International Journal of Current Engineering and Technology
E-ISSN 2277 – 4106, P-ISSN 2347 - 5161

Available at http://inpressco.com/category/ijcet

Research Article

Design of Model Reference Self Tuning Mechanism for PID like Fuzzy Controller

N. J. Patil[Ȧ*]

[Ȧ]Instrumentation Engineering, D. N. Patel College of Engineering, Shahada, Dist Nandurbar, India

Accepted 04 March 2014, Available online 01 April 2014, **Vol.4, No.2 (April 2014)**

Abstract

This paper presents a method for tuning the scaling factors of PID type fuzzy controllers. The method tunes the input scaling factor related to change in error and the output scaling factor related to integral output of PID type of fuzzy controller using model error, error between reference model and process. Fuzzy Controller used is typical fuzzy PID type of controller in which the scaling factors are set at their initial values using classical approach. An adaptive mechanism block based on model error is added in the classical fuzzy controller. This block tunes one input scaling factor and one output scaling factor based on function of model error. The algorithm is tested on the second order plus delay time processes. The results are compared with other methods such as PID, classical FLC and tuning method based on relative rate observer. The impact of the present method is checked using both graphical and analytical criteria.

Keywords: *Fuzzy, PID, Self Tuning, Time Delay, Reference Model.*

1. Introduction

Fuzzy Logic controllers are mostly used for process control applications where the system is complex or shows a nonlinear behavior. After the first application of fuzzy control algorithm, fuzzy logic is applied quite often in most of the control problems. Although fuzzy controller application was successful compared to classical controllers, the design procedure is dependent on the experience and knowledge of the operator and it has limitation of using heuristic rule base (E. Mamdani, 1974).

In literature, various types of fuzzy PID (including fuzzy PI and fuzzy PD) controllers have proposed. The PID type controllers have structures analogous to the conventional PID controller (Z. Woo *et al*, 2000). Fuzzy PI type control is mostly used because of the limitation of fuzzy PD type controller such as steady state error problem. Fuzzy PI controller also has disadvantage of showing poor performance in case of higher order systems due to internal integration operation (J. Xu *et al*, 2000). For Fuzzy PID controller three dimensional rule table is needed which makes the design more complex. So combination of fuzzy PD and fuzzy PI is used to improve the performance (M. Golob, 2001; M. Guzelkaya, 2001; R. Mudi and N. Pal, 2000; O. Karasakal, 2005). The PID type FLC can be formed either by using a combination of one PI and one PD type FLC with two different rule base given in or by using one PD type FLC with an integrator and summation unit at the output presented in (S. Chopra *et al*, 2008; H. Li and H. Gatland, 1996).

The design parameters of fuzzy controller are classified as (M. Guzelkaya *et al*, 2003):

(1) Structural parameters,
(2) Tuning Parameters

First type of parameters covers input /output variables to fuzzy inference, fuzzy linguistic sets, membership functions, fuzzy rules, inference mechanism and defuzzification mechanism. Second category includes input/ output scaling factors, and parameters of membership functions such as centre of membership functions etc. Generally first types of parameters are selected during offline design while the others can be calculated during online adjustments of the controller to improve the performance of process.

A non-adaptive fuzzy controller is one in which these parameters do not change once the controller is being used on-line. If any one of these parameters is altered on-line, controller is called as adaptive fuzzy controller. Adaptive Fuzzy controller that modifies the rules is called self organizing controllers and fuzzy controllers in which scaling factors are modified are called self tuning controllers (D. Drainkov *et al*, 1993). A class of self organizing controller is discussed in literature of Xu called as fuzzy learning controllers (L. Zhen and L. Xu, 2000).

In this paper, we will use fuzzy PID type controller formed using one PD type FLC with an integrator at the output. In this type of FLC number of scaling factors required is decreased as compared to the structure of combination of PI and PD type FLC having two different rules base. There are different tuning methods are presented in previous literature (C. Tseng, 2006; P. Mary and N. Marimuthu, 2009, D. Seborg *et al*, 2005). Here a modified method is presented for tuning the scaling factors of PID type FLC. This new method adjusts the input scaling factor related to change in error and output scaling

*Corresponding author: **N. J. Patil**

factor related to integral coefficient in similar manner to the methods given in (P. Mary and N. Marimuthu, 2009, D. Seborg *et al*, 2005).

Rest of the paper is organized in following sections. In section 2 presents tuning mechanism based on model error for scaling factors. In section 3 comparative results using tuning mechanism with conventional PID, classical PID, self tuning fuzzy from previous literature are given. Conclusions on the proposed method are given in section 4.

2. Model Reference Self Tuning Fuzzy PID Controller

The fuzzy PID controller used for this work is shown in Fig. 1. The output of fuzzy PI and PD controller is given by (Z. Woo *et al*, 2000)

$$U_{FPI}(k) = U(k-1) + \Delta U(k)$$
$$U_{FPD}(k) = GCU\ \Delta u(k) \quad (1)$$

Where $U_{FPI}(k)$ and $U_{FPD}(k)$ are output fuzzy PI and fuzzy PD controller, GCU is scaling factor for fuzzy PD controller, $\Delta u(k)$ is output of fuzzy controller.

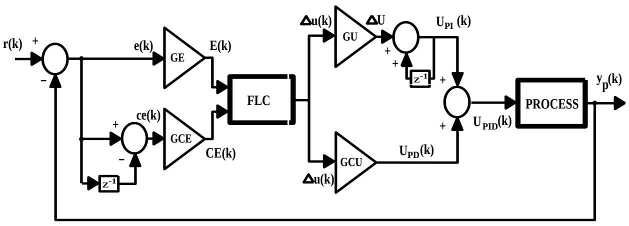


Fig.1 PID like Fuzzy Controller (Z. Woo *et al*, 2000)

Inputs of fuzzy controller are scaled using scaling factors also known as fuzzy gains as:

$$E(k)=GE\ e(k)\ ;$$
$$CE(k)=GCE\ ce(k); \quad (2)$$
$$\Delta U(k)=GU\ \Delta u(k)$$

Where $e(k)$, $ce(k)$, GE, GCE are error, change in error and scaling factor related to error and change in error respectively. The output of fuzzy PID controller is given as

$$\begin{aligned} U_{FPID}(k) &= U_{FPI}(k) + U_{FPD}(k) \\ &= U(k-1) + \Delta U(k) + GCU\ \Delta u(k) \\ &= U(k-1) + GU\ \Delta u(k) + GCU\ \Delta u(k) \\ &= f(GE, GCE, GU, GCU) \end{aligned} \quad (3)$$

The tuning of scaling factors related to integral coefficient, *GU* and derivative (change in error) coefficient, *GCE* has more effect on the performance of the system.

Here we introduce a reference model in structure of fuzzy controller to generate model error given by

$$e_m(k) = y_m(k) - y_p(k) \quad (4)$$

$y_m(k)$ represents the desired performance and $e_m(k)$ represents the difference between actual process value and expected value of output. By considering the fact we will use the following method and define the factors as functions of error as follows:

$$f(k) = \alpha_1 \times \left(e_m(k)\right) + \alpha_2$$
$$g(k) = \beta_1 \times \left(1 - e_m(k)\right) + \beta_2 \quad (5)$$

where α_i, β_i are constant values. These constants are needed due to initial and intermediate conditions. When model error is minimum or zero i.e. $e_m(k) = 0$ means $y_p(k) = y_m(k)$ then,

$$f(0) = \alpha_2$$
$$g(0) = \beta_1 + \beta_2 \quad (6)$$

For positive model error, $e_m(k) > 0$ means $y_p(k) < y_m(k)$ then the above equation becomes,

$$f(+) = \alpha_1 + \alpha_2$$
$$g(+) = \beta_2 - \beta_1 \quad (7)$$

For positive model error, $e_m(k) < 0$ means $y_p(k) > y_m(k)$ then the above equation becomes,

$$f(-) = \alpha_2 - \alpha_1$$
$$g(-) = \beta_1 + \beta_2 \quad (8)$$

The basic idea of tuning the scaling factor is to multiply by amount $f(k)$ and $g(k)$ to the initial tuning values by small amount to increase or decrease them according value of process output. This will generate the relationship for adaptive mechanism as follows

$$GU(k) = GU \times f(k)$$
$$GCE(k) = GCE \times g(k) \quad (9)$$

From above equation, it is observed that we are making the two scaling factors function of model error. This will force the overall system to behave like a reference model.

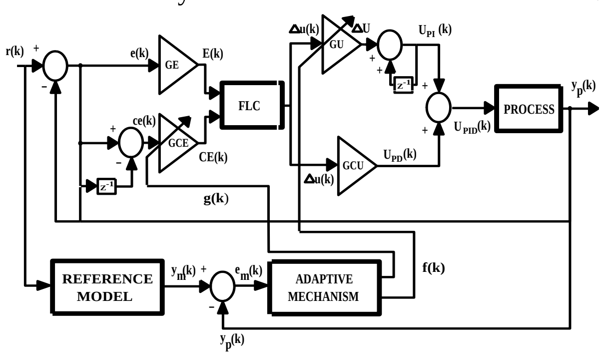


Fig.2 Model reference self tuning adaptive fuzzy PID controller

The function related to integral factor i.e. $f(k)$ decreases as the model error decreases while the function related to derivative factor increases i.e. *g(k)* which is directly proportional to *GU* and *GCE* thus decreasing *GU* and increasing *GCE* and vice versa. The overall block diagram using above tuning scheme is depicted in Fig. 2.

It is well known that in PID type controller when the integration factor is weak the system response will probably slow. But when the integration factor is too strong then it will make the system unstable. From the above discussion it is said that the response of system can be fastened to have better rise time which can be achieved by having large integration and small derivative at initial stages and to avoid instability and oscillations the factors are reversed in later stages which can be achieved by using relation (9). Values of α_1, β_1 < 1 as these are scaling factors for error and α_2, β_2 > 1 as they are constants used for avoiding the situations at initial and positive and negative model error values. For this study these factors are α_1= 0.2, β_1 = 0.8, α_2=1.7, and β_2 =4.5.

The equivalent PID and fuzzy control components according to (M. Guzelkaya *et al*, 2003) are represented by

$$GE = 1;\quad GCE = \frac{K_D}{K_P};\quad GU = K_P;\quad GCU = \frac{K_I}{K_P} \tag{10}$$

The membership functions for error, change in error, and controller output are shown in Figure 3 and rule base used is given in Table 1.

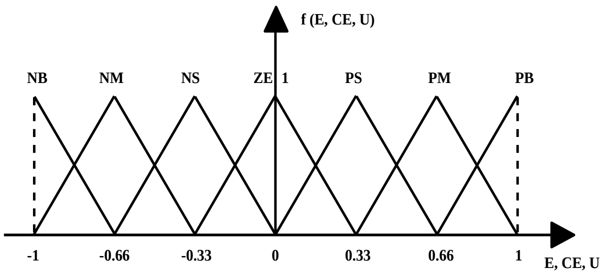


Fig.3 Input Output Membership Functions (D. Drainkov *et al*, 1993)

Table 1 Fuzzy Rule Table (D. Drainkov *et al*, 1993)

E/CE	**NB**	**NM**	**NS**	**ZE**	**PS**	**PM**	**PB**
NB	NB	NB	NB	NB	NM	NS	ZE
NM	NB	NB	NB	NM	NS	ZE	PS
NS	NB	NB	NM	NS	ZE	PS	PM
ZE	NB	NM	NS	ZE	PS	PM	PB
PS	NM	NS	ZE	PS	PM	PB	PB
PM	NS	ZE	PS	PM	PB	PB	PB
PB	ZE	PS	PM	PB	PB	PB	PB

Self tuning procedure

1. Compute model error $e_m(k)$ using eq. (4) i.e. $e_m(k) = \left[y_m(k) - y_p(k) \right]$.
2. Generate model error based adaptation functions $f(\cdot)$ *and* $g(\cdot)$.
3. There are three possibilities.

(a) If process output is same as model output $y_m(k) = y_p(k)$ i.e. if model following is perfect then

$$f(\cdot) = \alpha_2$$
$$g(\cdot) = \beta_1 + \beta_2$$

Go to step 4

(b) If process output is lagging model output $y_m(k) > y_p(k)$ then

$$f(\cdot) = \alpha_1 + \alpha_2$$
$$g(\cdot) = \beta_2 - \beta_1$$

Go to step 4

(c) If process output is leading model output $y_m(k) < y_p(k)$ then

$$f(\cdot) = \alpha_2 - \alpha_1$$
$$g(\cdot) = \beta_1 + \beta_2$$

Go to next step

4. Update the scaling factors GCE and GU using

$$GCE(\cdot) = GCE \times g(\cdot)$$
$$GU(\cdot) = GU \times f(\cdot)$$

5. While in this selection of α_i, β_i is depends on the initial values of scaling factors (*GE*, *GCE*, *GU* and *GCU*) computed using the relations given in eq. (10) from PID parameters.

3. Simulation Results

This section provides the results of the proposed self tuning method compared with PID, classical fuzzy, self tuning scheme based on relative rate observer fuzzy logic controller (RROFLC, (M. Guzelkaya *et al*, 2003)) and proposed scheme Model Reference Self Tuning Adaptive Fuzzy Controller (MRSTAFC) is given. IAE is used as performance index. Simulations are done on a second order plus dead time (SOPDT) process models with different combinations of parameters given in (D. Seborg *et al*, 2005).

3.1 Reference Model

The reference model is used to indicate the required performance which follows desired specifications such as time constants, gain etc. In this work we used the second order model with transfer function of the form

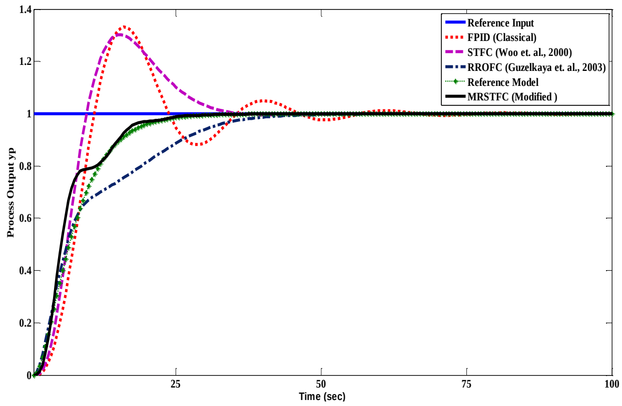
$$G_1(s) = \frac{2}{(3s^2 + 2s + 1)} \tag{11}$$

Example 1

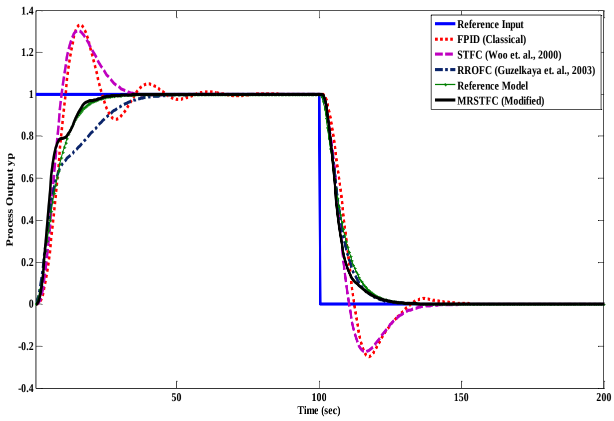
Let us consider second order plus delay time system from Woo's literature (Z. Woo *et al*, 2000)

$$G(s)=\frac{1}{(s+1)(2s+1)}e^{-0.25s}$$

The values of *GE*, *GCE*, *GU* and *GCU* are 1, 0.25, 1 and 0.2 respectively. The response with various controllers is illustrated in fig. 4. Analytical comparison for various controllers is given in Table 2. It is clear that the system response is better with model reference self tuning fuzzy controller (MRSTFC) as compared to the other controllers. The sudden set point change applied after the system is settled at 50 sec to check the robustness. In this case also the response given by MRSTFC is better as compared to the other controllers. Also it is observed that it is tracking the reference model.



(a)



(b)

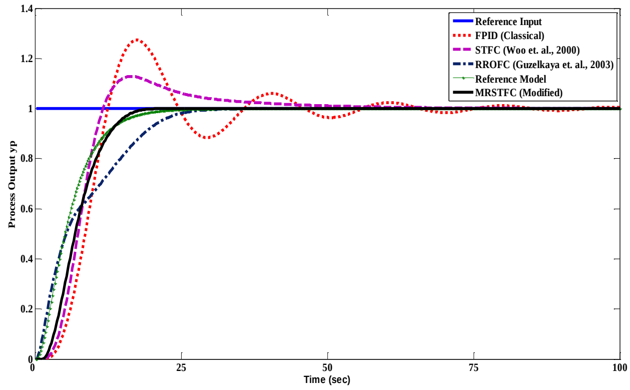
Fig.4 Response for Example (1)

Example 2

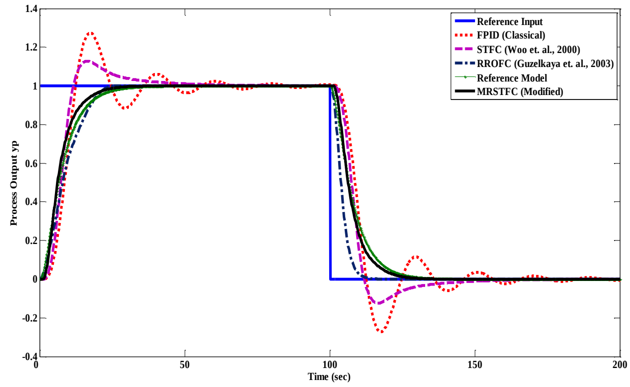
Consider the second order plus dead time process with repeated roots

$$G_2(s)=\frac{2}{(2s+1)^2}e^{-s}$$

The initial fuzzy settings are obtained as *GE*, *GCE*, *GU* and *GCU* are 1, 0.45, 2.43 and 0.154 respectively. The performance with various controllers is illustrated in fig. 5. It is observed that the performance of the self tuning adaptive fuzzy controller is better than the other methods.



(a)



(b)

Fig. 5 Response for Example (2)

Example 3

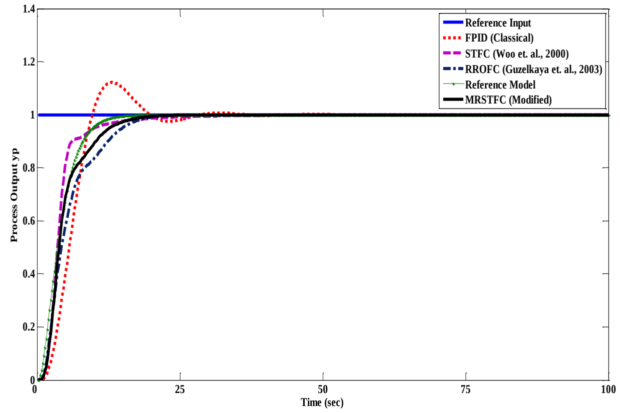
Consider the higher order process with transfer function

$$G_3(s)=\frac{27}{(s+1)(s+3)^3}$$

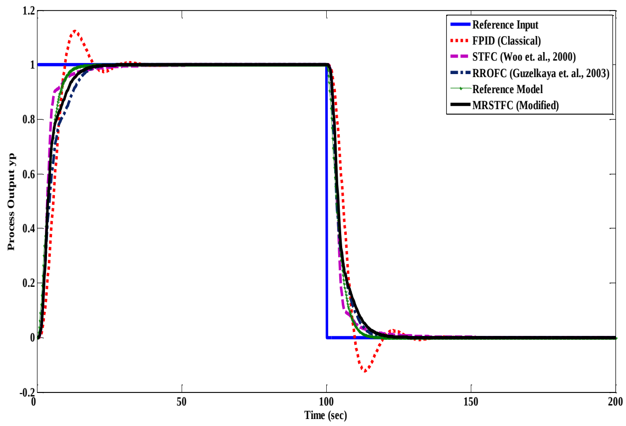
The initial fuzzy settings are obtained as *GE*, *GCE*, *GU* and *GCU* are 1, 0.42, 2.38 and 0.168. The performance with various controllers is illustrated in fig. 6. It is observed that the performance of the self tuning adaptive fuzzy controller is better than the other methods.

Table 2 Controller Performance Comparison

Process	**Controller**	**Performance Index IAE**	
		Given Set point	**Change in Set point**
1	FPID	2.168	4.194
	RROFLC	2.013	3.481
	STFC	1.863	3.631
	MRSTFC	1.314	2.683
2	FPID	6.027	12.16
	RROFLC	3.954	7.574
	STFC	4.952	9.265
	MRSTFC	3.643	6.535
3	FPID	2.758	6.458
	RROFLC	2.757	5.121
	STFC	2.758	4.993
	MRSTFC	2.408	4.137



(a)



(b)

Fig. 6 Response for example (3)

4. Conclusions

In this paper a self tuning method based on model error to tune the coefficients of fuzzy PID controller is proposed. The tuning scheme is based on the function of model error. The scaling factors related to derivative coefficient and integral coefficients are tuned using adaptive mechanism to improve the performance of the system. The tuning performed is online i.e. when the process is running. Due to model error the adaptive mechanism tunes the scaling factors such that the overall system behaves like a reference model.

It is concluded that method for tuning of scaling factors based on model error function is found efficient than the classical self tuning fuzzy and other controllers. This method may be extended for tuning of other two factors but it will increase the complexity of controller rather than improving results. The other method is to use function of change in error instead of using function of error as it gives rate of change of error. This may improve the performance of the overall system.

References

Mamdani E. (1974), Application of fuzzy logic algorithms for control of simple dynamic plant, *proc. of Institute of Electrical Engineers*, 121(12), pp. 1585-1588

Woo Z., Chung H., and Lin J. (2000), A PID type fuzzy controller with self tuning scaling factors, *Fuzzy sets and systems*, 115(2), pp. 321- 326.

Xu J., Hang C., and Liu C. (2000), Parallel structure and tuning of a fuzzy PID controller, *Automatica*, 36(5), pp. 673-684.

Golob M. (2001), Decomposed fuzzy proportional – integral – derivative controllers, *Applied soft computing*, 1(3), pp. 201-214.

Guzelkaya M., Eksin I., and Gurleyen F. (2001), A new methodology for designing a fuzzy logic controller and PI, PD blending mechanism, *Journal of intelligent and fuzzy systems*, 11(2), pp. 85-98.

Mudi R., and Pal N. (2000), A self tuning fuzzy PI controller, *Fuzzy sets and systems*, 115(2), pp. 327-338.

Karasakal O., Yesil E., Guzelkaya M., and Eksin I. (2005), Implementation of a New Self Tuning Fuzzy PID Controller on PLC, *Int. Journal of Electrical Engineering*, 13(2), pp. 277-286.

Chopra S., Mitra R., and Kumar V. (2008), Auto tuning of fuzzy PI type controller using fuzzy logic," *International Journal of Computational Cognition*, 6(1), pp. 12- 18.

Li H., and Gatland H. (1996), Conventional fuzzy control and its enhancement, *IEEE trans. on systems, man and cybern*. –part B, 26(5), pp. 791-797.

Guzelkaya M., Eksin I., and Yesil E. (2003), Self tuning of PID type fuzzy logic controller coefficients via relative rate observer, *Engineering applications of artificial intelligence*, 16(3), pp. 227-236.

Drainkov D., Hellendoorn H., and Reinfrank M. (1993), *An introduction to fuzzy control*, USA, Springer-Verlag.

Zhen L., and Xu L. (2000), Fuzzy learning enhanced speed control of an indirect field control induction machine drive, *IEEE Transactions on Control Systems Technology*, 8(2), pp. 270 - 278.

Tseng C. (2006), Model reference output feedback fuzzy tracking, *IEEE trans. on fuzzy systems*, 14(1), pp. 58-70.

Mary P., and Marimuthu N. (2009), Design of self-tuning fuzzy logic controller for the control of an unknown industrial process, *IET control theory and applications*, 3(4), pp. 428-436.

Seborg D., Edgar T., and Mellichamp D. (2005), *Process dynamics and control*, Wiley Blackwell, UK.