

Digital Control Systems

Summary # 8

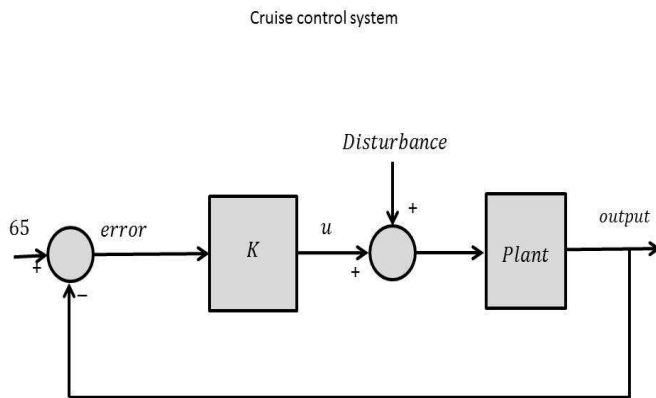


Fig. 1. Cruise control system with proportional control

Saturation and bias term

The error between the desired set point and the actual output is defined as

$$e(t) = r(t) - y(t) \quad (1)$$

where $r(t)$ is the desired set point. The set point can be constant or time varying. Consider the example of cruise control shown in figure 1 where u is the throttle signal adjusting the flow of fuel to the engine. Assume that the desired speed is 65. Under proportional control, we have

$$u(t) = K_p(r(t) - y(t)) \quad (2)$$

If the output is also 65, there is no action by the controller since $u = 0$. Therefore, it is not possible to achieve zero steady state error using the control law of equation 2. A small modification is needed as follows

$$u(t) = u_0 + K_p e(t) \quad (3)$$

where u_0 is called the bias or null value. It can be defined as the value of u that cause $y = r$ when there is no disturbance. This bias is not always needed, but there are situations where it is necessary. For digital controllers, u is usually expressed as a percentage.

Saturation is a nonlinear phenomenon. It is in general a characteristic of the physical limitations of the hardware. For example a valve cannot be more open than its max value and a motor cannot rotate more than its maximum rated

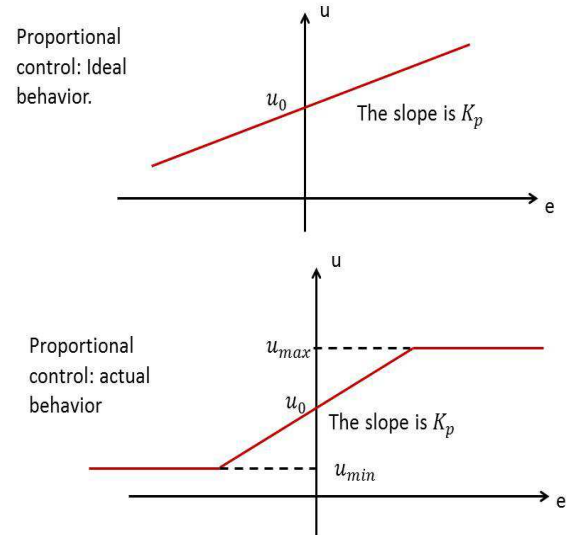


Fig. 2. Proportional control with and without saturation

speed. These limitations need to be taken into account. The proportional controller with saturation and bias is illustrated in figure 2. The saturation shown in the figure is an important limitation that needs to be taken into account.

Standard PID form

The PID control given by

$$u(t) = K_p + K_i \int e(t)dt + K_d \frac{e(t)}{dt} \quad (4)$$

is called ideal parallel form. Another form called the standard form is widely used in industry. The standard form is discussed below.

PI control under standard form

The proportional integral control is widely used because of its important practical advantage: it eliminates the steady state error. The continuous time standard form transfer function of a PI controller is

$$K_p \left(1 + \frac{1}{\tau_I s}\right) \quad (5)$$

PD control under standard form

PD control has two terms, a proportional term and a derivative term. This controller is also called rate action, pre-act control and anticipatory control. The PD continuous time standard form transfer function is

$$K_p (1 + \tau_D s) \quad (6)$$

Proportional, integral and derivative under standard form.

PI and PID control have been dominant control algorithms for process control for decades (about 95% of process controllers utilize some form of PID). The continuous time standard form transfer function for PID controller is given by

$$K_p(1 + \frac{1}{\tau_I s} + \tau_D s) \quad (7)$$

Position and velocity algorithms for digital controllers

There exist several variants of PID control. Three different variants are discussed below.

Position algorithm

The discrete time equation for the digital PID controller output is given by

$$u(k) = u_0 + K_p \left[e(k) + \frac{T}{\tau_I} \sum_{j=1}^k e(j) + \frac{\tau_D}{T} (e(k) - e(k-1)) \right] \quad (8)$$

where T is the sampling time. This form is called position algorithm.

Velocity algorithm

In the position PID control, we calculated the actual value of the PID controller output. In the velocity form we calculate the change in the controller output. The velocity form can be derived from the position form as follows:

$$u(k-1) = u_0 + K_p \left[e(k-1) + \frac{T}{\tau_I} \sum_{j=1}^{k-1} e(j) \right] \quad (9)$$

$$+ K_p \left[\frac{\tau_D}{T} (e(k-1) - e(k-2)) \right] \quad (10)$$

The velocity form is based on the difference given by

$$\Delta u = u(k) - u(k-1) \quad (11)$$

and therefore

$$\Delta u = K_p \left[e(k) - e(k-1) + \frac{T}{\tau_I} e(k) \right] \quad (12)$$

$$+ K_p \left[\frac{\tau_D}{T} (e(k) - 2e(k-1) + e(k-2)) \right] \quad (13)$$

It is possible to obtain an explicit formula for the controller as follows

$$u(k) = u(k-1) + K_p \left[e(k) - e(k-1) + \frac{T}{\tau_I} e(k) \right] \quad (14)$$

$$+ K_p \left[\frac{\tau_D}{T} (e(k) - 2e(k-1) + e(k-2)) \right] \quad (15)$$

PID with set point weighting

The input is weighted as follows:

$$u = K_p(\alpha r - y) + K_i \int_0^\infty (r(\tau) - y(\tau)) d\tau + K_d(\beta \frac{dr}{dt} - \frac{dy}{dt}) \quad (16)$$

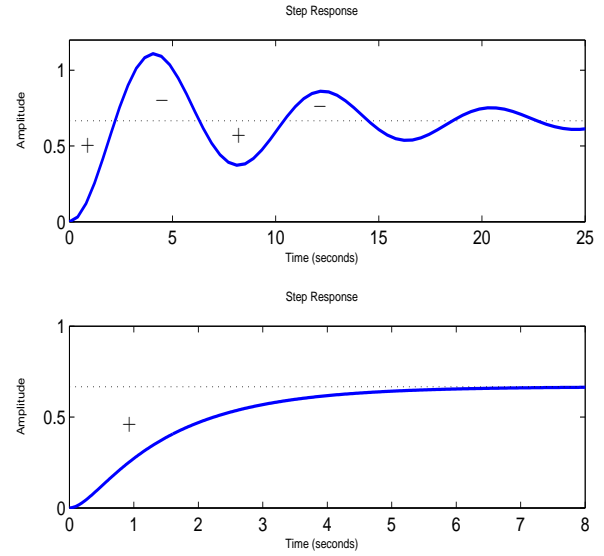


Fig. 3. Illustration of integral wind up. Top: sign of the error alternates, positive and negative values cancel out. Bottom: error sign is always positive, saturation is more likely to happen

where α and β are small constants. They are called set point weights.

Problems with PID controllers: Derivative kick and integral windup

A sudden change in the error that usually results from the change in the set point will cause the derivative part to become very large. This spike is undesirable. One way to solve this issue is by using $\frac{-dy}{dt}$ instead of $\frac{de}{dt}$. Integral wind up (or reset wind up) is another problem that happens when the integral output becomes large and the controller output becomes saturated. The build up of the integral terms is called integral windup. Consider the time response of figure 3. In figure 3-top, the integral term initially increases but begins decreasing again when the error changes sign. The positive and negative terms cancel out and the controller moves away from the saturation point. In figure 3-bottom, the integral term keeps increasing and this may result in in integral wind up. The PID velocity form is called anti-integral wind up because the summation is eliminated.

Performance criteria for closed loop systems

The performance criteria are as follows

- The closed loop system is stable
- Steady state error is eliminated
- Desired transient response
- Robust, the closed loop system is insensitive to changes in the plant conditions and modeling inaccuracies.
- Disturbance rejection

It is not possible to achieve all these goals simultaneously because of the conflicts and trade offs that may exist.

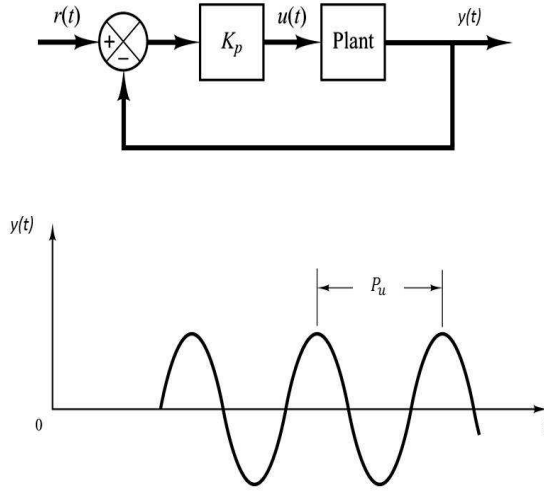


Fig. 4. Illustration of the continuous cycling method

PID control design methods

Several techniques are used

- Model based design methods
- Computer simulations
- Online tuning

Model and simulation-based methods use approximation of the plant model. Online tuning methods are experimental. Methods based on the plant model allow for initial setting of the controller. Computer simulation (Matlab, Labview and many other platforms) allow for comparison between the different alternatives.

Experimental online tuning

Experimental online tuning is also called field tuning. It is based on testing, trial and error. Having a good initial model can be very helpful in experimental tuning because it makes the tuning process more straightforward and less time consuming. Here we focus our discussions on the continuous cycling and the step reactive curve methods.

Continuous cycling

Introduced by Ziegler and Nichols in 1942, it is based on trial and error. After the system has reached its steady state, the integral and proportional actions are eliminated. The experiment's configuration is shown in figure 4-top. The gain K_p is set to a small value and then increased slowly until sustained oscillations with constant amplitude occur. The numerical value of the gain for which sustained oscillations with constant amplitude occur is called the ultimate gain (K_{pu}), its corresponding period is called the ultimate period (P_u). Once K_{pu} and P_u are determined, the table below is used to calculate the PID parameters. Illustration of the continuous cycling method is shown in figure 4.

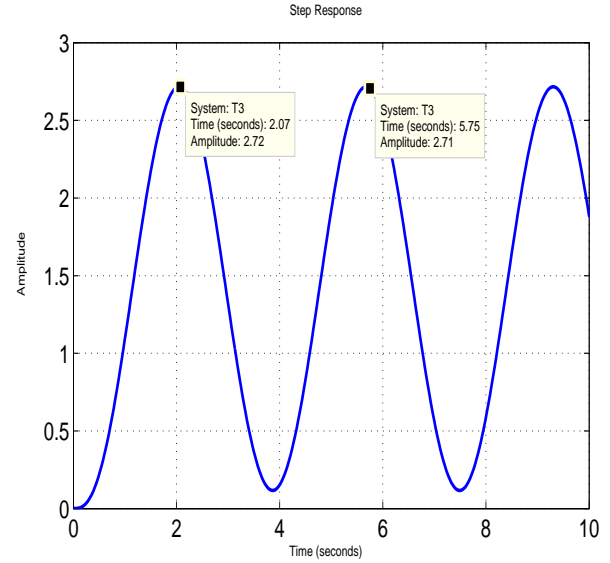


Fig. 5. Sustained oscillations for the continuous cycling method

$$\begin{bmatrix} \text{Ziegler - Nichols} & K_p & \tau_I & \tau_D \\ P & 0.5K_{pu} & - & - \\ PI & 0.45K_{pu} & 0.83P_u & - \\ PID & 0.6K_{pu} & 0.5P_u & 0.125P_u \end{bmatrix} \quad (17)$$

Example

Design a PID control for the system whose response is shown in figure 5. The ultimate gain for the sustained oscillations is 6.

It is clear from the oscillations that $P_u = 3.68s$. Therefore

$$K_p = 0.6K_{pu} = 3.6 \quad (18)$$

$$\tau_I = P_u/2 = 1.8 \quad (19)$$

$$\tau_D = P_u/8 = 0.45 \quad (20)$$

The main drawback of this method is that it pushes the system to the limit of stability, which can result in hazardous situations.

Astrom and Hagglund tuning method

This method is an effective alternative to the continuous cycling method. It uses an on-off controller with dead-zone. After the on-off controller is connected to the plant, the closed loop system exhibits sustained oscillations. The principle of the method is illustrated in figure 6. Astrom and Hagglund derived an approximation of the ultimate gain as follows

$$K_{pu} = \frac{4d}{\pi a} \quad (21)$$

where a, d are shown in figure 7. Once K_{pu}, P_u are determined, the Ziegler-Nichols table (17) can be used to calculate the controller parameters. One important advantage of the Astrom and Hagglund method is that it can be easily automated.

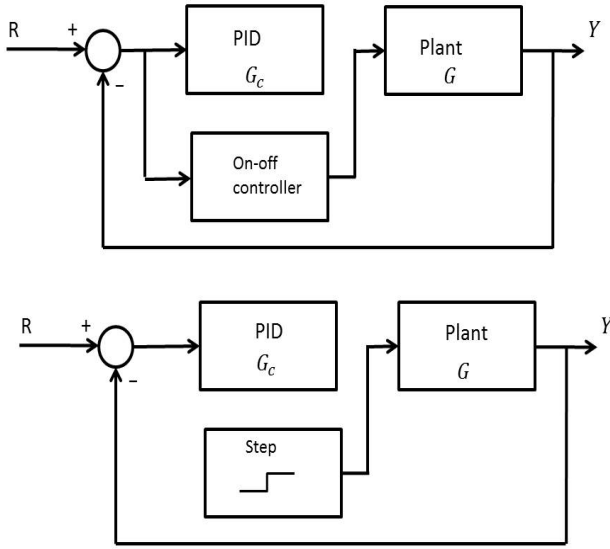


Fig. 6. Top: Astrom and Hagglund tuning method and bottom: step test method

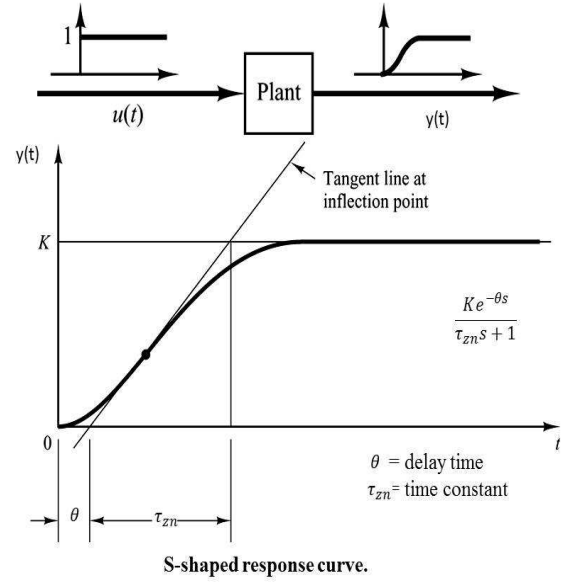


Fig. 8. Experiment setting and step response reaction curve

where K, τ_{zn}, θ can be obtained from the reaction curve. Let us define

$$K_0 = \frac{\tau_{zn}}{K\theta} \quad (23)$$

The PID parameters can be determined from the table below

Ziegler – Nichols	K_p	τ_I	τ_D
P	K_0	---	---
PI	$0.9K_0$	3.3θ	---
PID	$1.2K_0$	2θ	0.5θ

(24)

Example

Consider the reaction curve shown in figure 9. The goal is derive the PID control parameters. From the graph we have: $K = 3, \theta = 2, \tau_{zn} = 1$, from which we get

$$K_p = 0.2 \quad (25)$$

$$\tau_I = 4 \quad (26)$$

$$\tau_D = 1 \quad (27)$$

Additional reading

Model based design methods

Two methods are discussed here: the direct synthesis method and the internal model control.

Direct synthesis method

The controller is designed based on the desired closed loop transfer function. The method can be used to design but not limited to PID controllers. The closed loop transfer function is given by

$$\frac{Y(s)}{R(s)} = \frac{G_c(s)G(s)}{1 + G_c(s)G(s)} \quad (28)$$

where

$$\tilde{G} = \frac{Ke^{-\theta s}}{\tau_{zn}s + 1} \quad (22)$$

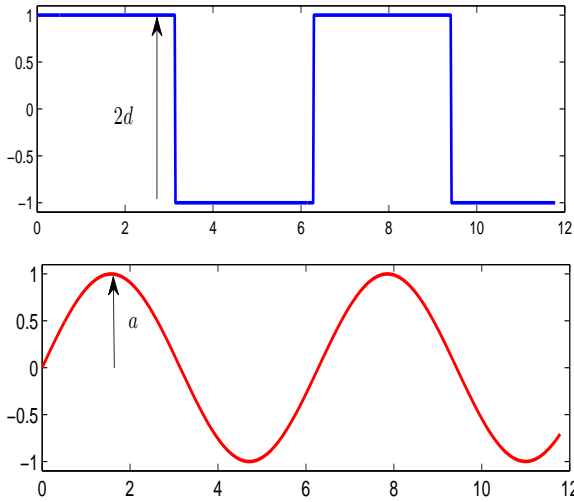


Fig. 7. Top: on-off controller output signal and bottom: sustained oscillations response

Step test method

This is another method proposed by Ziegler and Nichols in 1942. This method is also called the plant reaction curve or Ziegler and Nichols open loop method. After the system has reached its steady state, a small step change is introduced, the plant reaction curve is obtained as shown in figure 8-bottom. The controller parameters are derived based on the reaction curve parameters using table 24. The open loop transfer function whose response is shown in figure 8-bottom is

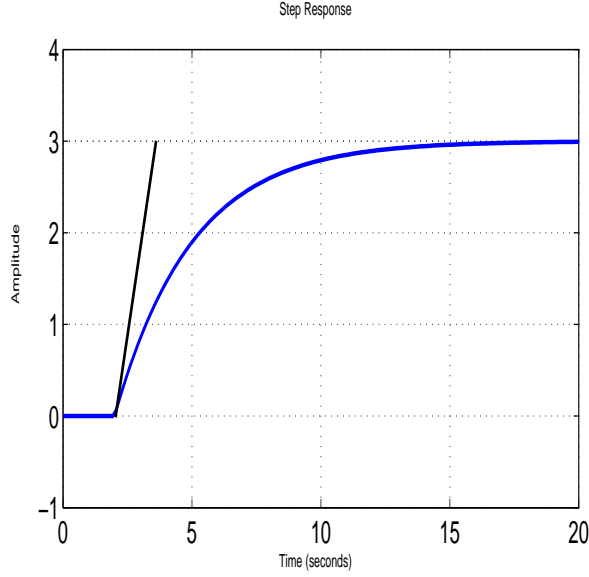


Fig. 9. Step test reaction curve

- $G_c(s)$ is the controller transfer function
- $\tilde{G}(s)$ is the open loop transfer function.

Rearranging the terms and solving for the controller transfer function yields the following equation

$$G_c(s) = \frac{1}{\tilde{G}(s)} \frac{\frac{Y(s)}{R(s)}}{1 - \frac{Y(s)}{R(s)}} \quad (29)$$

In order to solve for the controller's transfer function we need to know:

- the desired closed loop transfer function
- the open loop transfer function.

We can define a desired closed loop transfer function based on the desired characteristics of the system. We call this function

$$\left(\frac{Y(s)}{R(s)}\right)_d \quad (30)$$

Assuming we can determine an approximation \tilde{G} of the open loop transfer function, the controller equation becomes

$$G_c(s) = \frac{1}{\tilde{G}(s)} \frac{\left(\frac{Y(s)}{R(s)}\right)_d}{1 - \left(\frac{Y(s)}{R(s)}\right)_d} \quad (31)$$

Several possibilities exist for the desired closed loop system, Ideally we want the closed loop response to be equal to the input, that is

$$\left(\frac{Y(s)}{R(s)}\right)_d = 1 \quad (32)$$

This is too perfect and is not possible in practice. A more realistic approach is to use a first order system.

First order system

In this case the desired closed loop system is a simple first order system

$$\left(\frac{Y(s)}{R(s)}\right)_d = \frac{1}{\tau_c s + 1} \quad (33)$$

where τ_c is the closed loop time constant. The controller equation is reduced to the following equation

$$G_c = \frac{1}{\tilde{G}(s)} \frac{1}{\tau_c s} \quad (34)$$

If the process has a delay, we can use the closed loop transfer function with delay

$$\left(\frac{Y(s)}{R(s)}\right)_d = \frac{e^{-\theta s}}{\tau_c s + 1} \quad (35)$$

where θ is a time delay. The equation for the controller is

$$G_c = \frac{1}{\tilde{G}(s)} \frac{e^{-\theta s}}{\tau_c s + 1 + e^{-\theta s}} \quad (36)$$

Assuming a small delay, the delay can be approximated as

$$e^{-\theta s} = 1 - \theta s \quad (37)$$

and the controller as

$$G_c = \frac{1}{\tilde{G}(s)} \frac{e^{-\theta s}}{(\tau_c + \theta)s} \quad (38)$$

Clearly, equations (34) and (38) depend on the approximation of the open loop system. This discussed below.

A. Open loop transfer function approximation

Several methods exist to approximate the open loop transfer function such as the step response and the frequency response. Here we consider first and second order approximations with delay.

B. First order approximation

The open loop system is approximated by a first order transfer function as follows

$$\tilde{G}(s) = \frac{K e^{-\theta s}}{(\tau s + 1)} \quad (39)$$

where τ is the open loop time constant. Using equation (38) and knowing that the controller's equation is

$$G_c = K_p \left(1 + \frac{1}{\tau_I s}\right) \quad (40)$$

we get

$$K_p = \frac{\tau}{K(\tau_c + \theta)} \quad (41)$$

$$\tau_I = \tau \quad (42)$$

C. Second order approximation

The transfer function is approximated by a second order transfer function as follows

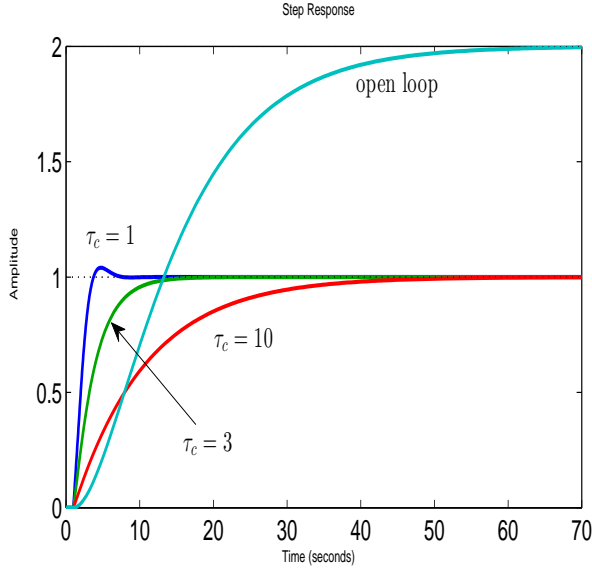


Fig. 10. Closed loop response for different values of the time constant

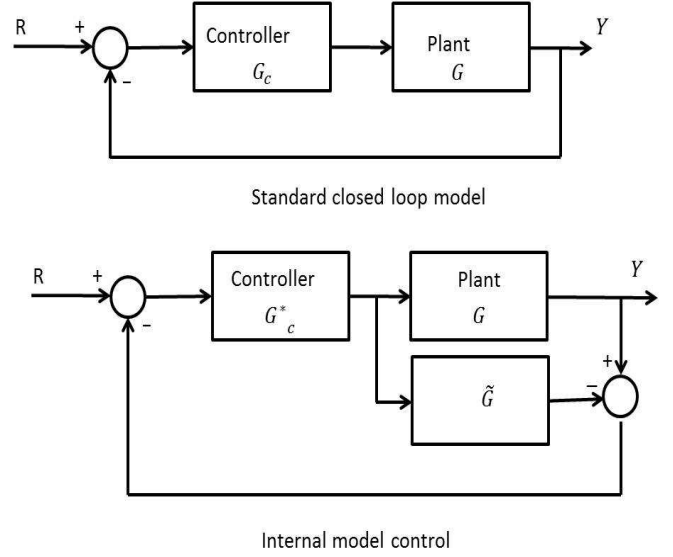


Fig. 11. Standard and IMC models

$$\tilde{G}(s) = \frac{Ke^{-\theta s}}{(\tau_1 s + 1)(\tau_2 s + 1)} \quad (43)$$

By substituting in the controller equation we get

$$K_p = \frac{\tau_1 + \tau_2}{K(\tau_c + \theta)} \quad (44)$$

$$\tau_I = \tau_1 + \tau_2 \quad (45)$$

$$\tau_D = \frac{\tau_1 \tau_2}{\tau_1 + \tau_2} \quad (46)$$

where

$$G_c = K_p \left(1 + \frac{1}{\tau_I s} + \tau_D s\right) \quad (47)$$

Example

Consider the following approximation of the open loop transfer function

$$\tilde{G}(s) = \frac{2e^{-1s}}{(10s + 1)(5s + 1)} \quad (48)$$

Design a PID controller when $\tau_c = 1s, 3s, 10s$ and simulate the system behavior.

- $\tau_c = 1s \Rightarrow K_p = 3.7500, \tau_I = 15; \tau_D = 3.3333$
- $\tau_c = 3s \Rightarrow K_p = 1.8750, \tau_I = 15; \tau_D = 3.3333$
- $\tau_c = 10s \Rightarrow K_p = 0.68, \tau_I = 15; \tau_D = 3.3333$

A simulation is shown in shown in figure 10

Internal Model Control

The IMC method uses a simplified model of the control loop where another controller called IMC controller is introduced. The standard control loop and the IMC model are shown in figure 11. The IMC controller is denoted by G_c^* . The first step is to establish the relationship between the two

controllers. Assuming the block diagrams are equivalent, it is possible to write

$$G_c = \frac{G_c^*}{1 - G_c^* \tilde{G}} \quad (49)$$

There is one to one relationship between G_c and G_c^* . This implies that for each IMC controller there is a corresponding standard controller. The IMC method first finds an expression for G_c^* and then derives an equation for G_c . This is done in two steps:

- Step 1: The open loop model is written as

$$\tilde{G} = (\tilde{G}^+)(\tilde{G}^-) \quad (50)$$

where \tilde{G}^+ contains any time delays and right half plane zeros.

- Step 2: The IMC controller is derived using the following equation

$$\tilde{G}_c^* = \frac{1}{(\tilde{G}^-)(\tau_c s + 1)^r} \quad (51)$$

where τ_c is the closed loop time constant and r is a positive integer that characterizes the system order

Example

Use IMC to design a PID controller for the following system

$$\tilde{G}(s) = \frac{Ke^{-1s}(-1 + s)}{(0.5s + 1)(3s + 1)} \quad (52)$$

We have

$$\tilde{G}^-(s) = \frac{K}{(0.5s + 1)(3s + 1)} \quad (53)$$

for $r = 1$, we have

$$G_c^*(s) = \frac{(0.5s + 1)(3s + 1)}{K(\tau_c s + 1)} \quad (54)$$

After rearranging the terms, we get

$$K_p = \frac{13}{K(4\tau_c + 1)} \quad (55)$$

$$\tau_I = 3.5 \quad (56)$$

$$\tau_D = 3/13 \quad (57)$$

It is important to note that the choice of τ_c plays an important role. High values of τ_c result in a more conservative controller and low values result in more aggressive controller.