Proceeding of the 2004 American Control Conference
Boston, Massachusetts June 30 - July 2, 2004

Fuzzy Proportional Integral - Proportional Derivative (PI-PD) Controller

M. P. Veeraiah S. Majhi Chitralekha Mahanta

Abstract— **In this paper we propose a fuzzy PI-PD controller that is tuned by using Genetic Algorithm(GA). The fuzzy PI-PD controller preserves the linear structure of the conventional one, but has self-tuned gains. The proportional, integral and derivative gains are nonlinear functions of their input signals having certain adaptive capability in set-point tracking performance. The proposed design is then optimized using the GA. Our proposed controller is validated by applying both linear and nonlinear test signals. The results demonstrate that gain optimization using GA leads to better transient performance of the proposed fuzzy PI - PD controller.**

I. Introduction

Proportional-Integral-Derivative(PID) Controllers[1], [2], [3], [4] are extensively used in process industries. The classical PID controller, apart from the derivative kick, is suitable for controlling stable "time constant" plants with small time delays, which are typical of many plant transfer functions. Difficulties are often faced, however, while controlling plants with resonance, integral or unstable transfer functions[5].

PI-PD controller proposed by Majhi S and Atherton D.P[6] is a modified form of PID controller. PI-PD controller, which corresponds to PI control of the plant transfer function changed by the PD feedback, can produce improved control in several situations. This implementation avoids the derivative kick problem associated with derivative action in the forward path, which still exists when filter is included. Further, the PD in the inner feedback loop can enable placement of the open loop poles in appropriate positions, thereby providing good control for open loop system transfer functions having resonances, unstable or integrating poles. The parameters of the PI-PD controller are obtained by minimization of the integral of squared time weighted error (ISTE) criterion, which usually produces a step response of desirable form, although other criteria could be used.

Conventional PI-PD controllers generally do not work well for nonlinear systems, higher order and time-delayed linear systems, and particularly complex and vague systems which do not have precise mathematical models. In order to overcome these difficulties, a class of nonconventional fuzzy PI-PD controllers have been designed by us.

S. Majhi is with the Department of Electronics and Communication Engineering, IITGuwahati, Guwahati-781039, Assam, India smajhi@iitg.ernet.in

C. Mahanta is with the Department of Electronics and Communication Engineering, IITGuwahati, Guwahati-781039, Assam, India chitra@iitg.ernet.in

Our proposed fuzzy PI-PD controller has the following features:

(1) It has the same linear structure as that of the conventional PI-PD controller, but has non-constant coefficient and self-tuned control gains (they are the nonlinear functions of the input signal).

(2)The controller is designed based on the classical discrete PI-PD controller, from which the fuzzy control law is derived.

(3)Membership functions are simple triangular ones with only four fuzzy logic if-then rules.

The fuzzification, control- rule execution, and defuzzification steps are embedded in the final formulation of the designed fuzzy control law. The resulting control law is a small set of explicit conventional formulas. Therefore, the controller works just like a conventional PI-PD controller. The fuzzification rules and defuzzification routine are not needed throughout the entire control process.

Optimal fuzzy PI-PD gain parameters are obtained by using Genetic Algorithm[7], [8], [9]. The outstanding performance of our proposed fuzzy PI-PD controller is demonstrated by computer simulations on a couple of linear and non linear systems.

This paper is organized as follows. In Section II we discuss the design procedure of our proposed fuzzy PI - PD controller. Fuzzification, control rule base and defuzzification procedure are elaborated in Section III. Computer simulation results are presented in Section IV. Section V contains our conclusions.

II. Fuzzy PI-PD controller

We have used an arrangement of fuzzy PI-PD control units as shown in Fig.1. This arrangement is often desirable if the reference input contains discontinuities[10]. The derivation of the fuzzy control law is performed in two steps: one for the output of the fuzzy PI[11] controller and the other for the output of the fuzzy PD[12] controller. The final control law combines these two individual control laws together in an appropriate way, as described in more detail in the following sections.

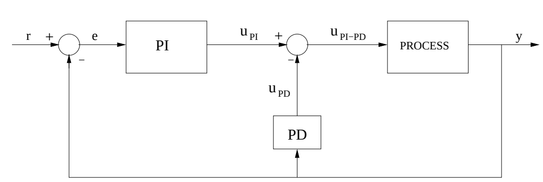

Fig. 1. The conventional continuous-time PI-PD control system

0-7803-8335-4/04/$17.00 ©2004 AACC

The fuzzy PI-PD controller that we have designed is a digital controller. Hence, we start with a continuous conventional PI-PD controller[6] and then use the standard bilinear transformation to convert it to the corresponding digital controller.

A. Derivation of the fuzzy PI controller

The output of a conventional analog PI controller in the frequency domain is given by

$$u_{PI}(s) = \left(K_p^c + \frac{K_i^c}{s}\right)E(s), \tag{1}$$

where K_p^c and K_i^c are the proportional and integral gains, respectively, and $E(s)$ is the error signal. This equation can be transformed into the discrete version by applying the bilinear transformation

$$s = \frac{2}{T}\frac{(z-1)}{(z+1)}, \tag{2}$$

where $T > 0$, is the sampling period. The resulting equation has the following form:

$$u_{PI}(z) = \left(K_p^c - \frac{K_i^c T}{2} + \frac{K_i^c T}{1-z^{-1}}\right)E(z). \tag{3}$$

Let,

$$K_p = K_p^c - \frac{K_i^c T}{2}$$

and

$$K_i = K_i^c T.$$

Taking inverse z-transform in (3), we have

$$u_{PI}(nT) - u_{PI}(nT-T) = K_p\Big[e(nT) - e(nT-T)\Big] + K_i e(nT). \tag{4}$$

Dividing the above equation by T, we obtain

$$\triangle u_{PI}(nT) = K_p e_v(nT) + K_i e_p(nT), \tag{5}$$

where

$$\triangle u_{PI}(nT) = \frac{u_{PI}(nT) - u_{PI}(nT-T)}{T},$$

$$e_v(nT) = \frac{e(nT) - e(nT-T)}{T},$$

and

$$e_p(nT) = e(nT).$$

More precisely, $\triangle u_{PI}(nT)$ is the incremental control output of the PI controller, $e_p(nT)$ is the error signal and $e_v(nT)$ is the rate of change of the error signal. We can rewrite (4) as

$$u_{PI}(nT) = u_{PI}(nT-T) + T\triangle u_{PI}(nT). \tag{6}$$

In the design of the fuzzy PI controller to be discussed later, we will replace the term $T\triangle u_{PI}(nT)$ by fuzzy control action $K_{uPI}\triangle u_{PI}(nT)$, so that

$$u_{PI}(nT) = u_{PI}(nT-T) + K_{uPI}\triangle u_{PI}(nT), \tag{7}$$

where K_{uPI} is a fuzzy control gain to be determined later.

B. Derivation of the Fuzzy PD controller

In Fig.1, it is clearly seen that the PD controller has y as its input and u_{PD} as its output. So, the conventional analog PD controller is represented as

$$u_{PD}(t) = K_p^{c'} y(t) + K_d^c \dot{y}(t) \tag{8}$$

In the frequency domain, the above equation becomes

$$u_{PD}(s) = (K_p^{c'} + K_d^c s)Y(s) \tag{9}$$

The above continuous domain equation is converted into discrete domain by applying bilinear transformation,

$$s = \frac{2}{T}\frac{(z-1)}{(z+1)}, \tag{10}$$

where $T > 0$ is sampling period. The resulting equation has the following form:

$$u_{PD}(z) = (K_p^{c'} + \frac{2}{T}\frac{(z-1)}{(z+1)}K_d^c)Y(z). \tag{11}$$

The above equation (11) is simplified as

$$u_{PD}(nT) + u_{PD}(nT-T) = K_p'(y(nT) + y(nT-T)) + K_d(y(nT) - y(nT-T)), \tag{12}$$

where

$$K_p{}' = K_p^{c'}$$

and

$$K_d = \frac{2}{T}K_d^c.$$

Dividing the above equation by T, we get

$$\frac{u_{PD}(nT) + u_{PD}(nT-T)}{T} = \frac{K_p'(y(nT) + y(nT-T))}{T} + \frac{K_d(y(nT) - y(nT-T))}{T},$$

$$\textit{or}, \quad \triangle u_{PD}(nT) = K_p' d(nT) + K_d \triangle y(nT),$$

where

$$\triangle u_{PD}(nT) = \frac{u_{PD}(nT) + u_{PD}(nT-T)}{T},$$

$$d(nT) = \frac{y(nT) + y(nT-T)}{T},$$

$$\triangle y(nT) = \frac{y(nT) - y(nT-T)}{T}.$$

Again, we have,

$$u_{PD}(nT) = -u_{PD}(nT-T) + T\triangle u_{PD}(nT). \tag{13}$$

$\triangle u_{PD}(nT)$ becomes the fuzzy control action in the new design, where we will use K_{uPD} as a fuzzy control gain, which will be determined later in the design and rewrite it as

$$u_{PD}(nT) = -u_{PD}(nT-T) + K_{uPD}\triangle u_{PD}(nT). \tag{14}$$

C. Combination of the fuzzy PI-PD controller

The overall fuzzy PI-PD control law can be obtained by algebraically summing the fuzzy PI control law (7) and the fuzzy PD control law (14) together. The resulting law is

$$u_{PI-PD}(nT) = u_{PI}(nT) - u_{PD}(nT)$$

or

$$u_{PI-PD}(nT) = u_{PI}(nT-T) + K_{uPI}\triangle u_{PI}(nT) + u_{PD}(nT-T) - K_{uPD}\triangle u_{PD}(nT) \quad (15)$$

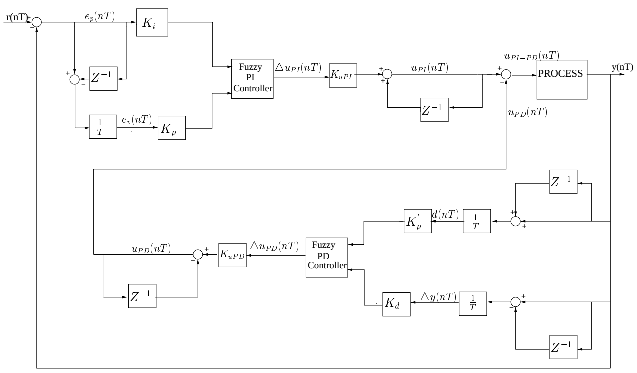

Fig. 2. The fuzzy PI-PD control system

The fuzzy PI-PD control system (15) is shown in Fig. 2. This fuzzy PI-PD controller is obtained by inserting the fuzzy PI and the fuzzy PD controllers in the conventional PI -PD controllers[6].

III. Fuzzification, control rule base and defuzzification

In this section, we follow the standard procedure of fuzzy controller design consisting of fuzzification, control rule base formulation and defuzzification.

A. Fuzzification

We fuzzify the PI and PD components of the PI-PD control system individually and then combine the desired fuzzy control rules for each of them. The overall PI-PD fuzzy control law given in (15) is taken into consideration during fuzzification. Similar to the fuzzy PI controller[13], the input and the output membership functions of the PI component are shown in Fig.3.

The fuzzy PI controller employs two inputs, namely, the error signal $e_p(nT)$, and the rate of change of error signal $e_v(nT)$. The fuzzy PI controller has a single output called the incremental control output and is denoted by $\triangle u_{PI}(nT)$ as shown in Fig.3.

The fuzzy PD controller has two inputs, namely, average change of output $d(nT)$ and the rate of change of output $\triangle y(nT)$. The fuzzy PD controller has a single output called the incremental control output and is denoted by $\triangle u_{PD}(nT)$. The inputs to the fuzzy PD controller have to be fuzzified before being fed into the controller. The membership functions for the two inputs and the output of the controller are shown in Fig.4. These are the simplest possible functions to use for this purpose.

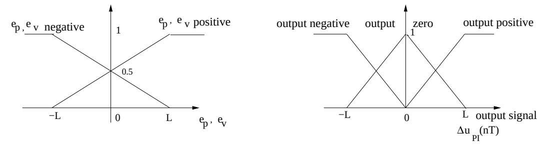

Fig. 3. Membership functions for error, rate of change of error and incremental control output

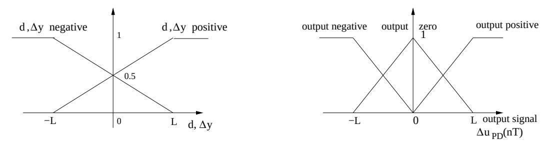

Fig. 4. Membership functions for average output, rate of change of output and incremental control output

B. Control rule base

Using the aforementioned membership functions, the following control rules are established for the fuzzy PI controller:

1) R1: IF $e_p.n$ AND $e_v.n$ THEN PI-output = o.n.
2) R2: IF $e_p.n$ AND $e_v.p$ THEN PI-output = o.z.
3) R3: IF $e_p.p$ AND $e_v.n$ THEN PI-output = o.z.
4) R4: IF $e_p.p$ AND $e_v.p$ THEN PI-output = o.p.

In these rules, $e_p = r - y$ is the error, $e_v = \dot{e_p} = 0 - \dot{y} = -\dot{y}$ is the rate of error, "PI-output" is the fuzzy PI control output $\triangle u_{PI}(nT)$, "$e_p.p$" means"error positive" and "o.p" means "output positive" etc. Also,"AND" is the Zadeh's logical "AND" [14] defined by

$$\mu_A AND \mu_B = min\{\mu_A \cdot \mu_B\}$$

for any two membership values μ_A and μ_B on the fuzzy subsets A and B respectively.
Likewise, from the membership functions of the fuzzy PD controller, the following control rules are used for the PD components:

1) R5: IF $y.p$ AND $\triangle y.p$ THEN PD-output = o.z.
2) R6: IF $y.p$ AND $\triangle y.n$ THEN PD-output = o.p.
3) R7: IF $y.n$ AND $\triangle y.p$ THEN PD-output = o.n.
4) R8: IF $y.n$ AND $\triangle y.n$ THEN PD-output = o.z.

C. Defuzzification

In the defuzzification step, for both the fuzzy PI and PD controllers, the commonly used "centre of mass" formula[11], [15] is employed to defuzzify the incremental control of the fuzzy control law (15). The "centre of mass" formula for defuzzification reads as:

$$\Delta(nT) = \frac{\Sigma membership\ value\ of\ input \times output\ corresponding\ to\ that\ membership}{\Sigma membership\ value\ of\ input}$$

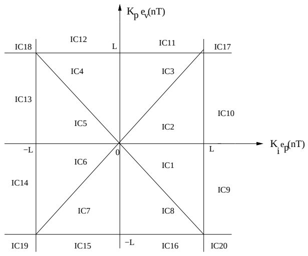

Fig. 5. Regions of the fuzzy PI controller input-combination values

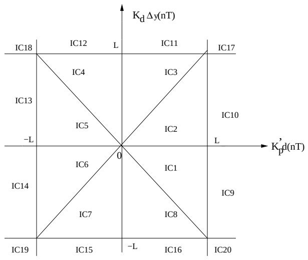

Fig. 6. Regions of the fuzzy PD controller input-combination values

The control rules for the fuzzzy PI controller (R1)-(R4), with membership functions and input-combination (IC) regions together, are used to evaluate appropriate fuzzy control laws for each region. The ranges of values of the two inputs, namely, the error and the rate of change of error, are actually decomposed into 20 adjacent IC regions. We put the membership function of the error signal (given in Fig.3) over the horizontal $K_i.e_p(nT)$-axis and put the membership function of the rate of change of the error signal over the vertical $K_p.e_v(nT)$-axis as shown in Fig.5.
Working through all regions, we obtain the following formulas for the $20 IC$ regions:

$$\begin{aligned}
\triangle u_{PI}(nT) &= \frac{L[K_i.e_p(nT)+K_p.e_v(nT)]}{2(2L-K_i.|e_p(nT)|)} \\
&\qquad (in\ IC1, IC2, IC5, IC6) \\
&= \frac{L[K_i.e_p(nT)+K_p.e_v(nT)]}{2(2L-K_p.|e_v(nT)|)} \\
&\qquad (in\ IC3, IC4, IC7, IC8) \\
&= \frac{1}{2}[K_p.e_v(nT)+L]\,(in\quad IC9, IC10) \\
&= \frac{1}{2}[K_i.e_p(nT)+L]\,(in\quad IC11, IC12) \\
&= \frac{1}{2}[K_p.e_v(nT)-L]\,(in\quad IC13, IC14) \\
&= \frac{1}{2}[K_i.e_p(nT)-L]\,(in\quad IC15, IC16) \\
&= 0\,(in\quad IC18, IC20) \\
&= L\,(in\quad IC17) \\
&= -L\,(in\quad IC19)
\end{aligned} \tag{16}$$

Similarly, defuzzification of the fuzzy PD controller follows the same procedure as described above for the PI component, with the exception that the input signals in this case are different. We put the membership function of the average change of the output signal (given in Fig.4) over the horizontal $K_p' d(nT)$-axis and the membership function of the incremental control output signal over the vertical $K_d \triangle y(nT)$-axis as shown in Fig.6. Similar to the formulas in (16), we use the values $o.p = L, o.n = -L, o.z = 0$ and apply the straight line formulas obtained from the geometry of Fig.6 given by:

$$\begin{aligned}
d.p &= \frac{K_p'.d(nT)+L}{2L}, \quad d.n = \frac{-K_p'.d(nT)+L}{2L}, \\
\triangle y.p &= \frac{K_d.\triangle y(nT)+L}{2L}, \quad \triangle y.n = \frac{-K_d.\triangle y(nT)+L}{2L}.
\end{aligned} \tag{17}$$

Hence, we obtain the following formulas for the twenty IC regions:

$$\begin{aligned}
\triangle u_{PD}(nT) &= \frac{L[K_p'.d(nT)-K_d.\triangle y(nT)]}{2(2L-K_p'.|d(nT)|)} \\
&\qquad (in\ IC1, IC2, IC5, IC6) \\
&= \frac{L[K_p'.d(nT)-K_d.\triangle y(nT)]}{2(2L-K_d.|\triangle y(nT)|)} \\
&\qquad (in\ IC3, IC4, IC7, IC8) \\
&= \frac{1}{2}[-K_d.\triangle y(nT)+L]\,(in\quad IC9, IC10) \\
&= \frac{1}{2}[K_p'.d(nT)-L]\,(in\quad IC11, IC12) \\
&= \frac{1}{2}[-K_d.\triangle y(nT)-L]\,(in\quad IC13, IC14) \\
&= \frac{1}{2}[K_p'.d(nT)+L]\,(in\quad IC15, IC16) \\
&= 0\,(in\quad IC18, IC20) \\
&= L\,(in\quad IC17) \\
&= -L\,(in\quad IC19)
\end{aligned} \tag{18}$$

IV. Computer Simulation Results

In this chapter, we present computer simulation results using the proposed fuzzy PI-PD controller. The nonlinear defuzzification algorithm was used for simulation results. In these simulations, first order and fourth order plants with time delay are used to compare the performance of our proposed fuzzy PI-PD controller with that of the conventional one for unit step input. The fuzzy PI-PD controller shows remarkable improvement in performance over conventional PI-PD controllers as regards overshoot. We have also considered the case of nonlinear processes for our simulation where both overshoot and steady state errors are seen to be minimized in the response.
Let us consider a first order plant with time delay [6] having the following transfer function

$$H(s) = \frac{4e^{-2s}}{4s-1} \tag{19}$$

The conventional PI-PD gain parameters are $K_p = 0.0637, K_i = 0.0858, K_p' = 0.602$, $K_d = 11.56$ and the sampling period T is 0.1 sec. The fuzzy PI-PD gain parameters are $K_p = 2.4, K_i = 0.2, K_p' = 3.5, K_d = 0.5, K_{uPI} = 0.005, K_{uPD} = 0.0279$ and $L = 300$.

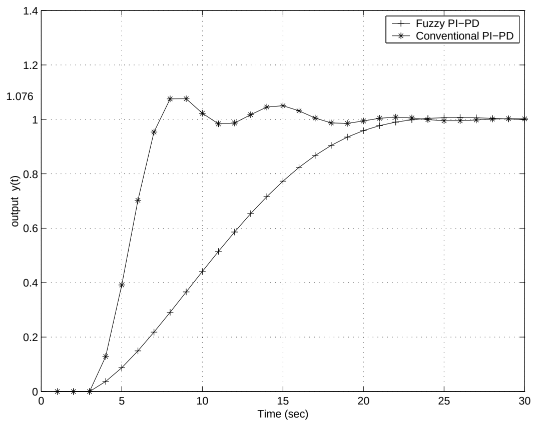

Fig. 7. Output responses of conventional and fuzzy controller for a first order process with time delay

The output response of our proposed fuzzy PI-PD controller and that of the conventional PI-PD controller are shown in Fig. 7. It is clearly seen that the fuzzy controller has zero overshoot, but the conventional controller has 7.6% overshoot. The settling time of the conventional and the fuzzy controller are 19 sec and 23 sec respectively. Settling time is more in the fuzzy controller because we considered only overshoot as the objective function in the GA.
Next we consider a higher order system with time delay. The higher order system under consideration is a fourth order plant with time delay[6] having transfer function

$$H(s) = \frac{e^{-0.2s}}{s(s+1)^3} \tag{20}$$

The conventional PI-PD gain parameters are $K_p = 0.2542$, $K_i = 0.1221$, $K_p' = 0.5123$ and $K_d = 24.7$. Sampling period T is 0.1sec. The fuzzy PI-PD gain parameters are $K_i = 1.4662$, $K_p = 7.0342$, $K_p' = 0.4897$, $K_d = 0.0127$, $K_{uPI} = 0.0142$, $K_{uPD} = 0.7035$ and $L = 1.5393$.

The output response obtained in our proposed fuzzy PI-PD controller is compared with that of a conventional PI-PD controller in Fig.8. It clearly reveals that the fuzzy PI-PD controller after optimisation via the GA, has better transient response as regards overshoot. The fuzzy controller has no overshoot. Moreover in this case, the fuzzy controller has a settling time of 17.2 sec which is pretty faster than the settling time of 20.1 sec in the case of the conventional controller.

Next, a nonlinear process[15] is considered for our simulation study. The nonlinear plant is described by

$$\dot{y}(t) = y(t) + \sqrt{y(t)} + u(t) \tag{21}$$

The fuzzy PI-PD gain parameters are $K_p = 7.5344, K_i = 1.1646, K_p' = 0.6911, K_d = 0.0246, K_{uPI} = 6.7445, K_{uPD} = 7.4524$ and $L = 146.2357$, and the input is a unit step. The sampling period T is 0.1 sec. The output response is shown in Fig.9, which clearly reveals that the fuzzy PI-PD controller tracks the set point without any oscillations. The peak overshoot is zero and the settling time is 1.96 sec.

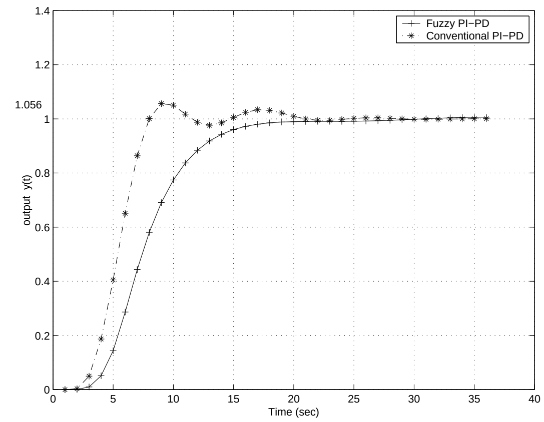

Fig. 8. Output responses of conventional and fuzzy controller for a fourth order process with time delay

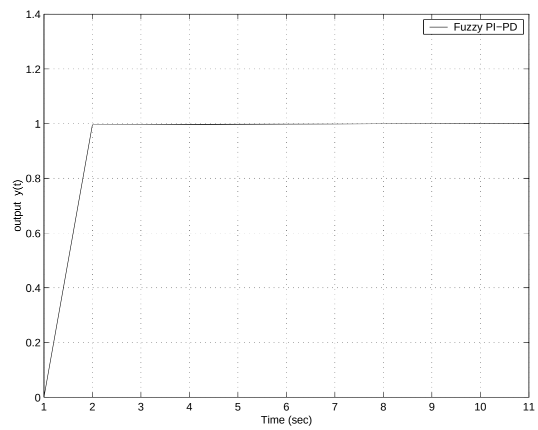

Fig. 9. Output response of fuzzy PI-PD controller for nonlinear system

The above simulation results clearly reveal that the fuzzy PI-PD controller has better transient response than the conventional PI-PD controller when the system under control is a high-order linear processes. When the controlled process is a nonlinear one, both the transient and steady state performance of the fuzzy PI-PD controller are excellent.

V. Conclusion

We have described the design principle of a fuzzy PI-PD controller and also investigated the merits of a fuzzy PI-PD controller over the conventional one by using nonlinear defuzzification algorithm. Our proposed controller is a discrete-time fuzzy version of the conventional PI-PD controller having self-tuning gain capability. The optimal fuzzy gain parameters are obtained by using Genetic Algorithm. First order and fourth order plants with time delay are used for simulation of linear systems. For these two examples, the output response of the fuzzy controlled system has no overshoot. The output response of the fuzzy controlled system has excellent transient response in the case of nonliner systems. The fuzzy “if-then” rules used in this design are generic type in the sense that they do not depend on the specific structure of the system under control.

REFERENCES

[1] B. C. Kuo, *Automatic Control Systems*, 7th ed. Prentice-Hall, 1997.

[2] J. J. D'Azzo and H. Houpis, *Linear Control Systems Analysis And Design*, 4th ed. McGraw-Hill, 1995.

[3] J. H. Park, S. W. Sung, and I. Lee, “An enhanced PID control strategy for unstable processes,” *Automatica.*, vol. 34, no. 6, pp. 751–756, 1998.

[4] C. C. Valentine and M. Chidambaram, “PID control of unstable time delay systems,” *Chem. Eng. Comm.*, no. 162, pp. 63–74, 1997.

[5] D. P. Atherton and S. Majhi, “Limitations of PID controllers,” *American control conference*, vol. 6, pp. 3913–3917, 1999.

[6] S. Majhi, “Relay feedback process identification and controller design,” Ph.D. dissertation, Univ. of Sussex, Brighton, UK, August 1999.

[7] D. Beasley, D. R. Bull, and R. R. Martin, “An overview of genetic algorithms: Part1 fundamentals,” *University of Computing*, vol. 15, no. 2, pp. 58–69, 1993.

[8] Y.P.Kuo and T.H.S.Li, “Ga-based fuzzy pi/pd controller for automotive active suspension system,” *IEEE Transaction on Industrial Electronics*, vol. 46, no. 6, pp. 1051–1066, 1999.

[9] E.N.Sanchez and V.Flores, “Real-time fuzzy pi+pd control for an underactuated robot,” in *Proc. IEEE Int.Symp.Intelligent Control*, 2002, pp. 137–141.

[10] C. T. Chen, *Analog and Digital Control System Design*. Orlando, FL: Saunders College, 1993.

[11] G. Chen and H. Ying, “On the stability of fuzzy PI control systems,” in *Proc. 3rd internat. conf. on Fuzzy Logic Applications*, Dec 1993, pp. 128–133.

[12] H. A. Malki, H. Li, and G. Chen, “New design and stability analysis of fuzzy proportional-derivative control systems,” *IEEE Transactions on Fuzzy Systems*, vol. 2, no. 4, pp. 245–253, 1994.

[13] H. Ying, W. Siler, and J. J. Buckley, “Fuzzy control theory: a nonlinear case,” *Automatica*, vol. 26, pp. 513–520, 1990.

[14] L. Zadeh, “Fuzzy sets,” *Information and Control*, vol. 8, pp. 338–352, 1965.

[15] D. Misir, H. A. Malki, and G. Chen, “Design and analysis of a fuzzy proportional integral derivative controller,” *Fuzzy Set and Systems*, vol. 79, pp. 297–314, 1996.