International Journal of Research
Available at https://edupediapublications.org/journals
p-ISSN: 2348-6848
e-ISSN: 2348-795X
Volume 03 Issue 13
September 2016

Classy Controller Using Fuzzy Logic - Controller for Concert Enhancement

***Khaled Mohammed Beer Gamal, **S. S Handa, ***M V Ramana Murthy**

*Research Scholor, Dept. Computer Science &Engg., Manav Rachna International University, Faridabad, India.

**Professor in Dept. of Computer Science &Engg., Manav Rachna International University, Faridabad, India.

*** Head Dept. of Mathematics & Computer Science, Osmania UniverSity, Hyderabad, India.

Abstract—
Besides the advantages, there are some limitations of Fuzzy Logic Controllers. The tuning of an FLC is a very difficult task. To make the tuning easy and efficient, there are several structures of fuzzy logic controllers are available. This paper considers new structures of fuzzy logic controllers. Two structures, self-tuning FLC and fuzzy supervised conventional controller are considered. In Fuzzy Logic control rules and scaling factors play very important role. In this paper, a self-tuning PI-like FLC is used for the tuning of output scaling factor. In FSPI, we tune the PI controller parameters with FLC. So, it adds the advantages of PID controller and FLC. The design of these controllers is discussed. These are implemented to control three non-linear example systems. The concert of FLC, STFLC and FSPI controller is compared with each other. It is found that STFLC and FSPI controller give better concert than the conventional PI controller or simple FLC.

Index Terms— Fuzzy logic controller (FLC), Fuzzy supervised controller, Membership Function , Self-tuning FLC , Scaling factor

INTRODUCTION

As the complexity of the controlled processes/systems is increasing researchers concentrated their efforts on providing simple and easy control algorithms. The design method for a controller should enable full flexibility in the modification of the control surface. The systems involved in practice are, in general, complex and time variant, with delays and nonlinearities, and often with poorly defined dynamics. The most control solutions developed earlier were based on precise mathematical models of the systems. But for practical systems, it is difficult to describe them by mathematical relations; hence, these model-based design approaches may not provide satisfactory solutions. FLC is not based on a mathematical model of the plant and is widely used to solve problems which are uncertain and vague and those with high nonlinearities. Due to their characteristics, FLCs have been implemented successfully in various applications such as process control and robotics [3, 4, & 5]. Fuzzy logic provides a certain level of artificial intelligence to the conventional PID controllers. Fuzzy PID controllers have self-tuning ability and on-line adaptation to nonlinear, time varying, and uncertain systems Fuzzy PID controllers provide a promising option for industrial applications with many desirable features [7].

Besides the advantages, there are some limitations of FLCs. The main limitation of FLC is the lack of existence of a systematic procedure for design and analysis of the control system. It is well-known that tuning of an FLC is a difficult task. To tune an FLC is a much more difficult job than tune a conventional controller because there are many more parameters to adjust in an FLC such as SFs, MFs and control rules. While in conventional PI, PD or PID controllers, there are only two or three parameters. There is a lot of research work have done on FLC tuning. However, still there is no standard and systematic methodology for the tuning of FLCs. Most of the tuning approaches for the FLC parameters (SFs, MFs and rule-base) are time consuming. In designing FLCs there are many other difficulties, such as lack of completeness of the rule-base and lack of definite criteria for the selection of the scaling factors, for the shape of MFs, the number of MFs, the total number of rules required, the inference mechanism, and also the defuzzification scheme. There are various types of adaptive FLCs have been developed to overcome the above mentioned problems [10].

Most of the practical processes under automatic control are non-linear higher order systems and may have a considerable dead time. Due to the problems associated with dead time and higher order nonlinearities, it is very difficult to design an effective controller. For having a satisfactory concert, the controller output should be a non-linear function of the process state (e and Δe). In such a situation to eliminate this drawback, fixed valued SFs and predefined MFs may not be sufficient. To control such a situation lots of research work on tuning of FLCs has been reported where either the input-output SFs or the definitions of fuzzy sets are tuned on-line to match the current plant characteristics [13].

Most of the controllers in operation today are PID controllers. Many PID controllers are poorly tuned in practice. A quite obvious way to automate the operator's task is to employ an artificial intelligence technique. Fuzzy control, occupying the boundary line between artificial intelligent and control engineering, can be considered as an obvious solution, which is confirmed by engineering practice [14]. If we tune the PID controller parameters on-line by an adaptive mechanism based on fuzzy logic, the control of nonlinear systems may become easy and effective. The combination of PID controller and Fuzzy logic is known as fuzzy supervised PID (FSPID).

FUZZY LOGIC

Fuzzy logic is a logic having many values, approximate reasoning and have a vague boundary. The variables in fuzzy logic system may have any value in between 0 and 1 and hence this type of logic system is able to address the values of the variables (called linguistic variables) those lie between completely truths and completely false. Each linguistic variable is described by a membership function which has a certain degree of membership at a particular instance.

The human knowledge is incorporated in fuzzy rules. The fuzzy inference system formulates suitable rules and based on these rules the decisions are made. This whole process of decision making is mainly the combination of concepts of fuzzy set theory, fuzzy IF-THEN rules and fuzzy reasoning. The fuzzy inference system makes use of the IF-THEN statements and with the help of connectors present (such as OR and AND), necessary decision rules are constructed.

The fuzzy rule base is the part responsible for storing all the rules of the system and hence it can also be called as the knowledge base of the fuzzy system. Fuzzy inference system is responsible for necessary decision making for producing a required output.

The fuzzy control systems are rule-based systems in which a set of fuzzy rules represent a control decision mechanism for adjusting the effects of certain system stimuli. The rule base reflects the human expert knowledge, expressed as linguistic variables, while the membership functions represent expert interpretation of those variables.

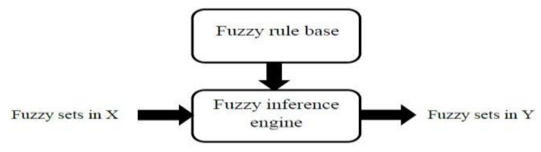

Fig. 1 A pure fuzzy system

The block diagram of a fuzzy control system is shown in Fig. 2. A fuzzy logic controller is composed of the following four elements:

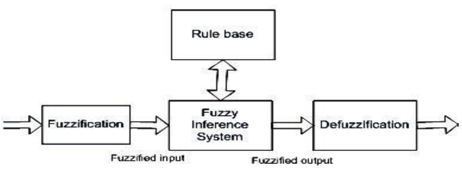

Fig. 2 Block diagram of fuzzy control system

1. A rule-base (a set of If-Then rules), which contains a fuzzy logic quantification of the expert's linguistic description of how to achieve good control.
2. An inference mechanism (also called an "inference engine" or "fuzzy inference" module), which emulates the expert's decision making in interpreting and applying knowledge about how best to control the plant.
3. A fuzzification interface, which converts controller inputs into information that the inference mechanism can easily use to activate and apply rules.
4. A defuzzification interface, which converts the conclusions of the inference mechanism into actual inputs for the process.

The crisp inputs are applied to the input side of fuzzification unit. The fuzzification unit converts the crisp input into fuzzy variable. The fuzzy variables are then passed through the fuzzy rule base. The fuzzy rule base computes the input according to the rules and gives the output. The output is then passed through defuzzification unit where the fuzzy output is converted into crisp output.

SELF-TUNING FUZZY LOGIC CONTROLLER

A self-tuning fuzzy controller is that in which the control rules, the membership function, or the scaling factors are self-adjusted. Among them, the control rules and the scaling factors play important roles [13]. We have used a real time tuning for output scaling factors. Its main advantages over a general FLC are stronger control capability, increased flexibility and robustness.

PI-type fuzzy controller gives very good steady-state concert but has transient concert not so good. For obtaining good transient and steady state concert, we may follow a strategy. The strategy is that at the beginning system has a positive large acceleration so we increase the scaling factor such that the rise time and settling time are reduced. Near the set-point when the system has a negative acceleration we reduce the scaling factors such that the overshoot is reduced or eliminated [12].

While controlling a plant, a skilled human operator manipulates the process input (i.e., controller output) based on error 'e' and change in error 'Δe' to minimize the error within the minimum time. Fuzzy logic control is a knowledge based system. The output SF should be determined very carefully for the successful and effective implementation of an FLC.

A. Development of Self Tuning Fuzzy Controller

Developing a generalized tuning method for FLCs is not an easy task because the computation of the optimal values of tuning parameters needs the required control objectives as well as a fixed model for the controller. A self-tuning PI-like FLC (STFLC) was used for the tuning of output scaling factor. Based on this self-tuning mechanism, the incremental change in controller output is obtained by the following equation:

$$\Delta u = (\alpha . Ku).\ \Delta uN \qquad (1)$$

where

$$\alpha\,(k) = f\,(e(k),\, ce(k)) \qquad (2)$$

control concert under load disturbance, the gain should be sufficiently large around the steady-state condition. Note that immediately after a large load disturbance, error may

International Journal of Research
International Journal of Research
Available at https://edupediapublications.org/journals
p-ISSN: 2348-6848
e-ISSN: 2348-795X
Volume 03 Issue 13
September 2016

be small but change in error will be sufficiently large (they will be of same sign) and, in that case, ‘α’ is needed to be large to increase the gain. At steady state (i.e., when error and change in error both are zero) controller gain should be very small. to avoid chattering problem around the set point we use the rule
IF ‘e’ is ‘ZE’ and ‘ce’ is ‘ZE’ THEN ‘α’ is ‘S’.

The rule base for α may also be modified further according to the type of response. It is very important to note that the rule base for computation of ‘α’ always depends on the choice of the rule base used in the main controller. Any significant change in the controller rule base may call for changes in the rule base for ‘α’ accordingly. Fig. 3, 4 and 5 shows membership functions for both inputs i.e. ‘e’ and ‘ce’ and for ‘α’ respectively.

here f, is a nonlinear function (computational algorithm) of e and ce, which is described by the rule base shown in Table 1 and the associated inference scheme.

Thus, when the controller is in operation, the gain of the self-tuning FLC will modify at each sampling time by the gain updating factor α. ‘α’ is obtained online based on fuzzy logic reasoning using the error and change of error at each sampling time rather than keeping it fixed. For improving the overall control concert, we use the rule base in Table 1, for the computation of α.

Table I Fuzzy Rules for Computation of α

e(k) / ce(k)	N	ZE	P
N	B	M	S
ZE	M	S	M
P	S	M	B

Practical processes or systems are often subjected to load disturbances. A good controller should provide regulation against changes in load; or we can say, it should bring the system to the stable state within a short time in the event of load disturbance. This is accomplished by making the gain of the controller as high as possible. Hence, to improve the

seven values (NB, NM, NS, ZE, PS, PM, PB). Membership function for inputs (e, ce) and output (K_p) are shown in fig. 7, 8 and 9 respectively. The rules for calculating ‘K_p’ are in table

2. In the implementation of FSPI scheme the proportional gain was tuned online by using FIS, but a constant integral gain was considered. So we can say this is semi fuzzy supervised PI controller.

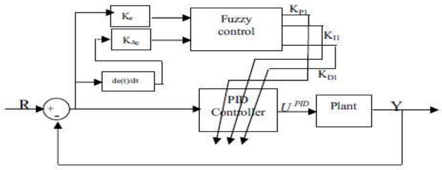

Fig. 3 Membership functions for ‘a’ of the ST FLC

FUZZY SUPERVISED PID CONTROLLERS (FSPID)

For controlling of a nonlinear process a conventional controller is not enough to obtain a desired concert. To ensure good concerts and stability for all the operation set point in nonlinear process, the controller gains should change to adopt the variation of physical parameters. PID controller may be used for such processes, but there is a significant need to develop methods for the automatic tuning of PID controllers for nonlinear systems. We can use fuzzy inference system to tune the PID controller gains for improving system concert. This provides a nonlinear mapping from the error signal e (t) and change in error ce(t), to the PID gain parameters K_p , K_i , and K_d [8].

A. *Design and Structure of FSPID Controller*

The fuzzy supervisor tunes the parameters of the conventional controller according to the nonlinearities of the process. Input to the fuzzy block can be any signal indicating a change in operation conditions which demands different values for the control parameters. Fig. 6 shows the basic structure of FSPID controller.

The fuzzy supervised PID controller, which takes error "e" and rate of change-in-error "ce" as the input to the controller makes use of the fuzzy control rules to modify PID gain parameters on-line. The supervising of the PID controller refers to finding the fuzzy relationship between the three parameters of PID K_p, K_i, and K_d and error "e" and change in error "ce" and according to the principle of fuzzy control modifying the three parameters in order to meet different requirements for control parameters when "e" and "ce" are different and making the control object produce a good dynamic and static concert.

The number of linguistic variables describing the fuzzy subset of a variable varies according to the application. Each of the input variables (“e”, “ce”) is assigned five fuzzy values (NB, NS, ZE, PS, PB). Here (NB, NS, ZE, PS, PB) is the set of linguistic values. For Kp and Ki the linguistic variables are

Kp	e(t)					
		NB	NS	ZE	PS	PB
Δe(t)	NB	VB	VB	VB	VB	VB
	NS	B	B	B	MB	VB
	ZE	ZE	ZE	ZE	S	S
	PS	B	B	B	MB	VB
	PB	VB	VB	VB	VB	B

Fig. 4 Membership functions for the input 'e' of the FLC for Kp.

The simulink model of different fuzzy controllers i.e. fuzzy logic controller (FLC), self-tuned fuzzy logic controller (STFLC), and fuzzy supervised PI controller (FSPI) are shown in fig. 10, 11 and 12 respectively.

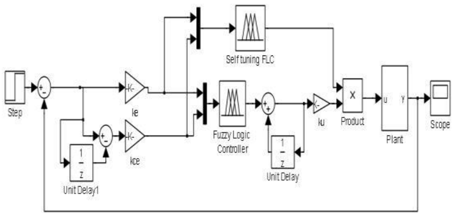

fig. 5 Membership functions for the output 'u' of the FLC for Kp

RISULTS & DISCUSSION

Example 2: A second order process with a time delay of 0.02 was considered as a second example. The plant transfer function is given by:

$$G_2(s) = \frac{500}{s^2 + 30s + 1000} e^{-0.02s} \qquad (4)$$

Example 3: It is also a second order process with time delay but in this the time delay is increased to 0.2 sec. The plant transfer function is:

$$G_3(s) = \frac{500}{s^2 + 30s + 1000} e^{-0.2s} \qquad (5)$$

For the comparision of different controllers the parameters considered were peak overshoot (M_P), Rise time (T_R) & Settling time (T_S). Comparisons between step responses of different systems using conventional PID controller and FLC of different types (FLC, GSFLC, STFLC, and FSPID) are shown in fig. 13, 14, and 15 and the significant values are given in Tables 3, 4 and 5 for systems described by equation 3, 4 and 5 respectively.

Simulation results were obtained using MATLAB/ SIMULINK. Simulation rsults are discussed for different types of Fuzzy logic controller. By comparing the experimental results, we can observe that, out of different types of FLCs (FLC, STFLC, FSPI), FSPI controller gives the best results. However, FSPI requires more expert knowledge and the rules should be judiciously chosen.

CONCLUSION

In this work the design of a PI-like fuzzy logic controller is described first. Then we described the design of a self-tuning fuzzy logic controller, where the output scaling factor was adjusted online depending on the process trend. Another category of fuzzy PI controllers "The Fuzzy supervised PI controller" was designed, where the proportional gains are tuned online based on fuzzy inference rules and reasoning and we use conventional integral gain.

Different configurations of the PI-type FLCs were implemented for different examples and the experimental results demonstrated the effectiveness of the PI-type FLCs. By comparing their concert, it is seen that the self-tuning PI-like FLC performed better than the PI-like FLC, but needs an extra set of inference rules for the online tuning of updating factor. This requires significantly more implementation effort than the PI-like FLC. Also, the FSPI Controller performed significantly better than the two kinds of PI-like FLCs, but the FSPI controller requires prior experience to design the proportional gain. This demonstrates the fact that employment of more information from the knowledge of experienced operator into the controller design, the concert of the controlled system increases.

REFERENCES

1. Wang, R. J., Zeng, J. & Zhou, D. W. (2012). Determination of temperature difference in squeeze casting hot work tool steel. *International journal of material forming*. 5(4), 317-324.

2. Parappagoudar M. B., Pratihar, D. K. & Datta, G. L. (2008). Forward and reverse mapping,s in green sand mould system using neural networks. *Applied Soft Computing*. 8(1), 239-260.

3. Parappagoudar, M. B., Pratihar, D. K. & Datta, G. L. (2007). Modelling of input–output relationships in cement bonded moulding sand system using neural networks. *International Journal of Cast Metals Research*. 20(5), 265-274.

4. Kittur, J. K. & Parappagoudar, M. B. (2012). Forward and reverse mappings in die casting process by neural networkbased approaches. *J. Manuf. Sci. Prod.* 12(1), 65-80.

5. Parappagoudar, M. B., Pratihar, D. K. & Datta, G. L. (2008 Neural network-based approaches for forward and reverse mappings of sodium silicate-bonded, carbon dioxide gas hardened moulding sand system. *Materials and Manufacturing Processes*.

6. Benguluri, S., Vundavilli, P. R., Bhat, R. P. & Parappagoudar, M. B. (2011). Forward and reverse mappings
in metal casting—A step towards quality casting and automation, (11-009). *AFS Transactions-American Foundry Society*. 119(19), 1-15,

7. Surekha, B., Vundavilli, P. R., Parappagoudar, M. B. & Srinath, A. (2011). Design of genetic fuzzy system for

forward and reverse mapping of green sand mould system. *International Journal of Cast Metals Research*. 24(1), 53-64.

8. Surekha, B., Vundavilli, P.R. & Parappagoudar, M.B. (2012). Forward and reverse mappings of the cement-bonded sand mould system using fuzzy logic, *The International Journal of Advanced Manufacturing Technology*. 61(9-12), 843-854.

9. Vundavilli, P. R., Parappagoudar, M. B., Kodali, S. P. & Benguluri, S. (2012). Fuzzy logic-based expert system for prediction of depth of cut in abrasive water jet machining process. *Knowledge-Based Systems*. 27, 456-464.

10. Surender, Y. & Pratihar, D. K. (2013). Fuzzy Logic Logic-Based Techniques for Modeling the Correlation between the Weld Bead Dimension and the Process Parameters in MIG Welding. *International Journal of Manufacturing Engineering*. http://dx.doi.org/10.1155/2013/230463.

11. Patel, G. C. M., Mathew, R., Krishna, P. & Parappagoudar, M. B. (2014). Investigation of squeeze cast process parameters effects on secondary dendrite arm spacing using statistical regression and artificial neural network models, (Accepted for publication in Procedia Technology Elsevier Journal).

12. Pratihar, D. K. (2008). *Soft computing*, Narosa publishing house pvt. Ltd, India.

13. Azar, A. T. (2010). Adaptive neuro-fuzzy systems, *Fuzzy systems*, 85-110, Feburaury 2010, INTECH, Croatia, ISBN 978-953-7619-92-3.

14. Haykin, S. (1994). *Neural networks: a comprehensive foundation*. Prentice Hall PTR.