Truncation Effects of the Fuzzy Logic Controllers

Byung Soo Moon*, Je Sun Moon*, and Jongmin Lee*

ABSTRACT

Fuzzy logic controllers are often found to behave better than PI controllers. One of the major reasons for this is that the fuzzy logic inferences used can produce nonlinear type controllers. For some applications, however, linear fuzzy logic controllers also perform better than PI controllers. In this paper, we examine linear fuzzy logic controllers to show that the truncation effects of the fuzzy logic controllers make them perform much better than the PI controllers. In terms of a performance index we used, the truncation effects reduced the index value by up to 80% for examples we studied.

I. Introduction

A linear fuzzy logic controller(FLC) for a single input system is equivalent to a proportional integral controller(PI) under ceriain choices of fuzzy logic operations, while they are not if some other fuzzy logic operations are chosen [1]. It is also proved [2] that when a PI controller is given, one can design an FLC whose output is identical to that of the PI controller and that an FLC designed by using specified fuzzy logic operations is essentially a PI controller.

We find, however, that even when we follow the specified fuzzy logic operations, if less number of fuzzy sets are used than what is necessary, then the resulting FLC performs better than the given PI controller. In this paper, we examine how the truncation effects caused by using less number of fuzzy sets than necessary affects the performance of the FLC. We choose our system to be controlled as

$$y' = -5(y-3) - 3we^{-5t} \sin(wt + \Pi) \tag{1}$$
$$y(0) = 0, \text{ set point} = 3.0,$$

where the value of w varies from 65 to 520.

The above equation (1) may be viewed as representing the response of a system when no controller is applied. One can easily compute the analytical solution of this system as

$$y(t) = 3 + 3e^{-5t} \cos(wt + \Pi) \tag{2}$$

*Korea Atomic Enengy Research Institute

whose graph for $w=65$ is shown in fig.1. In fact, we have started with (2) to derive the system (1) to be controlled.

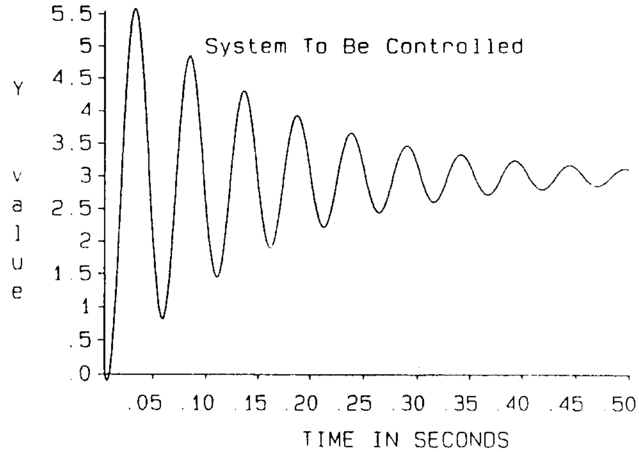

fig 1. The System to be Controlled

Ⅱ. A Proportional Integral Controller

In order to apply a PI controller to the system (1), we start with an analysis on how to tune the PI controller. Note that a desired form of the controller output is of the form

$$Z=3e^{-5kt}\cos(wt+\Pi)+3 \tag{3}$$

where k is a positive integer and w is of the same value as in (1). Since the settling time for Z becomes shorter as k increases, we will be adjusting k for the tuning of the PI controller.

From equation (1), we have $y''=-5y'-h'(t)$ where $h(t)=3we^{-5t}\sin(wt+\Pi)$ and a further caluculation shows

$$\begin{aligned} y'' &= -10y'-(25+W^2)(y-3) \\ &= F(t, y, y') \end{aligned} \tag{4}$$

Repeating a similar process for (3), we also obtain

$$Z''=-10kZ'-(25k^2+W^2)(Z-3) \tag{5}$$

Normally, the controller should satisfy $Z'=f(t, Z)+u$, where $f(t, Z)=-5(Z-3)-h(t)$ which is derived from (1), and u is tuned so that a performance index is of a minimal value.

In this case, however, we try our controller u to satisfy

$$Z'' = F(t, z, z') + u' = -10Z' - (25 + W^2)(Z-3) + u'. \quad (6)$$

Equating (5) and (6), we obtain

$$u' = -10(k-1)Z' - 25(k^2-1)(Z-3) = -10(k-1)e' - 25(k^2-1)e, \quad (7)$$

where $e = z - 3$ is the input error. If we integrate both sides of (7), we get

$$u = -10(k-1)e - 25(k^2-1)\int_0^t e(s)ds + u_o$$

or equivalently,

$$u_n = -10(k-1)\, e_n - 25(k^2-1)\, h \sum_{i=1}^{n} e_i + u_o \quad (8)$$

Where h is the time step size and $e_i = z_i - 3$. We apply this PI controller to the system

$$Z' = -5(Z-3) - 3we^{-5t}\sin(wt + \prod) + u \quad (9)$$

When the Adams-Moulton's method is applied to the case of $k = 101$, $w = 260$, and with the scanning period to be the same value as $h = 0.0001$, we obtain the result shown in fig.2.

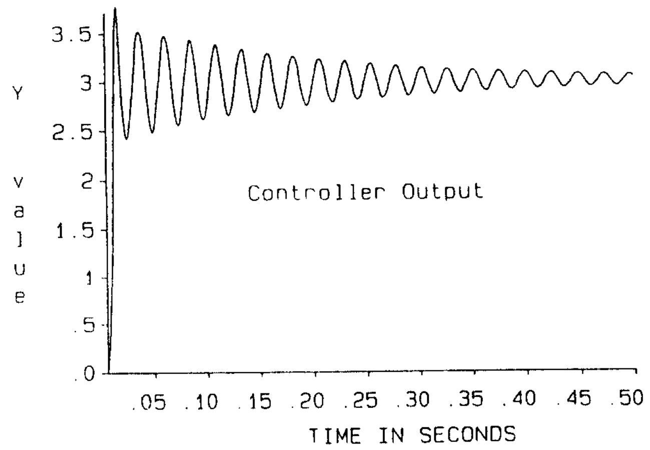

fig 2. Controlled Result by a PI Controller

Ⅲ. An Equivalent Fuzzy Logic Controller

In this section, we will describe a fuzzy logic controller which produces the identical output as the PI controller (8) with $k=101$, $u_o=o$. For the fuzzification of variables e_n, s_n and u_n-u_o, we first take a bound M common for all of $|10(k-1)|e_n|=|1000e_n|$, $|25(k^2-1)hs_n|=|25.5s_n|$, $|u_n-u_o|=|u_n|$, $n=1, 2,\cdots 5000$. The value of M is selected based on the response of the PI controller and adjusted afterwards.

We take an arbitrary positive number $m \geq 1$ and let $2m+1$ be the number of fuzzy sets for e_n and s_n respectively, while we use $4m+1$ fuzzy sets for the controller output U_n-U_0. The results shown in section 4 are obtained using $m=5$ but it turns out that the choice of m does not affect the controller output in our case.

We then define

$$\nu=\frac{M}{m}, \quad \lambda=\frac{\nu}{10(k-1)}, \quad \mu=\frac{\nu}{25(k^2-1)h} \tag{10}$$

and use the standard triangular functions defined on equally spaced points with the interval length of λ for e_n, μ for s_n, and ν for u_n as our fuzzy sets. These fuzzy sets are numbered from left toward right as 1, $2,\cdots(2m+1)$ or $(4m+1)$.

The fuzzy control rules are defined by

$$r=4m+3-(p+q) \tag{11}$$

were p is the fuzzy set number for e_n, q is for s_n and r is for u_n. Hence, if P, Q, R are the corresponding fuzzy sets with numbers p, q and r respectively, then the control rules will be of the form

If e is P and s is Q, then u is R
If e is A and s is B, then u is C. (12)

For the fuzzy set intersection operation, we use the one by Dubois & Praide with $\alpha=1$ [3, p50], i.e.

$$\mu_{P\cap Q}(e_n, s_n)-\mu_P(e_n)\mu_Q(s_n), \tag{13}$$

and the Larsen's Product Operation Rule is used for the fuzzy logic implications [4, p429]. Thus, the fuzzy sets resulted from (12) when $e=e_n$, $s=s_n$ are

$$\mu_p(e_n)\mu_Q(S_n)\mu_R \quad \text{and} \quad \mu_A(e_n)\mu_B(s_n)\mu_c$$

Finally, for the combination of the two rules in (12) and for the defuzzification of the result when $e=e_n$, and $s=s_n$, we use

$$(r\mu_p(e_n)\mu_Q(s_n)+c\mu_A(e_n)\mu_B(s_n))/(r+c) \tag{14}$$

Where r and c are the centers of the support for the fuzzy sets R and C respectively.

It is proved in [2] that the fuzzy logic controller designed above should produce the same result as the PI controller which we started with in section 2. The simulation experiments described in the next section also showed that the above is true.

Ⅳ. The Truncation Effects and Simulation Experiments

In order to study the truncation effects on the FLC, we may either reduce the number of fuzzy sets so that we use less than what is necessary or decrease the uniform support lengths λ, μ and ν by certain factors. We choose the latter method and apply a common factor ('truncation factor' in Tables 1, 2) for both e_n and s_n. As it turned out, the truncation effect for u_n is minor for our examples, and it is not considered in this paper.

For a quantitative comparison of the fuzzy logic controller against the PI controller, one must define a performance index. We define one by

$$PI=\sum_{i=1}^{n} e_i^2 + \sum_{i=1}^{n} \left(\frac{u_i}{2500}\right)^2 \tag{15}$$

The quantity 2500 is chosen so that the two sums on the right hand side of (15) become of the same order of magnitude.

With a truncation factor of 0.6 common for e_n and s_n, the fuzzy logic controller is applied to the system (9) to obtain the result shown in fig. 3. Table 1 shows the comparison of the performance index between the PI controller and the FLC when the same factor of 0.6 is applied to the system with different w values. The row titled SSE in Table 1 is for the sum of squared errors (e_n). Note that both the performance index and the SSE are reduced to about 50% of those for the PI controller.

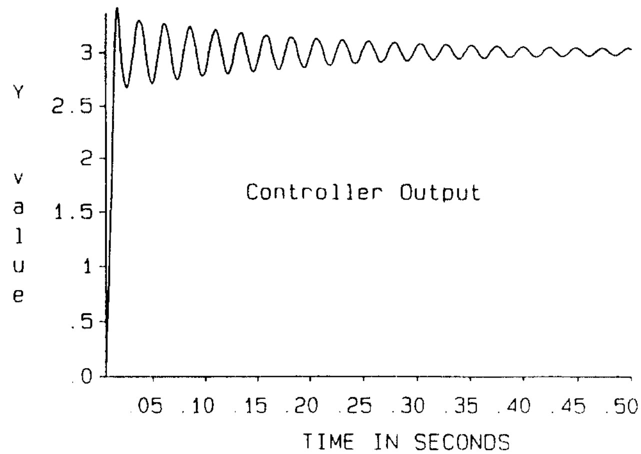

fig 3. Controlled Result by FLC with Truncation Factor 0.6

Table 2 shows the result obtained when the common truncation factor for e_n and s_n is varied from 0.25 to 1.0. Note that when the factor is 1.0, we are not truncating at all and hence the result should be the same as the one obtained by the PI controller in Table 1.

Table 1. Comparison Between PI and FLC(Truncation Factor = 0.6)

		Exp.1 w=65	Exp.2 w=130	Exp.3 w=260	Exp.4 w=520
PI	Performance Index	39.04	68.01	296.95	1497.5
	SEE	31.43	50.04	230.62	1247.6
FLC	Performance Index	23.86	40.48	142.26	676.99
	SEE	16.85	23.24	82.10	437.41

Table 2. Truncation Effects of FLC

	Factor 0.25	Factor 0.5	Factor 0.75	Factor 1.0
Performance Index	292.40	533.43	941.35	1497.5
SSE	76.69	299.61	695.43	1247.6

V. Conclusion

For linear fuzzy logic controllers, we have shown that by utilizing the truncation effects, one can enhance the performance of the controller greatly, i.e reduce the settling time and reduce the overshooting. In other words, one of the major reasons why the fuzzy logic controllers perform better than the PI controllers is that the truncation effects can be utilized in the FLCs. This truncation effects, however, can also be implemented on a digital PI controller if one wishes to do so.

References

1. William siler and Hao Ying, "Fuzzy Control Theory: The Linear Case," Fuzzy sets and systems, Vol. 33(1989), 275-290.
2. Byung Soo Moon, "Equivalence Between Fuzzy Logic Controllers and PI Controllers For single Input Systems," Fuzzy Sets and Systems, To be accepted, 1994.
3. George J. Klir and Tina A. Folger, "Fuzzy Sets, Uncertainty, and Information," Prentice Hall(1988).
4. Chuen Chien Lee, "Fuzzy Logic in Control Systems: Fuzzy Logic Controller-Part II," IEEE Trans. on Systems, Man and Cybernetics, Vol.28, No2 (1990), 419-435.