Iván D. Díaz-Rodríguez · Sangjin Han · Shankar P. Bhattacharyya

Analytical Design of PID Controllers

Springer

Analytical Design of PID Controllers

Iván D. Díaz-Rodríguez ·
Sangjin Han · Shankar P. Bhattacharyya

Analytical Design of PID Controllers

Springer

Iván D. Díaz-Rodríguez
McAllen Higher Education Center
Texas A&M University
Mcallen, TX, USA

Sangjin Han
Electrical and Computer Engineering
Texas A&M University
College Station, TX, USA

Shankar P. Bhattacharyya
Electrical and Computer Engineering
Texas A&M University
College Station, TX, USA

ISBN 978-3-030-18227-4 ISBN 978-3-030-18228-1 (eBook)
https://doi.org/10.1007/978-3-030-18228-1

MATLAB® is a registered trademark of The MathWorks, Inc., 1 Apple Hill Drive, Natick, MA 01760-2098, USA, http://www.mathworks.com.

Mathematics Subject Classification (2010): 9, 93C05, 93C35, 93C55, 93C80, 93B36, 93B50, 93B51, 93B52, 93D09, 93D15, 93D25

This Springer imprint is published by the registered company Springer Nature Switzerland AG.
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Dedicated to Prof. J. B. Pearson (1930–2012).

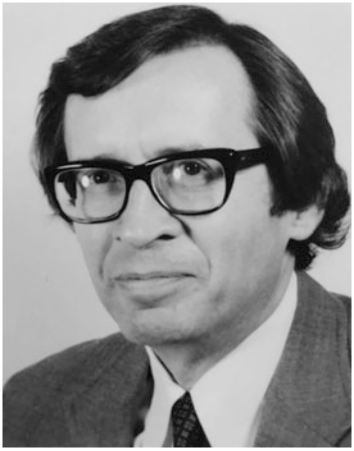


Photo courtesy of Jane Pearson

Preface

The Proportional–Integral–Derivative (PID) controller dominates the control industry across the traditional fields of Aerospace, Electrical, Mechanical, and Chemical Engineering as well as emerging fields such as driverless cars, autonomous robots, and unmanned aerial vehicles. Indeed, they account for 99% of all the controllers in use in the world.

This universal presence of PID controllers in applications contrasts sharply with the relative lack of interest in them from the control theory community which, until recently, was mainly focused on designing high-order optimal controllers by state-space methods. This situation began to change in 1997 when it was demonstrated that high-order controllers rendered the closed-loop system dysfunctionally fragile with respect to controller parameters even if they were robust with respect to plant parameters. This helped to usher in a renewed interest in low-order controllers.

The PID controllers are the simplest of low-order controller structures providing servo and disturbance rejection capabilities provided closed-loop stability can be achieved. In the last 20 years, significant progress has been made in computing the complete set of stabilizing PID controllers for linear time-invariant continuous- and discrete-time plants of arbitrary order. These were reported in the monographs "Structure and Synthesis of PID Controllers" by A. Datta, Ming-Tzu Ho, and S. P. Bhattacharyya and "PID Controllers for Time-Delay Systems" by Guillermo Silva, A. Datta, and S. P. Bhattacharyya. The present monograph is the third in this sequence.

The main results presented here demonstrate how multi-objective designs can be carried out using PID controllers, by exploiting the availability of the stabilizing set. By superimposing gain margin, phase margin, H_∞, and time domain specifications calculated in terms of design parameters, on the stabilizing set, in a systematic and constructive manner one can effectively execute hitherto impossible multi-objective designs and determine the limits of achievable performance. The results are presented here for continuous-time and discrete-time systems in a unified and self-contained manner and are illustrated by examples. A recent extension of these results to multivariable systems is also given.

We expect the concepts and design methods presented here to be useful in engineering and other applications and to further drive research on PID controllers into Adaptive Control, Machine Learning, Computer Science, Biological Systems, and other areas.

The results given here would not have been possible without the support and collaboration of numerous colleagues. Especially, we would like to express our gratitude to L. H. Keel, A. Datta, Ming-Tzu Ho, Guillermo Silva, and Navid Mohsenizadeh for their contributions.

Iván D. Díaz Rodríguez would like to thank his beloved parents for their love, support, encouragement, and sacrifices. Sangjin Han would like to thank his parents and brother.

College Station, TX, USA
February 2019

Iván D. Díaz-Rodríguez
Sangjin Han
Shankar P. Bhattacharyya

Contents

Chapter 1
Introduction to Control

Abstract In this chapter, we describe control systems informally, emphasizing the key elements of tracking, disturbance rejection, stability, and robustness. Next, we show why integral control driven by tracking error provides the correct feedback architecture to try to achieve these goals. This leads naturally to the Proportional–Integral–Derivative (PID) controller structure, where the proportional, integral, and derivative gains k_p, k_i, and k_d now become the design parameters which need to be tuned to achieve robust stability and time domain response specifications. A brief description is given of some classical and existing tuning approaches. We conclude the chapter with an examination of why optimal control, and in particular quadratic optimization, is absent from PID design theory and show that the reason lies in the inherent fragility of the high-order controllers invariably produced by optimization. The contents of this chapter should serve as background, perspective, and motivation for the rest of the book.

1.1 Introduction

Control theory and control engineering deal with a variety of dynamic systems such as aircraft, spacecraft, ships, trains, automobiles, and robots. They also deal with industrial processes such as distillation columns and rolling mills, electrical systems such as motors, generators, and power systems. Nowadays, they are ubiquitous in biomedical applications, power electronics, driverless cars, and autonomous robots.

In each case, the *setting* of the control problem is represented by the following elements:

Sections 1.1, 1.2, 1.4 and 1.5.4 are reproduced from S. P. Bhattacharyya, A. Datta, L. H. Keel, *Linear System Theory: Structure, Robustness, and Optimization. Taylor & Francis LLC Books,* with permission © 2008 Taylor & Francis LLC Books.

I. D. Díaz-Rodríguez et al., *Analytical Design of PID Controllers*,
https://doi.org/10.1007/978-3-030-18228-1_1

1. There are dependent variables, called *outputs*, to be controlled, which must be made to behave in a prescribed way. For instance, it may be necessary to *assign* the temperature and pressure at various points in a process, or the position and velocity of a vehicle, or the voltage and frequency in a power system, to given desired fixed values, despite uncontrolled and unknown variations at other points in the system.
2. Specific independent variables, called *inputs*, such as a voltage applied to the motor terminals, or valve position, are available to regulate and to control the behavior of the system. Other dependent variables, such as position, velocity, or temperature, are accessible as dynamic *measurements* on the system.
3. There are unknown and unpredictable *disturbances* impacting the system. These disturbances could be, for example, the fluctuations of a load in a power system, disturbances such as wind gusts acting on a vehicle, external weather conditions acting on an air conditioning plant, or the fluctuating load torque on an elevator motor, as passengers enter and exit.
4. The equations describing the plant dynamics, and the parameters contained in these equations, are not known at all or are known imprecisely. This uncertainty can arise even when the physical laws and equations governing a process are known well, for instance, because these equations were obtained by linearizing a nonlinear system about an operating point. As the operating point changes so do the system parameters.

The previous considerations suggest the following general representation of the *plant* or system to be controlled. In Fig. 1.1, the inputs or outputs shown could be representing a vector of signals. In such cases, the plant is said to be a *multivariable plant* as opposed to the case where the signals are scalar, in which case the plant is said to be a *scalar or monovariable plant*.

Control is exercised by feedback, which means that a device, driven by the available measurements, generates the corrective control input to the plant. Thus, the *feedback* or *closed-loop system* in Fig. 1.2 represents the controlled system.

The control design problem is the problem of determining the characteristics of the controller so that the controlled outputs can be

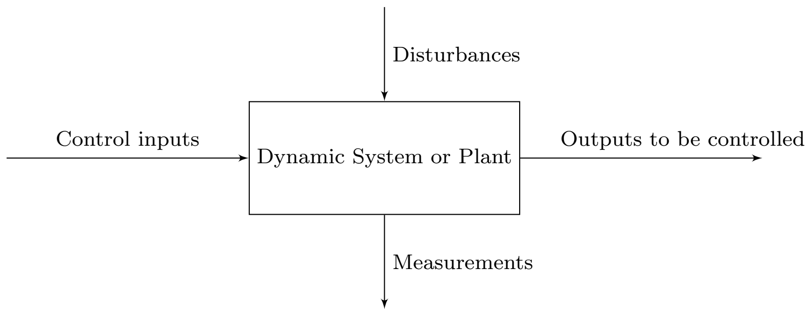


Fig. 1.1 A general plant. © Taylor & Francis LLC Books. Reproduced from [8] with permission

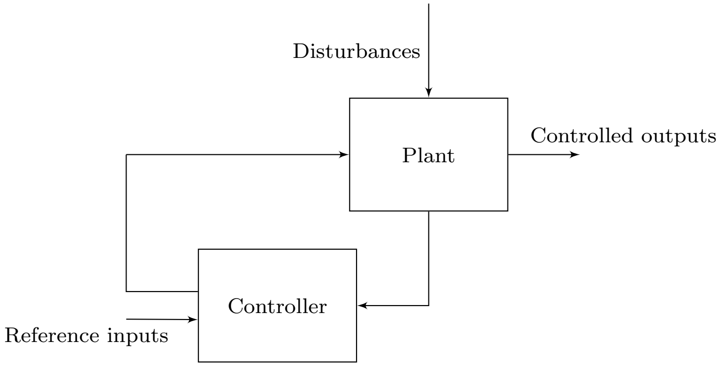


Fig. 1.2 A feedback control system. © Taylor & Francis LLC Books. Reproduced from [8] with permission

1. Set to prescribed values called *references*;
2. Maintained at the reference values despite the unknown disturbances;
3. Conditions (1) and (2) are met despite the inherent uncertainties and changes in the plant dynamic characteristics.

The first requirement above is called *tracking*. The second is called *disturbance rejection*. The third condition is called the *robustness* of the system. The simultaneous satisfaction of (1), (2), and (3) is called *robust tracking and disturbance rejection*, and the control systems designed to achieve this are called *servomechanisms*.

In the next section, we discuss how integral control is useful in the design of servomechanisms.

1.2 The Magic of Integral Control

Integral control is used almost universally in the control industry to design robust servomechanisms. Computer control most easily implements integral action. It turns out that hydraulic, pneumatic, electronic, and mechanical integrators are also commonly used elements in control systems. In this section, we explain how integral control works in general to achieve robust tracking and disturbance rejection.

Let us first consider an integrator as shown in Fig. 1.3.

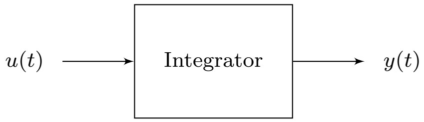


Fig. 1.3 An integrator. © Taylor & Francis LLC Books. Reproduced from [8] with permission

The input–output relationship is

$$y(t) = K \int_0^t u(\tau)d\tau + y(0) \tag{1.1}$$

or, in differential form,

$$\frac{dy(t)}{dt} = Ku(t), \tag{1.2}$$

where K is a nonzero real number called the integrator gain.

Now *suppose* that the output $y(t)$ is a *constant* for a segment of time $[t_1, t_2]$. It follows from (1.2) that

$$\frac{dy(t)}{dt} = 0 = Ku(t) \text{ for } t \in [t_1, t_2]. \tag{1.3}$$

Equation (1.3) proves the following essential facts about the operation of an integrator:

Fact 1 *If the output of an integrator is* constant *over a segment of time, then the input must be identically* zero *over that same segment.*

Fact 2 *The output of an integrator changes as long as the input is nonzero.*

The simple facts stated above suggest how an integrator can be used to solve the servomechanism problem. If a plant output $y(t)$ is to track a constant reference value r, despite the presence of unknown constant disturbances, it is enough to

A. attach an integrator to the plant and make the error

$$e(t) = r - y(t) \tag{1.4}$$

the input to the integrator;

B. ensure that the closed-loop system is asymptotically stable so that under constant reference and disturbance inputs, all signals, including the integrator output, reach constant steady-state values.

This structure is depicted in the block diagram shown in Fig. 1.4.

If the feedback system, shown in Fig. 1.4, is asymptotically stable, and the inputs r and d (disturbances) are constant, it follows that all signals in the closed loop will converge to constant values. In particular, the integrator output $v(t)$ tends to a constant value. Therefore, by the fundamental fact about the operation of an integrator established in Fact 1 above, it follows that the integrator input tends to *zero*. Since we have arranged that this input is the tracking error, it follows that $e(t) = r - y(t)$ goes to zero and hence $y(t)$ tracks r as $t \to \infty$.

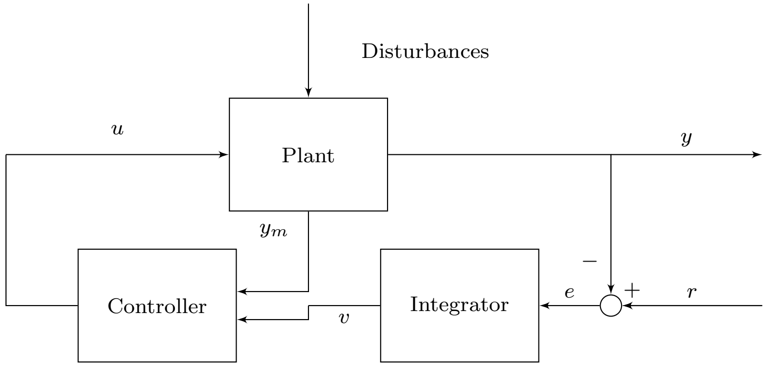


Fig. 1.4 Servomechanism. © Taylor & Francis LLC Books. Reproduced from [8] with permission

We emphasize that the steady-state tracking property established above is *very robust*. It holds as long as the closed loop is asymptotically stable and is (1) independent of the particular values of the constant disturbances or references, (2) independent of the initial conditions of the plant and controller, and (3) independent of whether the plant and controller are linear or nonlinear. Thus, the tracking problem is reduced to guaranteeing that stability is assured. In many practical systems, stability of the closed-loop system can even be ensured without detailed and exact knowledge of the plant characteristics and parameters; this is known as *robust stability*.

We next discuss how several plant's outputs $y_1, y_2, \ldots, y_m$ can track prescribed but arbitrary constant reference values $r_1, r_2, \ldots, r_m$ in the presence of unknown but constant disturbances $d_1, d_2, \ldots, d_q$. The previous argument can be extended to this multivariable case by attaching m integrators to the plant and driving each integrator with its corresponding error input $e_i = r_i - y_i$, $i = 1, \ldots, m$. This is shown in the configuration in Fig. 1.5. This requires the existence of $u_1, u_2, \ldots, u_r$ that makes $y_i = r_i, i = 1, \ldots, m$ for arbitrary $r_i, i = 1, \ldots, m$. Therefore, the plant's equations relating $y_i, i = 1, \ldots, m$ to $u_j, j = 1, \ldots, r$ must be invertible for constant inputs. In the case of Linear Time-Invariant (LTI) systems, this is equivalent to the requirement that the corresponding transfer matrix be right invertible or equivalently possess rank equal to m at $s = 0$. Sometimes, this is restated as two conditions: (1) $r \geq m$ or at least as many control inputs as outputs to be controlled and (2) $G(s)$ has no transmission zero at $s = 0$. The architecture of the block diagram of Fig. 1.5 is easily modified to handle servomechanism problems for more general classes of reference and disturbance signals such as ramps or sinusoids of a specified frequency. The only modification required is to replace the integrators by the corresponding signal generators of these external signals.

In general, the addition of an integrator to the plant tends to make the system less stable. This is because the integrator is inherently an unstable device; for instance, its response to a step input, a bounded signal, is a ramp, an unbounded signal. Therefore,

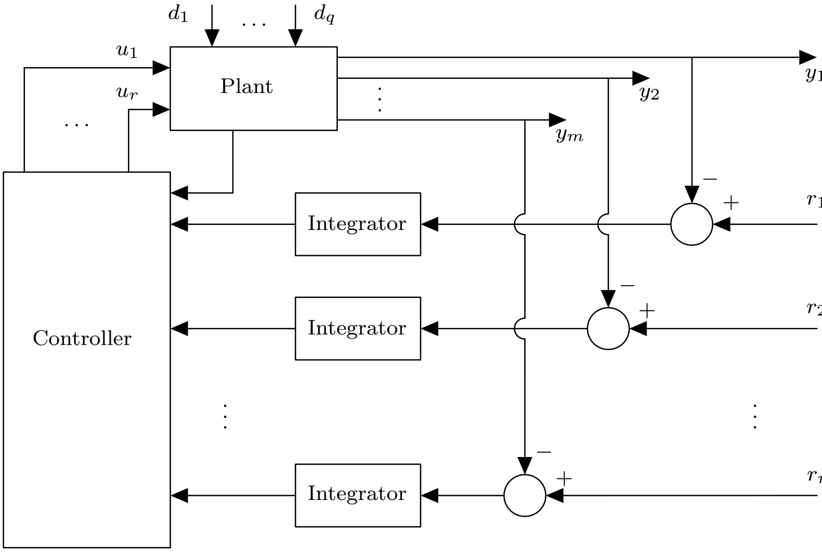


Fig. 1.5 Multivariable servomechanism. © Taylor & Francis LLC Books. Reproduced from [8] with permission

the problem of stabilizing the closed loop becomes a critical issue even when the stand-alone plant is stable.

Since the integral action and thus the attainment of zero steady-state error is *independent* of the particular value of the integrator gain K, we can see that this gain can be adjusted to try to stabilize the system. This single degree of freedom is sometimes insufficient for attaining stability and acceptable transient response, and additional gains are introduced as explained in later sections. The addition of gains naturally leads to the PID controller structure commonly used in industry.

1.3 Overview of PID Controller Design Approaches

PID controllers are the most widely used controllers in the control industry in motion control, process control, power electronics, hydraulics, pneumatics, and manufacturing. In fact, in process control, more than 95% of the control loops are of PID type with most loops using PI control. Their popularity is due to their simple structure, easy implementation, and straightforward maintenance. Also, they provide satisfactory performance with a cost/benefits ratio that is hard for other types of controllers to match. For this same reason, they are also popular in modern applications such as driverless cars, unmanned aerial vehicles, and autonomous robots.

1.3.1 PID Controller Structure

The PID controller is the name given to a controller which consists of the addition of three control actions (see Fig. 1.6). These actions are an action proportional to the error, an action proportional to the integral of the error, and an action proportional to the first derivative of the error in (1.4).

- **Proportional (P) controller**. The proportional action deals with the present values of the error signal; it is proportional to the size of the process error signal increasing the magnitude of the control variable when the error signal increases. When using only a P controller, we notice that increasing the proportional gain k_p may in general speed up the time response. However, it is possible that a steady-state error will occur. In general, under proportional control the steady-state error is zero if and only if k_p is very large.
- **Integral (I) controller**. The integral action is used to reduce the steady-state error to zero. When using an integral gain, increasing the value of k_i can give a broad range of response types in addition to the elimination of the offset error. The control signal is

$$u(t) = k_i \int e(t)dt. \tag{1.5}$$

  The integral of the error $e(t)$ is proportional to the area under the error curve. The control signal u will continuously change depending on whether the error signal is positive or negative. If the control signal $u(t)$ is constant, then the error signal must be identically zero, as expressed in Fact 1 in Sect. 1.2.
- **Derivative (D) controller**. The derivative action is often used to improve damping and closed-loop stability. It deals with the possible future values of the error signal based on its current rate of change, anticipating the trend of the error. The control signal here is

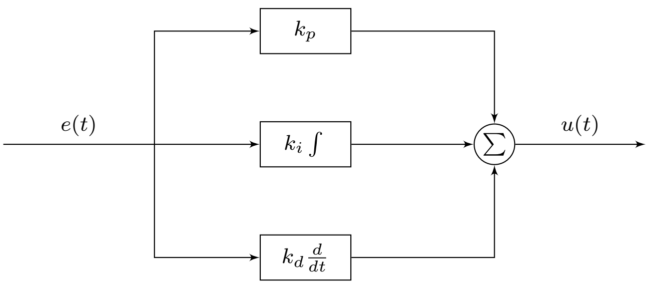


Fig. 1.6 PID controller block diagram

$$u(t) = k_d \frac{de(t)}{dt}. \tag{1.6}$$

The derivative part is proportional to the predicted error.

1.3.2 PID Controller Representations

Commonly used PID controller structures are of parallel and series types.

- **Parallel type**. This controller type has the following control law:

$$u(t) = K_c \left(e(t) + \frac{1}{T_i} \int e(t)dt + T_d \frac{de(t)}{dt} \right) \tag{1.7}$$

  where $k_p = K_c$ is the proportional gain, T_i is the integral time of the controller with $k_i = \frac{K_c}{T_i}$, and T_d is the derivative time of the controller with $k_d = K_c T_d$. This representation is known as *ideal*.
- **Series type**. This controller type has the following control law:

$$\begin{aligned} e_1(t) &= e(t) + T_d \frac{de(t)}{dt}, \\ u(t) &= K_c \left(e_1(t) + \frac{1}{T_i} \int e(t)dt \right). \end{aligned} \tag{1.8}$$

  In this case, all three portions of this PID structure are affected by the gain K_c. However, the proportional term is also affected by the values of the integral and derivative tuning parameters T_d and T_i. Therefore, adjusting T_i affects both the I and P actions, adjusting T_d affects both the D and P actions, and adjusting K_c affects all three actions.

1.3.3 Classical PID Controller Tuning

Due to the popularity of PID controllers in industry and their widespread use, there exist many approaches for their design and implementation developed over the years. The classical methods found in the literature can be classified as follows.

- **Trial and Error Method**. This method is applied when there is no systematic approach to follow when designing the controller. The method is based on experience about the effects of adjusting the individual k_p, k_i, k_d gains trying to get a better time response regarding speed and closed-loop stability. The typical effects of increasing each gain are represented in a table as in Table 1.1.

Table 1.1 Effects of adjusting individual PID gains on the system

Parameter	Steady-state error	Speed	Stability
k_p	Reduces	Increases	Decreases
k_i	Eliminates	Reduces	Increases
k_d	No effect	Increases	Increases

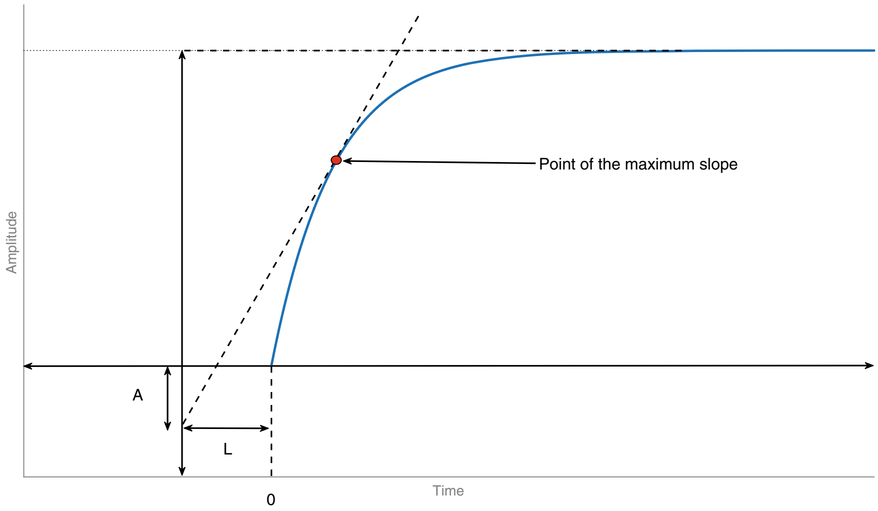


Fig. 1.7 Ziegler–Nichols step response method

The advantage of the trial and error method is that it does not require any mathematical model or mathematical derivation. However, it requires some experience to adequately adjust the controller gains to satisfy the desired performance regarding the speed of response and stability margins.

- **The Ziegler–Nichols step response method**. This PID tuning method was developed between 1941 and 1942 at the Taylor Instrument Company, USA. Since that time, this method has been extensively used in its original form and with some variations. The method is based on the measured step response of the open-loop stable system. For instance, see Fig. 1.7. The procedure is the following:

1. Calculate or determine experimentally the step response of the open-loop system.
2. Draw a tangent line with the maximum slope possible from the step response, see Fig. 1.7.
3. Calculate L, which is the distance from the intersection of the slope and vertical axis to the starting point of the step response.
4. Calculate A, which is the distance from the intersection of the slope and the vertical axis to the horizontal axis.

5. Compute the PID gains from the following formulas:

$$\begin{aligned} k_p &= \frac{1.2}{A}, \\ k_i &= \frac{0.6}{AL}, \\ k_d &= \frac{0.6L}{A}. \end{aligned} \tag{1.9}$$

- **The Ziegler–Nichols frequency response method**. This PID tuning method considers a proportional controller attached to the system in a closed-loop configuration. The objective is to find the ultimate frequency where the phase of the process is $-180°$. That is the ultimate gain, where the system reaches the stability boundary. The tuning procedure is the following.

1. Connect a proportional controller to the system in a closed-loop configuration.
2. Slowly increase the proportional gain until the output starts oscillating. This gain is called ultimate gain K_u.
3. Measure the period of the oscillation in the output. This period is called the ultimate period T_u.
4. Compute the PID gains from the following formulas:

$$\begin{aligned} k_p &= 0.6K_u. \\ k_i &= \frac{1.2K_u}{T_u}. \\ k_d &= 0.075K_u T_u. \end{aligned} \tag{1.10}$$

This tuning method is capable of finding the PID controller gains for the system. However, it requires some experience and skill because the system is taken to its limits of instability and it may become very close to getting damaged. It is emphasized that the Ziegler–Nichols design procedure assumes that the plant is of first-order cascaded with a delay.

- **Relay Tuning Method**. This PID tuning method was developed by K. Åström and T. Hägglund as an alternative to the Ziegler–Nichols frequency response PID tuning method. This method is very similar to the Ziegler–Nichols method, but instead of increasing a proportional gain until the system's output oscillates, a relay is used to generate an oscillation in the output, see Figs. 1.8 and 1.9. The relay connected to the system generates a square wave signal with specific amplitude and frequency. Then, a signal in the output approximated to a sinusoid is generated. The tuning procedure is the following:

1. The system should be working at the operating point.
2. Set the amplitude of the square signal in the relay.
3. Calculate the ultimate period T_u, see Fig. 1.9.

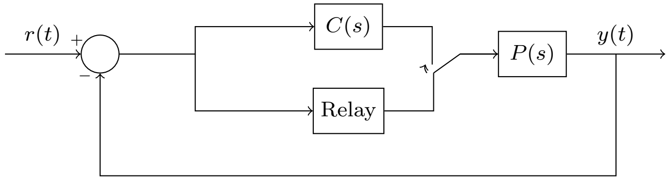


Fig. 1.8 Unity feedback block diagram with a relay

4. Calculate the controller parameters k_p, k_i, and k_d using the Ziegler–Nichols table using $K_u = K_e$, where $K_e = A_u/A_e$. Where $A_u = 4A/\pi$ and $A_e = E$ with E being the amplitude of the oscillations in the control error signal.

The advantage of this method is that it does not require one to force the system to be close to instability. Therefore, it keeps the system safer and reduces the possibility of damage. Also, this relay method can be automated since the output oscillation amplitude is proportional to the amplitude of the relay signal.

- **The Cohen-Coon Method**. This is an open-loop PID tuning method, which follows the same procedure as the Ziegler–Nichols step response method. In Fig. 1.10, we show the step response of the open-loop system for which the parameters k_p, L, and T can be determined. The gain k_p is determined by taking the ratio between the amplitude increment of the output and the increase in the control signal. That is

$$k_p = \frac{\Delta y}{\Delta u}. \tag{1.11}$$

The variables L and T can be found from Fig. 1.10, which represents the step response. Then, considering the PID controller of parallel type as in (1.7), the Cohen-Coon method recommends the following formulas to calculate the PID gains:

$$\begin{aligned} K_c &= \frac{1}{k_p}\left(0.25 + \frac{1.35T}{L}\right), \\ T_i &= \frac{2.5 + \frac{0.46L}{T}}{1 + \frac{0.61L}{T}} L, \\ T_d &= \frac{0.37}{1 + \frac{0.19L}{T}} L. \end{aligned} \tag{1.12}$$

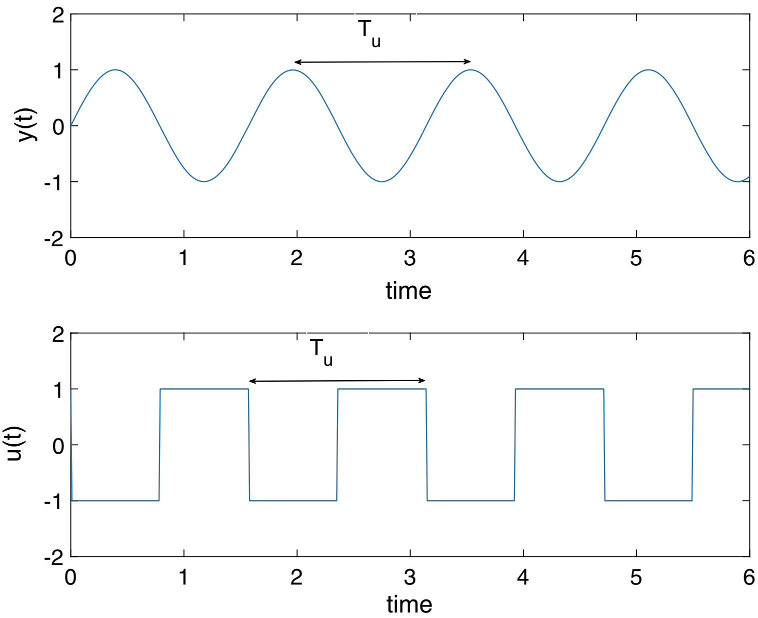


Fig. 1.9 Output oscillation and relay signals

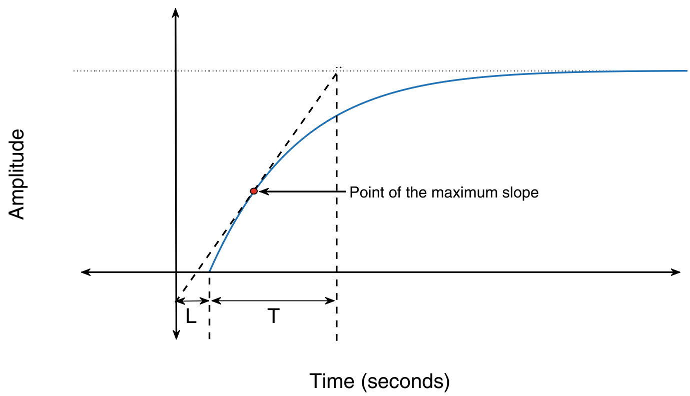


Fig. 1.10 Cohen-Coon method

1.3.4 PID Controller Tuning Methods

After the appearance of the classical PID controller tuning techniques, the complexity of the systems and performance demands from the control designer made necessary the development of new tuning design techniques. Over the years, many useful results were developed toward PID tuning methods for more performance- specific require-

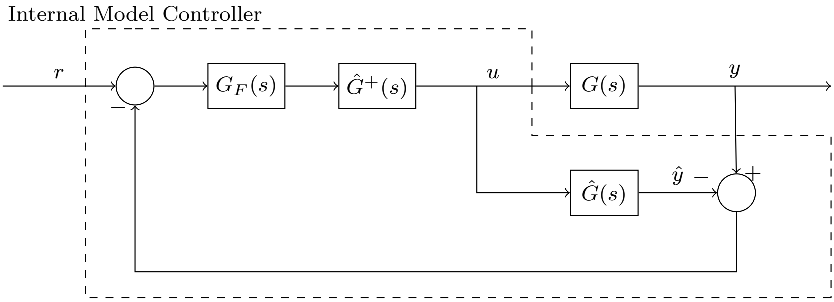


Fig. 1.11 Closed-loop system block diagram with internal model controller

ments and to deal with more complex systems. Some of these approaches are the following:

- **Internal Model Control design**
  This controller approach considers stable systems. Consider the closed-loop system block diagram presented in Fig. 1.11. Where $\hat{G}(s)$ is an approximation of the system $G(s)$, $G_F(s)$ is a low pass filter, and $\hat{G}^{+}(s)$ is the inverse of $\hat{G}(s)$. Then, the controller design objective is to cancel the poles and zeros of the original system $G(s)$ by connecting it in parallel with $\hat{G}(s)$. This approach is called internal model control because the controller contains a model of the system internally. The purpose of $G_F(s)$ is to make the system less sensitive to modeling errors. The controller $C(s)$ is given by

$$C(s) = \frac{G_F(s)\hat{G}^{+}(s)}{1 - G_F(s)\hat{G}^{+}(s)\hat{G}(s)}. \tag{1.13}$$

  Consider the case when this approach is applied to PI and PID controllers. For the case of plants which are first-order plus time-delay systems, we have that

$$P(s) = \frac{K}{1+sT}e^{-sL}. \tag{1.14}$$

$$\hat{G}^{+}(s) = \frac{1+sT}{K}. \tag{1.15}$$

$$G_F(s) = \frac{1}{1+sT_f}. \tag{1.16}$$

  Then, by a first-order Padé approximation for the time delay

$$e^{-sL} \approx \frac{1-sL/2}{1+sL/2}, \tag{1.17}$$

we have the controller of the PID form

$$C(s) = \frac{(1+sL/2)(1+sT)}{Ks(L+T_f+sT_fL/2)} \approx \frac{(1+sL/2)(1+sT)}{Ks(L+T_f)} = \frac{k_d s^2 + k_p s + k_i}{s}, \tag{1.18}$$

where

$$k_d = \frac{LT}{2K(L+T_f)}, \tag{1.19}$$

$$k_p = \frac{(L+2T)}{2K(L+T_f)}, \tag{1.20}$$

$$k_i = \frac{1}{K(L+T_f)}. \tag{1.21}$$

- **Pole Placement Design**
  Pole placement is a controller design method based on the knowledge of the system's transfer function, where the objective is to determine the closed-loop pole locations on the complex plane by setting the controller gains. It is known that the system's closed-loop pole locations influence the behavior of the system. Therefore, the designer can apply this method to place the locations of the poles for a desirable behavior of the closed-loop system.
  PI and PID controllers can be used with a pole placement design as long as the plant transfer function system is of the first or second order. For higher order systems, one way to use a PI or PID controller is to approximate the system's transfer function by a first- or second- order transfer function.
  For the first-order case, the system can be described by

$$P(s) = \frac{K}{1+Ts}, \tag{1.22}$$

where K is the system's gain and T is the time constant. Using a PI controller, we have

$$C(s) = K_c\left(1 + \frac{1}{T_i s}\right), \tag{1.23}$$

where K_c is the controller gain and T_i the integral time. The closed-loop transfer function is

$$G(s) = \frac{C(s)P(s)}{1+C(s)P(s)}. \tag{1.24}$$

The characteristic equation becomes of second order

$$\delta(s) = s^2 + \left(\frac{1 + KK_c}{T}\right)s + \left(\frac{KK_c}{TT_i}\right). \tag{1.25}$$

A second-order characteristic equation can be represented in terms of the relative damping ζ and the natural frequency ω_n as

$$\delta(s) = s^2 + 2\zeta\omega_n s + \omega_n^2, \tag{1.26}$$

where the parameters ζ and ω_n determine the time response of the second-order system.
Comparing (1.25) and (1.26) we must have

$$K_c = \frac{2\zeta\omega_n T - 1}{K}. \tag{1.27}$$

$$T_i = \frac{2\zeta\omega_n T - 1}{\omega_n^2 T}. \tag{1.28}$$

For a second-order plant without zeros, the plant can be described by

$$P(s) = \frac{K}{(1 + T_1 s)(1 + T_2 s)}. \tag{1.29}$$

Using the PID controller

$$C(s) = \frac{K_c\left(1 + T_i s + T_i T_d s^2\right)}{T_i s}, \tag{1.30}$$

the characteristic equation becomes of third order

$$\delta(s) = s^3 + \left(\frac{1}{T_i} + \frac{1}{T_2} + \frac{KK_cT_d}{T_1T_2}\right)s^2 + \left(\frac{1}{T_1T_2} + \frac{KK_c}{T_1T_2}\right)s + \frac{KK_c}{T_1T_2T_i}. \tag{1.31}$$

A third-order characteristic equation can also be represented in terms of the relative damping ζ and the natural frequency ω_n as

$$\delta(s) = (s + \alpha\omega_n)(s^2 + 2\zeta\omega_n s + \omega_n^2). \tag{1.32}$$

Combining (1.31) and (1.32) we have

Table 1.2 Cohen-Coon formulae for dominant pole placement controller design

Controller	K_c	T_i	T_d
PI	$\frac{0.9}{a}\left(1+\frac{0.92\tau}{1-\tau}\right)$	$\frac{3.3-3.0\tau}{1+1.2\tau}L$	
PID	$\frac{1.35}{a}\left(1+\frac{0.18\tau}{1-\tau}\right)$	$\frac{2.5-2.0\tau}{1-0.39\tau}L$	$\frac{0.37-0.37\tau}{1-0.81\tau}L$

$$K_c = \frac{T_1T_2\omega_n^2(1+2\alpha\zeta)-1}{K}. \tag{1.33}$$

$$T_i = \frac{T_1T_2\omega_n^2(1+2\alpha\zeta)-1}{T_1T_2\alpha\omega_n^3}. \tag{1.34}$$

$$T_d = \frac{T_1T_2\omega_n(\alpha+2\zeta)-T_1-T_2}{T_1T_2\omega_n^2(1+2\alpha\zeta)-1}. \tag{1.35}$$

- **Dominant Pole Placement Design**
  This controller design approach follows the same idea of the previous pole placement design. However, this method is focused on higher order systems. The objective is to select a pair of dominant poles, which have more influence on the behavior of the system time response than the rest of the closed-loop poles.
  For PI and PID controllers design using a dominant pole placement method, there is an approach developed by Cohen-Coon for first-order plus time-delay systems such as the one shown in (1.14). The central design criterion is the rejection of load disturbances by placing the dominant poles that give a quarter amplitude decay ratio in the time response. For PID controllers, two complex dominant poles and one real pole are placed to satisfy the quarter amplitude decay ratio in the time response. The following table presents some formulae to calculate the PI and PID controller gains (Table 1.2),
  where

$$a = \frac{KL}{T}. \tag{1.36}$$

$$\tau = \frac{L}{(L+T)}. \tag{1.37}$$

- **Time Domain Optimization Methods**
  In time domain optimization methods, the controller gains are calculated based on numerical optimization methods where an objective function is specified. For PID controllers, an objective function is defined by one of the forms

$$J(\theta) = \int_0^\infty t|e(\theta,t)|dt, \tag{1.38}$$

$$J(\theta) = \int_0^\infty |e(\theta,t)|dt, \tag{1.39}$$

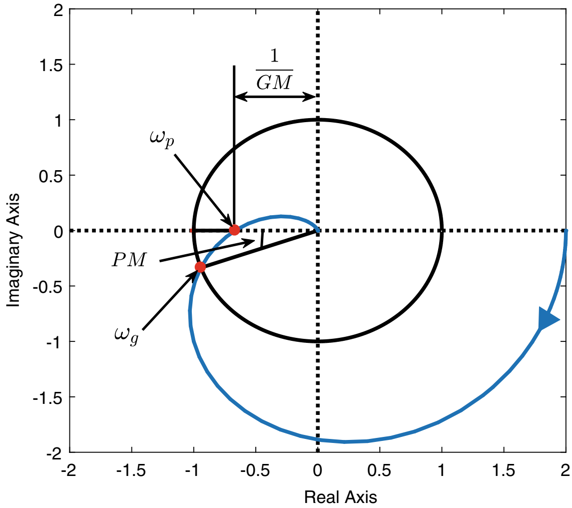
$$J(\theta) = \int_0^\infty e(\theta, t)^2 dt, \tag{1.40}$$

where θ represents a vector with the PID gains and $e(\theta, t)$ is the error signal of the control system. The objective function in (1.38) is called integral time-weighted absolute error (ITAE); this function integrates the absolute error multiplied by time as a weight. The objective function (1.39) is called integral absolute error (IAE). This function integrates the absolute error without weights. The objective function (1.40) is called integral square error (ISE), which only integrates the square of the error.

The parameters of the controller are obtained after minimizing a selected objective function to obtain a better performance of the closed-loop system .

- **Gain and Phase Margin Based Design**
  Gain and phase margins can indicate how stable the system is. These margins are calculated from the open-loop system to determine how robust the closed-loop system is. The gain margin is the amount of gain necessary to make the system unstable and the phase margin is the amount of phase reduction necessary to make the system unstable. These margins are considered in classical control designs associated with the frequency response of the system. The gain and phase margins can be obtained from the Nyquist plot of $P(j\omega)C(j\omega)$ $\omega \in [0, \infty)$ as in Fig. 1.12. In Fig. 1.12, GM represents the gain margin, PM is the phase margin, ω_p is the phase crossover frequency, and ω_g the gain crossover frequency. Over time, there has been a research interest in developing new controller design approaches to achieve specific gain and phase margins for the closed-loop system. There is a significant number of research papers with different approaches for PI and PID controllers to achieve specific gain and phase margins. These different approaches for PI/PID controller design generally consider first-order or second-order plants cascaded with a time delay. In this monograph, we have presented a general approach for simultaneously achieving prescribed gain and phase margins for an arbitrary order plant.
- **Adaptive Control Design**
  In the adaptive controller design, the controller gains are to be readjusted in response to the changes in the system or due to the presence of perturbations. There are two types of adaptive control called direct and indirect methods. In the direct approach, the adaptive controller design approach known as *model reference adaptive control* is considered. In this a reference model, representing desired performance, is specified in terms of the characteristics of a dynamic system. Then, the difference between the output of the plant and the reference model, is used by an adaptation algorithm to *directly* adjust the parameters of the controller in real time to force the plant model error to zero. In the *indirect* approach, a model of the plant is estimated from the available input–output measurements. Then, the adaptive control scheme is called *indirect* because the readjustment of the controller parameters is made by first performing the estimation of the plant and then the

Fig. 1.12 Gain and phase margins from a typical Nyquist plot



computation of the controller parameters is based on the current estimated plant model. Recursive parameter estimation is used to update the process model. These types of adaptive controller techniques are widely used for PID controllers.

1.4 Integrator Windup

An essential element of the controller is the actuator, which applies the control signal u to the plant. However, all actuators have limitations that make them nonlinear elements. For instance, a valve cannot be more than fully opened or less than fully closed. During the regular operation of a control system, it can very well happen that the control variable reaches the actuator limits. When this situation arises, the feedback loop is broken, and the system runs as an open loop because the actuator will remain at its limit independently of the process output. If the controller is of the PID type, the error will continue to be integrated. This condition, in turn, means that the integral term may become very large, which is commonly referred to as *windup*. The error signal needs to have an opposite sign for an extended period to return to a normal state. As a consequence of all this, a system with a PID controller may give large transients when the actuator saturates.

The phenomenon of windup has been known for a long time. It may occur in connection with large setpoint changes or large disturbances or equipment malfunction may cause it. Several techniques are available to avoid windup when the integrator is in the controller. We describe some of these techniques in this section.

1.4.1 Setpoint Limitation

The easiest way to avoid integrator windup is to introduce limiters on the setpoint variations so that the controller output will never reach the actuator bounds. However, this approach has several disadvantages: (a) it leads to conservative bounds; (b) it imposes limitations on the controller performance; (c) it does not prevent windup caused by disturbances.

1.4.2 Back-Calculation and Tracking

This technique is illustrated in Fig. 1.13. We notice that the controller has an extra feedback path. This path is generated by measuring the actual actuator output $u(t)$ and forming the error signal $e_s(t)$ as the difference between the output of the controller $v(t)$ and the signal $u(t)$. This signal $e_s(t)$ is fed to the input of the integrator through a gain $1/T_t$.

When the actuator is within its operating range, the signal $e_s(t)$ is zero. Thus, it will not have any effect on the normal operation of the controller. When the actuator saturates, the signal $e_s(t)$ is different from zero. The regular feedback path around the process is broken because the process input remains constant. However, there is a new feedback path around the integrator due to $e_s(t) \neq 0$, and this prevents the integrator from winding up. The feedback gain $1/T_t$ governs the rate at which the controller output is reset. The parameter T_t can thus be interpreted as the time constant that determines how quickly the integral action is reset. In general, the smaller the value of T_t, the faster the integrator is reset. However, if the parameter T_t is chosen too small, spurious errors can cause saturation of the output, which accidentally resets the integrator.

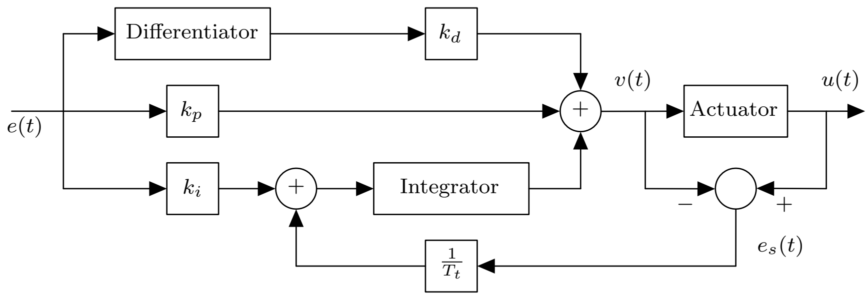


Fig. 1.13 Controller with antiwindup. © Taylor & Francis LLC Books. Reproduced from [8] with permission