Asian Journal of Information Technology 5 (1) : 34-37, 2006

Design with Simulation of an Iterative Fuzzy Logic Controller (IFLC)

Mohammed Mahdi Ali
Philadelphia University, Jordan

Abstract: As the need for control is extended to systems of increasing complexity which is also often highly non-linear, rule-based fuzzy logic control was one of the successful solutions. An Iterative Fuzzy Logic Controller (IFLC) is designed based on finding the best input-output scale factors settings that are found during iterations to give an acceptable controlled behaviour. To verify the ability of the proposed IFLC, a promising simulation study on different kinds of control systems has been pointed out.

Key words: Iterative Fuzzy Logic Controller (IFLC), Fuzzy Production Rules (FPRs), Input-Output scale factors (Ge, Gce and Gu)

INTRODUCTION

Fuzzy concepts derive from fuzzy phenomena that commonly occur in the natural world. The concepts formed in human brains for perceiving, recognition and categorizing natural phenomena are often fuzzy concepts[1].

Fuzzy logic was first introduced by Zadeh[2,3], as a means for handling and processing vague, linguistic information. Zadeh reasoned that:-Conventional quantitative techniques of system analysis are unsuited for dealing with humanistic systems, thus fuzzy logic has been developed to provide soft information processing algorithms which can reason about and utilize imprecise data[4]. It allows variables to be partial members of a particular set and uses generalization of the conventional Boolean logical operators to manipulate this information. By allowing partial membership of a set, it is possible to represent the smooth transition from one rule to another as the input is varied smoothly, which is a very desirable property in modeling and control applications. In contrast, conventional expert systems reason using hard, crisp rules and are unable to represent a smooth input-output transformation.

Fuzzy logic is widely used in intelligent control to reason about vague rules which describe the relationship between imprecise, linguistic assessment of the system's input and output states. These production rules are generally natural language representation of human's (or expert's) knowledge and provide an easily understood knowledge representation scheme for explaining information learnt by a computer. Fuzzy Production Rules (FPRs) are described using vague terms such as *small* or *medium* to categorize the input and output variables and the set of all these rules forms a fuzzy algorithm[3,4].

Recently, FLCs are finding increasing use in industry. The main benefit of this approach can be stated as:

- The difficulty of real-time process control is due to its nonlinear, time-varying behavior. Also, in some modern plants with process control, plants models have been used to calculate the required controller settings. The plant models based on parameter estimations methods are approximations to the real process and may require a large amount of computer time. All these problems can be avoided using FLC[5,6].

While many questions concerning this approach[5,7] arise mainly from:-

- Its non-numerical nature.
- Stability of the overall control system.
- Completeness of the rules.

IFLC Design: Two precise inputs, the error and its rate of change as for control applications, are presented to the fuzzy system. The degree of membership of each of the linguistic fuzzy sets is calculated (fuzzification) and this knowledge is combined to represent the degree of belief of the antecedent in the Fuzzy Production Rule (FPR). The output of this fuzzy rule is the respective fuzzy output set which is scaled by a parameter which signifies the confidence in this rule is true and the overall fuzzy output set is formed from the contributions of each fuzzy rule. A real-valued output is then obtained by defuzzifying this output set. The block diagram of the fixed rules Fig.1 FLC[4,5].

The function of the input scaling factors Ge and Gce is to map the measured variables of error and its rate of

Corresponding Author: Mohammed Mahdi Ali, Philadelphia University, Jordan

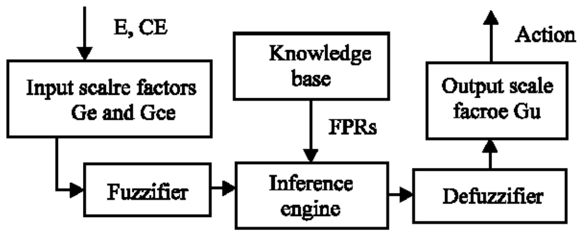

Fig. 1: FLC schematic diagrame

	NCEB	NCEM	NCES	ZCE	PCES	PCEM	PCEB
NEB	NUB	NUB	NUB	NUB	NUM	NUS	ZU
NEM	NUB	NUM	NUM	NUM	NUS	ZU	PUS
NES	NUB	NUM	NUS	NUS	ZU	PUS	PUM
ZE	NUB	NUM	NUS	ZU	PUS	PUM	PUB
PES	NUM	NUS	ZU	PUS	PUS	PUM	PUB
PEM	NUS	ZU	PUS	PUM	PUM	PUM	PUB
PEB	ZU	PUS	PUM	PUB	PUB	PUB	PUB

Fig. 2: 49-fuzzy production rule

change into their predefined universe of discourse respectively. Trial and error procedure to find the best setting for input gains and output gain GU to get an acceptable output response is very difficult problem especially when a real-time implementation is desired.

In[8] some suggestions were presented to obtain suitable input gains settings as below:

Without loss of generality the initial value of the process output y(0) is set to be zero, then for the first iteration the max absolute value of the measured error $|e_m|_{max}$= Input and therefore Ge can be initially set, however, the max absolute value of the measured rate of change of error $|ce_m|_{max}$ cannot be easily decided in advance. Fortunately, it can be obtained during iterations. At the end of each iteration, all the data of are available and the maximum absolute value $|ce_m|_{max}$ is found then Gce can be set.

With regard to the output scaling factor GU, it is simply set to:

1/max[Ge, or Gce], in order to ensure that the control action will be applicable and will not drift the system to unstability especially during the first iteration, while in the next iterations all gains will be set automatically as summarized below:

For the first iteration (initial gains settings):-

$$Ge = Vn / Input,\ Gce = 1.0 \text{ and } GU = 1/Ge \quad (1)$$

But for the next iterations:-

$$Ge=Vn/|e_m|max \quad Gce=Wm/|ce_m|max$$
$$\text{and } GU = 1/max[Ge, \text{ or } Gce] \quad (2)$$

Where;

Vn is the error universe of discourse with positive limit V and card n. Wm is the change of error universe of discourse with positive limit W and card m.

For the underlying cases of study V and W are set to 10, while n and m are set to 21.

A stopping iteration criterion is taken based on minimizing a Performance Index (P.I) of the form:

$$P.I = 0.5 \int_0^T e_m{}^2 dt \quad (3)$$

On the condition that the controlled output response meets the transient and steady-state requirements.

A triangular fuzzifier is used to map crisp values into 7-fuzzy sets of NB, NM, NS, Z, PS, PM, PB for error, change of error and control action to bring 49-fuzzy production rules as in Fig. 2 finally a center of gravity defuzzifier is implemented.

RESULTS

The main aim of the simulation is to examine the behaviour of the proposed IFLC in dealing with different kinds of control systems. The situations concerned in this simulation study are divided into three main groups: Third order, unstable and non-linear control systems.

Third order system: A continuous type 1 third order system described by:

$$G(s) = \frac{2}{s(s^2 + 3s + 2)} \quad (4)$$

is taken to be controlled by the proposed iterative FLC with Ts = 0.05 sec. During 3-iteration with U = -18 to 18 step 1.8 the final gains are found to be Ge = 10.0,Gce = 9.7 and Gu = 0.1 that give P.I = 11.9. The controlled response is compared with that of conventional PID controller[9] and with that of a hybrid neural-network based on self organizing controller (SONNC)[10] along with the uncontrolled one through Fig. 3. It shows that the IFLC behaves well with no overshoot compared with that of the PID at the expense of less speed, while (SONNC) is obtained based on a sluggish reference model of the form:

$$G_m(s) = \frac{1}{s^2 + 5.4s + 1} \quad (5)$$

since the implemented neurocontroller diverges by using fast model in controlling high order systems[10].

Unstable second order system:
Transfer function of the form:

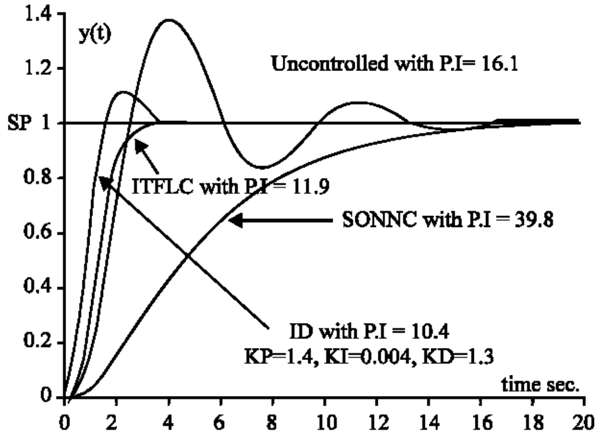

Fig. 3: Comparison of third order system controlled responses with that of uncontrolled one.

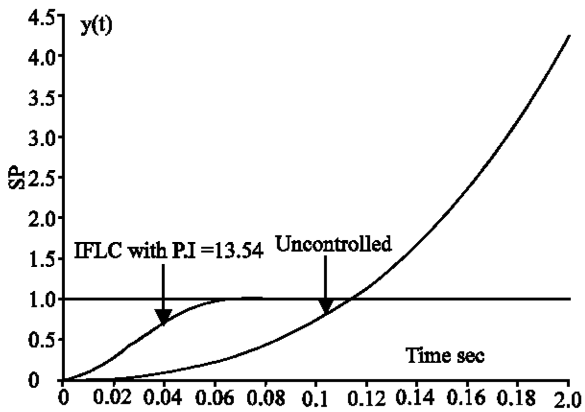

Fig. 4: IFLC response along with uncontrolled one of unstable system

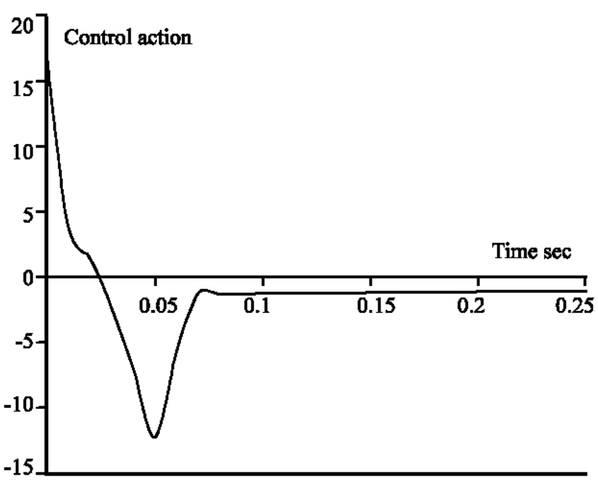

Fig. 5: Control action provided by IFLC to control the unstable system.

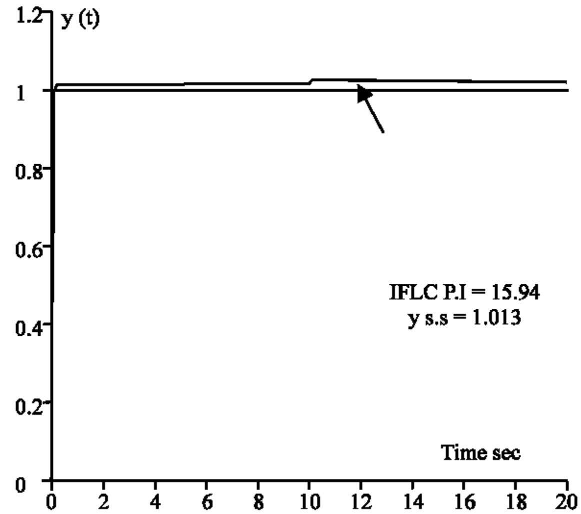

Fig. 6: Effect of disturbance on the controlled response of Fig. 4

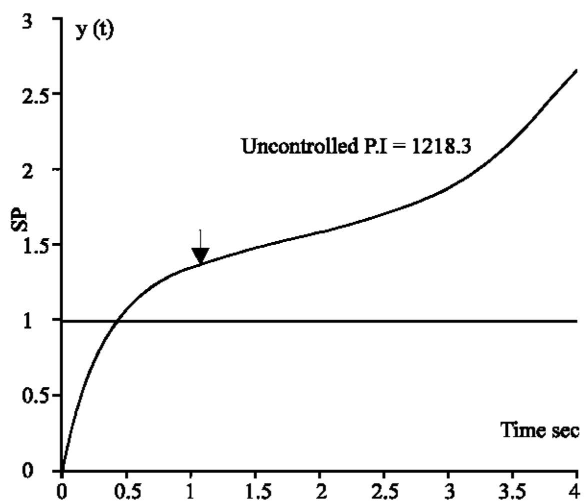

Fig. 7: Uncontrolled response of the non linear system

$$G(s) = \frac{100}{s^2 - 100\,s - 100} \qquad (6)$$

describes a continuous unstable second order system[11] and is taken to be controlled using the already designed controller with Ts = 0.001 sec. Using the IFLC for 5-iteration with U = -200 to 200 step 20 the final gains are found to be Ge = 10.0, Gce = 0.419 and Gu = 0.1 with a controlled response with P.I of 13.64. The large values of the control action vectors are used because of the high system's nstability at high speed due to the inherently large gain. Fig. 4 illustrates the IFLC response along with that of the uncontrolled system for a very short period of time for better recognition. On the other

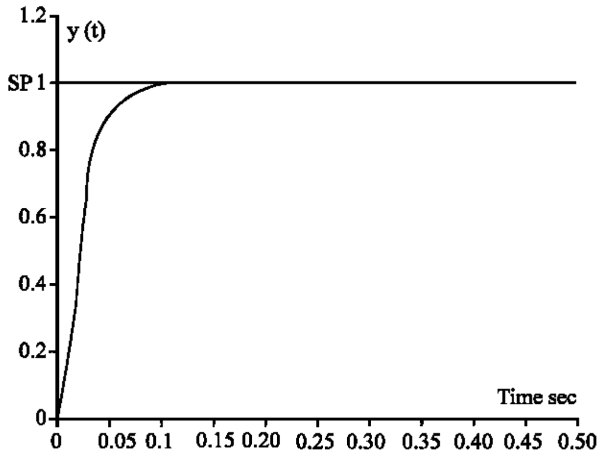

Fig. 8: IFLC response of the non-linear system

hand, (Fig 5) clarifies the reasonable control action provided by the IFLC.

Figure 6 shows a challenge of using a disturbance of 0.7 of input applied to the output of the controller and fed to the controlled process all along the period of interest. The excellent behaviour of the IFLC controller which yields a steady-state deformation of 0.013 is noticed.

Nonlinear second order system: A state and output equations of a nonlinear second order system are described as:

$$\left.\begin{aligned} \dot{x}_1 &= \sin x_1 + x_2 + 4\,u \\ \dot{x}_2 &= x_1^2 - \sin x_2 + u \\ y &= x_1, \text{ with } \underline{x}(0) = 0 \end{aligned}\right] \qquad (7)$$

With free input (u = 0.0), the singular point of the given nonlinear system is (0,0). Fig. 7 shows the uncontrolled unity feedback system response for only four seconds using Ts = 0.001 sec. The IFLC is used with an accumulated change in control of: U = -10.0 t0 10.0 step 1.0. Within 3 iterations the final gains are found to be: Ge = 10.0, Gce = 0.258 and Gu = 0.1 with a P.I = 8.8 as shown in Fig. 8.

CONCLUSIONS

Many points of high importance can be concluded as below: -

- FLC is a good remedy for systems that can not be modeled easily.
- Using Input-output scale factors helps in mapping variables into a unified Input-Output universe of discourses.
- ITFLC shows a powerfull feature in controlling systems through a small number of iterations.
- ITFLC can be used easily especially when a real-time implementation is considered.
- ITFLC results comparision were found better than others (conventional and modern) controllers.

REFERENCES

1. Li, H., *et al,* 2001. Fuzzy Neural Intelligent Systems: Mathematical Foundation and the Applications in Engineering C RC Press LLC,
2. Zadeh, L.A., 1965. Fuzzy sets, information and control, pp: 338-353.
3. Zadeh, L.A., 1973. Outline of a new approach to the analysis of complex systems and decision processes, IEEE Transactions on Systems, Man and Cybernetics, pp: 28-44.
4. Brown, M. and C. Harris, 1994. Neurofuzzy adaptive modelling and control, Prentice Hall International Ltd., UK.
5. Mamdani, E.H., 1974. Application of fuzzy algorithms for control of simple dynamic plant, IEEE Proceeding, pp: 1585-1588.
6. Self, K., 1990. Designing with fuzzy logic, IEEE Spectrum, pp: 42-44.
7. Umbers, I.G. and P.J. King, 1979. An Analysis of human decision-making in cement kiln and the implication for automation, Crown Copyright, pp: 369-381.
8. Linkens, D.A. and J. Nie, 1993. Constructing rule-bases for multivariable fuzzy control by self-learning-Part 2, Intl. J. Sys. Sci, pp: 129-157.
9. Phillips, C.L. and H.T. Nagel, 1984. Digital control system analysis and design, Prentice-Hall Inc., Englewood Cliffs, USA.
10. Nie, J. and D.A. Linkens, 1994. A hybrid-neural-based Self-Organizing controller, Intl. J. Control, pp: 197-222.
11. Chen, Y.Y. and K.Z. Lin, 1992. Learning behaviour of fuzzy controller with neuron adaptive elements, ACC/TM3.