ECE 680 Fall 2009

Proportional-Integral-Derivative (PID) Control

Stanislaw H. Żak

1 Introduction

The proportional-integral-derivative (PID) controllers are without a doubt the most widely used controllers in industry today. Åström and Hägglund [1, p. 198] regard the PID controller as the "bread and butter" of control engineering. Knospe [3] estimates that over 90% of control loops employ PID control. The PID control strategy is most useful when a mathematical model of the process to be controlled is not available to the control engineer.

To better appreciate the effectiveness of the PID control strategies, we first discuss a general control design principle referred to as the internal model principle (IMP).

2 The Internal Model Principle (IMP)

Consider a block diagram of a closed-loop system shown in Figure 1.

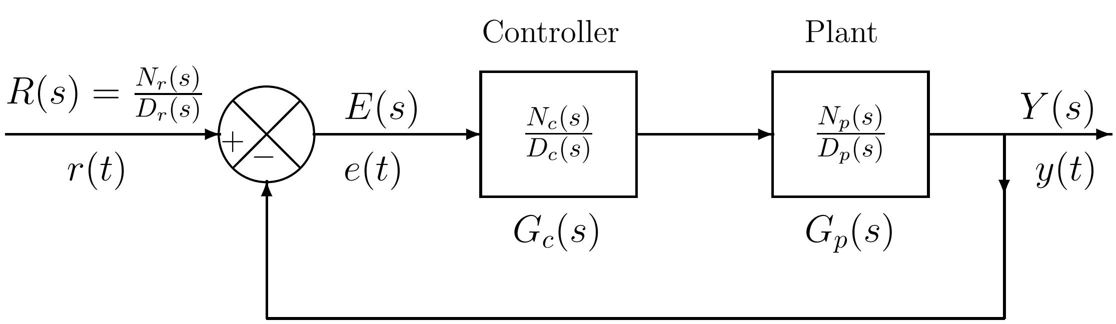

Figure 1: Block diagram of a closed-loop control system.

Using basic rules of the block diagram algebra yields

$$\begin{aligned} E(s) &= R(s) - G_p(s)G_c(s)E(s) \\ &= \frac{N_r(s)}{D_r(s)} - \frac{N_p(s)N_c(s)}{D_p(s)D_c(s)}E(s). \end{aligned}$$

Performing simple manipulations, we obtain

$$\boxed{E(s) = \frac{D_p(s)D_c(s)}{D_p(s)D_c(s) + N_p(s)N_c(s)} \frac{N_r(s)}{D_r(s)}} \tag{1}$$

Our objective is to design a controller $G_c(s) = \frac{N_c(s)}{D_c(s)}$ so that

$$\lim_{t\to\infty} e(t) = \lim_{t\to\infty} (r(t) - y(t)) = 0,$$

where $e(t) = \mathcal{L}^{-1}(E(s))$ is the inverse Laplace transform of $E(s)$.

Suppose that the poles of the Laplace transform of the reference signal $r(t)$ are in the closed right half-plane, that is, they belong to the set $\{s : \Re(s) \geq 0\}$, where $\Re(s)$ denotes the real part of the complex variable s. The polynomial

$$P_c(s) = D_p(s)D_c(s) + N_p(s)N_c(s)$$

is the closed-loop characteristic polynomial (CLCP) of the closed-loop system depicted in Figure 1 and the polynomial zeros are the closed-loop poles. Using (1), we can easily prove the following result that we refer to as the Internal Model Principle (IMP).

Theorem 1 (Internal Model Principle) *In the configuration depicted in Figure 1, where the poles of $R(s)$ are in the closed right half-plane,*

$$\lim_{t\to\infty} e(t) = 0$$

if and only if

1. *the closed-loop poles are in the open left-half plane;*
2. *$D_r(s)$ is a factor of the open-loop characteristic polynomial $D_p(s)D_c(s)$, that is, there is a polynomial, say $Q(s)$, such that $D_p(s)D_c(s) = Q(s)D_r(s)$.*

The second condition of the IMP an be interpreted as follows: the controller, $G_c(s)$, must be chosen in such a way that the open-loop transfer function, $G_p(s)G_c(s)$, contains a model

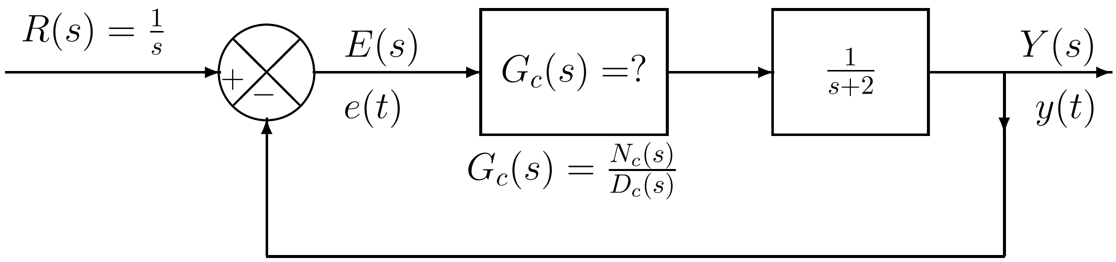

Figure 2: A closed-loop control system of Example 1.

of the reference signal to be tracked. If none of the poles of $R(s)$ is a pole of the plant's transfer function, $G_p(s)$, then we can restate the IMP as follows:

Any good tracking controller must stabilize the closed-loop system and must contain a model of the reference signal.

For an alternative approach to the IMP, we recommend Wolovich [7, pp. 254–255].

We now illustrate the IMP with two numerical examples.

Example 1 For the closed-loop system shown in Figure 2, our objective is to construct a transfer function $G_c(s)$ such that $\lim_{t\to\infty} e(t) = 0$. The Laplace transform of the error is

$$E(s) = \frac{1}{1 + \frac{1}{s+2}\frac{N_c(s)}{D_c(s)}}\frac{1}{s} = \frac{(s+2)D_c(s)}{(s+2)D_c(s) + N_c(s)}\frac{1}{s}.$$

Let $N_c(s) = 1$ and $D_c(s) = s$, that is, let

$$G_c(s) = \frac{1}{s},$$

Then,

$$E(s) = \frac{(s+2)s}{s^2+2s+1}\frac{1}{s} = \frac{s+2}{s^2+2s+1}.$$

It is easy to check using, for example, the final value theorem that $\lim_{t\to\infty} e(t) = 0$. Thus, in this case, a simple integral (I) controller does the job.

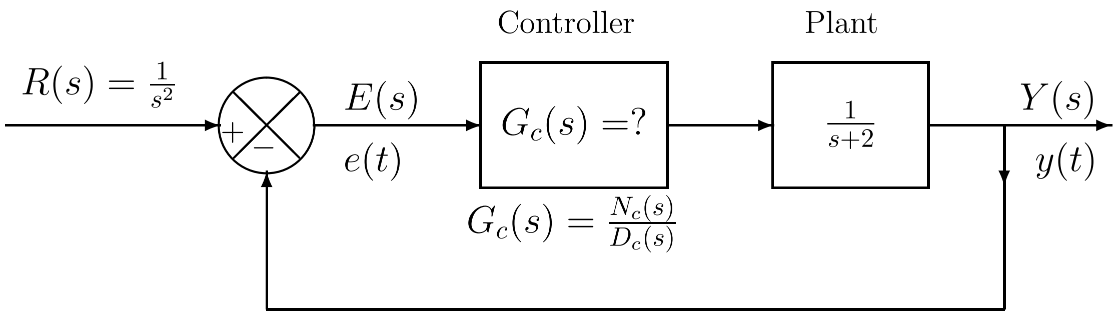

Figure 3: A closed-loop control system of Example 2.

Example 2 Consider the closed-loop system shown in Figure 3. We wish to construct a controller $G_c(s) = \frac{N_c(s)}{D_c(s)}$ so that $\lim_{t\to\infty} e(t) = 0$, where the reference signal is the unit ramp function. In Example 1, the simple integrator was all that we needed for the unit step as the reference input. Let us try the same controller here. Then, the Laplace transform of the error signal is

$$\begin{aligned} E(s) &= \frac{1}{1+\frac{1}{s+2}\frac{N_c(s)}{D_c(s)}}\frac{1}{s^2} \\ &= \frac{(s+2)D_c(s)}{(s+2)D_c(s)+N_c(s)}\frac{1}{s^2} \\ &= \frac{s+2}{s^2+2s+1}\frac{1}{s}. \end{aligned}$$

Note that now $e(\infty) = 2$. Thus the simple integrator is not enough now to force the steady-state error to zero. Note that in this case the polynomial $D_r(s)$ was not a factor of $D_c(s)$.

Let us then try a controller that would satisfy the second condition of the IMP, the divisibility condition. If we take the double integrator then the divisibility condition will be satisfied and we obtain

$$E(s) = \frac{s+2}{s^3+2s^2+1}.$$

However, now $e(t)$ is divergent because the poles of $E(s)$ are not stable. The three poles of $E(s)$ are located at -2.2056, and $0.1028 \pm j0.6655$. This time the first condition of the Internal Model Principle is not satisfied.

Finally, let us try a controller of the form

$$G_c(s) = \frac{N_c(s)}{D_c(s)} = \frac{s+a}{s^2}, \tag{2}$$

where a is a design parameter that should be chosen to satisfy the both conditions of the IMP. Note that the above controller contains the model of the reference signal. Thus, we only need to select the design parameter a so that the CLCP, $P_c(s)$, has its zeros in the open left half-plane, where

$$\begin{aligned} P_c(s) &= D_p(s)D_c(s) + N_p(s)N_c(s) \\ &= (s+2)s^2 + s + a \\ &= s^3 + 2s^2 + s + a \end{aligned}$$

Selecting $a = 1$ yields $P_c(s)$ with its zeros at

$$-1.7549, \ -0.1226 \pm j0.7449$$

and

$$E(s) = \frac{s+2}{s^3 + 2s^2 + s + 1}$$

We can now apply the final value theorem to $E(s)$ to obtain, $\lim_{t\to\infty} e(t) = 0$.

The selection of $a = 1$ results in the desired steady-state behavior of the closed-loop system. However, the above controller may, or may not, guarantee the desired transient performance of the closed-loop system. Thus further tuning of the parameter a may be necessary.

In industrial applications, control engineers usually specify the performance of the controlled system based on the system step response. In such a case $r(t)$ is a step function, which sometimes is also referred to as the constant *set-point*, which can be represented as

$$r(t) = \begin{cases} 0 & \text{for} \quad t < 0 \\ A & \text{for} \quad t \geq 0, \end{cases}$$

where A is the constant set-point value. Recall that the Laplace transform of the above signal is

$$R(s) = \mathcal{L}(r(t)) = \frac{A}{s}.$$

Taking the above into account it is easy to see that the integral (I) controller contains the model of the unit step function as well as any scalar multiple of it. If, in addition, the zeros of the CLCP are all in the open left-half plane, then $e(\infty) = 0$ for any step input. For this reason, the I controller is often referred to as as *reset controller* because it automatically ensures $e(\infty) = 0$ without an operator manually altering or "resetting" the set-point value to achieve a desired output value.

3 PID Controller Architecture

We begin our discussion with a basic PID controller architecture, also known as the type A PID controller; it consists of three-terms. Later we discuss two modifications, B and C, of the type A PID control architecture. The three terms of a PID controller attempt to fulfill the transient and steady-state specifications.

To proceed, we need some notation. Let $u(t)$ denote the output of the controller, which is the input signal to the plant. Recall that $e(t)$ is the tracking error, that is, the difference between the desired plant output, $r(t)$, and the actual plant output, $y(t)$. Thus

$$e(t) = r(t) - y(t). \tag{3}$$

The three-term type A PID control law can be described as

$$u(t) = u_P(t) + u_I(t) + u_D(t), \tag{4}$$

where u_P is the proportional term, u_I is the integral term, and u_D is the derivative term. We now analyze each term one by one.

- The proportional term, u_P, has the form

  $$u_P(t) = K_P e(t), \tag{5}$$

  where K_P is the proportional gain.

  The proportional term responds immediately to the current tracking error. Typically, however, it cannot achieve the desired tracking accuracy without excessively large gain.

- The integral term, u_I, is given by

  $$u_I(t) = \frac{K_P}{T_I} \int_0^t e(\tau) d\tau, \tag{6}$$

  where T_I is the integral time constant.

  The integral term yields a zero steady-state error in tracking a step function (a constant set-point). This term is slow in response to the current tracking error.

- The derivative term, u_D, is

$$u_D(t) = K_P T_D \frac{de(t)}{dt}, \tag{7}$$

where T_D is the derivative time constant.

The derivative term is especially effective for plants with significant dead-time resulting in a poor representation of previous control actions in the current tracking error. This, in turn, as pointed out by Knospe [3], may lead to large transient error when PI control is used. The derivative action combats this problem by generating its action based on a prediction of future tracking error. But there is a cost to be paid—the derivative term amplifies higher frequencies sensor noise. To alleviate the problem, a filtering of the differentiated signal is employed thus resulting in a PID controller with a derivative filter referred to as a PIDF controller. We present a PIDF controller in a section on implementing PID controllers.

Combining the above terms together, we obtain the type A PID control action, in the time domain,

$$\begin{aligned} u(t) &= u_P(t) + u_I(t) + u_D(t) \\ &= K_P \left(e(t) + \frac{1}{T_I} \int_0^t e(\tau) d\tau + T_D \frac{de(t)}{dt} \right). \end{aligned} \tag{8}$$

It is advantageous to work in the Laplace domain when analyzing the PID-controlled plant. The Laplace transfer function of the type A PID controller, denoted $G_{PID}(s)$, is

$$G_{PID}(s) = \frac{U(s)}{E(s)} = K_P \left(1 + \frac{1}{T_I s} + T_D s \right), \tag{9}$$

where $U(s) = \mathcal{L}(u(t))$ is the Laplace transform of $u(t)$ and $E(s) = \mathcal{L}(e(t))$ is the Laplace transform of the tracking error $e(t)$. In Figure 4, we show a PID-controlled system with the type A PID controller in the forward path.

4 PID Controller Tuning

The control engineer's task is to select the PID controller's parameters to meet given performance specifications. This process is referred to as the controller tuning. The most widely known PID tuning rules were proposed by Ziegler and Nichols [8] in 1942. They proposed two sets of tuning rules of the type A PID controllers. In both cases the controller's parameters are selected based on the experimental measurements of the controlled process.

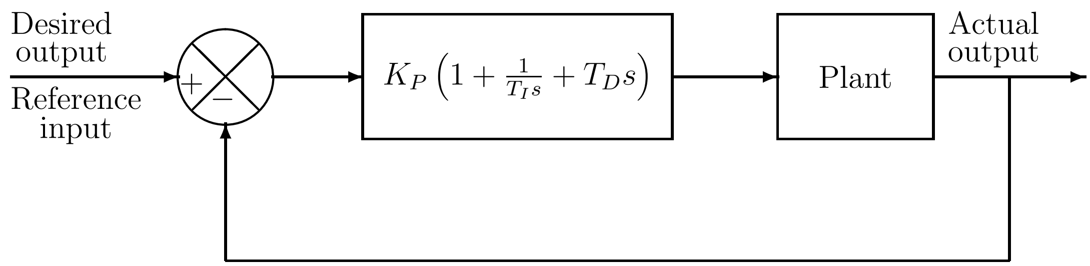

Figure 4: A PID-controlled system.

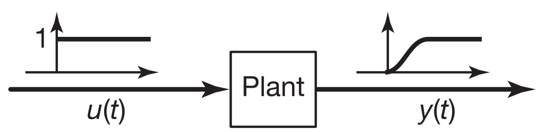

Figure 5: Open-loop test on the plant—measuring the step response of the plant.

4.1 The Open-Loop Method—Tuning Based on Open-Loop Tests on the Plant

The first method of choosing control settings for the PID controller, proposed by Ziegler and Nichols, is based on open-loop tests on the plant as shown in Figure 5. There is a restriction on the applicability of this method. The method can only be applied if the step response is S-shaped as shown in Figure 6. The S-shaped step response can be characterized using three parameters: the delay time (dead-time) L, the time constant T, and the steady-state gain (process gain) K.

We add that a plant with the S-shaped step response can be approximately modeled by a first-order transfer function with transportation delay,

$$\frac{Y(s)}{U(s)} = G_d(s) = \frac{Ke^{-Ls}}{Ts+1}. \tag{10}$$

We illustrate the procedure of approximating a system with an S-shaped step response using a first-order transfer function with transportation delay given by (10) with the following example.

Example 3 We consider, as in Ogata [5, p. 444], a plant modeled by the second-order

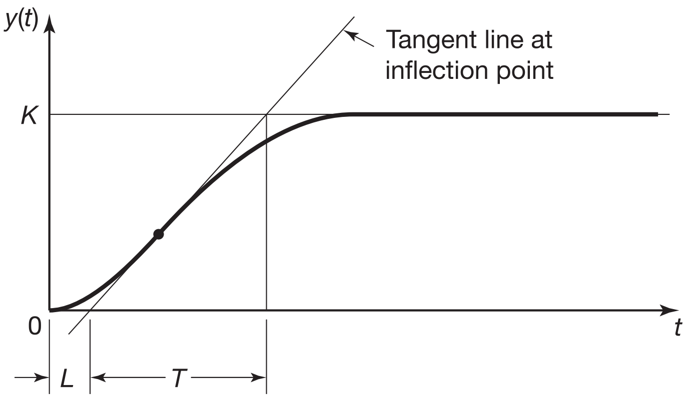

Figure 6: An S-shaped step response and its characterization using three parameters.

transfer function

$$G_p(s) = \frac{10}{(s+1)(s+5)} = \frac{10}{s^2+6s+5}. \tag{11}$$

Using the procedure illustrated in Figure 6, we approximate the second-order transfer function (11) using the first-order transfer function with transportation delay,

$$G_d(s) = \frac{Ke^{-Ls}}{Ts+1} = \frac{2e^{-0.053s}}{0.798s+1}. \tag{12}$$

In Figure 7, we compare the step responses of the above two transfer functions. One can see that approximating the S-shaped step step response of a dynamic system using a first-order transfer function with transportation delay is a challenging problem. Global optimization methods, such as genetic algorithms and particle swarm optimizers are the tools that can be used here effectively to determine parameters L and T that result in $G_d(s)$ that "best" approximates a given $G_p(s)$.

Ziegler and Nichols proposed a set of tuning rules, formulas, using the parameters that characterize the step response of the plant modeled by the transfer function (10). These tuning formulas are shown in Table 1.

Substituting the tuning parameters given in Table 1 into the transfer function of the PID controller gives

$$\begin{aligned} G_{PID}(s) &= K_P\left(1+\frac{1}{T_I}s+T_Ds\right) \\ &= 1.2\frac{T}{L}\left(1+\frac{1}{2Ls}+0.5Ls\right) \end{aligned}$$

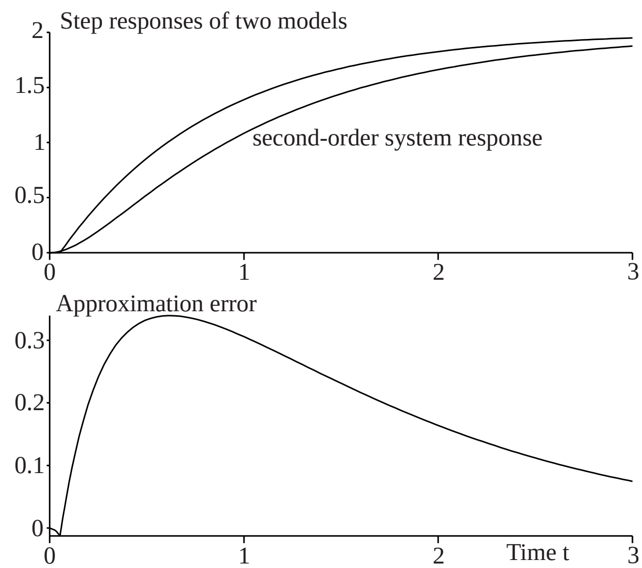

Figure 7: Comparison of step responses of the second-order system and its approximation using a first-order system with transportation delay in Example 3.

Table 1: Ziegler-Nichols tuning formulas based on the step response of the plant (open-loop method).

Controller	K_P	T_I	T_D
P	$\frac{T}{L}$	∞	0
PI	$0.9\frac{T}{L}$	$\frac{L}{0.3}$	0
PID	$1.2\frac{T}{L}$	$2L$	$0.5L$

$$= \quad 0.6T\frac{\left(s+\frac{1}{L}\right)^2}{s}. \tag{13}$$

Thus the type A PID controllers tuned using the formulas of the open-loop method can be parameterized by two parameters, T and L.

The ACT controller manufacturer, as well as Li, Ang, and Chong [4], present slightly modified Ziegler-Nichols open-loop tuning formulas shown in Table 2. The only difference between the formulas shown in Table 1 and those in Table 2 is in the first column where all the column elements of the first column of Table 2 are obtained by dividing by K the corresponding elements of Table 1.

Table 2: Modified Ziegler-Nichols tuning formulas based on the step response of the plant (open-loop method).

Controller	K_P	T_I	T_D
P	$\frac{T}{LK}$	∞	0
PI	$0.9\frac{T}{LK}$	$\frac{L}{0.3}$	0
PID	$1.2\frac{T}{LK}$	$2L$	$0.5L$

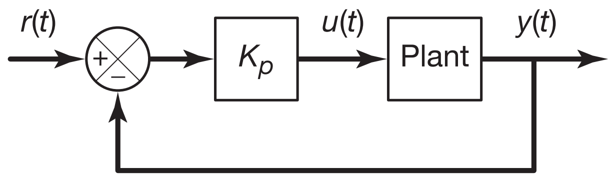

Figure 8: Experimental setup to determine the critical gain, K_{cr} and critical period of oscillation, P_{cr}.

4.2 The Closed-Loop Method—Tuning Based on Closed-Loop Tests on the Plant

The second method of Ziegler and Nichols of tuning PID controllers is also known as the the closed-loop method. This method requires an experiment on the closed-loop system as shown in Figure 8. We thus have a pure P-controller in the *feed-forward path*. A step is applied as the input and the response is observed. The test is repeated with increased or decreased controller gain until a sustained oscillation is achieved, as shown in Figure 9. This gain is called the "critical gain" and denoted K_{cr}. The period of the sustained oscillation is called the critical period and denoted P_{cr}. Using the parameters K_{cr} and P_{cr}, Ziegler and

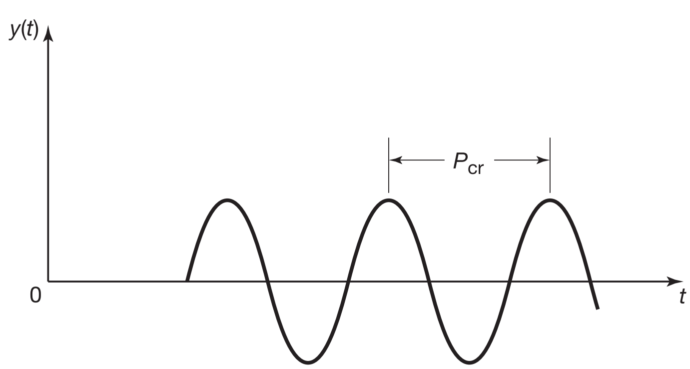

Figure 9: Measuring the period of the sustained oscillation, P_{cr}.

Nichols proposed tuning formulas depicted in Table 3. Substituting the tuning parameters

Table 3: Ziegler-Nichols tuning formulas based on critical gain K_{cr} and critical period P_{cr} (closed-loop method).

Controller	K_P	T_I	T_D
P	$0.5K_{cr}$	∞	0
PI	$0.45K_{cr}$	$\frac{P_{cr}}{1.2}$	0
PID	$0.6K_{cr}$	$0.5P_{cr}$	$0.125P_{cr}$

from Table 3 into the transfer function of the type A PID controller, we obtain

$$\begin{aligned} G_{PID}(s) &= K_P\left(1+\frac{1}{T_I}s+T_Ds\right) \\ &= 0.6K_{cr}\left(1+\frac{1}{0.5P_{cr}s}+0.125P_{cr}s\right) \\ &= 0.075K_{cr}P_{cr}\frac{\left(s+\frac{4}{P_{cr}}\right)^2}{s}. \end{aligned} \tag{14}$$

As in the open-loop method, the type A PID controllers tuned using the formulas of the closed-loop method can be parameterized by two parameters; this time by K_{cr} and P_{cr}.

In summary, both the open-loop and closed-loop Ziegler-Nichols tuning rules allow us to parameterize the type A PID controllers using just two parameters. That is, we can represent a family of type A PID controllers as

$$G_{A-PID} = K_c\frac{(s+a)^2}{s}, \tag{15}$$

where K_c and a are design parameters. The above representation can be used to computationally obtain parameters' values that satisfy design specifications.

5 Implementing PID Controllers

If the reference input is a step, then because of the presence of the derivative term, the controller output will involve an impulse function. In addition, the derivative term amplifies high frequency sensor noise. To alleviate the above problems, the pure derivative term is replaced with a derivative filter, which results in the PIDF controller.

5.1 PIDF Controller

In the PIDF controller, the pure differentiator is approximated with the pure differentiator cascaded with a first-order low-pass filter. Thus the pure derivative term is replaced by the approximate differentiator and so the derivative term contribution would take the form,

$$U_D(s) = \frac{T_D s}{\gamma T_D s + 1} E(s), \tag{16}$$

where γ is a small parameter, for example, $\gamma = 0.1$. The transfer function of the type A PIDF controller has the form

$$G_{PIDF} = K_P \left(1 + \frac{1}{T_I} s + \frac{T_D s}{\gamma T_D s + 1} \right). \tag{17}$$

For a step reference, the PIDF controller output will involve a sharp pulse rather than an impulse function. This is referred to as the *set-point kick phenomenon*.

Example 4 In this example, we use the open-loop Ziegler-Nichols tuning formulas given in Table 1 to tune the PIDF controller for plant models in Example 3. We first apply the PIDF controller to the second-order transfer function given by (11). We add that the proportional gain was tuned as

$$K_P = 1.2 \frac{T}{L} = 18.068$$

because for the modified gain, $K_P = 1.2 \frac{T}{LK}$, the closed-loop system becomes unstable. The remaining parameters are:

$$T_I = 2L = 0.106, \quad T_D = 0.5L = 0.0265, \quad \text{and} \quad \gamma = 0.1.$$

The plant output and the control signal plots versus time are shown in Figure 10. The system output is very oscillatory and takes long time to settle down. We can observe the set-point kick phenomenon in the plot of the control effort versus time.

We then applied the type A PIDF controller to the transfer function given by (12). Here, on the other hand, we used $K_P = 1.2 \frac{T}{LK}$ because for $K_P = 1.2 \frac{T}{L}$ the closed-loop system becomes unstable. The plant output and the control signal plots versus time are shown in Figure 11. Here, as in the previous case, we can observe a set-point kick phenomenon. The closed-loop system is much faster here but the overshoot is at unacceptable level.

We conclude that the open-loop Ziegler-Nichols tuning rules do not necessarily yield excellent performance of the PID-controlled system. The controller requires careful additional tuning. To illustrate this point, we doubled the values of K_P, T_I, and T_D. The re-tuned

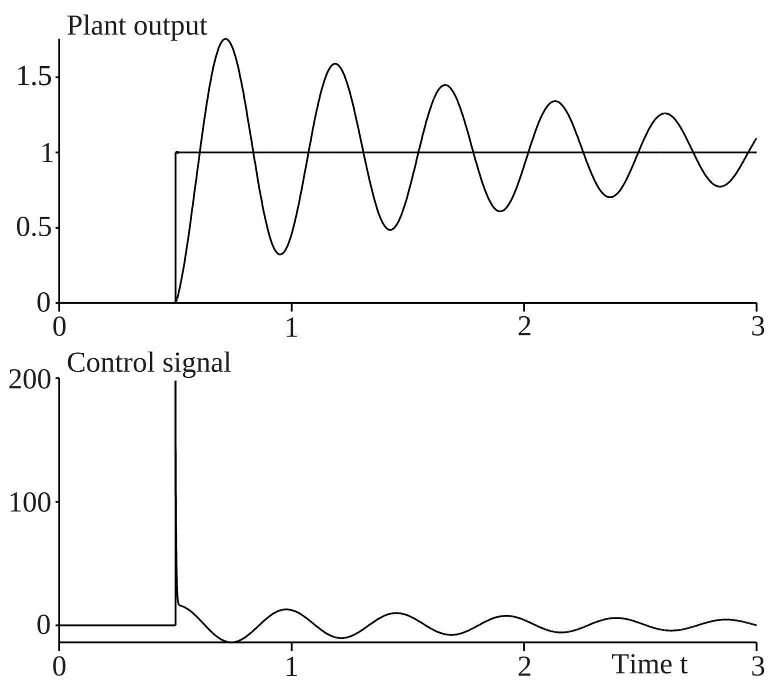

Figure 10: PIDF-controlled second-order system's step response of Example 4, where the reference input is a step function delayed by 0.5 s in order to better observe the set-point kick phenomenon.

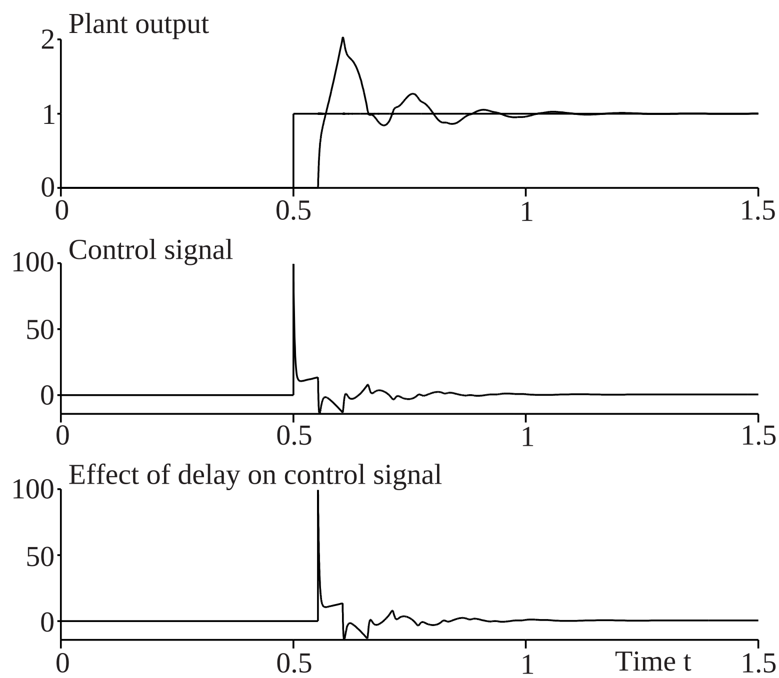

Figure 11: Step response of the PIDF-controlled first-order system with transportation delay in Example 4.

PIDF controller performance applied to the second-order plant is shown in Figure 12. This re-tuning was suggested by Ogata [6, p. 165]. However the re-tuned controller does not work for the first-order plant with transportation delay. The closed-loop system driven by this controller is in fact unstable.

As we observed before, in addition to the transient performance, the designer has to deal with the set-point kick phenomenon. We address this issue next.

5.2 Integrator Windup

When implementing a PID controller, we need to take into account physical constraints of the controlled plant. In particular, a control engineer has to remember that all actuators have physical limitations; a motor has a finite limited speed while a valve can only be opened to its limits. Thus a control engineer will be faced with the problem of actuator saturation.

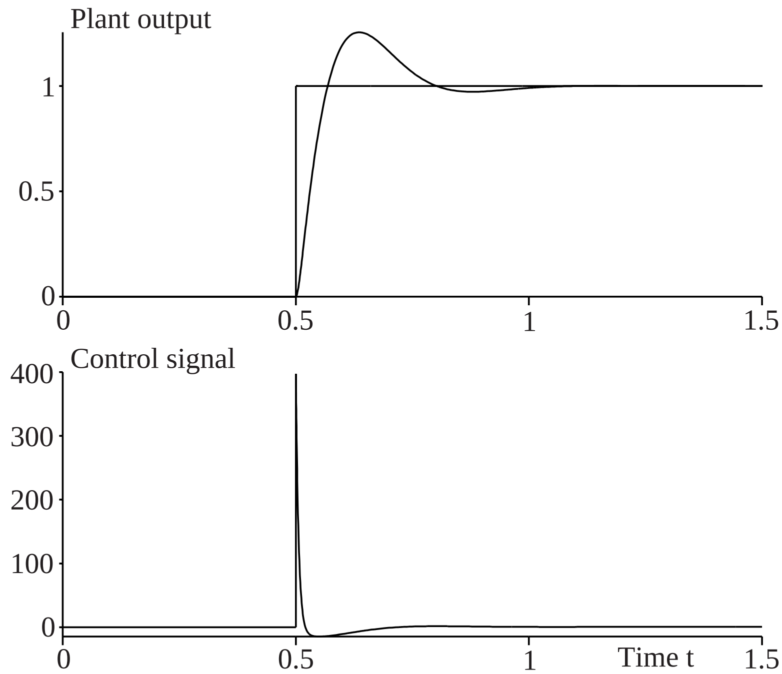

Figure 12: Step response of the closed-loop system with re-tuned PIDF controller and second-order plant in Example 4.

To illustrate the issue, consider Example 4. Suppose now that the actuator saturation limits are ± 10, that is,

$$|u(t)| \leq 10$$

These constraints are violated in the above example because of the set-point kick phenomenon. Let us now impose the constraints on the control action and see how this affects the closed-loop system behavior.

Example 5 First, we consider the second-order plant (11) simulated in the previous example. We now impose constraints on the actuator, $|u(t)| \leq 10$. The PIDF controller parameters are the same as the ones we used in the simulation shown in Figure 12. The output of the closed-loop system is shown in Figure 13.

As can be seen in Figure 13 and 14, in both cases the control signal saturates in the moment the reference input is applied. The input to the plant remains saturated in the sense that the feedback loop is broken because the controller's output is not responding to

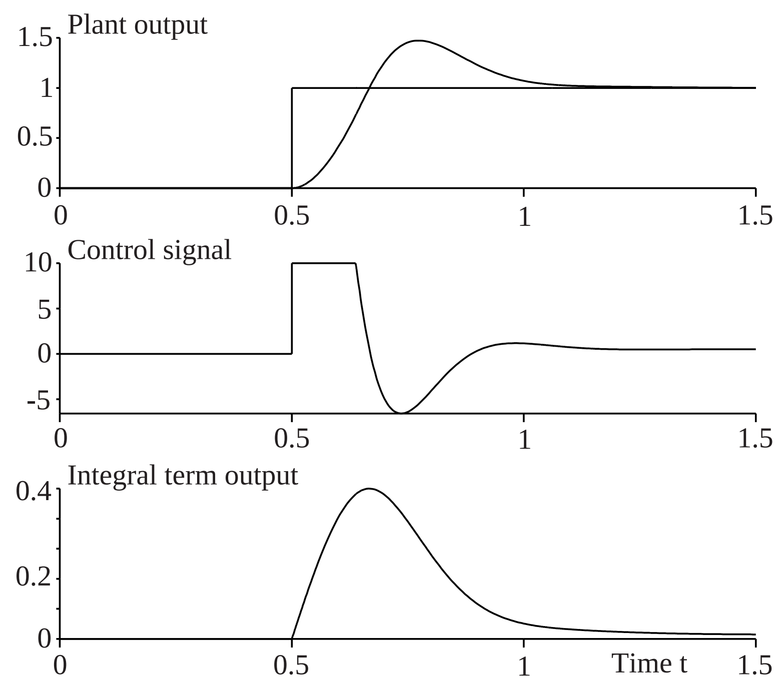

Figure 13: A PIDF-controlled second-order plant with with constrained actuator.

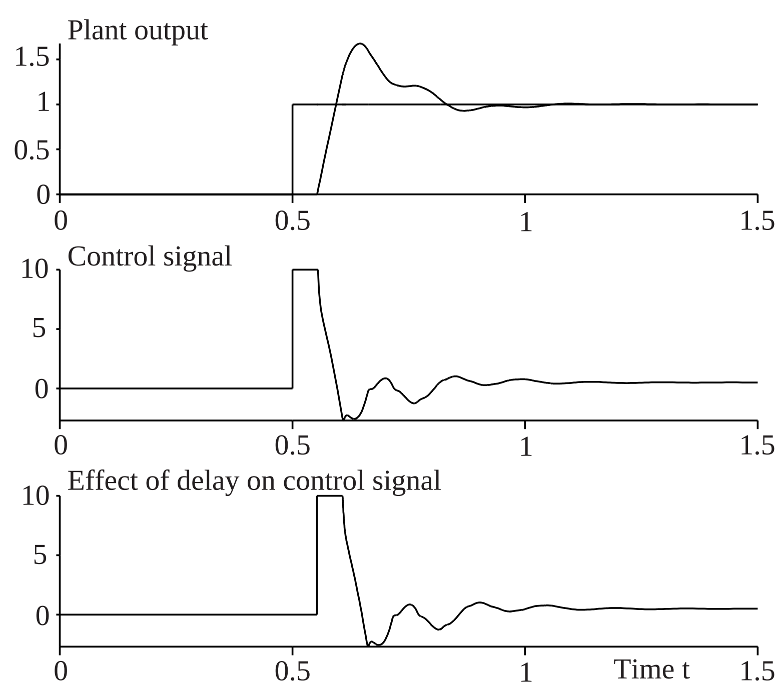

Figure 14: Plant output of the PIDF-controlled first-order plant with with transportation delay and constraints on input.

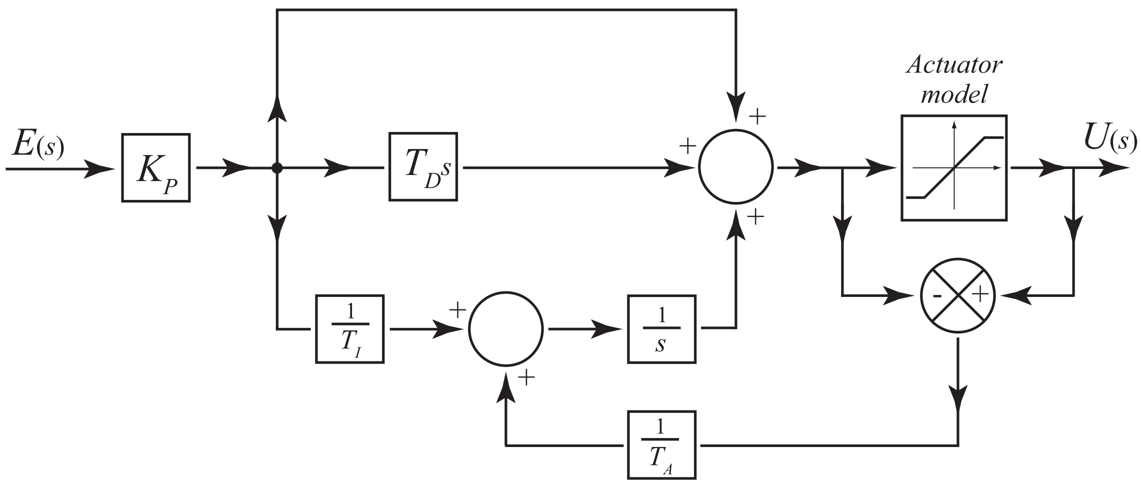

Figure 15: A type A PID controller with the integrator anti-windup.

its input. The output of the integral term continues to increase because the tracking error is positive. The output of the integral term starts decreasing when the plant's output becomes larger than the reference input. However, the controller stays saturated for some time because of the excessively large output of the integral term. As a result of the integral term excessive output, the plant's output has a large overshoot. This phenomenon of excessive integrator term output resulting in the large overshoot of the plant's output is called the *integrator windup*.

We can avoid or reduce the integrator windup by reducing the integral term output when the actuator saturates. A possible *anti-windup* scheme is shown in Figure 15. This anti-windup mechanism is referred to as tracking or back calculation. In Figure 16, we show the output of the PIDF-controlled second-order plant with the anti-wind, where $T_A = T_I/0.075$. In Figure 17, we show the output of the first-order plant with transportation delay controlled with a PIDF controller with the integrator windup. The PIDF controller parameters are the same as in the simulation of Figure 11, and $T_A = 0.21$.

6 Modified PID Control Schemes

We now discuss two alternative schemes to eliminate the set-point kick phenomenon. We first take another look at the type A PID control architecture as shown in Figure 18. In order to avoid differentiating a step reference input, it is proposed to operate the derivative action in the feedback as shown in Figure 19; hence its name, the PI-D controller also referred to as the type B PID controller. A practical implementation of this controller that we refer to

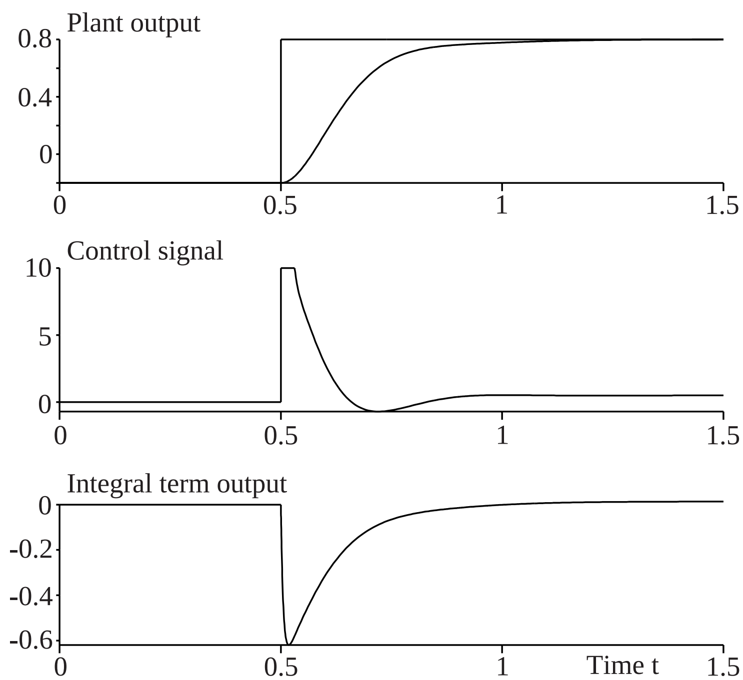

Figure 16: Plant output of the PIDF-controlled second-order plant with the integrator ani-windup circuit.

as the type B PIDF controller action can be described as

$$U(s) = K_P\left(1 + \frac{1}{T_I s}\right)E(s) - \frac{K_P T_D s}{\gamma T_D s + 1}Y(s).$$

We can go even further and move the proportional and derivative actions into feedback giving us the I-PD control or the type C PID controller shown in Figure 20. A practical implementation of this controller that we refer to as the type C PIDF controller action can be described as

$$U(s) = \frac{K_P}{T_I s}E(s) - K_P\left(1 + \frac{T_D s}{\gamma T_D s + 1}\right)Y(s).$$

7 Digital Implementation

In this section we present a method of implementing PID controllers using a computer. To obtain a computer implementation, we need to discretize the continuous-time differential equation describing a controller at hand. We now discuss discretization of different terms of the type A PIDF controller. The types B and C PIDF controllers can be easily obtained by

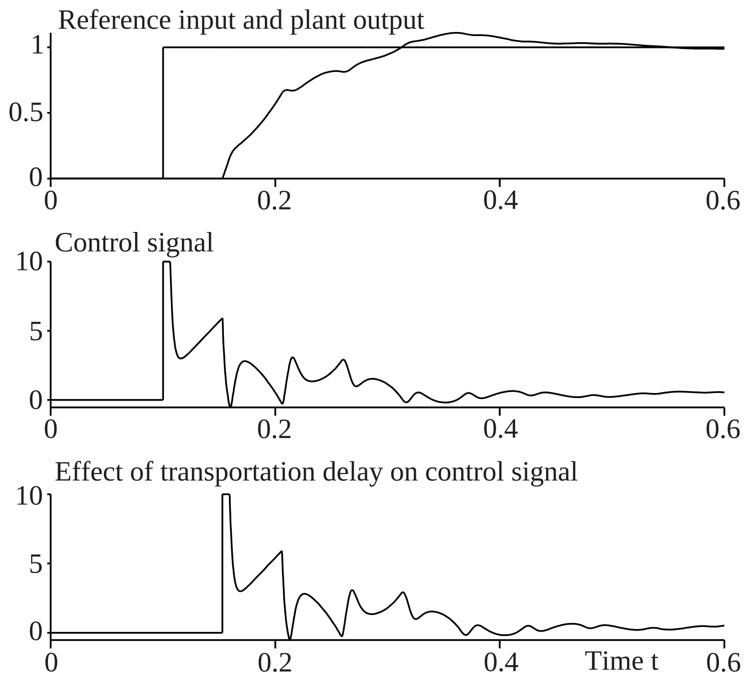

Figure 17: Plant output of the PIDF-controlled first-order plant with transportation delay and the integrator ani-windup.

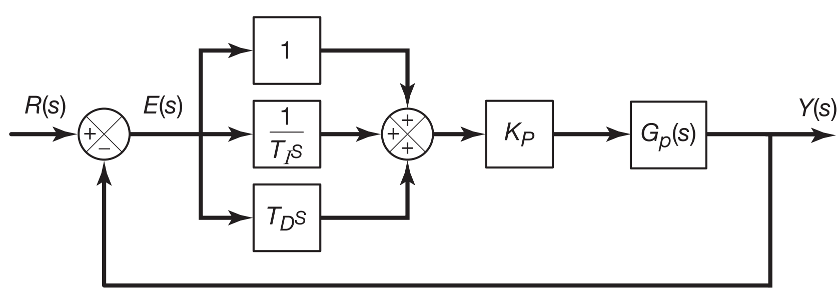

Figure 18: More detailed view of the type A PID controller.

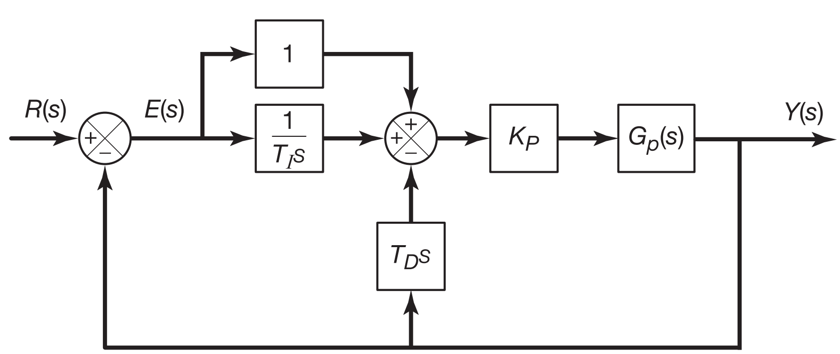

Figure 19: Type B PID controller.

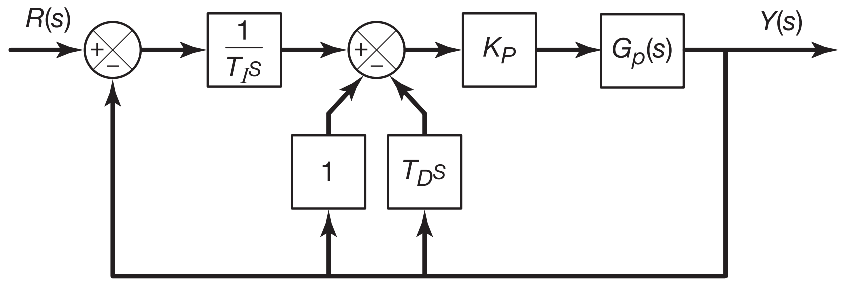

Figure 20: Type C PID controller.

modifying the A PIDF architecture. In our discussion, we assume that the sampling instants are equally spaced. We denote the sampling period as h, and sampling instants as t_k, where $k = 0, 1, 2, \ldots$.

- The discrete-time version of the proportional term, $u_P(t_k)$, has the form

$$u_P(t_k) = K_P e(t_k), \tag{18}$$

  where K_P is the proportional gain.

- Recall that the continuous-time integral term, $u_I(t)$, is given by

$$u_I(t) = \frac{K_P}{T_I} \int_0^t e(\tau) d\tau,$$

  Differentiating both sides of the above integral equation yields

$$\frac{du_I(t)}{dt} = \frac{K_P}{T_I} e(t).$$

  We can now use several methods to discretize the above differential equation. Using the forward difference discretization gives

$$u_I(t_{k+1}) = u_I(t_k) + \frac{K_P h}{T_I} e(t_k). \tag{19}$$

  For other discretization methods, the reader may wish to consult the section on numerical techniques.

- Performing cross-multiplication in (16) and representing the resulting polynomial equation, into the Laplace variable s, in the time domain gives the continuous-time differential equation describing the filtered derivative term,

$$\gamma T_D \frac{du_D(t)}{dt} + u_D(t) = K_P T_D \frac{de(t)}{dt}. \tag{20}$$

The forward difference discretization of the above differential equation yields

$$u_D(t_{k+1}) = \left(1 - \frac{h}{\gamma T_D}\right) u_D(t_k) + \frac{K_P}{\gamma}\left(e(t_{k+1}) - e(t_k)\right). \tag{21}$$

On the other hand, the backward difference discretization of the differential equation (20) gives,

$$u_D(t_k) = \frac{\gamma T_D}{\gamma T_D + h} u_D(t_{k-1}) + \frac{K_P}{\gamma T_D + h}\left(e(t_k) - e(t_{k-1})\right). \tag{22}$$

Combining (18), (19), and (22), we obtain a discrete-time approximation of the type A PIDF controller,

$$\left.\begin{array}{rcl} u_P(t_k) & = & K_P e(t_k) \\ u_D(t_k) & = & \frac{\gamma T_D}{\gamma T_D + h} u_D(t_{k-1}) + \frac{K_P}{\gamma T_D + h}\left(e(t_k) - e(t_{k-1})\right) \\ u(t_k) & = & u_P(t_k) + u_I(t_k) + u_D(t_k) \\ u_I(t_{k+1}) & = & u_I(t_k) + \frac{K_P h}{T_I} e(t_k) \end{array}\right\} \tag{23}$$

To implement the above controller, one can use the zero-order hold.

8 Notes

The first PID controllers for industrial applications were introduced in 1939 by the Taylor Instrument Company and the Foxboro Instrument Company. For an account of the PID controller development along with an overview of early process control devises, we recommend Bennett [2]. For an account of recent advances in PID control, see an article by Li, Ang, and Chong [4].

Many engineers are familiar with PID controllers and prefer to enhance the performance of a well-known solution through additional capabilities rather than switch to an untested solution. Knospe [3] observers that even alternative digital control methods have not supplanted PID's primacy in industrial control with a possible exception of model predictive control, where the control action is calculated on the fly. Knospe [3] adds that PID control still has a role even in this case by handling lower level loops.

References

[1] K. J. Åström and T. Hägglund. PID Control. In W. S. Levine, editor, *The Control Handbook*, chapter 10.5, pages 198–209. CRC Press in cooperation with IEEE Press, Boca Raton, Florida 33431, 1996.

[2] S. Bennett. Development of the PID controller. *IEEE Conrol Systems Magazine*, 13(6):58–65, December 1993.

[3] C. Knospe. PID control—Introduction to the special section. *IEEE Conrol Systems Magazine*, 26(1):30–31, February 2006.

[4] Y. Li, K. H. Ang, and G. C. Y. Chong. PID control system analysis and design: Problems, remedies, and future directions. *IEEE Conrol Systems Magazine*, 26(1):32–41, February 2006.

[5] K. Ogata. *Modern Control Engineering*. Prentice Hall, Upper Saddle River, New Jersey 07458, third edition, 1997.

[6] K. Ogata. *Solutions Manual to Modern Control Engineering*. Prentice Hall, Upper Saddle River, New Jersey 07458, third edition, 1997.

[7] W. A. Wolovich. *Automatic Control Systems: Basic Analysis and Design*. Saunders College Publishing, Harcourt Brace College Publishers, Fort Worth, 1994.

[8] J. G. Ziegler and N. B. Nichols. Optimum settings for automatic controllers. *Transactions of the American Society of Mechanical Engineers*, 64:759–768, November 1942.