5. Relay Feedback and its Variations

Suppose that a relay feedback system is stable and eventually yields limit cycles. The information contained in the limit cycles can be used for process frequency response estimation, as pioneered by Åström and co-workers. Fundamentals are first provided in Section 5.1, followed by their refinements in the subsequent sections, which expand applicability to more scenarios. Note that in this chapter, only stationery oscillations are used for process identification while the following chapter involves relay transient response. This chapter focuses on non-parametric models while Chapter 7 addresses conversion from frequency responses to transfer function models.

5.1 Fundamentals

Consider a single-input single-output process described by

$$\begin{aligned} \dot{x}(t) &= Ax(t) + bu(t-L), \\ y(t) &= cx(t), \end{aligned} \tag{5.1}$$

where $x(t) \in \mathbb{R}^n$, $y(t) \in \mathbb{R}$ and $u(t-\tau) \in \mathbb{R}$ are the state, output and control input, respectively; A, b, c are constant real matrices or vectors with appropriate dimensions; $L \geq 0$ indicates the time delay. More often, we will use the transfer function representation of the process:

$$Y(s) = G(s)U(s), \tag{5.2}$$

where

$$G(s) = G_0(s)e^{-Ls},$$

with $G_0(s) = c(sI - A)^{-1}b$ being a strictly proper rational function. Let $r(t)$ be the reference or the set-point for the process output $y(t)$ to track. The error between them is

$$e(t) = r(t) - y(t). \tag{5.3}$$

Assume that the process is controlled by relay feedback:

$$u(t) = \begin{cases} \mu_+, \text{ if } e(t) > \varepsilon_+, \text{ or } e(t) \geq \varepsilon_- \text{ and } u(t_-) = \mu_+, \\ \mu_-, \text{ if } e(t) < \varepsilon_-, \text{ or } e(t) \leq \varepsilon_+ \text{ and } u(t_-) = \mu_-, \end{cases} \tag{5.4}$$

where $\varepsilon_+, \varepsilon_- \in \mathbb{R}$ with $\varepsilon_- \leq \varepsilon_+$ indicating hysteresis; $\mu_-, \mu_+ \in \mathbb{R}$ and $\mu_- \neq \mu_+$. For easy reference, the relay function in (5.4), which maps $e(t)$ to $u(t)$, is denoted by $\rho(\epsilon_+, \epsilon_-, \mu_+, \mu_-)$. Its functionality is shown in Figure 5.1. A relay is said to have hysteresis if $\varepsilon_+ \neq 0$ or $\varepsilon_- \neq 0$; and to be symmetric if $\varepsilon_+ = \varepsilon$, $\varepsilon_- = -\varepsilon$, and $\mu_+ = \mu$, $\mu_- = -\mu$, denoted by $\rho(\varepsilon, \mu)$; otherwise, it is called a biased relay. The standard relay corresponds to a symmetric relay with no hysteresis and is denoted by $\rho(\mu)$.

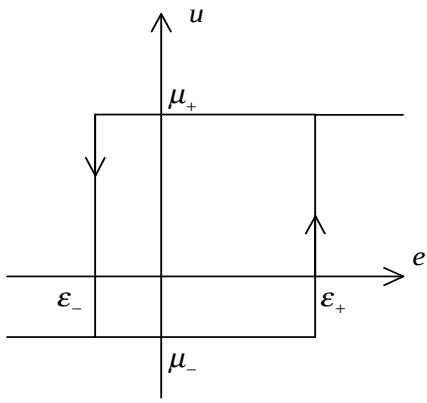


Fig. 5.1. General relay function $\rho(\varepsilon_+, \varepsilon_-, \mu_+, \mu_-)$

Due to time delay $L \geq 0$, we have to specify the initial function $u(\tilde{t})$ for $\tilde{t} \in [-L,\ 0]$. The most natural one, which is also used in practice, is

$$u(\tilde{t}) \equiv \begin{cases} \mu_+, & \text{if } e(0) > \varepsilon_+, \\ \mu_-, & \text{if } e(0) < \varepsilon_-, \\ u_0 \in \mathcal{U}, & \text{if } \varepsilon_- \leq e(0) \leq \varepsilon_+, \end{cases} \tag{5.5}$$

where

$$\mathcal{U} := \{\mu_-,\ \mu_+\}. \tag{5.6}$$

This completes the description of a linear process with relay feedback control.

We call (5.1)–(5.6) a relay feedback system (abbreviated as RFS), denote it by Σ_L, and depict it in Figure 5.2. Experience shows that a RFS is likely

to have limit cycle oscillations as its steady state. Readers are referred to Part I of this book for a detailed analysis of the existence and stability of limit cycles in RFS and to the next section for the simple case of first-order systems to get a rough idea on them. In relay feedback experiments for analysis and identification, the set-point r is always kept unchanged, and we thus assume $r(t) = 0$ throughout this chapter. Assume that a limit cycle results from a RFS. Our task for this chapter is to extract the process dynamic information from such a limit cycle. We will start with the simplest method, the describing function approximation.

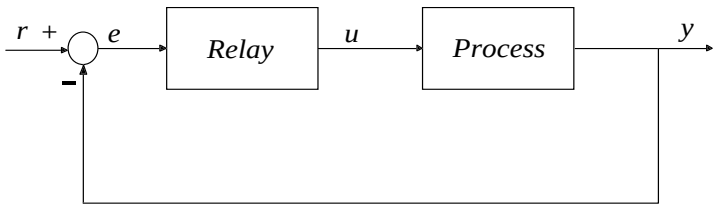


Fig. 5.2. Relay feedback system

Describing Function Method The describing function method approximates the relay with an "equivalent" linear time-invariant system. To this end, the input to the relay is assumed to be sinusoidal:

$$e(t) = a \sin \omega t,$$

and the resulting signals in the overall system are analyzed. Consider first the standard relay case. Then, the relay output $u(t)$ in response to $e(t)$ would be a square wave having a frequency ω and an amplitude equal to the relay output level μ. Using Fourier's series expansion, the periodic $u(t)$ can be written as

$$u(t) = \frac{4\mu}{\pi} \sum_{k=1}^{\infty} \frac{\sin(2k-1)\omega t}{2k-1}.$$

The describing function (DF) of the relay, $N(a)$, is simply the complex ratio of the fundamental component of $u(t)$ to the input sinusoid, i.e.

$$N(a) = \frac{4\mu}{\pi a}.$$

One sees that the DF analysis ignores harmonics beyond the fundamental component. The residual ϱ is the entire sinusoidally-forced relay output minus the fundamental component, i.e. the part of the output that is ignored in the DF development,

$$\varrho = \frac{4\mu}{\pi} \sum_{k=2}^{\infty} \frac{\sin(2k-1)\omega t}{2k-1}.$$

In the DF analysis of the relay feedback system, the relay is replaced with its quasi-linear equivalent DF, and a self-sustained oscillation of amplitude a and frequency ω_c is assumed. Then, for the process with the transfer function $G(s)$, it follows from Figure 5.2 that the variables in the loop satisfy the following relations,

$$E = -Y,$$
$$U = N(a)E,$$
$$Y = G(j\omega_c)U.$$

This implies

$$G(j\omega_c) = -\frac{1}{N(a)} = -\frac{\pi a}{4\mu}, \tag{5.7}$$

which gives an estimation of the process frequency response at one frequency, the RFS oscillation frequency.

The above DF analysis assumes that the Nyquist curve of $G(j\omega)$ intersects with the real axis at $-\frac{1}{N(a)}$ at ω_c in the complex plane. Recall that the intersection point of a process Nyquist curve with the real axis is called the critical point of the process and defines the critical or ultimate frequency, ω_u, of the process, for which

$$arg\{G(j\omega_u)\} = -\pi. \tag{5.8}$$

We can thus estimate the ultimate frequency and ultimate gain k_u by

$$\omega_u = \omega_c,$$

and

$$k_u := \frac{1}{|G(j\omega_u)|} = \frac{1}{|G(j\omega_c)|} = \frac{4\mu}{\pi a}.$$

For experiment design, the standard relay has only one parameter to tune, the relay output amplitude μ. Large μ will cause strong excitation of the process

and thus better identification. On the other hand, a large signal will make the process output deviate further from its set-point, which is not desirable. The choice of μ is thus a trade-off between identification and control performance, as always in any identification problem, and much depends on measurement noise in the process output. If permissible, the relay output level should be adjusted such that the oscillation amplitude of the process output is about three times as large as its noise band. If the adjustment is not possible for a limited testing time, μ may be set to 3–10 % of the maximum range of the manipulated variable.

For preservation of regular relay switchings and estimating robustness against noise, it may be advantageous to replace a standard relay by a symmetric relay with suitable hysteresis so that the resultant system is less sensitive to measurement noise. The describing function of a (symmetric) hysteretic relay, $\rho(\epsilon, \mu)$, is

$$N(a) = \frac{4\mu}{\pi(\sqrt{a^2 - \epsilon^2} + j\epsilon)}.$$

The process frequency response, like the standard relay case, is estimated as the inverse negative describing function of the new relay

$$G(j\omega_c) = -\frac{1}{N(a)} = -\frac{\pi}{4\mu}\left(\sqrt{a^2 - \epsilon^2} + j\epsilon\right).$$

In this case, the oscillation corresponds to the point where the negative inverse describing function of the relay crosses the Nyquist curve of the process as shown in Figure 5.3.

Noise is always present in output measurements and a big concern for process identification since it uses noisy data from measurements. As mentioned, hysteresis in the relay is a simple way to reduce the influence of measurement noise. Before a relay test is performed, the noise bank in the process output measurements can be estimated using simple statistics. The hysteresis width, ϵ, should be greater than the noise band to avoid wrong switchings in the relay output and is usually chosen to be twice the noise band (Hang *et al.*, 1993*b*) so that reliable stationery oscillations can be produced, maintained and observed easily. Filtering is another possible anti-noise measure (Åström and Wittenmark, 1984). Note that noise is usually of high frequency while most processes are of low-pass nature. A low-pass filter may be employed to pre-process noisy output and the pre-processed data are then used for model estimation. The filter bandwidth is usually set to be 3–5 times higher than the process critical frequency. Yet another anti-noise measure is to utilize multiple periods of

limit cycles instead of a single period so as to filter out noise by the averaging method. A detailed and quantified analysis of this is given in Section 5.3, where the disturbance issue is also addressed.

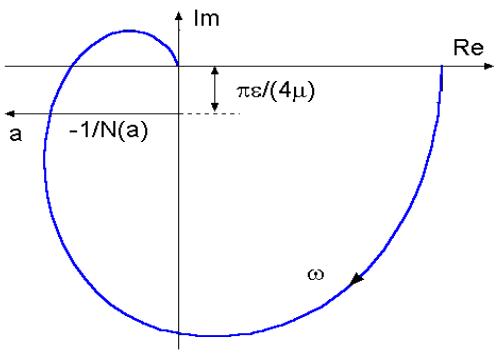


Fig. 5.3. Negative inverse describing function of hysteretic relay

Fourier Series Method The oscillation waveform of the relay input signal $e(t)$ under a RFS is usually not precisely sinusoidal as assumed in the DF analysis above, which will thus cause estimation error. The error increases as the waveform differs from the sinusoidal function. For a linear process, this error can easily be removed by extracting the fundamental harmonics of both input and output of the linear process $G(s)$ using the Fourier series method. The resulting formula for estimating the process frequency response at the oscillation frequency is

$$G(j\omega_c) = \frac{Y(j\omega_c)}{U(j\omega_c)} = \frac{\int_{1period} y(t)e^{-j\omega_c t}dt}{\int_{1period} u(t)e^{-j\omega_c t}dt}, \tag{5.9}$$

where $y(t)$ and $u(t)$ are one period of the process output and input stationery oscillations, respectively. This formula holds for a general relay and is precise if the system does not have any noise or disturbance.

Static Gain Estimation If the relay used is symmetric, the resulting limit cycles will also be symmetric. No DC components are contained in such oscillations. This enables one-point estimation only, as shown above. To further estimate the DC gain or static gain of the process, we may introduce a bias, either in the relay input ($\epsilon_+ \neq \epsilon_-$), or relay output ($\mu_+ \neq \mu_-$), or both, so as to create an asymmetric relay and asymmetric limit cycles in the process

output. If this is the case, the static gain can be obtained using the Fourier series expansion again as

$$G(0) = \frac{Y(0)}{U(0)} = \frac{\int_{1period} y(t)dt}{\int_{1period} u(t)dt}. \tag{5.10}$$

5.2 First-order Modelling

First-order plus dead time (FOPDT) transfer functions are often used in process modelling and control because of its simplicity although actual processes could be of high order. In general, relay feedback systems are a hard problem for theoretical analysis, see Part I of this book. For FOPDT processes, however, we are able to establish the complete results. They are presented in this section and give some idea of the existence of solutions, the existence of limit cycles, and the stability of limit cycles for relay feedback systems, without much mathematics. They also provide some feeling and insight into what will happen to relay feedback systems. The information on limit cycles is adequate to determine the FOPDT model and this is also covered in this section, after relay feedback theory.

Relay Feedback Theory Let the process be represented by the first-order plus dead time transfer function,

$$G(s) = \frac{K}{\tau s + 1} e^{-Ls}, \tag{5.11}$$

or in terms of a state space model,

$$\begin{aligned} \dot{x}(t) &= ax(t) + bu(t - L), \\ y(t) &= cx(t), \end{aligned} \tag{5.12}$$

where for a non-integral process with $\tau \neq \infty$, we have

$$a = -\frac{1}{\tau}, \quad cb = \frac{K}{\tau}, \tag{5.13}$$

while for an integral process with $\tau = \infty$, (5.11) reduces to $G(s) = \frac{\kappa}{s} e^{-Ls}$ so that

$$a = 0, \quad cb = \kappa. \tag{5.14}$$

Suppose that the process is under general relay feedback control as described by (5.4). Since $r(t) = 0$, the relay is given by

$$u(t) = \begin{cases} \mu_-, \text{ if } y(t) > -\varepsilon_-, \text{ or } y(t) \geq -\varepsilon_+ \text{ and } u(t_-) = \mu_-, \\ \mu_+, \text{ if } y(t) < -\varepsilon_+, \text{ or } y(t) \leq -\varepsilon_- \text{ and } u(t_-) = \mu_+, \end{cases} \tag{5.15}$$

and the initial condition is

$$u(\tilde{t}) \equiv \begin{cases} \mu_-, & \text{if } y(0) > -\varepsilon_-, \\ \mu_+, & \text{if } y(0) < -\varepsilon_+, \\ u_0 \in \mathcal{U}, & \text{if } -\varepsilon_+ \le y(0) \le -\varepsilon_-. \end{cases} \tag{5.16}$$

The resulting relay feedback system is rather simple and enables us to easily analyze it completely. It turns out that the results depend on the nature of the parameter τ, whether it is positive, negative or zero, and have to be presented separately for these three different cases.

In what follows, define the switching planes (which are "switching lines" for the present case of the first-order system):

$$\mathcal{S}_+ := \{\xi \in R : \ c\xi = -\varepsilon_-\}, \tag{5.17}$$

$$\mathcal{S}_- := \{\xi \in R : \ c\xi = -\varepsilon_+\}. \tag{5.18}$$

Proposition 5.2.1. *Consider a RFS for the stable process in the form of* (5.11) *with* $\tau > 0$ *(i.e.,* $a < 0$ *in (5.12)).*

(i) A unique solution exists for any given initial condition if and only if any of the following holds.

(a) $L > 0$,

(b) $L = 0$ *and* $-\varepsilon_+ > \max\{K\mu_-, \ K\mu_+\}$,

(c) $L = 0$ *and* $-\varepsilon_- < \min\{K\mu_-, \ K\mu_+\}$,

(d) $L = 0$ *and* $K\mu_+ \le -\varepsilon_-$ *and* $K\mu_- \ge -\varepsilon_+$.

(ii) A limit cycle exists if and only if $L > 0$ *and* $K\mu_+ > -\varepsilon_- \ge -\varepsilon_+ > K\mu_-$. *If this is the case, the limit cycle is unique with two switchings per period.*

(iii) If a limit cycle exists, then the limit cycle is globally stable. Moreover, for a given process, the limit cycle is the common trajectory after the first switch, independent of the initial conditions.

Proof. We first show (i). For $L > 0$, it is easy to show that there exists a unique solution for any given initial condition. We now concentrate on the case for $L = 0$. Let the initial state x_0 satisfy $cx_0 \ge -\varepsilon_+$ and the relay start at μ_-. Then the trajectory of $x(t)$ will be governed by

$$x(t) = e^{at}(x_0 + ba^{-1}\mu_-) - ba^{-1}\mu_-. \tag{5.19}$$

It is easy to see that if $-\varepsilon_+ > K\mu_-$, then $x(t)$ will intersect $\mathcal{S}_-$ at some instant $t_1 > 0$. However, if $-\varepsilon_+ \le K\mu_+$, after $t = t_1$, the trajectory $x(t)$ cannot evolve. Otherwise, for $t > 0$, we have

$$y(t_1 + t) = cx(t_1 + t) = \begin{cases} e^{at}(-\varepsilon_+ - K\mu_+) + K\mu_+ \ge -\varepsilon_+, & \text{for } u = \mu_+ \\ e^{at}(-\varepsilon_+ - K\mu_-) + K\mu_- < -\varepsilon_+, & \text{for } u = \mu_-, \end{cases}$$

which contradicts the control law (5.15). If $-\varepsilon_+ > K\mu_-$ and $-\varepsilon_+ > K\mu_+$ after the instant $t = t_1$, the trajectory will be governed by $x(t) = e^{at}(x_1 + ba^{-1}\mu_+) - ba^{-1}\mu_+$. Next, if $-\varepsilon_+ \leq K\mu_-$, we also check that if $-\varepsilon_- \geq K\mu_+$ holds, a unique solution exists for any initial condition. For $-\varepsilon_+ \leq K\mu_-$, if $-\varepsilon_- < K\mu_+$, then a similar analysis leads to a unique solution for any initial condition if $-\varepsilon_- < K\mu_-$ also holds. So far, (i) is proved.

Next we show (ii) and (iii). It is seen from the above that for $L = 0$, there is no limit cycle since the solution, if any, tends to $K\mu_-$ or $K\mu_+$. For $L > 0$, if and only if $K\mu_+ > -\varepsilon_- \geq -\varepsilon_+ > K\mu_-$, can the relay switch continuously. Moreover, any trajectory $x(t)$ will traverse $\mathcal{S}_-$ and $\mathcal{S}_+$ at fixed points $-\varepsilon_+/c$ and $-\varepsilon_-/c$, respectively. This proves (ii) and (iii).

Proposition 5.2.2. *Consider a RFS for the unstable process in the form* (5.11) *with* $\tau < 0$ *(i.e.,* $a > 0$ *in (5.12)).*

(i) A unique solution exists for any given initial condition if and only if any of the following holds.

(a) $L > 0$,

(b) $L = 0$ *and* $-\varepsilon_+ < \min\{K\mu_-, \quad K\mu_+\}$,

(c) $L = 0$ *and* $-\varepsilon_- > \max\{K\mu_-, \quad K\mu_+\}$,

(d) $L = 0$ *and* $K\mu_+ \geq -\varepsilon_- \geq -\varepsilon_+ \geq K\mu_-$.

(ii) A limit cycle exists if and only if $K\mu_+ < -\varepsilon_+ \leq -\varepsilon_- < K\mu_-$ *and*

$$0 < L < \min\left\{-\tau \ln \frac{K(\mu_+ - \mu_-)}{-\varepsilon_+ - K\mu_-}, \quad -\tau \ln \frac{K(\mu_- - \mu_+)}{-\varepsilon_- - K\mu_+}\right\}.$$

If this is the case, the limit cycle is unique with two switchings per period.

(iii) If a limit cycle exists, then the limit cycle is locally stable, and the stability range is $K\mu_+ < cx(0) < K\mu_-$. *Moreover, for the given process, the limit cycle is the common trajectory after the first switch, independent of the initial conditions in the stability range.*

Proof. We first show (i). For $L > 0$, it is easy to show that there exists a unique solution for any given initial condition. We now concentrate on the case $L = 0$. Let the initial state x_0 satisfy $cx_0 \geq -\varepsilon_+$ and the relay start at μ_-. Then the trajectory of $x(t)$ will be governed by

$$x(t) = e^{at}(x_0 + ba^{-1}\mu_-) - ba^{-1}\mu_-. \tag{5.20}$$

Since $a > 0$, it is easy to see that if $-\varepsilon_+ < K\mu_-$, then for $cx_0 \geq K\mu_-$, the relay will remain μ_- for all $t \geq 0$; and for $cx_0 < K\mu_-$, $x(t)$ will intersect $\mathcal{S}_-$ at some instant $t_1 > 0$. However, if $-\varepsilon_+ \geq K\mu_+$, after $t = t_1$, the trajectory $x(t)$ cannot evolve. Otherwise, for $t > 0$, we have

$$y(t_1+t) = cx(t_1+t) = \begin{cases} e^{at}(-\varepsilon_+ - K\mu_+) + K\mu_+ \geq -\varepsilon_+, & \text{for } u = \mu_+ \\ e^{at}(-\varepsilon_+ - K\mu_-) + K\mu_- < -\varepsilon_+, & \text{for } u = \mu_-, \end{cases}$$

which contradicts the control law (5.15). If $-\varepsilon_+ < K\mu_-$ and $\varepsilon_- \leq K\mu_+$, after the instant $t = t_1$, the trajectory will be governed by $x(t) = e^{at}(x_1 + ba^{-1}\mu_+) - ba^{-1}\mu_+$. Next, if $-\varepsilon_+ \geq K\mu_-$, we check that if also $-\varepsilon_- \leq K\mu_+$ holds, a unique solution exists for any initial condition. For $-\varepsilon_+ \geq K\mu_-$, if $-\varepsilon_- > K\mu_+$, then a similar analysis leads to a unique solution for any initial condition if $-\varepsilon_- > K\mu_-$ also holds. So far, (i) is proved.

Next we show (ii) and (iii). It is seen from the above that for $L = 0$, there is no limit cycle since the solution, if any, tends to $+\infty$ or $-\infty$, or is equivalent to $K\mu_-/c$ or $K\mu_+/c$. We now concentrate on $L > 0$. Without loss of generality, assume that $cx_0 > -\varepsilon_-$. It is easy to see, as for the case $L = 0$, that if $cx_0 \geq K\mu_-$, then the trajectory $x(t)$ starting from the initial condition x_0 exists for all $t \geq 0$ while it does not make the relay switch (i.e., $u(t) \equiv \mu_-$). Let the initial state x_0 satisfy $K\mu_- > cx_0 > -\varepsilon_-$. Then the trajectory of $x(t)$ will be governed by

$$x(t) = e^{at}(x_0 + ba^{-1}\mu_-) - ba^{-1}\mu_-$$

until for some time $t_1 > 0$, it satisfies $cx(t_1) = -\varepsilon_+$. After $t = t_1$, due to the time delay $L > 0$, the trajectory will satisfy

$$x(t_1+t) = e^{at}(x(t_1) + ba^{-1}\mu_-) - ba^{-1}\mu_-, \quad 0 \leq t \leq L$$

before the switch occurs at $t = L$. It is easy to check that $cx(t_1+t) < -\varepsilon_+$ for all $t \in (0, L]$. After time $t_1 + L$, the trajectory of $x(t)$ will be governed by

$$x(t_1+L+t) = e^{at}(x(t_1+L) + ba^{-1}\mu_+) - ba^{-1}\mu_+. \tag{5.21}$$

Similarly, the switch will occur if and only if

$$cx(t_1+L) + cba^{-1}\mu_+ > 0. \tag{5.22}$$

With some simple manipulation, we see that (5.22) is equivalent to

$$0 < L < -\tau \ln \frac{K(\mu_+ - \mu_-)}{-\varepsilon_+ - K\mu_-}. \tag{5.23}$$

Under condition (5.23), for some time $t_2 > 0$, the trajectory in (5.21) satisfies $cx(t_1+L+t_2) = -\varepsilon_-$. After time $t_1 + L + t_2$, due to the time delay $L > 0$, the trajectory will satisfy

$$x(t_1+L+t_2+t) = e^{at}(x(t_1+L+t_2) + ba^{-1}\mu_+) - ba^{-1}\mu_+, \quad 0 \leq t \leq L$$

before the switch occurs at $t_1 + L + t_2 + L$. Again, we can check that the next switch will occur if and only if

$$cx(t_1 + L + t_2 + L) + cba^{-1}\mu_- < 0. \tag{5.24}$$

Also, with some simple manipulation, we see that (5.24) holds if and only if

$$0 < L < -\tau \ln \frac{K(\mu_- - \mu_+)}{-\varepsilon_- - K\mu_+}. \tag{5.25}$$

So, by combining (5.23) and (5.23), (ii) and (iii) are proved by noting that after time t_1, the trajectory $x(t)$ will be a limit cycle with two switchings per period. Moreover, any trajectory $x(t)$ starting from $K\mu_+ < cx(0) < K\mu_-$ will traverse $\mathcal{S}_-$ and $\mathcal{S}_+$ at fixed points $-\varepsilon_+/c$ and $-\varepsilon_-/c$, respectively.

Proposition 5.2.3. *Consider a RFS for the integral process in the form* $G(s) = \frac{\kappa}{s}e^{-Ls}$ *(i.e., (5.12) with* (5.14)*).*

(i) A unique solution exists for any given initial condition if and only if any of the following holds.

(a) $L > 0$,

(b) $L = 0$ *and* $0 > \max\{\kappa\mu_-, \quad \kappa\mu_+\}$,

(c) $L = 0$ *and* $0 < \min\{\kappa\mu_-, \quad \kappa\mu_+\}$,

(d) $L = 0$ *and* $\kappa\mu_- \geq 0 \geq \kappa\mu_+$.

(ii) A limit cycle exists if and only if $L > 0$ *and* $\kappa\mu_+ > 0 > \kappa\mu_-$. *If this is the case, the limit cycle is unique with two switchings per period.*

(iii) If a limit cycle exists, then the limit cycle is globally stable. Moreover, for the given process, the limit cycle is the common trajectory after the first switch, independent of the initial conditions.

Proof. Noting that in this case the trajectories of $x(t)$ and $y(t)$ will be governed by

$$x(t) = but + x_0,$$
$$y(t) = \kappa ut + cx_0,$$

the proof is similar to but simpler than those for the case $a \neq 0$, and thus is omitted here.

It can be checked that if there is a limit cycle for system (5.12), the relay switches at the instants when the trajectory $y(t) = cx(t)$ reaches the peak values; see Figure 5.4. Based on the information on this and other points, we can derive expressions for the limit cycle amplitudes and periods and use them to find the parameters in the FOPDT model of the process.

Parameter Estimation for Non-integral Processes Consider the case $a \neq 0$ (i.e., $\tau \neq \infty$). From (5.12) and Figure 5.4(a), we can see that

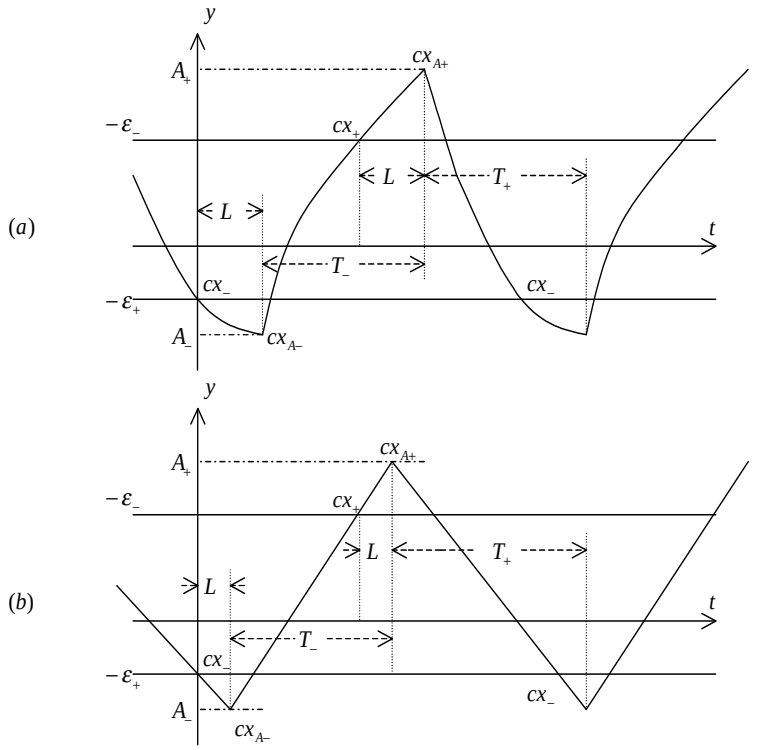


Fig. 5.4. Limit cycles for system (5.12) (a) $a \neq 0$; (b) $a = 0$

$$
\begin{aligned}
A_- &= cx_{A_-}, \\
x_{A_-} &= e^{aL}(x_- + a^{-1}b\mu_-) - a^{-1}b\mu_-, \\
cx_- &= -\varepsilon_+.
\end{aligned}
$$

Taking into account (5.13), the above yields

$$A_- = e^{-L/\tau}(-\varepsilon_+ - K\mu_-) + K\mu_-. \tag{5.26}$$

Similarly, from

$$
\begin{aligned}
A_+ &= cx_{A_+}, \\
x_{A_+} &= e^{aL}(x_+ + a^{-1}b\mu_+) - a^{-1}b\mu_+, \\
cx_+ &= -\varepsilon_-,
\end{aligned}
$$

we have

$$A_+ = e^{-L/\tau}(-\varepsilon_- - K\mu_+) + K\mu_+. \tag{5.27}$$

Let the time taken for the limit cycle to go from x_{A_-} (resp. x_{A_+}) to x_{A_+} (resp. x_{A_-}) be T_- (resp. T_+). Then, we get

$$
\begin{aligned}
x_{A_+} &= e^{aT_-}(x_{A_-} + a^{-1}b\mu_+) - a^{-1}b\mu_+, \\
cx_{A_+} &= A_+, \\
cx_{A_-} &= A_-,
\end{aligned}
$$

which gives, by taking into account (5.13),

$$
T_- = -\tau \ln \frac{e^{-L/\tau}(-\varepsilon_- - K\mu_+)}{e^{-L/\tau}(-\varepsilon_+ - K\mu_-) + K(\mu_- - \mu_+)}. \tag{5.28}
$$

Also, from

$$
\begin{aligned}
x_{A_-} &= e^{aT_+}(x_{A_+} + a^{-1}b\mu_-) - a^{-1}b\mu_-, \\
cx_{A_+} &= A_+, \\
cx_{A_-} &= A_-,
\end{aligned}
$$

we get

$$
T_+ = -\tau \ln \frac{e^{-L/\tau}(-\varepsilon_+ - K\mu_-)}{e^{-L/\tau}(-\varepsilon_- - K\mu_+) + K(\mu_+ - \mu_-)}. \tag{5.29}
$$

Luyben (1987) proposed the following method for first-, second- and third-order process modelling (called the ATV method): (i) The ultimate gain and ultimate frequency are obtained by using Åström's auto-tuning method. (ii) The dead time is read off from the initial response of the system to the auto-tuning test. (iii) The steady-state gain is obtained from a steady-state model of the process, or by using the step response method (Luyben, 1990). (iv) First-, second- and third-order transfer functions are fitted to the data at zero and the ultimate frequencies.

Table 5.1. Parameter estimation from biased relay

Case	Process			Biased relay				New method			ATV method		
	K	τ	L	T_+	T_-	A_+	A_-	K	τ	L	K	τ	L
1	1	2	2	2.79	3.91	0.859	-0.480	1.000	1.999	2.002	1	1.658	2
2	1	1	3	3.50	4.18	1.241	-0.670	1.000	0.999	3.006	1	1.042	3
3	1	5	2	3.44	5.46	0.497	-0.299	0.999	4.990	2.009	1	4.068	2
4	1	5	1	2.15	3.65	0.318	-0.209	1.001	5.003	1.004	1	4.055	1

In fact, the expressions (5.26)–(5.29) can be used to determine the three parameters in (5.11). However, they are coupled and nonlinear. Closed-form formulas for calculating the model parameters are not possible. Notice that for a biased relay, we can use (5.10) to find $G(0) = K$. Then, we obtain the normalized dead time, $\bar{L} = \frac{L}{\tau}$, from either (5.26) or (5.27), τ from (5.28) or (5.29), and finally obtain $L = \tau\bar{L}$. Simulation is carried out for processes

with different normalized dead time to illustrate the accuracy of the above identification method. The outputs of the biased relay are set at 1.3 and -0.7 respectively, and the hysteresis of relay is made symmetrical and set at 0.1. The resultant limit cycles and model parameters are presented in Table 5.1. For comparison, the parameters obtained by the ATV (Luyben, 1987) are also given in Table 5.1, where it is assumed that the steady-state gain is known and the dead time is read exactly.

In practice, many high-order processes can be well approximated by first-order plus dead time models. The proposed method can do so effectively. The results for some typical processes are listed in Table 5.2. The Nyquist curves of the real processes and the models are shown in Figure 5.5, and they are very close to each other over the phase range 0 to $-\pi$.

Table 5.2. FOPDT models for the high-order processes

Case	Process	Model
1	$\frac{1}{(2s+1)^2}e^{-2s}$	$\frac{1.00}{4.072s+1}e^{-2.93s}$
2	$\frac{1}{(2s+1)^5}e^{-2s}$	$\frac{1.00}{6.809s+1}e^{-7.26s}$
3	$\frac{1}{(s+1)(s^2+s+1)}e^{-0.5s}$	$\frac{1.00}{1.152s+1}e^{-2.1s}$
4	$\frac{-s+1}{(s+1)^5}e^{-s}$	$\frac{1.00}{2.99s+1}e^{-4.24s}$

Integral Processes We now turn to the case $a = 0$ (i.e., $\tau = \infty$), which implies integral processes. We compute as follows. It is obvious from Figure 5.4 that

$$A_- = cb\mu_- L - \varepsilon_+, \tag{5.30}$$

$$A_+ = cb\mu_+ L - \varepsilon_-. \tag{5.31}$$

From

$$A_+ = cb\mu_+ T_- + A_-,$$

$$A_- = cb\mu_- T_+ + A_+$$

and taking into account (5.14), we have

$$T_- = \frac{\kappa L(\mu_+ - \mu_-) - \varepsilon_- + \varepsilon_+}{\kappa\mu_+}, \tag{5.32}$$

$$T_+ = \frac{\kappa L(\mu_- - \mu_+) - \varepsilon_+ + \varepsilon_-}{\kappa\mu_-}. \tag{5.33}$$

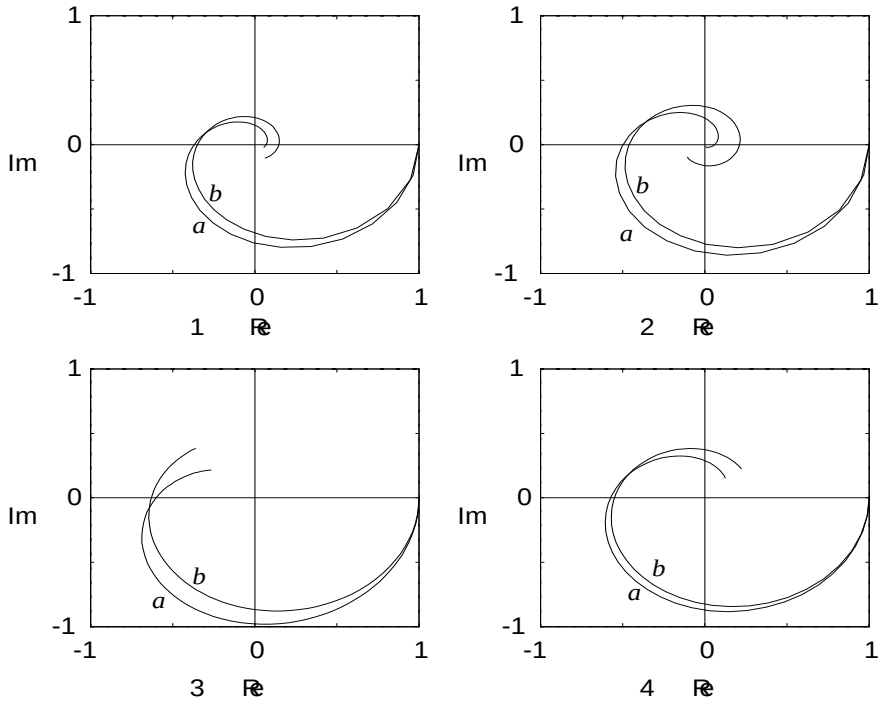


Fig. 5.5. Nyquist curves of processes and their FOPDT models
(a: real process; b: model)

5.3 Robustness Enhancement

It is well known that the difficulty in system identification is attributed to the existence of noise and disturbance. Noise is inevitable in practical situations and it contaminates the sampled data as picked up by the sensors. It is of so much a concern that once the samples are corrupted, there is basically no means by which they can be totally recovered. Distortions of the identification results are bound to arise when the samples used are subjected to random effects. Though it is true that recovery of the samples is not possible, many statistical methods, such as the stochastic least square algorithm employed in parameter estimation (Franklin *et al.*, 1994), have been devised to account for the effect of noise in the identification procedure. They essentially make use of the mean and variance characteristics of noise in their computations. In consideration of

the growing popularity of the use of relays in performing system identification, effective ways to minimize the effect of noise will be of much importance. This is especially so in situations where the amount of noise has become so large that it can no longer be ignored.

Before we can work on the relevant design, we need first to study the nature and characteristics of noise. In most applications, noise is modelled as white noise with zero mean. White noise is the most random type of signal possible, so that any samples taken at different instants are totally uncorrelated. If the white noise concerned has zero mean, then it is likely that the noise can apparently be rejected by use of averaging, although it cannot be removed from the associated signals, and this is the main idea behind the method introduced in this section. To test the validity of the concept, the accuracy of the process gains at the critical point and the static frequency are evaluated by taking different numbers of limit cycles in the relay experiment. It is found that as samples are averaged over a larger number of limit cycles, the relative error in the process gains at the two frequencies drops, and the results conform to expectations. Apart from noise, disturbances are another common source of error in many identification problems. They can appear in many different forms depending on their sources. Some of the types of disturbances are load disturbances, measurement errors and parameter variations (Åström and Writtenmark, 1984). In this section, disturbances due to offsets in measurement and load disturbances, which are typically modelled as steps acting in the loop, are considered.

Suppose that a SISO linear plant is subjected to a relay test as shown in Figure 5.6, where white noise $n(t)$ of zero mean acts in the loop through the sensor at the output of the plant, and $w(t)$ is a constant disturbance. Consider first the disturbance-free case, i.e., $w(t) = 0$. With noise present in the system, the contaminated plant output $y(t)$ is measured instead of the actual value $\tilde{y}(t)$. If $y(t)$ and $u(t)$ are employed directly for identification, impaired results will be obtained. Since noise samples do not follow a traceable pattern and can assume any random value at different instants of time when they are picked up at different locations in the loop, they cannot be calculated or predicted from previous records and thus, the actual output is not recoverable from the corrupted samples.

Despite the fact that signal corruption by noise is an irreversible process, the behaviour of noise can still be described by statistical measurements. Therefore, estimation of the actual signal $\tilde{y}(t)$ from the infected $y(t)$ to achieve more accurate identification results is always possible by examining the statistical properties of noise. We start by noting that the noise involved has zero mean.

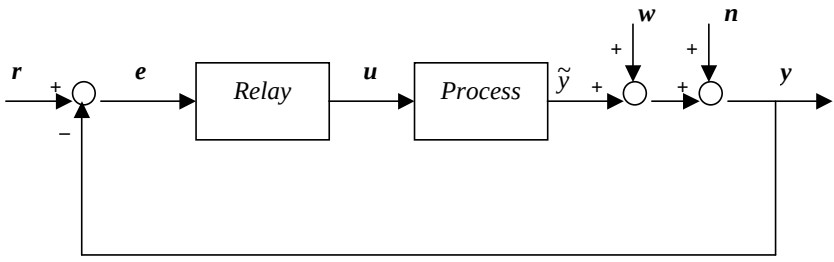


Fig. 5.6. Relay feedback system with noise and disturbance

This suggests that if a large number of samples extending over a sufficient number of periods (limit cycles) are collected and averaged out, it is possible to obtain a period of processed samples in which noise is significantly reduced. This is because noise attached to the different periods cancel each other and in the limit when the number of periods taken approaches infinity, the noise can be completely cancelled in the ideal situation. However, this is practically unattainable and it is usually sufficient to take up to a certain number of periods depending on the noise-to-signal ratio.

Let the relay feedback system give rise to limit cycle oscillations and the oscillation frequency be ω_c. Let $\hat{y}(t)$ be an estimate of one period of $\tilde{y}(t)$ from $y(t)$ using the averaging method. The associated frequency response $G(j\omega_c)$ can be obtained from $\hat{y}(t)$ and $u(t)$ using the equation

$$\hat{G}(j\omega_c) = \frac{\hat{Y}(j\omega_c)}{U(j\omega_c)} = \frac{\int_{1period} \hat{y}(t)e^{-j\omega_c t}dt}{\int_{1period} u(t)e^{-j\omega_c t}dt}. \tag{5.34}$$

Since the static gain of the process can be attained at the same time in the same relay test with no extra effort if biased relays are used in place of symmetrical relays, the second frequency point is conveniently chosen to correspond to the d.c. component of the process. The relevant frequency response $G(0)$ is obtained from the equation

$$\hat{G}(0) = \frac{\hat{Y}(0)}{U(0)} = \frac{\int_{1period} \hat{y}(t)dt}{\int_{1period} u(t)dt}. \tag{5.35}$$

It is, in essence, the ratio of the area of $\hat{y}(t)$ over one period to that of $u(t)$.

Two observations can be made in performing the relay experiment described above. First, if the noise power is high, the relay will be easily switched by noise on top of the switchings by the actual process output $\tilde{y}(t)$. To overcome this problem, a higher hysteresis level can be used for situations with higher noise

power. A convenient indicator of the noise power P_n is its standard deviation σ_n and the two are related by $P_n = \sigma_n^2$. It is found from simulation studies that correct switchings can be achieved if the hysteresis level is set at twice the standard deviation of the noise together with an upper limit equal to 0.95 times the minimum of the on and off relay output level.

The second observation concerns the reliability of judging the period of the limit cycles from the separation between two switching points of the relay output at high noise power. To derive a more accurate value for the period, samples taken during the transient are excluded from the entire span of $u(t)$. The total number of switching points N is then counted. If N is even, the last point is retrenched so as to keep N odd. This is to make sure that a window size of an integral number of periods is used since an odd value of N implies an even number of separations and each separation denotes a half-period. The total time covering these switching points, T, is then recorded and the quotient $2T/N$ is taken as the final answer for the period of the limit cycle. Since this method uses the average of the periods of all limit cycles, it is intuitively more credible in the presence of noise and is justified to be so in the simulation results to be presented later.

The noise power and signal power are defined as

$$P_n = \frac{\int_{T_i}^{T_f} n^2(t)dt}{T_f},$$

$$P_y = \frac{\int_{T_i}^{T_f} y^2(t)dt}{T_f - T_i},$$

respectively, where T_i and T_f are the corresponding initial and final time instants between which the samples $n(t)$ and $y(t)$ are taken for integration in the respective equations. The noise-to-signal ratio (NSR) may be measured by

$$NSR = \frac{P_n}{P_y}.$$

The relative estimation errors at the two frequency points are defined as

$$\text{for the point } s = j\omega_c, \quad \text{relative error} = \left| \frac{\hat{G}(j\omega_c) - G(j\omega_c)}{G(j\omega_c)} \right|,$$

$$\text{for the point } s = 0, \quad \text{relative error} = \left| \frac{\hat{G}(0) - G(0)}{G(0)} \right|.$$

The mean relative error (MRE) represents the average value of the errors in $G(0)$ and $G(j\omega_c)$. By varying the noise power and hence the NSR, the number of

limit cycles used to produce a pre-specified mean relative error is calculated. In such a way, the relationship between NSR and MRE is established. In practice, this can be used to decide how many limit cycle periods are required to reach the pre-specified estimation accuracy in terms of MRE, i.e. how long the testing should last.

Example 5.3.1. The proposed method is applied to a first-order plus delay process

$$G(s) = \frac{1}{s+1}e^{-3s}.$$

The noise power is gradually increased to raise the noise-to-signal ratio. The results of the computations are shown in Table 5.3. It can be concluded from the results that

- For the same NSR, as the number of limit cycles used increases, the relative errors decrease.
- As the NSR increases, the same relative errors can be achieved if more limit cycles are used.

It must be emphasized that the identification method used here serves only as a tool to show and verify the effectiveness of the proposed averaging method. The choice of method is entirely optional but the results found on the relationship between the number of limit cycles adopted, the relative error in the frequency response and the noise-to-signal ratio are applicable in general to all relay-based identification methods.

Consider next the situation where both noise and disturbance are present as shown in Figure 5.6. Apart from the white noise model $n(t)$ encountered earlier, also appearing in the sensor is a constant value disturbance represented by $w(t) = w$. Owing to the extra d.c. term introduced into the output signal by the disturbance and the fact that its value is uncertain and may change with time, the static component of the output due solely to the relay bias cannot be separated from $y(t)$ and hence the method described above for the estimation of $G(0)$ will fail. Nevertheless, the calculation of $G(j\omega_c)$ for the process is not affected since the disturbance contains no frequency content other than d.c. and therefore (5.34) remains applicable despite the unknown w. This can be verified mathematically by noting that

$$\frac{\int_{1period} y(t)e^{-j\omega_c t}dt}{\int_{1period} u(t)e^{-j\omega_c t}dt} = \frac{\int_{1period} y(t)e^{-j\omega_c t}dt - \int_{1period} we^{-j\omega_c t}dt}{\int_{1period} u(t)e^{-j\omega_c t}dt}$$

Table 5.3. Required limit cycles vs NSR and MRE without disturbance

		Mean Relative Error				
		10%	8%	6%	4%	2%
NSR	0.0744	3	3	3	3	11
	0.1382	3	3	3	3	13
	0.1938	3	3	3	11	11
	0.2510	3	3	6	10	22
	0.3025	4	4	4	6	16
	0.3502	5	5	12	80	80
	0.3968	10	22	22	74	80
	0.4520	33	33	39	80	80
		Minimum number of limit cycles required				

$$= \frac{\int_{1period}(y(t)-w)e^{-j\omega_c t}dt}{\int_{1period}u(t)e^{-j\omega_c t}dt} = G(j\omega_c). \tag{5.36}$$

In the above equation, $y(t)$ can be replaced by its average over a number of periods, in much the same way as in the averaging technique illustrated earlier. Therefore, although a measurement of the magnitude of the disturbance is not available, $G(j\omega_c)$ can still be deduced by using the output samples $y(t)$.

Example 5.3.2. Simulations are performed using the same model with transfer function $G(s) = \frac{1}{s+1}e^{-3s}$ as in the previous example and details of the adjustment of the hysteresis level, the determination of the period of limit cycles, and the calculation of noise power and signal power follow precisely those stipulated there. The final results are given in Table 5.4. The same conclusions as in the previous example are obtained.

5.4 Parasitic Relay

To get more information on a process, identifying multiple points on the process frequency response from one relay test is more appealing than the use of several relay tests. A standard or symmetric relay can excite the process at the limit cycle oscillation frequency ω_c, as well as $3\omega_c$, $5\omega_c$, ..., where ω_c is usually very close to the process critical frequency for which the process phase lag is π. Due

Table 5.4. Required limit cycles vs NSR and MRE with disturbance

		Mean Relative Error				
		10%	8%	6%	4%	2%
NSR	0.0819	3	3	3	3	3
	0.1499	3	3	3	3	4
	0.2144	3	3	3	3	7
	0.2665	3	3	3	3	9
	0.3165	3	3	6	9	16
	0.3649	8	8	8	18	18
	0.4076	18	21	27	27	42
	0.4489	28	28	28	28	28
		Minimum number of limit circle required				

to the low-pass nature of most practical processes, the signal-to-noise ratios at $3\omega_c$, $5\omega_c$, ..., are too low to enable meaningful estimation of the process frequency response at these points. Effectively, we can only get the critical point information from such a relay test. By adding a bias to the relay, we may obtain the process static gain as well. The frequency response information between zero and ω_c is most important for an understanding of the process dynamics and its use in controller design. To estimate more points around this region in one relay test, a modified relay is proposed in this section.

Our modified relay consists of a standard relay and a parasitic relay as shown in Figure 5.7. The standard relay operates as usual with amplitude of the sampled output $u_1(k)$ being μ_1, where $u_1(k)$ is the kth sample of $u_1(t)$. It is well known that this relay excites process mainly at frequency ω_c. In order to provide additionally effective excitation to the process at frequencies other than ω_c while maintaining the process output oscillation under such an arrangement, a parasitic relay with output amplitude $\alpha\mu_1$ and twice the period of $u_1(k)$ is introduced and superimposed on $u_1(k)$. This implies that the output $u_2(k)$ of the parasitic relay flip-flops immediately when every period of oscillations in $u_1(k)$ is reached. The parasitic relay is realized by

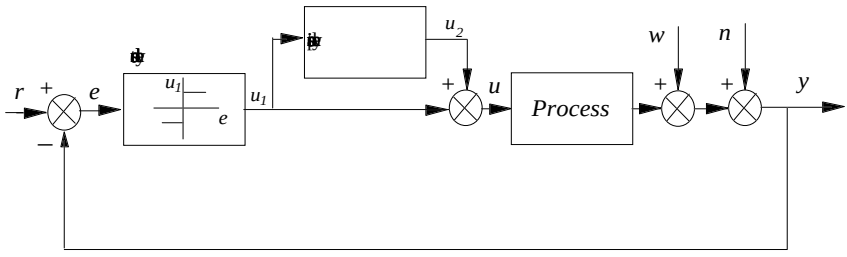


Fig. 5.7. Modified relay feedback system

$$\begin{cases} u_2(0) = \alpha\mu_1; \\ u_2(k) = -\alpha\mu_1 \cdot sign(u_2(k-1)), \text{ if } u_1(k-1) > 0 \text{ and } u_1(k) < 0; \\ u_2(k) = u_2(k-1), \text{ otherwise.} \end{cases} \tag{5.37}$$

The constant α should be large enough to sufficiently stimulate the process while it should also be small enough that the parasitic relay will not change the period of oscillation generated by the main relay by too much. According to extensive simulations, α is recommended to be $0.1 \sim 0.3$. The output of the modified relay test is thus given by

$$u(k) = u_1(k) + u_2(k),$$

and is sent to the process. In this way, the process is stimulated by two different excitations whose periods are T_c and $2T_c$. The resultant process output y from the modified relay test is shown in Figure 5.8 and reaches a stationary oscillation of period $2T_c$. Due to the two excitations in u, y consists of frequency components at $\frac{2\pi}{T_c}$, $\frac{\pi}{T_c}$ and their odd harmonics at $\frac{6\pi}{T_c}$, $\frac{10\pi}{T_c}$, ..., and $\frac{3\pi}{T_c}$, $\frac{5\pi}{T_c}$, ..., respectively. Let y_s and u_s be a period $(2T_c)$ of the stationary oscillations of $u(k)$ and $y(k)$ respectively. For a linear process, the process frequency response can be obtained by

$$G(j\omega_i) = \frac{\int_0^{2T_c} y_s(t)e^{-j\omega_i t}dt}{\int_0^{2T_c} u_s(t)e^{-j\omega_i t}dt}, \quad i = 1, 2, \ldots, \tag{5.38}$$

where

$$\omega_i = \frac{(2i-1)2\pi}{2^l T_c}, \quad l = 0, 1,$$

are the basic and odd harmonic frequencies in u_s and y_s. Equation (5.38) can be implemented using the FFT algorithm as

$$G(j\omega_i) = \frac{FFT(y_s)}{FFT(u_s)}. \tag{5.39}$$

Since the method adopts spectrum analysis instead of the describing function, it will lead to accurate process frequency response estimation. The proposed method employs the FFT only and the required computation burden is light. It can identify multiple points on the frequency response from a single relay test. Moreover, the method can easily be extended to find other points on the frequency response. You may flip-flop the parasitic relay every three or four periods of the main oscillations generated by the standard relay to get other frequency points. You may also use more than one parasitic relay in a relay test and find more points on the frequency response in one relay test. As discussed earlier, to estimate the static gain of a process, a bias has to be introduced to the relay input or output.

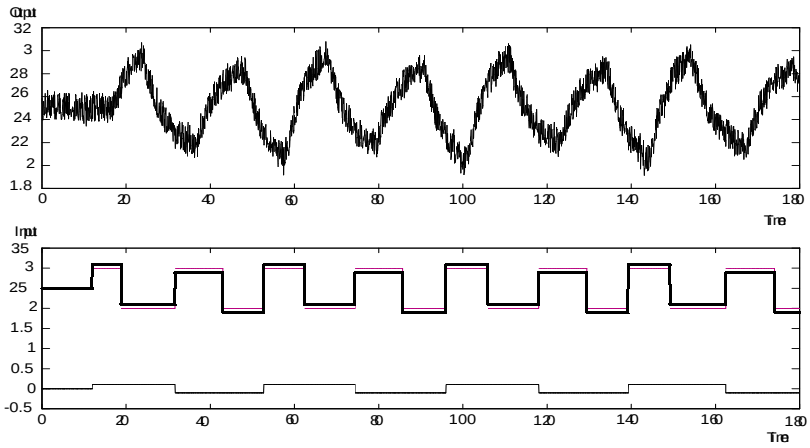


Fig. 5.8. Process input and output in the modified relay test
(—— u, ... u_1, —— u_2)

Reduction of Noise and Disturbance Effects on Estimation In a realistic environment, the major concerns for any identification method are distur-

bance and noise. As in Section 5.3, it can be shown that our new identification is also unaffected by a step-like load disturbance w as shown in Figure 5.7. As to measurement noise in the relay test, the same anti-noise measures as presented in Section 5.1 for simple relays, such as the introduction of hysteresis and a low-pass filter, can be used for the master relay in the new scheme. To reduce the noise effect further, especially in the case of large noise-to-signal ratio, we can average several periods of the stationary oscillations to enhance estimation robustness; see Section 5.3 for details. With these anti-noise measures, the proposed method can reject noise very effectively, and provide accurate frequency response estimation at frequencies $0.5\omega_c$, ω_c and $1.5\omega_c$. It should also be noted that a nonzero initial condition of the process at the start of a relay test has no effect on our estimation because only stationary oscillations u_s and y_s after the transient are used in the estimation, where u_s and y_s are independent of the initial condition.

For assessment of identification accuracy, the identification error is measured here by the worst-case error

$$ERR = \max_i \left\{ \left| \frac{\hat{G}(j\omega_i) - G(j\omega_i)}{G(j\omega_i)} \right| \times 100\%,\ i = 1, 2, 3 \right\}, \tag{5.40}$$

where $G(j\omega_i)$ and $\hat{G}(j\omega_i)$ are the actual and estimated process frequency responses respectively. The process frequency responses at $\frac{\pi}{T_c}$, $\frac{2\pi}{T_c}$ and $\frac{3\pi}{T_c}$ are considered since the frequency response in these region is especially important to controller design. To test estimation robustness against noise, the process output may be corrupted by some noise and the corrupted output used for identification. The noise level is judged, in the context of system identification, by the noise-to-signal ratio, which is usually defined as

$$\begin{aligned} N_1 &= \textit{Noise-to-Signal Power Spectrum Ratio} \\ &= \frac{\textit{mean power spectrum density of noise}}{\textit{mean power spectrum density of signal}}, \end{aligned} \tag{5.41}$$

or

$$\begin{aligned} N_2 &= \textit{Noise–Signal Mean Ratio} \\ &= \frac{mean(abs(noise))}{mean(abs(signal))}. \end{aligned} \tag{5.42}$$

In order to test our method in a realistic environment, real-time relay tests were performed using the *Dual Process Simulator KI 100* from KentRidge Instruments, Singapore. The simulator is an analogue process simulator and can be configured to simulate a wide range of industrial processes with different levels of noise and disturbance. The simulator is connected to a PC via an A/D and

D/A board. The window-based *DT VEE 3.0* from *DataTranslation* is used as the system control platform, on which the relay control code is written in C++. The fastest sampling time of the *VEE* system is 0.06 second. A few examples of real-time testing are presented below.

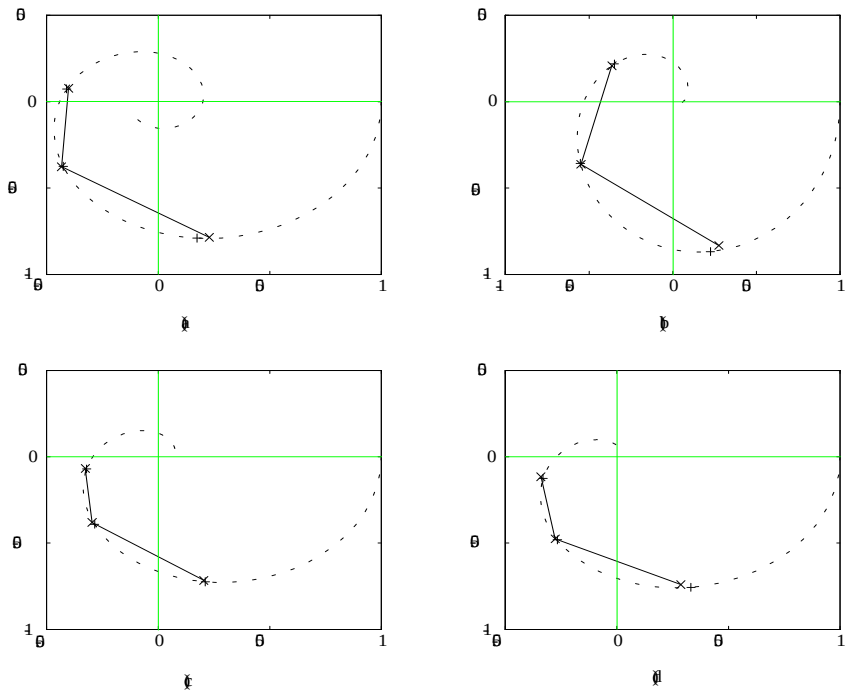


Fig. 5.9. Estimation of frequency response with $N_1 = 10\%$ noise (+ actual, x estimated)

Example 5.4.1. Consider a first-order plus dead time process:

$$G(s) = \frac{1}{5s+1}e^{-5s}.$$

In our relay test, the standard relay amplitude is chosen as 0.5 and the parasitic relay height is set to $20\% \times 0.5$. Without additional noise, the noise-to-signal ratio N_1 of the inherent noise in our test environment is 0.025% ($N_2 = 4\%$). The identification error ERR is 2.57%. To see noise effects, extra noise is introduced

with the noise source in the Simulator. Time sequences of $y(t)$ and $u(t)$ in a relay test under $N_1 = 10\%$ ($N_2 = 31\%$) are shown in Figure 5.8, where $t = 0 \sim 12$ is the "listening period", in which the noise bands of $y(t)$ and $u(t)$ at steady state are measured. At this noise level, the hysteresis is chosen as 0.3. With averaging of four periods of stationary oscillations, the estimated frequency response points at this noise level are shown in Figure 5.9(a). The result is pretty good. At noise level $N_1 = 10\%(N_2 = 31\%)$, real-time testing of the proposed method was also performed on other typical processes whose transfer functions are listed in Table 5.5. Figure 5.9 compares frequency responses of the actual processes and their respective estimates. The identification results are shown to be satisfactory.

To ensure estimation accuracy at different noise levels, the number of stationary oscillation periods adopted in average calculation should be different. The estimation error ERR vs the number of stationary oscillation periods adopted in the averaging is plotted in Figure 5.10, which can be used as a guide in deciding how many periods are needed to achieve a given estimation accuracy at a given noise level. Table 5.5 shows the identification accuracy of four real-time examples at different noise and disturbance levels.

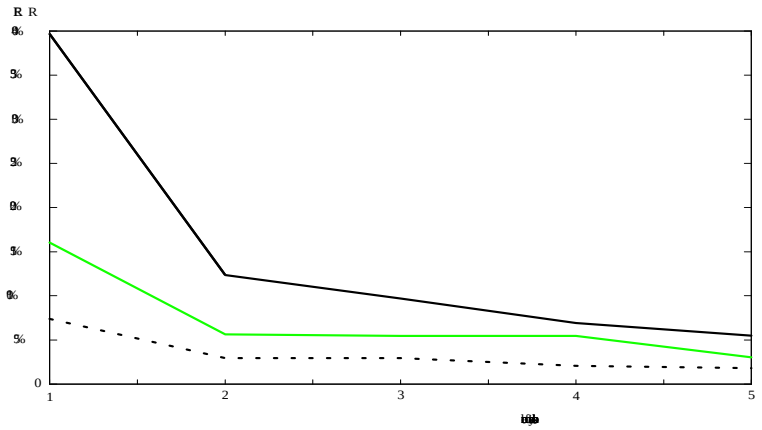


Fig. 5.10. ERR vs number of stationary oscillations adopted
(- - - $N_1 = 0\%$, ... $N_1 = 1\%$, —— $N_1 = 10\%$)

In this section, a new method for process frequency response identification has been developed in the context of the relay feedback test. The method has

several unique features. Firstly, it can estimate multiple points on the process frequency response simultaneously with one single relay experiment and this reduces testing time significantly. Secondly, the method is accurate since no approximation is made. The computations involved are simple so that it can be easily implemented on microprocessors. Thirdly, the method is insensitive to noise and step-like load disturbances, and nonzero initial condition. Various processes have been employed to demonstrate the effectiveness of the method in real time. The identified process frequency response is useful for process analysis and controller design.

Table 5.5. Identification error (ERR)

Disturbance	d=0			d=0.5
Noise	N_1=0%	1%	10%	10%
	N_2=4%	11%	31%	31%
Processes	ERR			
(a) $\frac{1}{5s+1}e^{-5s}$	2.57%	5.02%	6.83%	7.17%
(b) $\frac{1}{(s+1)^8}$	2.93%	5.46%	6.90%	6.35%
(c) $\frac{1}{(s+1)(5s+1)}e^{-2.5s}$	5.01%	5.08%	5.41%	5.16%
(d) $\frac{1-s}{(2s+1)^2(5s+1)}e^{-0.5s}$	3.55%	5.17%	6.38%	5.13%

5.5 Cascade Relay

In the preceding section, one notes that the amplitude of the parasite relay cannot be chosen freely. It should be large enough to sufficiently stimulate the process while it should also be small enough that the parasite relay will not change the period of oscillation generated by the main relay by too much. Since the recommended value for it is small, the resultant estimation at $0.5\omega_c$ might be sensitive to measurement noise due to small signal-to-noise ratio there. In this section, cascade relay feedback is proposed as an alternative to parasitic feedback. The former can achieve almost the same objectives and results as the latter while the generation of limit cycles is less restrictive in the former than the latter.

The proposed cascade relay feedback consists of a master relay in the outer loop and a slave relay in the inner loop, as shown in Figure 5.11. The slave relay

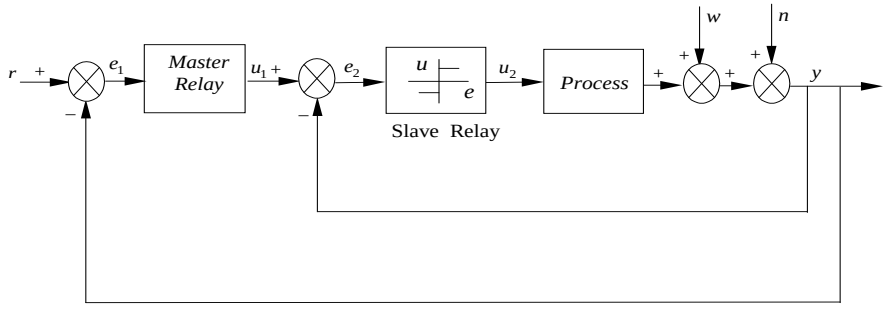


Fig. 5.11. Cascade relay feedback system

is just a standard relay with amplitudes of the sampled output $u_2(k)$ being d_2. With the inner loop closed, this relay can excite the process at the frequency ω_c sufficiently. In order to provide additional effective excitations to the process at other frequencies while maintaining the process output oscillation, the master relay in the outer loop is introduced with its output amplitude of d_1 and bias of μ_1, and operated at the frequency $0.5\omega_c$. It is realized by

$$u_1(k) = \begin{cases} -u_1(k-1) + 2\mu_1, & \text{if } e_1(k-1) < 0 \text{ and } e_1(k) > 0; \\ u_1(k-1) - 2d_1, & \text{if } e_1(k-1) > 0 \text{ and } e_1(k) < 0; \\ u_1(k-1), & \text{otherwise.} \end{cases} \tag{5.43}$$

The sampled output $u_1(k)$ from the master relay is a periodic stair wave with three amplitudes, $2d_1 + \mu_1$, μ_1 and $-2d_1 + \mu_1$, respectively. This relay is introduced to obtain persistent excitation at frequencies of $0.5\omega_c$ and $1.5\omega_c$, in addition to ω_c. In this way, the process is stimulated by two different excitations whose periods are T_c and $2T_c$. The waveforms for u_1, u_2, and the resultant output response are shown in Figure 5.12. The output reaches a stationary oscillation with period $2T_c$. The bias μ_1 is introduced to reduce possible unnecessary switchings due to noise and disturbances. One can see that the difference between the master relay output and the process output determines the switchings in the slave relay. Since the output of the master relay has three possible values, $2d_1 + \mu_1$, μ_1 and $-2d_1 + \mu_1$, load disturbance noise will not cause any relay switching unless its amplitude is larger than μ_1. Hence, a suitable μ_1 helps to establish robust oscillations in the process output at two fundamental frequencies.

Due to the two excitations in the input, y consists of frequency components at $\frac{2\pi}{T_c}$ and $\frac{\pi}{T_c}$ and their odd harmonics at $\frac{6\pi}{T_c}$, $\frac{10\pi}{T_c}$, ..., and $\frac{3\pi}{T_c}$, $\frac{5\pi}{T_c}$, ..., respectively. This enables process frequency response estimation at these points.

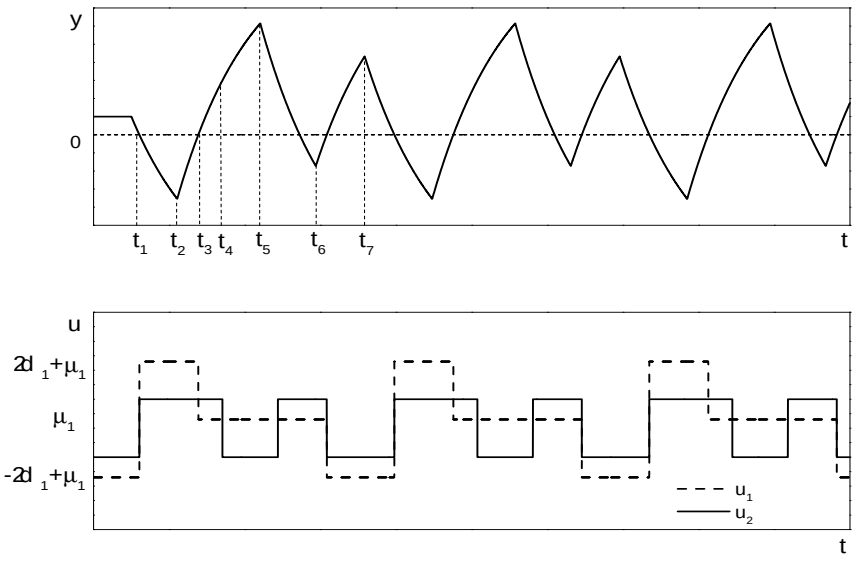


Fig. 5.12. Signals from cascade relay test

The estimation formula and its implementation are given by (5.38) and (5.39), respectively. Moreover, the cascade relay shares the same properties of estimation robustness against noise and constant disturbances as in the master and parasite relay case, and can use some anti-noise measures as discussed in Section 5.4.

It should be pointed out that in principle, the proposed method can be extended to find other points on the frequency response. One can realize the master relay and generate u_1 as a periodic stair wave which can enable y to have frequency components other than ω_c, $\omega_c/2$, and their odd harmonics. Another possible way is to use more than one cascade outer loop in a relay test and find more points on the frequency response in one relay test. Practically, the information about three points on the process frequency response available from the proposed cascade relay method is usually adequate to represent the process dynamics and to tune a good controller. Although more points can be identified from the extension as mentioned above, the structure will inevitably become more complicated, leading to implementation problems.

Guidelines for Relay Parameter Selection Most processes in industry are open-loop stable, and it is conjectured (Åström and Writtenmark, 1984) that most of them will exhibit a stable limit cycle with standard relay feedback. This is also true when the proposed cascade relay feedback is used. Extensive

simulation shows that stationary oscillation is obtained for most processes if the parameters are chosen properly. We are thus motivated to consider the parameter selection problem for the generation of stationary oscillation in the cascade relay test.

To simplify the problem and gain insight into the solution, let us consider, as in Section 5.2, a first-order plus dead time (FOPDT) model,

$$G(s) = \frac{K}{\tau s + 1} e^{-Ls}. \tag{5.44}$$

Lemma 5.5.1. *For the cascade relay feedback system of Figure 5.11 with the process given by* (5.44)*, if stationary oscillations at two different fundamental frequencies exist, then* $d_2 K > 0$ *and*

$$\mu_1 < \min \left\{ d_2 K,\ d_2 K (e^{\frac{L}{\tau}} - 1) \right\}. \tag{5.45}$$

Proof. Refer to Figure 5.12 for the definition of t_i. Consider for illustration that the output initial condition is $y_0 > 0$ and that the slave relay switches to $u_2 = -d_2$ at $t = 0$. Obviously, the other case can be proven similarly. y will decrease while e_1 increases monotonically after a delay L. The output response for $t \geq L$ is described by

$$y(t) = -d_2 K (1 - e^{-(t-L)/\tau}) + y_0 e^{(t-L)/\tau}.$$

At $t = t_1$, y becomes 0, which causes u_1 to switch to $2d_1 + \mu_1$ and u_2 to d_2. Before $u_2 = d_2$ takes its effect on y, the output after t_1 can be described by

$$y(t) = -d_2 K (1 - e^{-(t-t_1)/\tau}), \tag{5.46}$$

where t_1 can be calculated from $y(t_1) = 0$, i.e.,

$$-d_2 K (1 - e^{-(t_1 - L)/\tau}) + y_0 e^{(t_1 - L)/\tau} = 0,$$

as

$$t_1 = L + \tau \ln(1 + \frac{y_0}{d_2 K}).$$

However, $y(t)$ will not respond to the positive switching $u_2 = d_2$ until it has continued its monotonic downward trend for a time L, as seen from (5.46). At $t_2 = t_1 + L$, $y(t)$ reaches its first peak in the cycle, and its value can be calculated from (5.46) as

$$y(t_2) = y(t_1 + L) = -d_2 K (1 - e^{-L/\tau}). \tag{5.47}$$

For $t > t_2$, the output response can be described by

$$y(t) = d_2K(1-e^{-(t-t_2)/\tau})+y(t_2)e^{-(t-t_2)/\tau}. \tag{5.48}$$

The output increases monotonically. At time $t = t_3$, y becomes 0 and u_1 switches to μ_1. For $t_3 < t < t_4$, y can still be described by (5.48) and it keeps increasing with decreasing e_2. At $t = t_4$, e_2 becomes 0, which leads to the switching of u_2 to $-d_2$. t_4 can be calculated from the following equation:

$$y(t_4) = \mu_1,$$

or

$$d_2K(1-e^{-(t_4-t_2)/\tau})+y(t_2)e^{-(t_4-t_2)/\tau} = \mu_1. \tag{5.49}$$

However, $y(t)$ will not respond to the negative switching of u_2 until it has continued for a time L. Therefore, at $t = t_5 = t_4 + L$, y reaches its second peak point as

$$\begin{aligned} y(t_5) &= y(t_4 + L) \\ &= d_2K(1 - e^{-(t_4+L-t_2)/\tau}) + y(t_2)e^{-(t_4+L-t_2)/\tau} \\ &= d_2K - (d_2K - \mu_1)e^{-L/\tau}. \end{aligned} \tag{5.50}$$

The other two peaks in one cycle can be found similarly. Denote these four points as

$$\begin{aligned} y(t_2) &= -d_2K(1 - e^{-L/\tau}), \\ y(t_5) &= d_2K - (d_2K - \mu_1)e^{-L/\tau}, \\ y(t_6) &= -d_2K + (d_2K + \mu_1)e^{-L/\tau}, \\ y(t_7) &= d_2K(1 - e^{-L/\tau}). \end{aligned} \tag{5.51}$$

To enable the slave relay in the inner loop to switch between two output levels while the master relay in the outer loop switches between the three output levels with two different fundamental frequencies indefinitely, the following condition holds:

$$y(t_6) < 0,$$

or

$$\mu_1 < d_2K(e^{L/\tau} - 1).$$

It also follows from (5.49) that

$$e^{-(t_4-t_1)/\tau} = \frac{d_2K - \mu_1}{d_2K(2 - e^{-L/\tau})} > 0,$$

i.e.,

$$\mu_1 < d_2K.$$

This completes the proof of the lemma.

Several remarks are now made regarding this lemma.

- The condition therein provides a guideline for choosing the bias μ_1 to obtain limit cycles. Other parameters can be chosen similarly to the standard relay case. In practice, the relay amplitude is adjusted so that the oscillation at the process output is about three times the amplitude of the noise.
- Though Lemma 5.5.1 is derived for a hysteresis-free relay, it can easily be extended to relays with hysteresis.
- For a process with negative steady-state gain, u_1 and u_2 have to change their signs to attain stationary oscillations.
- The assumption $y_0 > 0$ is made only for simplicity of illustration, and can be removed without affecting the derivation. In fact, as can be seen from (5.51), y_0 has no influence on the four peaks.

Simulation A few simulation examples are presented below, and a comparison is made with the standard relay in Section 5.1 and parasite relay in Section 5.4. The relay parameters in these three cases are chosen such that the resultant output oscillations have almost the same amplitudes. Performance is measured by the worst-case error, ERR, as defined in (5.40), and the noise-to-signal ratio in the form of (5.41) and (5.42) is also adopted here.

Example 5.5.1. (**Simple Dynamics**) Consider a FOPDT process:

$$G(s) = \frac{1}{5s+1}e^{-5s}.$$

In our relay test, the slave relay amplitude d_1 is chosen as 1, the master relay height d_2 is set to 1, and its bias μ_1 is 0.5. The responses are shown in Figure 5.13, where u is the input to the process. For multiple-point estimation evaluation, the parasite relay test sets its standard relay amplitude to 0.5 and the parasitic relay height to 0.2×0.5. For the standard relay test, its height is set to 1 and only one point ω_c on the process frequency response is available and then used for calculating its ERR. In the noise-free case, the identification error ERR is 0.30% for the cascade relay test, 0.31% for the parasite relay

test and 11.19% for the standard relay test, respectively. Afterwards, noise is introduced using the band-limited white noise module in Matlab. Under this noise condition, hysteresis is set to 0.3 for all three relays. To reduce the noise effect, especially in the case of large noise-to-signal ratio, we use the average of the last two–four periods of oscillation as the stationary oscillation period, depending on the noise level. Further, the accuracy of the relay test depends on the reliability of judging the period of the limit cycles. To derive a more accurate value of the period, the averaging technique in Section 5.3 is adopted. With these noise rejection techniques, for $N_1 = 10\%$, ERR is 1.87% for the cascade relay test, 6.83% for the parasite relay and 10.01% for the relay test, respectively. By averaging four periods of stationary oscillations, the estimated frequency responses for noise levels of $N_1 = 10\%$ and 20%, are shown in (a) of Figure 5.14 and Figure 5.15, respectively.

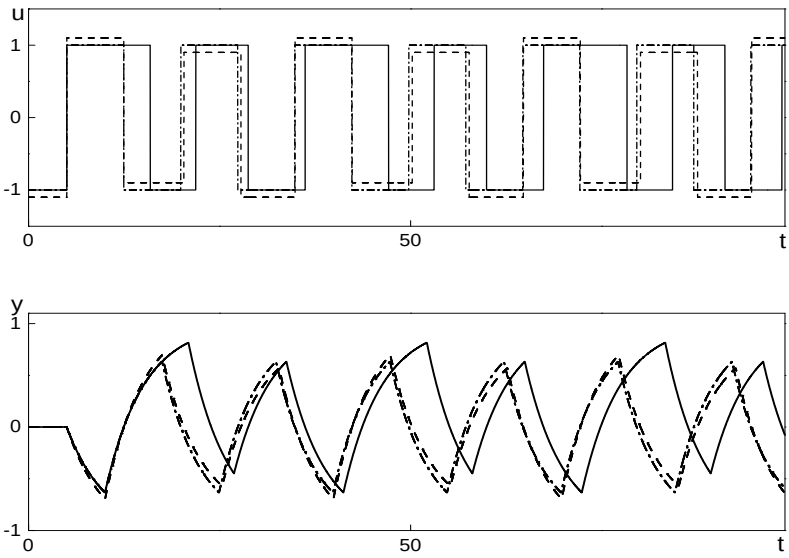
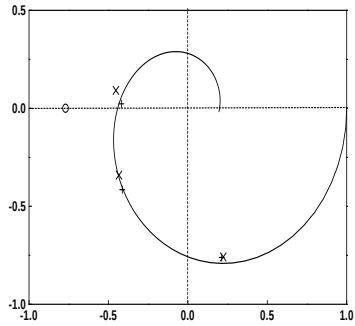
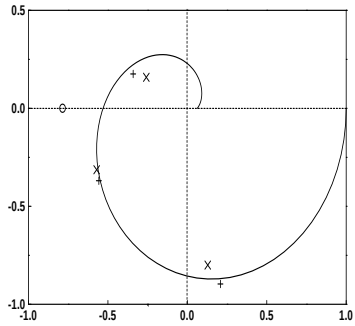


Fig. 5.13. Responses obtained during various relay tests
(- · - ·-: standard; - - - : parasite; —— : cascade)

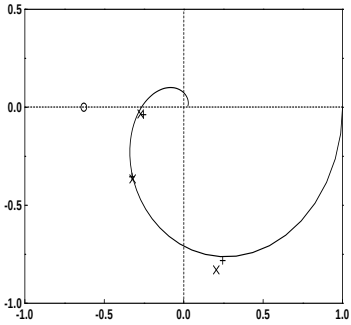
Example 5.5.2. (**Complex Dynamics**) Consider now three processes having different dynamics:



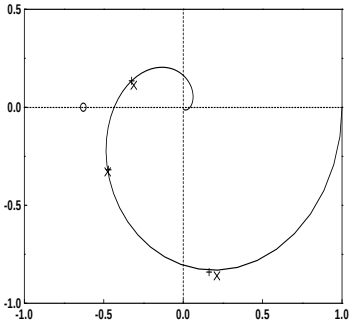
(a) $\frac{1}{5s+1}e^{-5s}$



(b) $\frac{1}{(s+1)^8}$



(c) $\frac{1}{(s+1)(5s+1)}e^{-2.5s}$



(d) $\frac{1-s}{(2s+1)^2(5s+1)}e^{-0.5s}$

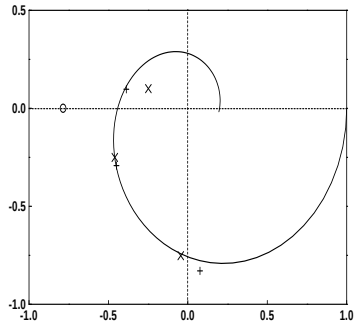
Fig. 5.14. Evaluation of $G(j\omega)$ for $N_1 = 10\%$
('+': cascade, 'x': parasite, 'o': standard)

$$G(s) = \frac{1}{(s+1)^8}; \tag{5.52}$$

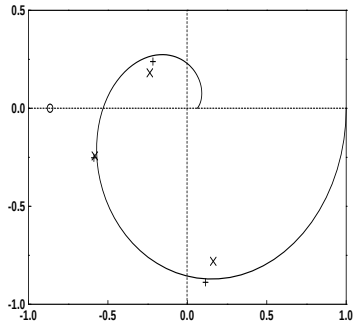
with a multi-lag high order,

$$G(s) = \frac{1}{(s+1)(5s+1)}e^{-2.5s};$$

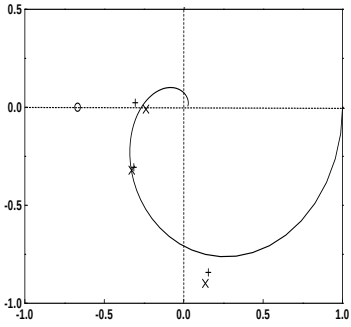
with different poles, and



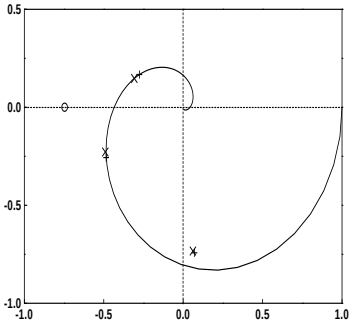
(a) $\frac{1}{5s+1}e^{-5s}$



(b) $\frac{1}{(s+1)^8}$



(c) $\frac{1}{(s+1)(5s+1)}e^{-2.5s}$



(d) $\frac{1-s}{(2s+1)^2(5s+1)}e^{-0.5s}$

Fig. 5.15. Evaluation of $G(j\omega)$ for $N_1 = 20\%$
('+': cascade, 'x': parasite, 'o': standard)

$$G(s) = \frac{1-s}{(2s+1)^2(5s+1)}e^{-0.5s};$$

with non-minimum phase plus dead time. The actual and estimated frequency responses are shown in (b), (c) and (d) of Figure 5.14 and Figure 5.15, respectively. Table 5.6 shows the identification accuracy obtained under different relay tests with/without noise. The identification results are seen to be satisfactory for both parasite and cascade relays. For the noisy case, one can see from

Table 5.6 that significant improvement is achieved by the cascade relay over the parasite relay for most processes. The processes used here for simulation are exactly the same as those in Section 5.4 and thus the results achieved with the cascade relay are typical and representative for the method.

Table 5.6. Identification errors (ERR)

Noise levels	Different relays	Processes			
		$\frac{e^{-5s}}{5s+1}$	$\frac{1}{(s+1)^8}$	$\frac{e^{-2.5s}}{(s+1)(5s+1)}$	$\frac{(1-s)e^{-0.5s}}{(2s+1)^2(5s+1)}$
$N_1 = 0\%$	Cascade	0.30%	0.62%	0.41%	0.31%
$(N_2 = 0\%)$	Parasite	0.31%	0.62%	0.42%	0.32%
w= 0	Standard	11.19%	0.53%	7.06%	3.71%
$N_1 = 10\%$	Cascade	1.87%	2.76%	5.38%	4.39%
$(N_2 = 29\%)$	Parasite	6.83%	6.90%	5.41%	6.38%
w= 0	Standard	10.01%	3.70%	10.73%	9.37%
$N_1 = 20\%$	Cascade	5.80%	3.91%	12.61%	9.62%
$(N_2 = 41\%)$	Parasite	14.52%	6.12%	14.20%	16.96%
w= 0	Standard	15.35%	7.41%	17.20%	25.31%
$N_1 = 10\%$	Cascade	2.01%	4.52%	4.93%	4.14%
$(N_2 = 29\%)$	Parasite	7.17%	6.35%	5.16%	5.13%
w= 0.5	Standard	17.28%	10.08%	36.91%	15.31%

All real processes have some nonlinearity. If the nonlinearity is associated with operating point change (which is the usual case), then the proposed method may be applied to each operating point with a linearized model and gain scheduling can be used to handle this change. When the nonlinearity is modest, our method can be applied without any gain adaptation.

Example 5.5.3. (**Nonlinearity**) Introduce a nonlinearity into a linear model such that the process is described by

$$y = \frac{1}{(s+1)^8} v,$$

where $v = k(u)$ and

$$k(u) = \begin{cases} u, & \text{if } |u| > 0.2, \\ 0, & \text{if } |u| \leqslant 0.2, \end{cases}$$

and u is the process input. Its input and output responses $u(t)$ and $y(t)$ under the cascade relay test are processed as usual with the proposed method to give $G(j\omega)$. Since the frequency response of a nonlinear process is not defined, the effectiveness of the proposed identification method is judged from the control performance. For illustration, a multiple-point fitting method (Wang *et al.*, 1998*a*) for PID tuning is designed with the resultant $\hat{G}(j\omega)$. The refined gain and phase method (Zhuang and Atherton, 1993), which uses only the critical point on the process frequency response available from the standard relay feedback test, is also applied for comparison. The resulting closed-loop response is shown in Figure 5.16, where the solid line is for the proposed method, and the dashed line is for the standard relay feedback test with Zhuang's tuning method. The effectiveness of the proposed method for nonlinear processes is verified.

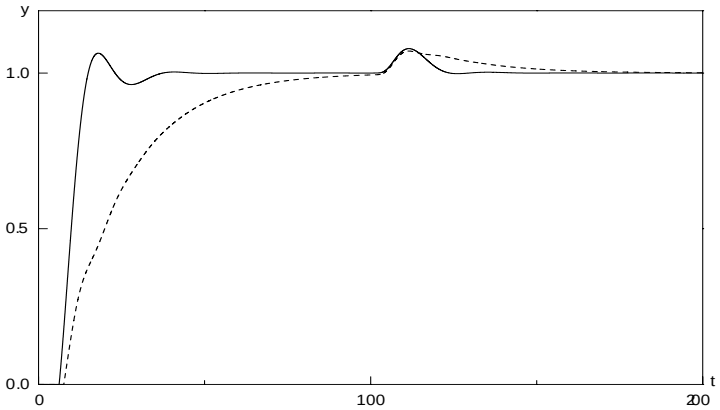


Fig. 5.16. Control performance for $\frac{1}{(s+1)^8}k(u)$

In this section, a new relay, the cascade relay, has been proposed for robust estimation of process frequency response. It shows some improvement over the master-and-parasite relay in terms of estimation results and the likehood of limit cycle generation.

5.6 Extension to MIMO Case

We have so far dealt with SISO processes only. We are looking for extension to the MIMO case. Obviously, it is too tedious and also unnecessary to consider all the types of relays covered so far. Instead, we will illustrate such an extension for the simple relays of Section 5.1 only, and extract the process frequency response matrix at the critical and zero frequencies. When the relay technique is extended to an MIMO system, there are three possible relay feedback schemes.

- *Independent Single Relay Feedback* (*IRF*): Only one loop at a time is subjected to relay feedback while all others are kept open.
- *Sequential Relay Feedback* (*SRF*): A loop is closed with a simple controller once a relay test has been made on that loop. This is repeated until all the loops have been tested.
- *Decentralized Relay Feedback* (*DRF*): All loops are subjected to relay feedback simultaneously, as shown in Figure 5.17.

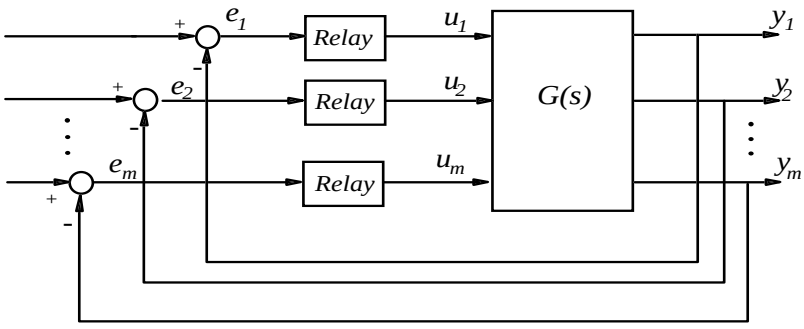


Fig. 5.17. Decentralized relay test

Among the three relay feedback schemes, decentralized relay feedback is the most desirable and will be used as our test for process frequency response matrix estimation in this section. Note that DRF is a *complete* closed-loop test, meaning that for an $m \times m$ plant at any instant during a test, all the m outputs are simultaneously under feedback control, while IRF and SRF are only partial closed-loop tests. For IRF, only one loop is closed, with $(m-1)$ open. For SRF, at the ith test, i loops are closed with $(m-i)$ loops open. Closed-loop testing

is preferred to open-loop testing since a closed-loop test keeps outputs close to the set-points so that it causes less perturbation to the process and makes the linear model assumption (such as frequency response or transfer function) valid.

Analysis of Decentralized Relay Feedback If a $m \times m$ process is controlled by decentralized relay feedback, its outputs usually oscillate in the form of limit cycles after an initial transient. Each output has its own oscillation frequency, denoted ω_{ic}, $i = 1, 2, \ldots, m$, and they are, in general, different. For instance, a 2×2 process consisting of two independent (or very lightly coupled) loops has different output oscillation frequencies. However, it was found in Atherton (1975) that for typical coupled multivariable processes, m outputs normally have the same oscillation frequencies, that is, $\omega_{1c} = \omega_{2c} = \cdots = \omega_{mc}$, but different phases. For ease of reference later, we call this kind of multivariable oscillation *oscillations of a common frequency* and the frequency as a *process critical frequency*, and denote it by ω_c.

The describing function method is extended in Loh and Vasnani (1994) to analyze multivariable oscillations under decentralized relay feedback control. In this context, it is assumed that the m-input and m-output process has low-pass characteristics in each element of its transfer function matrix and one of its characteristic loci has at least 180^o phase lag. Analysis of decentralized relay feedback based on the describing function provides a basic understanding of the behaviour of the resulting system and shows the effects of relay parameters on the behaviour so that insight and guidelines can be gained for the design of such relay tests. It is not intended to be comprehensive, but just to capture the major features of the system, as the analysis is approximate in nature. Therefore, for simplicity, suppose that each relay in the DRF is standard. Let the output amplitudes of standard relays be μ_i, and the inputs to the relays have amplitudes a_i. Then, the describing function matrix of such a decentralized relay controller is

$$N(a, \mu) = diag\left\{\frac{4\mu_i}{\pi a_i}\right\}.$$

Lemma 5.6.1. *(Loh and Vasnani, 1994). If the decentralized relay feedback system oscillates at a common frequency, then at least one of the characteristic loci of $N(a, \mu)G(j\omega)$ crosses the $(-1 + j0)$ point on the complex plane, and the oscillation frequency corresponds to the frequency at which the crossing occurs. Further, if the process is stable, then the limit cycle oscillation is stable, the outermost characteristic locus of $N(a, \mu)G(j\omega)$ passes through the $(-1 + j0)$*

point and the process critical frequency is the same as the critical frequency of the outermost characteristic locus.

It is noted that the crossing condition and the oscillation frequency in Lemma 5.6.1 are related to $N(a,\mu)$, which cannot be calculated until the oscillations are observed and the amplitudes a_i of relay inputs measured from the oscillation waveforms. It would be useful if the frequency could be given in terms of the information on the process only but independent of the relay controller. To this end, consider an $m \times m$ multivariable process $G(s)$ with row Gershgorin bands as shown in Figure 5.18. For each band, let $c_{i1} = g_{ii}(\omega_{i1})$ and $c_{i2} = g_{ii}(\omega_{i2})$ be the centres of the circles which are tangential to the negative real axis, and $(-\beta_{i1} + j0)$ and $(-\beta_{i2} + j0)$ be the points at which the outer-rim and inter-rim of the ith Gershgorin band intersect the negative real axis respectively. If the ith Gershgorin band does not intersect the negative real axis, $[\omega_{i1},\ \omega_{i2}]$ is defined to be empty. The following result gives an estimate for ω_c in terms of ω_{i1} and ω_{i2}.

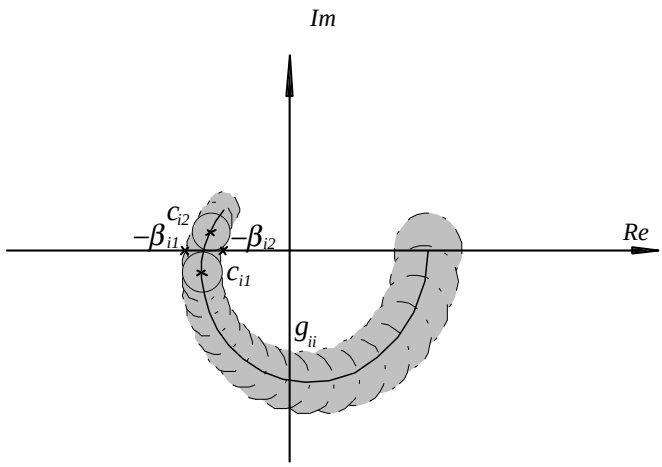


Fig. 5.18. Gershgorin bands

Proposition 5.6.1. *If the decentralized relay feedback system oscillates at a common frequency ω_c, then there exists a $k \in \{1, 2, \ldots, m\}$ such that $\omega_c \in [\omega_{k1},\ \omega_{k2}]$.*

Proof. By the Gershgorin theorem (Maciejowski, 1989), we know that the characteristic loci of $G(s)$ lie in the union of its Gershgorin bands. The point at which the i-th characteristic locus $\lambda_i(j\omega)$ of G crosses the negative real axis if it exists can only lie in the union of circles with centres from c_{i1} to c_{i2}. It follows that for $\lambda_i(j\omega)$, the critical frequency ω_{ic} at which the crossing occurs is in the range $[\omega_{i1},\ \omega_{i2}]$. Suppose now that the transfer function matrix $G(s)$ is multiplied by a diagonal constant matrix $K = diag\,\{k_i\}$ as

$$Q = KG = \begin{bmatrix} k_1 g_1 \\ \vdots \\ k_i g_i \\ \vdots \\ k_m g_m \end{bmatrix}$$

where g_i, $i = 1, \ldots, m$ are the row vectors of $G(s)$. The centres of the circles for the ith Gershgorin band of Q have now been shifted to $k_i g_{ii}$ with the radii of the circles magnified k_i times as shown in Figure 5.19. Since k_i is constant, the centre $k_i g_{ii}(\omega_{i1})$ has the same phase as that of $c_{i1} = g_{ii}(\omega_{i1})$ and is on the straight line drawn through the origin and $g_{ii}(\omega_{i1})$. Further, the magnitude $|k_i g_{ii}(\omega_{i1})|$ differs from $|g_{ii}(\omega_{i1})|$ by a factor $|k_i|$. Therefore, the distance between the point $k_i g_{ii}(\omega_{i1})$ and the negative real axis is $|k_i|$ times as large as that between the point $g_{ii}(\omega_{i1})$ and the axis, which is exactly the radius of the circle with centre $k_i g_{ii}(\omega_{i1})$. This implies that this circle is still tangential to the negative real axis and thus $\tilde{\omega}_{i1}$ for Q is equal to ω_{i1} for G. The same can be said for c_{i2} and $\tilde{\omega}_{i2} = \omega_{i2}$. It follows that the critical frequency $\tilde{\omega}_{ic}$ for the ith characteristic locus of $Q(s)$ is still in $[\omega_{i1},\ \omega_{i2}]$. Since the describing matrix $N(a, \mu)$ is also a constant diagonal matrix, the critical frequency for the ith characteristic locus of $N(a, \mu)G(s)$ is thus in $[\omega_{i1},\ \omega_{i2}]$. By Lemma 5.6.1, the limit cycle oscillation frequency must be in one of $[\omega_{i1},\ \omega_{i2}]$, $i = 1, 2, ..., m$ and our result follows.

In view of Lemma 5.6.1 and Proposition 5.6.1, the oscillation frequency ω_c for a stable process depends on which characteristic locus of $G(s)$ is moved to the outermost by the multiplication of the corresponding relay element describing function $N_i = \frac{4\mu_i}{\pi a_i}$. In general, one can enlarge the gain N_i by increasing the ratios of the relay amplitudes in the ith loop to those in other loops. We call this outermost loop the *dominant* loop. It is noted that the dominant loop remains dominant and the critical frequency varies very little with a fairly large

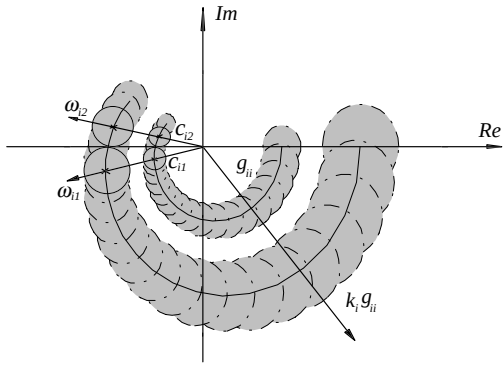


Fig. 5.19. Expansion of a Gershgorin band

change of relay amplitude ratios unless an inner characteristic locus becomes a new outermost. As an example, consider the following typical process (Wood and Berry, 1973):

$$G(s) = \begin{bmatrix} \frac{12.8e^{-s}}{1+16.7s} & \frac{-18.9e^{-3s}}{1+21s} \\ \\ \frac{6.6e^{-7s}}{1+10.9s} & \frac{-19.4e^{-3s}}{1+14.4s} \end{bmatrix}.$$

Let μ_1 and μ_2 be the relay amplitudes in loop 1 and loop 2 respectively. When $r := \frac{\mu_1}{\mu_2}$ varies from 1 to 2 by 100%, the process always exhibits oscillations of a common frequency, and the process critical frequency ω_c changes from 0.494 to 0.496, i.e., by 0.4%. This feature is addressed in the following proposition.

Proposition 5.6.2. *If the decentralized relay feedback system for a stable process oscillates at a common frequency and for some k, $N_k > \frac{N_i \beta_{i1}}{\beta_{k2}}$, $i = 1, 2, \ldots, m$, $i \neq k$, then only the kth characteristic locus of $N(a,\mu)G(j\omega)$ crosses the $(-1+j0)$ point and the oscillation frequency satisfies $\omega_c \in [\omega_{k1},\ \omega_{k2}]$.*

Proof. The conditions, $N_k > \frac{N_i \beta_{i1}}{\beta_{k2}}$, $i = 1, 2, \ldots, m$, $i \neq k$, guarantee that the kth Gershgorin band of $N(a,\mu)G(j\omega)$ is the outermost among all the m Gershgorin bands. Since the kth characteristic locus of $N(a,\mu)G(j\omega)$ is in this band, it is the outermost locus of $N(a,\mu)G(j\omega)$. It follows from Lemma 5.6.1 that the kth characteristic locus of $N(a,\mu)G(j\omega)$ crosses the $(-1+j0)$ point and the oscillation frequency ω_c is equal to ω_{kc}, which is in $[\omega_{k1},\ \omega_{k2}]$.

By Proposition 5.6.2, if we vary the relay amplitudes such that the resulting describing function gain matrix $N'(a,\mu)$ still satisfies $N_k' > \frac{N_i'\beta_{i1}}{\beta_{k2}}$, $i = 1, 2, \ldots, m$, $i \neq k$, then the resulting limit cycle oscillation frequency is expected to be in the range $[\omega_{k1},\ \omega_{k2}]$ and thus close to the previous value if the interval $[\omega_{k1},\ \omega_{k2}]$ is small. In general, the condition $N_k > \frac{N_i\beta_{i1}}{\beta_{k2}}$, $i = 1, 2, \ldots, m$, $i \neq k$, remains true if one increases the relay amplitude of the dominant loop or decreases one or more relay amplitudes in the other loops.

Estimation of Process Frequency Response An $m \times m$ multivariable process can be described in the frequency domain as

$$\begin{bmatrix} y_1(j\omega) \\ \vdots \\ y_m(j\omega) \end{bmatrix} = \begin{bmatrix} g_{11}(j\omega) & \ldots & g_{1m}(j\omega) \\ \vdots & \ddots & \vdots \\ g_{m1}(j\omega) & \ldots & g_{mm}(j\omega) \end{bmatrix} \begin{bmatrix} u_1(j\omega) \\ \vdots \\ u_m(j\omega) \end{bmatrix}. \tag{5.53}$$

We want to estimate the process frequency response $G(j\omega)$ at the critical oscillation frequency ω_c. In order to additionally identify the steady-state gain matrix of the process, a biased relay instead of a standard relay should be used in the dominant loop to make the process inputs and outputs have nonzero means. Thus, a test as shown in Figure 5.17 with a biased relay in the dominant loop and symmetric relays in the other loops is applied to the process. When the process becomes stationary, the process stationary inputs $u_i(t)$ and outputs $y_i(t)$, $i = 1, 2, \ldots, m$, are all periodic, and can be expanded into Fourier series. If the oscillations in m loops have a common frequency ω_c, then the direct-current components and the first harmonics of these periodic waves are extracted as

$$U^1(0) := \begin{bmatrix} \int_0^{T_c} u_1(t)dt \\ \vdots \\ \int_0^{T_c} u_m(t)dt \end{bmatrix}, \quad Y^1(0) := \begin{bmatrix} \int_0^{T_c} y_1(t)dt \\ \vdots \\ \int_0^{T_c} y_m(t)dt \end{bmatrix} \tag{5.54}$$

and

$$U^1(j\omega_c) := \begin{bmatrix} \int_0^{T_c} u_1(t)e^{-j\omega_c t}dt \\ \vdots \\ \int_0^{T_c} u_m(t)e^{-j\omega_c t}dt \end{bmatrix}, \quad Y^1(j\omega_c) := \begin{bmatrix} \int_0^{T_c} y_1(t)e^{-j\omega_c t}dt \\ \vdots \\ \int_0^{T_c} y_m(t)e^{-j\omega_c t}dt \end{bmatrix}. \tag{5.55}$$

Then

$$Y^1(0) = G(0)U^1(0), \tag{5.56}$$

and

$$Y^1(j\omega_c) = G(j\omega_c)U^1(j\omega_c). \tag{5.57}$$

Since (5.56) and (5.57) are vector equations, they are not sufficient to determine $G(j\omega_c)$ and $G(0)$ from Y^1 and U^1 only. Next, we slightly increase the relay amplitude of the dominant loop or decrease that of another loop, and repeat the above procedure until m tests have been completed. According to Proposition 5.6.2, the process is likely to have all the oscillation frequencies close to each other for the m tests. $Y^2(0), U^2(0), Y^2(j\omega_c), U^2(j\omega_c), \ldots, Y^m(0), U^m(0), Y^m(j\omega_c), U^m(j\omega_c)$ are obtained subsequently. We have

$$[Y^1(0) \quad \ldots \quad Y^m(0)] = G(0)[U^1(0) \quad \ldots \quad U^m(0)], \tag{5.58}$$

and

$$[Y^1(j\omega_c) \quad \ldots \quad Y^m(j\omega_c)] = G(j\omega_c)[U^1(j\omega_c) \quad \ldots \quad U^m(j\omega_c)]. \tag{5.59}$$

While (5.58) is accurate for any decentralized relay test, (5.59) is only approximate since ω_c is not exactly the same for all m tests. U^i, $i = 1, 2, \ldots, m$, are linearly independent since there is always a relay amplitude change for each test. It follows from (5.58) and (5.59) that the steady-state gain matrix $G(0)$ and frequency response matrix $G(j\omega_c)$ are determined, respectively, as

$$G(0) = [Y^1(0) \quad \ldots \quad Y^m(0)][U^1(0) \quad \ldots \quad U^m(0)]^{-1}, \tag{5.60}$$

and

$$G(j\omega_c) = [Y^1(j\omega_c) \quad \ldots \quad Y^m(j\omega_c)][U^1(j\omega_c) \quad \ldots \quad U^m(j\omega_c)]^{-1}. \tag{5.61}$$

Our relay experiment thus consists of m decentralized relay tests and continues from one to another without any stop in between. To design this experiment, one needs to specify relay amplitudes for each test. The following design parameters are recommended for use and are obtained through our extensive case studies. For the first test, the relay amplitude for each loop is set as in the single-variable case (see Section 1). In most circumstances, stationary oscillations of a common frequency will result in the system. For subsequent tests, either the relay amplitude in the dominant loop is increased or the relay amplitude in one of the other loops is decreased by 5–20%. This usually leads to oscillations with frequencies close to the previous ones.

It should be pointed out that m decentralized relays in our test scheme are reasonable and even necessary to identify an $m \times m$ system. Out test scheme

may actually need *less* time than those for IRF and SRF. To see this, our scheme uses m *non-stop* relays, while both IRF and SRF also contain m relays for a $m \times m$ system. Furthermore, between their m relays, there are additional $(m-1)$ control transients to bring outputs back to the set-points before the next relays can be performed. In the context of resonance approximations, the number of relays should be at least m in order to identify an $m \times m$ frequency response matrix $G(j\omega)$, as explained above. In our opinion, the main shortcoming of the decentralized relay test is that it may cause complicated multivariable oscillations (Atherton, 1975; Loh *et al.*, 1993; Zhuang and Atherton, 1993), where three modes of multivariable oscillations have been observed. If there are no oscillations or the oscillations have different frequencies at different outputs, our method cannot be used and this is a restriction on it. However, oscillations with a common frequency is the mode most likely to occur (Atherton, 1975) when the process has significant interaction, which is the case considered in this section.

Noise is an important issue in the identification problem. Like the SISO case, anti-noise measures such as hysteresis, low-pass filtering and multiple oscillation periods can also be used in the present case of a DRF for each relay. No further discussion is required.

Example 5.6.1. Consider the well-known Wood/Berry binary distillation column plant (Wood and Berry, 1973):

$$G(s) = \begin{bmatrix} \frac{12.8e^{-s}}{1+16.7s} & \frac{-18.9e^{-3s}}{1+21s} \\ \\ \frac{6.6e^{-7s}}{1+10.9s} & \frac{-19.4e^{-3s}}{1+14.4s} \end{bmatrix}.$$

It is a typical MIMO plant with strong interaction and significant time delays. For a tuning test, the relay in loop 1 is set as a symmetric relay with output switching levels of 1.00 and -1.00, and a relay with bias in its output giving switching levels 1.50 and -1.00 is used in loop 2. The system exhibits limit cycle oscillations having a common frequency with frequency $\omega_c^1 = 0.485$. The switching levels of the relay in loop 2 is then changed to 1.80 and -1.20. The system exhibits limit cycle oscillations having a common frequency with $\omega_c^2 = 0.484$ in this case. The steady-state gain matrix $\hat{G}(0)$ and frequency response matrix $\hat{G}(\omega_c)$ are computed from (5.60) and (5.61) as

$$\hat{G}(0) = \begin{bmatrix} 12.8 & -18.9 \\ 6.60 & -19.4 \end{bmatrix}, \text{ and } \hat{G}(j\omega_c) = \begin{bmatrix} 1.56e^{-1.92j} & 18.6e^{0.221j} \\ 1.21e^{1.46j} & 2.79e^{0.260j} \end{bmatrix},$$

where ω_c=$\frac{1}{2}(\omega_c^1+\omega_c^2)$=0.485. They are very accurate, compared with their true values:

$$G(0)=\begin{bmatrix}12.8 & -18.9\\ 6.6 & -19.4\end{bmatrix},\ \text{and}\ G(0.485j)=\begin{bmatrix}1.57e^{-1.93j} & 18.5e^{0.21j}\\ 1.23e^{1.50j} & 2.75e^{0.26j}\end{bmatrix}.$$

Example 5.6.2. Consider the process in Palmor *et al* (1993):

$$G(s)=\begin{bmatrix}\frac{0.5}{(0.1s+1)^2(0.2s+1)^2} & \frac{-1}{(0.1s+1)(0.2s+1)^2}\\ \frac{1}{(0.1s+1)(0.2s+1)^2} & \frac{2.4}{(0.1s+1)(0.2s+1)^2(0.5s+1)}\end{bmatrix}.$$

There are large interactions in this process. Two decentralized relay tests are performed on it. The relay in loop 1 is symmetric with unit switching levels, and the switching levels of the relay in loop 2 are 1.40 and -0.933 in the first test and changed to 1.50 and -1.00 in the second. Both tests result in limit cycle oscillations with the same frequency $\omega_c = 4.29$. The estimated steady-state gain matrix $\hat{G}(0)$ and frequency response matrix $\hat{G}(j\omega_c)$ are

$$\hat{G}(0)=\begin{bmatrix}0.500 & -1.00\\ 1.00 & 2.40\end{bmatrix},\quad \hat{G}(4.29j)=\begin{bmatrix}0.243e^{-2.25j} & 0.529e^{1.31j}\\ 0.529e^{-1.84j} & 0.537e^{-2.98j}\end{bmatrix}$$

while the true values are

$$G(0)=\begin{bmatrix}0.5 & -1\\ 1 & 2.4\end{bmatrix},\quad G(4.29j)=\begin{bmatrix}0.24e^{-2.23j} & 0.53e^{1.32j}\\ 0.53e^{-1.82j} & 0.54e^{-2.96j}\end{bmatrix}$$

In this section, multivariable oscillations under decentralized relay feedback control have been investigated. In particular, it is shown that for a stable $m\times m$ process, the oscillation frequencies remain almost unchanged under relatively large relay amplitude variations. Therefore, if m decentralized relay feedback tests are performed on the process, their oscillation frequencies are close to each other so that the process frequency response matrix can be estimated at that frequency. A bias may be introduced into the relay to additionally obtain the process steady-state matrix.

9. Single-variable Systems

The internal model control (IMC) is a powerful framework for control system design and implementation (Morari and Zafiriou, 1989), and it has sound theoretical foundation. Its stability analysis is extremely easy to carry out and the design trade-off between performance and robustness is clearly understood. It has attracted the attention of industrial users because there is only one user-defined tuning parameter, which is directly related to the closed-loop time constant or equivalently, the closed-loop bandwidth. On the other hand, the vast majority of controllers being used in industry are of the PID type due to its simplicity and popularity (Åström and Hägglund, 1995). Recently, great efforts have been made to develop PID tuning strategies for more general processes (Barnes *et al.*, 1993; Sung and Lee, 1996; Sung *et al.*, 1996; Datta *et al.*, 2000). Each method was derived for its particular optimization objectives and plant model assumptions, and therefore performs well only for its own class. It is common that practising control engineers may not be certain which tuning method should be chosen to provide good control in a given process. It would hence be desirable to develop a design method that works universally with high performance for general stable linear processes, and is capable of producing a high-order controller when the PID controller is no longer adequate.

This chapter presents a unified framework for control system design. The IMC controller is always designed first. If the IMC scheme cannot be implemented, the equivalent controller in a conventional unity output feedback configuration is derived from the IMC controller and simplified by model reduction to a realizable controller, whose structure can be specified by users as a PID type or general rational function type to suit real situations best. In this chapter, we exclusively consider stable processes except the last section where the method is extended to unstable processes.

9.1 Design Methodology

The schematic of the IMC system is depicted in Figure 9.1, where $G(s)$ is the given stable process to be controlled, $\hat{G}(s)$ a model of the process and $C(s)$ the IMC primary controller. The design procedure for IMC systems is

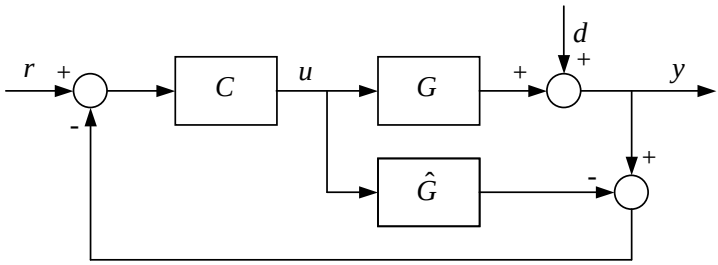


Fig. 9.1. IMC control system

well documented (Morari and Zafiriou, 1989), and highlighted as follows. The model is factorized as

$$\hat{G}(s) = \hat{G}_+(s)\hat{G}_-(s), \tag{9.1}$$

such that $\hat{G}_+(s)$ contains all the dead time and right half-plane zeros of $\hat{G}(s)$:

$$\hat{G}_+(s) = e^{-Ls}\left(\prod_i \frac{1-\beta_i s}{1+\beta_i s}\right), \quad \mathrm{Re}\,(\beta_i) > 0, \tag{9.2}$$

while $\hat{G}_-(s)$ is stable and of minimum phase with no predictors. The primary controller takes the form:

$$C = \hat{G}_-^{-1} f, \tag{9.3}$$

where f is a user-specified low-pass filter and usually chosen as

$$f(s,\tau) = \frac{1}{(\tau s+1)^m}, \tag{9.4}$$

where m is sufficiently large to guarantee that the IMC controller C is proper. τ is the only tuning parameter to be selected by the user to achieve the appropriate compromise between performance and robustness and to keep the action of the manipulated variable within bounds. A smaller τ provides faster closed-loop response but the manipulated variable is moved more vigorously, while a larger τ provides a slower but smoother response. A larger τ is also

less sensitive to model mismatches. In process control practice, the closed-loop bandwidth ω_{cb} can rarely exceed ten times the open-loop process bandwidth ω_{pb} (Morari and Zafiriou, 1989), i.e., $\omega_{cb} \leq 10\omega_{pb}$. Usually, the desired closed-loop bandwidth is chosen as $\omega_{cb} = \gamma\omega_{pb}$, $\gamma \in [0.5, 10]$. Using (9.4), it can be readily seen that

$$\tau = \frac{\sqrt{\sqrt[m]{2} - 1}}{\gamma\omega_{pb}}, \quad \gamma \in [0.5,\ 10]. \tag{9.5}$$

In the case of model uncertainty, τ should be increased just enough to meet the condition for which the system is robustly stable (Morari and Zafiriou, 1989).

In order to keep the action of the manipulated variable within bounds, we use a frequency-by-frequency analysis (Skogestad and Postlethwaite, 1996). Assume that at each frequency $|U(j\omega)| \leq \bar{U}$ and $|R(j\omega)| \leq \bar{R}$. The manipulated variable meets

$$U(s) = C(s)R(s) = \frac{1}{(\tau s + 1)^m}\hat{G}_-^{-1}(s)R(s).$$

One requires

$$\frac{1}{(\tau j\omega + 1)^m}\hat{G}_-^{-1}(j\omega)R(j\omega) \leq \bar{U}. \tag{9.6}$$

Consider the worst case of $|R(j\omega)| = \bar{R}$, and we require

$$\left|\frac{1}{(\tau j\omega + 1)^m}\right| \leq \left|\hat{G}_-(j\omega)\right| \frac{\bar{U}}{\bar{R}}. \tag{9.7}$$

To derive an inequality on τ imposed by input constraints, let $\omega = \omega_{ob}$, where ω_{ob} is the open-loop bandwidth, and notice that $\left|\hat{G}_-(j\omega_{ob})\right| = \frac{1}{\sqrt{2}}$, we have

$$\left|\frac{1}{(\tau j\omega_{ob} + 1)^m}\right| \leq \frac{1}{\sqrt{2}}\frac{\bar{U}}{\bar{R}}, \tag{9.8}$$

i.e.,

$$\tau \geq \sqrt{\sqrt[m]{\frac{2\bar{R}^2}{\bar{U}^2}} - 1}\Big/\omega_{ob}. \tag{9.9}$$

We choose τ to meet (9.5), (9.9) and any possible robustness specification. Then, the IMC control system has been designed and can be implemented according to Figure 9.1 with the controller in (9.3). To see performance for the case of no plant–model mismatch, the nominal closed-loop transfer function of the IMC system between the set point r and output y is

$$H = \hat{G}_+ f = (\prod_i \frac{1-\beta_i s}{1+\beta_i s}) \frac{1}{(\tau s+1)^m} e^{-Ls}. \tag{9.10}$$

If a user prefers a conventional (or single-loop) feedback control configuration, instead of the IMC scheme, for whatever reason, we can derive a controller for such a configuration from the IMC controller. This involves two issues. One is to convert the IMC system to an equivalent single-loop system. The other is to de-tune the IMC controller parameter to reflect the difference between the two control schemes and thus achievable performance limitations so that the performance from the properly de-tuned IMC system can be achieved by its single-loop (SL) equivalent.

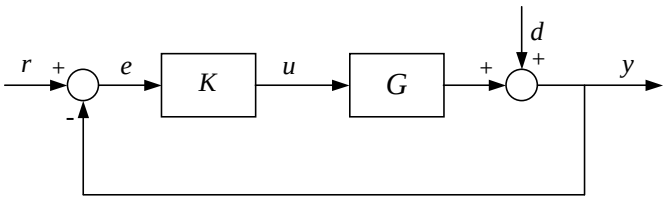


Fig. 9.2. Single-loop control system

The IMC system in Figure 9.1 can be formally redrawn into the equivalent single-loop (SL) feedback system in Figure 9.2, if the SL controller K is related to the IMC controller C via

$$K(s,\tau) = \frac{C(s,\tau)}{1-\hat{G}(s)C(s,\tau)}. \tag{9.11}$$

In Chien (1988), K is chosen as the PID type, and the value of τ is set according to (9.5) as if the single-loop PID controller could achieve the same performance as that of the more complex IMC controller. The dead time is approximated by either a first-order Padé or first-order Taylor series, and the PID controller parameters are obtained by matching the first few *Markov* coefficients of (9.11) for the selected specific process models. The results are listed in Table 9.1. However, it is noted that the use of the Padé approximation or a first-order Taylor expansion introduces extra modelling errors. Furthermore, Chien's rules are applicable only to first-order plus dead time (FOPDT) and second-order plus dead time (SOPDT) processes. This inevitably restricts the general applicability of the method and the performance of the resulting controller.

Our IMC-based design methodology described here can yield the best single-loop controller approximation to the IMC controller regardless of process order

Table 9.1. Chien's IMC–PID rules

Process model	$k_c k_p$	T_I	T_D
$\frac{k_p e^{-Ls}}{1+\tau_1 s}$	$\frac{\tau_1}{\tau+L}$	τ_1	–
$\frac{k_p(\tau_3 s+1)e^{-Ls}}{(\tau_1 s+1)(\tau_2 s+1)}$	$\frac{\tau_1+\tau_2-\tau_3}{\tau+L}$	$\tau_1+\tau_2-\tau_3$	$\frac{\tau_1\tau_2-(\tau_1+\tau_2-\tau_3)\tau_3}{\tau_1+\tau_2-\tau_3}$
$\frac{k_p(-\tau_3 s+1)e^{-Ls}}{(\tau_1 s+1)(\tau_2 s+1)}$	$\frac{\tau_1+\tau_2+\frac{\tau_3 L}{\tau+\tau_3+L}}{\tau+\tau_3+L}$	$\tau_1+\tau_2+\frac{\tau_3 L}{\tau+\tau_3+L}$	$\frac{\tau_3 L}{\tau+\tau_3+L}+\frac{\tau_1\tau_2}{\tau_1+\tau_2+\frac{\tau_3 L}{\tau+\tau_3+L}}$
$\frac{k_p e^{-Ls}}{s}$	$\frac{2\tau+L}{(\tau+L)^2}$	$2\tau+L$	–

and characteristics. The resulting single-loop performance can be better guaranteed and well predicted from the IMC counterpart. Our design idea is very simple: given the equivalent single-loop controller K in (9.11), which may be unnecessarily complicated to implement, apply a suitable model reduction to obtain the best approximation $\hat{K}$ to K. If the user specifies the type of $\hat{K}$ (say, PID), then the model reduction algorithm will generate its parameters. If the approximation accuracy is satisfactory, the design is completed; otherwise, the algorithm will adjust the IMC controller performance until its single-loop approximation is satisfactory. On the other hand, if the user has no preferred controller structure, our algorithm starts with a PID type, and gradually increases the controller complexity such that the simplest approximation $\hat{K}$ is attained with the guaranteed accuracy to K. This allows a unified treatment of all cases and facilitates auto-tuning applications.

A crucial issue in IMC-SL controller design is to get a suitable value for τ which leads to a good single-loop controller approximation to the corresponding IMC one. Note the inherent difference between IMC and SL systems in their configurations (Figures 9.1 and 9.2) where the former has output prediction while the latter does not. In fact, not all IMC systems can be approximated reasonably by single-loop systems (see the remark at the end of Section 9.3). The τ given by (9.5) is suitable for IMC systems, but it does not consider the performance limitations of single-loop feedback systems due to non-minimum-phase zero and dead time. Such limitations are usually expressed by integral relationships (Freudenberg and Looze, 1987). Recently, Åström (2000) proposed the following simple non-integral inequality for the gain crossover frequency ω_{og}

of the open-loop transfer function $\hat{G}K$, where

$$|\hat{G}(j\omega_{og})K(j\omega_{og},\tau)| = 1, \tag{9.12}$$

to meet

$$\arg \hat{G}_+(j\omega_{og}) \geq -180^\circ + \phi_m - \arg \hat{G}_-(j\omega_{og})K(j\omega_{og},\tau), \tag{9.13}$$

where ϕ_m is the desired phase margin. The selection of ϕ_m reflects the control system robustness to process uncertainty (Åström, 2000): large ϕ_m is required for large uncertainty. With a lack of information on uncertainty size, a typical range for ϕ_m would be 30°–80°. Our design objective is to achieve a non-oscillatory response as specified by (9.10) and yet have the response as fast as possible. This translates to a damping ratio of approximately $\xi = 0.7$, and the empirical formula $\phi_m = 100\xi$ (Franklin *et al.*, 1990) yields an estimate of $\phi_m = 70^\circ$ for $\xi = 0.7$. Our studies suggest that $\phi_m = 65^\circ$ is usually a good choice and we use this ϕ_m throughout this chapter. With ϕ_m specified, we then find the smallest τ^* which satisfies (9.12) and (9.13).

In short, for single-loop controller design the tuning parameter τ in the filter (9.4) should be, in general, chosen to meet (9.5), (9.9), (9.12) and (9.13) simultaneously. If the process is of minimum phase, (9.12), (9.13) vanish, while (9.5) and (9.9) are in action. On the other hand, if the process has any non-minimum element, our study shows that the τ derived from (9.9), (9.12) and (9.13) always appears in the range given in (9.5) so that (9.9), (9.12) and (9.13) would be enough to determine τ in this case. In the subsequent two sections, PID and general controllers are considered in detail.

9.2 PID Controller

Owing to its simple structure, the PID controller is the most widely used controller in the process industry, even though many advanced control algorithms have been introduced. Consider a PID controller in the form:

$$K_{PID} = k_p + \frac{k_i}{s} + k_d s, \tag{9.14}$$

where k_p is the proportional gain, k_i the integral gain (units of time), and k_d the derivative gain (units of time). Our task is to find the three PID parameters, so as to match $\hat{K} = K_{PID}$ to $K = \frac{C}{1-C\hat{G}}$ as well as possible. This objective can be realized by minimizing the loss function,

$$\min_{K_{PID}} J \triangleq \min_{K_{PID}} \sum_{i=1}^{M} |K_{PID}(j\omega_i) - K(j\omega_i)|^2, \quad k_p, k_i, k_d > 0, \tag{9.15}$$

whose solution is obtained by standard non-negative least squares to give the optimal PID parameters as $\left[k_p^* \; k_i^* \; k_d^*\right]^T = \theta^*$. Our studies suggest that the frequency range $[\omega_1, \; \omega_M]$ in the optimal fitting (9.15) be chosen as $(0.1\omega_{cb}, \; \omega_{cb})$ with steps of $(\frac{1}{100} \sim \frac{1}{10})\omega_{cb}$, where ω_{cb} is the desired closed-loop bandwidth.

Once a PID controller is found, the following criterion should be used to validate the solution:

$$ERR = \max_{\omega \in [0, \omega_{cb}]} \left| \frac{\hat{K}(j\omega) - K(j\omega)}{K(j\omega)} \right| \leq \epsilon, \tag{9.16}$$

where ϵ is the user-specified fitting error threshold. ϵ is specified according to the desired degree of performance, or accuracy of the SL approximation to the IMC one. Usually ϵ may be set as 3%. If (9.16) holds true, the design is complete.

On the other hand, if the given threshold cannot be met, one can always detune the PID controller by relaxing the IMC specification, i.e., increasing τ. A typical relationship between the tuning parameter τ and the approximation error is shown in Figure 9.3. In general, ERR decreases as τ increases. It provides a simple way to select a minimum τ with respect to the specific accuracy threshold. In practice, however, it is inconvenient to draw such a curve. It is found that the decreasing rate $d(ERR)/d\tau$ is highly influenced by plant dead time L and the right half-plane (RHP) zeros β_i^{-1}, which limit the achievable bandwidth. ω_{cb} is virtually unaffected by the presence of the filter (Rivera *et al.*, 1986) until τ reaches an order of magnitude comparable to L and β_i, respectively. Hence, it is effective and efficient to choose the increment of τ in the PID detuning procedure as the maximum of L and $\mathrm{Re}\,\beta_i$, i.e.,

$$\tau^{k+1} = \tau^k + \eta^k \max\left(L, \; \min_i(\mathrm{Re}\,(\beta_i))\right) \tag{9.17}$$

where k represents the kth iteration, and η is an adjustable factor reflecting the approximation accuracy of the present iteration and is set at $\frac{1}{4}$, $\frac{1}{2}$ and 1, when $3\% < ERR \leq 20\%$, $20\% < ERR \leq 100\%$ and $100\% < ERR$, respectively. The iteration continues until the accuracy bound is fulfilled.

Our detuning rule for τ in (9.17) implicitly assumes that ERR would be sufficiently small when τ is large enough. In this connection, it would be interesting to see if $\lim_{\tau \to \infty} ERR = 0$. Equation (9.5) can be rewritten as

$$\omega_{cb} = \frac{\sqrt{\sqrt[m]{2} - 1}}{\tau}. \tag{9.18}$$

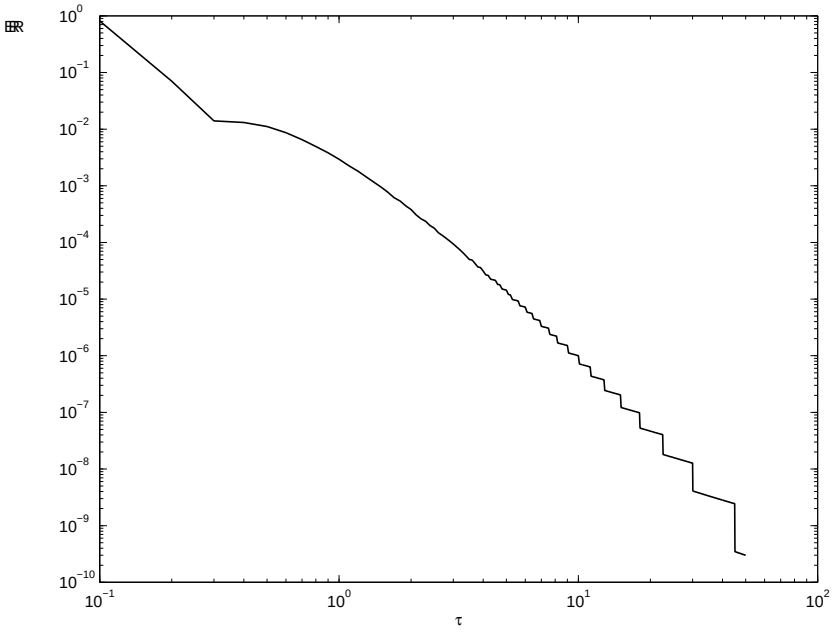


Fig. 9.3. Relationship between filter parameter (τ) and approximation error (ERR)

When τ increases to infinity, it is easy to see from (9.18) that $\lim_{\tau\to\infty} \omega_{cb} = 0$, $G(j\omega)$ can be replaced by $G(0)$ for $\omega \leq \omega_{cb}$, and K becomes $\frac{1}{G_-(0)((\tau s+1)^m-1)}$. For $m = 1$, $K = \frac{1}{sG_-(0)}$ is a pure integrator and can be realized precisely by a PID controller with no error. In general, the Nyquist curve of $K(j\omega)$ for $\omega \in (0, \omega_{cb})$ approaches a straight line as shown in Figure 9.4, when τ tends to infinity. Note that the Nyquist curve for the PID controller is always a vertical straight line, and can match that of $K(j\omega)$ as well as desired for $\tau \to \infty$. One thus expects ERR to converge to 0 as τ approaches infinity.

We now present some simulation examples to demonstrate our PID tuning algorithm and compare it with the original IMC and the PID tuning in Chien (1988). Chien (1988) implemented the following PID form:

$$\tilde{K}_{PID} = K_c(1 + \frac{1}{T_I s} + \frac{T_D s}{\frac{T_D}{N} s + 1}),$$

where the PID settings are given in Table 9.1. The ideal PID controller in (9.14) used for our algorithm development is not physically realizable and thus is replaced by

$$K_{PID} = k_c + \frac{k_i}{s} + \frac{k_d s}{\frac{k_d}{N} s + 1}. \tag{9.19}$$

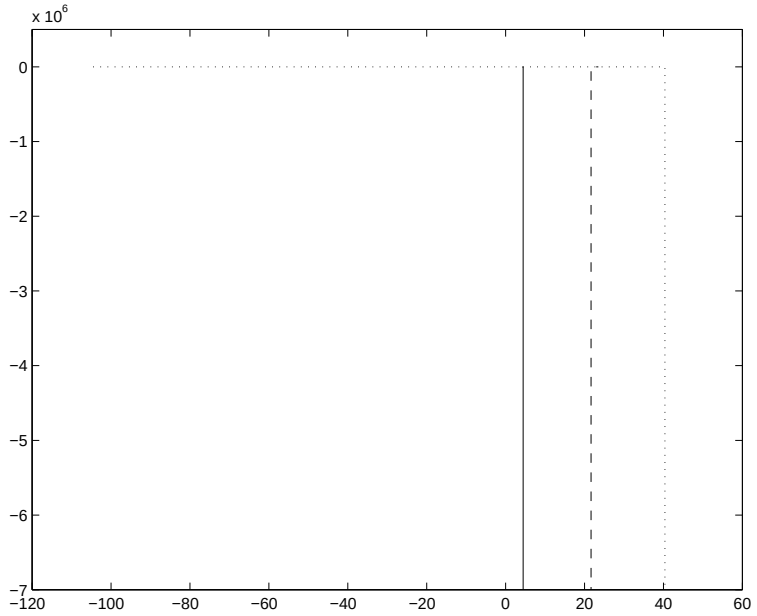


Fig. 9.4. Nyquist curve of K up to ω_{cb} ($m = 3$)
($\cdots \tau = 0.1$, $- - - \tau = 0.3$, —— $\tau = 1$)

In both cases, N is suggested to be chosen in the range $[5, 20]$. Simulations are done under the perfect model matching condition, i.e., $\hat{G} = G$ (model mismatch will be considered in Section 9.4). To have fair and comprehensive assessment of controller performance, most performance indices popularly used in process control are measured and they include both time domain ones such as percentage overshoot (M_p), rise time (from 10% to 90%) in seconds (t_r), setting time (to 1%) in seconds (t_s), integral absolute-error ($IAE = \int_0^\infty |r - y| dt$ where the upper limit ∞ may be replaced by T, which is chosen sufficiently large so that $e(t)$ for $t > T$ is negligible); and frequency domain error ERR defined in (9.16). Simulations were made for three typical plants, and the results are shown in Table 9.2.

Example 9.2.1. Consider a first-order plus dead time process:

$$G = \frac{e^{-0.5s}}{s+1}.$$

From (9.9), (9.12) and (9.13), one can readily find $\tau^* = 0.3333$. Then, Chien's formula gives a PI controller,

$$\tilde{K}_{PI} = 1.2029(1 + \frac{1}{s}),$$

Table 9.2. Simulation results for stable processes

Plant	τ	Scheme	$M_p(\%)$	t_r	t_s	$U_{\max}$	$ERR(\%)$	IAE
$G=\frac{e^{-0.5s}}{s+1}$	0.3333	Chien's PID	11.7728	0.72	4.15	1.8044	42.59	1.0571
		Proposed PID	0	0.73	2.31	6.3945	1.23	0.8351
		IMC	0	0.73	2.23	3.0184	0	0.8343
$G=\frac{(-0.5s+1)e^{-s}}{(s+1)(2s+1)}$	0.8328	Chien's PID	14.0690	1.54	11.66	6.3779	26.33	2.8728
		Proposed PID	0.9339	2.11	8.46	6.1204	2.66	2.8388
		IMC	0	2.17	6.18	4.8031	0	2.8378
$G=\frac{e^{-2s}}{(s^2+s+1)(s^2+0.6s+1)}$	0.6687	Chien				− − −		
		Proposed PID	7.4365	2.67	10.00	5.2860		17.48
		Proposed high-order	0	3.30	8.72	3.5488		0.0045
		IMC	0	3.30	8.72	5.0012		0
	1.1687	Chien				− − −		
		Proposed PID	0	5.77	10.00	5.1016		0.39
		IMC		0	5.77	10.00	1.0000	0

while the proposed method yields

$$K_{PID} = 1.4005 + \frac{1.2050}{s} + \frac{0.1856s}{\frac{0.1856}{N}s + 1},$$

which achieves the specified approximation accuracy $ERR \leq 3\%$. The Ziegler–Nichols step response tuning method (Ziegler and Nichols, 1942) gives

$$K_{PI-ZN} = 2.64(1 + \frac{1}{s} + \frac{0.25s}{\frac{0.25}{N}s + 1}),$$

while the Cohen–Coon step response method (Cohen and Coon, 1953) produces

$$K_{PI-CC} = 3.22(1 + \frac{0.9430}{s} + \frac{0.1703s}{\frac{0.1703}{N}s + 1}).$$

The closed-loop responses for different designs are shown in Figure 9.5. It

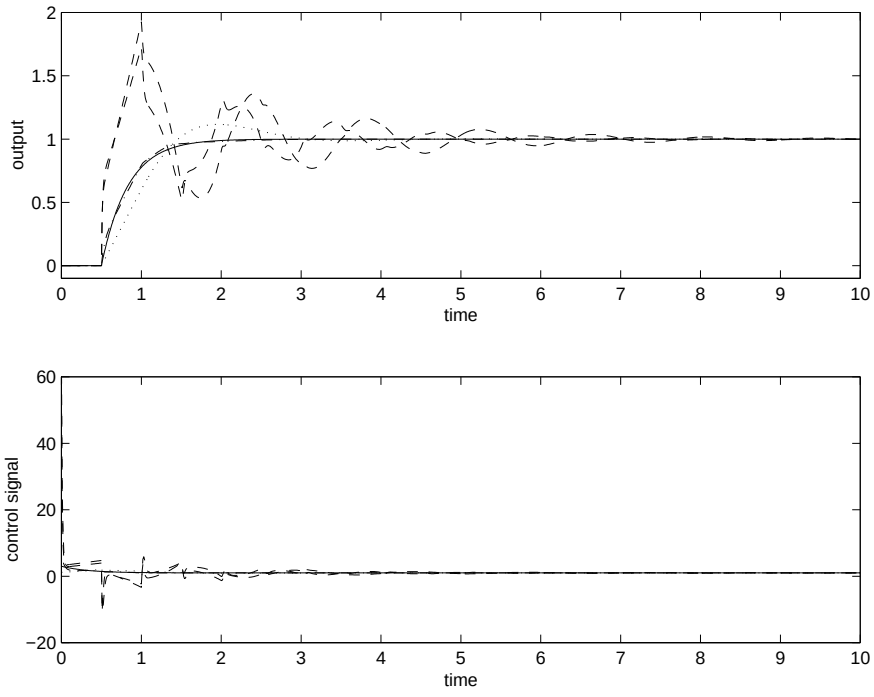


Fig. 9.5. Comparison of set-point responses for $\frac{e^{-0.5s}}{s+1}$
(– · – · – proposed PID, · · · Chien, —— IMC, – – – C-C and Z-N)

can be seen that both Chien's rule and the proposed method show much better performance than the conventional ZN and CC designs. The proposed method is almost identical to the IMC system.

Example 9.2.2. Consider a process with right half-plane zero:

$$G = \frac{(-0.5s+1)e^{-s}}{(s+1)(2s+1)}.$$

From (9.9), (9.12) and (9.13), one gets $\tau^* = 0.6687$, which gives rise to

$$\tilde{K}_{PID} = 1.3779(1+\frac{0.3111}{s}+\frac{0.8365s}{\frac{0.8365}{N}s+1})$$

by Chien's formula, and

$$K_{PID} = 1.1194+\frac{0.3569}{s}+\frac{0.9765s}{\frac{0.9765}{N}s+1}$$

by the proposed method with $ERR \leq 3\%$. The closed-loop responses are shown in Figure 9.6.

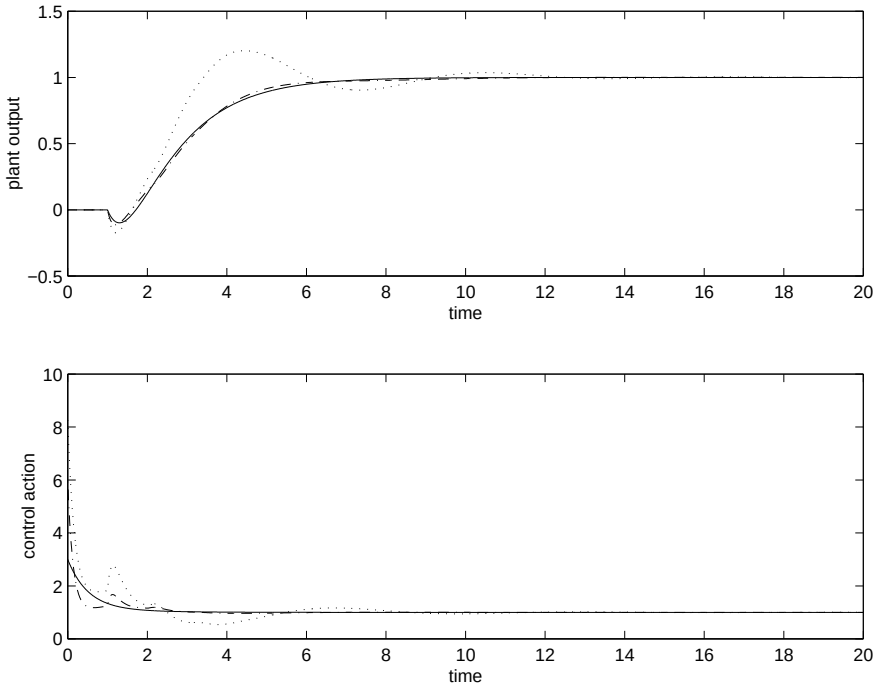


Fig. 9.6. Comparison of set-point responses for $\frac{(-0.5s+1)e^{-s}}{(s+1)(2s+1)}$
(– · – · – proposed PID, · · · Chien, —— IMC)

Example 9.2.3. Consider a high-order and oscillatory process:

$$G = \frac{e^{-2s}}{(s^2+s+1)(s^2+0.6s+1)}.$$

Our method produces $\tau^* = 0.6687$ and

$$K_{PID} = 0.2860 + \frac{0.2139}{s} + \frac{0.3962s}{\frac{0.3962}{N}s+1}. \tag{9.20}$$

This controller has the approximation error $ERR = 17.46\%$, which cannot fulfil the accuracy threshold, and the closed-loop response is very poor, as shown in Figure 9.7. Then τ is adjusted to $\tau^1 = \tau^0 + 0.25L = 0.6687 + 0.5 = 1.1687$

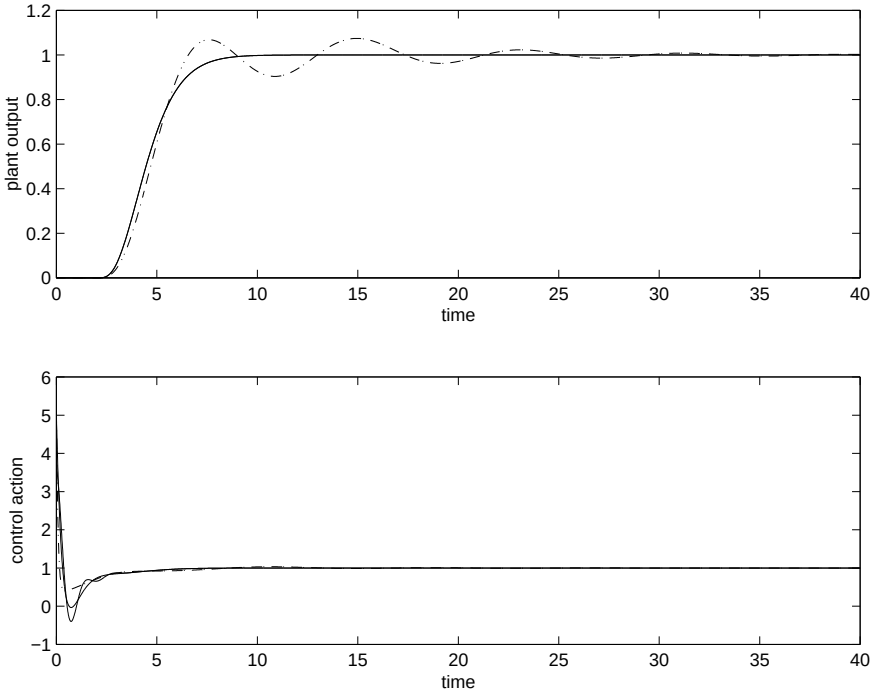


Fig. 9.7. Set-point responses for $\frac{e^{-2s}}{(s^2+s+1)(s^2+0.6s+1)}$ with $\tau = 0.6687$ (– · – · – proposed PID, – – – high-order controller, —— IMC)

according to the proposed tuning rule (9.17). The new τ results in

$$K_{PID} = 0.1016 + \frac{0.1498}{s} + \frac{0.1229s}{\frac{0.1229}{N}s+1}. \tag{9.21}$$

The approximation error ERR of the proposed method has met the specified approximation accuracy $ERR \leq 3\%$. The closed-loop responses are shown in

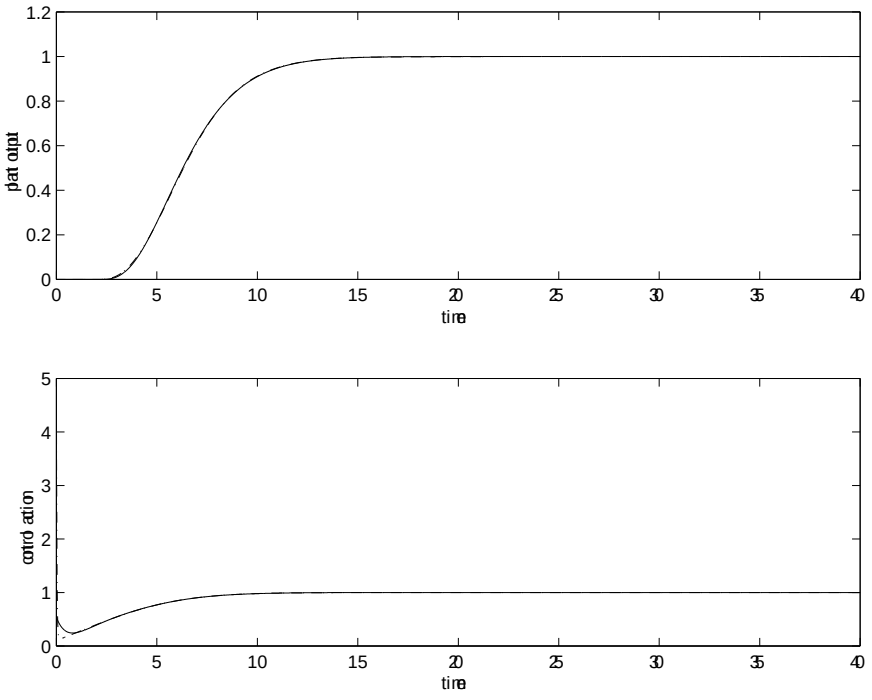


Fig. 9.8. Set-point responses for $\frac{e^{-2s}}{(s^2+s+1)(s^2+0.6s+1)}$ with $\tau = 1.1687$ (– · – · – proposed PID, —— IMC)

Figure 9.8. One observes that the difference between our SL system and the original IMC system is not discernible.

It can be seen from the simulation study in Table 9.2 that the proposed method always yields a PID controller that is a much better approximation to the IMC counterpart than Chien's method, regardless of what τ is chosen. Our experience indicates that for FOPDT and SOPDT processes and a slow closed-loop response requirement of $\tau > \frac{\sqrt{\sqrt[r]{2}-1}}{\omega_{pb}}$, both the proposed IMC-PID method and Chien's rules generate responses close to the IMC counterpart. In particular, the proposed method can always achieve $ERR < 3\%$, and thus the closed-loop performance can be well predicted from the corresponding IMC system. However, when fast closed-loop response, generally $\omega_{cb} > \omega_{pb}$, i.e., $\tau < \frac{\sqrt{\sqrt[r]{2}-1}}{\omega_{pb}}$, is required, the proposed method shows significant improvement over Chien's rules. The improvement is also evident for complex processes with slow responses. Moreover, under the fast response requirement, $\omega_{cb} > \omega_{pb}$, the PID controller derived from Chien's rules may cause large peaks in the manipulated

variable, which is harmful to the system. It is, however, noticed that for high-order processes with fast responses, none of the above two IMC-PID methods is able to generate PID systems similar to IMC ones. This implies that a controller in the PID form is insufficient to obtain the desired performance. In this case, a higher-order controller has to be considered for better fitting and performance, which will be discussed in the next section.

9.3 High-order Controller

If the single-loop controller is not limited to PID type and if PID is not adequate to control a given process, we have to consider a general type of proper rational function controller to meet the specifications. The task at hand is then to find an nth-order rational function approximation:

$$\hat{K} = \frac{b_n s^n + b_{n-1} s^{n-1} + ... + b_1 s + b_0}{s^n + a_{n-1} s^{n-1} + ... + a_1 s}, \tag{9.22}$$

with an integrator such that

$$J \triangleq \sum_{i=1}^{M} |W(j\omega_i)(\hat{K}(j\omega_i) - K(j\omega_i))|^2 \tag{9.23}$$

is minimized. The problem can be solved by one of two algorithms for transfer function modelling from frequency response presented in Chapter 7. If the recursive least squares methods (RLS) there is adopted, like the LS algorithm in the preceding section, the frequency range for the RLS is also chosen as $(0.1\omega_{cb},\ \omega_{cb})$ with steps of $(\frac{1}{100} \sim \frac{1}{10})\omega_{cb}$.

From the typical relationship of the relative fitting error ERR defined in (9.16) and the rational approximation order n shown in Figure 9.9, we can see that ERR decreases, as n increases. We try to find the minimum n which achieves the approximation bound $ERR < \epsilon$ with a user-specified τ. In general, if faster response is required, a higher-order controller has to be used.

The above algorithm deals with the problem of approximating a given, probably non-rational, transfer function by a rational function. Error bounds for such an approximation have been investigated (Wahlberg and Ljung, 1992; Yan and Lam, 1999). Wahlberg and Ljung (1992) proposed an approach based on weighted least squares estimation, and provided hard frequency-domain transfer function error bounds. However, it is not easy to calculate such a bound, and the convergence of estimation has not been addressed. In our work, we use a maximum likelihood index ERR to evaluate the approximation accuracy,

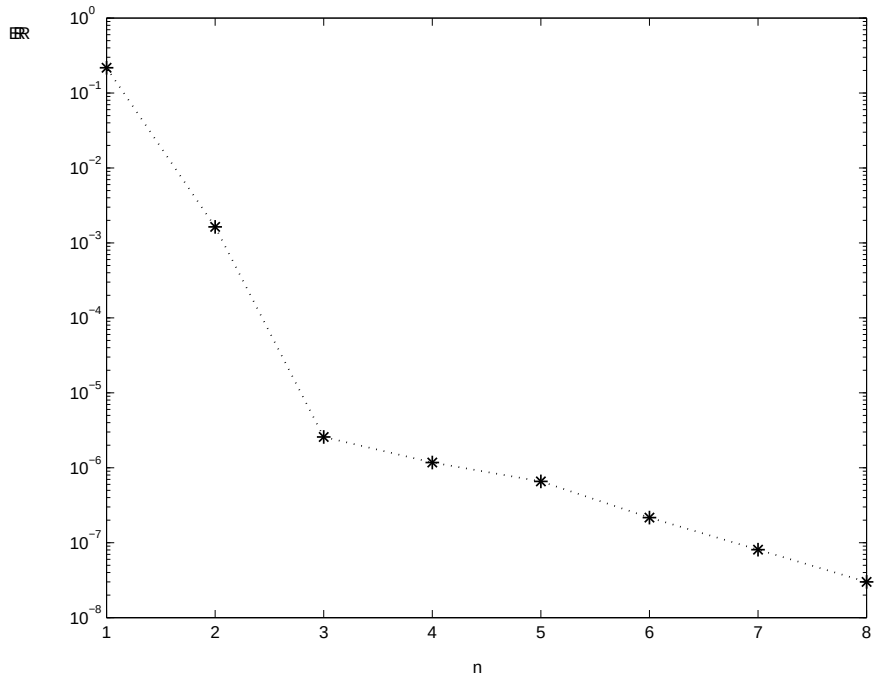


Fig. 9.9. Relationship between order n and approximation error ERR

and assume that the accuracy threshold can be achieved when the controller order is high enough.

When τ is chosen, we first find the PID controller using the standard least squares method and evaluate the corresponding approximation error ERR in (9.16) as described in the preceding section. If ERR cannot achieve the specified approximation accuracy ϵ (usually 3%), we recommend a high-order controller as in (9.22), and start with a controller of order 2 up to the smallest integer n such that $ERR \leq \epsilon$.

Tuning procedure

Step 1. Find the smallest τ^* from (9.9), (9.12) and (9.13), and let $\tau^0 = \tau^*$.

Step 2. Determine the PID controller from (9.15) and evaluate the corresponding approximation error ERR in (9.16). If ERR achieves the specified approximation accuracy ϵ (usually 3%), end the design.

Step 3. Otherwise, we have two ways to solve this problem: if a PID controller is desired, update τ by (9.17), and go to Step 2; else, go to Step 4.

Step 4. Adopt the high-order controller in (9.22), start with a controller of order 2 up to the smallest integer n for which $ERR \leq \epsilon$.

Example 9.2.3 (cont'd). Reconsider

$$G = \frac{e^{-2s}}{(s^2+s+1)(s^2+0.6s+1)},$$

for which $\tau^0 = 0.6687$ and a PID has been obtained with $ERR = 17.46\%$. For a high-order controller, our procedure ends with

$$\hat{K} = \frac{3.5488s^5 + 14.9135s^4 + 23.9669s^3 + 29.6333s^2 + 18.2712s + 9.2024}{s^5 + 4.5140s^4 + 25.8787s^3 + 22.8686s^2 + 43.0211s}, \tag{9.24}$$

with fitting error ERR less than $\epsilon = 3\%$. The closed-loop step responses are shown in Figure 9.7, and their performance indices are also given in Table 9.2. We can see that the new controller $\hat{K}$ restores the IMC performance, while the previous PID controller in (9.20) is not capable of that under such a tight performance specification.

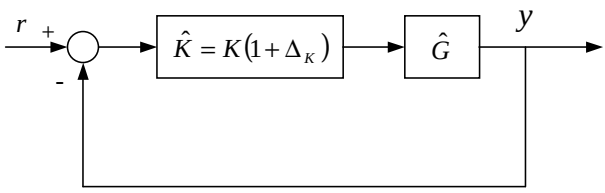
If τ is chosen to be smaller than the value suggested by (9.9), (9.12) and (9.13), this overcomes the limitation of single-loop feedback systems, but then no single-loop controller solution with stability could be found for the corresponding IMC system. For instance, in the above example, choosing τ 50% less than τ^0, i.e., $\tau = 0.3344$, we could not find a controller in the form of (9.22) with ERR less than 3%, which implies that SL controllers are unlikely to achieve a performance tighter than that specified by (9.9), (9.12) and (9.13).

It is observed from our simulation study that usually, the approximation error magnitude of the high-order controller obtained by the RLS is of the order 10^{-4} or less, the controller order is less than 6, and the controller yields a closed-loop response very close to that of the IMC loop provided that τ is set by (9.9), (9.12) and (9.13). The high-order controller does provide significant performance enhancement over PID for complex processes. The proposed method is a simple, effective, and efficient way to design such high-performance controllers.

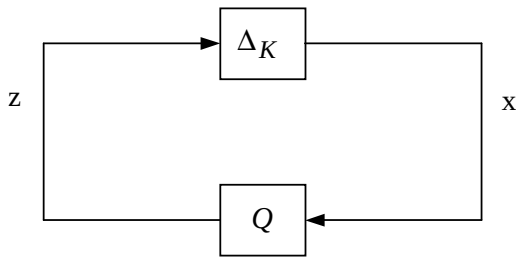
9.4 Stability Analysis

From the results obtained so far, it is possible to state that the single-loop system with $\hat{K}$ derived using the proposed method has a performance close to the corresponding IMC loop. Thus the stability of the resulting single-loop control system is well related to that of the IMC system. In this section, we consider both nominal stability ($G = \hat{G}$) and robust stability ($G \neq \hat{G}$).

Assume $G = \hat{G}$ in the absence of model uncertainty, the nominal stability of the IMC system automatically guarantees the stability of the feedback system



(a)



(b)

Fig. 9.10. Block diagram of nominal system

in Figure 9.2 with K determined from (9.11). However, the proposed design makes a controller approximation $\hat{K}$ and thus gives the nominal single-loop system shown in Figure 9.10(a), where $\hat{K}(s) = K(s)(1 + \Delta_K(s))$. The system in Figure 9.10(a) can be redrawn as Figure 9.10(b), where

$$Q = -\frac{\hat{G}K}{1 + \hat{G}K}.$$

Using (9.11), $Q(s)$ can be written as $Q = -\hat{G}C$ and is stable. With the standard assumption that $\hat{K}$ has the same number of unstable poles as K, the nominal single-loop feedback system is stable (Green and Limebeer, 1995) if and only if

$$|\hat{G}(j\omega)C(j\omega)\Delta_K(j\omega)| < 1, \quad \forall\omega. \tag{9.25}$$

It follows from (9.1)–(9.3) that $\hat{G}C = \hat{G}_+ f$. But $|\hat{G}_+(j\omega)| = 1$, $\forall\omega$ and $|\hat{G}_+ f\Delta_K| = |f\Delta_K|$, (9.25) then becomes

$$|f(j\omega)\Delta_K(j\omega)| < 1, \quad \forall\omega. \tag{9.26}$$

Note from (9.4) that $|f(j\omega)|$ decays quickly for $\omega \geq \omega_{cb} = \frac{\sqrt{\sqrt[m]{2}-1}}{\tau}$ and (9.26) is likely to hold for high frequencies. Thus, assume that (9.26) is true for $\omega \geq \omega_{cb}$. One now needs to check (9.26) only for the working frequency range $[0,\ \omega_{cb}]$, and because $|f(j\omega)| \leq 1$ for all ω, the nominal closed loop is thus stable if

$$|\Delta_K| = |\frac{K(j\omega) - \hat{K}(j\omega)}{K(j\omega)}| < 1, \quad \omega \in [0,\ \omega_{cb}]. \tag{9.27}$$

In the proposed algorithm, the approximation accuracy has to meet (9.16), where ϵ is usually specified as 3%. The resulting controller $\hat{K}$ then satisfies (9.27) with a large margin and nominal stability of the designed single-loop system is thus expected.

Consider now model uncertainty. Let the actual plant be $G(s) = \hat{G}(s)(1 + \Delta_G(s))$. In the IMC design (Morari and Zafiriou, 1989) to achieve robust stability, the filter parameter τ is chosen big enough to meet the condition:

$$|\hat{G}(j\omega)C(j\omega)\Delta_G(j\omega)| < 1, \quad \text{or} \quad |f(j\omega)\Delta_G(j\omega)| < 1, \quad \forall\omega. \tag{9.28}$$

The single-loop system with process uncertainty is shown in Figure 9.11(a), where $|\Delta_K| \leq \delta_K(\omega)$ and $|\Delta_G| \leq \delta_G(\omega)$, and it can be redrawn into the standard form in Figure 9.11(b), where $\tilde{\Delta}(s)$ is the normalized uncertainty $\tilde{\Delta} = \text{diag}\{\tilde{\Delta}_K, \tilde{\Delta}_G\}$ with $|\tilde{\Delta}_K| \leq 1$ and $|\tilde{\Delta}_G| \leq 1$. The transfer function matrix between z and x has no uncertainty and is given by

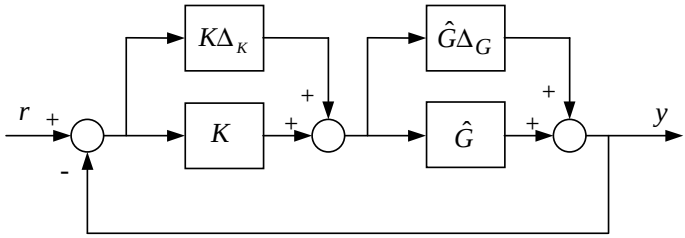
$$\begin{aligned} Q &= \begin{bmatrix} \delta_K & 0 \\ 0 & \delta_G \end{bmatrix} \begin{bmatrix} -\hat{G}K & -K \\ \hat{G} & -\hat{G}K \end{bmatrix} (1 + \hat{G}K)^{-1} \\ &= \begin{bmatrix} \delta_K & 0 \\ 0 & \delta_G \end{bmatrix} \begin{bmatrix} -\hat{G}C & -C \\ \hat{G}(1 - \hat{G}C) & -\hat{G}C \end{bmatrix}, \end{aligned}$$

whose stability is guaranteed by that of $\hat{G}$ and C. It follows from the stability robustness theorem (Doyle *et al.*, 1982) that the uncertain feedback system remains stable for all $\tilde{\Delta} = \text{diag}\{\tilde{\Delta}_K, \tilde{\Delta}_G\}$ if and only if

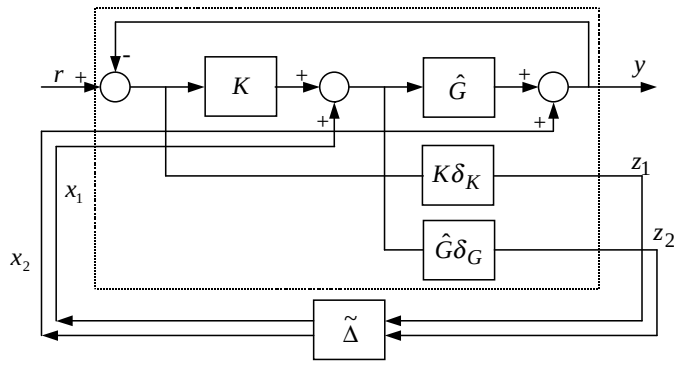
$$\|Q\|_\mu < 1, \tag{9.29}$$

where $\|Q\|_\mu = \sup_\omega \mu(Q(j\omega))$ and $\mu(\cdot)$ is the structured singular value with respect to $\tilde{\Delta}$. In our case, the structured singular value $\mu(Q(j\omega))$ can be calculated by

$$\mu(Q(j\omega)) = \mu(DQD^{-1}) = \inf_D \bar{\sigma}(DQD^{-1}),$$



(a)



(b)

Fig. 9.11. Block diagram of system with process uncertainty

where $D = \text{diag}\{d_1, d_2\}$, $d_1, d_2 > 0$ and $\bar{\sigma}(\cdot)$ represents the largest singular value.

After some calculations, we obtain

$$\mu(Q(j\omega)) = |GC| \cdot \cdot \sqrt{\frac{\delta_K^2 + \delta_G^2 + 2\delta_K\delta_G|\frac{1-\hat{G}C}{\hat{G}C}| + \sqrt{(\delta_K^2 + \delta_G^2 + 2\delta_K\delta_G|\frac{1-\hat{G}C}{\hat{G}C}|)^2 - 4\delta_K^2\delta_G^2|\frac{1}{\hat{G}C}|^2}}{2}}, \tag{9.30}$$

and the robust stability condition (9.29) becomes

$$\delta_K^2|\hat{G}C|^2 + \delta_G^2|\hat{G}C|^2 + 2\delta_K\delta_G|(1-\hat{G}C)\hat{G}C| + \\ +\sqrt{(\delta_K^2|\hat{G}C|^2 + \delta_G^2|GC|^2 + 2\delta_K\delta_G|(1-\hat{G}C)\hat{G}C|)^2 - 4\delta_K^2\delta_G^2|\hat{G}C|^2} \\ < 2, \quad \forall\omega. \tag{9.31}$$

Since $4\delta_K^2\delta_G^2|\hat{G}C|^2 \geq 0$, $\forall\omega$, and $|1-GC| \leq 1+|GC| \leq 2$, (9.31) is satisfied if

$$\delta_K^2|\hat{G}C|^2+\delta_G^2|\hat{G}C|^2+4\delta_K\delta_G|\hat{G}C| < 1, \quad \forall\omega, \tag{9.32}$$

i.e.,

$$\delta_K^2(\omega)|f(j\omega)|^2+\delta_G^2(\omega)|f(j\omega)|^2+4\delta_K(\omega)\delta_G(\omega)\,|f(j\omega)| < 1, \quad \forall\omega.$$

As $|f(j\omega)|$ decays quickly for $\omega \geq \omega_{cb} = \frac{\sqrt{\sqrt[n]{2}-1}}{\tau}$, then (9.32) is likely to hold for high frequencies. Thus, assume that (9.32) is true for $\omega \geq \omega_{cb}$. One now needs to check (9.32) only for the working frequency range $[0,\ \omega_{cb}]$, and because $|f(j\omega)| \leq 1$ for all ω, the closed loop is robustly stable if

$$\delta_K^2(\omega)+\delta_G^2(\omega)+4\delta_K(\omega)\delta_G(\omega) < 1, \quad \omega \in [0,\ \omega_{cb}].$$

In the proposed method, $|\Delta_K|$ is made small, i.e., $\delta_K(\omega) \leq 3\%$. Let $\delta_K = 3\%$, then the robust stability of the closed loop is guaranteed by

$$\delta_G(\omega) \leq 94.13, \quad \omega \in [0,\ \omega_{cb}]. \tag{9.33}$$

Note that for $\delta_K = 0$, i.e., no controller uncertainty, (9.30) reduces to $\mu(Q(j\omega)) = |\hat{G}C\delta_G|$. Then the robust stability condition (9.29) is simplified to $\sup_\omega|\hat{G}C\delta_G| < 1$, which is equivalent to the robust stability condition (9.28) of the IMC system.

Example 9.2.3 (cont'd). Reconsider

$$G = \frac{\alpha e^{-2s}}{(s^2+s+1)(s^2+0.6s+1)},$$

with nominal $\alpha = \alpha_0 = 1$. When $\tau = 0.6687$, the proposed method yields a 5th-order controller in (9.24) and the nominal performance is shown in Figure 9.7. It can be seen that the system is indeed nominally stable. To demonstrate robustness, introduce a 50% perturbation from the nominal gain of α, giving $\alpha = 1.5$. Figure 9.12 shows the resulting performances, and indicates that the single-loop high-order controller K derived using the proposed method exhibits a similar robust performance to the IMC loop.

When $\tau = 1.1687$, the proposed method yields a PID controller as in (9.21) and the nominal performance is shown in Figure 9.8. We also introduce a 50% perturbation in gain, giving $\alpha = 1.5$. Figure 9.13 shows the resulting performance, which is still stable, and more robust than that shown in Figure 9.12 for $\tau = 0.6687$.

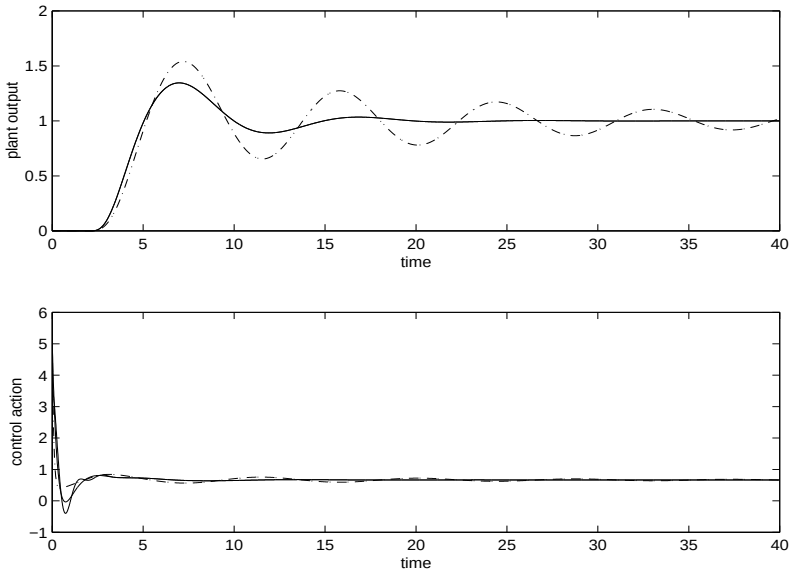


Fig. 9.12. Performance robustness
(– · – · – proposed PID, – – – high-order controller, — IMC)

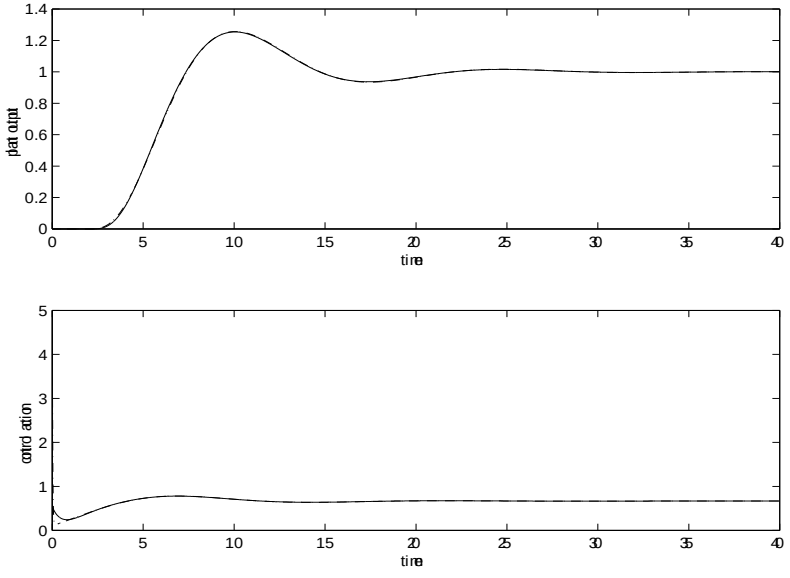


Fig. 9.13. Performances robustness
(– · – · – proposed PID, — IMC)

9.5 Unstable Processes

It is well known that the IMC scheme is internally unstable if the process $G(s)$ is unstable (Morari and Zafiriou, 1989) and is thus not implementable. However, Morari and Zafiriou (1989) suggest that one can still design the controller using the IMC method, and then implement the controller in an equivalent feedback structure as follows. In the nominal case (the uncertain case will be discussed below) of $G(s) = \hat{G}(s)$, the equivalent single-loop feedback system of Figure 9.2 is derived from the IMC system of Figure 9.1 with

$$K = C(1 - \hat{G}C)^{-1}. \tag{9.34}$$

The system is internally stable if

$$C \text{ and } (1 - \hat{G}C)\hat{G} \text{ are both analytic in the RHP.} \tag{9.35}$$

Generally, the feedback controller K in (9.34) should include an integrator to eliminate the steady-state error, and maintain stability. Thus we require

$$\begin{aligned} &K(\text{after all possible pole–zero cancellations}) \\ &\text{has no pole in the closed RHP except } s = 0. \end{aligned} \tag{9.36}$$

Let us consider a class of unstable processes with a single RHP pole only:

$$\hat{G}(s) = \frac{1}{1 - Ts}\hat{G}_-(s)e^{-Ls}, \tag{9.37}$$

where $\hat{G}_-(s)$ is rational, stable and of minimum phase. The optimal H_2 IMC controller for step inputs is

$$C = (1 - Ts)\hat{G}_-^{-1} f, \tag{9.38}$$

where f is a user-specified low-pass filter and chosen as

$$f(s, \tau) = \frac{\alpha s + 1}{(\tau s + 1)^{m+1}}, \quad \alpha = T(\tau/T + 1)^{m+1} e^{L/T} - T, \tag{9.39}$$

where m is an integer large enough to guarantee that the IMC controller C is proper. One can easily verify that (9.35) holds for all $\tau > 0$. Thus, $\tau > 0$ is the only tuning parameter to be selected by the user to meet (9.36) and achieve the appropriate trade-off between performance and robustness, and to keep the manipulated variable within bounds. We now address them separately.

For $\hat{G}$ as in (9.37) and C as in (9.38), K in (9.34) becomes

$$K(s, \tau) = \frac{(1 - Ts)(\alpha s + 1)}{\hat{G}_-((\tau s + 1)^{m+1} - (\alpha s + 1)e^{-Ls})}. \tag{9.40}$$

It can be verified that $s = \frac{1}{T}$ is both a zero and pole of $K(s)$. It should be cancelled to form the final $K(s)$ for actual implementation. $s = 0$ is another pole of $K(s)$, which is necessary to eliminate the steady-state error. Equation (9.36) requires that no roots of the denominator of $K(s,\tau)$ lie in the closed RHP, expect $s = \frac{1}{T}$, and $s = 0$. Since $\hat{G}_-$ is of minimum phase, from (9.3) we only need to investigate the root locations of

$$\delta(s,\tau,L) = (\tau s+1)^{m+1} - (\alpha s+1)e^{-Ls}. \tag{9.41}$$

Normalize L, τ, a, s as $\bar{L} = L/T$, $\bar{\tau} = \tau/T$, $\bar{\alpha} = \alpha/T$ and $\bar{s} = sT$, Equation (9.41) then becomes

$$\delta(\bar{s},\bar{\tau},\bar{L}) = (\bar{\tau}\bar{s}+1)^{m+1} - (\bar{\alpha}\bar{s}+1)e^{-\bar{L}\bar{s}}. \tag{9.42}$$

Equation (9.42) is a quasi-polynomial and can be written into a standard form,

$$Q(\bar{s},\bar{L}) = A(\bar{s}) + B(\bar{s})e^{-\bar{L}\bar{s}}, \tag{9.43}$$

where $A(\bar{s})$ and $B(\bar{s})$ are polynomials in $\bar{s}$. Walton and Marshall (1987) proposed a method to study the movement of the roots of (9.43) with respect to a given parameter and this can be employed to determine the minimum $\bar{\tau}$ at which roots of (9.43) lie on the imaginary axis. This $\bar{\tau}_{min}$ should meet

$$\cos(\omega_0\bar{L}) = \mathrm{Re}\,\{\frac{(j\omega_0\bar{\tau}_{min}+1)^{m+1}}{j\omega_0\bar{\alpha}+1}\}, \tag{9.44}$$

$$\sin(\omega_0\bar{L}) = \mathrm{Im}\,\{-\frac{(j\omega_0\bar{\tau}_{min}+1)^{m+1}}{j\omega_0\bar{\alpha}+1}\}, \tag{9.45}$$

where

$$\omega_0 = \{\min(\omega_0)|(\omega_0^2\bar{\tau}_{min}^2+1)^{m+1} - (\omega_0^2\bar{\alpha}^2+1) = 0,\ \omega_0 > 0\}. \tag{9.46}$$

Thus, for a given $\bar{L}$, $\bar{\tau}$ should be chosen to satisfy

$$\bar{\tau} > \bar{\tau}_{min}, \tag{9.47}$$

to ensure stability. For a given m, $\bar{\tau}_{min}$ depends on $\bar{L}$. When $m = 1,2,3$, the typical relationship between $\bar{\tau}_{min}$ and $\bar{L}$ is shown in Figure 9.14. It is interesting to note that when $\bar{\tau}_{min}$ tends to infinity, $\bar{L}$ tends to a constant, $\bar{L}_{max}$, which indicates that there is a limitation on tuning τ for stabilizability and that the process is stabilizable only if $\bar{L} \le \bar{L}_{max}$. $\bar{L}_{max}$ is determined by m only and can be obtained as

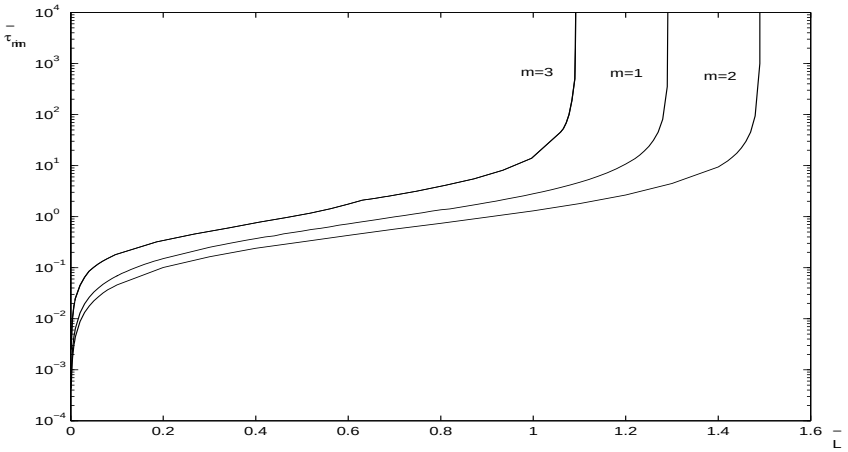


Fig. 9.14. Typical relationship between $\bar{\tau}_{min}$ and $\bar{L}$

$$\bar{L}_{max} = \begin{cases} \{\max(\bar{L}) \mid e^{\bar{L}}\bar{L}^m < (\frac{3\pi}{2})^m\}, m = 4l+1 \\ \{\max(\bar{L}) \mid e^{\bar{L}}\bar{L}^m < (\pi)^m\}, m = 4l+2 \\ \{\max(\bar{L}) \mid e^{\bar{L}}\bar{L}^m < (\frac{\pi}{2})^m\}, m = 4l+3 \\ \{\max(\bar{L}) \mid e^{\bar{L}}\bar{L}^m < (2\pi)^m\}, m = 4l+4 \end{cases} \tag{9.48}$$

where $l = 0, 1, 2 \cdots$.

For system performance, with controller $K(s)$ in (9.40), the closed-loop transfer function is

$$\eta(s) = \frac{G(s)K(s)}{1 + G(s)K(s)} = \hat{G}_+(s)f(s) = \frac{\alpha s + 1}{(\tau s + 1)^{m+1}} e^{-Ls}. \tag{9.49}$$

The maximum of the magnitude of η is related directly to $\bar{L}$ and $\bar{\tau}$ as

$$\|\eta\|_\infty = \sqrt{\frac{\frac{1}{m}((\frac{\bar{\alpha}}{\bar{\tau}})^2 - m - 1) + 1}{(\frac{1}{m}(1 - (m+1)(\frac{\bar{\tau}}{\bar{\alpha}})^2) + 1)^{m+1}}}. \tag{9.50}$$

A typical relation between $\|\eta\|_\infty$ and $\bar{\tau}$ is shown in Figure 9.15. The large amplitude of $\|\eta\|_\infty$ usually produces a peak overshoot in the step response in the time domain perspective (Kuo, 1991). To eliminate the overshoot, a prefilter $F = \frac{1}{\alpha s+1}$ is added with α given in (9.39), as shown in Figure 9.16.

For system robustness, let the actual process be $G(s) = \hat{G}(s)(1 + \Delta_G(s))$ with $|\Delta_G| \leq \delta_G(\omega)$. In implementation, the presence of dead time in the denominator of $K(s)$ increases the complexity of the controller. Moreover, due to the fact that the denominator of $K(s)$ is not in polynomial form, it is not possible to cancel $s = \frac{1}{T}$ in $K(s)$ explicitly. Thus, model reduction is applied

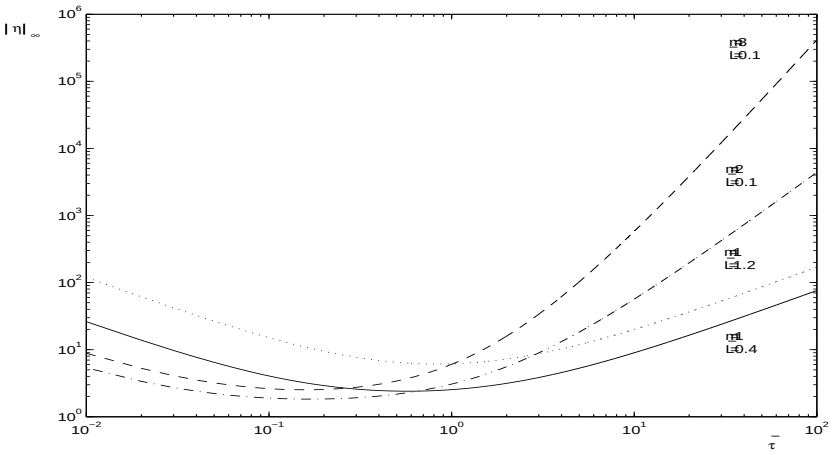


Fig. 9.15. Relation between $\|\eta\|_\infty$ and $\bar{\tau}$

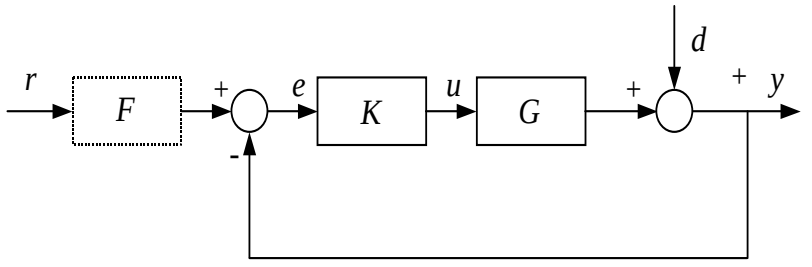


Fig. 9.16. Single-loop control system with pre-filter

to obtain the best approximation $\hat{K}$ to K, where $\hat{K}(s) = K(s)(1 + \Delta_K(s))$ and $|\Delta_K| \leq \delta_K(\omega)$. In our context, $\delta_K(\omega) \leq ERR \leq \epsilon$, where ϵ is the threshold for model reduction error and is specified prior to the design and usually taken as 5%. With the standard assumption that $\hat{K}$ has the same number of unstable poles as K, it follows from the stability robustness theorem (Doyle *et al.*, 1982) and some algebra that the uncertain feedback system remains stable for all $\Delta = \text{diag}\{\Delta_K, \Delta_G\}$ if

$$\delta_K^2(\omega)|\eta(j\omega)|^2 + \delta_G^2(\omega)|\eta(j\omega)|^2 + 4\delta_K(\omega)\delta_G(\omega)\,|\eta(j\omega)| < 1, \quad \forall\omega. \tag{9.51}$$

It is noted from (9.49) that $|\eta(j\omega)|$ has a peak value at a low frequency ω_p and decays quickly for higher frequencies. Then (9.51) is likely to hold if

$$\delta_K^2(\omega_p)||\eta||_\infty^2 + \delta_G^2(\omega_p)||\eta||_\infty^2 + 4\delta_K(\omega_p)\delta_G(\omega_p)\,||\eta||_\infty < 1. \tag{9.52}$$

An upper bound on $||\eta||_\infty$ can be obtained from (9.52):

$$||\eta||_\infty < \left.\frac{2\sqrt{\delta_K^2(\omega)+\delta_G^2(\omega)+4\delta_K^2(\omega)\delta_G^2(\omega)}-4\delta_K(\omega)\delta_G(\omega)}{2(\delta_K^2(\omega)+\delta_G^2(\omega))}\right|_{\omega=\omega_p}. \tag{9.53}$$

Combining (9.50) and (9.53) yields

$$\sqrt{\frac{\frac{1}{m}((\frac{\bar{\alpha}}{\bar{\tau}})^2-m-1)+1}{(\frac{1}{m}(1-(m+1)(\frac{\bar{\tau}}{\bar{\alpha}})^2)+1)^{m+1}}}$$

$$< \left.\frac{2\sqrt{\delta_K^2(\omega)+\delta_G^2(\omega)+4\delta_K^2(\omega)\delta_G^2(\omega)}-4\delta_K(\omega)\delta_G(\omega)}{2(\delta_K^2(\omega)+\delta_G^2(\omega))}\right|_{\omega=\omega_p} \triangleq \delta_p, \tag{9.54}$$

which gives a range of τ to achieve robust stability.

To consider the performance limitation imposed by input constraints, we still use frequency-by-frequency analysis. Assume that at each frequency $|U(j\omega)| \le \bar{U}$ and the reference signal satisfies $|R(j\omega)| \le \bar{R}$. The manipulated variable is

$$\begin{aligned} U(s) &= F(s)\frac{K(s,\tau)}{1+G(s)K(s,\tau)}R(s) \\ &= \frac{1-Ts}{(\tau s+1)^{m+1}}\hat{G}_-^{-1}(s)R(s). \end{aligned}$$

It is required that

$$\frac{1-Tj\omega}{(\tau j\omega+1)^{m+1}}\hat{G}_-^{-1}(j\omega)R(j\omega) \le \bar{U}. \tag{9.55}$$

Consider the worst case $|R(j\omega)| = \bar{R}$, which requires

$$\left|\frac{1}{(\tau j\omega+1)^{m+1}}\right| \le \left|\frac{1}{1-Tj\omega}\hat{G}_-(j\omega)\right|\frac{\bar{U}}{\bar{R}}. \tag{9.56}$$

To derive an inequality on τ imposed by the input constraint, let $\omega = \omega_{ob}$, where ω_{ob} is the open-loop bandwidth, and notice that $\left|\frac{1}{1-Tj\omega_{ob}}\hat{G}_-(j\omega_{ob})\right| = \frac{1}{\sqrt{2}}$, we require

$$\left|\frac{1}{(\tau j\omega_{ob}+1)^{m+1}}\right| \le \frac{1}{\sqrt{2}}\frac{\bar{U}}{\bar{R}}, \tag{9.57}$$

i.e.,

$$\tau \ge \sqrt{\sqrt[m+1]{\frac{2\bar{R}^2}{\bar{U}^2}}-1}\Big/\omega_{ob}. \tag{9.58}$$

Therefore, for open-loop unstable process controller design the tuning parameter τ in the filter (9.39) should be, in general, chosen to meet (9.47), (9.54) and (9.58) simultaneously and this will determine a suitable range, $\tau \in (\tau_{min}, \tau_{max})$.

Once the ideal single-loop controller $K(s)$ has been found, model reduction is again applied to obtain its PID or high-order controller approximation. The procedure is similar to the stable case discussed earlier and we only highlight possible differences and present simulation examples as follows.

9.5.1 PID Controller

Consider a PID controller in the form (9.14). Following the same steps as in Section 9.2, the optimal PID controller K_{PID} can be obtained. If the user-specified fitting error threshold in (9.16) holds true, the design is complete. On the other hand, one can increase τ by the de-tuning rule in (9.17). The iteration continues until the accuracy bound is fulfilled or $\tau^{k+1} > \tau_{max}$. If (9.16) cannot be fulfilled when $\tau^{k+1} > \tau_{max}$, a more complex controller than a PID is necessary. We now present some simulation examples to demonstrate our PID tuning algorithm and the performance is compared with the results of Huang and Chen (1997), Majhi and Atherton (2000), and Park *et al.* (1998). As in Section 9.2, the ideal PID controller in (9.14) is replaced by version (9.19) for implementation. In simulation examples of this subsection, we consider the nominal case and (9.54) is not used. Normally, (9.58) gives a smaller lower bound on τ and thus only (9.47) is utilized to derive τ_{min}, and we set $\tau^0 = \tau_{min}$ in Examples 9.5.1–9.5.3.

Example 9.5.1. Consider an unstable process (Huang and Chen, 1997)

$$G = \frac{4e^{-2s}}{1-4s}.$$

From (9.47) one obtains $\tau^0 = 1.7$. This results in

$$K_{PID} = 0.6407 + \frac{0.0626}{s} + \frac{0.5633s}{\frac{0.5633}{N}s + 1}. \tag{9.59}$$

This controller has the approximation error $ERR = 1.80\%$, which can meet the accuracy threshold. The PI-PD controller of Majhi and Atherton (2000) are $k_p(1+\frac{1}{T_i s}) = 0.131(1+\frac{1}{2s})$ and $K_f(T_d s+1) = 0.5(s+1)$. The PID-P controller of Park *et al.* (1998) has $K_p(1+\frac{1}{T_i s}+T_d s) = 0.068(1+\frac{1}{1.885s}+4.296s)$ and $K_f = 0.350$. The closed-loop responses are shown in Figure 9.17. It can be seen that the proposed method shows much better performance than the other designs.

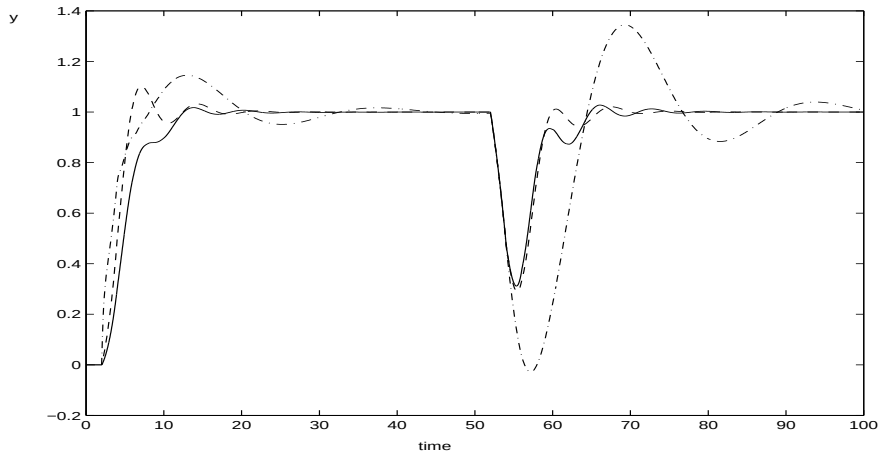


Fig. 9.17. Step response for Example 9.5.1
(—— proposed, – – – Majhi and Atherton, – · – · – Park *et al.*)

Example 9.5.2. Consider an unstable process (Park *et al.*, 1998)

$$G = \frac{e^{-0.5s}}{(1-2s)(0.5s+1)}.$$

The proposed design gives $\tau^0 = 0.6$ and

$$K_{PID} = 3.1308 + \frac{0.7452}{s} + \frac{1.5651s}{\frac{1.5651}{N}s+1}, \tag{9.60}$$

with $ERR = 0.47\%$. The PI-PD controller of Majhi and Atherton (2000) are $0.937(1+\frac{1}{1.339s})$ and $2.328(0.53s+1)$. The PID-P controller of Park *et al.* (1998) are $0.561(1+\frac{1}{1.165s}+1.478s)$ and $K_f = 1.687$. The closed-loop responses are shown in Figure 9.18.

Example 9.5.3. Consider an unstable process (Huang and Lin, 1995)

$$G = \frac{e^{-0.5s}}{(1-5s)(2s+1)(0.5s+1)}.$$

It follows that $\tau^0 = 1$ and

$$K_{PID} = 4.3794 + \frac{0.5106}{s} + \frac{7.2571s}{\frac{7.2571}{N}s+1}, \tag{9.61}$$

with $ERR = 26.07\%$, which cannot meet the accuracy threshold, and the closed-loop response is very poor. Then τ is adjusted to $\tau^1 = \tau^0 + L = 1.5$ according to the proposed tuning rule (9.17). The new τ results in

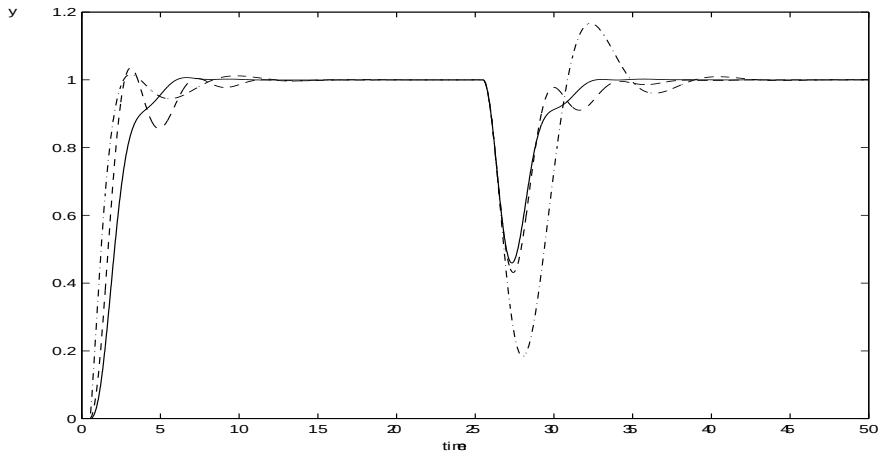


Fig. 9.18. Step response for Example 9.5.2
(—— proposed, – – – Majhi and Atherton, – · – · – Park *et al.*)

$$K_{PID} = 2.9886 + \frac{0.2335}{s} + \frac{4.6668s}{\frac{4.6668}{N}s + 1}, \tag{9.62}$$

which has $ERR = 2.06$ and meets the specified approximation accuracy $ERR \leq 5\%$. The controller in Huang and Chen (1997) is

$$K_{PID} = 6.1859(1 + \frac{0.1395}{s} + \frac{1.4724s}{\frac{1.4724}{N}s + 1}). \tag{9.63}$$

The closed-loop responses are shown in Figure 9.19. One observes that our design yields great improvement over Huang's method.

It can be seen from the simulation study that the proposed method always yields a PID controller with much better performance than the other methods, regardless of what τ is chosen. Our experience indicates that for FOPDT and SOPDT processes and a slow closed-loop response requirement, the proposed method can always achieve $ERR < 5\%$, and thus the closed-loop performance can be well predicted from the corresponding IMC design. It is, however, noticed that for high-order processes with fast responses, none of the above methods is able to generate PID systems with good performance. This implies that controllers in PID form are insufficient to obtain the desired performance. In this case, a higher-order controller has to be considered for better fitting and performance.

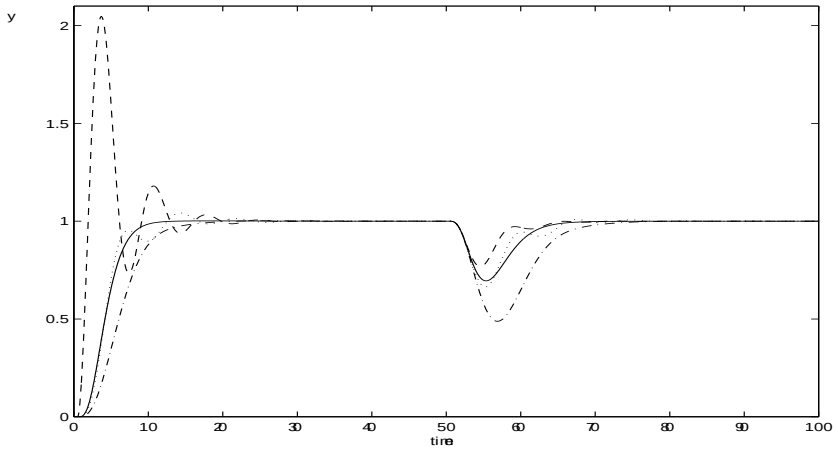


Fig. 9.19. Step response for Example 9.5.3
($\cdots\cdots$ proposed PID ($\tau = 1$), —— proposed high order ($\tau = 1$), $- - -$ Huang and Chen, $- \cdot - \cdot -$ proposed PID ($\tau = 1.5$))

9.5.2 High-order Controller

The idea here is the same as in Section 9.3, that is to find the lowest order of controller in the form of (9.22) that can match the ideal controller $K(s)$ as well as possible in the frequency range of interest and can meet a specified approximation accuracy. However, the rules for determining the IMC filter change for the present unstable processes. We thus need to summarize the tuning procedure again.

Tuning Procedure for Unstable Processes

Step 1. Find the smallest τ_{min} from (9.47) and (9.58), and let $\tau^0 = \tau_{min}$. Find the largest τ_{max} from (9.54) if the plant uncertainty δ_G is given.

Step 2. Determine the PID controller from (9.15) and evaluate the corresponding approximation error ERR in (9.16). If ERR achieves the specified approximation accuracy ϵ (usually 5%), end the design.

Step 3. Otherwise, we have two ways to solve this problem: if a PID controller is desired, update τ by (9.17), and go to Step 2 when $\tau < \tau_{max}$; else, go to Step 4.

Step 4. Adopt the high-order controller in (9.22), start with a controller of order 2 up to the smallest integer n for which $ERR \leq \epsilon$.

Example 9.5.3 (cont'd). Reconsider

$$G = \frac{e^{-0.5s}}{(1-5s)(2s+1)(0.5s+1)}.$$

for which $\tau^0 = 1$ and a PID has been obtained with $ERR = 26.07\%$. For a high-order controller, our procedure ends with

$$\hat{K} = \frac{29.8188s^3 + 75.8869s^2 + 39.4556s + 4.3412}{s^3 + 3.2042s^2 + 8.5984s}, \tag{9.64}$$

with fitting error ERR less than $\epsilon = 5\%$. The closed-loop step responses are shown in Figure 9.19. We can see that the new controller $\hat{K}$ restores the IMC performance, while the previous PID controller in (9.61) is not capable of that under such a tight performance specification.

Example 9.5.4. Consider an unstable process

$$G = \frac{e^{-1.2s}}{(1-s)(0.5s+1)}.$$

It follows that $\tau^0 = 2.7$, and

$$K_{PID} = 1.0134 + \frac{0.0063}{s} + \frac{1.0155s}{\frac{1.0155}{N}s + 1}, \tag{9.65}$$

with $ERR = 18.50\%$, which cannot meet the accuracy threshold, and the closed-loop response is very poor. One can de-tune the PID controller by increasing τ. However, this results in a sluggish response. For a high-order controller, our procedure ends with

$$\hat{K} = \frac{5.2221s^4 + 41.6265s^3 + 128.3411s^2 + 131.1146s + 0.7798}{s^4 + 14.0061s^3 + 9.6026s^2 + 123.0987s}, \tag{9.66}$$

with the fitting error ERR less than $\epsilon = 5\%$. The closed-loop step responses are shown in Figure 9.20. We can see that the high-order controller makes a significant improvement over the PID controller. To our knowledge, the PID controller design methods in the literature are not applicable to this example (Huang and Chen, 1997; Majhi and Atherton, 2000; Park *et al.*, 1998).

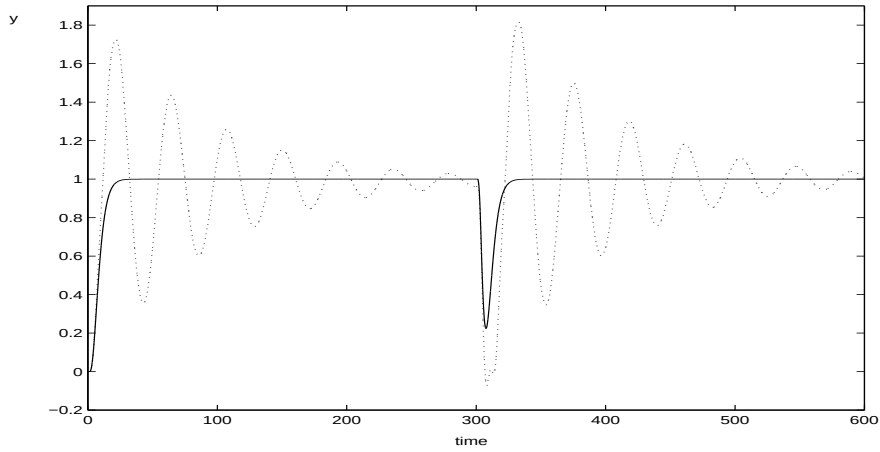


Fig. 9.20. Step response for Example 9.5.4
(····· proposed PID, —— proposed high order)