1

Basics of PID Control

1.1 Introduction

This chapter will introduce the basic ideas of PID control systems. It starts with an introduction to the roles of proportional control, integral control, and derivative control, followed by an introduction to the various tuning rules developed over the past several decades. These tuning rules are mainly developed for first order plus delay systems and are simple to use. However, they do not, in general, guarantee satisfactory control system performance. Simulation examples will be used to illustrate closed-loop performance.

This chapter is suitable for those who want to understand the very basics of PID control systems. By utilization of the tuning rules, it is possible to have an application of a PID control system without further exploration.

1.2 PID Controller Structure

There are four types of controllers that belong to the family of PID controllers: the proportional (P) controller, the proportional plus integral (PI) controller, the proportional plus derivative (PD) controller and the proportional plus integral plus derivative (PID) controller. To understand the roles of the controllers, in this section we will discuss each of the structures and the PID controller parameters. From the discussions, we will establish some basic knowledge about how to use these controllers in various applications.

1.2.1 Proportional Controller

The simplest controller is the proportional controller. With this term proportional, the feedback control signal $u(t)$ is computed in proportion to the feedback error $e(t)$ with the formula,

$$u(t) = K_c e(t) \tag{1.1}$$

where K_c is the proportional gain and the feedback error is the difference between the reference signal $r(t)$ and the output signal $y(t)$ ($e(t) = r(t) - y(t)$). The block diagram for the closed-loop feedback control configuration is shown in Figure 1.1 where $R(s)$, $E(s)$, $U(s)$, and $Y(s)$ are the Laplace transforms of the reference signal, feedback error, control

PID Control System Design and Automatic Tuning using MATLAB/Simulink, First Edition. Liuping Wang.

Companion website: www.wiley.com/go/Wang/PIDcontrolsystems

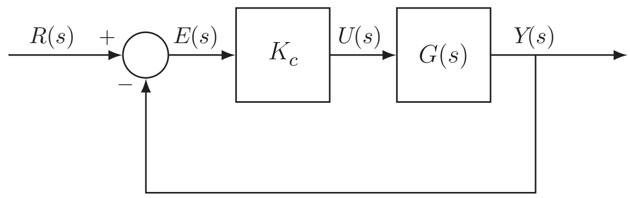

Figure 1.1 Proportional feedback control system.

signal, and output signal, respectively. $G(s)$ represents the Laplace transfer function of the plant.

Because of its simplicity, the proportional controller is often used in the cases when little information about the system is available and the required control performance in steady-state operation is not demanding. As the controller only involves one parameter to be determined, it is possible to choose K_c without detailed information about the plant.

One of the limitations of a simple proportional controller is that the steady-state error of the closed-loop control system will not be completely eliminated. We illustrate this point with the following example.

Example 1.1 *Suppose that the plant is a first order system with the following transfer function,*

$$G(s) = \frac{0.3}{s+1} \tag{1.2}$$

with proportional controller K_c *(*$K_c > 0$*). Suppose that the reference signal is a step signal with amplitude* 1 *and its Laplace transform is* $R(s) = \frac{1}{s}$*. Find the steady-state value of the output with respect to the reference signal.*

Solution. *From Figure 1.1, the closed-loop control system from the reference signal to the plant output has the transfer function,*

$$\frac{Y(s)}{R(s)} = \frac{K_c G(s)}{1 + K_c G(s)} = \frac{0.3K_c}{s + 1 + 0.3K_c}. \tag{1.3}$$

With any positive K_c*, the closed-loop system is stable where its pole is determined by the solution of the polynomial equation*[1]*,*

$$s + 1 + 0.3K_c = 0 \tag{1.4}$$

which is $-1 - 0.3K_c$*.*

The Laplace transform of the output, $Y(s)$*, is*

$$Y(s) = \frac{0.3K_c}{s + 1 + 0.3K_c} R(s) = \frac{0.3K_c}{s(s + 1 + 0.3K_c)}. \tag{1.5}$$

where $R(s) = \frac{1}{s}$*. Applying final value theorem to the stable closed-loop system, we calculate*

$$\lim_{t\to\infty} y(t) = \lim_{s\to 0} sY(s) = \lim_{s\to 0} s \times \frac{0.3K_c}{s(s + 1 + 0.3K_c)} = \frac{0.3K_c}{1 + 0.3K_c}. \tag{1.6}$$

1 This polynomial equation is called a closed-loop characteristic equation.

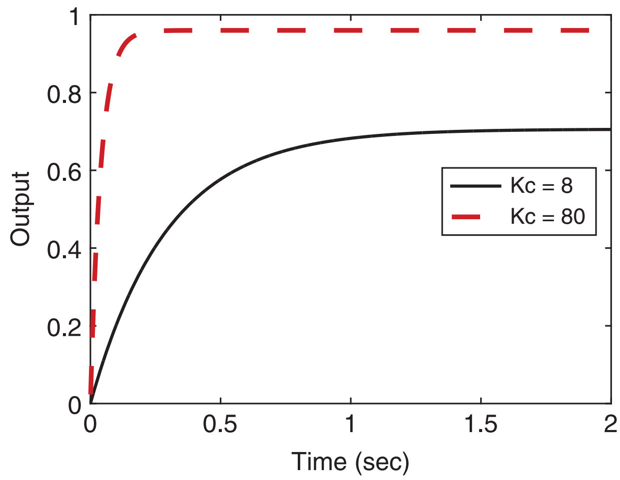

Figure 1.2 Closed-loop step response of a proportional control system (Example 1.1).

For any value of $0 < K_c < \infty$, $\lim_{t\to\infty} y(t) \neq 1$, *i.e. not equal to the desired value at the steady state response. Figure 1.2 shows the closed-loop step response with the proportional controller* $K_c = 8$ *and* $K_c = 80$, *respectively. It is seen that with the increased proportional gain* K_c, *the closed-loop response speed increases and the steady-state value becomes closer to the desired value 1.*

1.2.2 Proportional Plus Derivative Controller

In many applications, a proportional controller K_c is not sufficient to achieve a particular control objective such as stabilization or producing adequate damping for the closed-loop system. For instance, for a double integrator system with the transfer function:

$$G(s) = \frac{K}{s^2} \tag{1.7}$$

the closed-loop control system with a proportional controller K_c has a transfer function

$$\frac{Y(s)}{R(s)} = \frac{K_c K}{s^2 + K_c K}$$

which has a pair of closed-loop poles determined by the solutions of the polynomial equation:

$$s^2 + K_c K = 0.$$

These poles are at $\pm j\sqrt{K_c K}$. Thus, no matter what choice we make for K_c, the system still behaves in a sustained oscillatory manner because the pair of closed-loop poles are on the imaginary axis of the complex plane.

Now, assuming that we will additionally take the derivative of the feedback error signal into the control signal calculation, this leads to

$$u(t) = K_c e(t) + K_c \tau_D \dot{e}(t) \tag{1.8}$$

where τ_D is the derivative control gain.

The Laplace transfer function of (1.8) is calculated as

$$\frac{U(s)}{E(s)} = K_c + K_c \tau_D s.$$

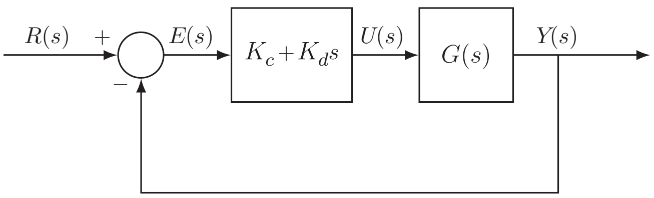

Figure 1.3 Proportional plus derivative feedback control system ($K_d = K_c\tau_D$).

This is what we called a proportional plus derivative (PD) controller.

The closed-loop feedback control configuration for a PD controller is shown in Figure 1.3. For the double integrator system (1.7), with the derivative term included in the controller, then the closed-loop transfer function becomes

$$\frac{Y(s)}{R(s)} = \frac{(K_c + K_c\tau_D s)G(s)}{1 + (K_c + K_c\tau_D s)G(s)} \tag{1.9}$$

$$= \frac{K_c K(1 + \tau_D s)}{s^2 + K_c K\tau_D s + K_c K}. \tag{1.10}$$

The closed-loop poles are determined by the solutions of the characteristic polynomial equation as

$$s^2 + K_c K\tau_D s + K_c K = 0$$

which are

$$s_{1,2} = \frac{-K_c K\tau_D \pm \sqrt{(K_c K\tau_D)^2 - 4K_c K}}{2}.$$

Clearly, we can choose the values of K_c and τ_D to achieve the desired closed-loop performance.

It is worthwhile emphasizing that almost without exception, the derivative term is different from the original form $K_c\tau_D s$ in the implementation. This is because the derivative term $K_c\tau_D(\frac{dr(t)}{dt} - \frac{dy(t)}{dt})$ is not practically implementable and the differentiation of the output signal $y(t)$ leads to amplification of the measurement noise. Thus, there are a few modifications with regard to the derivative terms. Firstly, in order to avoid the problem caused by a step reference trajectory change (the so-called derivative kick problem (Hägglund (2012))), the derivative action is only implemented on the output signal $y(t)$. Secondly, in order to avoid amplification of the measurement noise, a derivative filter is in a companionship with the derivative term $K_c\tau_D s$.

A commonly used derivative filter is a first order filter and has its time constant linked as a percentage to the actual derivative gain τ_D in the form:

$$F_D(s) = \frac{1}{\beta\tau_D s + 1} \tag{1.11}$$

where β is typically chosen to be 0.1 (10%). β is chosen to be larger if the measurement noise is severe.

With the derivative filter $F_D(s)$, a typical PD controller output is calculated as

$$u(t) = K_c(r(t) - y(t)) - K_c\tau_D\frac{dy_f(t)}{dt} \tag{1.12}$$

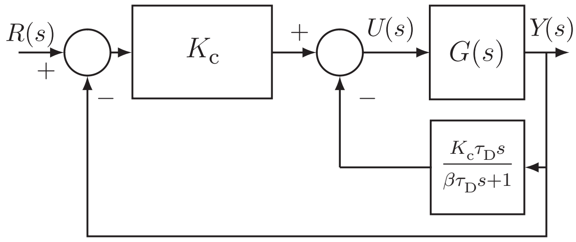

Figure 1.4 PD controller structure in implementation.

where $y_f(t)$ is the filtered output response using (1.11). In a Laplace transform, the control signal is expressed as

$$U(s) = K_c(R(s) - Y(s)) - \frac{K_c\tau_D s}{\beta\tau_D s + 1}Y(s). \tag{1.13}$$

Figure 1.4 shows the block diagram used for implementation of a PD controller with a filter.

If the derivative filter was not considered in the design, there is a certain degree of performance uncertainty due to the introduction of the filter. This may not be ideal for many applications. Designing a PD controller with the filter included will be discussed in Section 3.4.1.

1.2.3 Proportional Plus Integral Controller

A proportional plus integral (PI) controller is the most widely used controller among PID controllers. With the integral action, the steady-state error that had existed with the proportional controller alone (see Example 1.1) is completely eliminated. The output of the controller $u(t)$ is the sum of two terms, one from the proportional function and the other from the integral action, having the form,

$$u(t) = K_c e(t) + \frac{K_c}{\tau_I}\int_0^t e(\tau)\mathrm{d}\tau \tag{1.14}$$

where $e(t) = r(t) - y(t)$ is the error signal between the reference signal $r(t)$ and the output $y(t)$, K_c is the proportional gain, and τ_I is the integral time constant. The parameter τ_I is always positive, and its value is inversely proportional to the effect of the integral action taken by the PI controller. A smaller τ_I will result in a stronger effect of the integral action.

The Laplace transform of the controller output is

$$U(s) = K_c E(s) + \frac{K_c}{\tau_I s}E(s) \tag{1.15}$$

with $E(s)$ being the Laplace transform of the error signal $e(t)$. With this, the Laplace transfer function of the PI controller is expressed as

$$C(s) = \frac{U(s)}{E(s)} = \frac{K_c(\tau_I s + 1)}{\tau_I s} \tag{1.16}$$

Figure 1.5 shows a block diagram of the PI control system.

The example below is used to illustrate closed-loop control with a PI controller. For comparison purpose, we use the same plant as that used in Example 1.1.

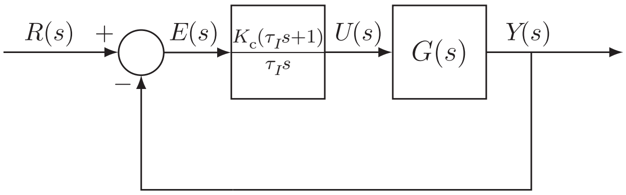

Figure 1.5 PI control system.

Example 1.2 *Assume that the plant is a first order system with the transfer function:*

$$G(s) = \frac{0.3}{s+1} \tag{1.17}$$

the PI controller has the proportional gain $K_c = 8$, and the integral time constant $\tau_I = 3$ and 0.5 respectively. Examine the locations of the closed-loop poles. With the reference signal $r(t)$ as a unit step signal, find the steady-state value of the closed-loop output $y(t)$.

Solution. *We calculate the closed-loop transfer function between the reference and output signals:*

$$\frac{Y(s)}{R(s)} = \frac{C(s)G(s)}{1 + C(s)G(s)}. \tag{1.18}$$

With $C(s)$ given in (1.16) and $G(s)$ in (1.17), we have

$$\frac{Y(s)}{R(s)} = \frac{0.3K_c\tau_I s + 0.3K_c}{\tau_I s^2 + \tau_I(1 + 0.3K_c)s + 0.3K_c}. \tag{1.19}$$

The closed-loop poles of this system are determined by the solutions of the closed-loop characteristic equation,

$$\tau_I s^2 + \tau_I(1 + 0.3K_c)s + 0.3K_c = 0 \tag{1.20}$$

which are

$$s_{1,2} = -\frac{1 + 0.3K_c}{2} \pm \frac{1}{2}\sqrt{(1 + 0.3K_c)^2 - \frac{1.2K_c}{\tau_I}}. \tag{1.21}$$

If the quantity

$$(1 + 0.3K_c)^2 - \frac{1.2K_c}{\tau_I} = 0$$

then there are two identical real poles located at

$$s_{1,2} = -\frac{1 + 0.3K_c}{2}.$$

If the quantity

$$(1 + 0.3K_c)^2 - \frac{1.2K_c}{\tau_I} > 0$$

then there are two real poles located at

$$s_{1,2} = -\frac{1 + 0.3K_c}{2} \pm \frac{1}{2}\sqrt{(1 + 0.3K_c)^2 - \frac{1.2K_c}{\tau_I}}.$$

If the quantity

$$(1 + 0.3K_c)^2 - \frac{1.2K_c}{\tau_I} < 0$$

then there are two complex poles located at

$$s_{1,2} = -\frac{1 + 0.3K_c}{2} \pm j\frac{1}{2}\sqrt{\frac{1.2K_c}{\tau_I} - (1 + 0.3K_c)^2}.$$

The closed-loop system is stable as long as K_c is positive and $0 < \tau_I < \infty$.

Applying the final value theorem, we calculate

$$\lim_{t\to\infty} y(t) = \lim_{s\to 0} sY(s) = \lim_{s\to 0} s \times \frac{0.3K_c\tau_I s + 0.3K_c}{\tau_I s^2 + \tau_I(1 + 0.3K_c)s + 0.3K_c}\frac{1}{s}$$
$$= \frac{0.3K_c}{0.3K_c} = 1, \tag{1.22}$$

where the steady-state value is equal to the reference signal, and it is independent of the value of integral time constant τ_I. Figure 1.6 shows for the same $K_c = 8$ as in Example 1.1 the closed-loop step response with $\tau_I = 3$ and $\tau_I = 0.5$, respectively. It is seen that as τ_I reduces, the closed-loop response speed becomes faster. Nevertheless, the steady-state responses with both τ_I values are equal to one.

It is often the case that the output of a PI control system exhibits overshoot to a step reference signal. The percentage of overshoot increases as higher control performance demanded. This may cause a conflict in the PI control performance specifications: on the one hand a fast control system response is desired, and yet on the other hand, the overshoot is not desirable when step reference changes are performed. The overshooting problem in reference change could be reduced by a small change in the configuration of the PI controller. This small change is to put the proportional control on the output signal $y(t)$, instead of the feedback error $e(t) = r(t) - y(t)$. More specifically, the control signal $u(t)$ is calculated using the following relation,

$$u(t) = -K_c y(t) + \frac{K_c}{\tau_I}\int_0^t (r(\tau) - y(\tau))d\tau. \tag{1.23}$$

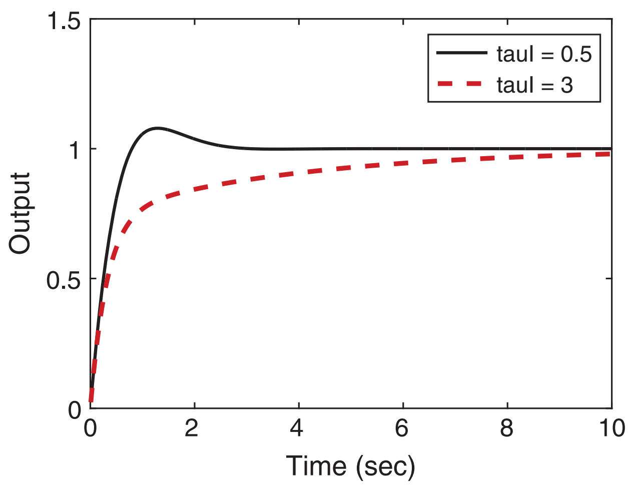

Figure 1.6 Closed-loop step response of a PI control system (Example 1.2).

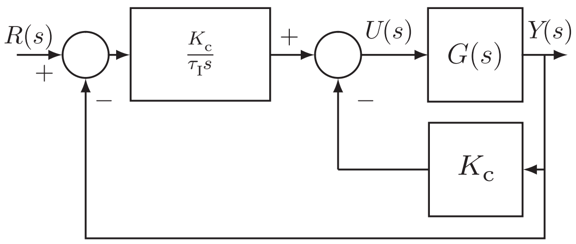

Figure 1.7 IP controller structure.

Applying a Laplace transform to this equation leads to the Laplace transform of the controller output in relation to the reference and the output as

$$U(s) = -K_c Y(s) + \frac{K_c}{\tau_I s}(R(s) - Y(s)). \tag{1.24}$$

Figure 1.7 shows a block diagram of this PI closed- loop control configuration. This type of implementation is called an IP controller in the literature, which is an alternative PI controller configuration. In Section 2.4, this PI controller structure is examined in the context of a reference filter within the framework of a two degrees of freedom control system. To demonstrate how this simple modification in the PI controller configuration can reduce the overshoot effect, we examine the following example.

Example 1.3 *Assume that the plant is described by the transfer function:*

$$G(s) = \frac{1}{s(s+1)^3} \tag{1.25}$$

and the PI controller has the parameters: $K_c = 0.56$, $\tau_I = 8$ [2]. *Find the closed-loop transfer function between the reference signal* $R(s)$ *and the output signal* $Y(s)$ *for the original PI controller structure (see Figure 1.5) and the IP controller structure (see Figure 1.7), and compare their closed-loop step responses.*

Solution. *With the PI controller in the original structure, the closed-loop transfer function between the reference signal* $R(s)$ *and the output signal* $Y(s)$ *is calculated using,*

$$\frac{Y(s)}{R(s)} = \frac{C(s)G(s)}{1 + C(s)G(s)}. \tag{1.26}$$

By substituting the plant transfer function (1.25) and the PI controller structure (1.16), the closed-loop transfer function is

$$\frac{Y(s)}{R(s)} = \frac{K_c(\tau_I s + 1)}{\tau_I s^2(s+1)^3 + K_c(\tau_I s + 1)}. \tag{1.27}$$

With the PI controller in the IP structure, the Laplace transform of the control signal $U(s)$ *is defined by (1.24). By substituting this control signal into the Laplace transform of the output* $Y(s)$ *via the following equation,*

$$Y(s) = \frac{1}{s(s+1)^3} U(s) \tag{1.28}$$

re-grouping and simplification lead to the closed-loop transfer function:

$$\frac{Y(s)}{R(s)} = \frac{K_c}{\tau_I s^2(s+1)^3 + K_c(\tau_I s + 1)}. \tag{1.29}$$

---

2 This PI controller was designed using frequency response data in Wang and Cluett (2000).

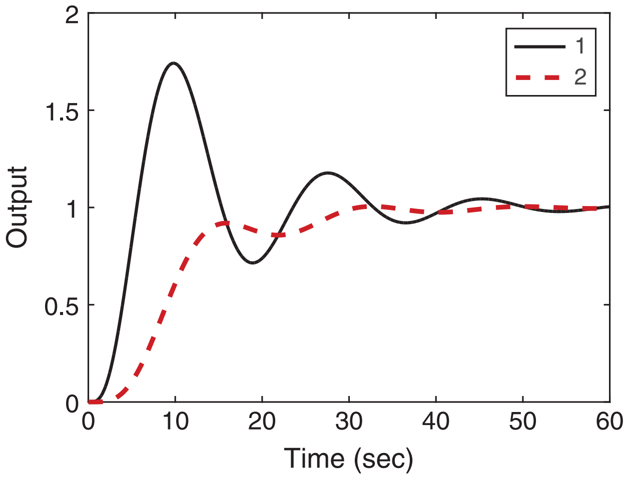

Figure 1.8 Closed-loop step response of PI control system (Example 1.3). Key: line (1) response from the original structure; line (2) response from the IP structure.

By comparing the closed-loop transfer function (1.27) from the original PI controller structure with the one (1.29) from the alternative structure, we notice that both transfer functions have the same denominator, however, the one from the original structure has a zero at $-\frac{1}{\tau_I}$. *Because of this zero, the original closed-loop step response may have an overshoot.*

Indeed, the closed-loop step responses for both structures are simulated and compared in Figure 1.8, which shows that the original PI closed-loop control system has a large overshoot; in contrast, the IP closed-loop control system has reduced this overshoot. The penalty for reducing the overshoot is the slower reference response speed.

The closed-loop transfer function obtained with IP controller structure can also be interpreted as a two degrees of freedom control system with a reference filter $H(s) = \frac{1}{\tau_I s+1}$. *This topic will be further discussed in Section 2.4.2.*

Another form of PI controller, perhaps more convenient for model-based controller design as in Chapter 3, is described by:

$$C(s) = \frac{c_1 s + c_0}{s} \tag{1.30}$$

This form of PI controller is identical to the original PI controller structure when the parameters of K_c and τ_I are selected as

$$K_c = c_1; \quad \tau_I = \frac{c_1}{c_0} \tag{1.31}$$

1.2.4 PID Controllers

A PID controller consists of three terms: the proportional (P) term, the integral (I) term, and the derivative (D) term. In an ideal form, the output $u(t)$ of a PID controller is the sum of the three terms,

$$u(t) = K_c e(t) + \frac{K_c}{\tau_I} \int_0^t e(\tau) d\tau + K_c \tau_D \frac{de(t)}{dt} \tag{1.32}$$

where $e(t) = r(t) - y(t)$ is the feedback error signal between the reference signal $r(t)$ and the output $y(t)$, and τ_D is the derivative control gain. The Laplace transfer function of the PID controller is

$$\frac{U(s)}{E(s)} = K_c(1 + \frac{1}{\tau_I s} + \tau_D s). \tag{1.33}$$

If the design is sound, the sign of τ_D is positive. If the sign of τ_D is negative, the derivative control term is to be neglected and instead a PI controller should be chosen.

Analogously to the proportional plus derivative controller described in Section 1.2.2, for most of the applications the derivative control is implemented on the output only with a derivative filter. For this reason, the control signal $U(s)$ is expressed in the following form:

$$U(s) = K_c(1 + \frac{1}{\tau_I s})(R(s) - Y(s)) - \frac{K_c \tau_D s}{\beta \tau_D s + 1} Y(s). \tag{1.34}$$

Figure 1.9 shows the block diagram of the PID controller structure.

To reduce the overshoot in output response to a step reference change, the proportional term in the PID controller may also be implemented on the plant output. In this case, the control signal is calculated using

$$u(t) = -K_c y(t) + \frac{K_c}{\tau_I} \int_0^t (r(\tau) - y(\tau)) d\tau - K_c \tau_D \frac{dy_f(t)}{dt}. \tag{1.35}$$

Accordingly, the Laplace transform of the control signal is expressed as

$$\begin{aligned} U(s) &= -K_c Y(s) + \frac{K_c}{\tau_I s}(R(s) - Y(s)) - \frac{K_c \tau_D s}{\beta \tau_D s + 1} Y(s) \\ &= \frac{K_c}{\tau_I s}(R(s) - Y(s)) - \frac{K_c(\tau_D(\beta + 1)s + 1)}{\beta \tau_D s + 1} Y(s). \end{aligned} \tag{1.36}$$

Figure 1.10 shows a block diagram of the alternative PID controller structure (called an IPD controller).

The example below is used to illustrate the effect of the derivative term in the closed-loop control. The starting point is the PI controller designed in Example 1.3, based on which a derivative term is introduced.

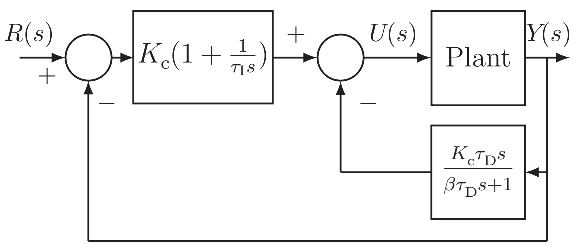

Figure 1.9 PID controller structure.

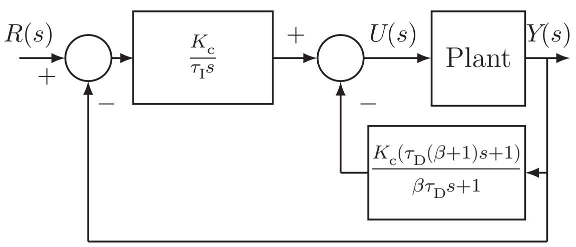

Figure 1.10 IPD controller structure.

Example 1.4 *Suppose that the plant is described by the transfer function:*

$$G(s) = \frac{1}{s(s+1)^3} \tag{1.37}$$

and the PI part of the controller has the parameters: $K_c = 0.56$, $\tau_I = 8$. Choosing $\tau_D = 0.1$ and 1, find the closed-loop transfer function for the PID control system shown in Figure 1.9 and simulate its closed-loop performance. Also, find the closed-loop transfer function of the IPD control system shown in Figure 1.10 and simulate its closed-loop step response.

Solution. *The control signal from Figure 1.9 has the Laplace transform given by Equation (1.34). Substituting $U(s)$ into the following equation,*

$$Y(s) = \frac{1}{s(s+1)^3}U(s) \tag{1.38}$$

re-grouping and re-arranging lead to the closed-loop transfer function:

$$\frac{Y(s)}{R(s)} = \frac{K_c(\tau_I s+1)(\beta\tau_D s+1)}{\tau_I s^2(s+1)^3(\beta\tau_D s+1) + K_c(\tau_I s+1)(\beta\tau_D s+1) + K_c\tau_I\tau_D s^2}. \tag{1.39}$$

There are two zeros in the closed-loop transfer function: $-\frac{1}{\tau_I}$ caused by the integral control, and $-\frac{1}{\beta\tau_D}$ caused by the derivative control. Figure 1.11(a) shows the closed-loop step responses for $\tau_D = 0.1$ and $\tau_D = 1$ respectively. With the increase in τ_D, the oscillation in the closed-loop response is reduced. However, there is a large overshoot in the output response.

Using the IPD structure as shown in Figure 1.10, we calculate the closed-loop transfer function by using the Laplace transform of the controller output given in Equation (1.36). Substituting this control signal into the plant output (see Equation (1.38)), the closed-loop transfer function is

$$\frac{Y(s)}{R(s)} = \frac{K_c(\beta\tau_D s+1)}{\tau_I s^2(s+1)^3(\beta\tau_D s+1) + K_c(\tau_I s+1)(\beta\tau_D s+1) + K_c\tau_I\tau_D s^2}. \tag{1.40}$$

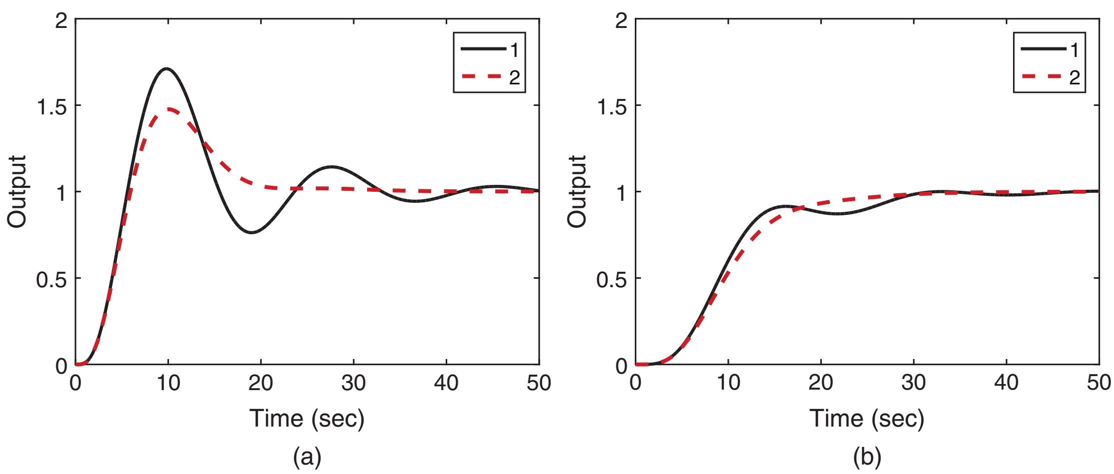

Figure 1.11 Step responses of PID control system (Example 1.4). (a) Responses for PID structure. (b) Responses for the IPD structure. Key: line (1) $\tau_D = 0.1$, line (2) $\tau_D = 1$ ($K_c = 0.56$, $\tau_I = 8$).

With this implementation, the denominator of the closed-loop transfer function is the same; however, there is only one zero at $-\frac{1}{\beta\tau_D}$ *caused by the derivative control. Figure 1.11(b) shows the closed-loop step responses with an IPD controller. In comparison with the responses from the previous case, it is seen that the overshoot in the closed-loop responses has been eliminated, however their response speed becomes slower.*

1.2.5 The Commercial PID Controller Structure

In the PID controller design, the following structure is commonly used for determining the parameters K_c, τ_I and τ_D, where the controller transfer function $C(s)$ is expressed as

$$C(s) = K_c(1 + \frac{1}{\tau_I s} + \tau_D s). \tag{1.41}$$

However, as demonstrated in this section, there are several variations in PID controller structure available for the realization of the control system, and different realization leads to different control system performance with the same set of PID controller parameters.

In order to be more flexible to the users, the commercial PID controllers (see Alfaro and Vilanova (2016)) from manufacturers such as ABB, Siemens, and National Instruments take the following general form with the Laplace transform of the control signal:

$$U(s) = K_c\left(\gamma_1 R(s) - Y(s) + \frac{1}{\tau_I s}(R(s) - Y(s)) + \frac{\tau_D s}{\beta\tau_D s + 1}(\gamma_2 R(s) - Y(s))\right). \tag{1.42}$$

where the coefficients $0 \le \gamma_1 \le 1$ and $0 \le \gamma_2 \le 1$ are for the weighting on the reference signal, and as before, the coefficient $0 \le \beta \le 1$ determines the appropriate derivative filtering action. There are several special combinations of the parameters γ_1, γ_2, and β that are commonly encountered.

1. When $\gamma_1 = 1$, $\gamma_2 = 0$, and $0 \le \beta \le 1$, the PID controller becomes identical to the case shown in Figure 1.9, where the derivative control with filter is implemented on the output only.
2. When $\gamma_1 = 0$ and $\gamma_2 = 0$, the PID controller becomes the IPD controller shown in Figure 1.10, where both the proportional control and derivative control are implemented on the output only.
3. When $\gamma_1 = 1$, $\gamma_2 = 1$, and $0 \le \beta \le 1$, the implementation of the PID controller puts proportional control, integral control, and derivative control with filter on the feedback error $R(s) - Y(s)$.
4. When $\gamma_1 = 1$, $\gamma_2 = 1$, and $\beta = 0$, the PID controller becomes the case where no derivative filter is used in the implementation. This will severely amplify the measurement noise.

It is worthwhile emphasizing that the parameters γ_1 and γ_2 only affect the closed-loop response to the reference signal $R(s)$ and they play no role in the closed-loop stability. We have examined the cases where γ_1 and γ_2 are either 0 or 1. However, we can extend the results to the situations where the parameters are between 0 and 1 and expect a compromised result. Upon understanding their roles, we can choose the appropriate coefficients according to the actual applications.

1.2.6 Food for Thought

1. The PID controllers are expressed in terms of the parameters K_c, τ_I and τ_D. What are the possible signs of K_c, τ_I and τ_D?
2. When you increase the magnitude of K_c, do you expect the action of proportional control to decrease or increase? When you increase τ_I, do you expect the action of integral control to decrease or increase? when you increase τ_D, do you expect the action of derivative control to decrease or increase?
3. What are the roles of integrator in a PID controller?
4. Can you implement the integrating control on output only? If not, explain the reason.
5. In many applications, we will put the proportional control on the feedback error, which is the original PI controller. Can you reduce the overshoot by using a ramp reference signal in the early part of the response?

1.3 Classical Tuning Rules for PID Controllers

This section will discuss the classical tuning rules that have existed for the past several decades and have withstood the test of time. Although all tuning rules are rule-based, there is still certain knowledge assumed for the system to be controlled.

1.3.1 Ziegler–Nichols Oscillation Based Tuning Rules

Ziegler–Nichols oscillation based tuning rules are to use closed-loop controlled testing to obtain the critical information needed for determining the PID controller parameters.

In the closed-loop control testing, the controller is set to proportional mode without integrator and derivative action. The sign of K_c must be the same as the steady-state gain of the plant for the reason of introducing negative feedback in the control system. With the proportional closed-loop control, the feedback gain K_c is set to be a very small value in magnitude to begin the experiment. The value of K_c is gradually increased until the control signal $u(t)$ exhibits sustained oscillation (see Figure 1.12). There are two parameters obtained from this test: the value of K_c that has caused the oscillation and the period of the oscillation. We denote this particular K_c as K_o and the period as P_o. With these two parameters, the PID controller parameters are presented for the Ziegler–Nichols tuning rules in Table 1.1.

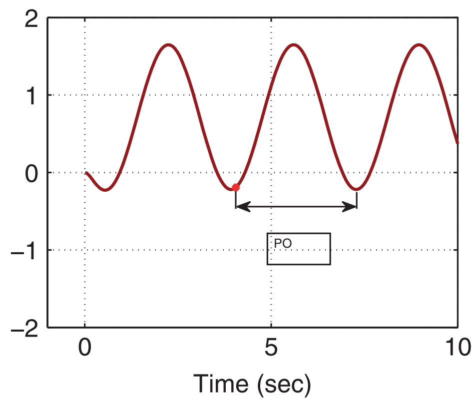

Figure 1.12 Sustained closed-loop oscillation (control signal).

Table 1.1 Ziegler–Nichols tuning rule using oscillation testing data.

	K_c	τ_I	τ_D
P	$0.5K_o$		
PI	$0.45K_o$	$\frac{P_o}{1.2}$	
PID	$0.60K_o$	$\frac{P_o}{2}$	$\frac{P_o}{8}$

A proportional control will not cause sustained oscillation for first order plant and second order plant with a stable zero. Thus, the tuning rule is not applicable to these two classes of stable plants. The following example is used to illustrate an application of the tuning rule.

Example 1.5 *Assume that a continuous time plant has the Laplace transfer function:*

$$G(s) = \frac{s-2}{(s+1)(s+2)(s+3)}. \tag{1.43}$$

Find the PI and PID controller parameters using Ziegler–Nichols tuning rule and simulate the closed-loop control systems.

Solution. *We build a Simulink simulation program for proportional control as illustrated in Figure 1.1. Since this system has a negative steady-state gain of* $-\frac{1}{3}$, *so the feedback control gain should be negative.*[3] *Beginning the tuning process by setting* $K_c = -1$ *and decreasing gradually to* $K_c = -7.5$, *the closed-loop control system exhibits sustained oscillation as shown in Figure 1.12. From this figure, the period of oscillation reads as* 3.35. *Based on Table 1.1, the proportional gain for the PI controller is* $K_c = 0.45 \times (-7.5) = -3.38$ *and the integral time constant* $\tau_I = \frac{3.35}{1.2} = 2.79$. *The proportional gain for the PID controller is* $K_c = 0.6 \times (-7.5) = -4.5$, $\tau_I = \frac{3.35}{2} = 1.68$, *and* $\tau_D = \frac{3.35}{8} = 4.2$. *The PI and PID control systems are simulated where the reference signal is a unit step signal. Figure 1.13 compares the closed-loop output responses based on the PI and PID controller structures. Here the derivative control is implemented on the output only with a filter time constant* $0.1\tau_D$. *It is seen that with the derivative term, the closed-loop oscillation existing in the PI controller is reduced.*

In general, the Ziegler and Nichols tuning rules using the oscillation method lead to quite aggressive responses with oscillations in closed-loop responses that are undesirable for many control applications. In Tyreus and Luyben (1992), a smaller K_c and a larger τ_I are recommended to reduce the oscillations, which have the following values:

$$K_c = 0.313K_o; \quad \tau_I = 2.2P_o.$$

It is important to point out that generating sustained oscillation by increasing the controller gain is not a safe operation because a small error in the tuning process could

3 We evaluate the steady-state gain of the transfer function by letting $s = 0$ and calculating the value of the transfer function.

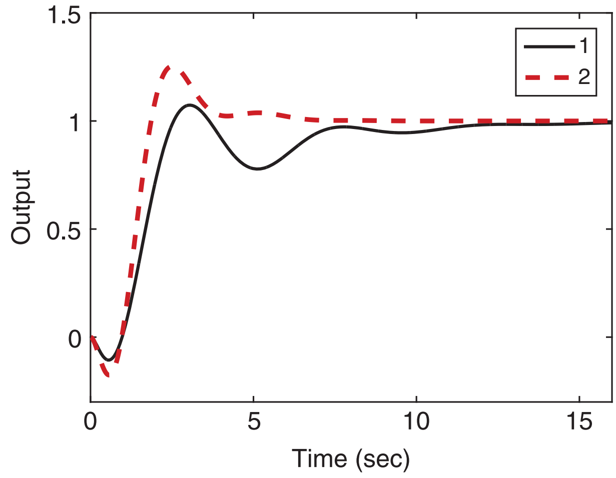

Figure 1.13 Comparison of closed-loop output response using Ziegler–Nichols rules (Example 1.5). Key: line (1) PI control; line (2) PID control.

cause the closed-loop system to become unstable. This unsafe procedure is replaced by using relay feedback control in Chapter 9, which also produces a sustained closed-loop oscillation.

1.3.2 Tuning Rules based on the First Order Plus Delay Model

The majority of tuning rules existing in the literature are based on a first order plus delay model, which has the following transfer function:

$$G(s) = \frac{K_{ss}e^{-ds}}{\tau_M s + 1} \tag{1.44}$$

where K_{ss} is the steady-state gain of the system, d is the time delay, and τ_M is the time constant. This is mainly because the primary applications of tuning rules are for process control where typically the process is stable with time delay. There are many methods available for obtaining a first order plus delay model. Among them is an incredibly simple procedure that is called fitting a reaction curve. This reaction curve is the so-called step response test.

The reaction curve is obtained by performing a plant step response test in open-loop operation, therefore we assume that the plant is stable, although it could be applied to integrating plant with caution. When performing this test, the plant input signal $u(t)$ takes a step change from an initial constant value U_0 to a normal operation value, U_s; the measurement of the plant output signal $y(t)$ in response to the step input change gives us the plant step response test data or the reaction curve. The response test completes when the value of the output signal reaches a constant or the signal fluctuates around a constant value due to noise and disturbances. Figure 1.14 illustrates a typical set of step response testing data, with the step change occurring at time $t = 0$. Figure 1.14(a) shows that the input takes a step change from the initial value U_0 to the final value U_s and Figure 1.14(b) shows the actual output response from the steady-state output position Y_0 to the steady-state output position Y_s.

The plant information obtained from the test includes the steady-state gain, defined as

$$K_{ss} = \frac{Y_s - Y_0}{U_s - U_0}. \tag{1.45}$$

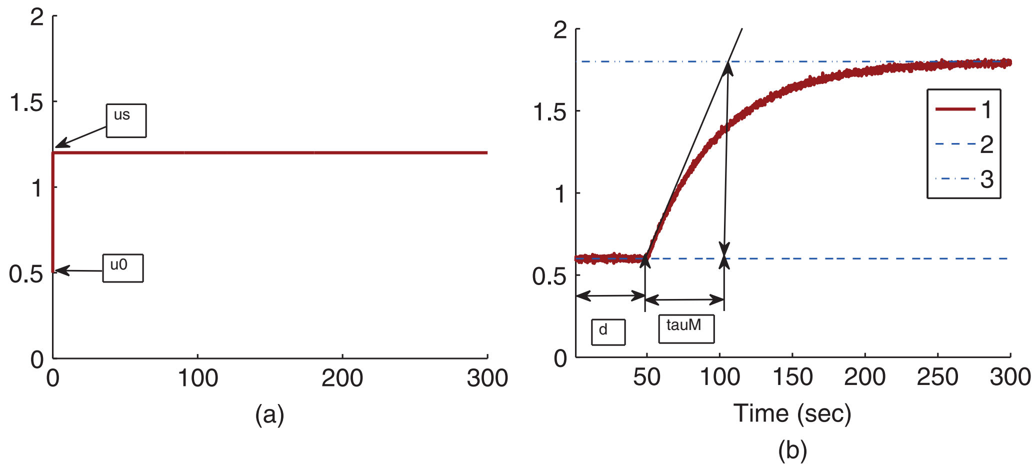

Figure 1.14 Step response data. (a) Input signal. (b) Output signal. Key: line (1) the output response; line (2) steady-state output position before the response (Y_0); line (3) steady-state output position in completion of the response (Y_s).

The time delay d is shown in the figure, which is the delayed time when the output responds to the change in the input signal. The parameter time delay d reflects the situation that the output response remains unchanged despite the step input signal being injected. Thus, it is estimated using the time difference between when the step reference change occurred ($t = 0$ for this figure) and when the output response moved away from its steady-state value (see the time interval in Figure 1.14(b) marked with the first set of arrows). A line with maximum slope is drawn on Figure 1.14(b), which is intersected with the line corresponding to the indicator of Y_s. The intersecting point shown in Figure 1.14(b) determines the value of τ_M that is a measurement of the dynamic response time.

Alternatively, because the step response of a first order system ($G(s) = \frac{K_{ss}}{\tau_M s+1}$) to a unit step input signal can be expressed as

$$g(t) = K_{ss}(1 - e^{\frac{-t}{\tau_M}})$$

and when the variable time $t = \tau_M$,

$$g(\tau_M) = K_{ss}(1 - e^{-1}) = 0.632K_{ss}$$

thus, we can determine the time constant τ_M using 63.2% of the rising time in the step response. This estimation of time constant gives a different value from the case when using the maximum slope approach. For the majority of the applications, this will result in a smaller time constant τ_M, and from the empirical tuning rules stated in the later part of the section, a smaller proportional gain K_c will follow. One can evaluate this approach as an exercise using Problem 1.2.

Essentially, the step response test gives the parameters in the first order plus delay description of the process as in (1.44).

There is a second set of Ziegler–Nichols tuning rules that is based on the plant step response test data. This is also called the Ziegler–Nichols tuning rules using reaction curve. With these parameters, Ziegler–Nichols tuning rules using a reaction curve are given in Table 1.2. By the nature of this testing procedure (open-loop testing), the tuning rules should apply to stable systems.

Table 1.2 Ziegler-Nichols tuning rules with a reaction curve.

	K_c	τ_I	τ_D
P	$\frac{\tau_M}{K_{ss}d}$		
PI	$0.9\frac{\tau_M}{K_{ss}d}$	$3d$	
PID	$1.2\frac{\tau_M}{K_{ss}d}$	$2d$	$0.5d$

Table 1.3 Cohen–Coon tuning rules with a reaction curve.

	K_c	τ_I	τ_D
P	$\frac{\tau_M}{K_{ss}d}\left(1+\frac{d}{3\tau_M}\right)$		
PI	$\frac{\tau_M}{K_{ss}d}\left(0.9+\frac{d}{12\tau_M}\right)$	$\frac{d(30\tau_M+3d)}{9\tau_M+20d}$	
PID	$\frac{\tau_M}{K_{ss}d}\left(\frac{4}{3}+\frac{d}{4\tau_M}\right)$	$\frac{d(32\tau_M+6d)}{13\tau_M+8d}$	$\frac{4d\tau_M}{11\tau_M+2d}$

There is another set of tuning rules that are derived based on the reaction curve, termed Cohen and Coon tuning rules. Table 1.3 gives the PID controller parameters calculated from Cohen and Coon tuning rules.

For the estimation of time delay d, τ_M, and K_{ss} when using MATLAB, it is a fairly straight forward procedure to draw the lines and pinpoint the data points. The MATLAB command for finding the point on a graph is called ginput. For example, by typing

```
[a,b]=ginput(1)
```

a cross hair will appear on the MATLAB figure and a double click on the point of interest will yield the exact values we need. This graphic procedure will be demonstrated in the example section (see Section 1.5).

1.3.3 Food for Thought

1. Can you apply Ziegler-Nichols oscillation tuning method to a first order system? Why?
2. Can you apply the reaction curve based tuning rules to unstable systems? Why?
3. How do we decide the sign of the proportional feedback controller gain when using the Ziegler-Nichols oscillation method?
4. Can you envisage any potential danger when using Ziegler- Nichols oscillation method?
5. How do you design a step response experiment?
6. What information will the step response experiment provide?

7. How do you determine steady-state gain, parameter τ_M and time delay d from a reaction curve?
8. What are your observations when comparing Ziegler-Nichols and Cohen-Coon tuning rules, in terms of signs and values of K_c, τ_I and τ_D?
9. Is there any desired closed-loop performance specification among the tuning rules?

1.4 Model Based PID Controller Tuning Rules

This section will discuss the PID controller tuning rules that are derived based on a first order plus delay model. These tuning rules worked well in applications.

1.4.1 IMC-PID Controller Tuning Rules

The internal model control (IMC)-PID tuning rules (Rivera et al. (1986)) are proposed on the basis of a first order plus delay model:

$$G_{\mathrm{M}}(s) = \frac{K_{\mathrm{ss}} e^{-ds}}{\tau_{\mathrm{M}} s + 1}.$$

When using the IMC-PID tuning rules, a desired closed-loop response is specified by the transfer function from the reference signal to the output:

$$\frac{Y(s)}{R(s)} = \frac{e^{-ds}}{\tau_{\mathrm{cl}} s + 1}$$

where τ_{cl} is the desired time constant chosen by the user. The PI controller parameters are related to the first order plus delay model and the desired closed-loop time constant τ_{cl}, which are given as:

$$\begin{aligned} K_{\mathrm{c}} &= \frac{1}{K_{\mathrm{ss}}} \frac{\tau_{\mathrm{M}}}{\tau_{\mathrm{cl}} + d} \\ \tau_{\mathrm{I}} &= \tau_{\mathrm{M}}. \end{aligned} \tag{1.46}$$

If the system has a second order transfer function with time delay in the following form:

$$G_{\mathrm{M}}(s) = \frac{K_{\mathrm{ss}} e^{-ds}}{(\tau_1 s + 1)(\tau_2 s + 1)}$$

then a PID controller is recommended. Assuming that $\tau_2 \leq \tau_1$, then the PID controller parameters are calculated as

$$\begin{aligned} K_{\mathrm{c}} &= \frac{1}{K_{\mathrm{ss}}} \frac{\tau_1}{\tau_{\mathrm{cl}} + d} \\ \tau_{\mathrm{I}} &= \tau_1 \\ \tau_{\mathrm{D}} &= \tau_2. \end{aligned} \tag{1.47}$$

Later on, it was realized that the choice of τ_{I} basically led to a pole-zero cancellation in the control system. Such a pole-zero cancellation will limit, as discussed in the next chapter, the control system performance in the disturbance rejection, particularly when τ_1 is large[4]. In Skogestad (2003), the IMC-PID tuning rules are modified to reduce the integral time constant as

$$\tau_{\mathrm{I}} = \min[\tau_1, 4(\tau_{\mathrm{cl}} + d)] \tag{1.48}$$

while K_{c} and τ_{D} are unchanged from (1.47).

The IMC-PID controller tuning rules are also extended to integrating systems in Skogestad (2003). Although the system has an integrator as part of its dynamics, integral control is still required for disturbance rejection (see Chapter 2).

Assuming that the system has the integrator with delay model:

$$G_{\mathrm{M}}(s) = K_{\mathrm{ss}}\frac{\mathrm{e}^{-ds}}{s} \tag{1.49}$$

then a PI controller is recommended with the following parameters:

$$\begin{aligned} K_{\mathrm{c}} &= \frac{1}{K_{\mathrm{ss}}(\tau_{\mathrm{cl}} + d)} \\ \tau_{\mathrm{I}} &= 4(\tau_{\mathrm{cl}} + d). \end{aligned} \tag{1.50}$$

If the transfer function for the integrating system has the form:

$$G_{\mathrm{M}}(s) = K_{\mathrm{ss}}\frac{\mathrm{e}^{-ds}}{s(\tau_1 s + 1)} \tag{1.51}$$

then a PID controller is recommended to have the following parameters:

$$\begin{aligned} K_{\mathrm{c}} &= \frac{1}{K_{\mathrm{ss}}(\tau_{\mathrm{cl}} + d)} \\ \tau_{\mathrm{I}} &= 4(\tau_{\mathrm{cl}} + d) \\ \tau_{\mathrm{D}} &= \tau_1. \end{aligned} \tag{1.52}$$

If the system has a double integrator with the transfer function

$$G_{\mathrm{M}}(s) = K_{\mathrm{ss}}\frac{\mathrm{e}^{-ds}}{s^2} \tag{1.53}$$

then a PID controller is recommended with the following parameters:

$$\begin{aligned} K_{\mathrm{c}} &= \frac{1}{K_{\mathrm{ss}}(\tau_{\mathrm{cl}} + d)^2} \\ \tau_{\mathrm{I}} &= 4(\tau_{\mathrm{cl}} + d) \\ \tau_{\mathrm{D}} &= 4(\tau_{\mathrm{cl}} + d). \end{aligned} \tag{1.54}$$

The IMC-PID controller tuning rules will be studied in Examples 2.1 and 2.2.

1.4.2 Padula and Visioli Tuning Rules

Several sets of tuning rules were introduced in Padula and Visioli (2011) and Padula and Visioli (2012). These tuning rules are based on the first order plus delay model:

$$G_{\mathrm{M}}(s) = \frac{K_{\mathrm{ss}}\mathrm{e}^{-ds}}{\tau_{\mathrm{M}} s + 1}.$$

4 This slow disturbance rejection problem will be analyzed using sensitivity analysis in Problem 2.8.

Table 1.4 Padula and Visioli tuning rules (PI controller).

	$M_s = 1.4$	$M_s = 2$
K_c	$\frac{1}{K_{ss}}\left(0.2958\left(\frac{d}{d+\tau_M}\right)^{-1.014} - 0.2021\right)$	$\frac{1}{K_{ss}}\left(0.5327\left(\frac{d}{d+\tau_M}\right)^{-1.029} - 0.2428\right)$
τ_I	$\tau_M\left(1.624\left(\frac{d}{\tau_M}\right)^{0.2269} - 0.5556\right)$	$\tau_M\left(1.44\left(\frac{d}{\tau_M}\right)^{0.4825} - 0.1019\right)$

Table 1.5 Padula and Visioli tuning rules (PID controller).

	$M_s = 1.4$	$M_s = 2$
K_c	$\frac{1}{K_{ss}}\left(0.1724\left(\frac{d}{d+\tau_M}\right)^{-1.259} - 0.05052\right)$	$\frac{1}{K_{ss}}\left(0.2002\left(\frac{d}{d+\tau_M}\right)^{-1.414} + 0.06139\right)$
τ_I	$\tau_M\left(0.5968\left(\frac{d}{\tau_M}\right)^{0.6388} + 0.07886\right)$	$\tau_M\left(0.446\left(\frac{d}{\tau_M}\right)^{0.9541} + 0.1804\right)$
τ_D	$\tau_M\left(0.5856\left(\frac{d}{\tau_M}\right)^{0.5004} - 0.1109\right)$	$\tau_M\left(0.6777\left(\frac{d}{\tau_M}\right)^{0.4968} - 0.1499\right)$

They were derived using optimization methods for minimizing an error function together with the sensitivity peak in the frequency domain (see Chapter 2).

Here, we only include two sets of the tuning rules introduced for disturbance rejection in their paper. Tables 1.4 and 1.5 present the tuning rules for PI and PID controllers, respectively. Each table contains two sets of rules. For the specification of $M_s = 1.4$ (M_s corresponds to the sensitivity peak (see Chapter 2)), the set of tuning rules is expected to produce a slower closed-loop response in comparison to that with $M_s = 2$. It is worthwhile mentioning from Tables (1.4) and (1.5) that both PI and PID controller parameters are invalid for the time delay $d = 0$ because as $d \to 0$, the proportional control gain $K_c \to \infty$.

A frequency response analysis for the Padula and Visioli tuning rules based control system will be presented in Chapter 2 where an example (see Example 2.4) will be given to show the sensitivity functions and their Nyquist diagrams.

1.4.3 Wang and Cluett Tuning Rules

In Wang and Cluett (2000), a first order plus delay model was used to derive several tuning rules for PID controllers. The rules were calculated using a frequency response analysis based on the ratio of the time constant τ_M to the time delay d, which is defined as $L = \tau_M/d$. A desired closed-loop time constant is chosen as a scale of the time delay d. Among them, which works quite well in applications, is a particular choice with the desired closed-loop time constant being equal to the time delay d. For this choice, the PID controller parameters are presented in Table 1.6.

Table 1.6 Wang–Cluett tuning rules with reaction curve ($L = \tau_M/d$).

	K_c	τ_I	τ_D
P	$\dfrac{0.13+0.51L}{K_{ss}}$		
PI	$\dfrac{0.13+0.51L}{K_{ss}}$	$\dfrac{d(0.25+0.96L)}{0.93+0.03L}$	
PID	$\dfrac{0.13+0.51L}{K_{ss}}$	$\dfrac{d(0.25+0.96L)}{0.93+0.03L}$	$\dfrac{d(-0.03+0.28L)}{0.25+L}$

1.4.4 Food for Thought

1. In the IMC-PID controller tuning rules, if a faster closed-loop response is desired, would you increase or decrease the desired closed-loop time constant τ_{cl}?
2. For the tuning rules derived by Padula and Visioli, are there any closed-loop performance parameters chosen by the user?
3. Intuitively, would you increase the proportional controller gain K_c if the PID controlled system is unstable when using the Padula and Visioli's tuning rules?
4. Would you increase the integral time constant τ_I if the PID controlled system is oscillatory when using the Padula and Visioli's tuning rules?

1.5 Examples for Evaluations of the Tuning Rules

Several examples are presented in this section for evaluation of the tuning rules that are based on the first order plus delay model.

1.5.1 Examples for Evaluating the Tuning Rules

The first example is based on a first order plus delay plant and the second example is based a high order plant so that an approximation is made during the graphic procedure.

Example 1.6 *The unit step response of a continuous time transfer function model*

$$G(s) = \frac{0.5e^{-20s}}{30s+1} \tag{1.55}$$

is shown in Figure 1.15. Instead of using the first order plus delay model directly, we will find the PI controller parameters using the values of τ_M, K_{ss}, and delay d.

Solution. *A Simulink simulator is built to collect the step response testing data and to produce a figure for the step response. On this figure, a line is drawn to reflect the maximum slope of the reaction curve; there are two arrows marking the points of interest. Using MATLAB command ginput(2), with a click on the bottom point, we find the coordinates $t_1 = 21$, $Y_0 = -0.02$; and with a click on the top point, we find $t_2 = 58$, $Y_s = 0.5$.*

From the readings of the two points, we find that

$$K_{ss} = \frac{Y_s - Y_0}{U_s - U_0} \approx 0.5 \tag{1.56}$$

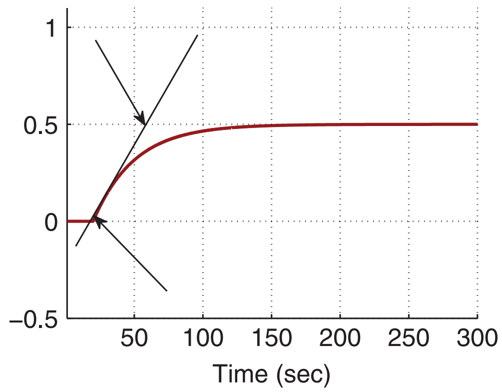

Figure 1.15 Unit step response (Example 1.6)

where $U_s - U_0$ is one since a unit step signal is used as the input. The time delay $d = t_1 = 21$, and the parameter $\tau_M = t_2 - t_1 = 58 - 21 = 37$. With these parameters, we calculate the PI controller parameters using the reaction curve based methods (see Tables 1.2, 1.3, and 1.6). The PI controller parameters are summarized in Table 1.7. Their closed-loop step responses are compared in Figure 1.16.

In reality, instead of a pure first order plus delay dynamics, there are more or less additional dynamics in the system. The tuning rules are applicable for a more complex system. We illustrate how to apply the tuning rules using the example below. This example also illustrates the fact that we need to be cautious when applying the tuning rules and be aware of their limitations.

Table 1.7 PI controller parameters with reaction curve.

	K_c	τ_I
Ziegler–Nichols	3.1714	63
Cohen–Coon	3.3381	32.7131
Wang–Cluett	2.0571	41.4811

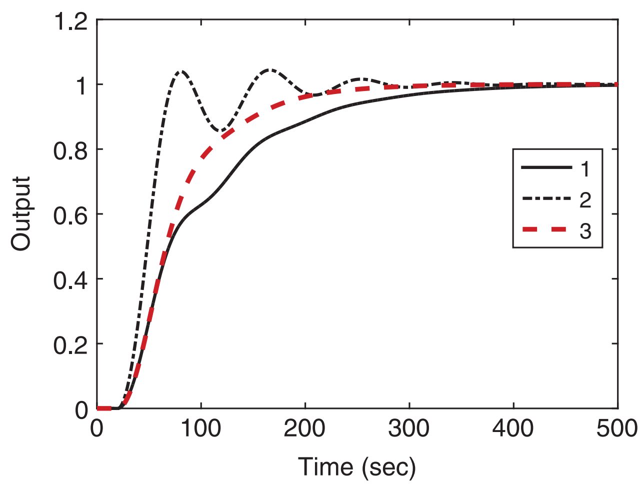

Figure 1.16 Closed-loop unit step response with PI controller (Example 1.6). Key: line (1) Ziegler–Nichols tuning rule; line (2) Cohen–Coon tuning rule; line (3) Wang–Cluett tuning rule.

Example 1.7 *A continuous time plant has the transfer function:*

$$G(s) = \frac{0.5e^{-20s}}{(30s+1)^3} \tag{1.57}$$

The unit step response of this transfer function model is shown in Figure 1.17. In this figure, a line is drawn to reflect the maximum slope of the reaction curve and the points of interest are marked by two arrows.

Using MATLAB command ginput(2), with a click on the bottom point, we will find the coordinates $t_1 = 36, Y_0 = -0.0022 \approx 0$; and a click on the top point, we will find $t_2 = 164, Y_s = 0.4981 \approx 0.5$. Find the PI and PID controllers using the reaction curve based tuning rules.

Solution. *The steady state gain $K_{ss} = \frac{Y_s - Y_0}{1} = 0.5$. The time delay is $d = t_1 = 36$ and the parameter $\tau_M = t_2 - t_1 = 164 - 36 = 128$. The PI controller parameters are calculated using the reaction curve based methods (see Tables 1.2, 1.3, and 1.6) and are summarized in Table 1.8. The closed-loop control systems with the PI controllers are simulated using the plant model (1.57). The unit closed-loop step responses are shown in Figure 1.18. From this figure, we can see that both PI controllers from Ziegler–Nichols and Cohen–Coon tuning rules failed to produce a stable closed-loop system. However, the PI controller using Wang–Cluett tuning rule gives a stable closed-loop system.*

The PI control systems will be subsequently used as examples for closed-loop stability analysis. The Nyquist plots of the PI controllers used in this example are analyzed in Example 2.3 of Chapter 2.

The following example illustrates the applications of the Padula and Visioli tuning rules.

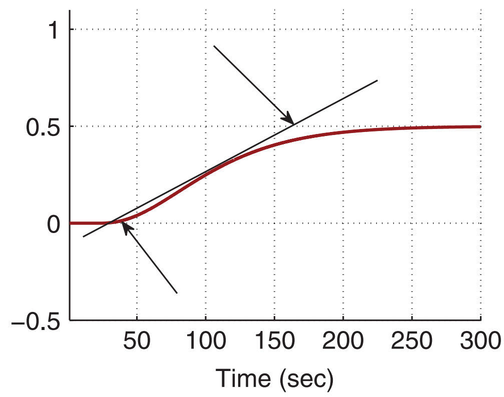

Figure 1.17 Unit step response (Example 1.7).

Table 1.8 PI controller parameters with reaction curve.

	K_c	τ_I
Ziegler–Nichols	6.4	108
Cohen–Coon	6.5667	75.9231
Wang–Cluett	3.8867	127.2154

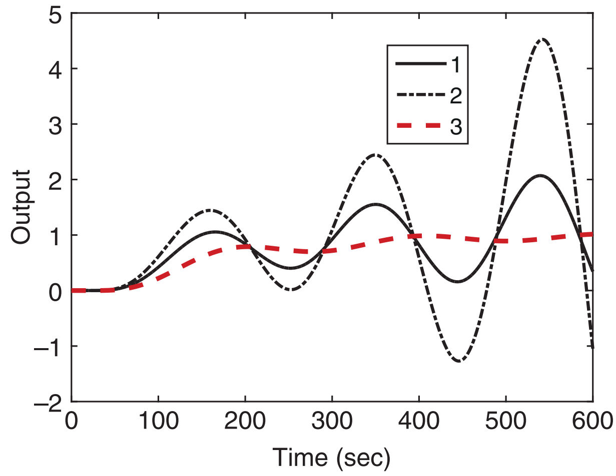

Figure 1.18 Closed-loop unit step response with PI controller (Example 1.7). Key: line (1) Ziegler–Nichols tuning rule; line (2) Cohen–Coon tuning rule; line (3) Wang–Cluett tuning rule.

Example 1.8 *Consider the same third order system with time delay used in Example 1.7. Find the PI and PID controller parameters using Padula and Visioli tuning rules and simulate their closed-loop step response.*

Solution. *The parameters used in the tuning rules are* $K_{ss} = 0.5$, $d = 36$, *and* $\tau_M = 128$. *To evaluate the PI controller performance, Table 1.4 is used to calculate the controller parameters. For* $M_s = 1.4$, *we have* $K_c = 2.3487$ *and* $\tau_I = 84.7649$. *For* $M_s = 2$, *we have* $K_c = 4.5861$ *and* $\tau_I = 86.9015$.

With sampling interval $\Delta t = 1$ *(s), the closed-loop step responses are compared in Figure 1.19, where the IP controller structure is used to reduce overshoot to the step reference signal. It is clearly seen that the closed-loop system with* $M_s = 1.4$ *is stable; however, with* $M_s = 2$ *it is not.*

Now, we evaluate the closed-loop performance for a PID controller with filter, where the filter time constant is chosen to be $0.1\tau_D$. *Based on Table 1.5, the PID controller parameters are calculated as for* $M_s = 1.4$, $K_c = 2.2253$, $\tau_I = 41.8298$, $\tau_D = 25.5365$, *and for* $M_s = 2$, $Kc = 3.54$, $\tau_I = 40.1098$, $\tau_D = 27.0037$.

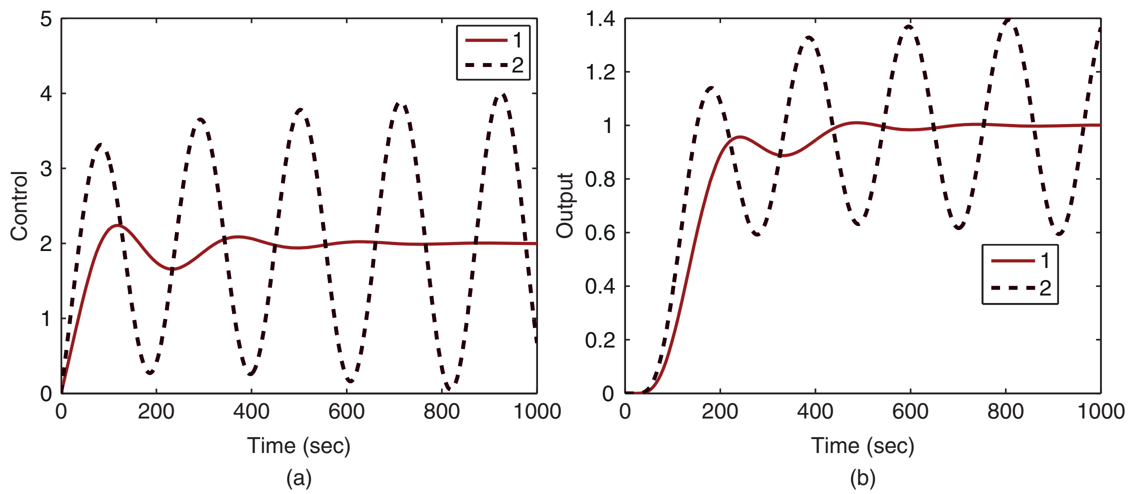

Figure 1.19 Comparison of closed-loop responses using Padula and Visioli PI controller (Example 1.8). (a) Control signal. (b) Output. Key: line (1) tuning rule with $M_s = 1.4$; line (2) tuning rule with $M_s = 2$.

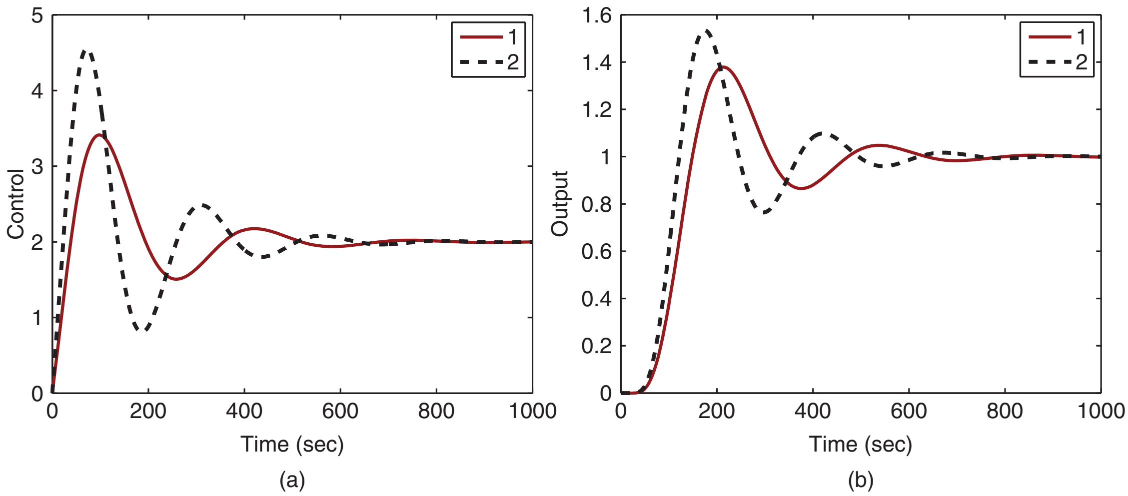

Figure 1.20 Comparison of closed-loop responses using Padula and Visioli PID controller (Example 1.8). (a) Control signal. (b) Output. Key: line (1) tuning rule with $M_s = 1.4$; line (2) tuning rule with $M_s = 2$.

With the same sampling interval $\Delta t = 1$, and both proportional and derivative control on output only (IPD structure) where the derivative filter time constant is selected as $0.1\tau_D$, the closed-loop responses are simulated. Figure 1.20 compares the closed-loop responses. Both tuning rules lead to stable closed-loop control systems. It is seen that there are overshoots in both reference responses, which was caused by the quite large derivative gains.

1.5.2 Fired Heater Control Example

A fired heater using gas fuel is a heating furnace that is typically used for household heating in the winter times. In this case study, the input to the heating furnace is the feed rate of the gas fuel and the output is the heater outlet or the room temperature in a house. Because the temperature sensors are located away from the heating source, there is a time delay in the measured temperature when the input feed rate changes. Additionally, depending on the operating conditions of the input feed rate, the dynamic response of the temperature is different. Two transfer function models are given in Ralhan and Badgwell (2000) to describe the operations of a gas fired heater at a low fuel operation and a high fuel operation. At the low fuel operating condition, the transfer function is described as

$$G_L(s) = \frac{3e^{-10s}}{(4s+1)^2}\frac{degC}{sm^3/s} \tag{1.58}$$

and at the high fuel operating condition,

$$G_H(s) = \frac{e^{-5s}}{(5s+1)^2}\frac{degC}{sm^3/s} \tag{1.59}$$

where the time constant is in minutes. Note that there are dramatically differences in time delay and the steady-state gain of the transfer function models.

In this study, we will introduce the effect of input disturbance in the closed-loop simulation by adding a step signal with negative magnitude to the control signal. This input disturbance represents a sudden change in the process that causes the output temperature drop. The effect of this disturbance is further discussed in Chapter 2.

Example 1.9 *In this example, we will show how to use the tuning rules to find the PID controller parameters for the fired heater at the lower operating condition using the transfer function (1.58) and simulate the closed-loop response with a step reference signal using sampling interval $\Delta t = 5$ (min) and a negative step disturbance entering at the half of the simulation time.*

The higher operating condition case is left as an exercise.

Solution. *Figure 1.21 shows the unit step response with the lines drawn to identify the time delay and time constant for a first order approximation. From the graph, the time delay is found as* 9.54 *min and the time constant* $\tau_M = 23 - 9.54 = 13.48$ *min. With the steady-state gain equal to* 3*, the approximation using first order plus model leads to the following transfer function:*

$$G(s) = \frac{3e^{-9.54s}}{13.48s + 1} \tag{1.60}$$

Now, applying the Ziegler–Nichols tuning rules (see Table 1.2), Cohen–Coon tuning rules (see Table 1.3) and Wang–Cluett tuning rules (see Table 1.6), we obtain the PI controller parameters for the fired heater process shown in Table 1.9. The PI controller parameters obtained are drastically different. The PI controllers using Ziegler–Nichols and Wang–Cluett tuning rules produce stable closed-loop system for the fired heater

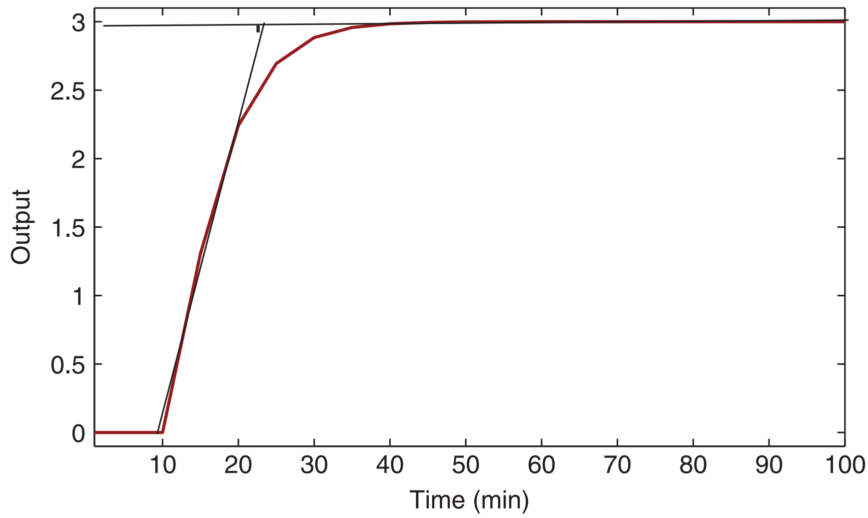

Figure 1.21 Unit step response of the fired heater process.

Table 1.9 PI controller parameters with reaction curve.

	K_c	τ_I
Ziegler–Nichols	0.4239	28.6200
Cohen–Coon	0.4517	13.2353
Wang–Cluett	0.2835	15.7610

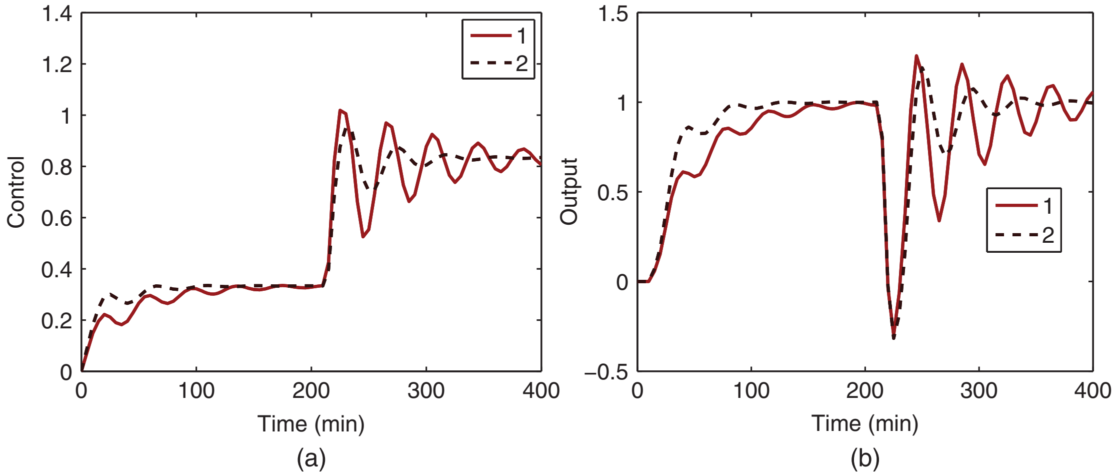

Figure 1.22 Comparison of closed-loop responses using Ziegler–Nichols and Wang–Cluett tuning rules (Example 1.9). (a) Control signal. (b) Output. Key: line (1) Ziegler–Nichols tuning rule; line (2) Wang–Cluett tuning rule.

process, however the PI controller using Cohen–Coon tuning rules does not lead to a stable closed-loop system, which was verified using closed-loop simulation. To evaluate the closed-loop control performance, a unit step input signal is used as a reference and a step input disturbance with magnitude of -0.5 *is added to the closed-loop simulation at half of the simulation time. Figure 1.22(a) shows the control signals generated by the PI controllers and Figure 1.22(b) shows the output responses to the reference change and the disturbance signal. Both closed-loop systems have oscillations, but in comparison, the controller using Wang–Cluett tuning rules leads to a slightly better closed-loop performance with less oscillations.*

1.6 Summary

This chapter has introduced the basics in PID control systems. It is important for us to understand the roles of proportional control, integral control and derivative control. The simple modification with the implementation of PID controller by putting the proportional control on output only reduces the overshoot in the step reference response. For some applications, avoiding the overshoot is important because the requirement is associated with the system's operational constraints while for other applications the overshoot is preferred if the PID controller is used in an inner-loop system (see Chapter 7). Additionally the derivative control should be implemented with a derivative filter to avoid amplification of measurement noise and the derivative control should be implemented on the output only to avoid the scenario of derivative 'kick'.

Several sets of tuning rules have been introduced in this chapter. These tuning rules are very simple and easy to use if the system can be approximated by a first-order-plus delay model. However, they offer no guarantees on the closed-loop performance as demonstrated by the simulation examples. Some of the examples used in this chapter will be analyzed using the Nyquist stability criterion and sensitivity functions in Chapter 2.

1.7 Further Reading

1. Text books in control engineering include, Franklin et al. (1998), Franklin et al. (1991), Ogata (2002), Golnaraghi and Kuo (2010), Goodwin et al. (2000) and Astrom and Murray (2008).
2. Process control books include Marlin (1995), Ogunnaike and Ray (1994), Seborg et al. (2010).
3. There are many books published on PID control, including Astrom and Hagglund (1995), Astrom and Hagglund (2006), Yu (2006), Johnson and Moradi (2005), Visioli (2006), Tan et al. (2012). PID control of multivariable systems is discussed in Wang et al. (2008).
4. Tuning rules are compiled as a book (O'Dwyer (2009)). PID controller tuning rules with performance specifications derived from frequency response analysis are introduced in Wang and Cluett (2000).
5. The survey and tutorial papers on PID control include Åström and Hägglund (2001), Ang et al. (2005), Li et al. (2006), Knospe (2006), Cominos and Munro (2002) and Visioli (2012), Blevins (2012).
6. A web-based laboratory for teaching PID control is introduced in Ko et al. (2001), Yeung and Huang (2003).
7. The issues associated with derivative filters in PID controller are addressed in Luyben (2001), Hägglund (2012), Hägglund (2013), Isaksson and Graebe (2002), Larsson and Hägglund (2011).
8. Improving reference tracking and reducing overshoot is discussed for an existing controller in an industrial environment (Visioli and Piazzi (2003)).
9. Ziegler-Nichols tuning formula is refined in Hang et al. (1991) and for unstable systems with time delay in De Paor and O'Malley (1989). The ZieglerNichols step response method is revisited from the point of view of robust loop shaping (Åström and Hägglund (2004)). There is a set of tuning rules for integrating plus delay model in Tyreus and Luyben (1992), which was extended to PID controllers in Luyben (1996).