Korean J. Chem. Eng., 40(6), 1277-1283 (2023)
DOI: 10.1007/s11814-022-1356-0

pISSN: 0256-1115
eISSN: 1975-7220

New non-interactive form of the proportional-integral-derivative-acceleration (PIDA) controller and its explicit tuning rule

Jae Pil Heo*, Sanghun Lim*, Chang Gyu Im*, Kyung Hwan Ryu,†, and Su Whan Sung*,†**

*Department of Chemical Engineering, Kyungpook National University, Daegu 41566, Korea
**Department of Chemical Engineering, Sunchon National University, 225 Jungang-ro, Suncheon, Jeollanam-do 57922, Korea
(*Received 17 July 2022 • Revised 26 October 2022 • Accepted 23 November 2022*)

Abstract−The physical meaning of the derivative and the acceleration term of the proportional-integral-derivative-acceleration (PIDA) controller was analyzed, and a new non-interactive form of the PIDA controller is proposed. Also, a new tuning rule for the PIDA controller was developed by combining the physical meaning of the two derivative terms with the previous integral of the time-weighted absolute value of the error (ITAE) tuning rule for the PID controller. The proposed tuning rule, composed of simple explicit algebraic equations, provides excellent performance for various processes without using any optimization methods and iterative computation.

Keywords: PIDA Controller, Acceleration, Second Derivative, Explicit Tuning Rule, ITAE

INTRODUCTION

The proportional-integral-derivative (PID) controller has been widely used due to its simple and intuitive structure, acceptable robustness to uncertainty, and good control performance for most processes. The PID controller has the three terms of the proportional, integral, and derivative terms, and three tuning parameters of the proportional gain, integral time, and derivative time [1,2]. The tuning parameters need to be appropriately determined by considering the dynamics of the process to achieve acceptable control performance.

To determine the fine-tuning parameters, many types of PID tuning methods have been proposed. Despite the theoretically optimal performance of the optimization-based tuning methods [3-7], they are not favorable in practice due to iterative computations. On the other hand, the tuning rules in the form of explicit algebraic equations have been widely used on account of their simplicity. The Ziegler-Nichols (ZN) [8], internal model control (IMC) [9], integral of the time-weighted absolute value of the error for the first-order plus time delay model (ITAE-1) [10], and ITAE-2 tuning rule for the second-order plus time delay model [11] are the representative tuning rules. These tuning rules have contributed much to the control industry. Especially the ITAE-2 PID tuning rule guarantees almost the same control performance as that of the optimal PID tuning minimizing the ITAE performance criterion, and it is composed of simple and explicit algebraic equations [10]. Also, tuning strategies to incorporate more various types of processes have been proposed by combining simple explicit tuning rules developed for low-order plus time delay models with model reduction methods [12-14].

†To whom correspondence should be addressed.
E-mail: khryu@scnu.ac.kr, suwhansung@knu.ac.kr

Meanwhile, since the proportional-integral-derivative-acceleration (PIDA) controller was first suggested [15], it has been successfully applied to various industries and achieved better control performance than the PID controller [16-21]. Note that some authors have used the term of the proportional-integral-derivative-second derivative ($PIDD^2$) control rather than that of the PIDA control. The two controllers of the PIDA and $PIDD^2$ are the same because acceleration means the second time derivative of the error. Such improvements by the PIDA controller could be possible because the PIDA utilizes more accurate future errors (equivalent to reducing the time delay of the process) by using the second time-derivative of the error additionally compared to the PID controller.

Many tuning rules of the PIDA controller have been proposed [22-25]. All the tuning approaches use optimization techniques, such as whale optimization [24], particle swarm optimization (PSO) [23], and firefly algorithm [22]. These optimization-based tuning approaches need iterative computations, and they are far from simple and explicit tuning formulas. Several PIDA tuning rules based on a direct synthesis or IMC control principle are also proposed [17]. These methods must determine a prespecified control performance, such as the IMC filter constant before using the tuning rules. Also, the control performance in the time domain can be different from the expectation when the controller is designed in the frequency domain.

In this research, a new form of the PIDA controller is proposed by analyzing the physical meaning of the derivative term and the acceleration (second derivative) term of the PIDA controller. A new simple and explicit tuning rule for the PIDA controller was developed by extending the ITAE-2 tuning rule for the PID controller to the PIDA controller by combining the physical meaning with the previous ITAE-2 tuning rule for the PID controller. The proposed PIDA tuning rule composed of simple explicit algebraic equations shows excellent performance for various processes without requiring any optimization methods or iterative computation.

PROPOSING A NON-INTERACTIVE FORM OF THE PIDA CONTROLLER AND ITS TUNING RULE

1. Mathematical Formulation of Proportional-integral-derivative-acceleration (PIDA) Controller

All the PIDA controllers published until now use the following parallel form:

$$u(t)=k_p e(t)+k_i\int_0^t e(\tau)d\tau+k_d\frac{de(t)}{dt}+k_a\frac{d^2e(t)}{dt^2} \tag{1}$$

where k_p, k_i, k_d, and k_a are the tuning parameters of the PIDA controller. e(t) and u(t) are the control error and the control output, respectively. The first, second, and third terms of the right-hand side of Eq. (1) are the proportional, integral, and derivative terms, respectively. The fourth term is the acceleration term of the PIDA controller. The parallel PIDA form of Eq. (1) makes it very difficult to determine the four tuning parameters in an intuitive and systematic way because they have no physical meaning, and two different units of the time unit and the error unit are not separated. A new form for the PIDA controller needs to be proposed for intuitive tuning and separation of the error unit and the time unit.

Consider the second-order Taylor approximation of the control error at the present time before deriving a new PIDA form:

$$e(t_f)\approx e(t)+\frac{de(t)}{dt}(t_f-t)+\frac{1}{2!}\frac{d^2e(t)}{dt^2}(t_f-t)^2 \tag{2}$$

where, t and t_f denote the present time and the future time, respectively. The Taylor approximation can be used to obtain the future error on the basis of the error (e(t)), error derivative (de(t)/dt), and error acceleration (second derivative, $d^2e(t)/dt^2$) at the present time. For example, the future error at τ_d+t (after τ_d from the present time) can be calculated as follows:

$$e(t+\tau_d)\approx e(t)+\tau_d\frac{de(t)}{dt}+\frac{\tau_d^2}{2!}\frac{d^2e(t)}{dt^2} \tag{3}$$

where, τ_d(de(t)/dt) is the contribution portion of the present first derivative to the future error and $(\tau_d^2/2!)(d^2e(t)/dt^2)$ is the contribution portion of the present second derivative to the future error. With generalizing this interpretation, the future error to be used for the PIDA controller can be defined by combining the two terms of τ_d(de(t)/dt) and $(\tau_a^2/2!)(d^2e(t)/dt^2)$ as follows:

$$e_{future}\approx e(t)+\tau_d\frac{de(t)}{dt}+\frac{\tau_a^2}{2!}\frac{d^2e(t)}{dt^2} \tag{4}$$

Finally, a new non-interactive form of the PIDA controller is derived by adding the integral term of $\int_0^t e(\tau)d\tau/\tau_i$ to reject the offset as follows:

$$u(t)=k_c\left(e(t)+\frac{1}{\tau_i}\int_0^t e(\tau)d\tau+\tau_d\frac{de(t)}{dt}+\frac{\tau_a^2}{2}\frac{d^2e(t)}{dt^2}\right) \tag{5}$$

$$G_{PIDA}(s)=\frac{u(s)}{e(s)}=k_c\left(1+\frac{1}{\tau_i s}+\tau_d s+\frac{\tau_a^2}{2}s^2\right) \tag{6}$$

where, t is the present time. k_c, τ_i, τ_d, and τ_a are the proportional gain, integral time, derivative time, and acceleration time, respectively. $G_{PIDA}(s)$ is the transfer function of the PIDA controller. The physical meaning of the derivative term (τ_d(de(t)/dt)) is the future error portion predicted by the present first derivative of the error as much as τ_d and the acceleration term $((\tau_a^2/2)(d^2e(t)/dt^2))$ is the future error portion predicted by the present acceleration (second derivative of the error) as much as τ_a. It is clear that the units of τ_i, τ_d, and τ_a are time and the unit of k_c is the control output (u(t)) unit over the control error (e(t)) unit.

Note that the derivative (D) and acceleration (A) terms of the PIDA controller reduce the time delay of the process because they contribute to using future error. Therefore, it is clear that the stability of the closed-loop control system using the PIDA controller is better than that of the P or PI controller. Also, the PIDA controller is more stable than the PID controller since the prediction (future error) accuracy of the PIDA controller is better than that of the PID controller, because the PIDA uses the acceleration term additionally. Now, it is concluded that the proportional gain can be set to stronger to achieve a faster closed-loop response because the time delay of the process is reduced as much as the prediction by the derivative and acceleration term and the closed-loop system becomes more stable.

Most industrial PID controllers have a noise-suppressing function by adding a first-order low-pass filter to the derivative term of the ideal PID controller [10]. Similarly, a noise-suppressing PIDA controller is proposed by adding a first-order low-pass filter and the second-order low-pass filter to the derivative term and acceleration term, respectively as follows:

$$G_{PIDA}(s)=\frac{u(s)}{e(s)}=k_c\left(1+\frac{1}{\tau_i s}+\frac{\tau_d}{\alpha_d\tau_d s+1}s+\frac{\tau_a^2}{2(\alpha_a\tau_a s+1)^2}s^2\right) \tag{7}$$

where, α_d and α_a are small constants. Our various simulations, it is recommended that α_d=0.05 and α_a=0.5α_d are good choices for the proposed method. The noise-suppressing effects become stronger, while the control performance degrades more and more as increasing α_d and α_a.

2. Development of an Explicit PIDA Tuning Rule

Assume that the following second-order plus time delay model is given before the tuning of the PIDA controller is performed:

$$G_m(s)=\frac{k_m\exp(-\theta_m s)}{\tau_m^2 s^2+2\tau_m\xi_m s+1} \tag{8}$$

where, k_m, τ_m, ξ_m, and θ_m are static gain, time constant, damping factor, and time delay, respectively. An explicit and straightforward tuning rule for the PIDA controller based on the second-order plus time delay model was developed in this study.

Before developing the proposed tuning rule, one needs to focus on the previously developed integral of the time-weighted absolute value of the error (ITAE) PID tuning rule based on the second-order plus time delay model (ITAE-2) [9,10]. The ITAE-2 tuning rule for the PID controller composed of explicit equations (with no optimization methods and no iterative computations) can provide almost optimal tuning parameters guaranteeing the optimal performance of the PID controller from the ITAE performance criterion point of view. Also, it can be applied to a high-order plus time delay model by converting the given model to a reduced second-order plus time delay model using a model reduction method and calculating the tuning parameters of the PID controller with the ITAE-2 tuning rule for the reduced second-order plus time delay model [10].

In this study, a new simple explicit tuning rule was developed by combining the ITAE-2 PID tuning rule with the fact that the acceleration term of the PIDA controller reduces the time delay of the process. First, assume that the acceleration time is chosen as $\tau_a=\tau_d f_a$ (f_a will be determined later). As a result, the time delay of the second order plus time delay model can be assumed to be reduced as much as $\theta_m f_d$ (f_d will be determined later) by the acceleration term $((\tau_a^2/2)(d^2e(t)/dt^2))$ of the PIDA controller, resulting in a new reduced time delay of $\theta_n=\theta_m-\theta_m f_d$. Then, k_c of the PIDA controller can be tuned to a bigger one compared to that of the PID controller to achieve better control performance because the time delay is reduced by adding the acceleration term to the PID controller. Meanwhile, note that the value of the integral term of the PIDA controller tends to be bigger as increasing k_c, resulting in too strong integral action. So, the integral time (τ_i) needs to be set to a bigger value to prevent excessive integral action. Many simulations confirmed that the problem resulting from increasing k_c can be overcome by increasing the time constant to half of the reduced time delay like $\tau_n=\tau_m+0.5\theta_m f_d$.

That is, the following new second order plus time delay model with the reduced time delay and the increased time constant is obtained by adding the acceleration term of $(\tau_a^2/2)(d^2e(t)/dt^2)$ to the PID controller (equivalent to using the PIDA controller).

$$G_n(s)=\frac{k_n\exp(-\theta_n s)}{\tau_n^2 s^2+2\tau_n\xi_n s+1},\ \theta_n=\theta_m-\theta_m f_d,$$
$$\tau_n=\tau_m+0.5\theta_m f_d,\ k_n=k_m,\ \xi_n=\xi_m \tag{9}$$

Now, it is straightforward to obtain the tuning parameters (k_c, τ_i, and τ_d) of the PIDA controller by applying the ITAE-2 PID tuning rule to the new second order plus time delay of Eq. (9) and the acceleration time constant of $\tau_a=\tau_d f_a$ is obtained.

Here, the optimal parameters of f_d and f_a are obtained by solving the following minimization problem of which the cost function is the integral of the time-weighted absolute value of the error (ITAE) for an extensive number of the second-order plus time delay model ($0.1\le\theta_m/\tau_m\le 2$ and $0.3\le\xi_m\le 2.0$) and a unit step set-point change of $y_s(s)=1/s$:

$$\min_{f_a, f_d}\left\{\int_0^\infty t|e(t)|dt\right\} \tag{10}$$

subject to

$$\frac{y(s)}{u(s)}=\frac{k_n\exp(-\theta_n s)}{\tau_n^2 s^2+2\tau_n\xi_n s+1},\ \theta_n=\theta_m-\theta_m f_d,$$
$$\tau_n=\tau_m+0.5\theta_m f_d,\ k_n=k_m,\ \xi_n=\xi_m \tag{11}$$

$$(k_c,\ \tau_i,\ \tau_d)=\text{ITAE-2}(k_n,\ \tau_n,\ \xi_n,\ \theta_n),\ \tau_a=\tau_d f_a,$$
$$\alpha_d=0.05,\ \text{and}\ \alpha_a=0.5\alpha_d \tag{12}$$

$$\frac{u(s)}{e(s)}=k_c\left(1+\frac{1}{\tau_i s}+\frac{\tau_d}{\alpha_d\tau_d s+1}s+\frac{\tau_a^2}{2(\alpha_a\tau_a s+1)^2}s^2\right),$$
$$e(s)=y_s(s)-y(s),\ y_s(s)=\frac{1}{s} \tag{13}$$

where, (k_c, τ_i, τ_d)=ITAE-2$(k_n, \tau_n, \xi_n, \theta_n)$ of Eq. (12) represents obtaining the tuning parameters (k_c, τ_i, and τ_d) of the PIDA controller by applying the ITAE-2 PID tuning rule to the new second order plus time delay model of Eq. (9). The explicit algebraic equations in Table 1 are fitted to the optimal data sets of f_d and f_a.

Putting it all together, the proposed PIDA tuning parameters can be summarized by the following procedure.

Step 1: Assume that the second-order plus time delay model of Eq. (8) is given, or it is obtained by applying the model reduction

Table 1. f_d and f_a of the proposed tuning rule for the second-order plus time delay model

$$\left.\begin{aligned}
f_d&=0.3425+0.0875\xi_m-0.12\frac{\theta_m}{\tau_m}\\
f_a&=0.273+(0.105+0.949\xi_m)\left(\frac{\theta_m}{\tau_m}+0.149\right)\ \text{for}\ \frac{\theta_m}{\tau_m}\le 0.37\\
&=0.273+(0.024+0.216\xi_m)\left(\frac{\theta_m}{\tau_m}+1.908\right)\ \text{for}\ 0.37<\frac{\theta_m}{\tau_m}\le 1.0\\
&=0.273+(0.009+0.081\xi_m)\left(\frac{\theta_m}{\tau_m}+6.407\right)\ \text{for}\ \frac{\theta_m}{\tau_m}>1.0
\end{aligned}\right\}\ \text{for}\ \xi_m\le 1.0$$

$$\left.\begin{aligned}
f_d&=0.5175-0.0875\xi_m-0.12\frac{\theta_m}{\tau_m}\\
f_a&=-0.768+(0.966+0.088\xi_m)\left(\frac{\theta_m}{\tau_m}+1.136\right)\ \text{for}\ \frac{\theta_m}{\tau_m}\le 0.37\\
&=-0.768+(0.22+0.02\xi_m)\left(\frac{\theta_m}{\tau_m}+6.255\right)\ \text{for}\ 0.37<\frac{\theta_m}{\tau_m}\le 1.0\\
&=-0.768+(0.0825+0.0075\xi_m)\left(\frac{\theta_m}{\tau_m}+18.0\right)\ \text{for}\ \frac{\theta_m}{\tau_m}>1.0
\end{aligned}\right\}\ \text{for}\ \xi_m>1.0$$

Table 2. ITAE-2 tuning rule for the PID controller for the setpoint change [11]

$$k_n k_c = -0.04 + \left\{0.333 + 0.949\left(\frac{\theta_n}{\tau_n}\right)^{-0.983}\right\}\xi_n \text{ for } \xi_n \le 0.9$$

$$k_n k_c = -0.544 + 0.308\left(\frac{\theta_n}{\tau_n}\right) + 1.408\,\xi_n\left(\frac{\theta_n}{\tau_n}\right)^{-0.832} \text{ for } \xi_n > 0.9$$

$$\frac{\tau_i}{\tau_n} = \left\{2.055 + 0.072\left(\frac{\theta_n}{\tau_n}\right)\right\}\xi_n \text{ for } \frac{\theta_n}{\tau_n} \le 1$$

$$\frac{\tau_i}{\tau_n} = \left\{1.768 + 0.329\left(\frac{\theta_n}{\tau_n}\right)\right\}\xi_n \text{ for } \frac{\theta_n}{\tau_n} > 1$$

$$\frac{\tau_d}{\tau_n} = \left\{1 - \exp\left(-\frac{(\theta_n/\tau_n)^{1.06}\xi_n}{0.87}\right)\right\}\left\{0.55 + 1.683\left(\frac{\theta_n}{\tau_n}\right)^{-1.09}\right\}$$

method [10] to the given high-order plus time delay model.

Step 2: Calculate f_d and f_a using Table 1.

Step 3: Obtain the new second-order plus time delay model (k_n, τ_n, ξ_n, and θ_n) using Eq. (9)

Step 4: Calculate k_c, τ_i, τ_d using the ITAE-2 PID tuning rule in Tables 2-3 [11] for the new second-order plus time delay model obtained in Step 3.

Step 5: Calculate τ_a using $\tau_a = \tau_d f_a$.

Fig. 1 shows typical control performance of the PIDA controllers tuned by the proposed tuning rule (PIDA-proposed) and tuned by the optimal tuning parameters (PIDA-optimal) and the PID controller tuned by the ITAE-2 tuning rule (PID-ITAE2). Here, the optimal tuning parameters are obtained by solving the following nonlinear optimization problem of which the cost function is the ITAE performance criterion:

$$\min_{k_c,\ \tau_i,\ \tau_d,\ \tau_a}\left\{\int_0^\infty t|e(t)|dt\right\} \tag{14}$$

subject to

$$\frac{y(s)}{u(s)} = \frac{k_m \exp(-\theta_m s)}{\tau_m^2 s^2 + 2\tau_m \xi_m s + 1} \tag{15}$$

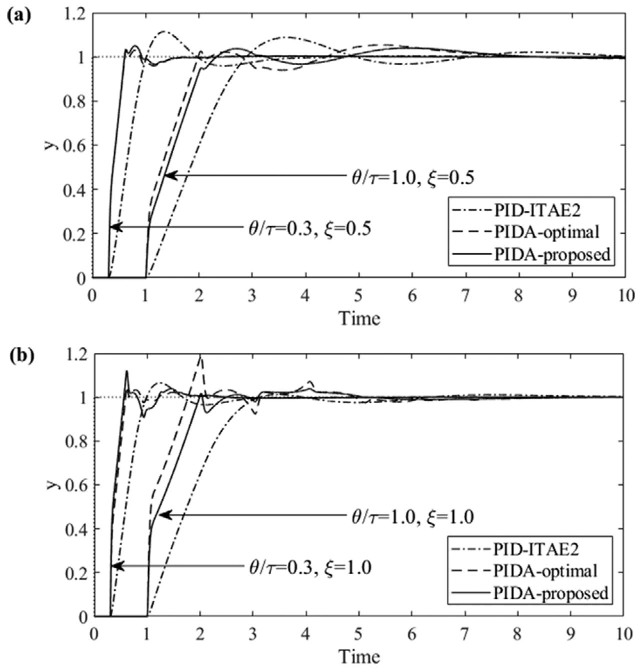

Fig. 1. Representative tuning results for the second-order plus time delay model: (a) with ξ_m=0.5 and (b) with ξ_m=1.0.

$$\frac{u(s)}{e(s)} = k_c\left(1 + \frac{1}{\tau_i s} + \frac{\tau_d}{\alpha_d \tau_d s + 1}s + \frac{\tau_a^2}{2(\alpha_a \tau_a s + 1)^2}s^2\right),$$

$$e(s) = y_s(s) - y(s),\ y_s(s) = \frac{1}{s} \tag{16}$$

As shown in Fig, 1, the proposed PIDA controller clearly shows much better control performance compared to the PID controller. Moreover, the control performance of the proposed tuning rule is close to the optimal PIDA controller with providing a smaller overshoot, demonstrating that the proposed tuning rule composed of explicit equations can provide remarkable performance without solving a nonlinear optimization problem or iterative computations.

SIMULATION STUDY

To confirm the performance of the proposed form of PIDA controller and its tuning rule, the following three different types of the

Table 3. ITAE-2 tuning rule for the PID controller for the disturbance rejection [11]

$$k_n k_c = -0.67 + 0.297\left(\frac{\theta_n}{\tau_n}\right)^{-2.001} + 2.189\,\xi_n\left(\frac{\theta_n}{\tau_n}\right)^{-0.766} \text{ for } \xi_n \le 0.9$$

$$k_n k_c = -0.365 + 0.26\left(\frac{\theta_n}{\tau_n} - 1.4\right)^2 + 2.189\,\xi_n\left(\frac{\theta_n}{\tau_n}\right)^{-0.766} \text{ for } \xi_n > 0.9$$

$$\frac{\tau_i}{\tau_n} = 2.2122\left(\frac{\theta}{\tau_n}\right)^{0.52} - 0.3 \text{ for } \frac{\theta}{\tau_n} < 0.4$$

$$\frac{\tau_i}{\tau_n} = -0.975 + 0.91\left(\frac{\theta_n}{\tau_n} - 1.845\right)^2 + \left\{1 - \exp\left[\frac{\xi_n}{0.15 + 0.33(\theta_n/\tau_n)}\right]\right\}\left\{5.25 - 0.88\left(\frac{\theta_n}{\tau_n} - 2.8\right)^2\right\} \text{ for } \frac{\theta_n}{\tau_n} \ge 0.4$$

$$\frac{\tau_d}{\tau_n} = -1.9 + 1.576\left(\frac{\theta_n}{\tau_n}\right)^{-0.53} + \left\{1 - \exp\left[\frac{\xi_n}{-0.15 + 0.939(\theta_n/\tau_n)^{-1.121}}\right]\right\}\left\{1.45 + 0.969\left(\frac{\theta_n}{\tau_n}\right)^{-1.171}\right\}$$

Table 4. Reduced model and tuning parameters for Example 1

Method	Reduced model			Set-point change				Disturbance rejection			
	τ	θ	ξ	k_c	τ_i	τ_d	τ_a	k_c	τ_i	τ_d	τ_a
IMC	3.589	1.254	-	2.690	4.216	0.534	-	2.690	4.216	0.534	-
ITAE-2	1.490	0.585	0.876	2.233	2.718	0.917	-	5.180	1.580	0.761	-
Proposed	1.490	0.585	0.876	3.778	2.900	0.940	0.718	10.87	1.167	0.593	0.453

Table 5. Reduced model and tuning parameters for Example 2

Method	Reduced model			Set-point change				Disturbance rejection			
	τ	θ	ξ	k_c	τ_i	τ_d	τ_a	k_c	τ_i	τ_d	τ_a
IMC	3.725	2.649	-	1.525	5.050	0.977	-	1.525	5.050	0.977	-
ITAE-2	2.046	1.556	0.863	1.312	3.726	1.385	-	2.174	3.069	1.448	-
Proposed	2.046	1.556	0.863	2.022	4.145	1.454	1.214	4.214	2.694	1.284	1.072

Table 6. Reduced model and tuning parameters for Example 3

Method	Reduced model			Set-point change				Disturbance rejection			
	τ	θ	ξ	k_c	τ_i	τ_d	τ_a	k_c	τ_i	τ_d	τ_a
IMC	3.540	4.038	-	1.101	5.559	1.286	-	1.101	5.559	1.286	-
ITAE-2	3.098	1.381	0.413	0.965	2.670	3.680	-	2.504	3.366	2.579	-
Proposed	3.098	1.381	0.413	1.465	2.847	3.933	2.123	5.503	2.798	1.857	1.002

processes were simulated.

Example 1: Third-order plus time delay process

$$G(s) = \frac{\exp(-0.2s)}{(s+1)^3} \tag{17}$$

Example 2: High-order process

$$G(s) = \frac{1}{(s+1)^5} \tag{18}$$

Example 3: Underdamped process with time delay

$$G(s) = \frac{\exp(-0.5s)}{(9s^2+2.4s+1)(s+1)} \tag{19}$$

In the present study, the internal model control (IMC) tuning rule [9], ITAE-2 tuning rule [11], and the proposed tuning rule are compared for the servo and regulatory problems. To obtain the controller parameters of each method, process models are reduced to the first order plus time delay models needed for the IMC tuning rule and the second order plus time delay model required for the ITAE-2 and the proposed method. The reduced models and tuning parameters for the servo problem and regulatory problem are enumerated in Tables 4-6, and simulation results of the control performance are depicted in Figs. 2-4.

As represented in Figs. 2(a)-4(a), the proposed method shows overwhelming performance (shorter rising time, smaller overshoot as well as shorter settling time) than previous tuning rules in all processes for the servo control problem. The integral time absolute error (ITAE) values of setpoint change/disturbance rejection for each method are: Example 1 of IMC: 4.57/7.79, ITAE2: 2.61/1.44, and proposed: 1.37/1.02; Example 2 of IMC: 19.18/29.08, ITAE2: 13.05/13.37, and proposed: 8.37/8.16; Example 3 of IMC: 22.22/18.74, ITAE2: 12.26/1.90, and proposed: 5.65/1.45. The IMC tun-

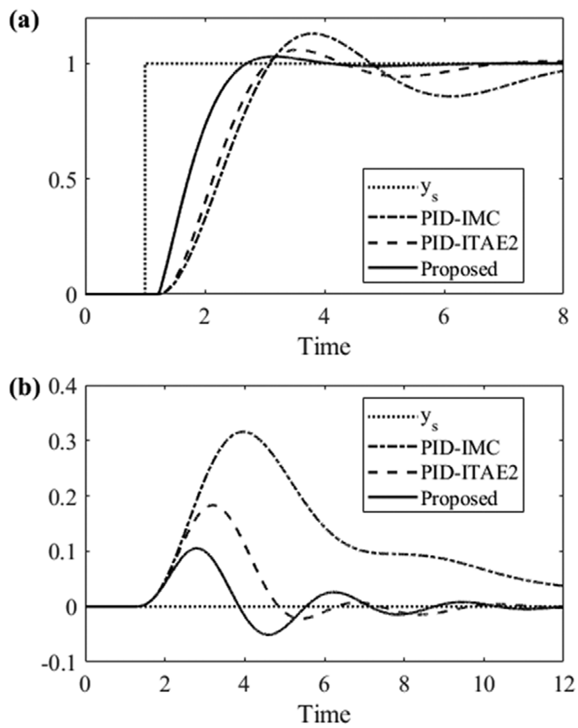

Fig. 2. Control performance of the proposed tuning rule and the previous tuning rules for Example 1: (a) the servo control problem and (b) the regulatory control problem.

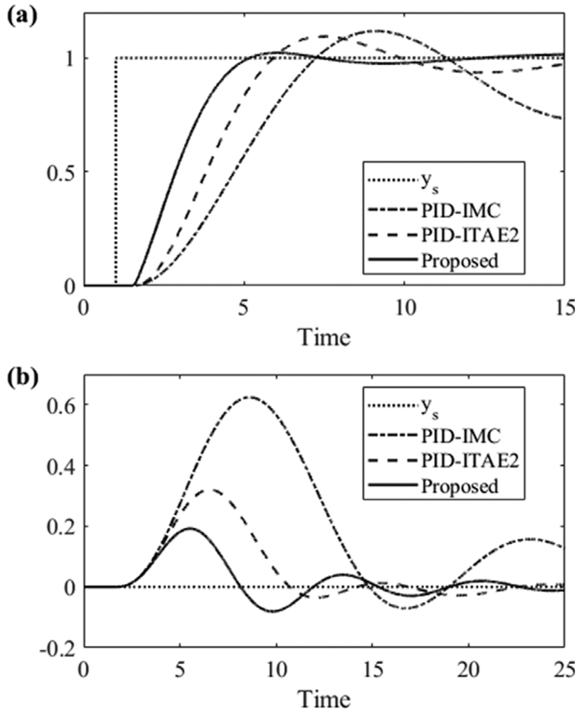

Fig. 3. Control performance of the proposed tuning rule and the previous tuning rules for Example 2: (a) the servo control problem and (b) the regulatory control problem.

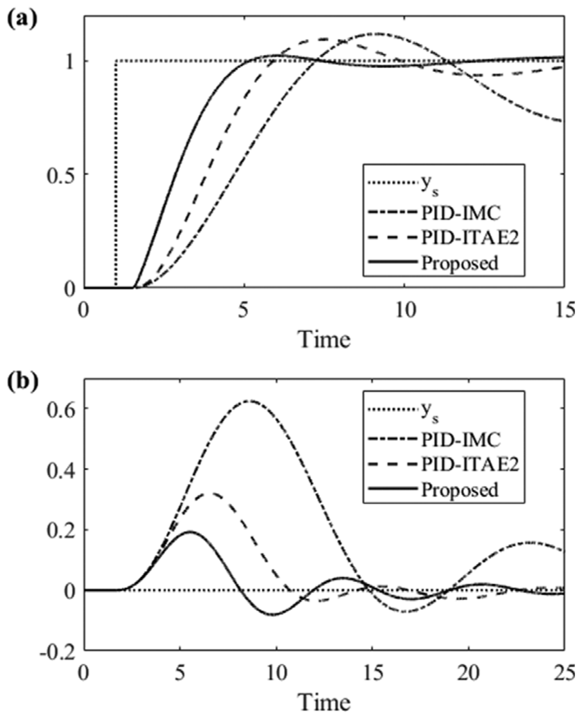

Fig. 4. Control performance of the proposed tuning rule and the previous tuning rules for Example 3: (a) the servo control problem and (b) the regulatory control problem.

ing rule provides poor control performance in Example 3 due to unavoidable serious structural mismatches in reducing the underdamped third-order plus time delay process of Eq. (18) to the first-order plus time delay model. For the regulatory control problem shown in Figs. 2(b)-4(b), the peak value of the process output in the case of the proposed method is much smaller than those of the ITAE-2 and IMC tuning rule in all processes, confirming that the proposed method eliminates the input disturbance more effectively than do the previous methods.

CONCLUSION

A new, non-interactive form of the proportional-integral-derivative-acceleration (PIDA) controller is proposed by analyzing the physical meaning of the error derivative term and the error acceleration term. Also, a new PIDA tuning rule was developed by combining the previous ITAE-2 tuning rule for the PID controller with the fact that the acceleration term of the PIDA controller reduces the time delay of the process. The proposed tuning rule composed of explicit algebraic equations provides excellent control performance without solving any nonlinear optimization problem or iterative computation. Simulation studies confirm that the proposed method provides superior control performance compared to the previous approaches with shorter rising time, smaller overshoot as well as shorter settling time for the servo control problem, and smaller peak for the regulatory control problem.

REFERENCES

1. C. Kravaris and I. K. Kookos, *Understanding process dynamics and control*, Cambridge University Press (2021).
2. S. W. Sung, J. Lee and I.-B. Lee, *Process identification and PID control*, John Wiley & Sons (2009).
3. L. Fan and E. M. Joo, *Design for auto-tuning PID controller based on genetic algorithms*, in: 2009 4th IEEE Conf. Ind. Electron. Appl., IEEE, 1924 (2009).
4. A. Zribi, M. Chtourou and M. Djemel, *J. Circuits, Syst. Comput.*, **27**, 1850065 (2018).
5. D. C. Meena and A. Devanshu, *Genetic algorithm tuned PID controller for process control*, in: 2017 Int. Conf. Inven. Syst. Control, IEEE, 1 (2017).
6. N. J. Killingsworth and M. Krstic, *IEEE Control Syst. Mag.*, **26**, 70 (2006).
7. M. I. Solihin, L. F. Tack and M. L. Kean, *Tuning of PID controller using particle swarm optimization (PSO)*, in: Proceeding Int. Conf. Adv. Sci. Eng. Inf. Technol., 458 (2011).
8. J. G. Ziegler and N. B. Nichols, Optimum settings for automatic controllers, Trans. ASME. 64 (1942).
9. M. Morari and E. Zafiriou, *Robust process control*, Prentice Hall (1989).
10. S. W. Sung, J. Lee and I.-B. Lee, *Process identification and PID control*, John Wiley & Sons (2009).
11. S. W. Sung, I.-B. Lee, J. Lee and S.-H. Yi, *J. Chem. Eng. Jpn.*, **29**, 990 (1996).
12. S. W. Sung and I.-B. Lee, *Ind. Eng. Chem. Res.*, **35**, 2596 (1996).
13. S. Skogestad, *J. Process Control.*, **13**, 291 (2003).
14. C. K. Yoo, H. J. Kwak and I.-B. Lee, *Chem. Eng. Res. Des.*, **79**, 754 (2001).

15. S. Jung and R. C. Dorf, *Analytic PIDA controller design technique for a third order system*, in: Proc. 35th IEEE Conf. Decis. Control, IEEE, 2513 (1996).
16. S. Buakaew, W. Narksarp, C. Wongtaychatham and W. Sangpisit, *PIDA controller realized on commercial IC current feedback operational amplifiers*, in: Proc. Int. MultiConference Eng. Comput. Sci. (2017).
17. M. Kumar and Y. V. Hote, *Robust IMC-PIDA controller design for load frequency control of a time delayed power system*, in: 2019 IEEE 58th Conf. Decis. Control, IEEE, 8380 (2019).
18. D. Puangdownreong, *ICA*, **3**, 303 (2012).
19. D. K. Sambariya and D. Paliwal, *Optimal design of PIDA controller using harmony search algorithm for AVR power system*, in: 2016 IEEE 6th Int. Conf. Power Syst., IEEE, 1 (2016).
20. M. Huba and D. Vrančič, *IFAC-PapersOnLine*, **51**, 954 (2018).
21. M. A. Sahib, *Eng. Sci. Technol. an Int. J.*, **18**, 194 (2015).
22. D. K. Sambariya and D. Paliwal, *Optimal design of PIDA controller using firefly algorithm for AVR power system*, in: 2016 Int. Conf. Comput. Commun. Autom., IEEE, 987 (2016).
23. K. Donuk, N. Özbey, M. İnan, C. Yeroğlu and D. Hanbay, *Investigation of PIDA controller parameters via PSO algorithm*, in: 2018 Int. Conf. Artif. Intell. Data Process., IEEE, 1 (2018).
24. A. M. Mosaad, M. A. Attia and A. Y. Abdelaziz, *Ain Shams Eng. J.*, **10**, 755 (2019).
25. A. Sharma, H. Sharma, A. Bhargava and N. Sharma, *Int. J. Metaheuristics*, **5**, 278 (2016).