MODELING, IDENTIFICATION AND CONTROL, 2004, VOL. 25, NO. 2, 85–120

Simple analytic rules for model reduction and PID controller tuning*

SIGURD SKOGESTAD†

Keywords: *Process control; feedback control; IMC; PI-control; integrating process; time delay*

The aim of this paper is to present analytic rules for PID controller tuning that are simple and still result in good closed-loop behavior. The starting point has been the IMC-PID tuning rules that have achieved widespread industrial acceptance. The rule for the integral term has been modified to improve disturbance rejection for integrating processes. Furthermore, rather than deriving separate rules for each transfer function model, there is a just a single tuning rule for a first-order or second-order time delay model. Simple analytic rules for model reduction are presented to obtain a model in this form, including the 'half rule' for obtaining the effective time delay.

1. Introduction

Although the proportional-integral-derivative (PID) controller has only three parameters, it is not easy, without a systematic procedure, to find good values (settings) for them. In fact, a visit to a process plant will usually show that a large number of the PID controllers are poorly tuned. The tuning rules presented in this paper have developed mainly as a result of teaching this material, where there are several objectives:

1. The tuning rules should be well motivated, and preferably model-based and analytically derived.
2. They should be simple and easy to memorize.
3. They should work well on a wide range of processes.

In this paper a simple two-step procedure that satisfies these objectives is presented:

Step 1. Obtain a first- or second-order plus delay model. The effective delay in this model may be obtained using the proposed half-rule.

Step 2. Derive model-based controller settings. PI-settings result if we start from a first-order model, whereas PID-settings result from a second-order model.

There has been previous work along these lines, including the classical paper by Ziegler & Nichols (1942), the IMC PID-tuning paper by Rivera *et al.* (1986), and the closely related direct synthesis tuning rules in the book by Smith & Corripio (1985). The Ziegler–Nichols settings result in a very good disturbance response for integrating processes, but are otherwise known to result in rather aggressive settings (Tyreus & Luyben, 1992; Astrom & Hagglund, 1995), and also give poor performance

*Reprinted from *Journal of Process Control*, Vol. 13, pp. 291–309, Copyright 2003, with permission from Elsevier. Additional remarks from the author are given at the end of the paper.

†Department of Chemical Engineering, Norwegian University of Science and Technology, N-7491 Trondheim Norway, tel.: +47-7359-4154, email: skoge@chemeng.ntnu.no

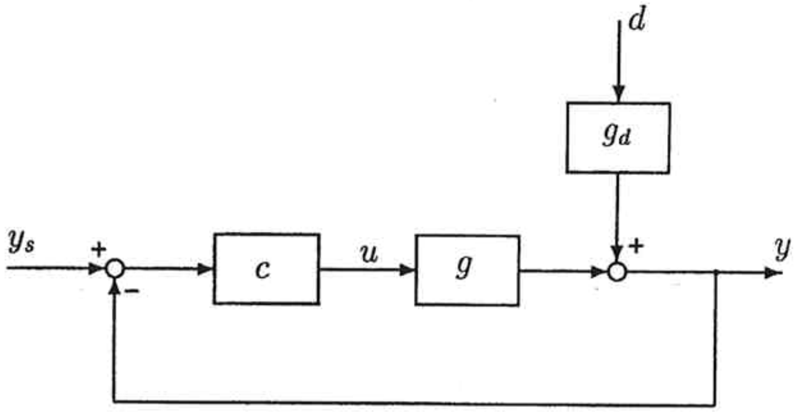

Figure 1. Block diagram of feedback control system. In this paper we consider an input ('load') disturbance ($g_d = g$).

for processes with a dominant delay. On the other hand, the analytically derived IMC-settings in Rivera *et al.* (1986) are known to result in a poor disturbance response for integrating processes (e.g. Chien & Fruehauf,1990; Horn *et al.*, 1996), but are robust and generally give very good responses for setpoint changes. The single tuning rule presented in this paper works well for both integrating and pure time delay processes, and for both setpoints and load disturbances.

1.1. *Notation*

The notation is summarized in Figure 1, where u is the manipulated input (controller output), d the disturbance, y the controlled output, and y_s the setpoint (reference) for the controlled output. $g(s) = (\Delta y/\Delta u)$ denotes the process transfer function and $c(s)$ is the feedback part of the controller. The Δ used to indicate deviation variables is deleted in the following. The Laplace variable s is often omitted to simplify notation. The settings given in this paper are for the series (cascade, 'interacting') form PID controller:

$$\text{Series PID: } c(s) = K_c \cdot \left(\frac{\tau_I s + 1}{\tau_I s}\right) \cdot (\tau_D s + 1) \tag{1}$$
$$= \frac{K_c}{\tau_I s}(\tau_I \tau_D s^2 + (\tau_I + \tau_D)s + 1)$$

where K_c is the controller gain, τ_I the integral time, and τ_D the derivative time. The reason for using the series form is that the PID rules with derivative action are then much simpler. The corresponding settings for the ideal (parallel form) PID controller are easily obtained using equation (36).

1.2. *Simulations*

The following series form PID controller is used in all simulations and evaluations of performance:

$$u(s) = K_c\left(\frac{\tau_I s + 1}{\tau_I s}\right)\left(y_s(s) - \frac{\tau_D s + 1}{\tau_F s + 1} y(s)\right) \tag{2}$$

with $\tau_F = \alpha\tau_D$ and $\alpha = 0.01$ (the robustness margins have been computed with $\alpha = 0$). Note that we, in order to avoid 'derivative kick' do not differentiate the setpoint in equation (2). The value $\alpha = 0.01$ was chosen in order to not bias the results, but in practice (and especially for noisy processes) a larger value of α in the range 0.1–0.2 is normally used. In most cases we use PI-control, i.e. $\tau_D = 0$, and the above implementation issues and differences between series and ideal form do not apply. In the time domain the PI-controller becomes

$$u(t) = u_0 + K_c\left((by_s(t) - y(t)) + \frac{1}{\tau_I}\int_0^t (y_s(\tau) - y(\tau))\mathrm{d}\tau\right) \tag{3}$$

where we have used $b = 1$ for the proportional setpoint weight.

2. Model approximation (Step 1)

The first step in the proposed design procedure is to obtain from the original model $g_0(s)$ an approximate first- or second-order time delay model $g(s)$ in the form

$$\begin{aligned} g(s) &= \frac{k}{(\tau_1 s + 1)(\tau_2 s + 1)} e^{-\theta s} \\ &= \frac{k'}{(s + 1/\tau_1)(\tau_2 s + 1)} e^{-\theta s} \end{aligned} \tag{4}$$

Thus, we need to estimate the following model information (see Figure 2):

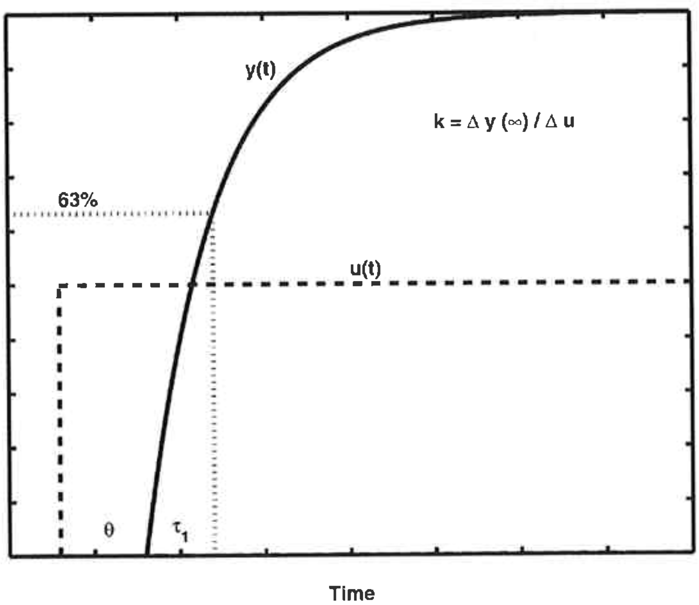

Figure 2. Step response of first-order plus time delay process, $g(s) = ke^{-\theta s}/(\tau_1 s + 1)$.

- Plant gain, k
- Dominant lag time constant, τ_1
- (Effective) time delay (dead time), θ
- Optional: Second-order lag time constant, τ_2 (for dominant second-order process for which $\tau_2 > \theta$, approximately)

If the response is lag-dominant, i.e. if $\tau_1 > 8\theta$ approximately, then the individual values of the time constant τ_1 and the gain k may be difficult to obtain, but at the same time are not very important for controller design. Lag-dominant processes may instead be approximated by an integrating process, using

$$\frac{k}{\tau_1 s+1} \approx \frac{k}{\tau_1 s} = \frac{k'}{s} \tag{5}$$

which is exact when $\tau_1 \to \infty$ or $1/\tau_1 \to 0$. In this case we need to obtain the value for the

$$\text{Slope, } k' \stackrel{\text{def}}{=} k/\tau_1$$

The problem of obtaining the effective delay θ (as well as the other model parameters) can be set up as a parameter estimation problem, for example, by making a least squares approximation of the open-loop step response. However, our goal is to use the resulting effective delay to obtain controller settings, so a better approach would be to find the approximation which for a given tuning method results in the best closed-loop response (here 'best' could, for example, be to minimize the integrated absolute error (IAE) with a specified value for the sensitivity peak, M_s). However, our main objective is not 'optimality' but 'simplicity', so we propose a much simpler approach as outlined next.

2.1. *Approximation of effective delay using the half rule*

We first consider the control-relevant approximation of the fast dynamic modes (high-frequency plant dynamics) by use of an effective delay. To derive these approximations, consider the following two first-order Taylor approximations of a time delay transfer function:

$$\mathrm{e}^{-\theta s} \approx 1-\theta s \quad \text{and} \quad \mathrm{e}^{-\theta s} = \frac{1}{\mathrm{e}^{\theta s}} \approx \frac{1}{1+\theta s} \tag{6}$$

From equation (6) we see that an 'inverse response time constant' T_0^{inv} (negative numerator time constant) may be approximated as a time delay:

$$(-T_0^{\text{inv}} s + 1) \approx \mathrm{e}^{-T_0^{\text{inv}} s} \tag{7}$$

This is reasonable since an inverse response has a deteriorating effect on control similar to that of a time delay (e.g. Skogestad & Postlethwaite, 1996). Similarly, from equation (6) a (small) lag time constant τ_0 may be approximated as a time delay:

$$\frac{1}{\tau_0 s+1} \approx \mathrm{e}^{-\tau_0 s} \tag{8}$$

Furthermore, since

$$\frac{-T_0^{\text{inv}} s+1}{\tau_0 s+1} \mathrm{e}^{-\theta_0 s} \approx \mathrm{e}^{-\theta_0 s} \cdot \mathrm{e}^{-T_0^{\text{inv}} s} \cdot \mathrm{e}^{-\tau_0 s} = \mathrm{e}^{-(\theta_0 + T_0^{\text{inv}} + \tau_0) s} = \mathrm{e}^{-\theta s}$$

it follows that the effective delay θ can be taken as the sum of the original delay θ_0, and the contribution from the various approximated terms. In addition, for digital implementation with sampling period h, the contribution to the effective delay is approximately $h/2$ (which is the average time it takes for the controller to respond to a change).

In terms of control, the lag-approximation equation (8) is conservative, since the effect of a delay on control performance is worse than that of a lag of equal magnitude (e.g. Skogestad & Postlethwaite, 1996). In particular, this applies when approximating the largest of the neglected lags. Thus, to be less conservative it is recommended to use the simple *half rule*:

- **Half rule:** The largest neglected (denominator) time constant (lag) is distributed evenly to the effective delay and the smallest retained time constant.

In summary, let the original model be in the form

$$\frac{\prod_j (-T_{j0}^{\mathrm{inv}} s+1)}{\prod_i \tau_{i0} s+1)} \mathrm{e}^{-\theta_0 s} \tag{9}$$

where the lags τ_{i0} are ordered according to their magnitude, and $T_{j0}^{\mathrm{inv}} > 0$ denote the inverse response (negative numerator) time constants. Then, according to the half-rule, to obtain a first-order model $\mathrm{e}^{-\theta s}/(\tau_1 s+1)$, we use

$$\tau_1 = \tau_{10} + \frac{\tau_{20}}{2}; \qquad \theta = \theta_0 + \frac{\tau_{20}}{2} + \sum_{i \geqslant 3} \tau_{i0} + \sum_j T_{j0}^{\mathrm{inv}} + \frac{h}{2} \tag{10}$$

and, to obtain a second-order model equation (4), we use

$$\tau_1 = \tau_{10}; \quad \tau_2 = \tau_{20} + \frac{\tau_{30}}{2}; \quad \theta = \theta_0 + \frac{\tau_{30}}{2} + \sum_{i \geqslant 4} \tau_{i0} + \sum_j T_{j0}^{\mathrm{inv}} + \frac{h}{2} \tag{11}$$

where h is the sampling period (for cases with digital implementation).

The main basis for the empirical half-rule is to maintain the robustness of the proposed PI- and PID-tuning rules, as is justified by the examples later.

Example E1. The process

$$g_0(s) = \frac{1}{(s+1)(0.2s+1)}$$

is approximated as a first-order time delay process, $g(s) = k\mathrm{e}^{-\theta s+1}/(\tau_1 s+1)$, with $k=1$, $\theta = 0.2/2 = 0.1$ and $\tau_1 = 1 + 0.2/2 = 1.1$.

2.2. *Approximation of positive numerator time constants*

We next consider how to get a model in the form of eqaution (9), if we have positive numerator time constants I_0 in the original model $g_0(s)$. It is proposed to cancel the numerator term $(T_0 s+1)$ against a 'neighbouring' denominator term $(\tau_0 s+1)$ (where both T_0 and τ_0 are positive and real) using the following approximations:

$$\frac{T_0 s+1}{\tau_0 s+1} \approx \begin{cases} T_0/\tau_0 & \text{for } T_0 \geqslant \tau_0 \geqslant \theta & \text{(Rule T1)} \\ T_0/\theta & \text{for } T_0 \geqslant \theta \geqslant \tau_0 & \text{(Rule T1a)} \\ 1 & \text{for } \theta \geqslant T_0 \geqslant \tau_0 & \text{(Rule T1b)} \\ T_0/\tau_0 & \text{for } \tau_0 \geqslant T_0 \geqslant 5\theta & \text{(Rule T2)} \\ \dfrac{(\tilde{\tau}_0/\tau_0)}{(\tilde{\tau}_0 - T_0)s+1} & \text{for } \tilde{\tau}_0 \stackrel{\text{def}}{=} \min(\tau_0, 5\theta) \geqslant T_0 & \text{(Rule T3)} \end{cases} \tag{12}$$

Here θ is the (final) effective delay, which exact value depends on the subsequent approximation of the time constants (half rule), so one may need to guess θ and iterate. If there is more than one positive numerator time constant, then one should approximate one T_0 at a time, starting with the largest T_0.

We normally select τ_0 as the closest *larger* denominator time constant ($\tau_0 > T_0$) and use Rules T2 or T3. The exception is if there exists no larger τ_0, or if there is smaller denominator time constant 'close to' T_0, in which case we select τ_0 as the closest *smaller* denominator time constant ($\tau_0 < T_0$) and use rules T1, T1a or T1b. To define 'close to' more precisely, let τ_{0a} (large) and τ_{0b} (small) denote the two neighboring denominator constants to τ_0. Then, we select $\tau_0 = \tau_{0b}$ (small) if $T_0/\tau_{0b} < \tau_{0a}/T_0$ and $T_0/\tau_{0b} < 1.6$ (both conditions must be satisfied).

Derivations of the above rules and additional examples are given in the Appendix.

Example E3. For the process (Example 4 in Astrom *et al.*, 1998)

$$g_0(s) = \frac{2(15s+1)}{(20s+1)(s+1)(0.1s+1)^2} \tag{13}$$

we first introduce from Rule T2 the approximation

$$\frac{15s+1}{20s+1} \approx \frac{15s}{20s} = 0.75$$

(Rule T2 applies since $T_0 = 15$ is larger than 5θ, where θ is computed below.) Using the half rule, the process may then be approximated as a first-order time delay model with

$$k = 2 \cdot 0.75 = 1.5; \qquad \theta = \frac{0.1}{2} + 0.1 = 0.15; \qquad \tau_1 = 1 + \frac{0.1}{2} = 1.05$$

or as a second-order time delay model with

$$k = 1.5; \qquad \theta = \frac{0.1}{2} = 0.05; \qquad \tau_1 = 1; \qquad \tau_2 = 0.1 + \frac{0.1}{2} = 0.15$$

Derivation of PID tuning rules (Step 2)

Direct synthesis (IMC tuning) for setpoints

Next, we derive for the model in equation (4) PI-settings or PID-settings using the method of direct synthesis for setpoints (Smith & Corripio, 1985), or equivalently the Internal Model Control approach for setpoints (Rivera *et al.*, 1986). For the system in Figure 1, the closed-loop setpoint response is

$$\frac{y}{y_s} = \frac{g(s)c(s)}{g(s)c(s)+1} \tag{14}$$

where we have assumed that the measurement of the output y is perfect. The idea of direct synthesis is to specify the desired closed-loop response and solve for the corresponding controller. From equation (14) we get

$$c(s) = \frac{1}{g(s)} \frac{1}{\frac{1}{(y/y_s)_{\text{desired}}} - 1} \tag{15}$$

We here consider the second-order time delay model $g(s)$ in equation (4), and specify that we, following the delay, desire a simple first-order response with time constant τ_c (Rivera *et al.*, 1986; Smith & Corripio, 1985):

$$\left(\frac{y}{y_s}\right)_{\text{desired}} = \frac{1}{\tau_c s + 1} e^{-\theta s} \tag{16}$$

We have kept the delay θ in the 'desired' response because it is unavoidable. Substituting equations (16) and (4) into equation (15) gives a 'Smith Predictor' controller (Smith, 1957):

$$c(s) = \frac{(\tau_1 s + 1)(\tau_2 s + 1)}{k} \frac{1}{(\tau_c s + 1 - e^{-\theta s})} \tag{17}$$

τ_c is the desired closed-loop time constant, and is the sole tuning parameter for the controller. Our objective is to derive PID settings, and to this effect we introduce in equation (17) a first-order Taylor series approximation of the delay, $e^{-\theta s} \approx 1 - \theta s$. This gives

$$c(s) = \frac{(\tau_1 s + 1)(\tau_2 s + 1)}{k} \frac{1}{(\tau_c + \theta)s} \tag{18}$$

which is a series form PID-controller equation (1) with Rivera *et al.* (1986); Smith & Corripio (1985)

$$K_c = \frac{1}{k} \frac{\tau_1}{\tau_c + \theta} = \frac{1}{k'} \frac{1}{(\tau_c + \theta)}; \qquad \tau_I = \tau_1; \qquad \tau_D = \tau_2 \tag{19}$$

3.2. *Modifying the integral time for improved disturbance rejection*

The PID-settings in equation (19) were derived by considering the *setpoint* response, and the result was that we should effectively cancel the first order dynamics of the process by selecting the integral time $\tau_I = \tau_1$. This is a robust setting which results in very good responses to setpoints and to disturbances entering directly at the process *output*. However, it is well known that for lag dominant processes with $\tau_1 \gg \theta$ (e.g. an integrating processes), the choice $\tau_I = \tau_1$ results in a long settling time for *input* (*'load'*) *disturbances* (Chien & Fruehauf, 1990). To improve the load disturbance response we need to reduce the integral time, but not by too much, because otherwise we get slow oscillations caused by having almost have two integrators in series (one from the controller and almost one from the slow lag dynamics in the process). This is illustrated in Figure 3, where we for the process

$$e^{-\theta s}/(\tau_1 s + 1) \quad \text{with} \quad \tau_1 = 30, \theta = 1$$

consider PI-control with $K_c = 15$ and four different values of the integral time:

- $\tau_I = \tau_1 = 30$ ('IMC-rule', see equation (19): excellent setpoint response, but slow settling for a load disturbance.

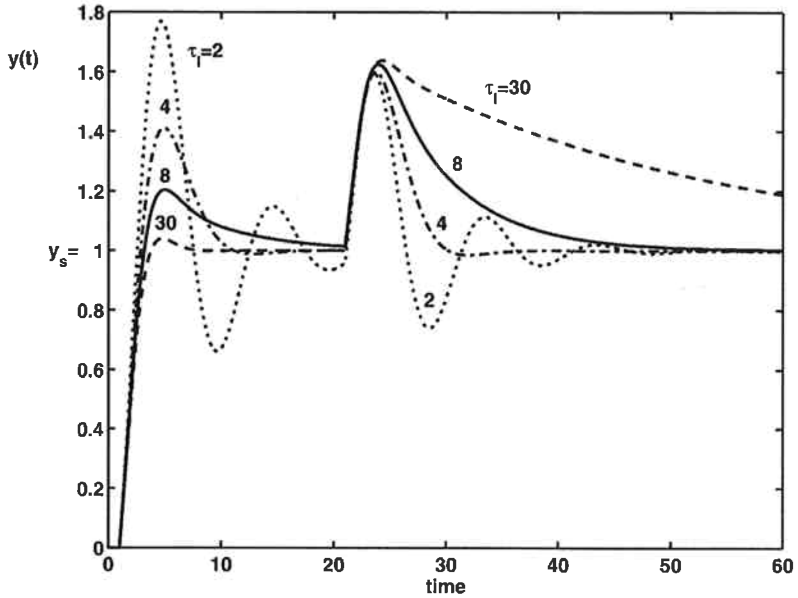

Figure 3. Effect of changing the integral time τ_I for PI-control of 'almost integrating' process $g(s) = e^{-s}/(30s+1)$ with $K_c = 15$. Unit setpoint change at $t = 0$; load disturbance of magnitude 10 at $t = 20$.

- $\tau_I = 8\theta = 8$ (SIMC-rule, see below): faster settling for a load disturbance.
- $\tau_I = 4$: even faster settling, but the setpoint response (and robustness) is poorer.
- $\tau_I = 2$: poor response with 'slow' oscillations.

A good trade-off between disturbance response and robustness is obtained by selecting the integral time such that we just avoid the slow oscillations, which corresponds to $\tau_I = 8\theta$ in the above example. Let us analyze this in more detail. First, note that these 'slow' oscillations are *not* caused by the delay θ (and occur at a lower frequency than the 'usual fast' oscillations which occur at about frequency $1/\theta$). Because of this, we neglect the delay in the model when we analyze the slow oscillations. The process model then becomes

$$g(s) = k\frac{e^{-\theta s}}{\tau_1 s + 1} \approx k\frac{1}{\tau_1 s + 1} \approx \frac{k}{\tau_1 s} = \frac{k'}{s}$$

where the second approximation applies since the resulting frequency of oscillations ω_0 is such that $(\tau_I\omega_0)^2$ is much larger than 1.[1] With a PI controller $c = K_c(1 + (1/\tau_I)s)$ the closed-loop characteristic polynomial $1 + gc$ then becomes

$$\frac{\tau_I}{k'K_C}s^2 + \tau_I s + 1$$

which is in standard second-order form, $\tau_0^2 s^2 + 2\tau_0\zeta s + 1$, with

[1] From equations (20) and (22) we get $\tau_0 = \tau_I/2$, so $\omega_0\tau_1 = (1/\tau_0)\tau_1 = 2(\tau_1/\tau_I)$. Here $\tau_1 \gg \tau_I$ and it follows that $\omega_0\tau_1 \gg 1$.

$$\tau_0 = \sqrt{\frac{\tau_I}{k'K_c}}; \qquad \zeta = \frac{1}{2}\sqrt{k'K_c\tau_I} \tag{20}$$

Oscillations occur for $\zeta < 1$. Of course, some oscillations may be tolerated, but a robust choice is to have $\zeta = 1$ (see also Marlin (1995), p. 588), or equivalently

$$K_c\tau_I = 4/k' \tag{21}$$

Inserting the recommended value for K_c from equation (19) then gives the following modified integral time for processes where the choice $\tau_I = \tau_1$ is too large:

$$\tau_I = 4(\tau_c + \theta) \tag{22}$$

3.3. *SIMC-PID tuning rules*

To summarize, the recommended SIMC PID settings[2] for the second-order time delay process in equation (4) are[3]

$$K_c = \frac{1}{k}\frac{\tau_1}{\tau_c + \theta} = \frac{1}{k'}\frac{1}{\tau_c + \theta} \tag{23}$$

$$\tau_I = \min\{\tau_1, 4(\tau_c + \theta)\} \tag{24}$$

$$\tau_D = \tau_2 \tag{25}$$

Here the desired first-order closed-loop response time τ_c is the only tuning parameter. Note that the same rules are used both for PI- and PID-settings, but the actual settings will differ. To get a PI-controller we start from a first-order model (with $\tau_2 = 0$), and to get a PID-controller we start from a second-order model. PID-control (with derivative action) is primarily recommended for processes with dominant second order dynamics (with $\tau_2 > \theta$, approximately), and we note that the derivative time is then selected so as to cancel the second-largest process time constant.

In Table 1 we summarize the resulting settings for a few special cases, including the pure time delay process, integrating process, and double integrating process. For the double integrating process, we let $\tau_2 \to \infty$ and introduce $k'' = k'/\tau_2$ and find (after some algebra) that the PID-controller for the integrating process with lag approaches a PD-controller with

$$K_c = \frac{1}{k''} \cdot \frac{1}{4(\tau_c + \theta)^2}; \qquad \tau_D = 4(\tau_c + \theta) \tag{26}$$

This controller gives good setpoint responses for the double integrating process, but results in steady-state offset for load disturbances occurring at the input. To remove this offset, we need to reintroduce integral action, and as before propose to use

$$\tau_I = 4(\tau_c + \theta) \tag{27}$$

It should be noted that derivative action is required to stabilize a double integrating process if we have integral action in the controller.

[2] Here SIMC means 'Simple control' or 'Skogestad IMC'.

[3] The derivative time in equation (25) is for the series form PID-controller in equation (1).

Table 1. SIMC PID-settings (equations (23)–(25)) for some special cases of equation (4) (with τ_c as a tuning parameter)

Process	$g(s)$	K_c	τ_I	τ_D[d]
First-order	$k\dfrac{e^{-\theta s}}{(\tau_1 s+1)}$	$\dfrac{1}{k}\dfrac{\tau_1}{\tau_c+\theta}$	$\min\{\tau_1, 4(\tau_c+\theta)\}$	—
Second-order, equation (4)	$k\dfrac{e^{-\theta s}}{(\tau_1 s+1)(\tau_2 s+1)}$	$\dfrac{1}{k}\dfrac{\tau_1}{\tau_c+\theta}$	$\min\{\tau_1, 4(\tau_c+\theta)\}$	τ_2
Pure time delay[a]	$ke^{-\theta s}$	0	0[e]	—
Integrating[b]	$k'\dfrac{e^{-\theta s}}{s}$	$\dfrac{1}{k'}\cdot\dfrac{1}{(\tau_c+\theta)}$	$4(\tau_c+\theta)$	—
Integrating with lag	$k'\dfrac{e^{-\theta s}}{s(\tau_2 s+1)}$	$\dfrac{1}{k'}\cdot\dfrac{1}{(\tau_c+\theta)}$	$4(\tau_c+\theta)$	τ_2
Double integrating[c]	$k''\dfrac{e^{-\theta s}}{s^2}$	$\dfrac{1}{k'}\cdot\dfrac{1}{4(\tau_c+\theta)^2}$	$4(\tau_c+\theta)$	$4(\tau_c+\theta)$

[a] The pure time delay process is a special case of a first-order process with $\tau_1 = 0$.
[b] The integrating process is a special case of a first-order process with $\tau_1 \to \infty$.
[c] For the double integrating process, integral action has been added according to equation (1).
[d] The derivative time is for the series form PID controller in equation (27).
[e] Pure integral controller $c(s) = \dfrac{K_I}{s}$ with $K_I \overset{\text{def}}{=} \dfrac{K_c}{\tau_I} = \dfrac{1}{k(\tau_c+\theta)}$.

3.4. *Recommended choice for tuning parameter τ_c*

The value of the desired closed-loop time constant τ_c can be chosen freely, but from equation (23) we must have $-\theta < \tau_c < \infty$ to get a positive and nonzero controller gain. The optimal value of τ_c is determined by a trade-off between:

1. Fast speed of response and good disturbance rejection (favored by a small value of τ_c)
2. Stability, robustness and small input variation (favored by a large value of τ_c).

A good trade-off is obtained by choosing τ_c equal to the time delay:

$$\text{SIMC-rule for fast response with good robustness:} \quad \tau_c = \theta \tag{28}$$

This gives a reasonably fast response with moderate input usage and good robustness margins, and for the second-order time delay process in equation (4) results in the following SIMC-PID settings which may be easily memorized ($\tau_c = \theta$):

$$K_c = \frac{0.5}{k}\frac{\tau_1}{\theta} = \frac{0.5}{k'}\frac{1}{\theta} \tag{29}$$

$$\tau_I = \min\{\tau_1, 8\theta\} \tag{30}$$

$$\tau_D = \tau_2 \tag{31}$$

The corresponding settings for the ideal PID-controller are given in equations (37) and (38).

Table 2. Robustness margins for first-order and integrating time delay process using the SIMC-settings in equations (29) and (30) ($\tau_c = \theta$)

Process $g(s)$	$\frac{k}{\tau_1 s + 1} e^{-\theta s}$	$\frac{k'}{s} e^{-\theta s}$
Controller gain, K_c	$\frac{0.5}{k}\frac{\tau_1}{\theta}$	$\frac{0.5}{k'}\frac{1}{\theta}$
Integral time, τ_I	τ_1	8θ
Gain margin (GM)	3.14	2.96
Phase margin (PM)	61.4°	46.9°
Sensitivity peak, M_s	1.59	1.70
Complementary sensitivity peak, M_t	1.00	1.30
Phase crossover frequency, $\omega_{180} \cdot \theta$	1.57	1.49
Gain crossover frequency, $\omega_c \cdot \theta$	0.50	0.51
Allowed time delay error, $\Delta\theta/\theta$	2.14	1.59

The same margins apply to a second-order process (equation (4)) if we choose $\tau_D = \tau_2$, see equation (31).

4. Evaluation of the proposed tuning rules

In this section we evaluate the proposed SIMC PID tuning rules in equations (23)–(31) with the choice $\tau_c = \theta$. We first consider processes that already are in the second-order plus delay form in equation (4). In Section 4.2 we consider more complicated processes which must first be approximated as second-order plus delay processes (Step 1), before applying the tuning rules (Step 2).

4.1. *First- or second-order time delay processes*

4.1.1. *Robustness* The robustness margins with the SIMC PID-settings in equations (29)–(31), when applied to first- or second-order time delay processes, are always between the values given by the two columns in Table 2.

For processes with $\tau_1 \leqslant 8\theta$, for which we use $\tau_I = \tau_1$ (left column), the system always has a gain margin GM = 3.14 and phase margin PM = 61.4°, which is much better than the typical minimum requirements GM > 1.7 and PM > 30° (Seborg *et al.*, 1989). The sensitivity and complementary sensitivity peaks are $M_s = 1.59$ and $M_t = 1.00$ (here small values are desired with a typical upper bound of 2). The maximum allowed time delay error is $\Delta\theta/\theta = \text{PM[rad]}/(\omega_c \cdot \theta)$, which in this case gives $\Delta\theta/\theta = 2.14$ (i.e. the system goes unstable if the time delay is increased from θ to $(1 + 2.14)\theta = 3.14\theta$).

As expected, the robustness margins are somewhat poorer for lag-dominant processes with $\tau_1 > 8\theta$, where we in order to improve the disturbance response use $\tau_I = 8\theta$. Specifically, for the extreme case of an integrating process (right column) the suggested settings give GM = 2.96, PM = 46.9°, $M_s = 1.70$ and $M_t = 1.30$, and the maximum allowed time delay error is $\Delta\theta = 1.59\theta$.

Of the robustness measures listed above, we will in the following concentrate on M_s, which is the peak value as a function of frequency of the sensitivity function $S = 1/(1 + gc)$. Notice that $M_s < 1.7$ guarantees GM > 2.43 and PM > 34.2° (Rivera *et al.*, 1986).

4.1.2. *Performance.* To evaluate the closed-loop performance, we consider a unit step setpoint change ($y_s = 1$) and a unit step input (load) disturbance ($g_d = g$ and $d = 1$), and for each of the two consider the input and output performance:

4.1.2.1. *Output perfomance.* To evaluate the output control performance we compute the integrated absolute error (IAE) of the control error $e = y - y_s$.

$$\text{IAE} = \int_0^\infty |e(t)|\,dt$$

which should be as small as possible.

4.1.2.2. *Input performance.* To evaluate the manipulated input usage we compute the total variation (TV) of the input $u(t)$, which is sum of all its moves up and down. TV is a bit difficult to define compactly for a continuous signal, but if we discretize the input signal as a sequence, $[u_1, u_2, \ldots, u_i, \ldots]$, then

$$\text{TV} = \sum_{i=1}^{\infty} |u_{i+1} - u_i|$$

which should be as small as possible. The total variation is a good measure of the 'smoothness' of a signal.

In Table 3 we summarize the results with the choice $\tau_c = \theta$ for the following five first-order time delay processes:

Case 1. Pure time delay process
Case 2. Integrating process
Case 3. Integrating process with lag $\tau_2 = 4\theta$
Case 4. Double integrating process
Case 5. First-order process with $\tau_1 = 4\theta$

Note that the robustness margins fall within the limits given in Table 2, except for the double integrating process in case 4 where we, from equation (27), have added integral action and robustness is somewhat poorer.

4.1.2.3. *Setpoint change.* The simulated time responses for the five cases are shown in Figure 4. The setpoint responses are nice and smooth. For a unit setpoint change, the minimum achievable IAE-value for these time delay processes is IAE $= \theta$ (e.g. using a Smith Predictor controller equation (17) with $\tau_c = 0$). From Table 3 we see that with the proposed settings the actual IAE-setpoint-value varies between 2.17θ (for the first-order process) to 7.92θ (for the more difficult double integrating process).

To avoid 'derivative kick' on the input, we have chosen to follow industry practice and not differentiate the setpoint, see equation (2). This is the reason for the difference in the setpoint responses between cases 2 and 3, and also the reason for the somewhat sluggish setpoint response for the double integrating process in case 4. Note also that the setpoint response can always be modified by introducing a 'feedforward' filter on the setpoint or using $b \neq 1$ in equation (3).

4.1.2.4. *Load disturbance.* The load disturbance responses in Figure 4 are also nice and smooth, although a bit sluggish for the integrating and double integrating processes. In the last column in Table 3 we compare the achieved IAE-value with

Table 3. SIMC settings and performance summary for five different time delay processes ($\tau_c = \theta$)

Case	$g(s)$	K_c	τ_I	τ_D[c]	M_s	Setpoint[a] IAE(y)	Setpoint[a] TV(u)	Load disturbance IAE(y)	Load disturbance TV(u)	Load disturbance $\frac{\text{IAE}}{\text{IAEmin}}$
1	$ke^{-\theta s}$	0	0[d]	—	1.59	2.17θ	$1.08\frac{1}{k}$	$2.17\,k\theta$	1.08	1.59
2	$k'\frac{e^{-\theta s}}{s}$	$\frac{0.5}{k'}\cdot\frac{1}{\theta}$	8θ	—	1.70	3.92θ	$1.22\frac{1}{k'\theta}$	$16\,k'\theta^2$	1.55	3.27
3	$k'\frac{e^{-\theta s}}{s(4\theta s+1)}$	$\frac{0.5}{k'}\cdot\frac{1}{\theta}$	8θ	$\tau_2=4\theta$	1.70	5.28θ	$1.23\frac{1}{k'\theta}$	$16\,k'\theta^2$	1.59	5.41
4	$k''\frac{e^{-\theta s}}{s.^2}$	$\frac{0.0625}{k''}\cdot\frac{1}{\theta^2}$	8θ	8θ	1.96	7.92θ	$0.205\frac{1}{k''\theta^2}$	$128\,k''\theta^3$	2.34	5.49
5	$k\frac{e^{-\theta s}}{4\theta s+1}$	$\frac{0.5}{k}\frac{\tau_1}{\theta}=\frac{2}{k}$	$\tau_1=4\theta$	—	1.59	2.17θ	$4.11\frac{1}{k}$	$2\,k\theta$	1.08	2.41

[a] The IAE and TV-values for PID control are without derivative action on the setpoint.
[b] IAEmin is for the IAE-optimal PI/PID-controller of the same kind.
[c] The derivative time is for the series form PID controller in equation (1).
[d] Pure integral controller $c(s) = \frac{K_I}{s}$ with $K_I = \frac{K_c}{\tau_I} = \frac{0.5}{k\theta}$.

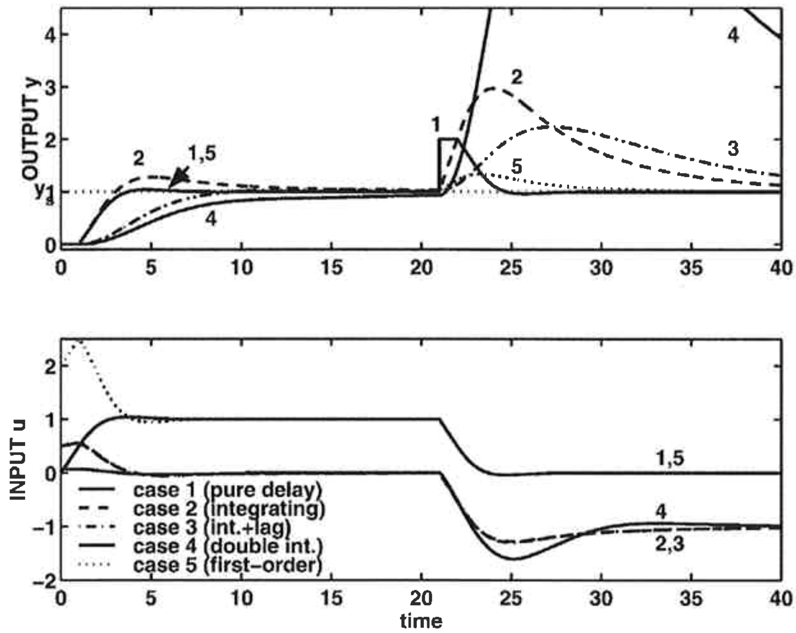

Figure 4. Responses using SIMC settings for the five time delay processes in Table 3 ($\tau_c = \theta$). Unit setpoint change at $t = 0$; Unit load disturbance at $t = 20$. Simulations are without derivative action on the setpoint. Parameter values: $\theta = 1$, $k = 1$, $k' = 1$, $k'' = 1$.

that for the IAE-optimal controller of the same kind (PI or series-PID). The ratio varies from 1.59 for the pure time delay process to 5.49 for the more difficult double integrating process.

However, lower IAE-values generally come at the expense of poorer robustness (larger value of M_s), more excessive input usage (larger value of TV), or a more complicated controller. For example, for the integrating process, the IAE-optimal PI-controller ($K_c = (0.91/k') \cdot (1/\theta)$, $\tau_I = 4.1\theta$) reduces IAE(load) by a factor 3.27, but the input variation increases from TV = 1.55 to TV = 3.79, and the sensitivity peak increases from $M_s = 1.70$ to $M_s = 3.71$. The IAE-optimal PID-controller ($K_c = (0.80/k') \cdot (1/\theta)$, $\tau_I = 1.26\theta$, $\tau_D = 0.76\theta$) reduces IAE(load) by a factor 8.2 (to IAE $= 1.95k'\theta^2$), but this controller has $M_s = 4.1$ and TV(load) = 5.34. The lowest achievable IAE-value for the integrating process is for an ideal Smith Predictor controller equation (17) with $\tau_c = 0$, which reduces IAE(load) by a factor 32 (to IAE $= 0.5k'\theta^2$). However, this controller is unrealizable with infinite input usage and requires a perfect model.

4.1.2.5. *Input usage.* As seen from the simulations in the lower part of Figure 4 the input usage with the proposed settings is very smooth in all cases. To have no steady-state offset for a load disturbance, the minimum achievable value is TV(load) = 1 (smooth input change with no overshoot), and we find that the achieved value ranges from 1.08 (first-order process), through 1.55 (integrating process) and up to 2.34 (double integrating process).