) = g(f(x)) \qquad (1)$$

where *f* is the objective function, *g* the transform that maps the value of *f* to a nonnegative number, and *F* is the resulting relative fitness. Individuals with higher fitness value will have higher probability of being selected as candidates for further examination. Number of transformations required to map objective function into fitness function to obtain non negative value for certain genetic operators is discussed in [22].

Step 3: Using the fitness values obtained in step 2, select the population called 'parents'.

Step 4: Apply three genetic operators namely selection, crossover and mutation on the selected parents to generate a new population called next generation.

Selection Operator:

Reproduction operator is also called as selection operator. Selection means extracting subset of genes from existing population, according to any definition of quality [21]. In principle, individuals from the population are copied to a "mating pool", according to the principle highly fit individuals get more chance to be copied than unfit individuals, that fitter the parent, the more times it is likely to be selected to reproduce. There are many types of selection criteria. The most commonly type method and also used in this work is the

Roulette-wheel selection:

In which parents are given a probability of being selected that is directly proportional to their fitness Thus, the ith string in the population is selected with a probability proportional to Fi. Since the population size is usually kept fixed in a simple GA, the sum of the probability of each string being selected for the mating pools must be one. Therefore, the probability for selecting the ith string is

$$P_i = f_i \,/ \textstyle\sum_{i=1}^{n} \quad fi \qquad (2)$$

where n is the population size. One way to implement this selection scheme is to imagine a roulette-wheel with it's circumference marked for each string proportionate to the string's fitness. roulette-wheel is spun n times. Since the circumference of the wheel is marked according to a string's fitness, this roulette-wheel mechanism is expected to make f_i/f' copies of the ith string in the mating pool. The average fitness of the population is calculated as

$$f' = \textstyle\sum_{i=1}^{n} \quad fi \qquad (3)$$

Using the fitness value f_i of all strings, the probability of selecting s string pi can be calculated. Thereafter, the cumulative probability Pi of each string being copied can be calculated by adding the individual probabilities from the top of the list. In order to choose n strings, n random numbers between zeros to one are created at random. Thus, string that represents the chosen random number in the cumulative probability range (calculated from the fitness values) for the string is chosen to the mating pol. This way the string with a higher fitness value will represent a larger range in the cumulative probability values and therefore has a higher of being copied into the mating pool. On the other hand, a string with a smaller fitness value represents a smaller range in cumulative probability values and has a smaller probability of being copied into the mating pool.

Crossover Operator:

After the construction of the intermediate population using selection operator, the next step is to recombine selected parents through a process called 'crossover' to generate new sample points (offspring) in the search space. It helps to explore the most promising regions of search space that leads to converge to the final solution in a more purposeful, efficient and directional way.

One – point crossover is the oldest and most commonly used and will be explained in this work. There are some alternatives to one – points crossover among which there are two – point crossover and uniform crossover (Syswerda), which can avoid some disadvantages that occur in one – point crossover. In single point crossover, a locus is selected randomly and the subsequences before and after that locus between two chromosomes (parents) are exchanged to produce two new offspring. For example, if we consider two individuals (Parents) with eleven binary variables each

Table 1: Population before crossover is applied

Parent 1	10001110011
Parent 2	11100110111

Suppose, locus selected randomly is at 5th position, so exchanging the subsequences before and after the locus point between these two parents, the following two offspring come into existence.

Table 2: Population after crossover

Offspring 1	10001110111
Offspring 2	11100110011

A proper selection of a crossover operator plays an important role in the success of an application using a genetic algorithm.

Mutation Operator:

After crossover, mutation operator is applied by which one or more bits of a parent chromosome are inverted randomly to drag the candidate solutions away from trapping in local extremum, which otherwise cannot be ensured by crossover alone. In fact, mutation is vital to ensure genetic diversity within the population. In mutation process, the locus is selected randomly and bit value at that position is just inverted. A simple example is given below for more clarification.

Table 3: String of population before and after mutation

Before Mutation	10**0**110
After Mutation	101110

In this example, randomly position 3rd is chosen as locus, so third bit from left is changed after mutation.

Step 5: Repeat steps 2 to 4 until optimal or near optimal (i.e. satisfying) solution are obtained.
These are different steps which explain working of GA to obtain optimized result. Pseudo code for GA is given as follows:

```
Begin
    Define parameters for GA
    Generation =0
     Initialize Population P
     While (Termination criterion)
          Evaluate p
          A=Select [P]
          B=Crossover [P]
          C=Mutation [P]
          P=Merge [A, B, C]
          Generation=Generation +1
     End
End
```

Fig 2: Pseudo code for genetic algorithm

3. Identification of Fuzzy Model Through GA

Fuzzy modeling is the task of identifying the parameters of a fuzzy inference system so that a desired behavior is attained due to linguistic and numeric requirements; the fuzzy modeling process has generally to deal with an important trade-off between the *accuracy* and the *interpretability* of the model. The first fuzzy modeling works were very similar to, and inspired by, the knowledge engineering methods used in expert systems. The increasing availability of input-output data of the modeled processes, which is not specifically used to determine the structure or the parameters of the fuzzy model in the direct approach, motivated the use of more automatic approaches to fuzzy modeling, in which only a part of the fuzzy model is built from *a priori* knowledge .Different modeling or identification approaches are discussed in [23].

GA like other evolutionary algorithms has capability to find optimal or near optimal solution in given complex search space and can be used to modify parameters of fuzzy models. Evolutionary algorithms offer a number of advantages over other search methods such as they integrate elements of direct and stochastic search. Also they do not require any auxiliary information except objective function values.
The application of GA for fuzzy model identification involves number of important considerations. The first step in applying such an algorithm is to create a random population of solution The next important step is to define an objective function which represents quality of solution. Mean Square Error (MSE), has been used for rating the quality of fuzzy model.

$$\mathrm{MSE} = \frac{1}{N}\sum_{k=1}^{N}[y(k) - \tilde{y}(k)]^2 \qquad (4)$$

Where,

$y(k)$ = Actual output

$\tilde{y}(k)$ = Computed output of model

N = Number of data points taken for model validation

For the Purpose of encoding, we consider two inputs, Temperature and temperature gradient divided into 3 and 2 fuzzy sets respectively, and single output Charging rate divided into 5 fuzzy sets. First and last membership function of each input and output variable are represented with Z-type and sigma type membership functions respectively. The Procedure for identification of fuzzy model through GA can be represented in the form of flow chart as shown in figure 3:

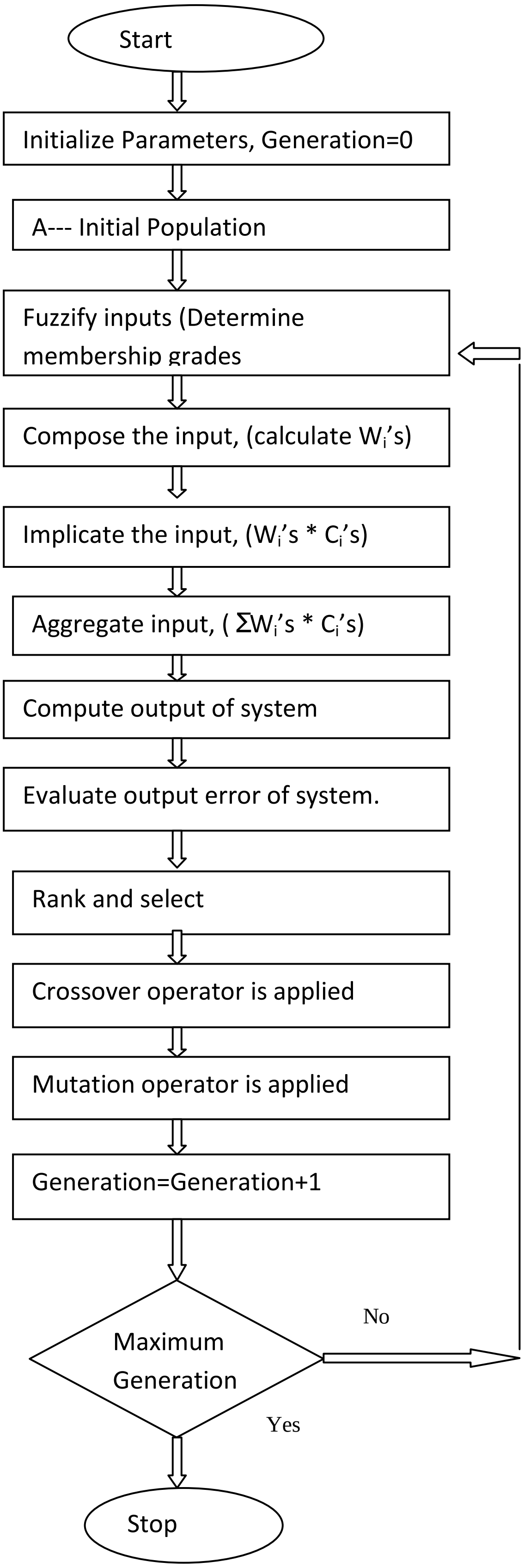
Start
Initialize Parameters, Generation=0
A--- Initial Population
Fuzzify inputs (Determine membership grades
Compose the input, (calculate W$_i$'s)
Implicate the input, (W$_i$'s * C$_i$'s)
Aggregate input, (ΣW$_i$'s * C$_i$'s)
Compute output of system
Evaluate output error of system.
Rank and select
Crossover operator is applied
Mutation operator is applied
Generation=Generation+1
Maximum Generation
No
Yes
Stop

Fig 5. Methodology for identification of fuzzy models using GA

4. Problem Formulation

Sugeno type fuzzy systems consists of 4 modules Fuzzifier, Rule composition module (means 'MIN' operator), implication module, and defuzzification module. Overall computed output in this case can be written as :

$$\text{Computed output} = \sum_i (w_i * c_i) / \sum w_i \qquad (5)$$

For any set of inputs w_i are easily computed by fuzzifier, the right hand side of above equation can be evaluated if proper values for c_is are chosen. For given data set of a system values of c_i are chosen such that difference between computed and actual output as given in data is minimum.

Computed output and actual output as given in data set is compared and error is calculated.

Error E = Actual Output as given in data set – Computed Output (6)

Very next step is to minimize objective function E.

Subject to constraint that C_i belongs to specified set of consequence. (7)

4.1 Application to battery charger

GA has been applied for identification of fuzzy model for rapid Ni-Cd battery charger. The main objective is to charge the batteries as quickly as possible but without doing any damage to them. The two input variables to control charging Rate (Ct) are absolute temperature (T) and its Temperature gradient. The input output variables along with universe of discourse are listed in table 4.

Table 4: Input output variables for battery charger with Universe of discourse

INPUT VARIABLES	MINIMUM VALUE	MAXIMUM VALUE

Temperature	0	50˚C
Temperature gradient	0	1
OUTPUT VARIABLE		
Charging Current	0	4A

The Sugeno type model for battery charger with two inputs temperature and temperature gradient and output to be controlled is charging current. Rules for model designed, are

1) If Temp is LOW and Temp gradient is LOW then Charging current is ULTRAFAST.
2) If Temp is LOW and Temp gradient is HIGH then Charging current is ULTRAFAST.
3) If Temp is MED and Temp gradient is LOW then Charging current is HIGH.
4) If Temp is MED and Temp gradient is HIGH then Charging current is HIGH.
5) If Temp is HIGH and Temp gradient is LOW then Charging current is LOW.
6) If Temp is HIGH and Temp gradient is HIGH then Charging current is TRICKLE.

Input Temperature with universe of discourse 0-50 degree centigrade has been partitioned into 3 fuzzy sets Low, Medium and High. Temperature gradient is divided into two fuzzy sets Low and High. Initially parameters of membership functions of input are set to arbitrary values. Once fuzzification of input is carried, 6 combinations of input membership function representing 6 antecedents of rule are obtained. These 6 rules form the rule base for the system under identification. The specified set of consequents in this particular case is C_1 = Low=1 Amp, C_2 = High = 3 Amp, C_3 = Ultrafast = 4 Amp, C_4 = Trickle = 0.01 Amp. Parameters of antecedent and consequents are chosen in such a way that they satisfy condition given by expression 5.

Parameters of input Temperature and temperature gradient before optimization is shown in figure

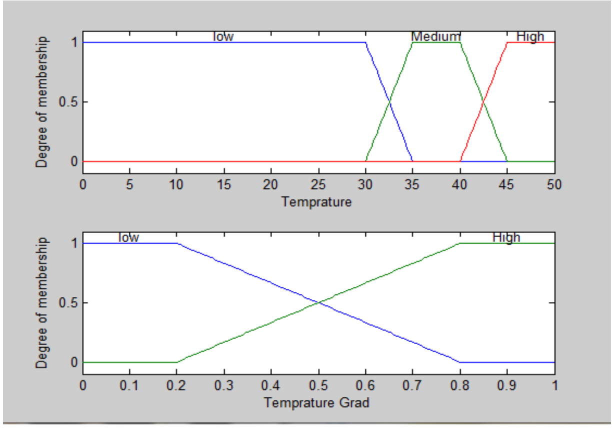

Fig 3: Membership functions before optimization

4.2 Simulation results

Complete methodology used to obtain results is implemented in Matlab m-file. Operating parameters used for identification of model are a) Number of Individuals = 300, b) Number of iterations = 500.Membership functions for input temperature and temperature gradient after optimization are shown in figure 4 below:

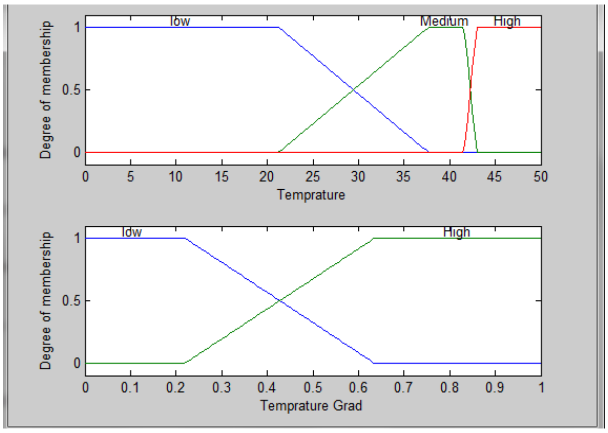

Fig 4: Membership functions after optimization

Simulation results shows that fuzzy model with tuning of membership function for 100 iterations lead to **MSE of 0.3573** ,however on increasing the iterations, system performance gets improved, as for 500 iterations MSE of model gets reduced to **0.1719.**A graph between number of iterations and average error of system is shown below:

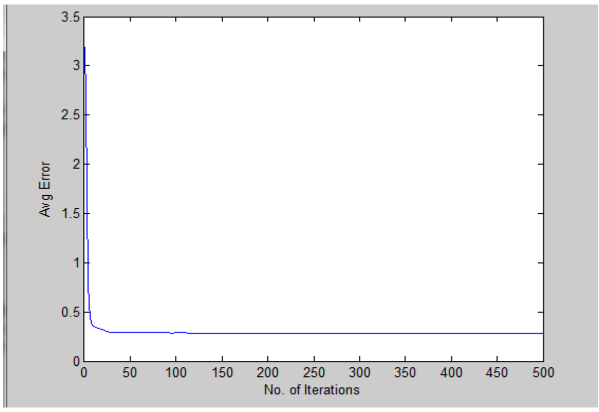

Fig 5: Graph showing improvement in system with increase in number of iterations

It is clear from the simulation results and graph shown, that with increase in number of iterations, system performance gets improved. Weighted average defuzzification technique was selected for singleton fuzzy model.

5. Conclusion

This paper shows the application of Genetic Algorithm for optimization problem. First of all, problem is encoded into number of solutions or individuals representing the parameters to be optimized. The difference between computed output and actual output given in training example gives the error. This error is an objective function which represents the quality of solution to find out best optimal solution out of number of solutions available for the problem. In this paper we have successfully applied GA to optimize the membership function .Simulation results shows that MSE (Mean square error, which is used for rating quality of model), for 500 iterations and 300 numbers of individuals is 0.1719.

6. References

[1] L.A. Zadeh, “Fuzzy sets”, Information and control, Vol. 8, pp. 338-353, 1965.

[2] M. Sugeno and T. Yasukawa, “A Fuzzy logic based approach to qualitative modeling ,” IEEE Transactions on fuzzy systems, Vol. 1, pp.7-31, 1993.

[3]Li-Xin Wang and Jerry M. Mendel, “Generating fuzzy rules by learning from examples, “IEEE transaction on systems, Man and cybernetics.vol.22, No.6, pp. 1414-1427, 1992.

[4] Ying-Chin Lee, Chyi Hwang, and Yen-Ping Shih, “A Combined Approach to Fuzzy Model Identification”, IEEE transactions on systems, man, and cybernetics, vol. 24, no 5, may 1994.

[5] Chen-Wei Xu, “Fuzzy systems identification,” IEEE proceedings, vol. 136, pt. d, no. 4, july 1989.

[6] Hector Pomares, Ignacio Rojas, Jesús González, and Alberto Prieto, “Structure Identification in Complete Rule-Based Fuzzy Systems “ , IEEE Transactions on fuzzy systems, vol. 10, no. 3, june 2002.

[7] George Tsekourasy, Haralambos Sarimveis and George bafas, “A method for fuzzy system identification based on clustering analysis”, SAMS, 2002, Vol. 42(6), pp. 797–823.

[8] Di Wang, Xiao-Jun Zeng and John A. Keane2, “An Input-Output Clustering Method for Fuzzy System Identification”, IEEE Transactions on fuzzy systems, 2007.

[9] Arun Khosla Shakti Kumar K.K. Aggarwal,” Identification of Fuzzy Controller for Rapid Nickel-Cadmium Batteries Charger through Fuzzy c-means Clustering Algorithm”, IEEE Transactions on fuzzy systems, 2003.

[10] Vishwas Puttige, Sreenatha Anavatti, “Real-time System Identification of Unmanned Aerial Vehicles: A Multi-Network Approach”, Journal of computers, vol. 3, no. 7, july 2008.

[11]Goldberg, D.E. Genetic Algorithms in Search, Optimization and Machine Learning, Addison- Wesley. 1989.

[12] Qing Wang, Pieter Spronck, Rudolf Tracht’ “An overview of genetic algorithms applied to control engineering problems”, Proceedings of the Second International Conference on Machine Learning and Cybernetics, Xi’an, 2-5 November 2003.

[13] Mantas Paulinas, Andrius Ušinskas, “A survey of genetic algorithms applications for image enhancement and segmentation”, issn 1392 – 124x information technology and control, 2007, vol.36, no.3

[14] Prof. Swati V. Chande, Dr. Madhavi Sinha, “Genetic algorithm: a versatile optimization tool”, Bharati Vidyapeeth’s Institute of Computer Applications and Management, New Delhi.

[15] Ibrahiem M.M. El-Emary and 2Mona M. Abd El-Kareem, “On the Application of Genetic Algorithms in

Finger Prints Registration", World Applied Sciences Journal 5 (3): 276-281, 2008

[16] Nathan Fortier, Michele Van Dyne, "A Genetic Algorithm Approach to Improve
Automated Music Composition", International journal of computers Issue 4, volume 5, 2011.

[17] Arun Khosla , Shakti Kumar and K.K. Aggarwal, "A Framework for Identification of Fuzzy Models through Particle Swann Optimization Algorithm", IEEE Indicon. 2005 Conference, Chennai, India, II - 1 3 Dec-2005,pp-388-391.

[18] Shakti Kumar, Parvinder Kaur and Amarpratap Singh, "Fuzzy System Identification: An ACO Based Approach", Proceedings of the workshop on intelligent System Engineering ,ISTK ,Kalawad, Yamunanagar, India, July 26-30,pp 93-105,2010.

[19] Shakti Kumar, Parvinder Kaur and Amarpratap Singh, "Fuzzy Rule Base Generation Part II : BBO Approach", Proceedings of the workshop on intelligent System Engineering ,ISTK ,Kalawad, Yamunanagar, India, July 26-30,pp 83-91,2010.

[20] Kalyanmoy Deb, "An introduction to Genetic Algorithm" , Sadhana, Vol 24, Parts 4 and 5 , August and October 1999, pp 293-351.

[21]http/: Rk Chakraborthy, www.myreaders.com

[22] Goldberg David E, "Genetic Algorithms in search, optimization and machine learning", Pearson Education, 2003.

[23] Disha Sharma, "Designing and Modeling Fuzzy Control Systems ",International Journal of Computer Applications
Volume 16– No.1, February 2011