Available online at www.sciencedirect.com
ScienceDirect
ISA Transactions 47 (2008) 45–52
www.elsevier.com/locate/isatrans

An improved auto-tuning scheme for PI controllers

Rajani K. Mudi[a,*], Chanchal Dey[b], Tsu-Tian Lee[c]

[a] *Department of Instrumentation and Electronics Engineering, Jadavpur University, Sector III, Block LB/8, Salt-lake, Calcutta 700098, India*
[b] *Department of Applied Physics, University of Calcutta, 92, A.P.C. Road, Calcutta 700009, India*
[c] *Department of Electrical and Control Engineering, National Chiao-Tung University, Hsinchu 300, Taiwan*

Received 21 August 2006; accepted 21 July 2007
Available online 12 September 2007

Abstract

Ziegler–Nichols tuned PI and PID controllers are usually found to provide poor performances for high-order and nonlinear systems. In this study, an improved auto-tuning scheme is presented for Ziegler–Nichols tuned PI controllers (ZNPICs). With a view to improving the transient response, the proportional and integral gains of the proposed controller are continuously modified based on the current process trend. The proposed controller is tested for a number of high-order linear and nonlinear dead-time processes under both set-point change and load disturbance. It exhibits significantly improved performance compared to ZNPIC, and Refined Ziegler–Nichols tuned PI controller (RZNPIC). Robustness of the proposed scheme is established by varying the controller parameters as well as the dead-time of the process under control.

Keywords: PI/PID controllers; Ziegler–Nichols tuning; Auto-tuning; Load disturbance

1. Introduction

In process industries PI and PID controllers are generally used due to their simple design and tuning methods [1,2]. Due to the presence of measurement noise, PI controllers are more preferable than PID controllers. The absence of derivative action makes a PI controller simple and less sensitive to noise [1]. In practice, nearly 90% of all industrial PID controllers have their derivative action turned off [2]. The most important step for a successful controller design is its tuning. Ziegler–Nichols (ZN) ultimate cycle method [3] is widely used to determine reasonably good settings of PI and PID controllers [4,5]. Ziegler–Nichols tuned PI controllers (ZNPICs) exhibit good performance for first-order processes, but they usually fail to provide satisfactory performance for high-order and/or nonlinear systems, which represent most of the practical processes. Specially, performances of ZNPICs under set-point change are not acceptable in many cases due to excessive oscillation associated with a large overshoot [6–8].

To overcome these limitations of PI controllers, various auto-tuning schemes are suggested to modify the controller gains through some adaptation mechanisms [7,9–12]. The Refined Ziegler–Nichols (RZN) tuning formula [7] modifies the tuning rules by characterizing the process in terms of normalized gain and normalized dead-time. Basilio and Matos [8] proposed a PI controller based on transient performance specification with monotonic step response. Panda et al. [9] developed a gain scheduled PI controller, which continuously updates its proportional and integral gains depending on the instantaneous process error. Based on information about the transient response, an iterative gain tuning method is proposed and implemented in [12]. Different set-point weighting methods are proposed to avoid the large overshoot in the step response [7,13–15]. Hang et al. [7] used fixed set-point weighting factors in ZNPICs, which can provide smaller overshoots, but no improvement in load regulation. A variable set-point weighting [13] is also proposed to further improve the transient response. Set-point weighting techniques for large dead-time, and unstable processes are respectively suggested in [14,15]. Most of these controllers [7–9,12–15] are developed for linear first- or second-order models, which are not always possible for practical processes.

* Corresponding author. Tel.: +91 33 23352587; fax: +91 33 23357254.
E-mail addresses: rkmudi@yahoo.com, rkmudi@iee.jusl.ac.in (R.K. Mudi), cdaphy@caluniv.ac.in (C. Dey), ttlee@cn.nctu.edu.tw (T.-T. Lee).

0019-0578/$ - see front matter
doi:10.1016/j.isatra.2007.07.002

Soft-computing tools like fuzzy logic and neural networks are being used for designing improved PI controllers with increased robustness [16–21]. The overall gain of a simple PI-type fuzzy controller [16,17] is continuously adjusted using fuzzy IF–THEN rules defined on the current process states. Similar types of fuzzy rules are also used to parameterize a PI controller with a dead-beat format [18]. Lee [19] proposed a PI-type fuzzy controller with resetting action to avoid the large accumulation of controller output, responsible for the excessive oscillation in set-point response. Kulic et al. [20] developed a neural network based gain scheduled PI controller. In [21] a fuzzy auto-tuning scheme is used for a DC brushless servo system. In spite of a number of merits, there are many limitations while designing a fuzzy or neuro-fuzzy controller, since there is no standard methodology for its various design steps, and no well-defined criteria for selecting suitable values for its large number of tunable parameters.

The above discussion reveals that many researchers have tried to improve the performance of PI controllers. Several of the proposed techniques [8,11,12,14,15] are inherently complicated from their practical implementation point of view. Some of them [7,9,13] can provide improved performance under set-point change but fail to do so under load disturbance. In the present study, a new auto-tuning scheme is proposed to enhance the performance of a ZNPIC under both set-point change and load disturbance. The proposed controller can be considered as an augmented Ziegler–Nichols tuned PI controller, and will be denoted by AZNPIC. The tuning policy of AZNPIC is based on the idea; *when the process is moving towards the desired set-point, control action is made weak, on the other hand when it moves away from the set-point control action is made aggressive*. We tried to incorporate this idea in the proposed AZNPIC in a very simple way by introducing an online gain updating factor, which continuously modifies its proportional and integral gains based only on the recent process trend. Observe that, RZNPIC [7] modifies the ZN-tuned proportional and integral gains based on normalized gain and normalized dead-time of the process. But our AZNPIC modifies those gains based on the current process trend in terms of error (e) and change of error (Δe) of the controlled variable. Therefore, unlike RZNPIC the modified parameters of AZNPIC are not dependent on the model of the process under control, although both RZNPIC and AZNPIC are refined forms of ZNPIC. Moreover, RZNPIC is found to provide improved performance for a limited range of normalized gain and normalized dead-time. On the other hand, there is no such constraint for AZNPIC.

The proposed controller is tested with several high-order linear and nonlinear dead-time processes. It shows better performance than ZNPIC as well as RZNPIC. The rest of the paper is divided into three sections. In Section 2, the design and auto-tuning strategy of the proposed AZNPIC are described in detail. The simulation results for various second-order and third-order processes including a pH process are presented in Section 3. There is a conclusion in Section 4.

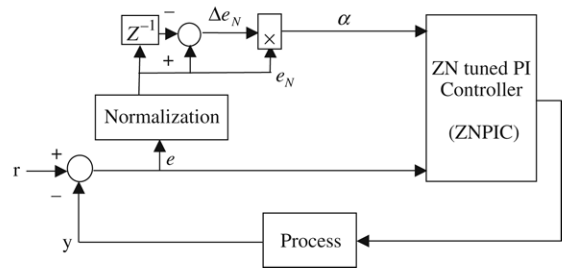

Fig. 1. Block diagram of the proposed AZNPIC.

2. The proposed controller — AZNPIC

Block diagram of the proposed AZNPIC is shown in Fig. 1. It shows that the gain updating factor α, a function of error (e) and change of error (Δe) of the controlled variable continuously adjusts the parameters of a ZNPIC. Next, its design and tuning strategy are discussed in detail.

2.1. Design of AZNPIC

The discrete form of a PI controller is expressed as

$$u(k) = k_p \left[e(k) + \frac{\Delta t}{T_i} \sum_{i=0}^{k} e(i) \right], \tag{1}$$

$$\text{or } u(k) = k_p e(k) + k_i \sum_{i=0}^{k} e(i), \tag{2}$$

where $k_i = k_p(\Delta t / T_i)$.

In Eq. (1), $e(k) = r - y(k)$ is the error, where r is the set-point, and $y(k)$ is the process output at kth instant. k_p and T_i are the proportional gain and integral time respectively, and Δt is the sampling interval. In Eq. (2), k_i is the integral gain. For a ZNPIC, k_p and T_i are obtained according to the following ZN ultimate cycle tuning rules [3]:

$$k_p = 0.45 k_u, \tag{3}$$

$$\text{and } \quad T_i = 0.833 t_u, \tag{4}$$

where k_u and t_u are the ultimate gain and ultimate period respectively. In AZNPIC, these ZN-tuned proportional and integral gains (i.e., k_p and k_i of Eq. (2)) are proposed to modify using the gain updating factor α through the following empirical relations:

$$k_p^t = k_p(1 + k_1 |\alpha(k)|), \tag{5}$$

$$k_i^t = k_i(0.5 + k_2 \alpha(k)), \tag{6}$$

where, k_p^t and k_i^t are the modified proportional and integral gains respectively at kth instant. k_1 and k_2 are two positive constants, which are used to determine the required variations of k_p^t and k_i^t from their initial values to achieve the desired response. The gain updating factor α is defined by

$$\alpha(k) = e_N(k) \times \Delta e_N(k), \tag{7}$$

$$\text{where, } e_N(k) = \frac{e(k)}{e_{\max}}, \tag{8}$$

$$\text{and } \Delta e_N(k) = e_N(k) - e_N(k-1). \tag{9}$$

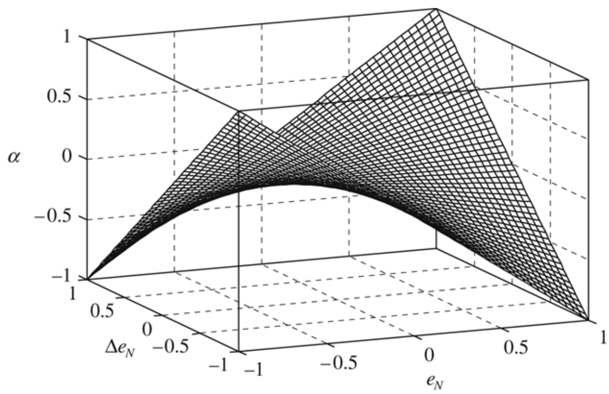

Fig. 2. Variation of α with e_N and Δe_N.

Here, $e_N(k)$ and $\Delta e_N(k)$ are taken as the normalized values of $e(k)$ and $\Delta e(k)$ respectively, and $e_{\max}$ is the maximum possible value of process error. According to Eq. (7), without loss of generality it can be assumed that the possible variation of α will be within the range $[-1, 1]$ for all close-loop stable systems. Fig. 2 shows the highly nonlinear variation of α, which indicates that unlike ZNPIC, AZNPIC has nonlinear proportional (k_p^t) and integral (k_i^t) gains. Finally, the proposed AZNPIC can be expressed as

$$u^t(k) = k_p^t e(k) + k_i^t \sum_{i=0}^{k} e(i), \tag{10}$$

where $u^t(k)$ is the modified control action. To achieve the desired performance, controller parameters k_p^t and k_i^t are to be properly tuned, which means appropriate values of k_1 and k_2 are to be selected as indicated by Eqs. (5) and (6). These may be obtained through some suitable optimization techniques for achieving some specific performance indices. In the present study, the values of k_1 and k_2 are selected heuristically keeping in mind an overall improved performance under both set-point change and load disturbance, and those same values are used in simulation experiments for all the processes. An important feature of AZNPIC is that as the traditional control structure is preserved, an existing conventional PI controller can be modified into the proposed form by incorporating the easily computable dynamic factor α. Moreover, the proposed scheme is model free, since α depends only on the recent process states, e_N and Δe_N.

2.2. Tuning strategy

The objective of the proposed auto-tuning scheme is that, subsequent to any set-point change or load disturbance, the proportional and integral gains of AZNPIC will be continuously modified to have a quick recovery of the process without a large oscillation. While designing AZNPIC, the following important points are taken into consideration to provide the appropriate control action in different operating phases. For easy understanding, a typical close-loop response of a second-order under-damped system is illustrated in Fig. 3.

(i) When the controlled variable is far from the set-point and moving towards it, e.g., point A or C in Fig. 3, proportional gain should be reasonably large to reach the set-point quickly but at the same time integral gain should be small enough to prevent the large accumulation of control action, which may result in a large overshoot or undershoot in future. In such situations (i.e., e and Δe are of opposite sign), α becomes negative, which will make the proportional gain k_p^t larger than k_p (proportional gain of ZNPIC) and integral gain k_i^t smaller than k_i (integral gain of ZNPIC) according to Eqs. (5) and (6) respectively. This type of gain variation will try to reduce the overshoot and/or undershoot without sacrificing the speed of response. Moreover, a reduced integral gain is important to avoid integral windup problem, especially, when the process dead-time becomes considerably large.

(ii) When the controlled variable is far from the set-point and moving further away from it (e.g., point B in Fig. 3), proportional as well as integral gains should be large enough to bring back the controlled variable to its desired value. Under such situations, both e and Δe have large values with the same sign, thereby making α large and positive according to Eq. (7). Such a large and positive α makes both k_p^t and k_i^t larger than their respective initial values according to Eqs. (5) and (6). As a result, the control action becomes more aggressive (i.e., $u^t > u$ according to Eqs. (2) and (10)) during such operating phases. Therefore, AZNPIC satisfies the need for a strong control action to improve the process recovery.

(iii) Industrial processes are often subjected to load disturbances. An efficient controller should provide a good regulation against such sudden changes in load, and restore the desired process state within a shortest possible time. This can be accomplished by increasing the controller gain. Observe that, at the event of any load disturbance, process will move away rapidly from the set-point, e.g., point D in Fig. 3. In such a situation, α becomes positive as both e and Δe are of the same sign. Therefore, both proportional and integral gains of AZNPIC will be increased according to Eqs. (5) and (6). These higher gains will help to provide a better load rejection.

The above discussion indicates that the proposed scheme always attempts to modify the control action in the right direction to improve the transient response due to both set-point change and load disturbance, irrespective of the type of process under control. In the next section this fact will be justified through extensive simulation experiments on a wide range of linear and nonlinear dead-time processes.

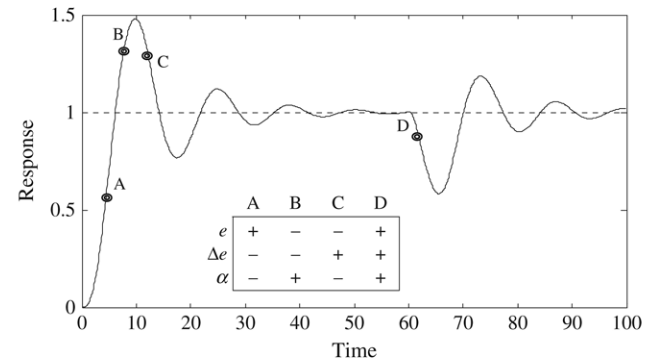

Fig. 3. Typical close-loop response of an under-damped second-order process.

Table 1
Performance analysis of the second-order linear process in (11) with $L = 0.2$ s

	ZNPIDC	ZNPIC	RZNPIC	AZNPIC				
				$k_1 = 1, k_2 = 30$	$k_1 = 1, k_2 = 24$	$k_1 = 1, k_2 = 36$	$k_1 = 0.8, k_2 = 30$	$k_1 = 1.2, k_2 = 30$
$\%OS$	60.61	57.28	9.93	4.04	2.68	5.99	4.21	3.86
t_r (s)	1.00	1.20	2.50	2.30	1.90	2.50	2.30	2.30
t_s (s)	4.00	11.40	5.10	2.60	5.00	4.20	2.60	2.60
IAE	2.22	4.08	3.07	2.12	2.13	2.18	2.13	2.12
ITAE	17.30	47.64	33.99	21.81	23.59	20.53	21.81	21.82

3. Results

Effectiveness of the proposed scheme is verified through simulation experiments on linear as well as nonlinear processes including a pH control system with dead-time (L). In addition to response characteristics, performance of the proposed AZNPIC is compared with ZNPIC, RZNPIC, and ZN-tuned PID controller (ZNPIDC) with respect to a number of indices, such as percentage overshoot ($\%OS$), rise-time (t_r), settling-time (t_s), integral-absolute-error (IAE), and time-integral-absolute-error (ITAE). The values of tuning parameters, k_1 and k_2 are empirically chosen as 1 and 30 respectively (i.e., $k_1 = 1$ and $k_2 = 30$). Fourth-order Runge–Kutta method is used for numerical integration. The detailed performance analysis for various types of processes is discussed below.

3.1. Second-order linear process

Transfer function of the second-order linear process is given by

$$G_p(s) = \frac{e^{-Ls}}{(1+s)^2}. \tag{11}$$

Response characteristics for the process in (11) with $L = 0.2$ s under ZNPIC, RZNPIC, and AZNPIC is shown in Fig. 4(a). Various performance indices are listed in Table 1. Fig. 4(a) and Table 1 clearly reveal a significant performance improvement under AZNPIC due to both set-point change and load disturbance over ZNPIC. For example, percentage overshoot has been reduced from more than 57% to nearly 4%, and the settling-time is reduced over 80% with respect to ZNPIC. Moreover, Table 1 and Fig. 4(a) indicate that the performance of AZNPIC is better than that of RZNPIC. Even the overshoot of ZNPIDC (60.61%) is found to be too high to be acceptable. To study the robustness of the proposed scheme a 50% higher value of dead-time, i.e., $L = 0.3$ s is considered with the same controller setting as that of $L = 0.2$ s, and the corresponding responses are shown in Fig. 4(b). Fig. 4(b) also shows the superiority of AZNPIC over RZNPIC and ZNPIC. To investigate the sensitivity of the tuning parameters k_1 and k_2 on the performance of AZNPIC, their values are changed by ±20% from their initial settings $k_1 = 1$ and $k_2 = 30$. Table 1 indicates that with such deviated values of k_1 and k_2, AZNPIC still maintains nearly the same level of performance as that with the initial values.

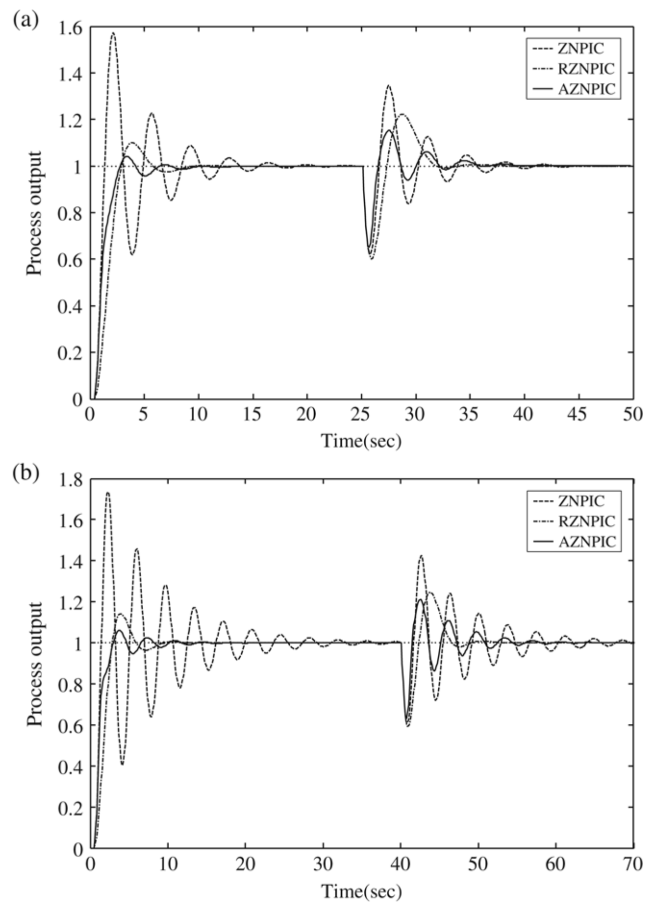

Fig. 4. Responses of the second-order linear process in (11) with (a) $L = 0.2$ s and (b) $L = 0.3$ s.

3.2. Second-order nonlinear process

The performance of the proposed auto-tuner is tested for different nonlinear processes. One of them is described by

$$\frac{d^2y}{dt^2} + \frac{dy}{dt} + 0.2y^2 = u(t-L). \tag{12}$$

Fig. 5(a) shows the responses of (12) with $L = 0.2$ s under ZNPIC, RZNPIC, and AZNPIC. Table 2 presents the different performance indices. Though the controllers are tuned for $L = 0.2$ s, a higher value, i.e., $L = 0.3$ s is also considered keeping the controller's setting same. Response characteristics for $L = 0.3$ s is illustrated in Fig. 5(b). From Fig. 5 and Table 2 it is found that in case of AZNPIC, $\%OS$ is considerably reduced, and steady state is reached faster, which reveal that

Table 2
Performance analysis of the second-order nonlinear process in (12) with $L = 0.2$ s

	ZNPIDC	ZNPIC	RZNPIC	AZNPIC				
				$k_1 = 1, k_2 = 30$	$k_1 = 1, k_2 = 24$	$k_1 = 1, k_2 = 36$	$k_1 = 0.8, k_2 = 30$	$k_1 = 1.2, k_2 = 30$
$\%OS$	69.85	80.73	84.31	36.80	41.38	33.03	36.58	37.03
t_r (s)	1.30	1.50	2.00	1.90	1.80	2.00	1.90	1.90
t_s (s)	7.80	30.60	38.90	14.10	14.00	14.30	14.10	14.10
IAE	3.10	8.83	11.69	4.18	4.36	4.05	4.18	4.19
ITAE	39.06	195.41	300.83	86.24	87.94	84.90	86.18	86.31

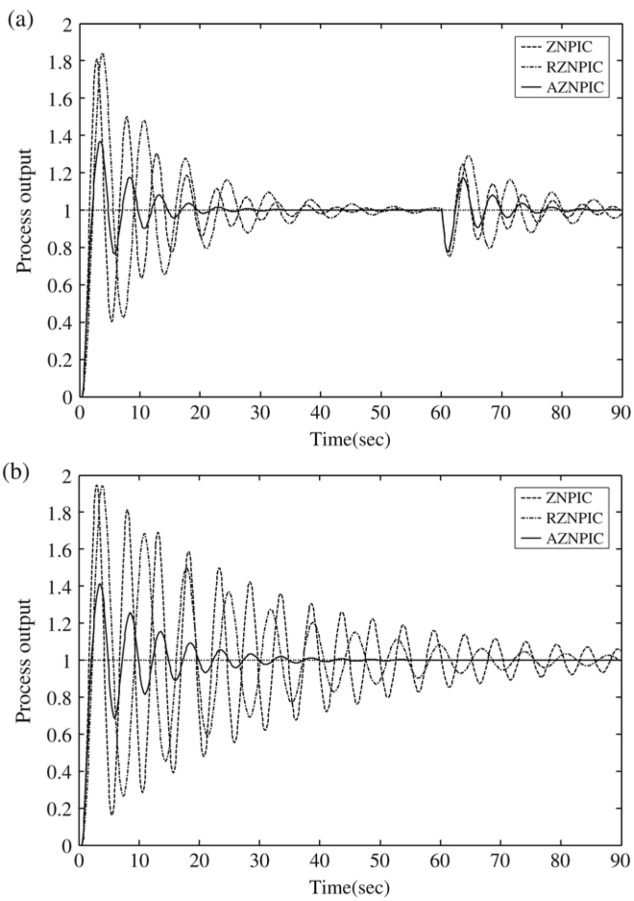

Fig. 5. Responses of the second-order nonlinear process in (12) with (a) $L = 0.2$ s and (b) $L = 0.3$ s.

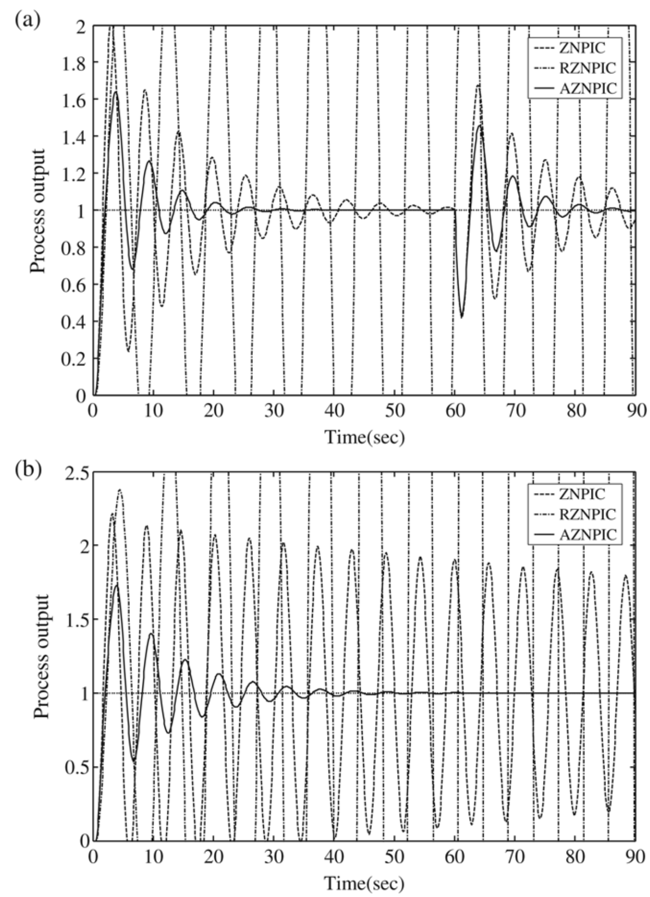

Fig. 6. Responses of the marginally stable process in (13) with (a) $L = 0.2$ s and (b) $L = 0.3$ s.

unlike ZNPIC and RZNPIC, AZNPIC is capable of providing an acceptable performance. From Table 2 it is observed that even ZNPIDC fails to restrict the large $\%OS$ (about 70%). Like the previous case, the performance of AZNPIC is not affected much due to a reasonable change in the values of k_1 and k_2 from their initial settings (Table 2).

3.3. Second-order marginally stable process

Transfer function of the second-order marginally stable process is

$$G_p(s) = \frac{e^{-Ls}}{s(s+1)}. \tag{13}$$

Responses of this integrating process with $L = 0.2$ s as shown in Fig. 6(a), and various indices of Table 3 justify that ZNPIC provides a very poor performance due to excessively large overshoot and oscillation. The $\%OS$ of ZNPIDC is also very large (about 80%). Whereas, AZNPIC is found to reduce the $\%OS$ and t_s by about 40% and 60%, respectively compared to ZNPIC. When the process dead-time is increased from 0.2 to 0.3 s without changing the controller parameters, the system goes to the verge of instability under ZNPIC as illustrated by Fig. 6(b). However, AZNPIC still provides a highly stable performance. Observe that, in both cases ($L = 0.2$ s and 0.3 s) the integrating process in (13) becomes completely unstable under RZNPIC. Once again, Table 3 shows that AZNPIC maintains almost the same level of performance in spite of considerable variations in k_1 and k_2. These facts establish the robust feature of the proposed scheme.

Table 3
Performance analysis of the marginally stable process in (13) with $L = 0.2$ s

	ZNPIDC	ZNPIC	RZNPIC	AZNPIC				
				$k_1 = 1, k_2 = 30$	$k_1 = 1, k_2 = 24$	$k_1 = 1, k_2 = 36$	$k_1 = 0.8, k_2 = 30$	$k_1 = 1.2, k_2 = 30$
%OS	79.40	102.79	140.73	63.88	66.58	62.12	63.76	64.00
t_r (s)	1.30	1.50	2.00	1.80	1.70	1.90	1.80	1.80
t_s (s)	8.70	42.60	Unstable	17.90	17.90	18.00	17.90	18.00
IAE	4.43	16.64	138.97	7.33	7.41	7.30	7.32	7.34
ITAE	96.25	566.25	7445.67	221.07	222.65	220.07	221.08	221.09

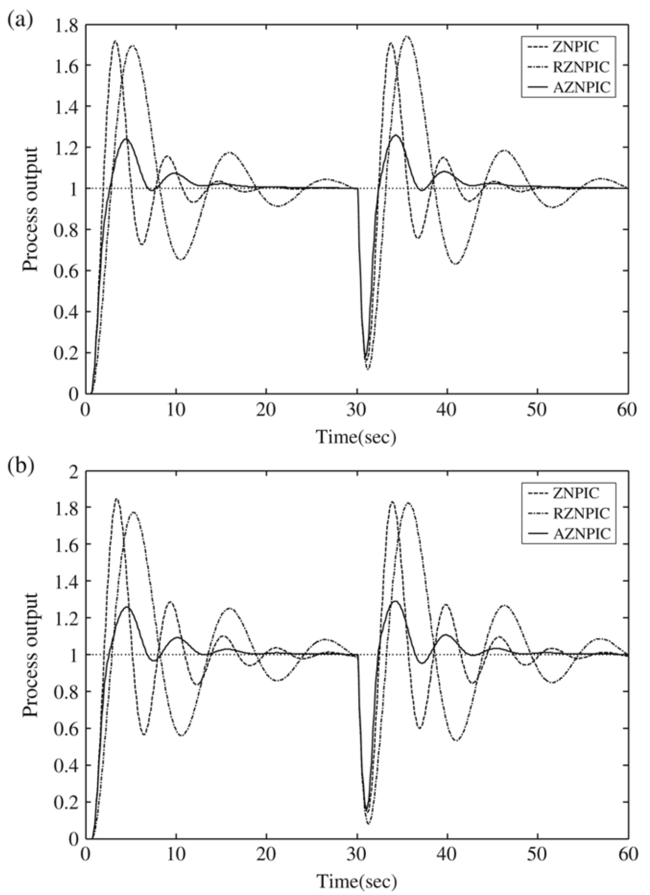

Fig. 7. Responses of the third-order linear process in (14) with (a) $L = 0.4$ s and (b) $L = 0.5$ s.

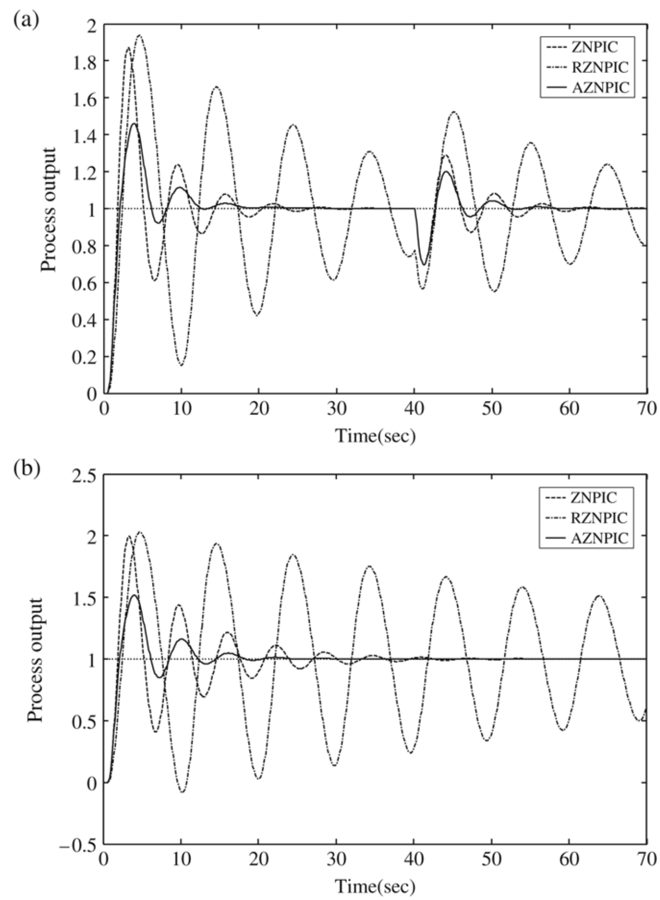

Fig. 8. Responses of the third-order nonlinear process in (15) with (a) $L = 0.3$ s and (b) $L = 0.4$ s.

3.4. Third-order linear process

Though most of the industrial processes can be fairly approximated by a second-order plus dead-time (SOPDT) model, in this study third-order linear as well as nonlinear process models are also tested to justify the effectiveness of the proposed scheme. Such a linear system is expressed by the following transfer function

$$G_p(s) = \frac{e^{-Ls}}{s^3 + s^2 + s + 0.2}. \quad (14)$$

The process in (14) is tested with comparatively larger values of dead-time, i.e., $L = 0.4$ s and 0.5 s. Figs. 7(a) and (b) present the responses of (14), which indicate that the performance of RZNPIC is even inferior to ZNPIC. Various performance indices of Table 4 reveal that the performance of ZNPIDC is also not satisfactory due to a very large overshoot. Response characteristics (Figs. 7(a) and (b)) and Table 4 clearly show a remarkable improvement in the performance of AZNPIC over ZNPIC and RZNPIC. Table 4 also indicates that the performance of AZNPIC remains almost same against ±20% variation of k_1 and k_2.

3.5. Third-order nonlinear process

The following third-order nonlinear process is considered

$$\frac{d^3 y}{dt^3} + \frac{d^2 y}{dt^2} + 0.5y\frac{dy}{dt} = u(t - L). \quad (15)$$

Figs. 8(a) and (b) respectively show the responses of (15) for two different values of dead-time $L = 0.3$ s and 0.4 s. Table 5 includes the various performance indices for $L =$

Table 4
Performance analysis of the third-order linear process in (14) with $L = 0.4$ s

	ZNPIDC	ZNPIC	RZNPIC	AZNPIC				
				$k_1 = 1, k_2 = 30$	$k_1 = 1, k_2 = 24$	$k_1 = 1, k_2 = 36$	$k_1 = 0.8, k_2 = 30$	$k_1 = 1.2, k_2 = 30$
$\%OS$	70.28	71.89	69.58	23.98	25.64	25.28	24.03	23.92
t_r (s)	1.30	1.60	2.40	2.10	1.90	2.40	2.10	2.10
t_s (s)	6.90	12.80	23.10	11.10	10.60	11.70	11.20	11.10
IAE	4.56	7.41	13.09	4.68	4.73	4.73	4.69	4.67
ITAE	64.82	137.18	295.84	81.49	83.89	80.27	81.77	81.23

Table 5
Performance analysis of the third-order nonlinear process in (15) with $L = 0.3$ s

	ZNPIDC	ZNPIC	RZNPIC	AZNPIC				
				$k_1 = 1, k_2 = 30$	$k_1 = 1, k_2 = 24$	$k_1 = 1, k_2 = 36$	$k_1 = 0.8, k_2 = 30$	$k_1 = 1.2, k_2 = 30$
$\%OS$	74.82	87.11	93.71	46.04	49.96	43.12	45.84	46.23
t_r (s)	1.30	1.50	2.10	1.80	1.70	1.90	1.80	1.80
t_s (s)	7.80	16.70	Unstable	11.50	11.30	11.80	11.50	11.50
IAE	3.77	6.70	22.49	4.25	4.30	4.24	4.25	4.25
ITAE	38.61	104.17	603.70	61.86	61.77	62.17	61.87	61.85

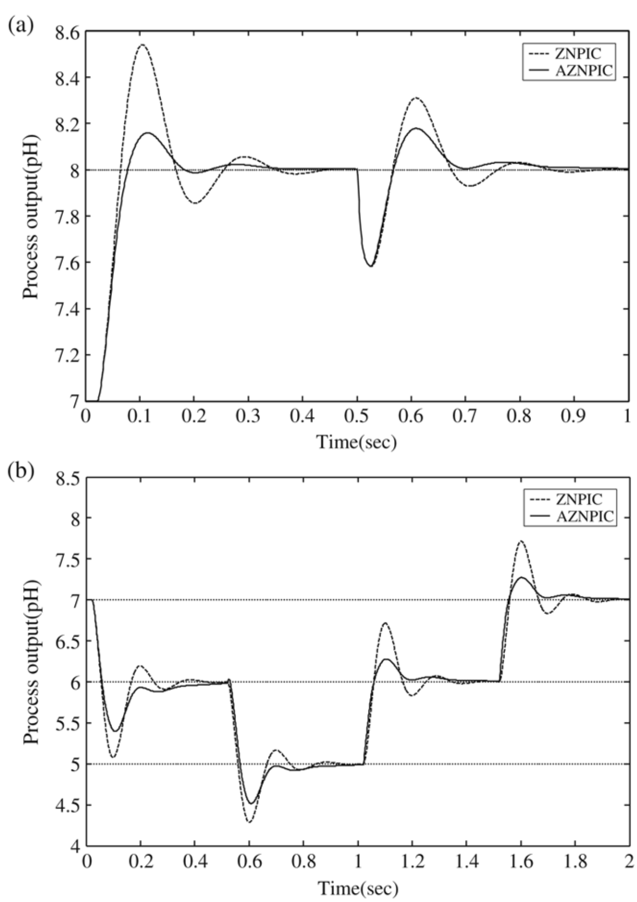

Fig. 9. Responses of pH process in (16) with $L = 0.02$ s for (a) both set-point change and load disturbance and (b) set-point change from different operating points.

0.3 s. Like third-order linear process of (14), the proposed AZNPIC exhibits better performance compared to ZNPIC for the nonlinear process in (15). Here also, RZNPIC shows a very poor performance. Observe that, even the $\%OS$ under AZNPIC (about 45%) is considerably smaller than that of ZNPIDC (about 75%). Reasonable variations (±20%) of k_1 and k_2 reflect little influence on the performance of AZNPIC (Table 5).

3.6. pH process

Now, we evaluate the performance of AZNPIC for a practical process, pH neutralization. For this process both ZNPIDC and RZNPIC are not suitable. In the case of ZNPIDC the system becomes completely unstable, whereas the calculated values of normalized gain and normalized dead-time for RZNPIC are found to be out of the proposed ranges [7]. In most of the cases, linear models are assumed for the pH process [22–24]. We use the same model considered in [22], which is defined by

$$G_p(s) = \frac{e^{-Ls}}{(1+s)(1+0.1s)^2}. \tag{16}$$

Performances for (16) with $L = 0.02$ s under AZNPIC and ZNPIC are shown in Fig. 9 and Table 6. Fig. 9(a) shows the responses due to both set-point change and load disturbance, and Fig. 9(b) depicts the responses due to step set-point change from different operating points. Considerably improved performance of AZNPIC over ZNPIC is noticed from Fig. 9 and Table 6. For example, there is a large reduction in $\%OS$ from 54% to 16% with a significant improvement (about 50%) in t_s.

From the above results for various processes it is evident that in each case the proposed AZNPIC shows consistently improved performance over ZNPIC and RZNPIC under both set-point change and load disturbance, it even produces considerably smaller overshoots than those of ZNPIDC.

4. Conclusion

A simple model independent auto-tuning scheme has been proposed for Ziegler–Nichols tuned PI controllers. It continuously adjusts the controller (ZNPIC) gains through a single

Table 6
Performance analysis of the pH process in (16) with $L = 0.02$ s

	ZNPIC	AZNPIC				
		$k_1 = 1, k_2 = 30$	$k_1 = 1, k_2 = 24$	$k_1 = 1, k_2 = 36$	$k_1 = 0.8, k_2 = 30$	$k_1 = 1.2, k_2 = 30$
%OS	53.99	15.94	17.84	14.38	15.88	16.01
t_r (s)	3.10	3.30	3.30	3.40	3.30	3.30
t_s (s)	30.90	16.20	15.80	16.80	16.20	16.20
IAE	13.77	9.72	9.80	9.69	9.72	9.72
ITAE	347.77	241.58	243.13	240.79	241.69	241.77

nonlinear parameter α, defined on the instantaneous process states. It can be easily incorporated in an existing control loop. Effectiveness of the proposed controller (AZNPIC) has been tested through extensive simulation experiments for a wide range of processes. In each case, AZNPIC has shown better performance in transient as well as steady state conditions compared to ZNPIC and RZNPIC. Robustness of the proposed scheme has been established by varying the dead-time without changing the controller parameters for a given process, and using the same values of k_1 and k_2 for all the processes in simulation experiments. Moreover, for a considerable variation of the tuning parameters k_1 and k_2 from their initial values, the proposed controller has maintained almost the same level of performance.

In the present study, we have used empirical values of k_1 and k_2. Further works may be done to find their more appropriate values. Stability analysis for this nonlinear controller (AZNPIC) may also be tried to make this study more meaningful.

Acknowledgment

This work is partially supported by the National Science Council, Taiwan, ROC, Grant No. NSC-94-2811-E-009-020.

References

[1] Shinsky FG. Process control systems — application, design, and tuning. New York: McGraw-Hill; 1998.
[2] Bissel C. Control engineering. New York: Chapman and Hall; 1994.
[3] Ziegler JG, Nichols NB. Optimum settings for automatic controllers. ASME Trans 1942;64:759–68.
[4] Hågglund T, Åström KJ. Revisiting the Ziegler–Nichols tuning rules for PI control. Asian J Control 2002;4(4):364–80.
[5] Åström KJ, Hägglund T. Revisiting the Ziegler–Nichols step response method for PID control. J Process Control 2004;14(6):635–50.
[6] Åström KJ, Hang CC, Person P, Ho WK. Towards intelligent PID control. Automatica 1992;28(1):1–9.
[7] Hang CC, Åström KJ, Ho WK. Refinements of Ziegler–Nichols tuning formula. IEE Proc-D 1991;138(2):111–8.
[8] Basilio JC, Matos SR. Design of PI and PID controllers with transient performance specification. IEEE Trans Education 2002;45(4):364–70.
[9] Panda SK, Lim JMS, Dash PK, Lock KS. Gain scheduled PI speed controller for PMSM Drive. In: Proc. IEEE industrial electronics society international conference IECON'97, vol. 2. 1997. p. 925–30.
[10] Seborg DE, Edgar TF. Adaptive control strategies for process control: A survey. AICHE J 1986;32(6):881–913.
[11] Kristiansson B, Lennartson B. Robust and optimal tuning of PI and PID controllers. IEE Proc Control Theory Appl 2002;149(1):17–25.
[12] Tursini M, Parasiliti F, Zhang D. Real time gain tuning of PI controllers for high performance PMSM drives. IEEE Trans Industry Appl 2002; 38(4):1018–26.
[13] Hang CC, Cao L. Improvement of transient response by means of variable set point weighting. IEEE Trans Industrial Electron 1996;43(4):477–84.
[14] Khan BZ, Lehman B. Set point PI controllers for systems with large normalized dead time. IEEE Trans Control Syst Technol 1996;4(4): 459–66.
[15] Padama Sree R, Chidambaram M. Simple method of calculating set point weighting parameter for unstable systems with a zero. Comput Chem Eng 2004;28(11):2433–7.
[16] Mudi RK, Pal NR. A robust self-tuning scheme for PI and PD type fuzzy controllers. IEEE Trans Fuzzy Syst 1999;7(1):2–16.
[17] Mudi RK, Pal NR. A self-tuning fuzzy PI controller. Fuzzy Sets Syst 2000;115(2):327–38.
[18] Bandyopadhyay R, Patranabis D. A fuzzy logic based PI autotuner. ISA Trans 1998;37(3):227–35.
[19] Lee J. On methods for improving performance of PI-type fuzzy logic controllers. IEEE Trans Fuzzy Syst 1993;1(4):298–301.
[20] Kulic F, Kukolj D, Levi M. Artificial neural network as a gain scheduled for PI speed controller in DC motor drives. In: Proc. fifth seminar on neural network applications in electrical engineering — NEUREL-2000. 2000. p. 199–203.
[21] Inoue K. An advanced DC brushless servo drive system with a fuzzy logic-based self-tuning control scheme and its practical evaluations. Internat J Electron 1996;80(2):223–33.
[22] Åström KJ, Wittenmark B. Adaptive control. New York: Addison Wesley; 1995.
[23] Böling JM, Seborg DE, Hespanha JP. Muli-model adaptive control of a simulated pH neutralization process. Control Eng Practice 2007;15(6): 663–72.
[24] Ali E. pH control using PI control algorithms with automatic tuning method. Chemical Eng Res Design 2001;79(A5):611–20.

Rajani K. Mudi is a Reader in the Dept. of Instrumentation and Electronics Engg, Jadavpur University, India. He received B. Tech and M. Tech in 1990 and 1992 respectively from Calcutta University, India, and Ph.D. from Jadavpur University in 1999. His research interests are in intelligent control, qualitative modeling and bioinformatics. He visited National Taiwan University, Taiwan from August 2006 to May 2007. He was the co-ordinator of *AFSS-2002* and the secretary of *ICONIP-2004*. He co-edited a volume 'Neural Information Processing' by Springer-Verlag, Germany and served as a Guest co-editor of a special issue: 'Computational Intelligence for Decision Making Systems' of *Int. Journal of Intelligent Systems*.

Chanchal Dey received the B. Tech degree in Instrumentation and Electronics Engineering in 1996 and M. Tech degree in Instrumentation and Control Engineering in 1999, respectively from Jadavpur University and Calcutta University, India. From 1996 to 1997 he worked as an Engineer in Gas Authority of India Ltd. He was a guest lecturer in the Department of Instrumentation and Electronics Engineering at Jadavpur University from 1999 to 2005. Since 2003, he has been a lecturer in the Department of Applied Physics, Calcutta University. His research includes intelligent PID control, fuzzy and neuro-fuzzy control.