ISA Transactions 41 (2002) 473–484

ISA TRANSACTIONS®

New tuning method for PID controller

Jing-Chung Shen*

Department of Automation Engineering, National Huwei Institute of Technology, Huwei, 632, Yunlin, Taiwan

(Received 20 March 2001; accepted 9 July 2001)

Abstract

In this paper, a tuning method for proportional-integral-derivative (PID) controller and the performance assessment formulas for this method are proposed. This tuning method is based on a genetic algorithm based PID controller design method. For deriving the tuning formula, the genetic algorithm based design method is applied to design PID controllers for a variety of processes. The relationship between the controller parameters and the parameters that characterize the process dynamics are determined and the tuning formula is then derived. Using simulation studies, the rules for assessing the performance of a PID controller tuned by the proposed method are also given. This makes it possible to incorporate the capability to determine if the PID controller is well tuned or not into an autotuner. An autotuner based on this new tuning method and the corresponding performance assessment rules is also established. Simulations and real-time experimental results are given to demonstrate the effectiveness and usefulness of these formulas. © 2002 ISA—The Instrumentation, Systems, and Automation Society.

Keywords: PID controller; Controller tuning; Genetic algorithm; Performance assessment

1. Introduction

The proportional-integral-derivative (PID) controller is the most common control algorithm. Due to their simplicity and robustness, PID controllers are widely used. There are many tuning formulas for PID and PI control for processes with monotonic transient responses [1–15]. Recently, Astrom and Hagglund [2] utilized the dominant pole assignment method to derive a new tuning method for PID and PI controllers. They applied the dominant pole assignment method to a test batch of plants and then found simple formulas that described the correlation between the process characteristics and the controller parameters. In the dominant pole assignment method, the dominant poles of the system are assigned such that the integration of the error (IE) caused by a step load disturbance is minimized subject to the constraint on maximum sensitivity. A set-point weighting is then selected to improve the set-point response of the system. The set-point response of the system is not optimized. Moreover, when the system is not well damped, the IE criteria are not a good measure of the load disturbance rejection.

It is known that the genetic algorithm (GA) is a powerful technique for solving nonlinear optimization problems [16,17]. GA's are parallel, global search techniques that emulate natural genetic operators. Because a GA simultaneously evaluates many points in the parameter space, it is more likely to converge toward the global solution. In this study, a genetic algorithm was utilized to optimize the set-point and load disturbance responses for a batch of test processes controlled by PID controllers. The equations that describe the relations between the process characteristics and the controller parameters are found and used to tune the PID controllers. Simulations from various ex-

**E-mail address*: jcshen@sparc.nhit.edu.tw

amples show that this new formula gives more satisfactory results than the formula proposed by Astrom and Hagglund.

When automating controller tuning, it is attractive to incorporate the capability to judge if the PID controller is properly tuned into the autotuner. The rules for assessing the performance of a PID controller tuned using this new method are also derived. In order to show the effectiveness of this new tuning formula and the usefulness of the performance assessment rules, an autotuner is established and real-time experimental results are given.

This paper is organized as follows: In Sec. 2, the design method based on genetic algorithm is described. Section 3 shows how to derive the tuning formulas. The way to derive the performance assessment rules is given in Sec. 4. Simulation examples and experimental results are given in Sec. 5. Conclusions are given in Sec. 6.

2. Genetic algorithm based design method

The new tuning formula proposed in this paper is based on a genetic algorithm. Thus the genetic algorithm based PID controller design method is described in this section.

Let the PID controller be implemented as follows:

$$u=K\left((by_r-y)+\frac{1}{T_i}\int e\,dt+T_d\frac{de}{dt}\right),\quad e=y_r-y, \tag{1}$$

where u, y_r, y, K, b, T_i, T_d, and e are the controller output, set point, process output, controller gain, set-point weighting, integral time, derivative time, and error, respectively. Notice that the controller parameters K, b, T_i, and T_d must be positive constants. Denote the loop transfer function of the closed-loop system as $G_l(s)$ and define M_s as

$$M_s=\max_{\omega}\left|\frac{1}{1+G_l(j\omega)}\right|. \tag{2}$$

Notice that M_s is the inverse of the shortest distance from the Nyquist curve of the loop transfer function to the critical point -1 and is a measure of stability robustness. Typical value of M_s is in the range from 1.4 to 2.0 and the standard value is 2.0 [2].

Let e_s denote the error caused by a unit step set-point change and e_d denote the error caused by a unit step disturbance at the process input, respectively. Define the integrated absolute errors for set-point response and load disturbance response as

$$J_s(K,T_i,T_d,b)=\int_0^{\infty}|e_s(t)|dt,$$

and

$$J_d(K,T_i,T_d,b)=\int_0^{\infty}|e_d(t)|dt,$$

respectively.

In this study, the PID controller parameters K, T_i, T_d, and b were chosen such that the performance index

$$J(K,T_i,T_d,b)=J_s(K,T_i,T_d,b)+J_d(K,T_i,T_d,b) \tag{3}$$

is minimized under the constraint

$$K>0,\ \ T_i>0,\ \ T_d>0,\ \ b>0,\ \text{ and }\ M_s\leqslant m, \tag{4}$$

where constant $m>1$ represents the minimal requirement of stability robustness. Clearly, this is a constrained nonlinear optimization problem.

GA's are stochastic optimization algorithms that were originally motivated by the mechanisms of natural selection and evolutionary genetics, and have found extensive applications in solving global optimization problems [16,17]. A simple GA is an iterative procedure that maintains a constant size population of individuals. Each individual represents a potential solution to the problem. Each individual is evaluated to give some measure of its fitness. Some individuals undergo stochastic transformations by means of genetic operations to form new individuals. There are two types of transformation: *crossover*, which creates new individuals by combining parts from two individuals, and *mutation*, which creates new individual by making changes in a single individual. New individuals, called *offspring*, are then evaluated. Selecting the best-fit individuals from the parent and offspring populations then forms a new population. After several generations, the algorithm converges to the best individual, which hopefully represents an optimal or suboptimal solution to the problem.

GA's are parallel, global search techniques. Because a GA simultaneously evaluates many points in the parameter space, it is more likely to converge toward the global solution. It does not need to assume that the search space is differentiable or continuous. A GA is suitable for solving the PID controller design problem described above due to the following reasons. First, the search space is large and there exist many locally optimal points. Second, unlike the conventional gradient-based algorithms, the GA requires no gradient calculations. Third, the likely optimal solutions are less likely to be destroyed under a genetic operator, thereby often leading to faster convergence.

The details for employing the GA to the PID controller design problem will be described in the following.

Encoding and initialization of population

In this study, real-number encoding [17] is used; that is, the PID controller parameters K, T_i, T_d, and b are coded to (K,T_i,T_d,b)—called a chromosome.

In most tuning formulas, the values for the PID controller parameters are in the ranges $0<K<K_u$, $0<T_i<T_u$, $0<T_d<T_u$, and $0<b<1$ [2,7,14], where K_u and T_u are the ultimate gain and ultimate period of the process, respectively. Therefore the initial population generation is randomly chosen within these ranges.

Fitness and cost function

In this study, the cost function is $J(K,T_i,T_d,b)$. Our objective is to search (K,T_i,T_d,b) such that $J(K,T_i,T_d,b)$ is minimized under the constraints in Eq. (4). Thus an individual that has a lower $J(K,T_i,T_d,b)$ should be assigned a larger fitness value. The GA then tries to generate better offspring to improve the fitness. Therefore a better PID controller could be obtained by better fitness in GA's. Intuitively, we have

$$F(K,T_i,T_d,b) \propto \frac{1}{J(K,T_i,T_d,b)},$$

where $F(K,T_i,T_d,b)$ is the fitness of an individual with chromosome (K,T_i,T_d,b). In this study, $F(K,T_i,T_d,b)$ is defined as

$$F(K,T_i,T_d,b) = \frac{1}{J(K,T_i,T_d,b)} \tag{5}$$

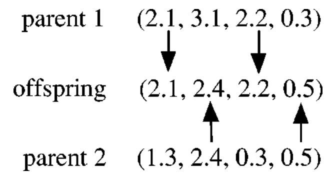

Fig. 1. Crossover operation.

and $F(K,T_i,T_d,b)=0$ if any of the following conditions are true:

(a) A system with PID parameters K, T_i, T_d, and b is unstable.

(b) Any of the parameters K, T_i, T_d, and b are negative.

(c) $M_s>m$.

The crossover and mutation operations are described in the following. When performing crossover and mutation, an individual that belongs to the parent population must be selected for operation. In this study, the roulette wheel selection method [16,17] was used. In roulette wheel selection, the jth individual of the parent population with fitness value F_j is given a proportional probability P_j of being selected, according to the distribution

$$P_j = \frac{F_j}{\sum_{i=1}^{R} F_i}, \tag{6}$$

where R is the population size. It is shown that the higher the fitness value, the higher the probability of being selected. Hence the individual with high fitness will have high probability of being chosen for crossover and mutation.

Crossover

Crossover exchanges the information of two chromosomes and provides a mechanism to mix and match the desired qualities through a random process. The crossover operation in this study proceeded in three steps. First, two chromosomes that belong to the parent population are chosen by roulette wheel selection. Second, a component is randomly taken from one chromosome to form the corresponding component in the offspring. Third, the second step is repeated until the offspring's components are completed. The crossover operation is shown in Fig. 1.

Mutation

Mutation is a way to add new genetic material to the population. This is done to help the GA avoid getting stuck at a local optimal. For a roulette wheel selected parent (K, T_i, T_d, b), if K is selected for mutation, the resulting offspring is $(\tilde{K}, T_i, T_d, b)$, where $\tilde{K}$ is randomly selected from the following two possible choices:

$$\tilde{K} = K + r\lambda\left(1 - \frac{i}{I}\right), \quad \tilde{K} = K - r\lambda\left(1 - \frac{i}{I}\right),$$

where r is an uniformly distributed random number from (0,1), i is the generation number, I is the maximum generation number, and $\lambda = K_u$ ($\lambda = T_u$ if T_i or T_d is selected for mutation and $\lambda = 1$ if b is selected for mutation).

For convenience, the genetic algorithm based (GAB) controller design algorithm is summarized as follows:

(*Step* 1) Compute the ultimate gain and ultimate period of a given process.

(*Step* 2) Generate a population of chromosomes.

(*Step* 3) Simulate the closed-loop systems with PID controllers that correspond to chromosomes and calculate the fitness value for each chromosome.

(*Step* 4) Create the offspring via the genetic operators, crossover, and mutation.

(*Step* 5) Calculate the fitness value for each offspring.

(*Step* 6) Selecting the best-fit chromosomes from the parent and offspring populations to form a new population.

(*Step* 7) If the allowable generation is attained, then stop the search; else go to *Step* 4.

3. Derivation of the new tuning formulas

The PID controller design method described in last section is a time consuming procedure. If we can find simple formulas that describe the relations between the parameters that characterize the process dynamics and the parameters of the PID controller obtained using the GAB design method, the user can obtain proper PID controller parameters easily and need not run the entire design procedure.

In this section, the tuning formulas based on the GAB design method will be presented. For deriving the new tuning formulas, a test batch of processes is chosen to represent the processes with monotonic transient responses. The GAB controller design method is then applied to design the PID controllers for the processes in this test batch. We derived the tuning formulas by finding simple formulas that described the relations between the parameters that characterize the process dynamics and the controller parameters.

In this study, the following systems chosen by Astrom and Hagglund [2] were used as the test batch:

$$G_1(s) = \frac{e^{-s}}{(1+sT)^2},$$

$$T = 0.1, 0.3, 0.5, 0.7, 0.9, 2, 4, 6, 8, 10,$$

$$G_2(s) = \frac{1}{(s+1)^n}, \qquad n = 3, 4, 8,$$

$$G_3(s) = \frac{1}{(1+s)(1+\alpha s)(1+\alpha^2 s)(1+\alpha^3 s)},$$

$$\alpha = 0.2, 0.5, 0.7,$$

$$G_4(s) = \frac{1-\alpha s}{(s+1)^3},$$

$$\alpha = 0.1, 0.2, 0.5, 1.2. \tag{7}$$

The GAB design method with $m = 2.0$ is applied to design the PID controller for each process in the test batch. In the design, the population size R and the maximum allowable generation number I were chosen as 500 and 50, respectively. Once the PID controller is designed, the controller parameters obtained can be correlated with parameters that characterize the process dynamics.

Generally, the tuning rules of PID controller are based on the step-response or frequency-domain method. When the step-response method is used, the transfer function of the process is approximated by

$$G_m(s) = \frac{K_s e^{-sL}}{(1+sT)}. \tag{8}$$

The static-gain K_s, apparent dead-time L, and apparent time constant T can be determined by a step response experiment. In this study, L was determined as the intersection of the tangent with the steepest slope with the time axis. Parameter T was determined as the time when the step response reaches 63% of its final value minus the apparent dead-time L. When the frequency-domain method is used, the process is characterized by the ultimate gain K_u, the ultimate period T_u, and the gain ratio $\delta = 1/K_s K_u$ [2,7,14]. In general, the ultimate gain K_u and the ultimate period T_u are estimated by relay experiment [1,2].

3.1. Tuning formula for step-response method

The tuning formula for the step-response method is derived next. For convenience, we utilized dimension free parameters

$$\alpha = K_s \frac{L}{T} \quad \text{and} \quad \tau = \frac{L}{L+T} \tag{9}$$

to characterize the process under control, and normalized the controller parameters as αK, T_i/L, T_d/L, and b. This is the same normalization used in the Astrom-Hagglund and Ziegler-Nichols rules. For deriving the tuning rules, the relations between the normalized controller parameters and the parameters that characterize the process dynamics are determined. The normalized controller parameters were plotted as a function of τ. We then investigated whether the normalized controller parameter αK, for example, can be expressed as

$$\alpha K = f(\tau)$$

with analogous expressions for the other parameters. It was found that the curves could be reasonably well approximated using functions of the form

$$f(\tau) = \exp(a_0 + a_1 \tau + a_2 \tau^2). \tag{10}$$

Fig. 2 shows the normalized controller parameters as functions of τ. The curves drawn correspond to the results obtained by least-square curve fitting. For comparison, the tunings obtained by Astrom-Hagglund rules for $M_s = 2.0$ are shown by the dashed lines in the figure. From the figure, it can be found that the normalized gain and normalized integral time are similar to the Astrom-Hagglund rules when $\tau > 0.3$. The set-point weighting factor of the new tuning rules is larger than Astrom-Hagglund rules. It can be expected that the new tuning rules will have a faster set-point response than the Astrom-Hagglund rules. The reason is that when deriving the new tuning rules, the controller parameters (include set-point weighting) are chosen by optimizing both the set point and load disturbance responses. The corresponding function coefficients of the form (10) are given in Table 1.

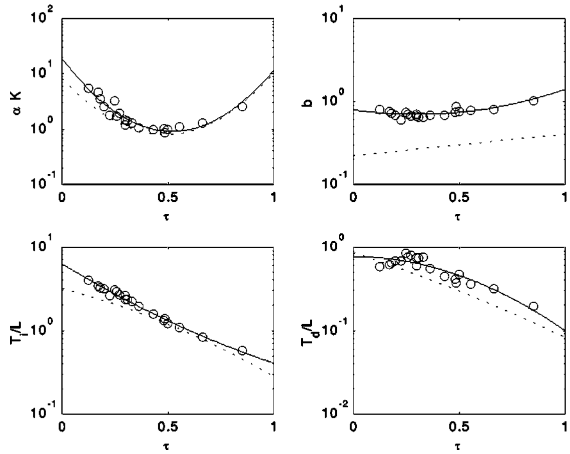

Fig. 2. Tuning diagrams for PID control. Controller parameters are obtained by genetic algorithm based design method with $m = 2.0$. The dashed lines correspond to Astrom-Hagglund's tuning rules for $M_s = 2.0$.

3.2. Tuning formula for frequency-response method

In the frequency-response method, the tuning formulas are derived in the same way as the step-response method. The parameters K_u, T_u, and δ are used to characterize the process under control. The controller parameters are normalized as K/K_u, T_i/T_u, T_d/T_u, and b, respectively. The relations between the parameters that characterize the processes and the normalized controller parameters are then determined. Fig. 3 depicts the results where the normalized controller parameters

Table 1
Coefficients of the tuning rules obtained by step-respond method.

	a_0	a_1	a_2
αK	2.94	−11.63	11.15
T_i/L	1.88	−3.63	0.86
T_d/L	−0.25	−0.06	−1.99
b	−0.22	−0.90	1.45

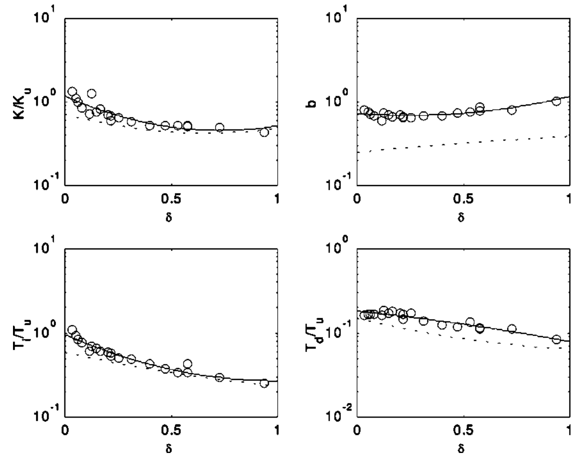

Fig. 3. Tuning diagrams for PID control based on T_u, K_u, and δ. The dashed lines correspond to the Astrom-Hagglund's tuning formulas.

are given as functions of δ. The corresponding coefficients of the tuning formulas are given in Table 2.

4. Performance assessment formulas

In this section, deriving the performance assessment formulas will be described. Following the studies in Ref. [18], two parameters the normalized rise time and the normalized peak load error are introduced to characterize the closed-loop response. The formulas described in the last section were applied to tune the PID controller for the batch of test processes and the resulting closed-loop systems were simulated. Simple formulas that describe the relations between the parameters that characterize the open-loop process dynamics and the parameters that characterize the closed-loop response were then determined. Once these formulas are found, they can be used to assess the control performance of the PID controller that is tuned by the formulas obtained in last section.

Table 2
Tuning formulas for PID control based on the frequency-response method. The table gives parameters of functions of the form $f(\delta)=\exp(a_0+a_1\delta+a_2\delta^2)$ for the normalized controller parameters.

	a_0	a_1	a_2
K/K_u	0.17	−2.62	1.79
T_i/T_u	−0.02	−2.62	1.34
T_d/T_u	−1.70	−0.59	−0.25
b	−0.30	−0.48	0.93

The closed-loop rise time, t_r, is a measure of the response speed of the closed-loop system. To obtain a dimension-free parameter, t_r will be normalized by the apparent time constant T for the step-response method and normalized by the ultimate period T_u for the frequency-response method respectively. The normalized rise time is defined as t_r/T for the step-response method and t_r/T_u for the frequency response method.

The response to the step load disturbance is an important factor when evaluating control systems. Consider the closed-loop system and assume that the disturbance enters at the process input. Since there exists an integral action in a PID controller, the static error due to a step load disturbance is zero. Therefore a meaningful measure is the maximum error. Following the studies in Ref. [18], the normalized maximum load error is defined as

$$l_n=\frac{l_{\max}}{l_oK_s}, \tag{11}$$

where l_o is the amplitude of the step load disturbance and $l_{\max}$ the maximum error due to the step load disturbance.

The performance assessment formulas were derived in the same way as the tuning formulas. The normalized rise time and the normalized maximum load error are plotted as functions of τ and δ. Fig. 4 shows the result of plotting. The curves drawn in Fig. 4 correspond to the results obtained by least-square curve fitting. The functions obtained by curve fitting are

$$t_r/T=\exp(-2.32+6.61\tau-3.23\tau^2), \tag{12}$$

$$t_r/T_u=\exp(-1.36+1.19\delta-1.38\delta^2), \tag{13}$$

$$l_n=\exp(-6.51+27.23\tau-39.67\tau^2+19.61\tau^3), \tag{14}$$

and

$$l_n=\exp(-3.76+16.0\delta-24.9\delta^2+13.0\delta^3). \tag{15}$$

Formulas (12)–(15) can be used to evaluate the performance of closed-loop systems under closed-loop operation. Consider the formulas for normalized rise time. The rise time of a closed-loop system can be measured when the set point is

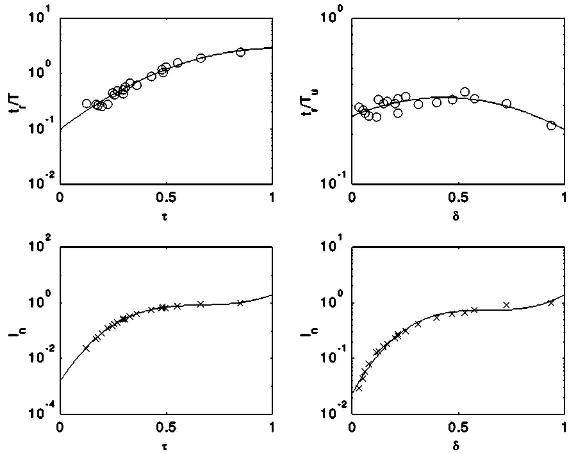

Fig. 4. Normalized rise time and normalized maximum load error of the closed-loop systems. Data are obtained by applying the tuning formulas for the step-response method and the frequency-response method to the processes in the test batch.

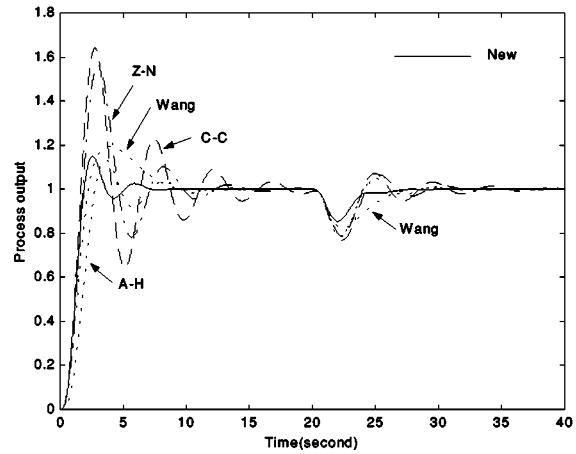

Fig. 5. Set-point response and load disturbance response of the process $1/(s+1)^3$ controlled by a PID controller tuned by the proposed method, Astrom-Hagglund's method, Wang's method, Ziegler-Nichols method, and Cohen-Coon's method.

changed. If the actual rise time is significantly different from the predicted value, for instance 30%, it indicates that the PID controller is poorly tuned. Similarly, the formulas for normalized maximum load error are used to determine poor controller tuning by introducing a small disturbance at the controller output and then measuring the maximum error. A maximum error that is significantly larger than that predicted by the formula indicates that the PID controller is poorly tuned.

5. Simulation examples and real-time experiment

In order to demonstrate the use of the tuning formulas and performance assessment formulas, we applied these formulas to a few processes. Comparisons will be made with Cohen-Coon's method [5], Ziegler-Nichols method [14], Wang's method [15], and Astrom-Hagglund's method [2].

Example 1: Consider the following process:

$$\frac{1}{(s+1)^3}.$$

It has apparent dead time $L=0.81$, apparent time constant $T=2.44$, static gain $K_s=1$, and $\tau=0.25$. Applying these values to the tuning formulas for step response, the controller parameters can be obtained as $K=6.28$, $T_i=2.27$, $T_d=0.57$, and $b=0.70$. Astrom-Hagglund's rules give $K=4.25$, $T_i=1.68$, $T_d=0.42$, and $b=0.26$. Ziegler-Nichols method gives $K=3.61$, $T_i=1.62$, and $T_d=0.41$. Cohen-Coon's method gives $K=4.32$, $T_i=1.79$, and $T_d=0.28$. Wang's method gives $K=3.76$, $T_i=2.23$, and $T_d=0.99$. Fig. 5 shows the set-point response and load disturbance response of the closed-loop system. The results show that the new tuning rules can provide shorter settling time and better load disturbance response than other methods.

The rise time and the maximum load disturbance error of the closed-loop system with the PID controller tuned by the proposed method are 1.15 and 0.146 s, respectively. By applying τ and L to Eq. (12) the predicted rise time can be obtained as 1.02 s. By applying τ and K_s to Eq. (14) the predicted maximum load disturbance error can be obtained as 0.153. The predicted errors of rise time and maximum load disturbance error are -11.3% and 5%, respectively.

Example 2: Consider a long apparent dead-time process

$$\frac{e^{-2.5s}}{(s+1)^2}.$$

This process has $K_u=1.55$, $T_u=8.45$, and static gain $K_s=1$. Applying these values to the tuning formulas for the frequency response method, we have $K=0.71$, $T_i=2.67$, $T_d=0.94$, and $b=0.8$. The Astrom-Hagglund rules for frequency re-

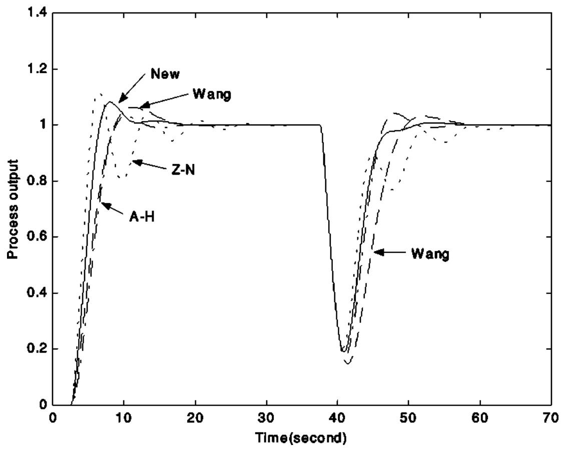

Fig. 6. Set-point response and load disturbance response of the process $e^{-2.5s}/(s+1)^2$ controlled by a PID controller tuned by the proposed method, Astrom-Hagglund's method, Wang's method, and Ziegler-Nichols method.

sponse give $K=0.67$, $T_i=2.53$, $T_d=0.65$, and $b=0.34$. The Ziegler-Nichols method gives $K=0.93$, $T_i=4.22$, and $T_d=1.06$. Wang's method gives $K=0.4, T_i=2, T_d=0.5$. Fig. 6 depicts the set-point and load disturbance responses of the closed-loop system. The proposed method and Astrom-Hagglund's rules give better response than the Ziegler-Nichols method. The load disturbance response of Wang's method is sluggish. Compared to Astrom-Hagglund's method, the improvement of the new method is not so significant in this example.

The rise time and the maximum load disturbance error of the closed-loop system with the PID controller tuned by the proposed method are 2.78 and 0.81 s while the predicted values are 2.63 and 0.733 s, respectively. The predicted errors are about -5.3% and -15.8%.

Table 3 lists the results of some processes controlled by a PID controller tuned using the proposed method. For comparison, controlled results from Wang's method and Astrom-Hagglund's method also are shown. The proposed method provides less integrated absolute error both in set-point response and load disturbance response. When τ or δ of the process is small, the improvement by the new method is significant in set-point response. For processes with larger τ or δ the improvement is not so significant. From the simulation results, we also found that Wang's method has the worst load disturbance response. The reason is that in Wang's method the zeros of the PID controller are used to cancel out the poles of the process model. Hence the load disturbance response is sluggish.

Simulation results also show that the performance assessment formulas can predict the rise time and the maximum load disturbance error of the closed-loop systems quite well (the predicted errors are inside $\pm 20\%$).

Real-time experiment. In order to show the applicability of the proposed tuning rules and performance assessment formulas, an autotuned PID controller was implemented. Experiments were carried out. The test was conducted on an analog computer simulated process. This autotuned PID controller consisted of one personal computer (PC) and a digital signal processor (DSP) card (manufactured by dSPACE Gmbh, DS1102) that plug in the ISA bus of the PC. The programs of

Table 3
Control results of more examples.

			J_s			J_d			t_r (Second)		$l_{\max}$	
Processes	K_u	T_u	Astrom	Wang	New	Astrom	Wang	New	Actual	Pred.	Actual	Pred.
$\frac{e^{-s}}{(1+10s)^2}$	20.6	14.2	7.07	16.74	4.06	0.67	1.96	0.57	3.92	3.84 (-2%)	0.045	0.048 (-6%)
$\frac{e^{-0.5s}}{(1+2s)(1+10s)}$	25.2	5.9	2.90	4.32	1.78	0.22	1.73	0.19	1.48	1.58 (6.8%)	0.037	0.042 (13%)
$\frac{e^{-0.5s}}{(s+1)(0.2s+1)}$	3.41	2.24	1.43	1.29	1.13	0.63	1.00	0.54	0.61	0.72 (18%)	0.41	0.42 (2.3%)
$\frac{e^{-5s}}{(s+1)^3}$	1.25	15.7	10.77	11.96	9.33	8.38	11.74	7.82	5.08	4.32 (-15%)	0.86	0.799 (-8%)

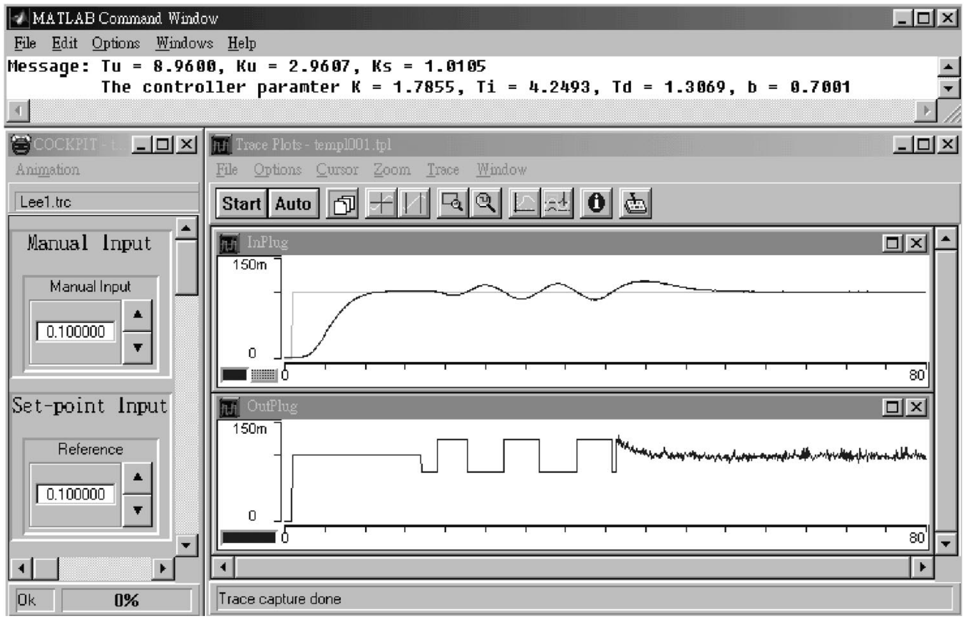

Fig. 7. Results of manual control and relay experiment.

PID control and relay control are implemented in DSP while the data analysis program and supervisory program were implemented using MATLAB language in PC.

The results on an analog computer simulated process with transfer function

$$\frac{1}{(s+1)^5}=\frac{1}{s^5+5s^4+10s^3+10s^2+5s+1}$$

are presented. The results are shown in Figs. 7–10 where the user interface of the PC is shown. At the top of the user interface, a message box is pro-

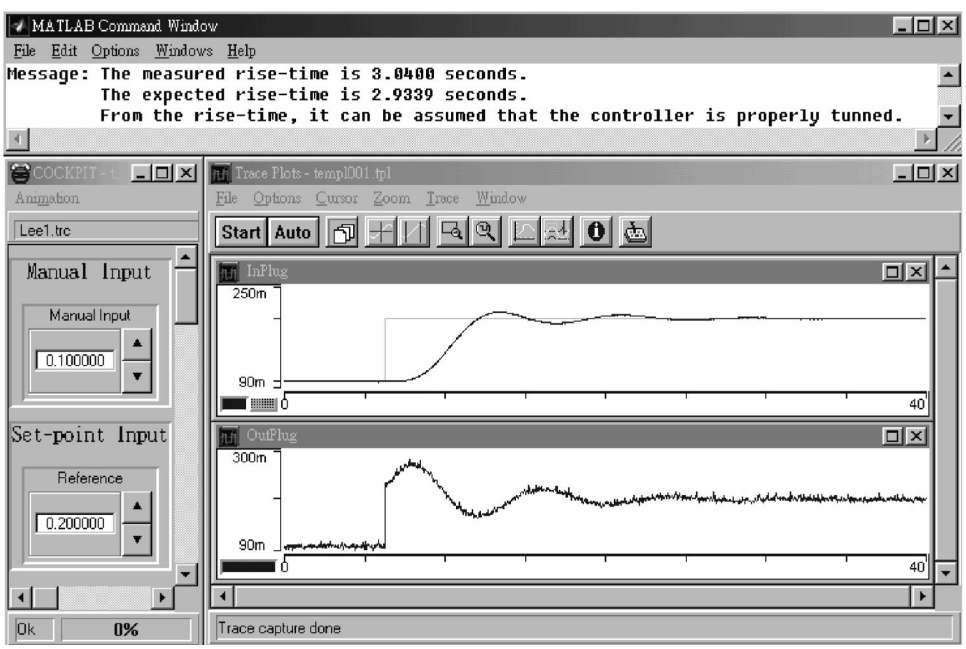

Fig. 8. Set-point response and the performance assessment.

Fig. 9. Performance assessment and retuning of the controller.

vided for to display messages. The process output and set point are shown on the upper display, while the control signal is shown on the lower display. The set-point input and the manual control input are placed on the left-hand side of the user interface where the user can set the values for the set point and change the controller output value when operating in the manual control mode.

Fig. 10. Set-point response and performance assessment of the retuned system.

Fig. 7 shows the manual control and relay experiment results. At the beginning, the controller works in the manual control mode. The user can set the controller output directly and bring the process output near the set point. In this phase, the static process gain K_s can be estimated. The relay experiment is executed where the ultimate period T_u, and the ultimate gain K_u are estimated. Once the K_s, T_u, and K_u values are estimated, the tuning formulas for the frequency domain are used to compute the controller parameters values. The PID controller is then used to control the process and display the controller parameters values in the message box.

Fig. 8 shows set-point change results. The rise time measured is 3.04 s while the predicted rise time [by performance assessment formula in Eq. (13)] is 2.93 s. The difference between these two values is about 3.8%. Therefore the PID controller can be assumed properly tuned. The measured rise time, the predicted rise time, and the assessment results are shown in the message box.

Fig. 9 shows the case when the dynamic characteristics of the process were changed. The transfer function of the process was changed to

$$\frac{1}{s^5+5s^4+10s^3+10s^2+6.9s+1}$$

before the set-point change was made. The measured rise time is 3.93 s and the predicted value is 2.93. The difference between the measured value and predicted value is large (about 34%). Therefore it was assumed that the PID controller was not properly tuned. The relay experiment was reintroduced to retune the PID controller.

Fig. 10 shows the control result of the changed process with new controller. From the measured rise time and predicted ones (error is about 15%), it can be assumed that the controller was properly tuned.

6. Conclusions

New tuning formulas and their performance assessment formulas for PID control of stable processes with monotonic step responses were given. This new formula is based on a genetic algorithm. The simulations and real-time experiment results were given to show the effectiveness and usability of these formulas.

Acknowledgment

National Science Council of Republic of China supported part of this work under grant NSC89-2218-E-150-003.

References

[1] Astrom, K. J., Hagglund, T., Hang, C. C., and Ho, W. K., Automatic tuning and adaptation for PID controllers—A survey. IFAC J. Control Eng. Practice **1**, 699–714 (1993).
[2] Astrom, K. J. and Hagglund, T., PID Controllers: Theory, Design, and Tuning. ISA, Research Triangle Park, NC, 1995.
[3] Chai, T. and Zhang, G., A new self-tuning of PID regulators based on phase and amplitude margin specifications. Acta Automatica Sin. **23**, 167–172 (1997).
[4] Chien, I. L. and Fruehauf, P. S., Consider IMC tuning to improve controller performance. Chem. Eng. Prog. **86**, 33–41 (1990).
[5] Cohen, G. H. and Coon, G. A., Theoretical consideration of retarded control. Trans. ASME **75**, 827–834 (1953).
[6] Abbas, A., A new set of controller tuning relations. ISA Trans. **36**, 183–187 (1997).
[7] Hang, C. C., Astrom, K. J., and Ho, W. K., Refinements of the Ziegler-Nichols tuning formula. IEE Proc.-D: Control Theory Appl. **138**, 111–118 (1991).
[8] Ho, W. K., Hang, C. C., and Cao, L. S., Tuning of PID controllers based on gain and phase margins specifications. Automatica **31**, 497–502 (1995).
[9] Morari, M. and Zafiriou, E., Robust Process Control. Prentice-Hall, Englewood Cliffs, NJ, 1989.
[10] Smith, C. A. and Corripio, C. B., Principles and Practices of Automatic Process Control. Wiley, New York, 1985.
[11] Shinkey, F. G., Process Control System Application, Design and Tuning, 3rd ed. McGraw-Hill, New York, 1988.
[12] Zhuang, G., Shao, C., and Chai, T., A new method for independently tuning PID parameters. Proceedings of the 35th CDC, 1996, Kobe, Japan, 2527–2532.
[13] Zhaung, M. and Atherton, D. P., Automatic tuning of optimal PID controllers. IEE Proc.-D: Control Theory Appl. **140**, 216–224 (1993).
[14] Ziegler, J. G. and Nichols, N. B., Optimal settings for automatic controllers. Trans. ASME **64**, 759–768 (1942).
[15] Wang, Q. G., Lee, T. H., Fung, H. W., Bi, Q., and Zhang, Y., PID Tuning for improved performance. IEEE Trans. Control Syst. Technol. **7**, 457–465 (1999).
[16] Chen, B. S., Cheng, Y. M., and Lee, C. H., A genetic approach to mixed H_2/H_∞ optimal PID control. IEEE Control Syst. Mag. **15**, 51–60 (1995).
[17] Gen, M. and Cheng, R., Genetic Algorithms & Engineering Optimization. Wiley, New York, 2000.
[18] Astrom, K. J., Hang, C. C., Persson, P., and Ho, W. K., Toward intelligent PID control. Automatica **28**, 1–9 (1992).

Jing-Chung Shen was born in Nantou, Taiwan in 1963. He received his B.S., M.S., and Ph.D. degrees in National Cheng Kung University in 1985, 1987, and 1990, respectively, in Electrical Engineering. From July 1990 to July 1992, he was an engineer in Computer and Communication Laboratory of Industrial Technology Research Institute, Hsin Chu, Taiwan. In August 1992, he joined the Department of Automation Engineering, National Huwei Institute of Technology. He is currently an Associate Professor. His research interests include robust control, intelligent control, and application of digital signal processors.