8 Process Reaction Curve and Relay Methods Identification and PID Tuning

Learning Objectives

The original methods arising from the process reaction curve approach avoided the explicit production of a process model and moved directly from some particular response measurements to a set of rules or formulae from which to compute the coefficients of the three terms in the PID controller. In this chapter, however, the PID controller tuning is based on the use of simple parametric models. Thus, first-order plus dead time (FOPDT) models or second-order plus dead time (SOPDT) models are identified using some characteristic measurements obtained from a transient step response test or from a relay feedback experiment. The design for the PI/PID controllers is then derived using these simple models and an internal model control (IMC) based methodology. This control design method uses an inverse process model and has degrees of freedom available which allow further control performance improvement. To complete the approach, the best achievable Integral of Absolute Error (IAE) cost function value and rise time of the system can be computed and used as a benchmark against which to assess the performance of the installed controller. Finally these methods all have the potential to be incorporated into a process controller unit to enhance the task of autotuning for the PID controllers.

The learning objectives for this chapter are to:

- Identify low-order process models from simple process reaction curve data.
- Use a relay feedback experiment to adaptively tune simple process models.
- Tune a PID controller using an inverse process model-based technique.
- Understand and use the idea of controller assessment based on classical performance indices.

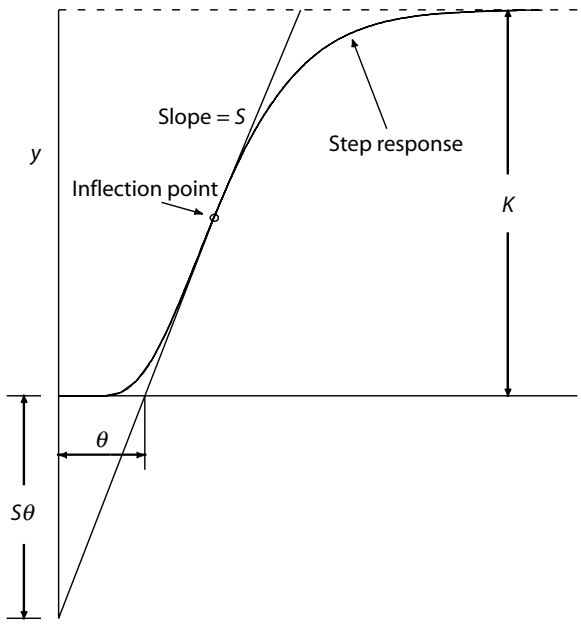
8.1 Introduction

In model-based controller design, simple models are used to characterise the dynamics of a given process. Based on the simple dynamic model obtained, the PID controller parameters are then computed. For example, in the parametric Ziegler–Nichols tuning method, a process model of the following form was assumed:

$$G_{\mathrm{p}}(s)=\left[\frac{K\mathrm{e}^{-\theta s}}{\tau s+1}\right]\approx\left[\frac{K\mathrm{e}^{-\theta s}}{\tau s}\right]=\left(\frac{K\theta}{\tau}\right)\left[\frac{\mathrm{e}^{-\theta s}}{\theta s}\right]$$

where the process gain, the time constant and the delay time are denoted by K, τ and θ respectively. Identification of this simple model structure used the step response of the process, as shown in Figure 8.1. In process engineering this type of system response is termed a process reaction curve or an S-curve, and this is the origin of the name for the family of tuning methods.

The process reaction curve is recorded and various response measurements made. As can be seen on Figure 8.1, the process gain K can be measured and the intersection of the tangent line with the baseline provides an estimate of the dead time θ. The slope of the tangent at the inflection point on the S-curve



Key
Process gain, K Delay time, θ Slope of tangent line, S

Figure 8.1 Process reaction curve of a system.

can also be measured. This measured slope has the theoretical formula $S = K\theta/\tau$. Although the PID tuning is based on an inherent process model with parameters K, τ, θ, the PID controller coefficients are computed from formula given in terms of the measured values of K, S, θ. Modelling for controller design in this way is only approximate due to the need for a tangent line at the inflection point of the S-curve.

There are also quite a few methods in the literature which propose to model the S-shape step response curve with the so called first-order plus dead time (FOPDT) models or alternatively, overdamped second-order plus dead time (SOPDT) models. Parametric methods to tune the PID controllers have been presented in many different ways for the FOPDT model given by

$$G_p(s) = \left[\frac{Ke^{-\theta s}}{\tau s + 1}\right]$$

For example, tuning rules to minimise an integral performance index based on the FOPDT dynamics have been given in the forms

$$P = a\left(\frac{\theta}{\tau}\right)^b \text{ and } P = a + b\left(\frac{\theta}{\tau}\right)^c$$

where P represents a controller parameter (such as k_p, τ_i and τ_d) to be tuned for a PID controller, and a, b and c are regression constants. Furthermore, since the 1980s, Internal Model Control (IMC) theory has been used to formulate PID controllers, and as a result the need for simple models is even more important.

There are a few methods reported in the literature for identifying simple SOPDT models. In an early work of Oldenbourg and Sartorius (1948), the model being identified is

$$G_p(s) = \left[\frac{Ke^{-\theta s}}{(\tau_1 s + 1)(\tau_2 s + 1)}\right]$$

where the process gain, the two process time constants and the delay time are denoted by K, τ_1, τ_2, θ, respectively. From the response as shown in Figure 8.2, the initial delay time θ and the lengths of segments I_1 and I_2 are measured.

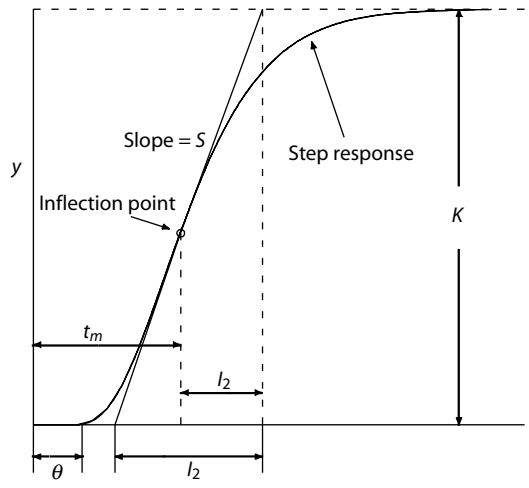


Figure 8.2 Process step response of a system.

Mathematically, the value of τ_1/I_1 is related to τ_2/I_2 by the following two equations:

$$\tau_1 I_1 = \left(\frac{\tau_2}{\tau_1}\right)^{\tau_2/(\tau_1-\tau_2)} = x^{x/(1-x)} \quad \text{and} \quad \frac{\tau_1}{I_1}x = \frac{\tau_2}{I_1}$$

where $x = \tau_2/\tau_1$.

On the other hand, the point $(\tau_1/I_1, \tau_2/I_1)$ also satisfies the relation

$$\frac{\tau_1}{I_1} + \frac{\tau_2}{I_2} = \frac{I_2}{I_1}$$

Consequently, based on the above equations and the measured segments, the two time constants can be found.

In an alternative method, Sundaresan *et al.* (1978) derived the following relation:

$$\lambda = (t_m - m)S = \chi e^{-\chi}, \text{ where } \chi = \frac{\ln \eta}{\eta - 1} \text{ and } \eta = \frac{\tau_2}{\tau_1} \leq 1.$$

The value of t_m is taken as the time at the inflection point of the response, and the quantity m is computed from the following integral:

$$m = \int_0^\infty [y(t_\infty) - y(t)]\mathrm{d}t$$

As the input is a unit step change, $y(t_\infty)$ equals K (see Figure 8.2). A plot of η versus λ can thus be prepared. Hence by calculating the value of λ from the response, the value of η can be calculated. Then the two time constants and the dead time can be obtained from the following formulae:

$$\tau_1 = \eta^{1(1-\eta)}/S, \tau_2 = \eta \cdot \tau_1 \text{ and } \theta = m - \tau_1 - \tau_2$$

where, as previously mentioned, S is the slope of the tangent at the inflection point of the response.

Another method for identifying simple overdamped SOPDT models is found in an early textbook due to Harriott (1964). Harriott's method is based on two observations from a class of step responses of the process represented by the transfer function

$$\frac{1}{(\tau_1 s + 1)(\eta\tau_1 s + 1)}, \quad \eta \in [0,1]$$

First, it is found that almost all the step responses mentioned reach 73% of the final steady state approximately at a time of 1.3 times $\tau_1 + \tau_2$. For easy reference, this response position and time are designated as y_{73} and t_{73}, respectively, as shown in Figure 8.3(a). Second, at a time of $0.5(\tau_1 + \tau_2)$, all the step responses separate from each other most widely. Again, for easy reference, the time at $0.5(\tau_1 + \tau_2)$ is designated as $t_\star$, and the corresponding response position is denoted as y^*. It is found that y^* is a function of $\tau_1/(\tau_1 + \tau_2)$ as shown in Figure 8.3(b). In other words, from the experimental response, the position of y_{73} is identified and the corresponding time t_{73} is recorded. Upon having a value for t_{73}, the value of $t_\star$ can be calculated. From this value of $t_\star$, the position of y^* can be read from the step response again. When y^* is found, the value of $\tau_1/(\tau_1 + \tau_2)$ can be obtained by making use of Figure 8.3(b), and the two time constants can be computed.

The method as given above was originally formulated for systems without delay time, but it can be easily extended to include the estimation of delay time by using the relationship for the area m given as

$$m = \tau_1 + \tau_2 + \theta$$

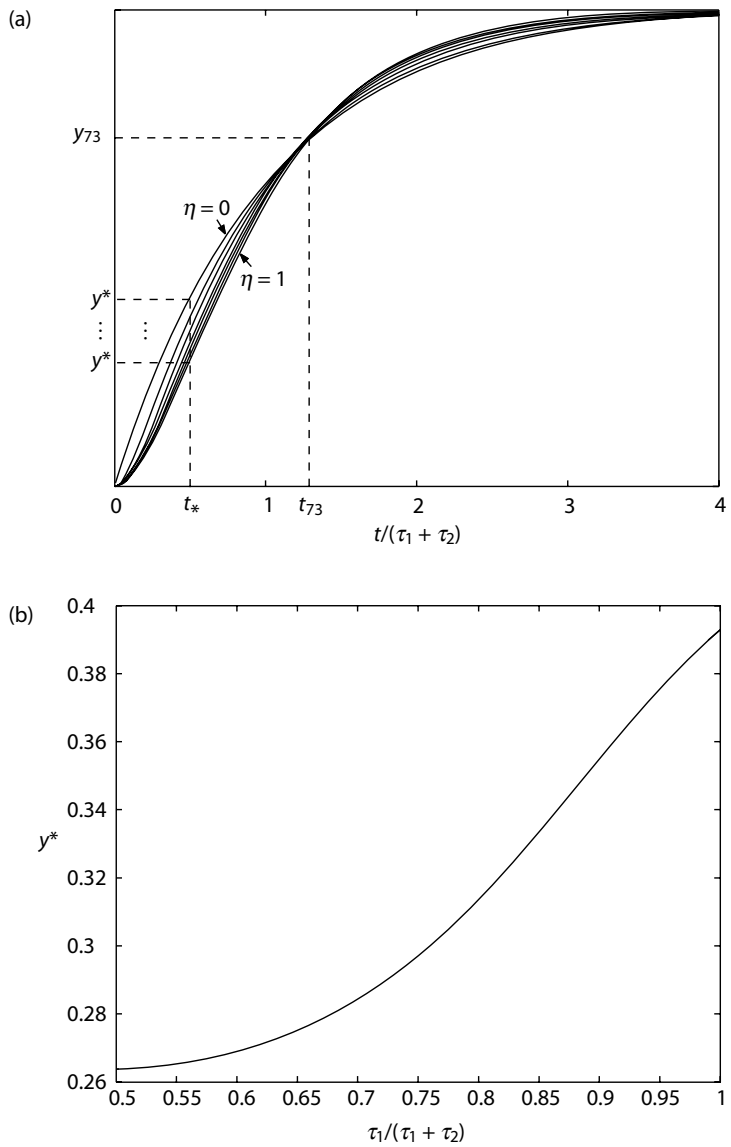


Figure 8.3 (a) Step responses of the processes; (b) relation between y^* and $\tau_1/(\tau_1 + \tau_2)$.

Thus, the model parameters can be estimated from the following equation set:

$$\theta = m - \frac{t_{73}}{1.3}$$

$$\tau_1 = \left(\frac{\tau_1}{\tau_1+\tau_2}\right)_{\text{at } y^*} \times \left(\frac{t_{73}-t_0}{1.3}\right)$$

$$\tau_2 = \left\{1-\left(\frac{\tau_1}{\tau_1+\tau_2}\right)_{\text{at } y^*}\right\} \times \left(\frac{t_{73}-t_0}{1.3}\right)$$

where t_0 is the time at which the process begins to respond initially.

All of the above methods for developing SOPDT models are characterised by the use of data from the step responses instead of using whole sets of time series data of both input and output as might typically occur for classical parameter estimation procedures. In this chapter, the focus will be on developing a systematic procedure to identify these simple models from transient step response curves or from relay feedback experiments. Following this, PID controllers will be synthesised using these simple models and a proposed inverse-based method. This methodology will not resort to rule-based methods based on optimisation procedures.

8.2 Developing Simple Models from the Process Reaction Curve

In this section, the identification of simple dynamic models using a process reaction curve is presented. A process reaction curve for identification is generated by disturbing the process with a manual step input, and recording the process variable (PV) output as shown in Figure 8.4. Thus the term "process reaction curve" is usually considered as being synonymous with the transient step response of an open-loop process, excluding the controller in a control loop. However, on a real plant the process controller would be put into manual mode and the step change generated from the controller unit. The step change signal would be a change, say 10% of setpoint, added to the constant process input needed to hold the process at the desired production output level. In this way the disturbance to the production unit operation is minimised and controlled. The perceived effect is to record the transient step response of an open-loop process with the feedback loop broken, as shown in Figure 8.4.

The simple SOPDT models used for identification include the following forms

$$\frac{Y(s)}{U(s)} = G_p(s) = \begin{cases} \dfrac{K(1+as)e^{-\theta s}}{\tau^2 s^2 + 2\tau\zeta s + 1} & 0 < \zeta < 1 \quad \text{(model I)} \\ \dfrac{K(1+as)e^{-\theta s}}{(\tau s+1)(\eta\tau s+1)} & 0 < \eta \le 1 \quad \text{(model II)} \end{cases}$$

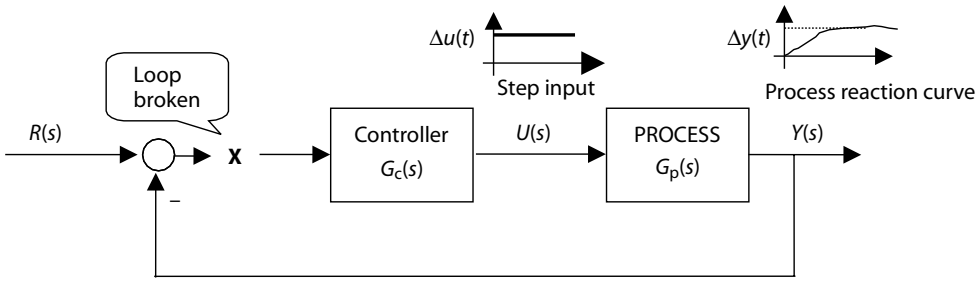


Figure 8.4 Generating the process reaction curve.

where model I is used for the underdamped second-order case with $\zeta < 1$ and model II is used for the overdamped second-order case with $\zeta \geq 1$.

In general, it is assumed that a process with high-order dynamics can be represented by the above models. For model-based controller design, models used are usually confined to the minimum phase case for which $a \geq 0$. Since the steady state gain K can be obtained easily by computing the ratio of changes of output Δy to the changes of input Δu, the identification will focus on the estimation of other parameters in the model.

The above simple SOPDT models for $G_p(s)$ can be changed into the dimensionless form by using the relations $\bar{s} = \tau s$ and $\bar{\theta} = \theta / \tau$. The dimensionless forms are given as

$$\frac{Y(s)}{KU(s)} = \frac{\bar{Y}(\bar{s})}{U(\bar{s})} = \bar{G}_p(\bar{s}) = \begin{cases} \dfrac{(1+\bar{a}\,\bar{s})e^{-\bar{\theta}\bar{s}}}{\bar{s}^2 + 2\zeta\,\bar{s} + 1} & 0 < \zeta < 1 \text{ (model I)} \\ \dfrac{(1+\bar{a}\,\bar{s})e^{-\bar{\theta}\bar{s}}}{(\bar{s}+1)(\eta\,\bar{s}+1)} & 0 < \eta \leq 1 \text{ (model II)} \end{cases}$$

where $\bar{Y} = Y/K$ and $\bar{a} = a/\tau$. The unit step response resulting from the dimensionless model above is given by the following equations.

Model I

$$\bar{y}(\bar{t}) = 1 - \left[\frac{\zeta - \bar{a}}{\sqrt{1-\zeta^2}} \sin(\sqrt{1-\zeta^2}\,\bar{t}) + \cos(\sqrt{1-\zeta^2}\,\bar{t}) \right] e^{-\zeta \bar{t}}$$

Model II

$$\bar{y}(\bar{t}) = 1 - \left(\frac{1-\bar{a}}{1-\eta} \right) e^{-\bar{t}} - \left(\frac{\eta - \bar{a}}{1-\eta} \right) e^{-\bar{t}/\eta}$$

where $\bar{t} = (t-\theta)/\tau$.

In the following, a method of identification will be presented based on the work of Huang *et al.* (2001).

8.2.1 Identification Algorithm for Oscillatory Step Responses

When the reaction curve is oscillatory (as shown in Figure 8.5), the dimensionless equation for model I will be adopted.

To distinguish between process models that do not have a process zero (where $\bar{a} = 0$) and those which do have a zero (where $\bar{a} \neq 0$) the following lemma is needed.

Lemma 8.1

Consider a system of type Model I having $\zeta < 1$ and $\bar{a} \in (-\infty, \infty)$. Let $t_{p,i}$ and $t_{m,i}$ be the time instants when the output $y(t)$ reaches its ith peak and ith valley, respectively. Then, if and only if $\bar{a} = 0$, the following relationship holds:

$$t_{p,1} - \theta = t_{p,i+1} - t_{m,i} = t_{m,i} - t_{p,i} = \frac{P}{2} = \frac{\pi\tau}{\sqrt{1-\zeta^2}} \qquad \forall i \geq 1$$

where P represents the oscillating period of the response, that is,

$$P = t_{p,i+1} - t_{p,i}, \quad \forall i \geq 1$$

Proof (Huang *et al.*, 2001) ❑

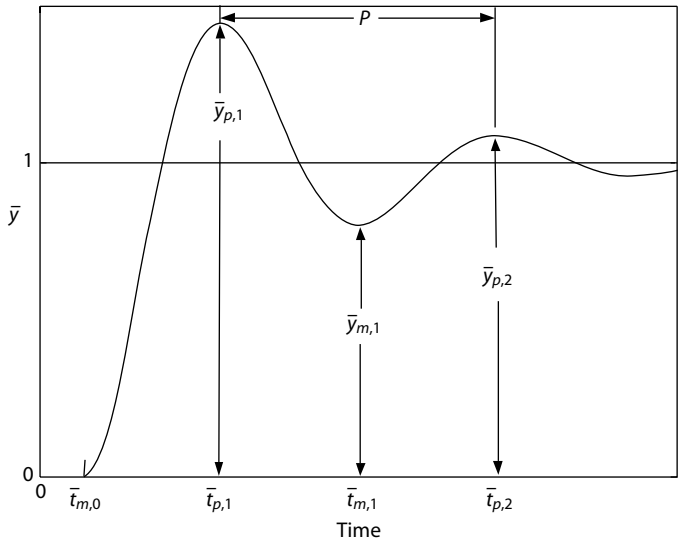


Figure 8.5 Step response for underdamped system with $\bar{a} \geq 0$.

Notice that in Figure 8.5 the time interval $t_{p,1} - \theta$ is denoted as $t_{p,1} - t_{m,0}$. Thus, the criterion to determine whether $\bar{a} = 0$ is simply to determine whether the time interval $t_{p,1} - t_{m,0}$ is equal to the time interval $t_{p,2} - t_{m,1}$.

Consequently, model parameters for model I processes can be estimated using the following algorithm.

Algorithm 8.1: SOPDT models for oscillatory step responses

Step 1 Complete the step response test, Measure P (see Figure 8.5).

Step 2 Compute the time constant τ using

$$\tau = \frac{P\sqrt{1-\zeta^2}}{2\pi} = \frac{P}{\sqrt{4\pi^2 + P^2x^2}}$$

where

$$x = \begin{cases} \dfrac{1}{t_{p,1} - t_{p,2}} \ln\left[\dfrac{\bar{y}_{p,2} - 1}{\bar{y}_{p,1} - 1}\right] & \forall \bar{a} \geq 0 \\ \dfrac{1}{t_{m,1} - t_{p,1}} \ln\left[\dfrac{\bar{y}_{p,1} - 1}{1 - \bar{y}_{m,1}}\right] & \forall \bar{a} < 0 \end{cases}$$

and

$$\bar{y}_{p,i} = \bar{y}(\bar{t}_{p,i}) \quad i = 1,2,\ldots$$

Step 3 Compute the damping ratio ζ using

$$\zeta = \begin{cases} \sqrt{\dfrac{\ln^2(\bar{y}_{p,1} - 1)}{\pi^2 + \ln^2(\bar{y}_{p,1} - 1)}} & \text{for } \bar{a} = 0 \\ \dfrac{Px}{\sqrt{4\pi^2 + P^2x^2}} & \forall \bar{a} \neq 0 \end{cases}$$

Step 4 Compute $\bar{a}$ and process delay time θ using
(a) for $\bar{a}=0$, compute $\theta = t_{p,1} + (\tau/\zeta)\ln(\bar{y}_{p,1}-1)$
(b) for $\bar{a}\neq 0, \bar{a}$ and θ are computed using

$$\bar{a} = \begin{cases} \zeta + \sqrt{\zeta^2 + \left[1 - \left(\dfrac{\bar{y}_{p,1}-1}{e^{-\zeta \bar{t}_{p,1}}}\right)^2\right]} & \forall \bar{a} > 0 \\ \zeta - \sqrt{\zeta^2 + \left[1 - \left(\dfrac{1-\bar{y}_{m,1}}{e^{-\zeta \bar{t}_{m,1}}}\right)^2\right]} & \forall \bar{a} < 0 \end{cases}$$

and

$$\theta = \begin{cases} t_{p,1} - \dfrac{P}{2\pi}\left[\pi - \tan^{-1}\dfrac{\bar{a}\sqrt{1-\zeta^2}}{1-\bar{a}\zeta}\right] & \forall \bar{a} > 0 \\ t_{m,1} + \dfrac{P}{2\pi}\left[\tan^{-1}\dfrac{\bar{a}\sqrt{1-\zeta^2}}{1-\bar{a}\zeta}\right] & \forall \bar{a} < 0 \end{cases}$$

Algorithm end

Example 8.1

Consider the following two processes:

Process (a) $$G_p(s) = \frac{e^{-s}}{(4s^2+2s+1)(s^2+s+1)}$$

Process (b) $$G_p(s) = \frac{(2s+1)e^{-2s}}{(4s^2+2s+1)(s^2+s+1)}$$

The step responses of these processes are as shown in Figure 8.6.

Both responses are oscillatory. It is found that only the response of process (a) has equal values for the time intervals $t_{p,2} - t_{m,1}$ and $t_{p,1} - t_{m,0}$. The identification results based on these observations are given in Table 8.1 and the unit step responses of the resulting identified models are also given in Figure 8.6.

8.2.2 Identification Algorithm for Non-Oscillatory Responses Without Overshoot

In this section the identification is presented for a process which has all the characteristics of an overdamped SOPDT model where the response will be non-oscillatory and without overshoot. As shown previously, the process response of a Model II type is

$$\bar{y}(\bar{t}) = 1 - \left(\frac{1-\bar{a}}{1-\eta}\right)e^{-\bar{t}} - \left(\frac{\eta-\bar{a}}{1-\eta}\right)e^{-\bar{t}/\eta}, \text{ where } \bar{t} = (t-\theta)/\tau$$

If it is assumed that $\bar{a}=0$ and the model has no process zero, then the $\bar{t}_x$ value at any given x is a function of ζ or η. This value can be easily obtained by solving the above response equation with a single MATLAB instruction. For later use, the values of $\bar{t}_x$ for $x = 0.3, 0.5, 0.7, 0.9$ have been calculated and represented as polynomial functions of η as follows:

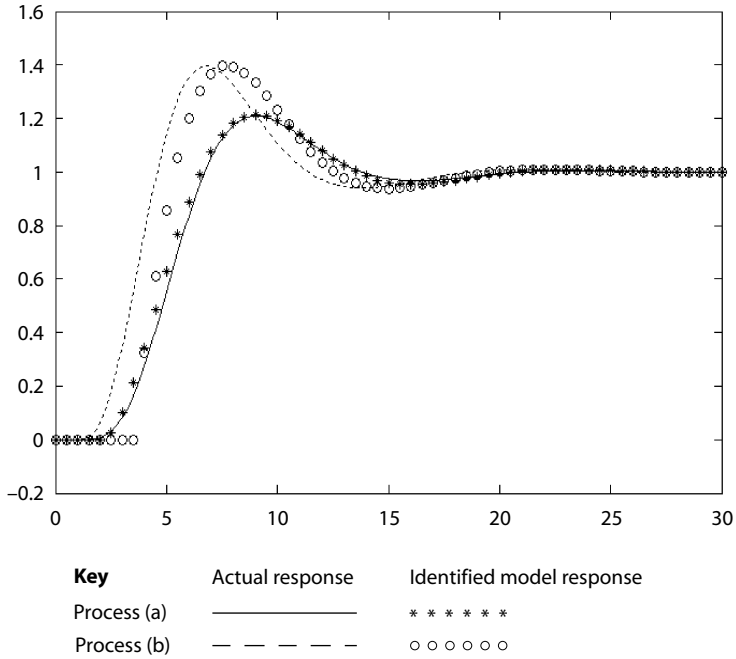


Figure 8.6 Step responses of processes (a) and (b) in Example 8.1.

Table 8.1 Identification results for the processes of Example 8.1.

Quantities computed		**Process (a)**		**Process (b)**	
		$\dfrac{e^{-s}}{(4s^2+2s+1)(s^2+s+1)}$		$\dfrac{(2s+1)e^{-2s}}{(4s^2+2s+1)(s^2+s+1)}$	
$t_{p,1}$	$y_{p,1}$	9.03	1.41	6.79	1.40
$t_{p,2}$	$y_{p,2}$	23.54	1.06	21.12	1.01
$t_{m,0}$	$t_{p,1} - t_{m,0}$	1.75	7.28	2.21	4.58
P		14.51		14.33	
χ		0.25		0.26	
τ		2.00		1.97	
ζ		0.44		0.51	
$\bar{a}$		0		1.33	
θ		2.03		3.50	
Identified models		$\dfrac{e^{-2.03s}}{(4.00s^2+1.76s+1)}$		$\dfrac{(2.62s+1)e^{-3.50s}}{(3.88s^2+2.01s+1)}$	

$$\bar{t}_{0.3} = 0.3548 + 1.1211\eta - 0.5914\eta^2 + 0.2145\eta^3$$
$$\bar{t}_{0.5} = 0.6862 + 1.1682\eta - 0.1704\eta^2 - 0.0079\eta^3$$
$$\bar{t}_{0.7} = 1.1988 + 1.0818\eta + 0.4043\eta^2 - 0.2501\eta^3$$
$$\bar{t}_{0.9} = 2.3063 + 0.9017\eta + 1.0214\eta^2 + 0.3401\eta^3$$

These forms for the function representation of $\bar{t}_x(\eta)$ are obtained from polynomial regressions of the graphs shown in Figure 8.7.

For identifying SOPDT processes, a ratio that characterises the response in time domain, designated as $R(x)$, is defined as follows:

$$R(x) = \frac{M_\infty - t_{(x-0.2)}}{t_x - t_{(x-0.2)}}, \text{ where } x > 0.2$$

In the above equation, the function $M(t)$ denotes the integration of $e(t) = y_\infty - y(t)$ with respect to time t, namely

$$M(t) \equiv \int_0^t e(t')\mathrm{d}t' = \int_0^t (y_\infty - y(t'))\mathrm{d}t' = K\int_0^t (1 - \bar{y}(t'))\mathrm{d}t'$$

The normalised value of $M(t)$, denoted $\overline{M}(\bar{t})$, is defined as

$$\overline{M}(\bar{t}) \equiv \int_0^{\bar{t}} (1 - \bar{y}(t'))\mathrm{d}t'$$

Recalling that the time delay is denoted θ, variable $M(t)$ is related to $\overline{M}(\bar{t})$ by the following analysis:

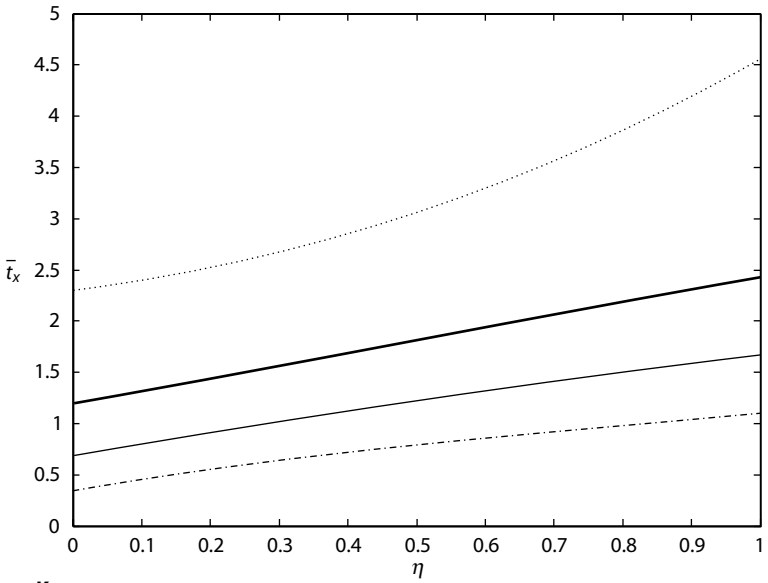


Figure 8.7 Graphs for $\bar{t}_x$ for an overdamped process with $\bar{a} = 0$.

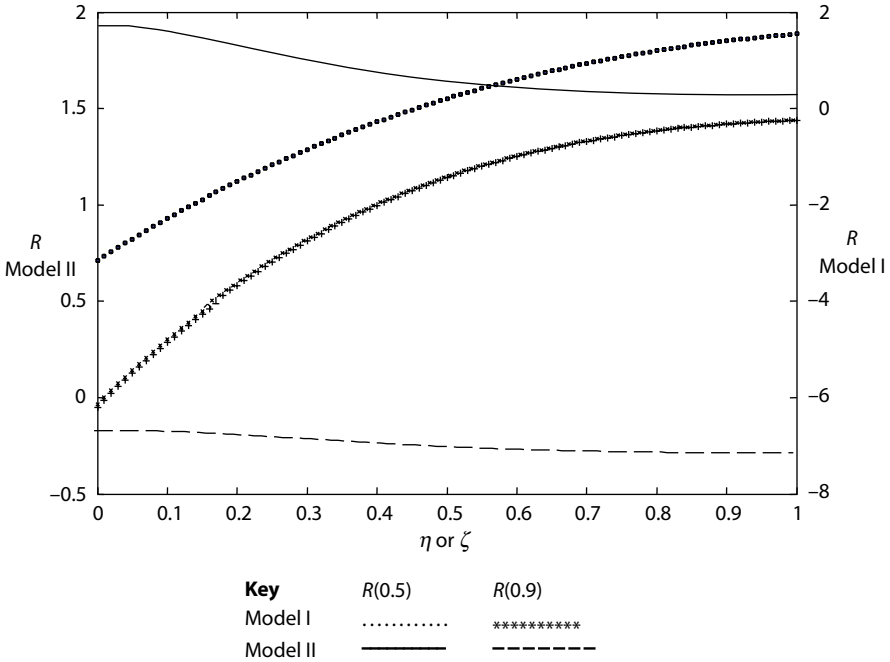


Figure 8.8 $R(x)$ for model I and model II with $\bar{a} = 0$.

$$M(t) = \int_0^t (y_\infty - y(t'))\mathrm{d}t' = \int_0^\theta y_\infty \mathrm{d}t' + \int_\theta^t (y_\infty - y(t'))\mathrm{d}t'$$
$$= K\theta + K\int_\theta^t (1 - \bar{y}(t'))\mathrm{d}t' = K\theta + K\int_\theta^{\bar{t}} (1 - \bar{y}(\bar{t}))\tau \mathrm{d}\bar{t}$$

and

$$M(t) = K[\theta + \bar{M}(\bar{t})\tau]$$

Values of $R(x)$ at $x = 0.5$ and at $x = 0.9$ are plotted in Figure 8.8.

It is found that for each value of $R(x)$ at a given x there is only one value of η. As a result, an average of these two values can be taken as the estimate for η. For later use, the values of $R(x)$ at $x = 0.5$ and $x = 0.9$ have been calculated and regression methods used to give functional representations in polynomials of η as follows:

$$R(0.5) = 1.9108 + 0.2275\eta - 5.5504\eta^2 + 12.8123\eta^3 - 11.8164\eta^4 + 3.9735\eta^5$$
$$R(0.9) = -0.1871 + 0.0736\eta - 1.2329\eta^2 + 2.1814\eta^3 - 1.5317\eta^4 + 0.3937\eta^5$$

Thus, by using the polynomial equations and the experimental values of R, the value of η can be obtained. With this estimated value of η, the values of $\bar{t}_x$ at $x = 0.3, 0.5, 0.7, 0.9$ can be calculated enabling the time constant value τ to be estimated by the following equation:

$$\tau = \frac{t_{x,i} - t_{x,j}}{\bar{t}_{x,i} - \bar{t}_{x,j}}$$

where i and j are any integers.

Consequently, model parameters for Model II processes can be estimated using the following algorithm.

Algorithm 8.2: SOPDT models for non-oscillatory overdamped step responses

Step 1 Complete the step response test

Step 2 *Finding* η

Calculate $R(0.5)$ and $R(0.9)$ from the experimental step response.

Determine η from Figure 8.8 or the equation set

$$R(0.5) = 1.9108 + 0.2275\eta - 5.5504\eta^2 + 12.8123\eta^3 - 11.8164\eta^4 + 3.9735\eta^5$$

$$R(0.9) = -0.1871 + 0.0736\eta - 1.2329\eta^2 + 2.1814\eta^3 - 1.5317\eta^4 + 0.3937\eta^5$$

$$\text{Take average value } \eta = \frac{\eta_{0.5} + \eta_{0.9}}{2}$$

Step 3 *Finding time constant* τ

Use η estimate in the equation set

$$\bar{t}_{0.3} = 0.3548 + 1.1211\eta - 0.5914\eta^2 + 0.2145\eta^3$$

$$\bar{t}_{0.5} = 0.6862 + 1.1682\eta - 0.1704\eta^2 - 0.0079\eta^3$$

$$\bar{t}_{0.7} = 1.1988 + 1.0818\eta + 0.4043\eta^2 - 0.2501\eta^3$$

$$\bar{t}_{0.9} = 2.3063 + 0.9017\eta + 1.0214\eta^2 + 0.3401\eta^3$$

Use $\bar{t}_{0.3}, \bar{t}_{0.5}, \bar{t}_{0.7}$ and $\bar{t}_{0.9}$ to compute

$$\tau = \frac{1}{3}\left[\frac{t_{0.9} - t_{0.7}}{\bar{t}_{0.9} - \bar{t}_{0.7}} + \frac{t_{0.7} - t_{0.5}}{\bar{t}_{0.7} - \bar{t}_{0.5}} + \frac{t_{0.5} - t_{0.3}}{\bar{t}_{0.5} - \bar{t}_{0.3}}\right]$$

Step 4 *Finding process delay time* θ

$$\text{Compute } \theta = \frac{t_{0.9} + t_{0.7} + t_{0.5} + t_{0.3}}{4} - \left(\frac{\bar{t}_{0.9} + \bar{t}_{0.7} + \bar{t}_{0.5} + \bar{t}_{0.3}}{4}\right)\tau$$

Algorithm end

A demonstration of Algorithm 8.2 is given in Example 8.2.

Example 8.2

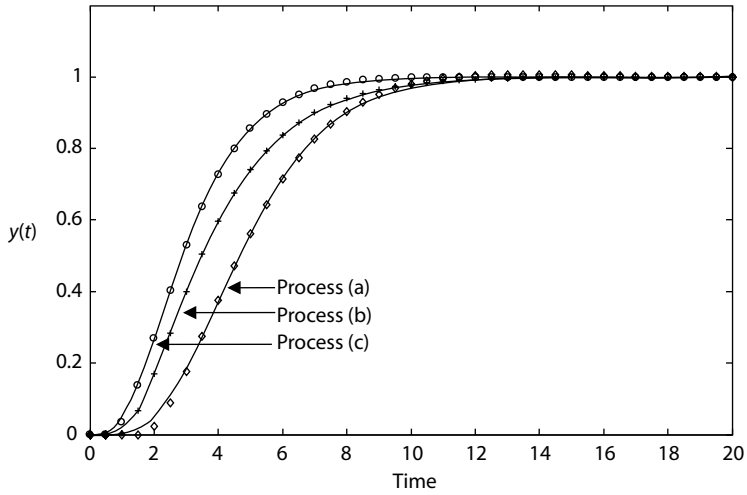
Consider the following three processes:

$$\text{Process (a)} \qquad G_p(s) = \frac{1}{(s+1)^5}$$

$$\text{Process (b)} \qquad G_p(s) = \frac{e^{-0.5s}}{(2s+1)(s+1)(0.5s+1)}$$

$$\text{Process (c)} \qquad G_p(s) = \frac{e^{-0.2s}}{(s+1)^3}$$

The actual step responses, as shown in Figure 8.9, are non-oscillatory and exhibit neither response overshoots nor inverse response features.



Key	Fitted models
Process (a)	◇
Process (b)	○
Process (c)	+
Actual process	Solid line

Figure 8.9 Actual and fitted step response graphs.

Table 8.2 Identification results for the processes of Example 8.2.

Quantities computed		**Process (a)**		**Process (b)**		**Process (c)**	
		$\frac{1}{(s+1)^5}$		$\frac{e^{-0.5s}}{(2s+1)(s+1)(0.5s+1)}$		$\frac{e^{-0.2s}}{(s+1)^3}$	
$t_{0.3}$	$t_{0.5}$	3.63	4.67	2.75	3.48	2.11	2.87
$t_{0.7}$	$t_{0.9}$	3.89	7.99	4.67	7.01	3.82	5.52
M_∞		5.00		4.00		3.20	
$R(0.5)$	$R(0.9)$	1.32	−0.42	1.57	−0.29	1.43	−0.36
ζ or η		0.85		0.74		0.92	
τ		2.02		1.80		1.41	
θ		1.53		0.86		0.61	
Identified models		$\frac{e^{-1.53s}}{(4.08s^2+3.43s+1)}$		$\frac{e^{-0.86s}}{(1.80s+1)(1.33s+1)}$		$\frac{e^{-0.61s}}{(1.99s^2+2.59s+1)}$	

The procedure of Algorithm 8.2 was applied and some important values relating to the steps of the identification are given in Table 8.2 along with the final fitted model parameters. The predicted step responses for each of the fitted models compared with those from the real process are also shown in Figure 8.9.

Summary Remarks

In this section, two algorithms were presented for estimating simple dynamic models using data that characterises the step response. These methods assumed that there is no right half-plane zero present in the process. A more general procedure for identifying simple models that include a right half-plane zero can be found in the work of Huang *et al.* (2001). The derivation and use of a criterion for determining model structure is not discussed here but results and more details can be also be found in Huang *et al.* (2001).

Additional similar methods for identifying models of specific second-order transfer function form have also been reported by Huang and co-workers (1993; 1994). Alternative methods that identify simple second-order transfer function models have been reported by Wang *et al.* (2001), but these have not been described here since these methods use the extensive response time sequence data of the step response.

8.3 Developing Simple Models from a Relay Feedback Experiment

Conventionally, a relay experiment is used to estimate the ultimate gain and ultimate frequency of a given process so that rule-based PID control design may be applied. In this section, a different type of relay experiment is introduced as an identification method that aims to produce simple models for tuning PID controllers. The general structure of this new relay feedback experiment is shown in Figure 8.10.

The identification is conducted by compensating the output from the relay feedback loop with a simple compensator. The objective is to make the compensated output behave like the one from an integral plus dead time (IPDT) process. An on-line mechanism to adjust the compensator is presented which will lead to the identification of the parameters of the desired simple model.

To obtain simple and reduced order models, the use of a relay feedback experiment is very attractive due to several advantages. These advantages include (Yu, 1999):

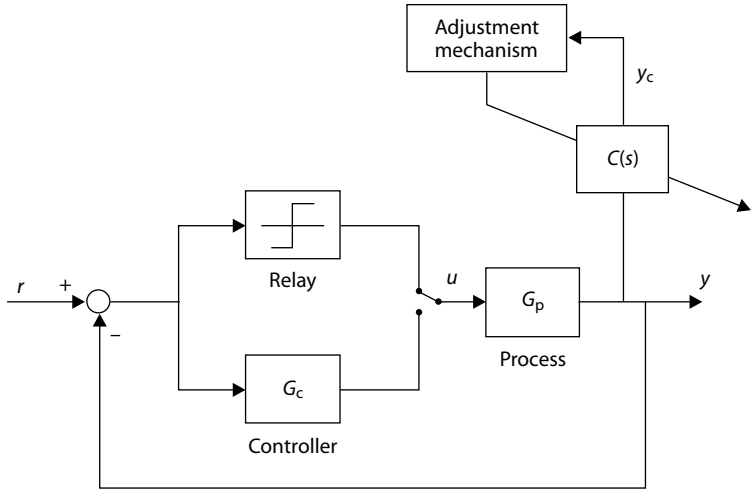


Figure 8.10 General structure of on-line identification scheme using relay feedback.

1. The process is operated in closed loop so that the output does not drift away from its desired target.
2. Relay feedback is a very time-efficient method for on-line testing procedures.
3. Although unknown disturbances and process noise can cause difficulties for the relay experiment, there are a number of simple remedies and modifications available.

Conventionally, the relay feedback test is used to implement the method of sustained oscillations and estimate the ultimate gain and ultimate frequency of a given process. However, the classical relay experiment has been extended in many works to identify parametric models (Li *et al.*, 1991; Sung *et al.*, 1996; Huang *et al.*, 2000; Luyben, 2001). Hang *et al.* (2002) have recently published a comprehensive review of this area. In general, a relay experiment provides data to estimate one single point on the Nyquist plot, for example the data point corresponding to phase crossover frequency. Although there are modifications which allow the identification of a limited number of other points, this is often insufficient for identifying simple FOPDT and SOPDT process models.

8.3.1 On-line Identification of FOPDT Models

In this section, the simple compensated relay feedback structure of Figure 8.10 is to be used to identify the usual first-order plus dead time model given as

$$G_p(s)=\frac{Ke^{-\theta s}}{\tau s+1}$$

In order to identify the FOPDT model, a dynamic compensator $C(s)$ is appended outside the loop (see Figure 8.10) to generate an instrumental output $y_c(t)$. During the experiment, the loop controller $G_c(s)$ is switched out and the loop is under relay feedback control. The compensator $C(s)$ is chosen to produce integral plus dead time output. Thus, consider a compensator of the form

$$C(s)=\frac{\hat{\tau}s+1}{s}$$

where $\hat{\tau}$ is an increasingly improving estimate of the process time constant τ.

The forward path output is given as

$$Y_c(s)=G_{fp}(s)U(s)=G_p(s)C(s)U(s)$$

Consequently, the forward path transfer function becomes

$$G_{fp}(s)=G_p(s)C(s)=K\left(\frac{1}{s}+\frac{\hat{\tau}-\tau}{\tau s+1}\right)e^{-\theta s}$$

and the output can be written in terms of two components, each driven by the same input:

$$Y_c(s)=\left[K\left(\frac{1}{s}\right)e^{-\theta s}\right]U(s)+\left[K\left(\frac{\hat{\tau}-\tau}{\tau s+1}\right)e^{-\theta s}\right]U(s)$$

If the value of $\hat{\tau}$ is properly adjusted to converge to the actual process time constant τ, the process output reduces to the residual output:

$$Y_c(s)=\left[K\left(\frac{1}{s}\right)e^{-\theta s}\right]U(s)$$

Thus the output curve $y_c(t)$ will be the same as one generated by an IPDT process driven by the same input sequence. When this is achieved, the time constant term of the FOPDT process has been exactly

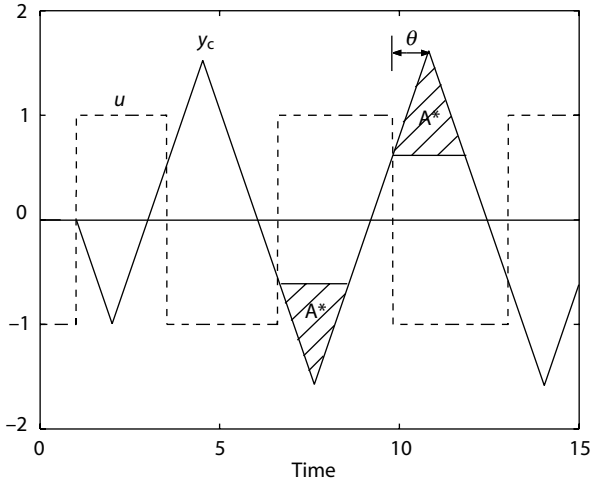


Figure 8.11 Input and compensated output waveforms for the IPDT process in an ideal relay feedback loop.

cancelled by the compensator $C(s)$ and the value of $\hat{\tau}$ will be equal to the actual value of the time constant τ.

The mechanisms which can be used to provide the adjustment scheme are derived from the knowledge that the desired output should be the output of an IPDT process driven by the relay input. Figure 8.11 shows this output and some measurable quantities which are available from this waveform.

An Estimate of Process Gain *K*

To estimate the process gain K, a biased relay with two levels (that is, $+h_1$ and $-h_2$) is used in the relay feedback loop. Then the process gain can be obtained as

$$K = \frac{\int_t^{t+P} y(t)\mathrm{d}t}{\int_t^{t+P} u(t)\mathrm{d}t}$$

where P is the oscillation period of output $y(t)$.

An Adjustment Mechanism for $\hat{\tau}$

The output shown in Figure 8.11 will be attained if the time constant estimate $\hat{\tau}$ converges to the unknown actual time constant τ. In the figure, the shaded triangular area A^* can be computed and used as the target value for a recursive algorithm to update the value of $\hat{\tau}$ to make it approach τ. When a biased-relay as mentioned is used to generate the response, the target value of A^* can be expressed as

$$A^* = \frac{1}{2} K h_1 \left(1 + \frac{h_1}{h_2}\right) \theta^2$$

Define E as the difference between the actual and target values of A^*, that is

$$E = A - A^*$$

The actual value of A^*, namely A, can be computed from a numerical integration of $y_c(t)$. Then an update equation can be given as

$$\hat{\tau}^{i+1} = \hat{\tau}^{i} - k_{\mathrm{a}} \frac{\mathrm{d}E}{\mathrm{d}\hat{\tau}} E$$

where k_{a} is an adjustable gain for updating $\hat{\tau}$. The initial value of $\hat{\tau}$, namely $\hat{\tau}^0$, can be computed as

$$\hat{\tau}^0 = \frac{Kh_1}{\dot{y}_{\mathrm{s}}}$$

where $\dot{y}_{\mathrm{s}}$ is the positive slope of output as it crosses the setpoint value, h_1 is the magnitude of the positive relay output, and K is the process gain. The convergence of this algorithm can be guaranteed if the value of k_{a} is properly selected (Huang and Jeng, 2003).

An Estimate of Process Time Delay θ

The value of the process time delay θ can be also read from the output time waveform $y_{\mathrm{c}}(t)$. Using Figure 8.11, the process time delay is found as the time that the output $y_{\mathrm{c}}(t)$ takes to go from the base of the shaded triangle to the waveform peak.

In a true FOPDT process, the time delay can be taken as the time, denoted as d, that $y(t)$ takes to go from the setpoint value to the waveform peak in a relay feedback test. Thus a simple means to justify whether the FOPDT model is adequate for representing the process is to check whether the obtained time delay is close enough to d using the following criterion:

$$\Delta = \left| 1 - \frac{\theta}{d} \right| \leq \varepsilon$$

where ε is an arbitrarily small value assigned to provide a tolerance of estimation errors.

8.3.2 On-line Identification of SOPDT Models

In cases where the unknown process does not exhibit oscillatory characteristics and the FOPDT model is found inadequate to represent processes of high-order dynamics, an SOPDT model should be identified instead. The SOPDT model is given as

$$G_{\mathrm{p}}(s) = \frac{K\mathrm{e}^{-\theta s}}{(\tau_1 s + 1)(\tau_2 s + 1)}$$

The identification scheme retains the basic adaptive structure shown in Figure 8.10, and the theory for the method follows the same outline steps as used for the on-line identification of the FOPDT model above. However, in the SOPDT case the dynamic compensator $C(s)$ is now given as

$$C(s) = \frac{(\hat{\tau}_1 s + 1)(\hat{\tau}_2 s + 1)}{s(\tau_{\mathrm{f}} s + 1)}$$

where $\hat{\tau}_1$, $\hat{\tau}_2$ are the estimated values for the process time constants, and τ_{f} is a small constant to make compensator $C(s)$ realisable.

As in the previous case, the forward path output is given as

$$Y_{\mathrm{c}}(s) = G_{\mathrm{fp}}(s)U(s) = G_{\mathrm{p}}(s)C(s)U(s)$$

Substituting for the process and the compensator transfer functions (and neglecting τ_{f} in $C(s)$) enables the output to be written in terms of two components, each driven by the same input:

$$Y_{\mathrm{c}}(s) = \left[K\left(\frac{1}{s}\right)\mathrm{e}^{-\theta s} \right] U(s) + \left[K\left(\frac{Bs + C}{(\tau_1 s + 1)(\tau_2 s + 1)} \right)\mathrm{e}^{-\theta s} \right] U(s)$$

where $B = \hat{\tau}_1 \hat{\tau}_2 - \tau_1 \tau_2$ and $C = (\hat{\tau}_1 + \hat{\tau}_2) - (\tau_1 + \tau_2)$.

Clearly, if $\hat{\tau}_1, \hat{\tau}_2$ converge to the process time constants τ_1, τ_2, then the process output converges to the residual output, that is

$$Y_c(s) = \left[K\left(\frac{1}{s}\right)e^{-\theta s} \right] U(s)$$

Consequently, the same principles used for identifying the FOPDT model can be applied to the case of SOPDT models, including the method of estimating the steady state gain.

An Adjustment Mechanism for $\hat{\tau}_1$ and $\hat{\tau}_2$

In the case of the SOPDT models, the output shown in Figure 8.11 will also be attained if the time constants are estimated correctly. In the waveform, the shaded triangular area A^* can be computed and used as the target value for a recursive update algorithm as follows.

As in the FOPDT case, define the area error E as

$$E = A - A^*$$

Then two update equations are defined as

$$\hat{\tau}_1^{i+1} = \hat{\tau}_1^i - k_{a1}\frac{\partial E}{\partial \hat{\tau}_1}E$$

and

$$\hat{\tau}_2^{i+1} = \hat{\tau}_2^i - k_{a2}\frac{\partial E}{\partial \hat{\tau}_2}E$$

where k_{a1} and k_{a2} are adjustable gains for updating $\hat{\tau}_1$ and $\hat{\tau}_2$, respectively. The vector

$$\nabla E = \left[\frac{\partial E}{\partial \hat{\tau}_1} \; \frac{\partial E}{\partial \hat{\tau}_2} \right]^T$$

can be considered as the gradient direction of area E with respect to the two process time constants. This gradient has to be estimated and can be considered to give the corresponding direction of change in each iteration. The initial value of $\hat{\tau}_1$ can be computed as

$$\hat{\tau}_1^0 = \frac{Kh_1}{\dot{y}_s}$$

as defined above and the initial value of $\hat{\tau}_2$ is set as 0.8 times $\hat{\tau}_1^0$.

An Estimate of the Process Time Delay θ

When the result converges, the process time delay is found as the time that the output $y_c(t)$ takes to go from the base of the shaded triangle to the waveform peak.

8.3.3 Examples for the On-line Relay Feedback Procedure

In order to illustrate the proposed on-line identification method, a second-order plus dead time process and a high-order process were used in two separate examples.

Example 8.3: Identifying an FOPDT model using the on-line procedure

Consider the problem of identifying an FOPDT model for an unknown second-order plus dead time process. The unknown process is taken to be represented by

$$G_p(s) = \frac{e^{-2s}}{(5s+1)(s+1)}$$

Some initial features of the FOPDT model for this process are obtained using a biased relay ($h_1 = 1.2$ and $-h_2 = -1.0$) experiment. According to the procedures, the slope $\dot{y}_s$ is 0.205 and the initialising response time d is 2.45. The estimated process time constant $\hat{\tau}$ is updated according to the recursive procedure of Section 8.3.1:

$$\hat{\tau}^{i+1} = \hat{\tau}^i - k_a \frac{dE}{d\hat{\tau}} E$$

where the adjustable gain k_a is taken as unity, the target waveform area A^* is set to be $1.32\theta^2$, the value of θ is recorded from the current waveform of $y_c(t)$, and the initial value of $\hat{\tau}$ is set as 5.85. The compensator used in the on-line scheme is

$$C(s) = \frac{\hat{\tau}s + 1}{s}$$

As a result, the estimated time constant $\hat{\tau}$ converges after three iterations where the FOPDT model obtained is

$$\hat{G}_p(s) = \frac{e^{-2.66s}}{5.28s + 1}$$

To verify this obtained model, the calculated Δ is found to be less than 0.1 and, thus, it is considered that the resulting FOPDT model is adequate for representing this second-order process.

Example 8.4: Identifying an SOPDT model using the on-line procedure

Consider the following high-order plus dead time process:

$$G_p(s) = \frac{e^{-0.8s}}{(2s+1)^3(s+1)^2}$$

As in the previous example, an FOPDT model was first identified. However, the calculated Δ was 0.4, which indicated that the FOPDT model was not adequate for representing such high-order dynamics. Therefore the identification of an SOPDT model was pursued instead. The identification scheme used a compensator of the form

$$C(s) = \frac{(\hat{\tau}_1 s + 1)(\hat{\tau}_2 s + 1)}{s(0.01s + 1)}$$

where $\hat{\tau}_1, \hat{\tau}_2$ are estimates for the process time constants τ_1, τ_2. The time constant update used the pair of equations:

$$\hat{\tau}_1^{i+1} = \hat{\tau}_1^i - k_{a1} \frac{\partial E}{\partial \hat{\tau}_1} E$$

and

$$\hat{\tau}_2^{i+1} = \hat{\tau}_2^i - k_{a2} \frac{\partial E}{\partial \hat{\tau}_2} E$$

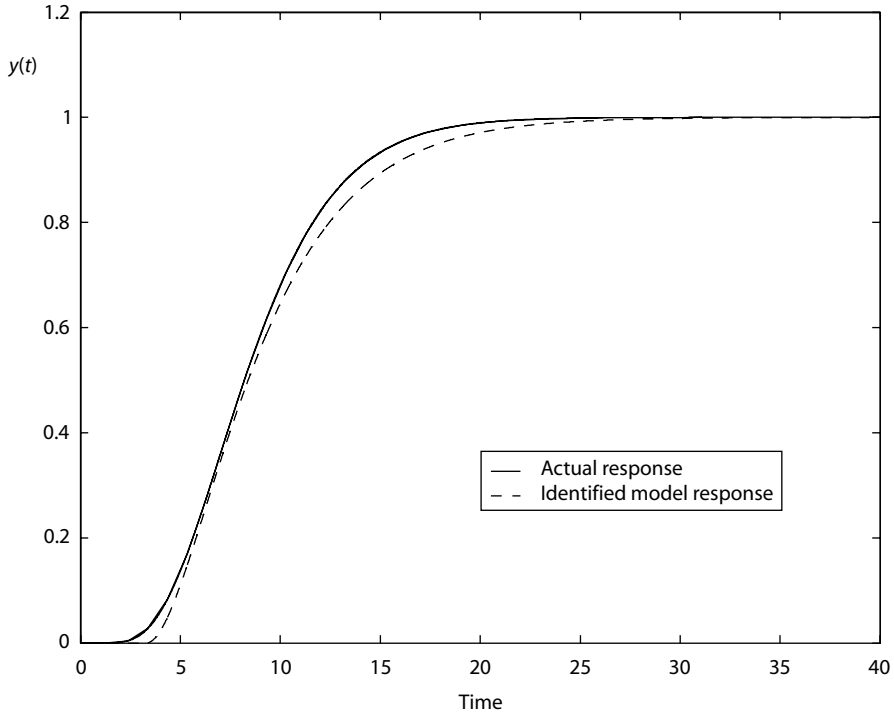


Figure 8.12 Step responses of actual process and identified model in Example 8.4.

where the adjustable gains were given values of $k_{a1} = k_{a2} = 0.5$. The estimation procedure converged after six iterations and the resulting SOPDT model was

$$\hat{G}_p(s) = \frac{e^{-3.30s}}{(3.60s + 1)(2.51s + 1)}$$

To show the result of the identification, a comparison of the responses is given in Figure 8.12.

8.3.4 Off-line Identification

In Sections 8.3.1 and 8.3.2, the identification procedure is conducted on-line with the relay feedback loop. This routine may be found to be time-consuming due to the iterative procedures and the slow process response. An off-line method is presented below which uses only the amplitude and the period of constant cycles.

If it is assumed that there is no process zero, the simple SOPDT models take the following two forms:

$$\frac{Y(s)}{U(s)} = G_p(s) = \begin{cases} \dfrac{Ke^{-\theta s}}{\tau^2 s^2 + 2\tau\zeta s + 1} & 0 < \zeta < 1 \quad \text{(model I)} \\ \dfrac{Ke^{-\theta s}}{(\tau s + 1)(\eta\tau s + 1)} & 0 < \eta \leq 1 \quad \text{(model II)} \end{cases}$$

where model I is used for the underdamped second-order case with $\zeta < 1$ and model II is used for the overdamped second-order case with $\zeta \geq 1$.

These simple SOPDT models for $G_p(s)$ can be changed into dimensionless forms by using the relations $\bar{s} = \theta s$ and $\bar{\tau} = \tau/\theta$ to yield

$$\frac{Y(s)}{KU(s)}=\frac{\overline{Y}(\bar{s})}{U(\bar{s})}=\overline{G}_{\mathrm{p}}(\bar{s})=\begin{cases}\dfrac{\mathrm{e}^{-\bar{s}}}{\overline{\tau}^2\bar{s}^2+2\overline{\tau}\zeta\bar{s}+1} & 0<\zeta<1 \ \ (\text{model I})\\[2ex] \dfrac{\mathrm{e}^{-\bar{s}}}{(\overline{\tau}\,\bar{s}+1)(\eta\,\overline{\tau}\,\bar{s}+1)} & 0<\eta\le 1 \ \ (\text{model II})\end{cases}$$

where $\overline{Y}=Y/K$.

A conventional relay feedback loop of which an ideal relay that has output magnitude h is then simulated. By disturbing the output from its equilibrium point with a pulse, the system will then be activated and the output $y(t)$ has an amplitude designated as a and a cycling period designated as P. Thus, by performing relay experiments numerically on model I and model II above, the responses in terms of normalised amplitude and normalised period are plotted as in Figure 8.13.

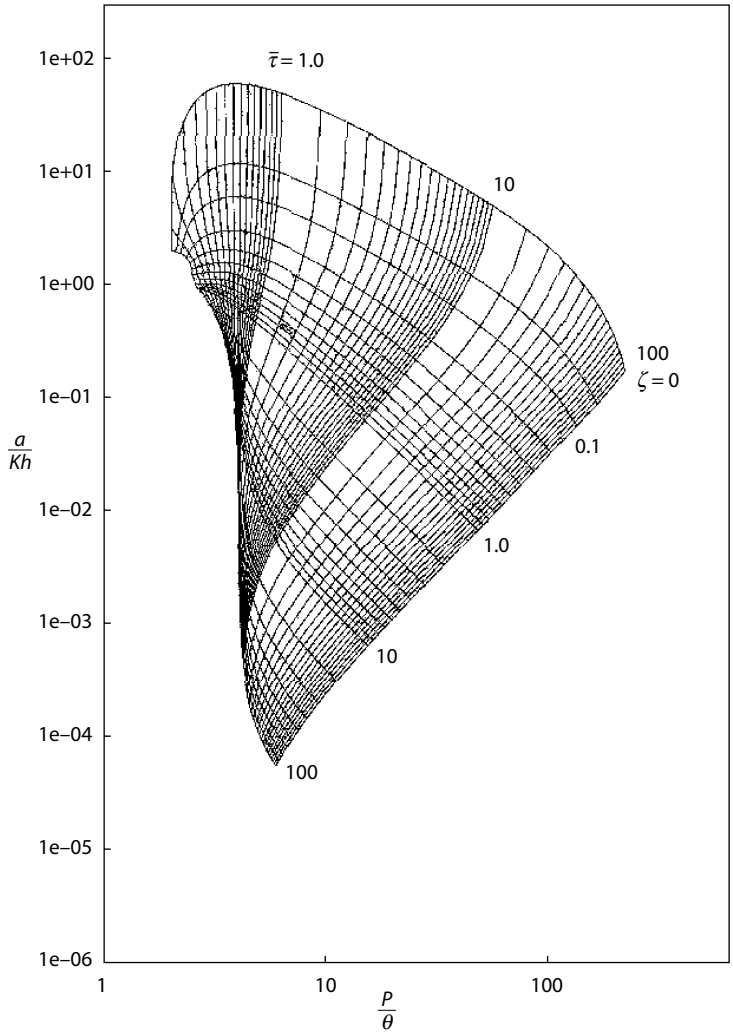


Figure 8.13 Normalised reaction curve $\overline{\tau}$ and ζ of relay feedback test.

It is then considered that in a dynamic system that can be represented by either of the model I or model II SOPDT model types, there will be a point in Figure 8.13 corresponding to the experimental magnitude and cycling period. However, there is one free parameter, namely the apparent dead time θ, to be determined. To fix this free parameter, one more equality relation has to be established. For control design purposes, it is desirable that the model and the actual process should have almost the same ultimate frequency ω_u. This design consideration becomes the additional equality, namely

$$\arg\{\hat{G}_p(j\omega_u)\} = \arg\left\{\hat{G}_p\left(j\frac{2\pi}{P}\right)\right\} = -\pi \text{ rad}$$

Based on the description given above, an iterative procedure for identification is given by the following algorithm.

Algorithm 8.3: Off-line relay feedback identification routine

Step 1 Compute or provide an initial value for the apparent dead time $\hat{\theta}$.

Step 2 Read new values of $\bar{\tau}$ and ζ from Figure 8.13 with the amplitude and cycling period obtained from the experiment.

Step 3 Use the resulting τ and ζ, together with $\hat{\theta}$, to check if $\arg\{\hat{G}_p(j\omega_u)\}$ equals $-\pi$. If not, go back to *Step 1* and repeat the procedures with a new guess of θ according to the one-dimensional search method until $\arg\{\hat{G}_p(j\omega_u)\}$ equals $-\pi$.

Algorithm end

Example 8.5

The results of this off-line procedure for a number of processes are given.

Case 1: Second-order plus dead time process

The process is

$$G_p(s) = \frac{e^{-2s}}{4s^2 + 3s + 1}$$

The identified SOPDT model is

$$\hat{G}_p(s) = \frac{e^{-2s}}{4s^2 + 3.2s + 1}$$

Case 2: High-order plus dead time process

The fifth-order process is

$$G_p(s) = \frac{e^{-0.8s}}{(2s+1)^3(s+1)^2}$$

which is the same one as in Example 8.4.

The identified SOPDT model is

$$\hat{G}_p(s) = \frac{e^{-3.31s}}{10.9561s^2 + 6.62s + 1}$$

Note that the identified model is similar to the result obtained in Example 8.4. The apparent time delay, that is 3.31, is larger than the true time delay of 0.8. This is because the reduction in dynamic order results in some extra time delay in the model to account for the missing process lags.

8.4 An Inverse Process Model-Based Design Procedure for PID Control

Three-term PID controllers have been widely used in chemical plants for process control. There is a wide variety of methods with different complexities that can be used to determine the parameters of PID controllers to meet given performance specifications. Through a very long period of development, PID controller design for SISO systems seems to be reaching a status of considered maturity. The book of Åström and Hägglund (1995) contains a good collection of PID control research papers along with a bibliography for the topic. There are also books (Åström and Hägglund, 1988, 1995; Hang *et al.* 1993; McMillan, 1994) that focus on autotuning principles and PID controller design. PID control tuning formulae based on the internal model control (IMC) principle for simple transfer function models have been given by Chien and Fruehauf (1990) and Hang *et al.* (1991). In this section, a contribution is made to PID autotuning with the derivation of tuning formulae for one degree of freedom (1 dof) PID controllers using an inverse process model-based approach.

8.4.1 Inverse Process Model-Based Controller Principles

Theoretically, non-minimum phase (nmp) elements in a process cannot be eliminated by any controller in a simple closed-loop system. On the other hand, in order to eliminate offsets, an integrator must be a part of the loop transfer function. As a result, the target forward path loop transfer function should consist of at least one integrator and the non-minimum phase elements of the process such as any right half-plane (RHP) zeros and the pure dead time if it is present. This combination of the integrator and the non-minimum phase elements constitute the basic loop transfer function of a control loop. To achieve this basic loop transfer function, many design methods try to use the controller in the loop as an explicit or implicit inverse process model. According to IMC theory, the nominal loop transfer function of a control system that has an inverse process model-based controller will be of the following form:

$$G_{\mathrm{lp}}(s)=\left(\frac{G_{\mathrm{p+}}(s)}{s}\right)F_{\mathrm{lp}}(s)$$

where $F_{\mathrm{lp}}(s)$ serves as a loop filter in the control system and the $G_{\mathrm{p+}}(s)$ contains the non-minimum phase elements of the process and represents the non-invertible part of process $G_{\mathrm{p}}(s)$.

Examples Due to Chien and Fruehauf (1990)

If the process is represented by a model of FOPDT transfer function

$$G_{\mathrm{p}}(s)=\frac{K\mathrm{e}^{-\theta s}}{\tau s+1}$$

then Chien and Fruehauf (1990) show that the resulting loop transfer function $G_{\mathrm{lp}}(s)$ (namely $G_{\mathrm{p}}(s)G_{\mathrm{c}}(s)$) will be

$$G_{\mathrm{lp}}(s)=\left(\frac{\mathrm{e}^{-\theta s}}{s}\right)\left(\frac{1}{\lambda+\theta}\right)$$

or

$$G_{\mathrm{lp}}(s)=\left(\frac{\mathrm{e}^{-\theta s}}{s}\right)\left(\frac{1+0.5\theta s}{(\lambda+\theta)(\tau_{\mathrm{f}}s+1)}\right)$$

Note that in the above loop transfer function the loop filter $F_{\mathrm{lp}}(s)$ has lead–lag form.

Similarly, for a process model of SOPDT type given by

$$G_p(s)=\frac{Ke^{-\theta s}}{\tau^2 s^2+2\tau\zeta s+1}$$

compensation by a controller yields a the loop transfer function

$$G_{lp}(s)=\left(\frac{e^{-\theta s}}{s}\right)\left(\frac{1}{\lambda+\theta}\right)$$

A Methodology Based on Inverse Model-Based Approach

The inverse model-based approach presented here differs from the IMC method in the way the controller is synthesised. In the IMC approach, a target overall transfer function is assigned and is embedded in the so-called IMC filter. On the other hand, the inverse model-based approach is to assign a target loop transfer function, designated as $G_{lp}(s)$ and synthesise the controller according to the following equation:

$$G_c(s)=G_p^{-1}(s)G_{lp}(s)$$

In this way, it is more direct without encountering the mathematical approximation to the process time delay in the derivation of controllers. Also, the gain margin and phase margin can be clearly defined from the assigned $G_{lp}(s)$. Recently, a study of Huang and Jeng (2002) showed that it is possible to obtain optimal performance and reasonable robustness using a system as shown in Figure 8.14 with a loop transfer function of the following form:

$$G_{lp}(s)=\frac{k_0(1+\alpha\theta s)e^{-\theta s}}{s}$$

This closed-loop system has an implicit assumption that a controller can be designed to achieve the target loop transfer function. In the following, the properties related to the robustness issue of the system in Figure 8.14 will be discussed.

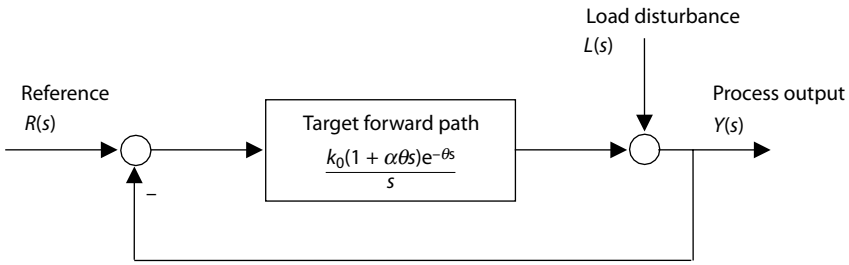


Figure 8.14 Equivalent inverse process model-based control system.

Introduce the normalising relationships $K_0=k_0\theta$ and $\bar{s}=\theta s$; then the above target loop transfer becomes

$$G_{lp}(s)=\frac{k_0(1+\alpha\theta s)e^{-\theta s}}{s}=\frac{K_0(1+\alpha\bar{s})e^{-\bar{s}}}{\bar{s}}$$

The lead element $(1+\alpha\theta s)$ in the above loop transfer function provides an extra degree of freedom for controller design. With the aid of this normalised loop transfer function, it is obvious that the phase crossover frequency of the loop ω_{pco} is only a function of parameter α. Hence, for any $\theta\neq 0$ the phase crossover frequency satisfies

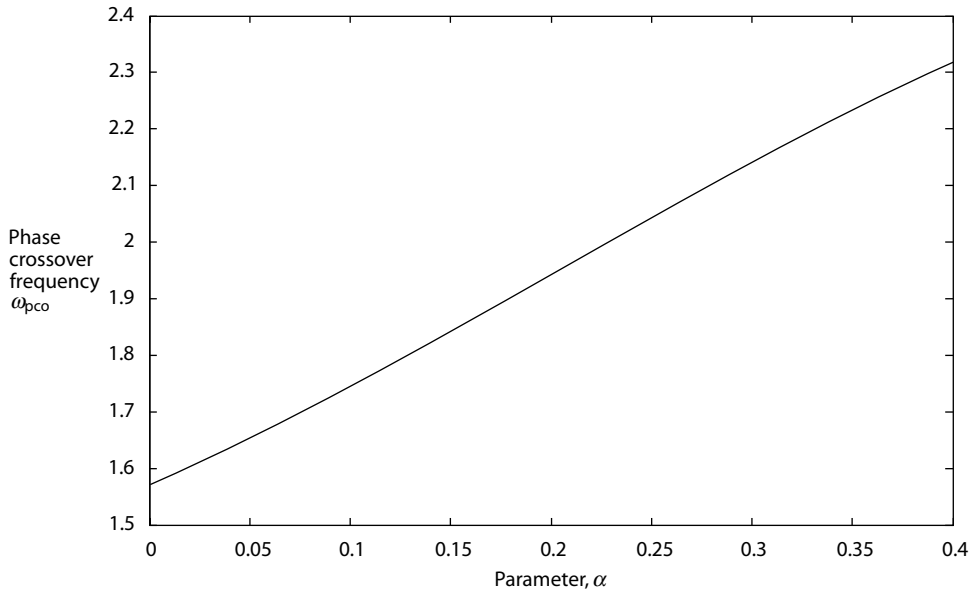


Figure 8.15 Graph of ω_{pco} versus α for the inverse process-based system.

$$-\omega_{pco}\alpha = \cot(\omega_{pco})$$

A graph of the phase crossover frequency ω_{pco} versus α is given in Figure 8.15, where it can be seen that the slope of ω_{pco} to the change of α is about 1.93. This graph leads to the following linear relationship: $\omega_{pco} = 1.93\alpha + 1.56$.

It is then easy to find the gain and phase margins of this equivalent system (Figure 8.14) as, respectively,

$$GM = \frac{\omega_{pco}}{K_0\sqrt{(1+\alpha^2\omega_{pco}^2)}}$$

and

$$\phi_{PM} = \frac{\pi}{2} - \frac{K_0}{\sqrt{(1+\alpha^2K_0^2)}} + \tan^{-1}\frac{\alpha K_0}{\sqrt{(1-\alpha^2K_0^2)}} \text{ rad}$$

Thus the gain and phase margins are functions of K_0 and α; both these equations are plotted in Figure 8.16 to show the relationship between the phase margin and gain margin over the range of K_0 and α. Note that the phase margin has been plotted in degrees in the Figure 8.16.

From Figure 8.16, it is found that the closed-loop system may have gain margin 2.1 and phase margin 60° if K_0 is set equal to 0.8 and α is set equal to 0.4. Simulation studies with the system structure of Figure 8.14 reveal that the value $\alpha = 0.4$ yields good performance and robustness properties and is selected as the proper choice for this parameter. If parameter α is chosen to have the value $\alpha = 0.4$, then the following relation is obtained:

$$\omega_{pco} = 1.93\alpha + 1.56 = 2.332$$

and the gain margin equation can be rearranged to give

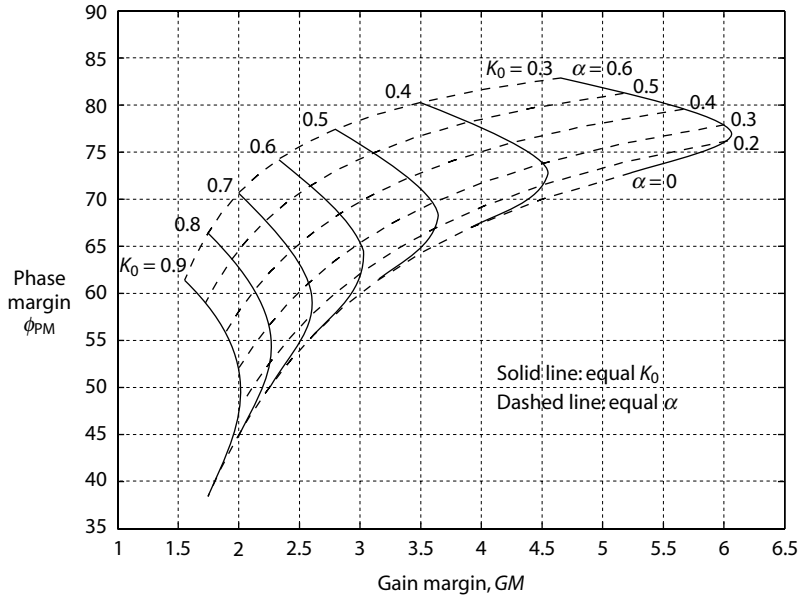


Figure 8.16 Gain margin versus phase margin (degrees) for range of K_0 and α.

$$K_0 = \frac{\omega_{\text{pco}}}{GM\sqrt{(1+\alpha^2\omega_{\text{pco}}^2)}} = \frac{1.7}{GM}$$

Thus, by specifying a gain margin, the overall normalised loop gain K_0 can be obtained, and with this loop gain the corresponding phase margin is also determined.

Finally, for practical synthesis it is necessary to introduce a low-pass filter into the target loop transfer function as follows:

$$G_{\text{lp}}(s) = \frac{K_0(1+0.4\bar{s})\text{e}^{-\bar{s}}}{\bar{s}(1+0.01\bar{s})}$$

Note that the time constant of this additional filter can be set to an arbitrarily small value, which is chosen as 0.01 here. It is this practical target loop transfer function which plays an important role in the synthesis of the three-term controllers described in the next section.

8.4.2 PI/PID Controller Synthesis

Based on the desired loop transfer function given above, a tuning formula for the PID controller can be easily derived for different processes. The key relationship is the forward path transfer equality:

$$G_{\text{p}}(s)G_{\text{c}}(s) = G_{\text{lp}}(s) = \frac{K_0(1+0.4\bar{s})\text{e}^{-\bar{s}}}{\bar{s}(1+0.01\bar{s})}$$

Into the forward path transfer $G_{\text{lp}}(s) = G_{\text{p}}(s)G_{\text{c}}(s)$ can be inserted different process models for $G_{\text{p}}(s)$ and different three-term controller structures, and the equality solved to synthesise the controller.

Case 1: FOPDT Process Model and PID Controller

The process model is

$$G_p(s)=\left(\frac{Ke^{-\theta s}}{\tau s+1}\right)$$

Let the PID controller be

$$G_c(s)=k_p\left(1+\frac{1}{\tau_i s}\right)\left(\frac{\tau_d s+1}{\tau_f s+1}\right)$$

The forward path relationship is

$$G_p(s)G_c(s)=\left(\frac{Ke^{-\theta s}}{\tau s+1}\right)k_p\left(\frac{\tau_i s+1}{\tau_i s}\right)\left(\frac{\tau_d s+1}{\tau_f s+1}\right)$$
$$=G_{lp}(s)=\frac{K_0(1+0.4\bar{s})e^{-\bar{s}}}{\bar{s}(1+0.01\bar{s})}$$

from which the controller parameters are determined as

$$k_p=\frac{0.65\tau_i}{K\theta},\ \tau_i=\tau,\ \tau_d=0.4\theta \text{ and } \tau_f=0.01\theta$$

Case 2: FOPDT Process Model and PI Controller

Let the PI controller be

$$G_c(s)=k_p\left(1+\frac{1}{\tau_i s}\right)$$

The forward path relationship is

$$G_p(s)G_c(s)=\left(\frac{Ke^{-\theta s}}{\tau s+1}\right)k_p\left(\frac{\tau_i s+1}{\tau_i s}\right)$$
$$=G_{lp}(s)\approx\frac{K_0(1+0.4\bar{s})e^{-\bar{s}}}{\bar{s}(1+0.01\bar{s})}$$

The synthesis of the PI controller is not as direct, since the target loop transfer function cannot be exactly achieved. As an approximation, the controller parameters are determined as

$$k_p=\frac{0.55\tau_i}{K\theta} \text{ and } \tau_i=0.4\theta+0.9\tau$$

Case 3: Overdamped SOPDT Process Model and PID Controller

Consider an overdamped SOPDT process of the form

$$G_p(s)=\frac{Ke^{-\theta s}}{(\tau s+1)(\eta\tau s+1)}$$

Let the PID controller be

$$G_c(s)=k_p\left(1+\frac{1}{\tau_i s}\right)\left(\frac{\tau_d s+1}{\tau_f s+1}\right)$$

The forward path relationship is

$$G_p(s)G_c(s) = \left(\frac{Ke^{-\theta s}}{(\tau s+1)(\eta\tau s+1)}\right)k_p\left(\frac{\tau_i s+1}{\tau_i s}\right)\left(\frac{\tau_d s+1}{\tau_f s+1}\right)$$
$$= G_{lp}(s) \approx \frac{K_0(1+0.4\bar{s})e^{-\bar{s}}}{\bar{s}(1+0.01\bar{s})}$$

This case is very similar to Case 2; hence, if the derivative time constant τ_d is set equal to τ, the controller parameters are determined as

$$k_p = \frac{0.55\tau_i}{K\theta},\ \tau_i = 0.4\theta + 0.9\eta\tau,\ \tau_d = \tau \text{ and } \tau_f = 0.01\theta$$

Case 4: Underdamped SOPDT Process Model and PID Controller
Consider the underdamped SOPDT process model

$$G_p(s) = \frac{Ke^{-\theta s}}{(\tau^2 s^2 + 2\tau\zeta s + 1)}$$

Let the PID controller have an extra filter, namely

$$G_c(s) = k_p\left(1 + \frac{1}{\tau_i s} + \tau_d s\right)\left(\frac{1}{\tau_f s+1}\right)$$

Then the forward path relationship is

$$G_p(s)G_c(s) = \left(\frac{Ke^{-\theta s}}{(\tau^2 s^2 + 2\tau\zeta s + 1)}\right)k_p\left(\frac{\tau_i\tau_d s^2 + \tau_i s + 1}{\tau_i s}\right)\left(\frac{1}{1+\tau_f s}\right)$$
$$= G_{lp}(s) = \frac{K_0 e^{-\bar{s}}}{\bar{s}(1+0.01\bar{s})}$$

Notice that in this case, the target loop transfer function used in the previous cases is not achievable. Based on the given loop transfer function, the controller parameters are determined as

$$k_p = \frac{0.55\tau_i}{K\theta},\ \tau_i = 2\tau\zeta,\ \tau_d = \frac{\tau}{2\zeta} \text{ and } \tau_f = 0.01\theta$$

8.4.3 Autotuning of PID Controllers

It remains to incorporate the identification of simple models with the tuning formula described in Section 8.4.2 to illustrate the autotuning of PI/PID controllers. The autotuning procedure requires a step response from the open-loop process or from a relay feedback experiment. In this section the results from the example processes have been identified in Sections 8.2 and 8.3. By applying the resulting models and the tuning formula, autotuning of PID controllers is demonstrated and assessed.

Example 8.6: Non-oscillatory high-order process – autotuning using models from reaction curve method

A demonstration of Algorithm 8.2 was given in Example 8.2 for a non-oscillatory process given by

Process (b) $$G_p(s) = \frac{e^{-0.5s}}{(2s+1)(s+1)(0.5s+1)}$$

The identification algorithm used a reaction curve approach and yielded the model

$$\hat{G}_p(s) = \frac{e^{-0.86s}}{(1.8s+1)(1.33s+1)}$$

The PID controller was given in the series form

$$G_c(s) = k_p\left(1+\frac{1}{\tau_i s}\right)\left(\frac{1+\tau_d s}{1+\tau_f s}\right)$$

According to the tuning formulae the controller parameters were found as

$k_p = 0.98$, $\tau_i = 1.54$, $\tau_d = 1.80$ and $\tau_f = 0.0086$

The responses resulting from the use of this and other controllers with the full process transfer function are shown in Figure 8.17. The performance of inverse-based controller tuning is satisfactory and similar that from IMC tuning. On the other hand, ZN tuning results in excessive overshoot and an over-oscillatory response.

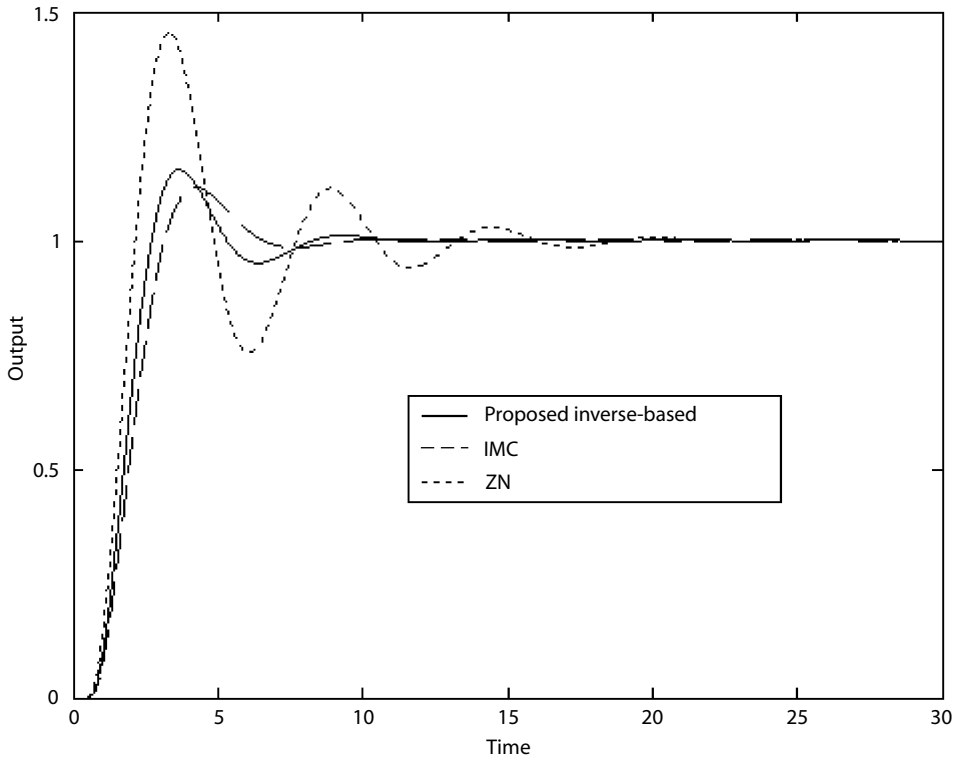


Figure 8.17 Comparison of step reference responses for the process $G_p(s) = \frac{e^{-0.5s}}{(2s+1)(s+1)(0.5s+1)}$.

Example 8.7: FOPDT model for a SOPDT process – autotuning using models from relay feedback test

In Example 8.3 the on-line relay feedback procedure was used to identify a FOPDT model for an unknown second-order plus dead time process. The unknown process is taken to be represented by

$$G_p(s) = \frac{e^{-2s}}{(5s+1)(s+1)}$$

The identified FOPDT model was

$$\hat{G}_p(s) = \frac{e^{-2.66s}}{5.28s + 1}$$

The PID controller was given in the series form as

$$G_c(s) = k_p\left(1 + \frac{1}{\tau_i s}\right)\left(\frac{1 + \tau_d s}{1 + \tau_f s}\right)$$

According to the tuning formulae the controller parameters were found as

$$k_p = 1.29,\ \tau_i = 5.28,\ \tau_d = 1.06 \text{ and } \tau_f = 0.0266$$

For PI control, the controller has the form

$$G_c(s) = k_p\left(1 + \frac{1}{\tau_i s}\right)$$

and the PI controller parameters are

$$k_p = 1.09,\ \tau_i = 5.82$$

The output responses of the system to a step setpoint change are given in Figure 8.18 for the PID controller and Figure 8.19 for the PI controller. The responses resulting from both inverse-based PID and PI controllers are similar to that of IMC, and are much better than those of ZN tuned systems.

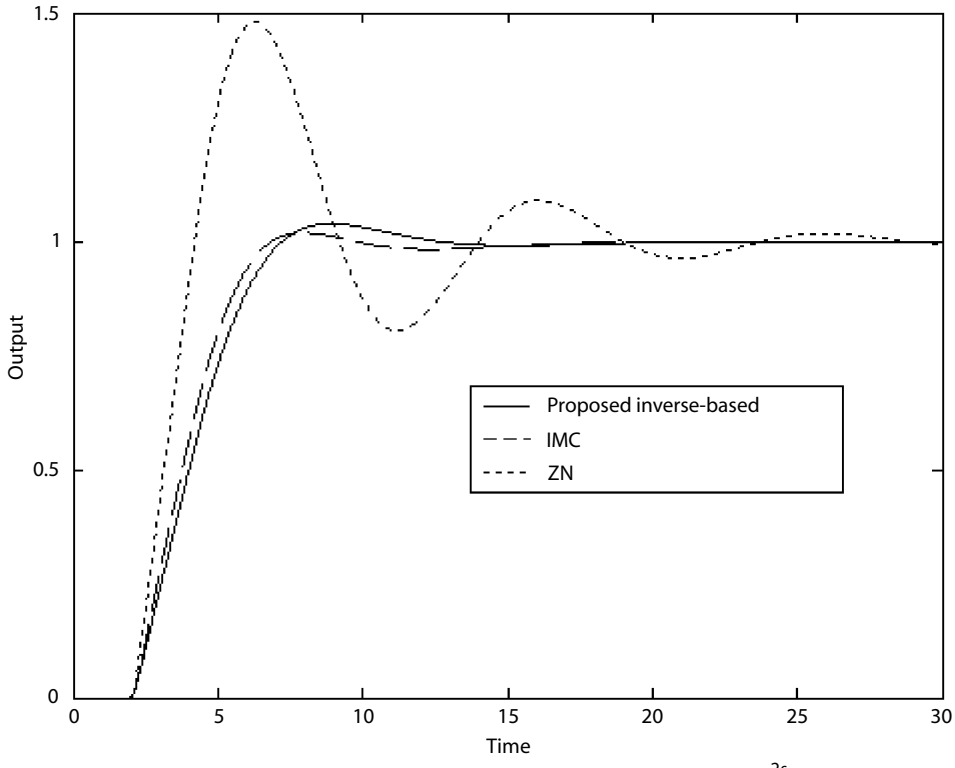


Figure 8.18 Comparison of PID control results for $G_p(s) = \dfrac{e^{-2s}}{(5s + 1)(s + 1)}$.

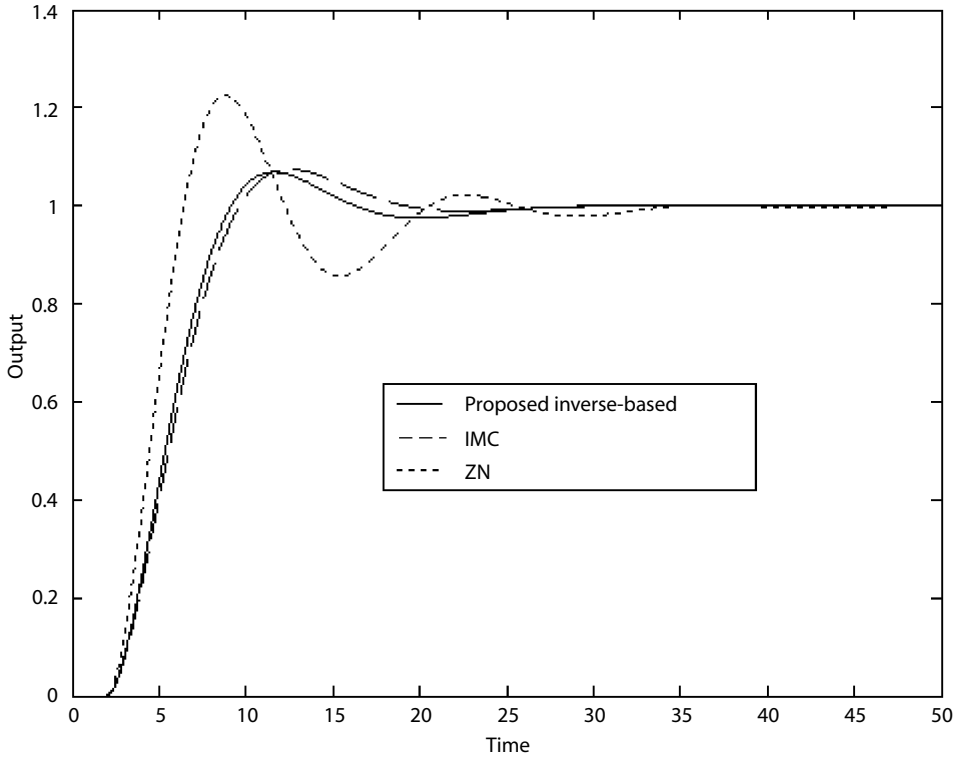


Figure 8.19 Comparison of PI control results for $G_p(s) = \dfrac{e^{-2s}}{(5s+1)(s+1)}$.

Example 8.8: Non-oscillatory high-order process – autotuning using models from relay feedback test

In Example 8.4 the on-line procedure was used to identify an SOPDT model for the following high-order plus dead time process:

$$G_p(s) = \frac{e^{-0.8s}}{(2s+1)^3(s+1)^2}$$

The outcome was the identified SOPDT model

$$\hat{G}_p(s) = \frac{e^{-3.3s}}{(3.6s+1)(2.51s+1)}$$

The PID controller took the series form

$$G_c(s) = k_p\left(1 + \frac{1}{\tau_i s}\right)\left(\frac{1+\tau_d s}{1+\tau_f s}\right)$$

and the controller parameters were found as

$k_p = 0.60$, $\tau_i = 3.58$, $\tau_d = 3.60$ and $\tau_f = 0.033$

The resulting output response to a step setpoint change is shown in Figure 8.20. The results from the other tuning rules are also given for comparison. The performance of inverse-based PID controller is satisfactory compared with those of IMC or ZN PID controllers.

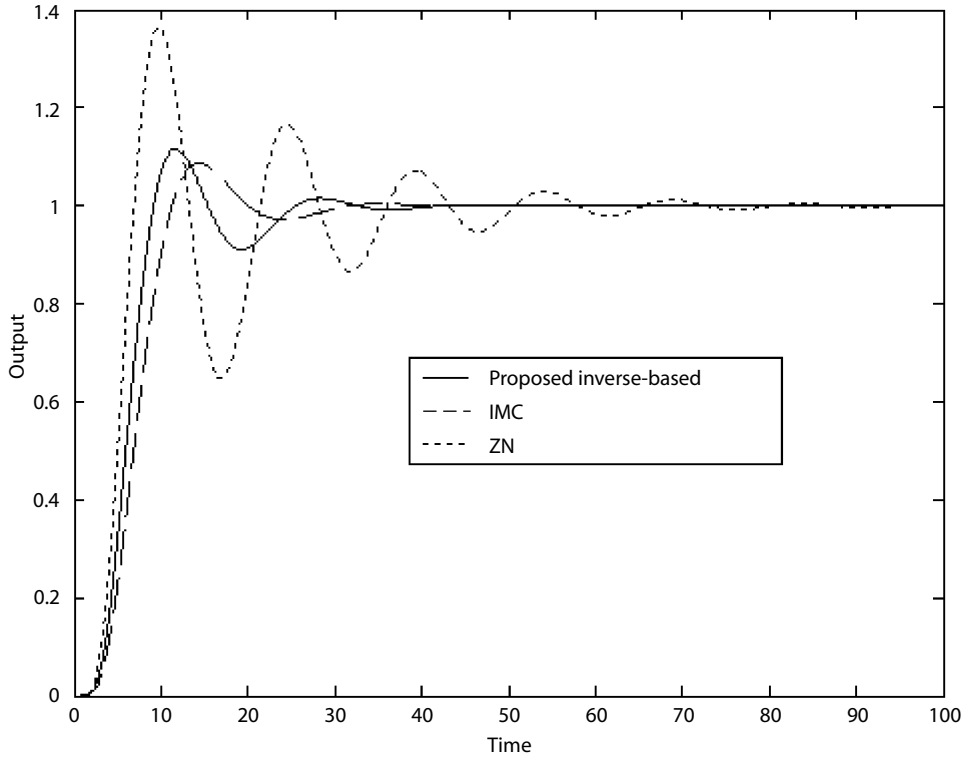


Figure 8.20 Comparison of control results for $G_p(s) = \dfrac{e^{-0.8s}}{(2s+1)^3(s+1)^2}$.

8.5 Assessment of PI/PID Control Performance

One of the common process control specifications is to provide good performance in tracking setpoint changes and rejecting output disturbances. Consequently, in real applications it would be invaluable to know how well the installed PID control system is performing against a benchmark of what is actually achievable from a particular loop. Quantifiable measures which can be used to perform an assessment of the performance of a conventional PID control system will be discussed in this section. In the spirit of the previous work in this chapter, the open-loop process in this control loop is assumed to be represented by simple models such as first-order or second-order plus dead time models. The benchmarks to be studied are the achievable optimal integral of absolute error (the so called IAE performance criterion, denoted J with optimal value denoted J^*) and the corresponding rise time t_r^* resulting from the closed-loop system tracking response. The optimal performance of the simple feedback systems is computed and represented with graphs of J^* and t_r^*. These graphs enable the performance of PID controllers to be assessed.

8.5.1 Achievable Minimal IAE Cost and Rise Time

The achievable dynamic performance of a system is important for the assessment of a current control system in operation. As with the results from the previous sections, this performance depends upon the open-loop process being controlled and the structure and tuning of the controller selected. One of the performance measures of a control system adopted for assessment in this section is the IAE value. The IAE performance criterion is defined as

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

where $e(t)$ is the error or difference between the controlled variable and its setpoint. Typically, IAE is a measure of performance since the size and length of the error in either direction is proportional to lost revenue (Shinskey, 1990).

Based on the open-loop process (FOPDT or SOPDT) and type of controller (PI or PID) used, the IAE value subject to an unit step setpoint change can be minimised by adjusting the controller parameters (denoted P) in the corresponding loop transfer function. This is equivalent to solving the following optimisation problem:

$$J = \min_P \text{IAE} = \min_P \int_0^\infty |e(t)|\,\mathrm{d}t$$

Huang and Jeng (2002) solved this optimisation problem numerically and the results were correlated into functions of process parameters. In Table 8.3 and Table 8.4, the optimal IAE values for PI and PID controllers are given as $\bar{J}^*_{\text{pi}}$ and $\bar{J}^*_{\text{pid}}$, which are normalised by time delay (thus $\bar{J}^*_{\text{pi}}$ and $\bar{J}^*_{\text{pid}}$ are optimal IAE values when time delay of the process is unity). Therefore these $\bar{J}^*_{\text{pi}}$ and $\bar{J}^*_{\text{pid}}$ values have to be multiplied by the time delay θ to become the actual optimal J^*_{pi} and J^*_{pid} values, that is, $J^*_{\text{pi}} = \bar{J}^*_{\text{pi}}\theta$ and $J^*_{\text{pid}} = \bar{J}^*_{\text{pid}}\theta$.

Table 8.3 Optimal PI control loop and IAE cost value $\bar{J}^*_{\text{pi}}$.

Process	$G_{\text{p}}(s)$	$G_{\text{lp}}(s) = G_{\text{p}}(s)G_{\text{c}}(s)$
FOPDT $\bar{\tau} = \tau/\theta \le 5$	$\dfrac{K\mathrm{e}^{-\bar{s}}}{\bar{\tau}\,\bar{s}+1}$	$\dfrac{K_0(B\bar{s}+1)}{\bar{\tau}\,\bar{s}+1}\dfrac{\mathrm{e}^{-\bar{s}}}{\bar{s}}$
	$\bar{J}^*_{\text{pi}} = 2.1038 - 0.6023\,\mathrm{e}^{-1.0695\bar{\tau}}$	
FOPDT $\bar{\tau} = \tau/\theta \le 5$	$\dfrac{K\mathrm{e}^{-\bar{s}}}{\bar{\tau}\,\bar{s}+1}$	$\dfrac{0.59\mathrm{e}^{-\bar{s}}}{\bar{s}}$
	$\bar{J}^*_{\text{pi}} = 2.1038$	
SOPDT $\zeta \le 2.0$	$\dfrac{K\mathrm{e}^{-\bar{s}}}{\bar{\tau}^2\bar{s}^2 + 2\bar{\tau}\zeta\bar{s}+1}$	$\dfrac{K_0(B\bar{s}+1)}{(\bar{\tau}^2\bar{s}^2 + 2\bar{\tau}\zeta\bar{s}+1)}\dfrac{e^{-\bar{s}}}{\bar{s}}$
	$\bar{J}^*_{\text{pi}} = \alpha(\zeta)\bar{\tau}^2 + \beta(\zeta)\bar{\tau} + \gamma(\zeta)$ $\alpha(\zeta) = 0.7444\zeta^3 - 1.4975\zeta^2 + 1.0202\zeta - 0.2525$ for $\zeta \le 0.7$ $\alpha(\zeta) = 0.0064\zeta - 0.0203$ for $0.7 < \zeta < 2.0$ $\beta(\zeta) = 1.1193\zeta^{-0.9339}$ for $\zeta \le 2.0$ $\gamma(\zeta) = -18.4675\zeta^2 + 17.9592\zeta - 2.7222$ for $\zeta \le 0.5$ $\gamma(\zeta) = -0.0995\zeta^2 + 0.4893\zeta + 1.4712$ for $0.5 < \zeta \le 2.0$	
SOPDT $\zeta > 2.0$	$\dfrac{K\mathrm{e}^{-\bar{s}}}{(\bar{\tau}_1\,\bar{s}+1)(\bar{\tau}_2\,\bar{s}+1)}$; $\bar{\tau}_1 \ge \bar{\tau}_2$	$\dfrac{K_0(B\bar{s}+1)}{(\bar{\tau}_1\,\bar{s}+1)(\bar{\tau}_2\,\bar{s}+1)}\dfrac{\mathrm{e}^{-\bar{s}}}{\bar{s}}$
	$\bar{J}^*_{\text{pi}} = -0.0173\bar{\tau}_2^2 + 1.7749\bar{\tau}_2 + 2.3514$	

Table 8.4 Optimal PID control loop and IAE cost value $\bar{J}^*_{\text{pid}}$.

Process	$G_{\text{p}}(s)$	$G_{\text{lp}}(s) = G_{\text{p}}(s)G_{\text{c}}(s)$
FOPDT $\bar{\tau} = \tau/\theta \le 3$	$\dfrac{K\text{e}^{-\bar{s}}}{\bar{\tau}\,\bar{s}+1}$	$\dfrac{K_0(A\bar{s}^2 + B\bar{s} + 1)}{\bar{\tau}\,\bar{s}+1}\dfrac{\text{e}^{-\bar{s}}}{\bar{s}}$
	$\bar{J}^*_{\text{pid}} = 1.38 - 0.1134\,\text{e}^{-1.5541\bar{\tau}}$	
FOPDT $\bar{\tau} = \tau/\theta > 3$	$\dfrac{K\text{e}^{-\bar{s}}}{\bar{\tau}\,\bar{s}+1}$	$\dfrac{0.76(0.47\bar{s}+1)\text{e}^{-\bar{s}}}{\bar{s}}$
	$\bar{J}^*_{\text{pid}} = 1.38$	
SOPDT $\zeta \le 1.1$	$\dfrac{K\text{e}^{-\bar{s}}}{\bar{\tau}^2\bar{s}^2 + 2\bar{\tau}\zeta\bar{s}+1}$	$\dfrac{K_0(A\bar{s}^2 + B\bar{s} + 1)}{(\bar{\tau}^2\bar{s}^2 + 2\bar{\tau}\zeta\bar{s}+1)}\dfrac{\text{e}^{-\bar{s}}}{\bar{s}}$
	$\bar{J}^*_{\text{pid}} = 2.1038 - \lambda(\zeta)\text{e}^{-\mu(\zeta)\bar{\tau}}$ $\lambda(\zeta) = 0.4480\zeta^2 - 1.0095\zeta + 1.2904$ $\mu(\zeta) = 6.1998\text{e}^{-3.8888\zeta} + 0.6708$	
SOPDT $\zeta > 1.1$	$\dfrac{K\text{e}^{-\bar{s}}}{(\bar{\tau}_1\,\bar{s}+1)(\bar{\tau}_2\,\bar{s}+1)};\quad \bar{\tau}_1 \ge \bar{\tau}_2$	$\dfrac{K_0(B\bar{s}+1)}{(\bar{\tau}_2\,\bar{s}+1)}\dfrac{\text{e}^{-\bar{s}}}{\bar{s}}$
	$\bar{J}^*_{\text{pid}} = 2.1038 - 0.6728\text{e}^{-1.2024\bar{\tau}_2}$	

IAE Cost Values and Rise Times for FOPDT Process Models

For FOPDT processes with PI control (see Table 8.3), the value of the optimal IAE criterion is a function of the ratio of τ to θ. Notice that the value of the optimal IAE criterion remains approximately constant at 2.1θ when $\bar{\tau} > 5$, and for $\bar{\tau} \le 5$ the value of the optimal IAE criterion lies in the range $1.7\theta \le J^*_{\text{pi}} \le 2.1\theta$. For those cases where $J^*_{\text{pi}} \approx 2.1\theta$, the reset time is approximately equal to the process time constant. In the case of FOPDT processes with PID control, a first-order filter is needed to fulfil the requirement for realisability. As a result, the value of the achievable optimal IAE criterion is a function not only of the ratio of τ to θ, but also of the time constant of the low pass filter. If this filter time constant is taken very small, the achievable IAE criterion will approach the value 1.38θ.

For both PI and PID control, the optimal rise time is obtained from simulation based on the optimal loop transfer function and is plotted as shown in Figure 8.21. For optimal PID control loops of FOPDT processes, it is found that the rise time is independent of the value of τ/θ. On the other hand, in the case of the optimal PI control loops, the rise time depends on the value of τ/θ when $\tau/\theta \le 5$, as shown in Figure 8.21.

IAE Cost Values and Rise Times for SOPDT Process Models

Similarly, for SOPDT processes with PI/PID control, the results of optimal IAE cost values are functions of parameters of open-loop processes. Notice that for the PI control loop the function of the optimal IAE cost value is divided into two parts depending on the value of ζ (see Table 8.3). The first part applies to

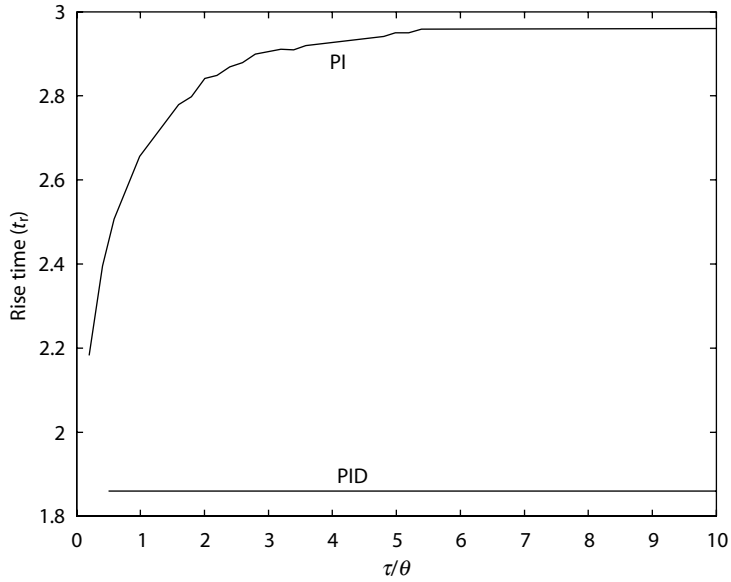


Figure 8.21 Optimal rise time t_r^* of a PI/PID control system for the FOPDT process model.

those systems which have $\zeta \leq 2.0$, where $\bar{J}_{pi}^{\star}$ is a function of $\bar{\tau}$ and ζ. The second part applies to those $\zeta > 2.0$, where $\bar{J}_{pi}^{\star}$ is only a function of the ratio of minor time constant to time delay $\bar{\tau}_2$. The IAE cost function values for the PID controller are also found to be given in two parts (see Table 8.4), that is for $\zeta \leq 1.1$ and for $\zeta > 1.1$. The rise times of these optimal systems are also obtained from simulation based on corresponding optimal loop transfer functions, and the results are plotted in Figure 8.22.

In fact, to justify which type of controller fits better to control a given process the system dynamic is not the only issue of concern. From the dynamical point of view, the actual achievable minimum IAE criterion value resulting from two designs based on different dynamic models can be compared. Although both FOPDT and SOPDT models can be used to model the same open-loop dynamics, the apparent dead time of the former is usually larger than that of the latter. Thus by applying the formulae in Table 8.3 and Table 8.4, two achievable minimal IAE values can be obtained and compared.

8.5.2 Assessment of PI/PID Controllers

PI/PID control systems have been widely used in the industrial process control and the performance of these control systems is dependent on the dynamics of the open-loop process. Huang and Jeng (2002) indicate that if the controller in a simple feedback loop has a general form, not restricted to the PI/PID controller, the achievable minimum IAE criterion value is found to be 1.38θ. To understand how well a PI/PID control system can perform referring to the optimal system with general controllers, an efficiency factor of the optimal PI/PID control systems (designated as η^*) for FOPDT and SOPDT processes is computed according to the formula

$$\eta^* = \frac{1.38\theta}{J_{pi}^{*}(\text{or } J_{pid}^{*})}$$

It is found that for open-loop processes with second-order plus delay time dynamics the efficiency of PID control is mostly around 65% only, but for FOPDT processes the efficiency is very close to 100%. Thus, compared with the optimal simple feedback system, the PID controller is not as efficient for

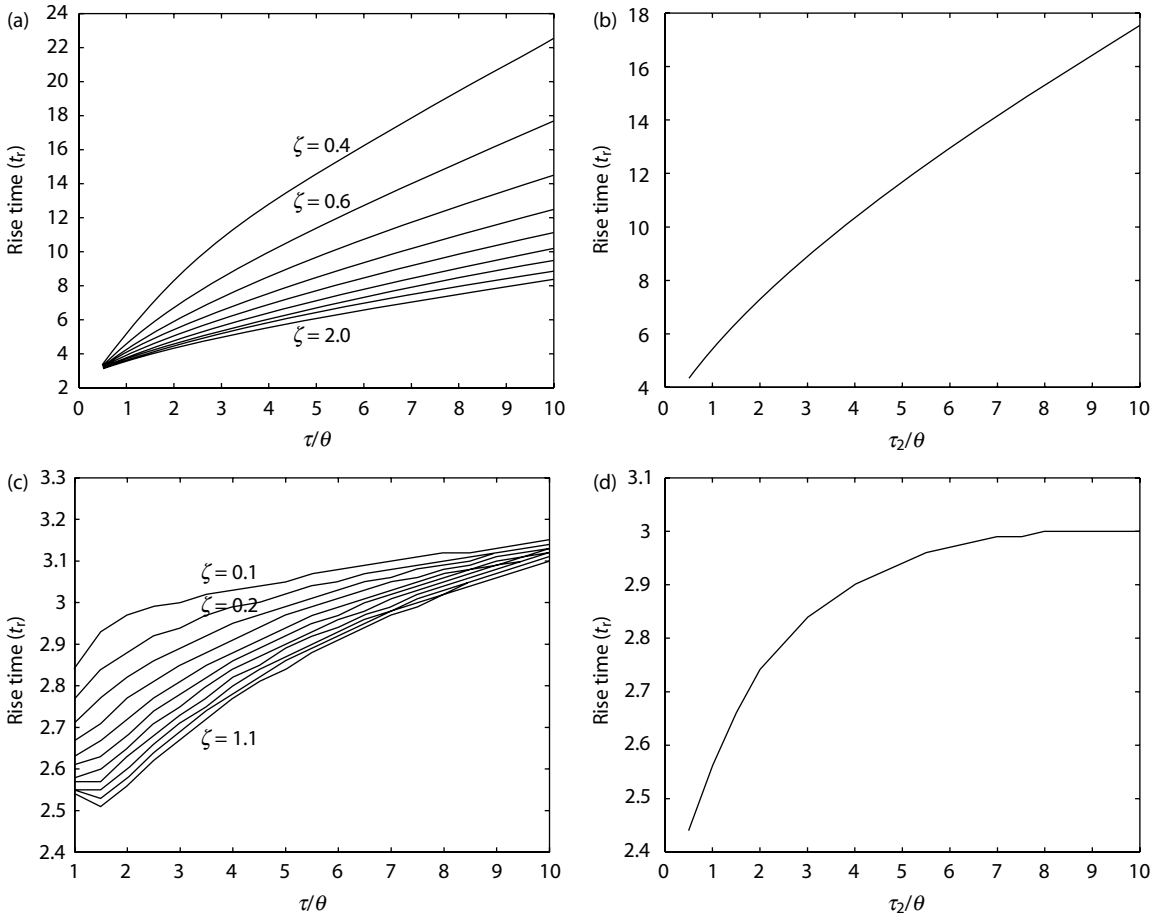


Figure 8.22 Optimal rise time t_r^* of PI/PID control systems for SOPDT processes. (a) Process G_p, $\zeta \le 2.0$; Controller G_c, PI. (b) Process G_p, $\zeta > 2.0$; Controller G_c, PI. (c) Process G_p, $\zeta \le 1.1$; Controller G_c, PID. (d) Process G_p, $\zeta > 1.1$; Controller G_c, PID.

SOPDT processes. On the other hand, the PID controller has the best efficiency for controlling FOPDT processes.

The graphs of $\bar{J}^*_{\text{pi}}$ and $\bar{J}^*_{\text{pid}}$ versus t_r^* as shown in Figures 8.23 and 8.24 can be used to highlight the performance of PI/PID loops. The status of the performance of a particular PI/PID control loop can be ascertained by computing the actual value for the loop integrated absolute error criterion $\bar{J}$, recording the actual loop rise time t_r and then locating the point $(\bar{J}, t_r)$ on the appropriate figure. If it is found that the point $(\bar{J}, t_r)$ is far away from the optimal region then the system is not performing well and re-tuning should be considered. Further, the actual location of the point will give an indication of the weakness of the control system. For example, if the point lies beneath the optimal region, it means that the tracking error is due to the response being too fast. It might be thought that the assessment requires knowledge of the open-loop dynamics, but from Figures 8.23 and 8.24 it can be seen that the optimal regions form fairly narrow bands. Consequently, the performance indicators should not be so sensitive to the actual process parameters; thus as long as the point $(\bar{J}, t_r)$ is located on the bands of the corresponding controller type, the performance of the system is close to optimal.

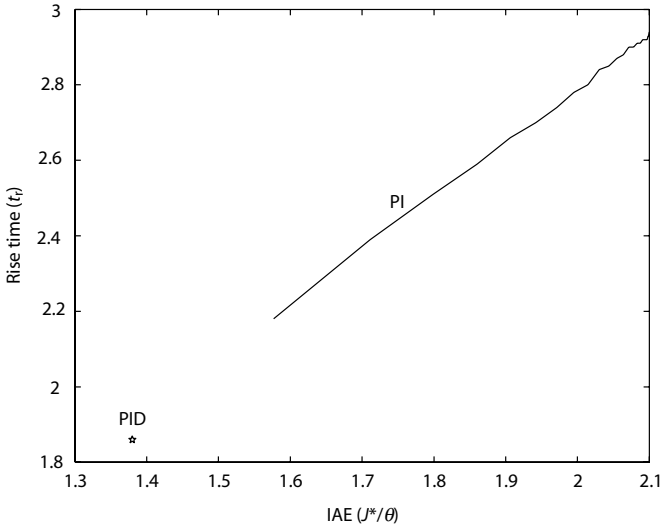


Figure 8.23 Graph of t_r^* vs. J^*/θ for optimal PI/PID control system of FOPDT processes.

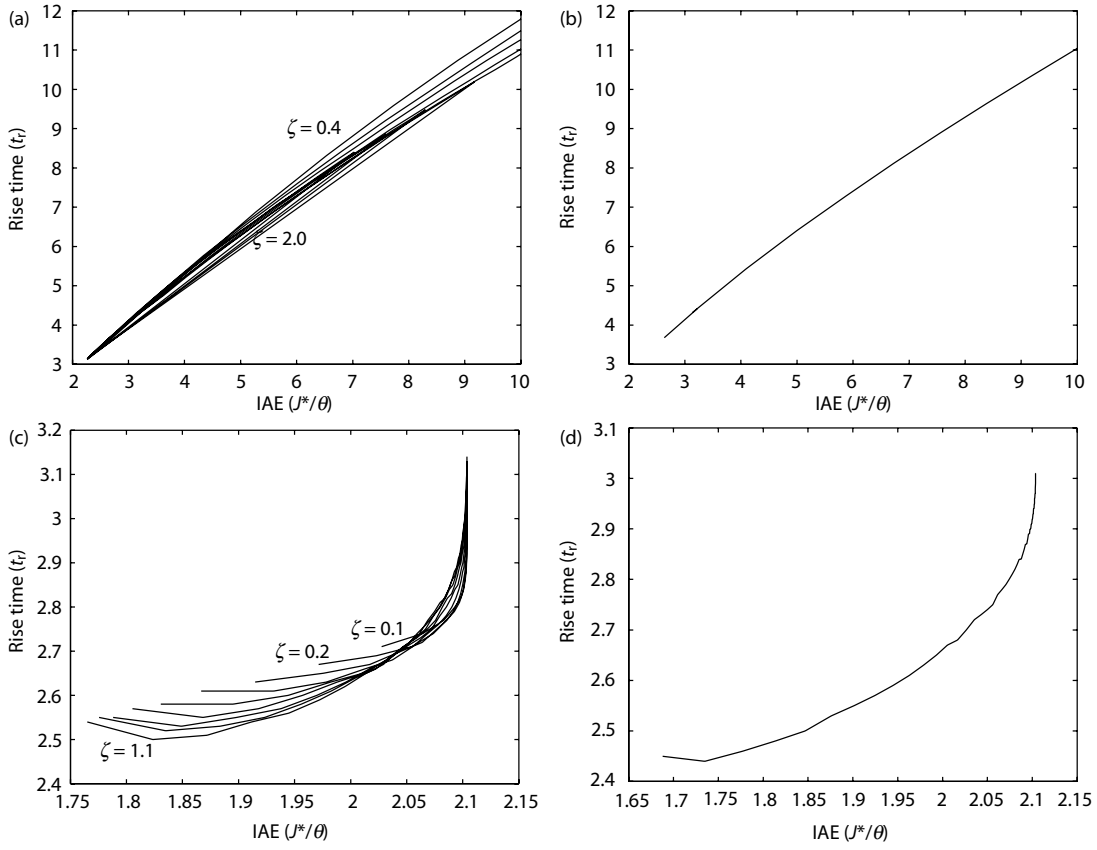


Figure 8.24 Graphs of t_r^* vs. J^*/θ for optimal PI/PID control system of SOPDT processes. (a) Process G_p, $\zeta \leq 2.0$; controller G_c, PI. (b) Process G_p, $\zeta > 2.0$; controller G_c, PI. (c) Process G_p, $\zeta \leq 1.1$; controller G_c, PID. (d) Process G_p, $\zeta > 1.1$; controller G_c, PID.

Example 8.9: Assessment of PID controller

Consider a PID control system for a process with the following model:

$$G_p(s) = \frac{e^{-s}}{4s^2 + 3.2s + 1}$$

The system starts with a PID controller with the following parameters:

$$k_p = 3.0,\ \tau_i = 3.0,\ \tau_d = 1.0$$

Both $\bar{J}$ and t_r are computed from the formula given in Table 8.5, and then the point $(\bar{J}, t_r)$ is located using the graph of Case (c) in Figure 8.24. It is found that the point is far off the optimal region, as shown in Figure 8.25 (point 1). Moreover, its location indicates that the large *IAE* value is due to the response being too fast; namely, the rise time is too small. For this reason, the controller is re-tuned to be more conservative to the second setting by decreasing k_p and increasing τ_i as given in Table 8.5. Although the *IAE* value is smaller than the first one, this setting is still not optimal due to its sluggish response (point 2 in Figure 8.25). In this manner, the parameters of PID controller are changed to the third and fourth sets as given in Table 8.5. The location of $(\bar{J}, t_r)$ as shown in Figure 8.25 indeed implies the status of the performance at each stage. Since the point $(\bar{J}, t_r)$ of the fourth setting falls into the optimal region, it is concluded that the current control system is close to optimal.

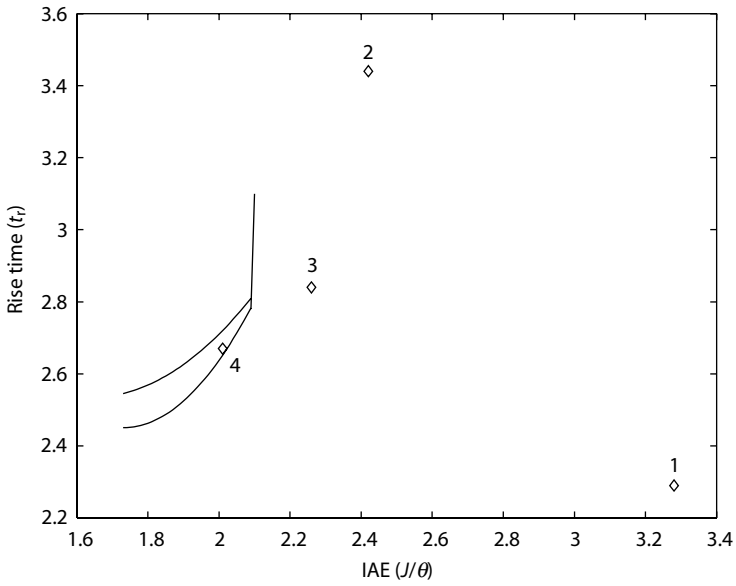


Figure 8.25 Assessment of PID control performance in Example 8.9.

Table 8.5 Parameters of PID controller and control performance in Example 8.9.

Setting	k_p	τ_i	τ_d	$\bar{J}$	t_r
Initial (1)	3.0	3.0	1.0	3.16	2.23
Second (2)	1.8	4.0	1.0	2.50	3.44
Third (3)	2.0	3.5	1.2	2.19	2.86
Fourth (4)	2.0	3.2	1.5	2.03	2.60

References

Åström, K.J. and Hägglund, T. (1988) *Automatic Tuning of PID Controllers*. Instrument Society of America, Research Triangle Park, NC.

Åström, K.J. and Hägglund, T. (1995) *PID Controllers: Theory, Design and Tuning*. Instrument Society of America, Research Triangle Park, NC.

Chien, I.L. and Fruehauf, P.S. (1990) Consider IMC tuning to improve controller performance. *Chemical Engineering Progress*, October, 33–41.

Hang, C.C., Åström, K.J. and Wang, Q.G. (2002) Relay feedback auto-tuning of process controllers – a tutorial review. *J. Proc. Cont.*, **12**, 143–162.

Hang, C.C., Åström, K.J. and Ho, W.K. (1991) Refinement of the Ziegler–Nichols tuning formula. *IEE Proceedings-D*, **138**, 111–118.

Hang, C.C., Lee, T.H. and Ho, W.K. (1993) *Adaptive Control*. Instrument Society of America, Research Triangle Park, NC.

Harriott, P. (1964) *Process Control*. McGraw-Hill, New York.

Huang, C.T. and Chou, C.J. (1994) Estimation of the underdamped second-order parameters from the system transient. *Ind. Eng. Chem. Res.*, **33**, 174–176.

Huang, C.T. and Huang, M.F. (1993) Estimating of the second-order parameters from process transient by simple calculation. *Ind. Eng. Chem. Res.*, **32**, 228–230.

Huang, C.T. and Clements Jr, W.C. (1982) Parameter estimation for the second-order-plus-dead-time model. *Ind. Eng. Chem. Process Des. Dev.*, **21**, 601–603.

Huang, H.P. and Jeng, J.C. (2002) Monitoring and assessment of control performance for single loop systems. *Ind. Eng. Chem. Res.*, **41**, 1297–1309.

Huang, H.P. and Jeng, J.C. (2003) Identification for monitoring and autotuning of PID controllers. *J. Chem. Eng. Japan*, **36**, 284–296.

Huang, H.P., Lee, M.W. and Chen, C.L. (2001) A system of procedures for identification of simple models using transient step response. *Ind. Eng. Chem. Res.*, **40**, 1903–1915.

Huang, H.P., Lee, M.W. and Chien, I.L. (2000) Identification of transfer-function models from the relay feedback tests. *Chem. Eng. Commun.*, **180**, 231–253.

Li, W., Eskinat, E. and Luyben, W.L. (1991) An improved autotune identification method. *Ind. Eng. Chem. Res.*, **30**, 1530–1541.

Luyben, W.L. (2001) Getting more information from relay-feedback tests. *Ind. Eng. Chem. Res.*, **40**, 4391–4402.

McMillan, G.K. (1994) *Tuning and Control Loop Performance, A Practitioner Guide*, 3rd edn. Instrument Society of America, Research Triangle Park, NC.

Oldenbourg, R.C. and Sartorius, H. (1948) The dynamics of automatic control. *Trans. ASME*, **77**, 75–79.

Shinskey, F. (1990) How good are our controllers in absolute performance and robustness? *Measurement and Control*, **23**, 114–121.

Sundareesan, K.R., Chandra Prasad, C. and Krishnaswamy, C. (1978) Evaluating parameters for process transients. *Ind. Eng. Chem. Process Des. Dev.*, **17**, 237–241.

Sung, S.W., Jungmin, O., Lee, I.B., Lee, J. and Yi, S.H. (1996) Automatic tuning of PID controller using second-order plus time delay model. *J. Chem. Eng. Japan*, **6**, 990–999.

Wang, Q.G., Guo, X. and Zhang, Y. (2001) Direct identification of continuous time delay systems from step response. *J. Proc. Contr.*, **11**, 531–542.
Yu, C.C. (1999) *Autotuning of PID Controllers*. Springer-Verlag, London.