International Journal of Soft Computing 2 (5): 599-605, 2007
ISSN: 1816-9503

Genetic Algorithm Based Gain Scheduled Fuzzy Logic Controller for an Unknown Process

[1]P. Melba Mary and [2]N.S. Marimuthu
[1]National College of Engineering, Anna University,
Maruthakulam Tirunleveli District,TamilNadu, India
[2]GCE, Anna University, Tamilnadu, India

Abstract: This study proposes a novel approach based on genetic algorithm optimized fuzzy logic controller, for the design of a temperature control process, capable of providing optimal performance over the entire operating range of the process. Since an optimum response of the Fuzzy Controller can be expected only for a limited range of inputs, here tuning the input , output gains and the scaling factor are done for other ranges of inputs. The proposed control system combines the advantages of Genetic Algorithm and Fuzzy Logic Control schemes. In order to evaluate the performance of the proposed control system methods, results from simulation of the process are presented.

Key words: Discrete time system, Fuzzy Inference Systems (FIS), fuzzy logic, Fuzzy Logic Control (FLC), gain scheduling, Genetic Algorithm (GA)

INTRODUCTION

Fuzzy logic control systems, which have the capability of transforming linguistic information and expert knowledge into control signals, as explained by Lee (1990a, b) are currently being used in a wide variety of engineering applications. The simplicity of designing these fuzzy logic systems has been the main advantage of their successful implementation over traditional approaches such as optimal and adaptive control techniques. Genetic Algorithm can be viewed as a general purpose optimization method and have been successfully applied to search, optimization and machine learning tasks. Karr and Gentry (1993) proved in their study that to provide practical solutions to such complex problems as those found in minerals and chemical industries GA represents a technique that acts as a valuable resource for the design of FLC's.

The application of GA's to FLC holds a great deal of promise in overcoming two of the major problems in fuzzy controller design., design time and design optimality Previous work has been done mainly in 2 areas, learning the fuzzy rules and learning membership functions. Shimojima *et al.* (1995) used GA to tune a type of Radial Basis Function (RBF) based fuzzy model, with only three fuzzy memberships for each variable. Abdollah and McCormic (1995) has shown that a GA's robustness enables it to cover a complex search space in a relatively short period of time while ensuring optimal solution.

Because of this capability, GA's are a natural match for fuzzy controller. Procyk and Mamdani (1979) introduced an iterative procedure for altering membership function using GA to improve the performance of an FLC, but it is heuristic and still subjective. Jang (1992) developed a self-learning FLC based on a neural network trained by temporal back-propagation. Rajani and Nikhil (1999) developed a robust self tuning scheme for fuzzy PI and PD controllers. Similarly, GA's are also currently being investigated for the development of adaptive or self-tuning fuzzy logic control systems.

It is shown in this study, that the performance of fuzzy control systems can be improved by incorporating a GA based gain tuning. The GA facilitates the derivation of the optimal input and output gains based on either a random selection or on their initial subjective selection. The design process involves determining parameters such as input and output scaling gains. For instance, if the range of parameters is given, then the process includes encoding the ranges to a bit code string , if the range of parameters is unknown, the GA can still determine the optimal gains, by defining a wider solution parameter space.

PROBLEM FORMULATION

The continuous-time temperature control system is described by Tanomaru (1992) as

Corresponding Author: P. Melba Mary, National College of Engineering, Anna University, Maruthakulam Tirunleveli District, TamilNadu, India

$$\frac{dy(t)}{dt} = \frac{f(t)}{C} + \frac{Y_0 - y(t)}{RC} \tag{1}$$

Where t denotes time, y(t) is the output temperature in °C, f (t) is the heat flowing inward the system, Y_0, is the room temperature (constant, for simplicity), C denotes the system thermal capacity and R is the thermal resistance between the system borders and surroundings. Assuming that R and C are essentially constants, obtaining the pulse transfer function for the system in (1) by the step response criterion results in the discrete-time system

$$y(k+1) = a(T_s)y(k) + b(T_s)u(k) \tag{2}$$

Where k is the discrete-time index, u(k) and y(k) denote the system input and output, respectively and T_s is the sampling period. Denoting by α and β some constant values depending on R and C, the remaining parameters can be expressed by

$$\left.\begin{aligned} a(T_s) &= e^{-\alpha(Ts)} \quad \text{and} \\ b(Ts) &= \frac{\beta}{a}(1 - e^{-a(Ts)}) \end{aligned}\right\} \tag{3}$$

The system described in Eq. 1-3 was modified to include a saturating non-linearity so that the output temperature cannot exceed some limitation.

The simulated control plant is described by

$$y(k+1) = a(T_s)y(k) + \frac{b(T_s)}{1 + e^{0.5y(k)-\gamma}}u(k) + [1 - a(T_s)]Y_0 \tag{4}$$

Where, $a(T_s)$ and $b(T_s)$ are given by (3). The parameters for simulation are α = 1.00151E-4, β = 8.67973E -3, γ = 40.0 and Yo = 25.0°C, which were obtained from a real water bath plant. The plant input u(k) was limited between 0 and 5 volts and it is also assumed that the sampling period is limited by

$$T_s \geq 10s \tag{5}$$

With the chosen parameters, the simulated system is equivalent to a Single Input Single Output (SISO) temperature control system of a water bath that exhibits linear behavior up to about 70°C and then becomes nonlinear and saturates at about 80°C. The Schematic Diagram of the real water bath process is depicted as in Fig. 1.

A personal computer reads the temperature of the waterbath through a link consisting of a diode- based

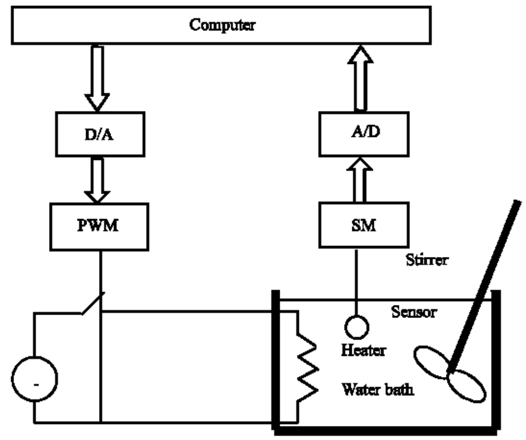

Fig. 1: Water bath control system

temperature Sensor Module (SM) and an 8- bit Analog to Digital converter. The plant input produced by the computer is limited between 0 and 5 and controls the duty cycle for a heater via a Pulse-Width-Modulation (PWM) scheme.

DESIGN PROCEDURE

Main idea of the FLC: In this study, we present the main ideas underlying the FLC. To highlight the issues involved, Fig. 2 shows the basic configuration of an FLC, taken from Lee (1990a) which comprises four principal components:

- A fuzzification interface,
- A knowledge base,
- A decision making logic,
- A defuzzification interface.

Fuzzification interface: Fuzzification is related to the vagueness and imprecision in a natural language. In fuzzy control applications, the observed data are usually crisp. Since the data manipulation in an FLC is based on fuzzy set theory, fuzzification is necessary in an earlier stage. The fuzzification module performs the following functions.

- It measures the values of input variable(s).
- Performs a scale mapping that transfers the range of values of input variable(s) into corresponding universe of discourse.
- Performs the function of fuzzification that converts input data into suitable linguistic values.

Knowledge base: The knowledge base comprises knowledge of the application domain and the control goals. It consists of a “data base’ and a ‘linguistic control

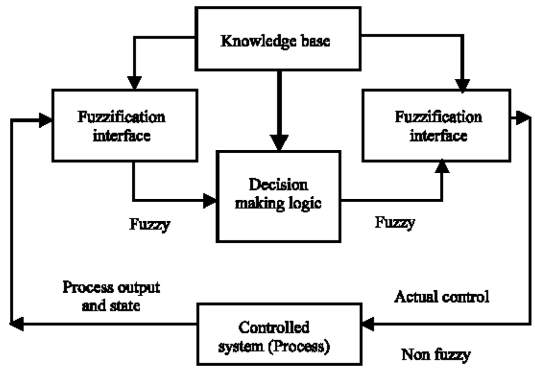

Fig. 2: Fuzzy logic controller

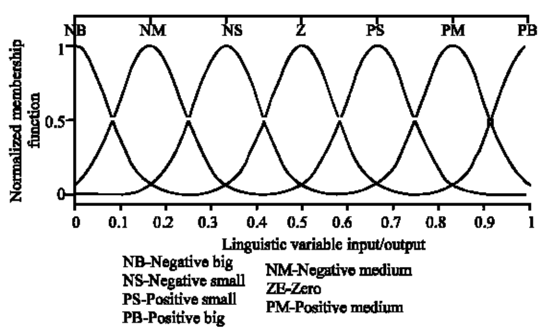

Fig. 3: Gaussian membership function

rule base'. The database provides necessary definitions, which are used to define linguistic control rules and fuzzy data manipulation in an FLC.

Decision making logic: The decision-making logic is the kernel of an FLC. It has the capability of simulating human decision-making based on fuzzy concepts and of inferring fuzzy control actions employing fuzzy implication and the rules of inference in fuzzy logic.

Defuzzification interface: The defuzzification interface performs the following functions.

- A scale mapping, which converts the range of values of output variables into corresponding universe of discourse.
- Defuzzification, which yields a non-fuzzy control action from an inferred fuzzy control action.

Membership functions: For defining the fuzzy set here we have used a functional definition of membership function. Such functions are used in FLC because they lead to themselves to manipulation through the use of fuzzy arithmetic. The functional definition can readily be adapted to a change in the normalization of a universe. The Gaussian membership function shown in Fig. 3, is used for defining the linguistic variables involved in the construction of rule base of the FLC.

Rule base structures: In an FLC, the dynamic behavior of a fuzzy system is characterized by a set of linguistic description rules based on expert knowledge. The expert knowledge is usually of the form.

IF (a set of conditions are satisfied) THEN (a set of consequences can be inferred).

Since the antecedents and the consequents of these IF-THEN rules are associated with fuzzy concepts (linguistic terms), they are often called fuzzy conditional statements. In our terminology, a fuzzy control rule is a fuzzy conditional statement in which the antecedent is a condition in its application domain and the consequent is a control action for the system under control.

Basically, fuzzy control rules provide a convenient way for expressing control policy and domain knowledge. Furthermore, several linguistic variables might be involved in the antecedents. Since in our case we have used two inputs and one output for the controller, our fuzzy system is denoted as Multi Input Single Output (MISO) system and the fuzzy control rules have the form:

R_1: if x is A_1 and y is B_1 then z is C_1
R_2: if x is A_2 and y is B_2 then z is C_2
……………………………………….
R_n: if x is A_n and y is B_n then z is C_n

Where x, y and z are linguistic variables representing two process state variables and one control variable; A_i , B_i and C_i are linguistic values of the linguistic variables x, y and z in the universes of discourse U , V and W , respectively, with i = 1,2; n and an implicit sentence connective also links the rules into a rule set or, equivalently a rule base.

Method of defuzzification: Though various methods are there for converting the inferred fuzzy control action to real value, we have used Centroid defuzzification method, a strategy which generates the centre of gravity of the possibility of a control action.

The following are the advantages of Centroid method of defuzzification over the other methods as per Lee (1990b):

- It yields superior results.
- It yields a better steady state performance and
- Yields lower mean square error.

Fuzzy inference systems: Mamdani type Fuzzy Inference system is used and the rule base framed by .Chih-Hsun Chou (2006) for the FLC is shown in Table 1, wherein E

Table 1: Inference rules for mamdani type flc

	E							
ΔE	CO	NB	NM	NS	Z	PS	PM	PB
NB		Z	NS	NS	NM	NM	NB	NB
NM		PS	Z	NS	NS	NM	NM	NB
NS		PS	PS	Z	NS	NS	NM	NM
Z		PM	PS	PS	Z	NS	NS	NM
PS		PM	PM	PS	PS	Z	NS	NS
PM		PB	PM	PM	PS	PS	Z	NS
PB		PB	PB	PM	PM	PS	PS	Z

and ΔE refer to the two inputs error and change in error to the FLC and CO denotes Controller Output. By trial and error approach 25 rules were considered out of forty nine rules for constructing the controller.

Genetic algorithm for input and output gain tuning: Genetic Algorithms (GAs) are adaptive heuristic search algorithms premised on the evolutionary ideas of natural selection and genetic. The basic concept of GAs is designed to simulate processes in natural system necessary for evolution, specifically those that follow the principles first laid down by Charles Darwin of survival of the fittest.

As such they represent an intelligent exploitation of a random search within a defined search space to solve a problem.

Though GA can be used as a flexible method for input space partitioning, it has the major disadvantage of consuming much time, we have proposed here to use GA to tuning of input output gains based on Wang *et al.* (2002) so that the convergence will be quicker and better result will be obtained compared to the plant without GA. Gain scheduling using GA's is shown in Fig.4, wherein G_e and G_d represents proportional error gain and derivative error gains, respectively in the input side of FLC and G_u is the output scaling gain.

Characteristics of genetic algorithms: GA's perform on the coding of the parameters and not on the exact parameters, therefore, it does not depend on the continuity of the parameter nor the existence of derivatives of the functions as needed in some conventional optimization algorithms.

The coding method allows GA's to handle multi parameters or multi model type of optimization problems easily, which is rather difficult or impossible to be treated by classical optimization methods.

The population strategy enables GA to search the near optimal solutions from various parts and directions within a search space simultaneously. Therefore, it can avoid converging to the local minimum or maximum points better.

GA processes each chromosome independently and makes it highly adaptable for parallel processing. It needs no more than only the relative fitness of the chromosomes; thus, it is rather suitable to be applied to systems that are ill defined. GA's can also work well for nondeterministic systems or systems that can only be partially modeled. GA's use random choice and probabilistic decision to guide the search, where the population improves toward near optimal points from generation to generation.

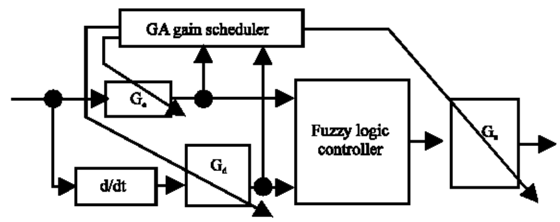

Fig. 4: Structure of the GA tuned FLC

Basic evolutionary processes: GA's consist of three basic operations reproduction, crossover and mutation which mimic the natural evolutionary processes.

Reproduction is the process where members of the population reproduced according to the relative fitness of the individuals, where the chromosomes with higher fitness have higher probabilities of having more copies in the coming generation. There are a number of selection schemes available for reproduction, such as "roulette wheel," "tournament scheme," "ranking scheme," etc. referred by Teo and Khalid (1999).

Crossover in GA occurs when the selected chromosomes exchange partially their information of the genes, i.e., part of the string is interchanged within two selected candidates.

Mutation is the occasionally alteration of states at a particular string position. Mutation is essentially needed in some cases where reproduction and crossover alone are unable to offer the global optimal solution. It serves as an insurance policy, which would recover the loss of a particular piece of information.

Parameters used in GA: The GA used here is called simple GA. We consider the problem to find the initial values of PD gains. In order to apply GA to find the initial values of PD gains, we use GA parameters as given in Table 2.

Procedural steps to do a simple GA:

- Initialize population with randomly generated individuals and evaluate the fitness value of each individual.
- Select two members from the population with probabilities proportional to their fitness values.

Table 2: Genetic parameters used in gain scheduling

GA Operators	Values
Number of variables(N)	3
Population size	10
Range of error(e)	0.05-0.07
Range of change in error (ce)	0.05-0.07
Range of output gain(G_u)	440-460
Number of bits for e and ce	20
Number of bits for output gain G_u	40
Crossover probability	0.8
Mutation probability	0.05

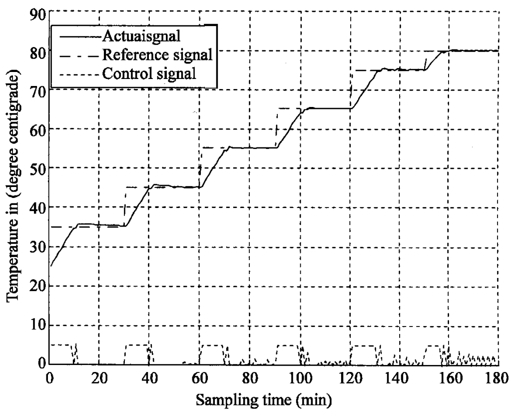

Fig. 5: Performance of the simulated system using conventional fuzzy logic controller(Ts=25s)

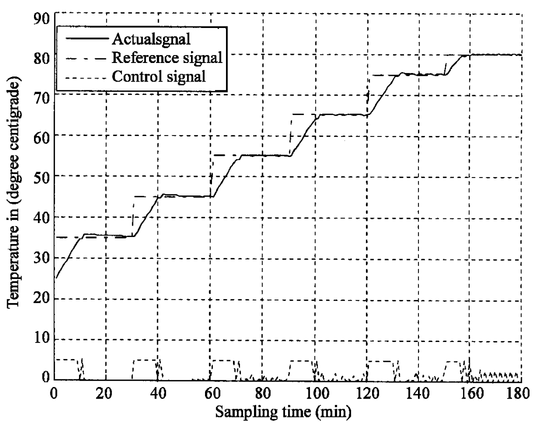

Fig. 6: Performance of the simulated system using GA tuned fuzzy logic controller (T_s=25s)

- Apply cross over with a probability equal to cross over rate.
- Apply mutation with a probability equal to mutation rate.
- Repeat the above three steps until the stopping criterion is met.

SIMULATION STUDY

Using the mathematical model of the real water bath plant our proposed approaches has been tested. From the

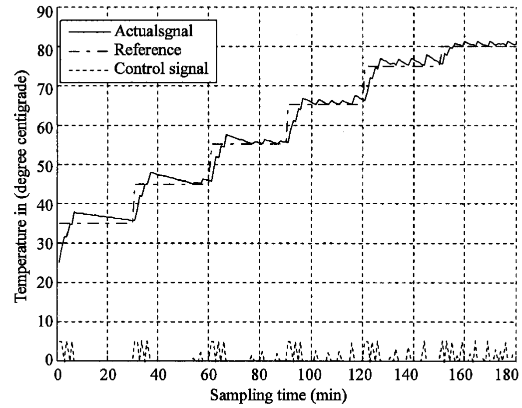

Fig. 7: Performance of the simulated system using conventional fuzzy logic controller when T_S =75s

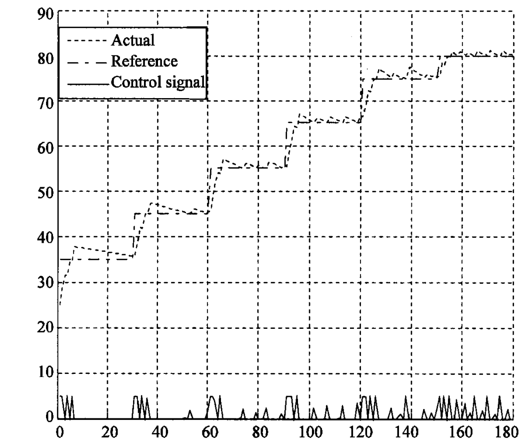

Fig. 8: Performance of the simulated system using Genetic Algorithm tuned fuzzy logic controller when T_S =75s

initial condition y(0) = Yo = 25°C, the target is to follow a control reference set to 35.0°C for 0 <=t <=30 min, 45.0°C for 30 < t <=60 min, 55.0°C for 60 < t <=90 min., 65.0°C for 90 < t <=120 min., 75.0°C for 120 < t <=150 min and 80.0°C for 150 < t <=180 min with sampling time T_s = 25 s.

The performances of the simulated water bath temperature process using the proposed approaches for T_s=25s are shown in Fig. 5 and 6.

Sampling time T_S is varied and the performance of the system is observed. Figure 7 and 8 show the response when T_S is set as 75s. When T_S is set as 25s, the response tracks the step input almost ideally. But when T_S is increased, response shows deviations from the desired step input and large oscillations are seen. We have tried and got results for various sampling times.

RESULTS

The comparison of the proposed approaches is made by calculating the performance index absolute error over the entire simulation period and is shown in Table 3. Results show that the absolute error is minimized in the design using GA tuned fuzzy controller compared with conventional fuzzy approach and self tuned fuzzy controller.

Table 4 shows the comparison of the proposed approaches choosing percent peak overshoot as the performance index. GA tuned FLC gives less percentage overshoot for all temperature ranges.

Table 5 and 6 shows the comparison of the proposed approaches choosing steady state error and settling time as the performance indices, respectively.

Table 7 shows the values of Gain scheduling parameters obtained in GA for a maximum generation of 10.

Table 3: Comparison of absolute error among the proposed methodologies T_S =25s

Methodology	Absolute error
Conventional fuzzy	306.2182
GA-tuned fuzzy	302.8760

Table 4: Comparison of peak overshoot among the proposed methodologies for different temperature ranges T_S =25s

Temperature in degree centigrade	Conventional fuzzy	GA tuned fuzzy
25-35	1.9739	0.6517
35-45	1.3515	0.9771
45-55	0.5205	0.3166
55-65	0.6715	0.4490
65-75	0.6199	0.5577
75-80	0.2518	0.1111

Table 5: Comparison of steady state error among the proposed methodologies for different temperature Ranges T_S =25s

Temperature in degree centigrade	Conventional fuzzy Controller	GA tuned fuzzy Controller
25-35	-0.2198	-0.0325
35-45	0.0261	0.0065
45-55	-0.0231	-0.0277
55-65	-0.0518	-0.3080
65-75	0.0399	-0.0841
75-80	0.0478	0.0479

Table 6: Comparison of settling time among the proposed methodologies for different temperature ranges T_S =25s

Temperature in degree centigrade	Conventional fuzzy	GA tuned fuzzy
25-35	4.1667	4.1667
35-45	4.1667	4.1667
45-55	4.1667	4.1667
55-65	4.1667	4.1667
65-75	4.5833	5
75-80	2.5000	2.9167

Table 7: Gain scheduling parameters in ga for a maximum generation of 10 T_S =25s

GAIN G_e	GAIN G_d	GAIN U	J
0.0690	0.0546	452.1369	3.2394
0.0597	0.0678	455.2419	3.2448
0.0591	0.0504	456.4281	3.2475
0.0589	0.0623	455.8387	3.2830
0.0684	0.0648	443.5253	3.2424
0.0581	0.0687	458.3381	3.2701
0.0582	0.0679	441.1578	3.3023
0.0571	0.0663	440.1972	3.2834
0.0528	0.0541	443.9744	3.2390
0.0513	0.0675	448.7844	3.2392

CONCLUSION

This study has presented a comparison performance of a conventional fuzzy and a GA tuned fuzzy controller where all of its gain parameters can be simultaneously tuned for a water bath temperature process. By appropriate coding of the FLC parameters, it can achieve self-tuning properties from an initial random state. By employing dynamic crossover and mutation probability rates, the tuning process by GA was further improved. In the simulation study, the control performance has been compared to a GA tuned FLC controller and the conventional FLC. Though it can be argued that in the proposed FLC, before GA can be used to optimize its parameters, initial encoding and settings are required, such procedures are somewhat relatively simpler and more systematic than heuristic.

In addition, this study shows the flexibility of the proposed methodology in applying the different types of performance indices (i.e., absolute error, steady state error, percentage overshoot etc.) for our simulated water bath process. The controller is adaptive for all temperature conditions and significantly improves the performance of the system, which are the effectiveness of the proposed approach. In addition to the above mentioned facts since gains are scheduled, the system will be a stable one whatever may be the gain values applied both on input and output sides and it was observed that the applied GA tuned FLC performed relatively better than the Conventional fuzzy logic controller.

ACKNOWLEDGEMENT

The authors would like to thank the Management and Principal of National College of Engineering for their constant encouragement and for having given permission to make use of the facilities available in the Electrical and Electronics Engineering laboratory which is well established with the installation of MATLAB-7 software with necessary tool boxes for doing simulation of water bath process.

REFERENCES

Chih-Hsun Chou, 2006. Genetic Algorithm based optimal fuzzy Controller Design in the linguistic space. IEEE Transaction on Fuzzy System, Vol. 14.

Homaifar, A. and E. McCormic, 1995. Simultaneous design of membership functions and rule sets for fuzzy controller using GA. IEEE. Trans. Fuzz Sys., 3: 129-139.

Julio Tanomaru, 1992. Process Control by On-Line Trained Neural Controllers, IEEE. Trans. Indus. Elec., 39: 511-521.

Jyh-Shing R. Jang, 1992. Self-organizing Fuzzy Controlles Based on Temporal Back Propogation. IEEE. Trans. Neural Networks, 3: 714-723.

Karr, C.L. and E.J. Gentry, 1993. Fuzzy control of pH using Genetic Algorithm. IEEE. Trans. Fuzz Sys., 1: 46-53.

Lee, C.C., 1990a. Fuzzy logic in control systems: Fuzzy logic controller-Part I, IEEE. Trans. Sys., Man, Cybern., 20: 404-418.

Lee, C.C., 1990b. Fuzzy logic in control systems: Fuzzy logic controller-Part II, IEEE. Trans. Sys., Man, Cybern., 20: 419-435.

Procyk, T.J. and E.H. Mamdani, 1979. A linguistic self-organizing process controller. Automatica, 15: 15-30.

Rajani K. Mudi and Nikhil R. Pal, 1999. A Robust Self-Tuning Scheme for PI- and PD-Type Fuzzy Controllers, IEEE. Trans. Fuzzy Sys., 7: 2-16.

Shimojima, K., T. Fukuda and Y. Hasegawa, 1995. Self-tuning fuzzy modeling with adaptive membership function, rules and hierarchical structure based on genetic algorithm. Fuzzy Sets Sys., 7: 295-309.

Teo Lian Seng and Marzuki Bin Khalid, 1999. Tuning of a Neuro-Fuzzy Controller by Genetic Algorithm, IEEE. Trans. Sys. Man and Cybernetics-Part B: Cybernetics, Vol. 29.

Wei Wang, Han-Xiong Li and Jingtao Zhang, 2002. Intelligence-Basedhybrid control for power plant boiler. IEEE. Trans. Control. Sys. Tech., 10: 280-286.