ANALYSIS OF SELF TUNING FUZZY PID INTERNAL MODEL CONTROL

Manoj Pandey*

Vinith Chauhan*

Pradeep Kumar Nathaney*

Shashank Kumar*

ABSTRACT

In this paper internal model control and fuzzy self-tuning PID controller is combined into a whole controller which make up a new controller fuzzy self-tuning PID internal model controller. First the internal model control system can be changed into conventional PID unity feedback control system through introducing pade series to approximate the time delay unit. Then using fuzzy inference to tune the PID parameters online, the fuzzy self tuning PID internal model controller is realized. This controller combines the advantage of fuzzy control, internal model control and PID control.

Keywords: Fuzzy logic controller, Controller tuning, PID, IMC

* Assistant Professor (ECE), St. Margaret Engineering College, Neemrana

1. INTRODUCTION

Internal model control is a useful method to overcome the time delay of real system. But when the mathematical model of real plant is changed, that is the internal model control model is not matched with the real plant, the response performance will grow poor and even cause the instability of the system. So the conventional internal model control cannot overcome the problem of parameter-time-varying of time delay system. In order to resolve this problem, internal model control and fuzzy self-tuning PID controller is combined into a whole controller which make up a new controller fuzzy self tuning PID internal model controller. Simulation results suggest system adopted fuzzy self-tuning PID internal model controller has better steady quality, good dynamic performance and comparatively strong robustness.

The principle of fuzzy self-tuning PID is firstly to find out the fuzzy relationship between three parameters of PID and error(e) and error changes(ec). Fuzzy inference engines modify three parameters to be content with the demands of the control system online through constantly checking e(e=r-y) and ec(ec=de/dt). Thus, the real plant will have better dynamic and steady performance. The structure of fuzzy self tuning PID is just as fig. 1.

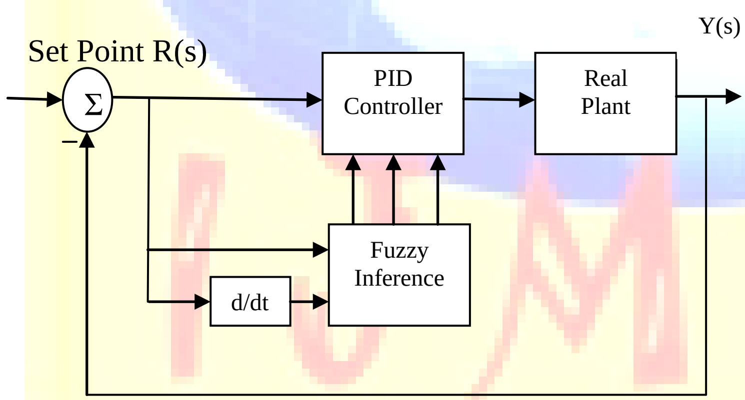

Fig. 1 Fuzzy self-tuning PID structure

The unity feedback controller can be realized by a PID controller with filter, then the internal model control can be found approximately through parameter-tuning of PID controller.

2. FUZZY LOGIC BASED SELF TUNING PID INTERNAL MODEL CONTROL

In this section the modelling of the fuzzy logic based self tuning PID internal model control is explained.

2.1 Controller design procedure

The fuzzy logic based self tuning PID IMC design consists of the following steps.

1) Identification of input and output variables.

2) Construction of control rules.

3) Establishing the approach for describing system state in terms of fuzzy sets, i.e. establishing fuzzification method and fuzzy membership functions.

4) Selection of the compositional rule of inference.

5) Defuzzification method, i.e., transformation of the fuzzy control statement into specific control actions.

The above steps are explained with reference to fuzzy logic based PID IMC in the following sections.

2.2 Fuzzy Logic based self tuning PID IMC

The principle of fuzzy self-tuning PID is firstly to find out the fuzzy relationship between three parameters of PID and error(e) and error changes(ec). Fuzzy inference engines modify three parameters to be content with the demands of the control system online through constantly checking e and ec . Thus, the real plant will have better dynamic and steady performance. The structure of fuzzy self tuning is already discussed in fig. 1.

2.2.1 Selection of input and output variables

Define input and control variables, that is, determine which states of the process should be observed and which control actions arc to be considered. Fuzzy self-tuning PID controller is adopted two input variables and three output variables. The inputs variables are e and ec, the output variables are kp, ki, kd. The dynamic performance of the system could be evaluated by examining the response curve of these variables. The values of kp, ki and kd is taken as the output from the fuzzy logic controller and then further these values are utilized to next module of control system.

2.2.2 Selection of Membership Function

The number of linguistic variables describing the fuzzy subsets of a variable varies according to the application. Usually an odd number is used. A reasonable number is seven. However, increasing the number of fuzzy subsets results in a corresponding increase in the number of rules. Each linguistic variable has its fuzzy membership function. The membership function maps the crisp values into fuzzy variables. The triangular membership functions are used to define the degree of membership. It is important to note that the degree of membership plays an important role in designing a fuzzy controller.

Each of the input and output fuzzy variables is assigned seven linguistic fuzzy subsets varying from negative big (NB) to positive big (PB).

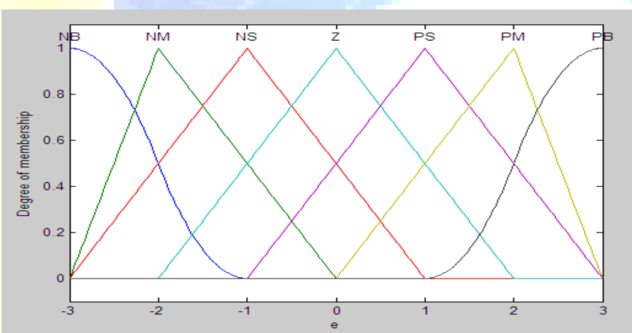

Fig. 2 (a) Membership functions for error 'e'

Each subset is associated with a triangular membership function to form a set of seven membership functions for each fuzzy variable.

The variables are given the range that may defer application to application. The membership functions of the input output variables have more than 50% overlap between adjacent fuzzy subsets. The membership functions for all inputs and outputs are shown in fig. 2.

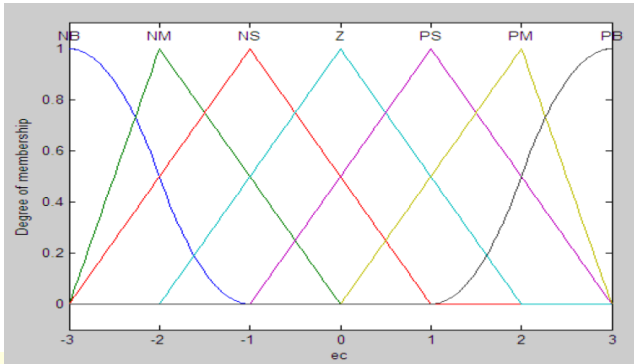

Fig. 2 (b) Membership functions for error derivative ‘ec’

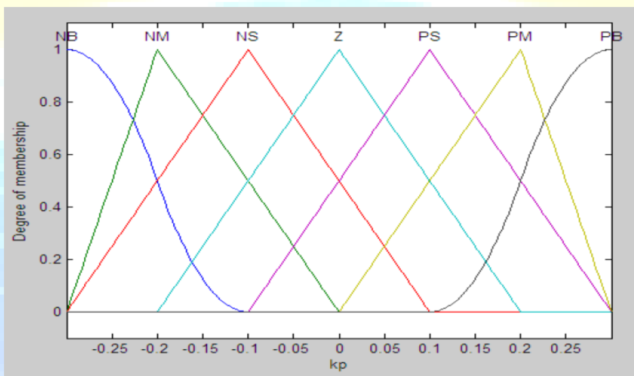

Fig. 2 (c) Membership functions for ‘Kp’

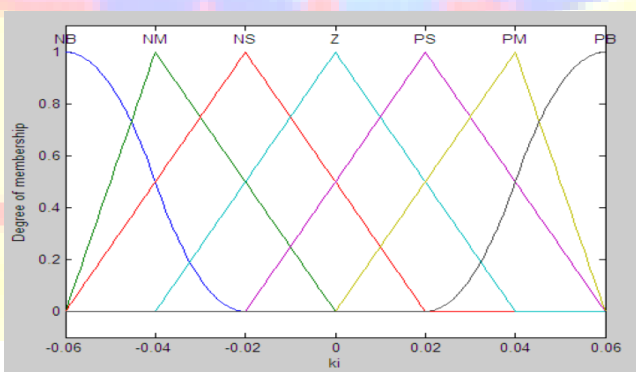

Fig. 2 (d) Membership functions for ‘Ki’

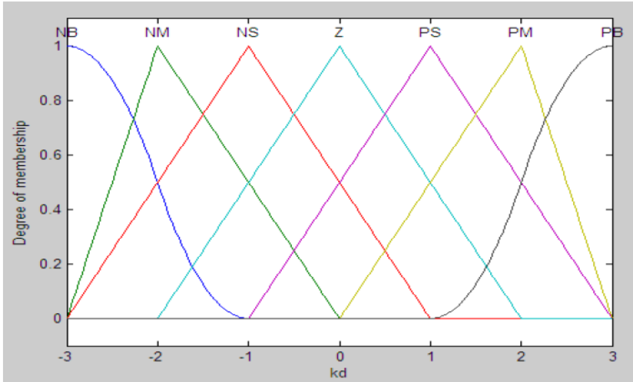

Fig. 2 (e) Membership functions for 'Kd'

2.2.3 Making Fuzzy rule base

A set of rules which define the relation between the input and output of fuzzy controller can be found using the available knowledge in the area of designing the system. The self tuning rule is deferent according to different e, ec, kp, ki and kd. These rules are defined using the linguistic variables. The two inputs, error and rate of change in error, result in 49 rules. The typical rules are having the following structure:

Rule 1: If (e is NB) and (ec is NB) then (kp is PB)(ki is NB)(kd is PS)

Rule 2: If (e is NB) and (ec is NM) then (kp is PB)(ki is NB)(kd is NS)

Rule 3: If (e is NB) and (ec is NS) then (kp is PM)(ki is NM)(kd is NB)

Rule 3: If (e is NB) and (ec is Z) then (kp is PM)(ki is NM)(kd is NB)

And so on….

If mod value (i.e. non negative) of 'e' is bigger, in order to make the systems having better tracking performance, kp value should be bigger, kd value should be smaller and ki value is prefer to be zero so as to avoid bigger overshoot. That is Integral influence should be limited.

If mod value 'e' is medium, in order to decrease the overshoot and keep the response speed rapid, kp value should be small, ki value should be medium and under this condition, kd value will have strong influence to the whole system. So the experience value is: if mod value 'ec' is comparative big, then kd value can be chosen small; if mod value 'ec' is comparative small, kd value can be chosen bigger.

However If mod value ‘e’ is small, in order to keep the systems have better steady performance, improve the ability of anti-disturbance and avoid oscillation, kp value and ki value should be bigger. At the time, in order to avoid oscillation occurring near the set value, kd value is critical. It can be chosen according to mod value ‘ec’ . If mod value ‘ec’ is bigger, then kd value can be smaller; if mod value ‘ec’ is smaller, kd value can be chosen larger.

Each output is obtained by applying a particular rule expressed in the form of membership functions. Finally the output membership function of the rule is calculated. This procedure is carried out for all of the rules and with every rule an output is obtained.

Using min-max inference, the activation of the i^{th} rule consequent is a scalar value which equals the minimum of the two antecedent conjuncts’ values. For example if error (e) belongs to NS with a membership of 0.025 and derivative of error (ec) belongs to NB with a membership of -1.72 then the rule consequence i.e. kp will be 0.192, Ki will be – 0.0384 and kd will be -1.02 .

The knowledge required to generate the fuzzy rules can be derived from an offline simulation. Some knowledge can be based on the understanding of the behaviour of the dynamic system under control. However, it has been noticed in practice that, for monotonic systems, a symmetrical rule table is very appropriate, although sometimes it may need slight adjustment based on the behaviour of the specific system. If the system dynamics are not known or are highly nonlinear, trial-and-error procedures and experience play an important role in defining the rules.

2.2.4 Defuzzification

The input for the defuzzification process is a fuzzy set (the aggregate output fuzzy set) and the output is a single crisp number. As much as fuzziness helps the rule evaluation during the intermediate steps, the final desired output for each variable is generally a single number. However, the aggregate of a fuzzy set encompasses a range of output values, and so must be defuzzified in order to resolve a single output value from the set. The most popular defuzzification method is the centroid calculation, which returns the center of area under the curve and therefore is considered for defuzzification.

For a discretised output universe of discourse

Y = (y1 , ... yp)

Which gives the discrete fuzzy centroid, the output of the controller is given by following expression:

$$u_k = \frac{\sum_{i=1}^{p} Y_i W_i}{\sum_{i=1}^{p} W_i}$$

3. SIMULATION RESULT

3.1 Performance with Fuzzy Logic based self tuning PID IMC

In this section, we have taken a first-order rational transfer

function model with delay-time. First, we consider a main model with some time delay as follows

$$G_p(s) = \frac{1}{S+1} e^{-0.1s}$$

Then using the 1/1 pade series to approximate the time delay unit we can get the approximate model of the system. That is

$$G_m(s) = \frac{1}{S+1} \left(\frac{1 - 0.05s}{1 + 0.05s} \right)$$

Using this model, unit step response is simulated with Matlab.

This is implemented using following FIS (fuzzy Inference System) properties:

And Method: Min

Or Method: Max

Implication: Min

Aggregation: Max

Defuzzification: Centroid

3.2 Self tuning of parameters

Tuning a system means adjusting three multipliers Kp, Ki and Kd adding in various amounts of these functions to get the system to behave the way we want.

At the first stage the Fuzzy logic controller determines the value of Kp from error and the rate of change of error, the second stage determines Kd from the error and finally Ki from the rate of change of error. The process is repeated until the error is zero. The Kp, Ki and Kd are set by the fuzzy logic controller to improve the performance of rise time, peak overshoot, oscillation and the settling time.

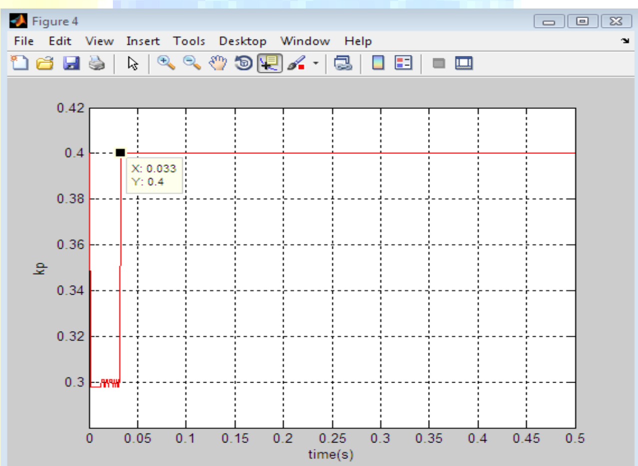

Fig. 3 (a) Tuning of Kp

Fig. 3 (a) shows the tuning of Kp. Kp is typically the main drive in a control loop, it reduces a large part of the overall error. For this system it becomes stable at 0.4 after 0.033 seconds.

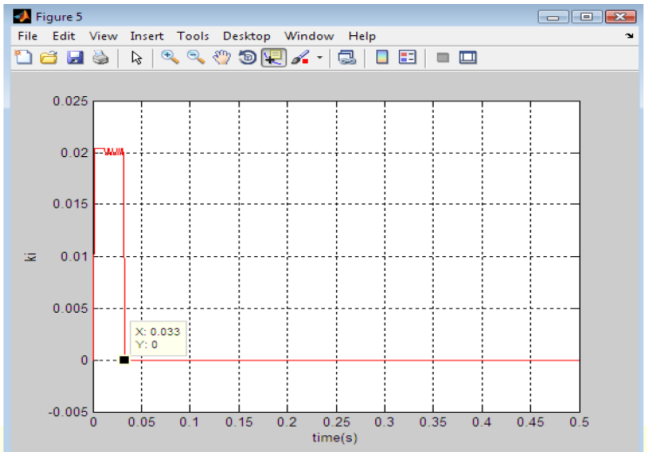

Fig. 3 (b) Tuning of Ki

Fig. 3(b) shows the tuning of Ki. Ki Reduces the final error in a system. Summing even a small error over time produces a drive signal large enough to move the system toward a smaller error. For this system it becomes stable at zero after 0.033seconds.

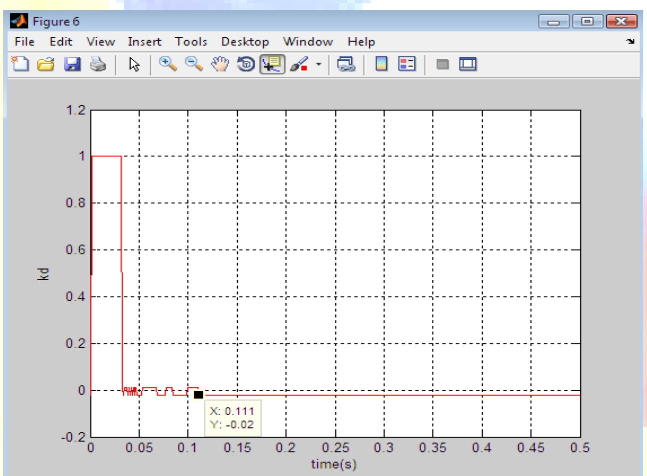

Fig. 3 (c) Tuning of Kd

Fig. 3(c) shows the tuning of Kd. The function of the parameter Kd is to reduce overshoot and ringing. It has no effect on final error. For this system Kd reduces to -0.02 and becomes stable at this value after 0.111 seconds.

3.3 Performance curve of Fuzzy Logic based self tuning PID IMC

Fig. 4 shows the step response of the designed Fuzzy Logic

based self tuning PID IMC. Different performance parameter of this control system is given below-

Rise time (tr) = 0.048 seconds

Peak time (tp) = 0.059 seconds

Maximum overshoot (Mp) = 0.078 (1.078-1= 0.078)

Settling time (ts) = 0.115 seconds

Here the settling time is the time needed to settle down the oscillation within 2% of the desired value of the output.

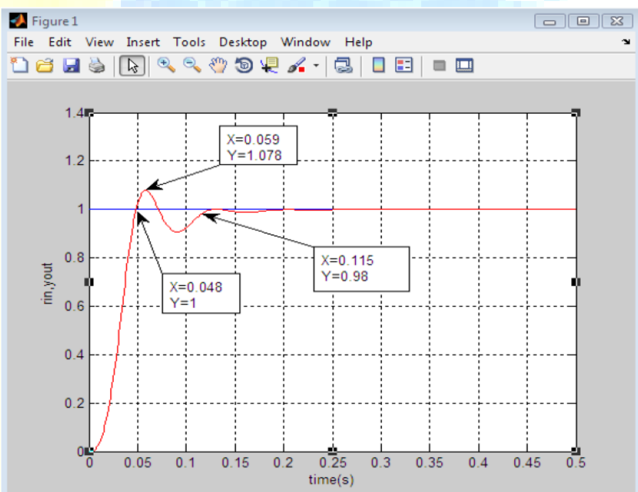

Fig. 4 Performance curve of Fuzzy logic based PID IMC

Fig. 5 shows how the error in transient response of system varies with time after 0.115 seconds error minimizes to 0.02. However as time increase, error further reduces and becomes nearly zero.

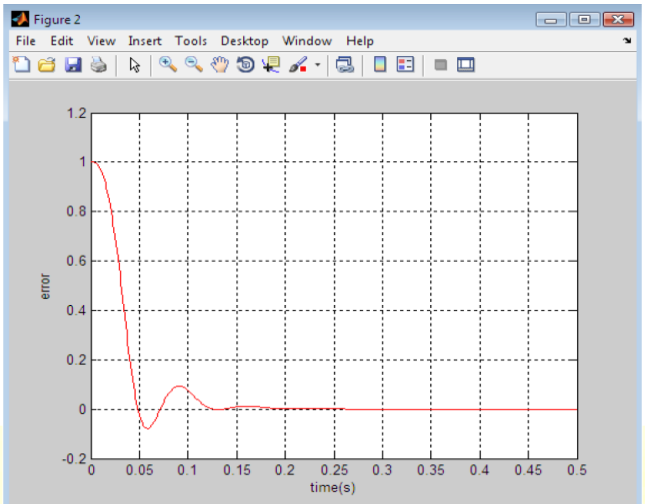

Fig. 5 Error Curve

3.4 Comparison of results

To compare the performance of fuzzy self-tuning PID internal model controller and fuzzy logic control, the step response is shown in fig. 6.

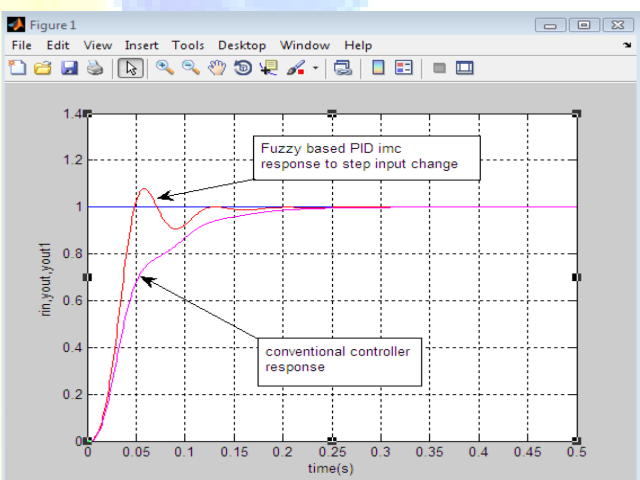

Fig. 6 Comparison of performance of fuzzy self-tuning PID internal model

controller with fuzzy logic control scheme to unit step change in input.

Although the adjusting time of fuzzy self-tuning PID internal model controller is prolonged too, the performance of fuzzy self-tuning PID internal model controller is better than when using only fuzzy logic control. When internal model control model is accurate, unity step input signal is added to the input variable r. The unit step response curve is shown in fig. 6. One curve shows fuzzy self-tuning PID internal model controller response. Other curve shows fuzzy logic control response. From fig. 6, we can say that if model of IMC matches with the real plant, fuzzy self-tuning PID internal model controller has faster rise time and shorter adjusting time. But on the whole, internal model control and fuzzy self-tuning PID internal model controller both have better response performance.

Therefore, it can inferred that the Fuzzy Logic based PID Internal Modal controller can be designed without requiring any complex mathematical Expression to get much improved response.

4. CONCLUSIONS

From simulation results, we can conclude that although the adjusting time of fuzzy selftuning PID internal model controller is prolonged too, the performance of fuzzy selftuning PID internal model controller is better than Fuzzy Logic control. Also comparing with Fuzzy Logic control alone , fuzzy self tuning PID internal model control has the advantages of system being faster rise time, good steady quality with shorter adjusting time and smaller steady error and when the plant's parameters change, system adopted fuzzy self-tuning PID internal model control has strong robustness which can regulate system into steady state fast.

REFERENCES

[1] L.A. Zadeh; "Fuzzy Sets", Information and Control, vol. 8, pp. 338-353, 1965

[2] E.H. Mamdani; "Application of fuzzy algorithms for control of simple dynamic plant", Proc. IEEE, vol 121, pp. 1585-1588, 1974

[3] Y.F.Li and C C.Lau; " Development of fuzzy algorithms for servo systems", Control System Mag,Vol.9,No.3, pp. 65-72, 1989

[4] C.C. LEE; "Fuzzy logic in control systems: fuzzy logic controller – Part I& II", IEEE Trans. Syst. Man Cybern., No. 20, pp. 404–435, 1990

[5] Yu Yiming and E.R. Fowler; "Rule based fuzzy logic inferencing, Systems, Man, and Cybernetics", 'Humans, Information and Technology', 1994 IEEE International Conference, Vol. 1, pp. 465 - 470, 1994

[6] Han-Xiong Li and H. B. Gatland; "Conventional fuzzy control and its enhancement", IEEE Trans. Syst., Man, Cybern., vol. 26, pp. 791–796, 1996

[7] P.J. Gawthrop ; "Self-tuning PID control structures", Getting the Best Our of PID in Machine Control (Digest No.: 1996/287), IEE Colloquium, pp. 1 – 4,1996

[8] T.J. Ross; "Fuzzy Logic with Engineering Applications", McGraw-Hill, ISBN: 0070539170 37, 1997

[9] Aniruddha Datta and Lei Xing ; "The theory and design of adaptive internal model control schemes", Proceeding of American Control Conference, vol.6, pp.3677-3684, 1998

[10] K Passino and S. Yurkovich; "Fuzzy Control", Reading, MA: Addison- Wesley, 1998

[11] W. F. Xie and A. B. Rad; "Fuzzy adaptive internal model control", IEEE Trans. Industrial Electronics, vol. 47, no. 1, pp. 193-202, 2000

[12] C. W. Tao and J.-S. Taur; "Flexible Complexity Reduced PID-like Fuzzy Controllers" , IEEE Transactions on Systems, Man, and Cybernetics, Part B, Vol. 30, No. 4, pp. 510-516, 2000

[13] Zhiqiang Gao, T.A. Trautzsch and J.G Dawson; "A stable self-tuning fuzzy logic control system for industrial temperature regulation", IEEE Transactions on Industry Applications, vol. 38, Issue 2, pp. 414- 424, 2002

[14]Keiji Watanabe and Eiichi Muramatsu; "Adaptive internal model control of SISO systems (Conference Paper style)" , SICE Annual Conference in Fukui, pp. 651-656, 2003

[15] B. W. Bequette; "Process Control: Modeling, Design And Simulation", Upper Saddle River, New Jersey: Prentice Hall, ISBN: 0-13-353640-8, 2003

[16] Y. Baba, T. Shigemasa , M. Yukitomo, F. Kojima, M. Takahashi and E. Sasamura; "Model-driven PID control system in single-loop controller", in Proceedings of the SICE annual conference Fukuri, Vol. 1, pp. 187-190, 2003

[17] Xiu-Zhang Jin, Jian-Ping Sun, Ji-Zhen Liu and Li-Feng Zhang; "Analysis and design of adaptive internal model control for unstable process with dead time", Proceedings of 2004 International Conference on Machine Learning and Cybernetics, vol.2, pp. 983 - 987, 2004

[18] Guihua Han, Lihua Chen, Junpeng Shao and Zhibin Sun.; "Study of Fuzzy PID Controller for Industrial Steam Turbine Governing System", proceedings of ISCIT, pp. 1228-1232, 2005

[19] Guillermo J. Silva, Aniruddha Datta. and S. P. Bhattacharyya; "PID Controllers for Time-Delay Systems", Boston, ISBN 0-8176-4266-8, 2005

[20] X.W. Gao, Y.P. Zhao, W.J. Gao and X.F. Yu; "Simulation and research of fuzzy immune adaptive PID control in coke oven temperature control system", The Sixth World Congress on Intelligent Control and Automation, pp.3315-3319, 2006

[21] L.A. Zadeh ; "Is There a Need for Fuzzy Logic?", in proceedings of Annual Meeting of the North American Fuzzy Information Processing Society, pp. 1-3, 2008

[22] Juan Chen, Lu Wang and Bin Du; "Modified internal model control for chemical unstable processes with time-delay", in proceedings of 7th World Congress on Intelligent Control and Automation, pp. 6353 – 6358, 2008

[23] Jun Liu, Wan-li Wang and Xiu-hua Dou; "Multiple Model Internal Model Control Based on Fuzzy Membership Function", in proceedings of IEEE International Conference on Automation and Logistics, pp. 881-885, 2008

[24] P.M. Mary and N.S. Marimuthu; "Design of self-tuning fuzzy logic controller for the control of an unknown industrial process", IET control theory & applications, vol. 3, pp. 428-436, 2009