PID Control with Parameter Scheduling Using Fuzzy Logic

Jae Hyuck Kwak and Gi Joon Jeon
Dept. of Electronics, Kyungpook National University, Taegu, KOREA and
Eng. Research Center for Adv. Control and Instr. of SNU by KOSEF

Abstract

This paper describes new PID control methods based on the fuzzy logic. PID gains are retuned after evaluating control performances of transient responses in terms of performance features. The retuning procedure is based on fuzzy rules and reasoning accumulated from the knowledge of experts on PID gain scheduling. For the case that the retuned PID gains result in worse CLDR (characteristics of load disturbance rejection) than the initial gains, an on-line tuning scheme of the set-point weighting parameter is proposed. This is based on the fact that the set-point weighting method efficiently reduce either overshoot or undershoot without any degradation of CLDR. The set-point weighting parameter is adjusted at each sampling instant by the fuzzy rules and reasoning. As a result, better control performances were achived in comparison with the controllers tuned by the Z-N (Ziegler-Nichols) parameter tuning formula or by the fixed set-point weighting parameter.

1. Introduction

The PID controller is perhaps the most common control strategy and used in practice broadly and frequently because of its simple structure and robust performance in a wide range of operating conditions. The important part in the design of such a controller is to determine three parameters, proportional, integral and derivative gains. According to the way of tuning parameters, the PID controllers can be divided into two main classes. In the first class, the controller parameters are determined analytically and/or automatically by a certain proper design procedure and they are fixed during operation [1]. While in the second class the parameters are updated continuously under a certain degree of knowledge of the process, and thus such controllers are called adaptive PID controllers [2-4]. Until today many researchers have presented many descriptions on on-line tuning of the PID controllers in the literature, and among these were several techniques based on fuzzy rules and reasoning [3, 4]. They could control processes successfully in the aspects of reducing the overshoot and the rise time. However, there are some drawbacks in that since the parameters are varying with time, it is very difficult to analyze the stability of the closed-loop control system, and even if the asymptotic stability is assured, wild start-up and abrupt set-point change may be intolerable [4]. In practical point of view, it is common for plant operators to have their own preferences for the best type of the transient response. In such situations, one speaks of the optimal response as the one that looks the best to the operator [2]. Therefore, it can be said that plant operators want to find the controller parameters which can produce the desired output under guarantee of stability. Such parameters can be determined by applying the knowledge of experts on PID control to the actual system through very boring and laborious procedures.

In this paper, we present a parameter tuning method of PID controllers based on fuzzy rules and reasoning to determine the controller parameters by means of several performance criteria of the step response. And for the case where the PID gains which improve the transient response make the CLDR worse [5, 6], we also propose a new PID control method based on on-line tuning of the set-point weighting parameter. The parameter is properly tuned at each sampling instant by the fuzzy rules using the output error and its first difference. Simulation results show that the proposed PID control scheme can effectively improve the control performances.

2. The PID Controller

The PID controller consists of three kinds of different controllers, namely, P(proportional), I(integral) and D(derivative) controllers where each controller supports others in operation. The transfer function of the PID controller has the following form :

$$G_c(s) = K_p + \frac{K_i}{s} + K_d s \qquad (1)$$

where, K_p, K_i and K_d are the proportional, integral and derivative gains, respectively. The discrete-time equivalent representation for the PID controller used in this paper is expressed as follows :

$$u(k) = K_p(\beta r - y(k)) + K_i T \sum_{i=1}^{k} e(i) + \frac{K_d}{T} ce(k) \qquad (2)$$

where, r stands for a constant reference input, β is the set-point weighting parameter, T the sampling time, $u(k)$ the control input at discrete-time k, $e(k)$ the error between the reference value and the process output ($y(k)$), and $ce(k)$ is the difference between two consecutive errors, that is, $ce(k) = e(k) - e(k-1)$.

The Z-N tuning formula which is based on the knowledge of the ultimate gain K_u and ultimate period T_u [7] was developed by empirical simulations on many different systems. Consequently, even though the dynamic model of the process is unknown, the PID controller tuned by the Z-N tuning method can be used. Though this tuning technique does not design to any specifications, years of experience by process control engineers have indicated that it produces rather good set-point responses [8], and Hang [5] finds that the tuning method gives a set of PID control parameters that are good at load disturbance rejection. However, the method has a drawback in that it often leaves closed-loop systems poorly undamped. Therefore, in order to satisfy the desired specifications which include overshoot, undershoot, rise time and so on, the PID controller gains should be retuned by very laborious trial and error procedures from the gains computed by the Z-N tuning formula, and it is also necessary to improve the excessive overshoot and undershoot in the step response without worsening the CLDR. In the following section, parameter scheduling methods of the PID controllers based on fuzzy rules are presented.

3. Fuzzy Parameter Tuning of PID Controllers

3.1. PID gain tuning scheme

Our scheme on the fuzzy parameter tuning is started with a nominal PID controller gain set. For the unknown plant, since the heuristic Z-N tuning technique can still be used, we make use of the gains computed by the Z-N tuning formula as initial values for the tuning procedure. The approach taken here is to exploit performance features of step responses to generate controller parameters using fuzzy rules and reasoning. Thus the performance features are utilized in fuzzy rules as inputs. A typical step response and several performance features are shown in Fig. 1, and in the figure, it is divided into four areas according to the signs of the output error and its first difference.

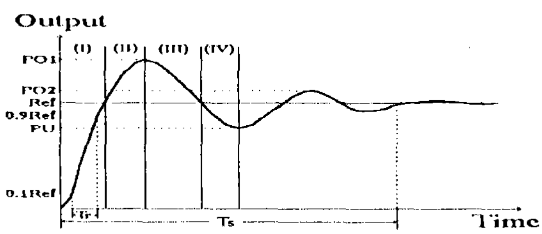

Fig 1. A typical step response

As shown in Fig. 1, in order to evaluate the degree of satisfaction for specifications, we consider the following six features which indicate the performance of transient responses : 1) percent overshoot (PO), 2) percent undershoot (PU), 3) rise time (T_r), 4) 5% settling time (T_S), 5) damping (D), 6) integral of the absolute error (IAE). The names of above features are self-explanatory except damping which is defined as

$$damping(D) \triangleq \frac{PO2 + PU}{PO1 + PU}$$

where PO1 and PO2 are, respectively, the percent overshoot of the first and second peaks. For the purpose of establishing the fuzzy rule base to determine the controller parameters which meet the desired control performance, the relationship between each controller parameter and the performance features is surveyed first, and it is also important to choose proper inputs for the fuzzy rules. By considering existing knowledge and through simulations on many plants, we can describe the ordinary effects of the change of each parameter on step responses as follows :

1) Increasing K_p decreases rise time, increases overshoot, undershoot and damping, and makes step responses a little oscillatory.
2) Increasing K_i increases overshoot and decreases rise time and undershoot especially when the ratio of overshoot to undershoot is smaller than 1.
3) Increasing K_d decreases overshoot and increases damping and rise time.

In order to express all of the above information in fuzzy rules, we employ the following inputs,

$$IN_1 = ISF_1 \cdot (\, PO1(desired) - PO1(actual)\,) \quad (3a)$$
$$IN_2 = ISF_2 \cdot (\, PO1(actual) - PU(actual)\,) \quad (3b)$$
$$IN_3 = ISF_3 \cdot (\, Tr(desired) - Tr(actual)\,) \quad (3c)$$
$$IN_4 = ISF_4 \cdot (\, D(desired) - D(actual)\,) \quad (3d)$$

where, ISF_i means the scaling factor for each input. The fuzzy rule bases are divided into three parts, and generally, since a specific demand on the overshoot is considered first, IN_1 is used as a common input in each rule base. The plant operators' various control aims can be easily expressed by suitably setting the weights for the outputs provided by each rule base and input/output scaling factors. Consequently, the controller parameters are determined by the fuzzy rules of the form :

$$\text{IF } IN_1 \text{ is } A_1{}^i \text{ and } IN_2 \text{ is } A_2{}^i \text{ THEN } \rho K_{p1} \text{ is } B_1{}^i \text{ and } \rho K_{i1} \text{ is } B_2{}^i \quad (4a)$$
$$\text{IF } IN_1 \text{ is } C_1{}^i \text{ and } IN_3 \text{ is } C_2{}^i \text{ THEN } \rho K_{p2} \text{ is } D_1{}^i \text{ and } \rho K_{i2} \text{ is } D_2{}^i \quad (4b)$$
$$\text{IF } IN_1 \text{ is } E_1{}^i \text{ and } IN_4 \text{ is } E_2{}^i \text{ THEN } \rho K_d \text{ is } F^i \quad (4c)$$
$$i = 1, 2, \cdots, n$$

where, A_1, A_2, C_1, C_2, E_1 and E_2 are input fuzzy sets, B_1, B_2, D_1, D_2 and F are output fuzzy sets, n is the number of rules of each rule base, and the outputs, ρK_p, ρK_i, and ρK_d, which are obtained by the rules and reasoning denote the incremental ratios of the corresponding gains. The MFs (membership functions) for input and output fuzzy sets are shown in Fig. 2 and Fig. 3, respectively. In these figures, μ stands for the grade of membership, and P represents positive, N negative, ZE zero, B big and S small. For instance, PB stands for Positive-Big. Note that the output linguistic variables may also be considered as fuzzy numbers which have a singleton MF. On the basis of afore-mentioned relations between the controller parameters and the performance features, the fuzzy parameter tuning rule bases are given in Tables 1, 2 and 3.

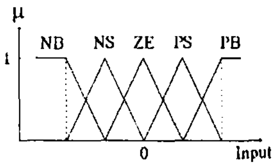

Fig. 2. MFs for inputs

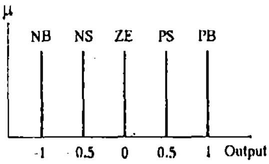

Fig. 3. MFs for outputs

The final output values, $\rho\overline{K_p}$, $\rho\overline{K_i}$, and $\rho\overline{K_d}$, obtained from the defuzzified output values (ρK_p, ρK_i, and ρK_d) denote the rates of change of the controller parameters. For example, $K_p^{n+1} = K_p^{n}\cdot(1+\rho\overline{K_p})$, where K_p^{n+1} means the adjusted proportional gain after n transients for the tuning procedure. To stop the tuning procedure, we introduce an objective function which is formulated to check the control performance after the parameters are adjusted. This function combines the pragmatic approach of the human tuner with the optimal response that the operator wants in a practical as well as numerical sense. The objective function, J, is defined as

Table 1. Rule base for adjusting K_p & K_i
(ρK_p and ρK_i are separated)

IN₂ / IN₁	N B		N S		Z E		P S		P B	
	ρK_p	ρK_i	ρK_p	ρK_i	ρK_p	ρK_i	ρK_p	ρK_i	ρK_p	ρK_i
N B	NB	PB	NB	PS	NS	ZE	NS	NS	ZE	NB
N S	NB	PB	NS	PS	NS	PS	ZE	NS	PS	NB
Z E	NS	PB	NS	PS	ZE	PS	ZE	ZE	PB	NS
P S	ZE	PB	ZE	PB	ZE	PB	PS	ZE	(ZE)	(ZE)
P B	PS	PB	ZE	PB	(ZE)	(ZE)	(ZE)	(ZE)	(ZE)	(ZE)

* ( ) means that the situation is hardly happened

Table 2. Rule base for adjusting K_p & K_i
(ρK_p and ρK_i are common)

IN₃ / IN₁	N B	N S	Z E	P S	P B
N B	ZE	NS	NB	NB	NB
N S	PS	ZE	NS	NB	NB
Z E	PB	PS	ZE	NS	NB
P S	PB	PB	PS	ZE	NS
P B	PB	PB	PB	PS	ZE

Table 3. Rule base for adjusting K_d

IN₄ / IN₁	N B	N S	Z E	P S	P B
N B	NS	PS	PS	PB	PB
N S	NS	NS	PS	PB	PB
Z E	NB	NS	ZE	PS	PB
P S	NB	NB	NS	PS	PS
P B	NB	NB	NS	NS	PS

$$J = \sum_i W_i C_i \qquad (5)$$

where C_i is the degree of satisfaction at a specific performance feature, and W_i is the weight for C_i. The weights are used to emphasize the order of priority of selected performance features, which are determined by the plant operator, and $\sum W_i = 1$. The tuning procedure is stopped when the value of the objective function becomes larger than a predefined specific value that stands for a tolerable degree of satisfaction at the overall control performance. The form of the function to compute C_i depends on the operator's preference, and we tentatively use a simple function as shown in Fig. 4. In this figure, $P_i(desired)$, $i=1,2,\cdots 6$, are the desired values of afore-mentioned six features of transient responses, for example, $P_1(desired)=10$ implies the desired value of the percent overshoot is 10%, and $a_i\cdot P_i(desired)$ means the boundary value on which the degree of satisfaction at the performance feature is 0, where a_i is a multiplicative constant not less than 1.0.

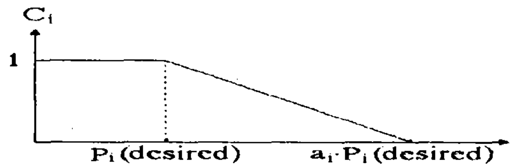

Fig. 4. The function C_i

The degrees of satisfaction, C_i, $i=1,2,\cdots 6$, are obtained from Fig. 4 as following equations and defuzzification and reasoning procedures will be presented in chapter 4.

$$C_i = \begin{cases} 0, & P_i(actual) > a_i P_i(ref) \\ 1, & P_i(actual) < P_i(ref) \\ \dfrac{P_i(actual)}{(1-a_i)P_i(ref)} - \dfrac{a_i}{(1-a_i)}, & elsewhere \end{cases} \qquad (6)$$

3.2. New approach based on on-line tuning of the set-point weighting parameter

The excessive overshoot in a step response can be effectively reduced by using the set-point weighting parameter, β in equation (2), whose value is smaller than 1. The set-point weighting method causes a little increased rise time, and no changes on the CLDR [9]. However, since it hardly affects undershoot, it can not be an effective method to improve the transient response when both the overshoot and the undershoot are excessive. In the meantime, the CLDR is mainly affected by the proportional and integral terms of PID controllers as briefly mentioned before. Here, for the case that the retuned PID gains lead to worse CLDR, we expect that the excessive overshoot and undershoot can be remarkably reduced without deteriorating the CLDR by the proper on-line tuning of the set-point weighting parameter. And this scheme based upon the fuzzy rules will be explained below.

The knowledge base of the tuning scheme for set-point weighting parameter, β, consists of collections of rules describing the roles of the parameter in control actions. These rules represent a critical starting point of the proposed method. The parameter is determined by a set of fuzzy rules of the form.

$$\text{IF } Se \text{ is } A_1^i \text{ and } Sce \text{ is } A_2^i \text{ THEN } \Delta\beta \text{ is } A_3^i,\quad i=1,2,\cdots,m \tag{7}$$

where, m is the number of rules. A_1^i and A_2^i stand for the input fuzzy sets, and A_3^i for the output fuzzy set on the corresponding supporting sets. Se and Sce are scaled output error and its first difference, and $\Delta\beta$ denotes the output of each rule used to compute the final output value at each sampling instant. The MFs of input and output fuzzy sets are shown in Fig. 5 and Fig. 6, respectively. In Fig. 6, the output variable, $\Delta\beta$, and the grade of the MFs, $\mu(\cdot)$, have the following relations.

$$\mu_{NB} = -\frac{1}{4}\ln(1+\Delta\beta) \quad \text{or} \quad \Delta\beta_{NB} = -1 + \exp(-4\mu) \tag{8a}$$

$$\mu_{NS} = -\frac{1}{4}\ln(-\Delta\beta) \quad \text{or} \quad \Delta\beta_{NS} = -\exp(-4\mu) \tag{8b}$$

$$\mu_{PS} = -\frac{1}{4}\ln(\Delta\beta) \quad \text{or} \quad \Delta\beta_{PS} = \exp(-4\mu) \tag{8c}$$

$$\mu_{PB} = -\frac{1}{4}\ln(1-\Delta\beta) \quad \text{or} \quad \Delta\beta_{PB} = 1 - \exp(-4\mu) \tag{8d}$$

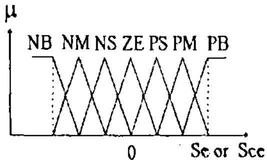

Fig. 5. MFs. for inputs

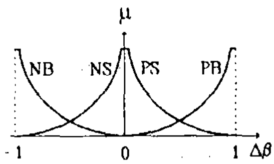

Fig. 6. MFs. for outputs

The fuzzy rules to determine proper values of the parameter at each sampling instant are extracted from the step response of the process. According to the signs of e and ce, the step response of the process can be characterized by four regions as shown in Fig. 1, and the signs of e and ce in each region are represented in Table 4.

Table 4. The signs of e and ce of each region

	(I)	(II)	(III)	(IV)
sign (e)	+	−	−	+
sign (ce)	−	−	+	+

The fuzzy rules for tuning β are established by several basic rules. First, in order to avoid an excessive overshoot, it is necessary to produce a small control signal in the latter half of the region (I) and all the region of (II). Therefore, we need to lower the value of β in such regions. Similary, in the latter part of the region (III) and throughout the region (IV), we expect a large control signal to avoid an excessive undershoot. Thus, the values of β should be large. On the other hand, to keep the rise time at least not longer than that of the simple set-point weighting method using a fixed β, the values of β should be kept a little large in the first half of the region (I). Under the above basic rules, and after many trial and errors through simulations on a variety of plants, the tuning rules for β are shown in Table 5.

Table 5. Rule for adjusting β

Se \ Sce	NB	NM	NS	ZE	PS	PM	PB
NB	NB	NB	NB	NS	PS	PS	PB
NM	NB	NB	NS	PS	PS	PB	PB
NS	NB	NS	NS	PS	PB	PB	PB
ZE	NB	NB	NS	PS	PS	PB	PB
PS	NB	NB	NB	NS	PS	PS	PB
PM	NB	NB	NS	NS	PS	PB	PB
PB	NB	NS	NS	PS	PB	PB	PB

4. Reasoning and Defuzzification

For simplicity, the reasoning and defuzzufication procedures for adjusting the parameters, K_p, K_i, K_d and β, are explained together in this chapter. In order to reduce the amount of computation, we use a logic based indirect reasoning method. The truth values of the i-th rules in equation (4) and (7), r_j^i and v^i, respectively, are obtained by the triangular norm (T norm) operations between the MF values of two inputs in each rule. That is,

$$r_j^i = \mu_{ILV_1^i}(IN_1)\cdot\mu_{ILV_2^i}(IN_j), \quad j=2,3,4 \tag{9a}$$

$$v_i = \mu_{A_1^i}(Se) \wedge \mu_{A_2^i}(Sce), \quad i=1,2,\cdots,n \tag{9b}$$

where, ILV^i is the specified input linguistic variables, and $\mu_{ILV^i}(\cdot)$ and $\mu_{A^i}(\cdot)$ are the MF values of the actual inputs for the input fuzzy sets of

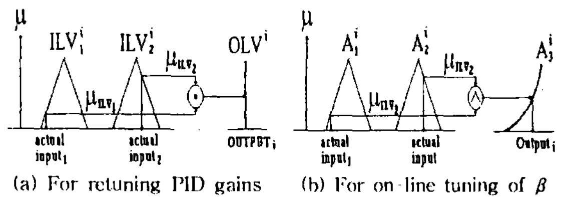

Fig. 7. Implication process for each rule

the i-th rule as shown in (4) and (7), respectively. The operations, '$\cdot$' and '$\wedge$' stand for 'algebraic product' and 'minimum' between two MF values, respectively. The implication process for each fuzzy rule is shown in Figs. 7(a) and 7(b). In these figures, $output_i$ is the value of one of the actual outputs (ρK_p, ρK_i, ρK_d and $\Delta\beta$) of the i-th rule corresponding to the truth value r_j^i or v_i. Then we express the truth values and outputs of the rules in matices as follows :

$$R = \begin{bmatrix} r_2^1 & r_2^2 & \cdot & \cdot & r_2^n \\ r_3^1 & r_3^2 & \cdot & \cdot & r_3^n \\ r_4^1 & r_4^2 & \cdot & \cdot & r_4^n \end{bmatrix},$$

$$\rho P = \begin{bmatrix} \rho K_{p1}^1 & \rho K_{p1}^2 & \cdot & \cdot & \rho K_{p1}^n \\ \rho K_{p2}^1 & \rho K_{p2}^2 & \cdot & \cdot & \rho K_{p2}^n \end{bmatrix},$$

$$\rho I = \begin{bmatrix} \rho K_{i1}^1 & \rho K_{i1}^2 & \cdot & \cdot & \rho K_{i1}^n \\ \rho K_{i2}^1 & \rho K_{i2}^2 & \cdot & \cdot & \rho K_{i2}^n \end{bmatrix},$$

$$\rho D = [\ \rho K_d^1 \ \ \rho K_d^2 \ \ \cdot \ \ \cdot \ \ \rho K_d^n\ ]$$

$$V = [\ v^1 \ \ v^2 \ \ \cdot \ \ \cdot \ \ v^m\ ],$$

$$\Delta B = [\ \Delta\beta^1 \ \ \Delta\beta^2 \ \ \cdot \ \ \cdot \ \ \Delta\beta^m\ ]$$

For the present work, we note that there are at most four rules whose truth values are not zero when the MFs of inputs in Fig. 2 or Fig. 5 are used. By using elements of the above matrices, defuzzification yields the value of each output and the final outputs ($\rho\overline{K_p}$, $\rho\overline{K_i}$, ρK_d and $\Delta\beta$) as follows.

$$\rho K_{p1} = \frac{\sum_{i=1}^{n} r_2^{\,i}\cdot\rho K_{p1}^{\,i}}{\sum_{i=1}^{n} r_2^{\,i}},\quad \rho K_{p2} = \frac{\sum_{i=1}^{n} r_3^{\,i}\cdot\rho K_{p2}^{\,i}}{\sum_{i=1}^{n} r_3^{\,i}} \qquad (10a)$$

$$\rho K_{i1} = \frac{\sum_{i=1}^{n} r_2^{\,i}\cdot\rho K_{i1}^{\,i}}{\sum_{i=1}^{n} r_2^{\,i}},\quad \rho K_{i2} = \frac{\sum_{i=1}^{n} r_3^{\,i}\cdot\rho K_{i2}^{\,i}}{\sum_{i=1}^{n} r_3^{\,i}} \qquad (10b)$$

$$\rho K_d = \frac{\sum_{i=1}^{n} r_4^{\,i}\cdot\rho K_d^{\,i}}{\sum_{i=1}^{n} r_4^{\,i}} \qquad (10c)$$

$$\Delta\beta = \frac{\sum_{i=1}^{m} v^i\cdot\Delta\beta^i}{\sum_{i=1}^{n} v^i} \qquad (11)$$

$$\rho\overline{K_p} = OSF_1\cdot(q_1\cdot\rho K_{p1} + q_2\cdot\rho K_{p2}) \qquad (12a)$$

$$\rho\overline{K_i} = OSF_2\cdot(z_1\cdot\rho K_{i1} + z_2\cdot\rho K_{i2}) \qquad (12b)$$

$$\rho\overline{K_d} = OSF_3\cdot\rho K_d \qquad (12c)$$

$$\overline{\Delta\beta} = OSF\cdot\Delta\beta \qquad (13)$$

where, OSF_i are output scaling factors, and q_1, q_2, z_1 and z_2 are output weights used to express the priority order ($q_1+q_2=z_1+z_2=1$). Once the final output values are obtained, the adjusted parameters are calculated from the following equations.

$$K_p^{\,n+1} = K_p^{\,n}\cdot(1+\rho\overline{K_p}) \qquad (14a)$$

$$K_i^{\,n+1} = K_i^{\,n}\cdot(1+\rho\overline{K_i}) \qquad (14b)$$

$$K_d^{\,n+1} = K_d^{\,n}\cdot(1+\rho\overline{K_d}) \qquad (14c)$$

$$\beta(k+1) = \beta_0 + \overline{\Delta\beta} \qquad (15)$$

In equation (15), β_0 represents a standard value that is the center of the possible range of β, in other words, the value of β is in the range [$\beta_0 - OSF$, $\beta_0 + OSF$] .

5. Simulation Results

To investigate the adequacy of the derived rules and the effects of the proposed tuning methods, two plants with the following transfer functions are considered.

$$Plant1 : G_1(s) = \frac{27}{(s+1)(s+3)^3}$$

$$Plant2 : G_2(s) = \frac{e^{-2.5s}}{(s+1)^2}$$

Unit step responses by the proposed parameter tuning method after enough transients to meet the required specifications are plotted in Fig. 8 and Fig. 9, respectively. The results obtained by using the Z-N tuned PID controllers are also presented in Figs. 8 and 9 for comparison, and Table 6 shows the representative simulation results. Simulation results show that all the actual values of performance features except T_r are very close approximation to the desired specifications, and the PID controller whose parameters are determined by the proposed tuning method can achieve much better control performance than the Z-N tuned controller does, which is confirmed by comparing the values of performance features.

$G_1(s)$ is a deadtime-free system, while the output of $G_2(s)$ is delayed for 2.5 seconds. Consequently, the value of K_i (and K_p) should be decreased to improve the transient response of the plant like $G_1(s)$, thus the CLDR get worse as shown in Fig. 10. For the plant 2, however, the controller retuned by the proposed tuning method results in better CLDR than those obtained by the Z-N tuned controller due to the increased integral gain, as shown in Fig. 11. Thus, in order to improve transient responses and prevent the CLDR from getting worse, we apply the PID control scheme based on on-line tuning of the set-point weighting parameter to the plant 1. The step response of the plant with load desturbance is represented in Fig. 12, and in this figure, the response obtained by using a fixed set-point weighting parameter is plotted together to compare their control performances. Note that the CLDR obtained by using the method (II) are the same as the characteristics of the Z-N tuned PID controller. The parameters used in each simulation are shown in Fig. 12. The simulation result indicates that the proposed control method (I) is more efficient than the method (II) which uses a fixed set-point weighting parameter, in the aspects of both the characteristics of transient response and load disturbance rejection.

Table 6. Numerical simulation result

	$G_1(s)$			$G_2(s)$		
	desired value	before retuning	after retuning	desired value	before retuning	after retuning
K_p		2.721	2.727		0.961	0.765
K_i		1.944	1.255		0.237	0.257
K_d		0.914	1.325		0.933	0.783
PO1(%)	15	44.82	15.21	10	14.67	10.05
PU(%)	5	10.86	2.87	5	20.25	3.93
D	0.5	0.32	0.498	0.5	0.74	0.49
T_r(sec)	0.7	0.7	0.8	2.0	2.0	2.3
T_5(sec)	4.0	5.9	2.9	8.0	11.8	8.3
IAE	14	17.58	13.57	45	50.78	45.37
J	0.95	0.44	0.958	0.9	0.52	0.904

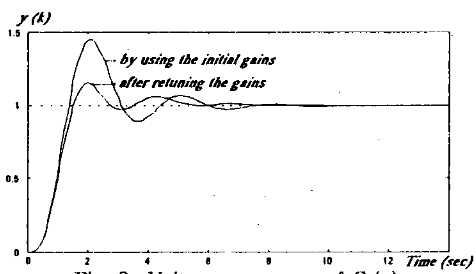

Fig. 8 Unit step response of $G_1(s)$

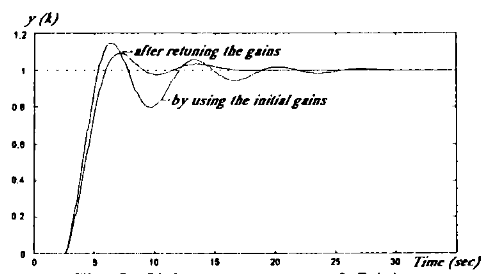

Fig. 9 Unit step response of $G_2(s)$

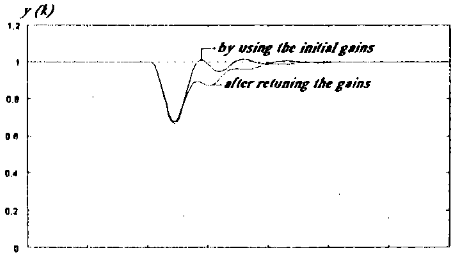

Fig. 10 Load disturbance rejection of $G_1(s)$

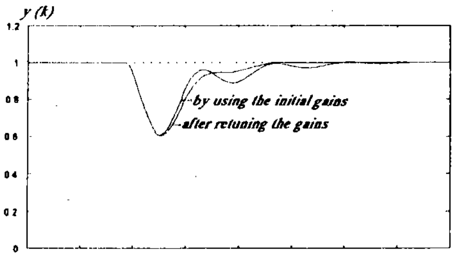

Fig. 11 Load disturbance rejection of of $G_2(s)$

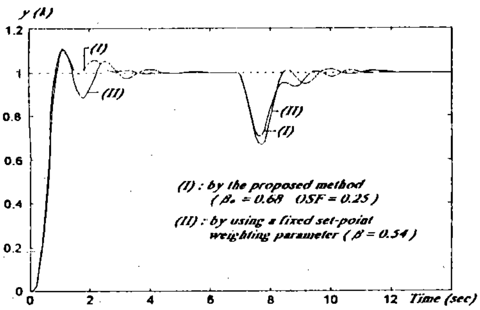

Fig. 12 Unit step responses with load disturbance rejection of $G_1(s)$

6. Conclusion

We proposed PID control schemes of retuning the PID gains and on-line tuning of the set-point weighting parameter to improve the overall control performances including CLDR. The basic objective of the proposed PID gain tuning is to retune the initial gains to improve the characteristics of transient responses. These retuned PID gains may have good or bad influences upon the CLDR mainly according to the amount of changes of the proportional and the integral gains. In this paper, we used the gains determined by the Z-N tuning formula as initial values. When the Z-N tuned PID controller is used, the performance of the controller can be predicted from the dimensionless parameter called 'normalized process gain' K, which is defined as the product of the ultimate gain K_u and the process steady state gain K_{ss} [10]. By considering the anticipated form of transient responses, we can predict to some degree of precision how much the value of each gain is increased or decreased by the retuning procedure and what influences the retuned gains have on the CLDR. Therefore, the on-line tuning of the set-point weighting parameter can be regarded as an alternative control method when the retuned PID gains which improve the transient responses make the CLDR worse. Simulations on a variety of plants have showed that the processes could be satisfactorily controlled by using the proposed parameter tuning algorithms.

7. References

[1] Aström, K.J., and Hägglund, T.: 'Automatic tuning of PID controllers', (Istrument Society of America, 1988)
[2] Litt, J.: 'An expert system to perform on-line controller tuning', *IEEE Contr. Syst. Mag.*, Apr. 1991, pp. 18-23.
[3] Tzafestas, S., and Papanikolopoulos, N.P.: 'Incremental fuzzy expert PID control', *IEEE Trans. Industrial Electronics*, Oct. 1990, Vol. 37, pp. 365-371.
[4] Zhao, Z.Y., Tomizuka, M., and Isaka, S.: 'Fuzzy gain scheduling of PID controllers', *IEEE Trans. Syst., Man, Cybern.*, 1993, Vol. 23, pp. 1392-1398.
[5] Hang, C.C.: 'The choice of controller zeros', *IEEE Control Systems*, 1989, Vol. 9, pp. 72-75.
[6] Takahashi, Y., Rabins, M.J., and Auslander, D.M.: 'Control and dynamic systems', (Addison-Wesley, 1970)
[7] Ziegler, J.G., and Nichols, N.B.: 'Optimum settings for automatic controllers', *Trans. ASME*, 1942, Vol. 64, pp. 759-768.
[8] Shahian, B., and Hassul, M.: 'Control system design using matlab', (Prentice Hall, 1993)
[9] Hang, C.C., Aström, K.J., and Ho, W.K.: 'Refinements of the Ziegler-Nichols tuning formular', *IEE Proc. D*, Mar. 1991, Vol. 138, pp. 112-11nn8.
[10] Aström, K.J., Hang, C.C., Persson, P., and Ho, W.K.: 'Towards intelligent PID control', *Automatica*, 1992, Vol. 28, pp. 1-9.