Xi'an Jiaotong-Liverpool University
西交利物浦大学

EEE220 Instrumentation and Control System

2018-19 Semester 2

Dr. Qing Liu

Email: qing.liu@xjtlu.edu.cn

Office: EE516

Department of Electrical and Electronic Engineering

2 May, 2019

Lecture 20

Outline

The Root Locus Method

- ❑ The Root Locus Concept
- ❑ The Root Locus Procedure
- ❑ The Root Locus Using Matlab
- ❑ Parameter Design by the Root Locus Method
- ❑ **PID Controllers**
  - **Concept**
  - **PID Tuning**
- ❑ **Design Examples**

PID Controllers

One form of controller widely used in industrial process control is the three-term, PID Controller. This controller has a transfer function

$$\boxed{\boldsymbol{G_c(s) = K_P + \frac{K_I}{s} + K_D s}}$$

For $E(s) = R(s) - Y(s)$, output is

$$U(s) \ = K_P E(s) \ + \frac{K_I}{s} E(s) + K_D s E(S)$$

Applying the inverse Laplace transform, the equation for the output in the time domain is

$$u(t) = \underbrace{K_P}_{\textbf{Proportional}} e(t) + \underbrace{K_I}_{\textbf{Integral}} \int_0^t e(\tau)\, dt + \underbrace{K_D}_{\textbf{Derivative}} \frac{de(t)}{dt}$$

The three-term controller is called a PID controller because it contains a **P**roportional, an **I**ntegral, and a **D**erivative term represented by K_P, K_I, K_D respectively.

Table 7.4 Effect of Increasing the PID Gains K_p, K_D, and K_I on the Step Response

PID Gain	Percent Overshoot	Settling Time	Steady-State Error
Increasing K_P	Increases	Minimal impact	Decreases
Increasing K_I	Increases	Increases	Zero steady-state error
Increasing K_D	Decreases	Decreases	No impact

- **Proportional plus integral (PI) controller** (set $K_D = 0$):

$$\boxed{G_c(s) = K_P + \frac{K_I}{s}}$$

- **Proportional plus derivative (PD) controller** (set $K_I = 0$):

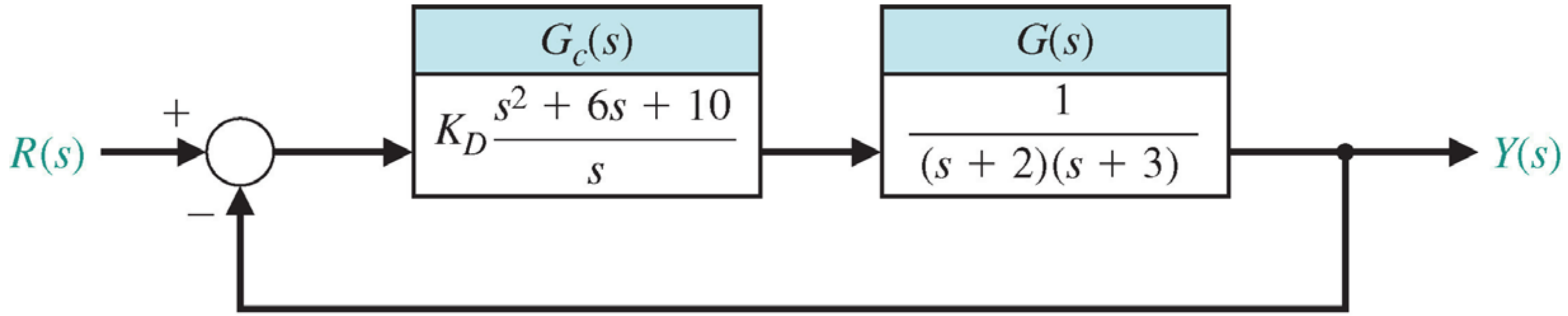
$$\boxed{G_c(s) = K_P + K_D s}$$

How a PID Controller Works

$$G_c(s) = K_p + \frac{K_I}{s} + K_D s = \frac{K_D\left(s^2 + \frac{K_P}{K_D}s + \frac{K_I}{K_D}\right)}{s}$$

A PID controller adds two zeros and one pole ($s = 0$) to the open-loop system.

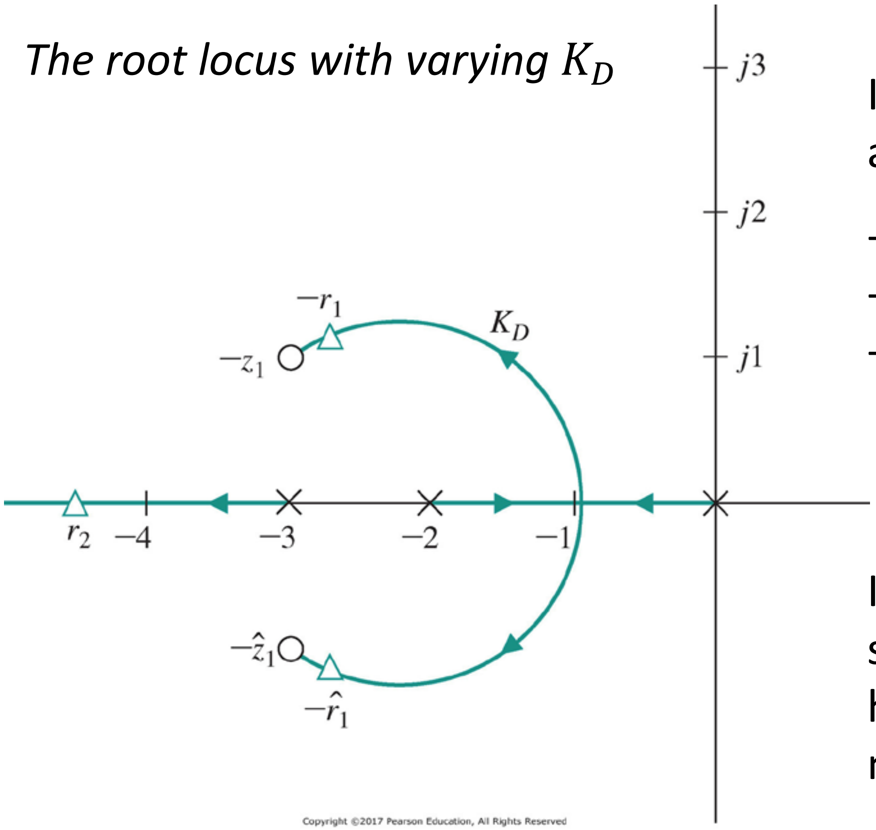
Example:



G_c in this system is actually a PID controller.

$$\Delta(s) = 1 + G(s)G_c(s) = 1 + \frac{K_D(s+z_1)(s+\hat{z}_1)}{s(s+2)(s+3)}$$

The root locus with varying K_D



If choose a value of K_D corresponding to $-r_1$ and -$\hat{r}_1$, then for a step input:

- the percent overshoot will be $P.O. \leq 2\%$,
- the steady-state error will be $e_{ss} = 0$;
- the settling time will be approximately $T_s = 1s$.

If a shorter settling time is desired, then we select $-z_1$ and $-\hat{z}_1$ to lie further in the left-hand s-plane and set K_D to drive the roots near the complex zeros.

PID Tuning

- The popularity of PID controllers can be attributed partly to their good performance over a wide range of operating conditions and partly to their functional simplicity that allows engineers to operate them in a simple, straightforward manner.
- To implement the PID controller, three parameters must be determined, the proportional gain K_P, integral gain K_I and derivative gain K_D.
- There are many methods available to determine acceptable values of the PID gains. The process of determining the gains is often called **PID tuning**.
- A common approach to tuning is to use manual PID tuning methods, whereby the PID control gains are obtained by trial-and-error with minimal analytic analysis using step responses obtained via simulation, or in some cases, actual testing on the system and deciding on the gains based on observations and experience.

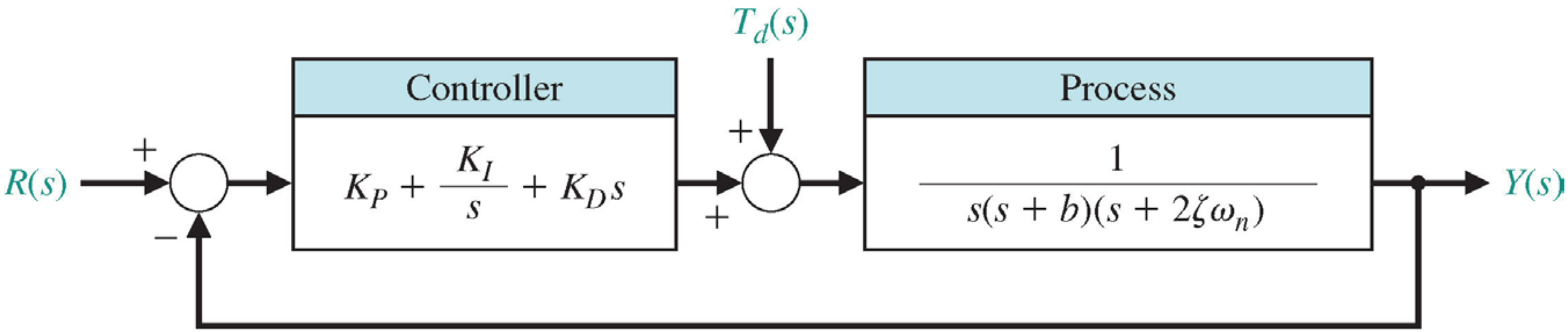
Manual Tuning Method

One approach to manually tuning is:

1. Set $K_I = 0$ and $K_\mathrm{D} = 0$;
2. Slowly increase the gain K_P until the output of the closed-loop system oscillates just on the edge of instability; - this can be done either in simulation or on the actual system if it cannot be taken off-line;
3. Then reduce the value of K_P to achieve what is known as the **quarter amplitude decay**, that is, the amplitude of the closed-loop response is reduced approximately to one-fourth of the maximum value in one oscillatory period. A rule-of-thumb is to start by reducing the proportional gain K_P by one-half;
4. Increase K_I and K_D manually to achieve a desired response.

Example 20.1

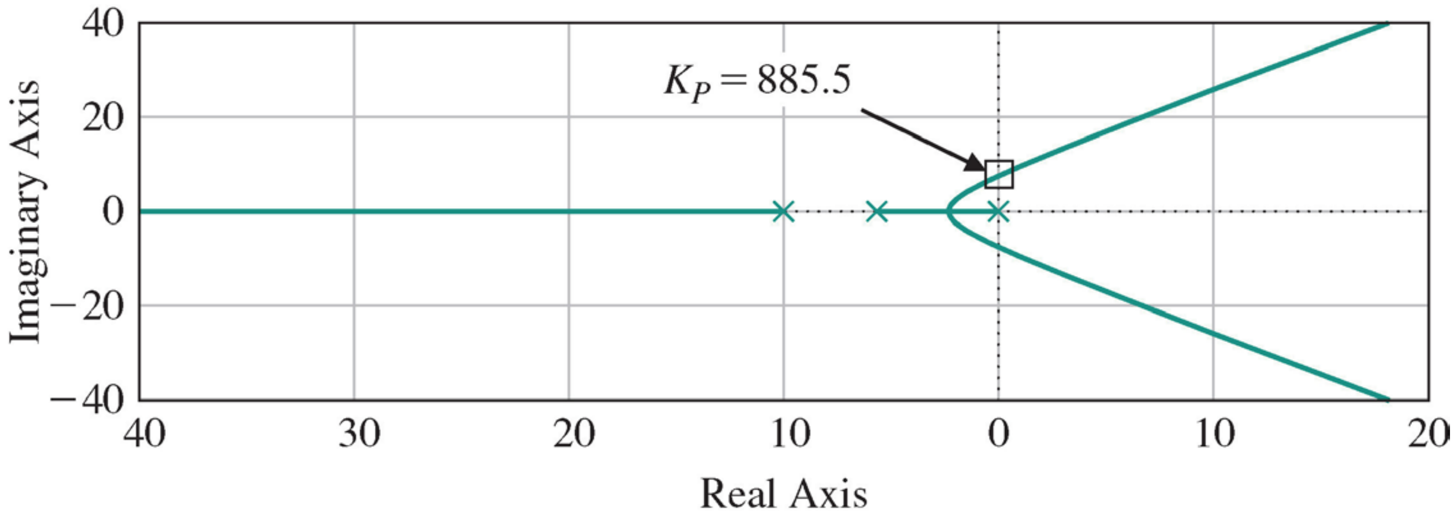
For the following system, where $b = 10, \zeta = 0.707$ and $\omega_n = 4$. Design the PID controller so that the percent overshoot $P.O. \leq 15\%$ and settling time with 2% criterion $T_s < 3s$.

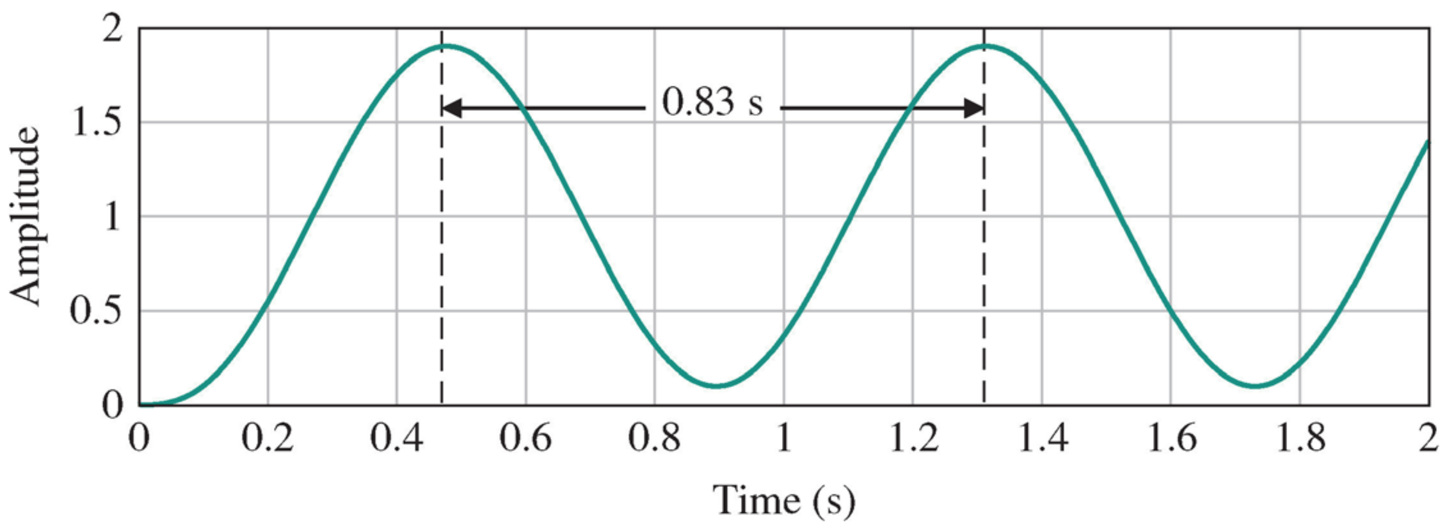


Solutions:

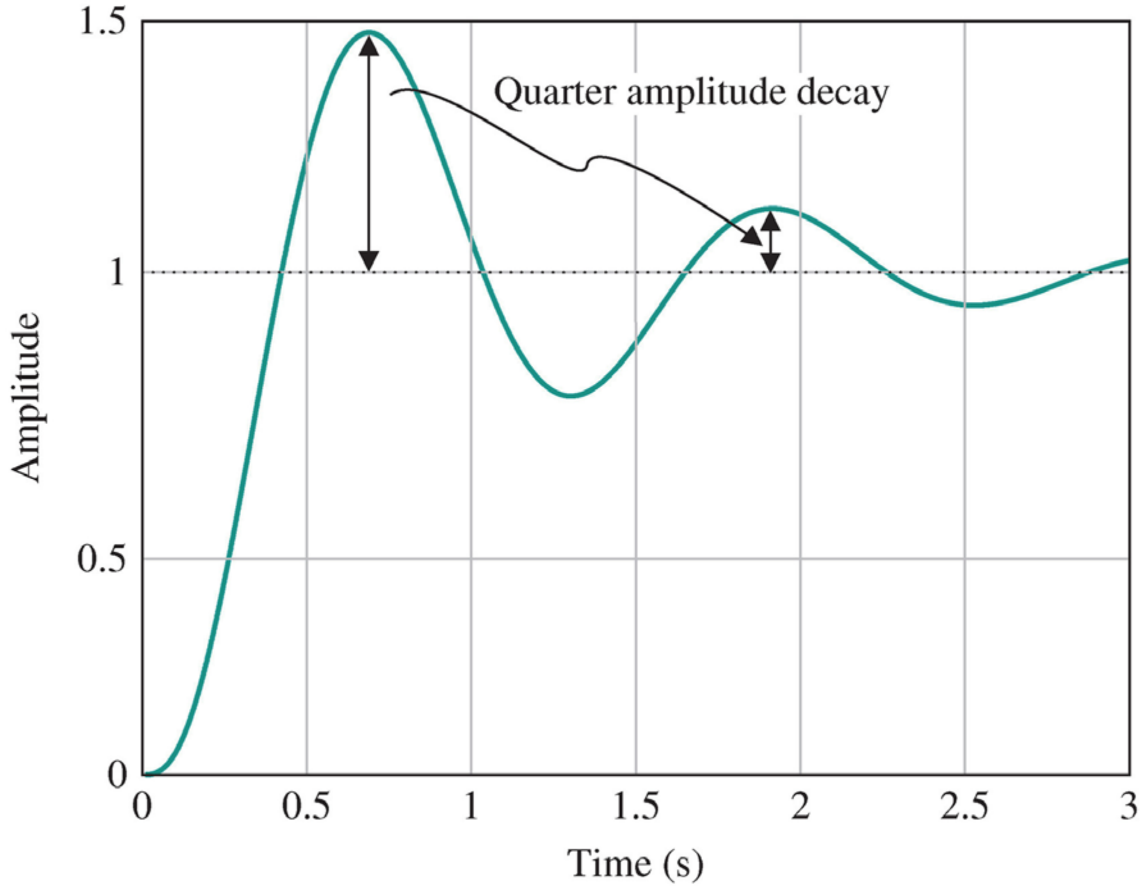
- Set $K_I = 0$ and $K_{\mathrm{D}} = 0$; find the gain on the edge of instability using Routh-Hurwitz Criterion. The root locus with varying K_P can be also sketched.

$$1 + K_P\left[\frac{1}{s(s+10)(s+5.66)}\right] = 0.$$


$K_P = 885.5$
40
20
0
−20
−40
Imaginary Axis
40
30
20
10
0
10
20
Real Axis


0.83 s
2
1.5
1
0.5
0
Amplitude
0
0.2
0.4
0.6
0.8
1
1.2
1.4
1.6
1.8
2
Time (s)

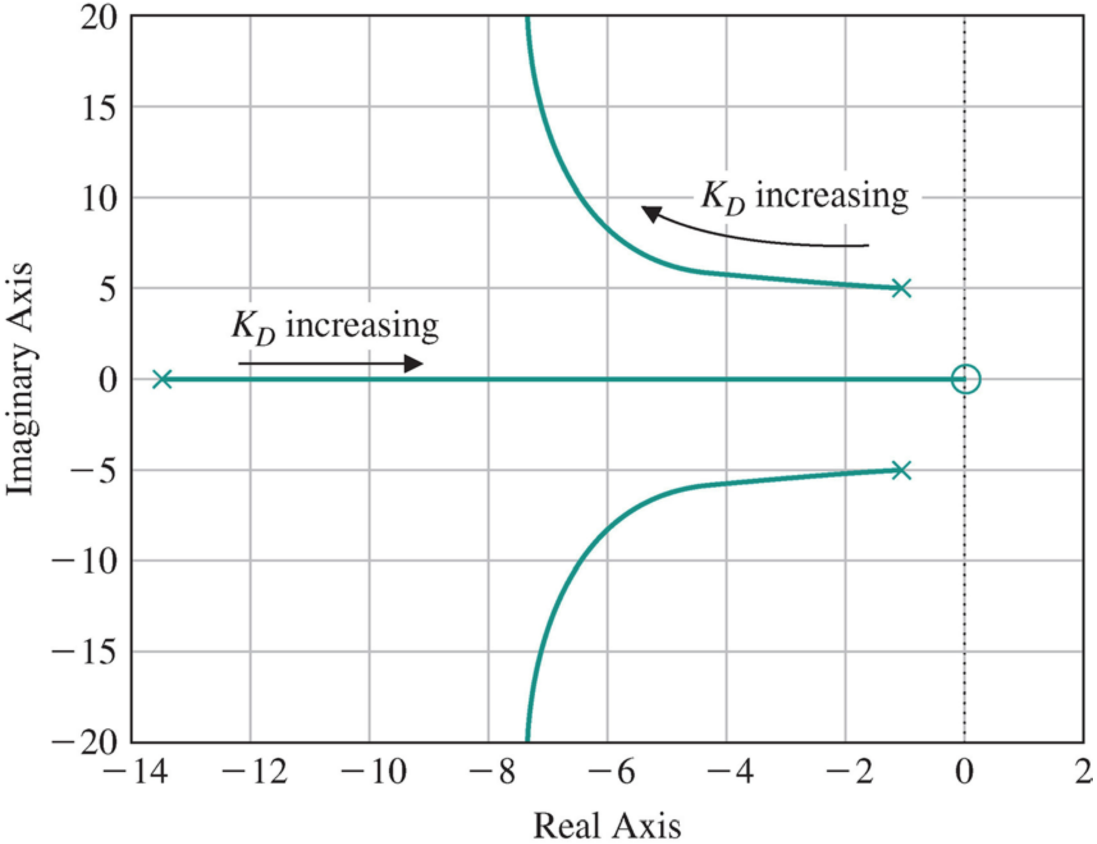
- Reduce $K_P = 885.5$ by half (start from $K_P = 442.75$) as a first step then keep reducing K_P to achieve a step response with approximately a quarter amplitude decay.



$K_P = 370$

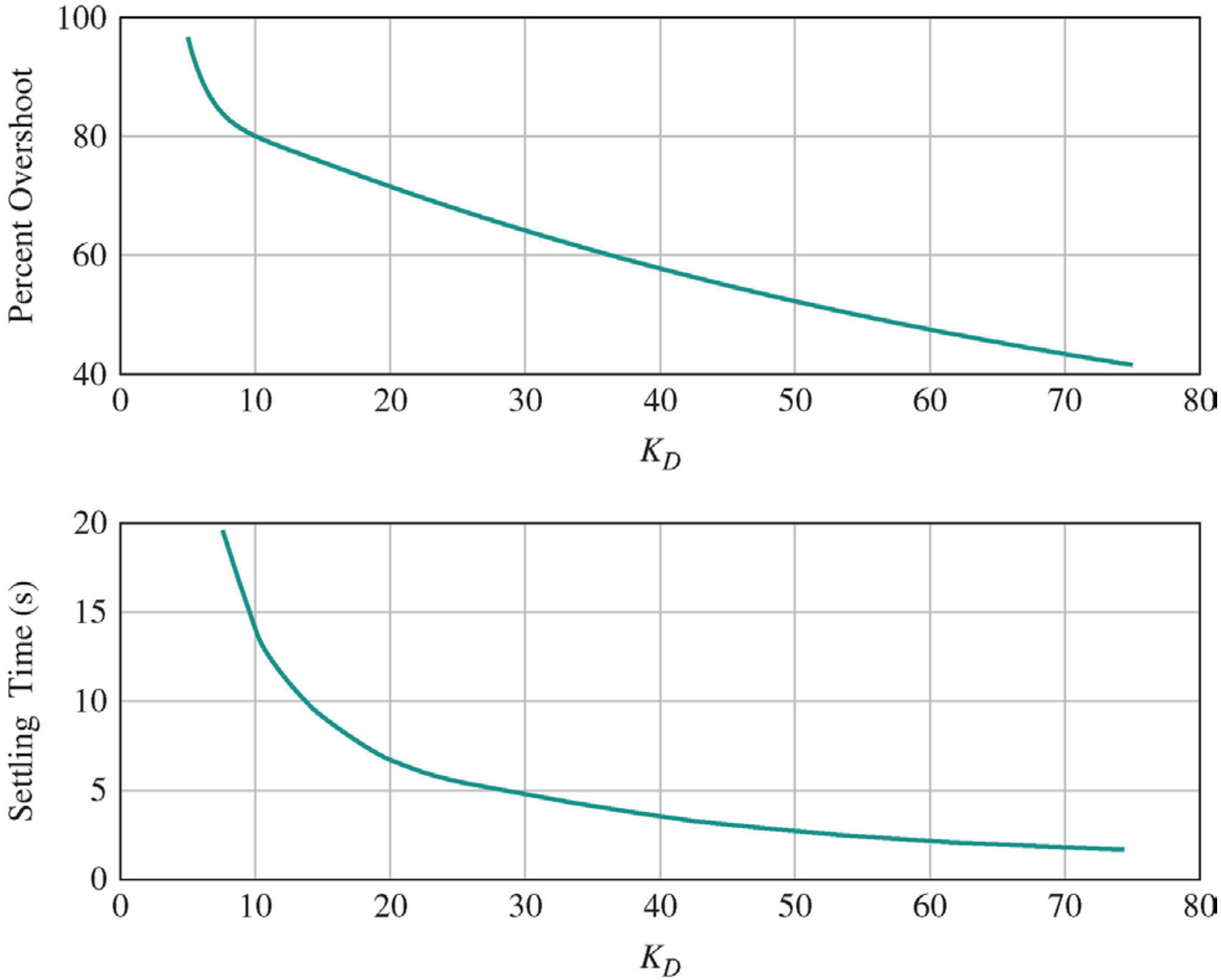
- Set $K_P = 370$ and $K_I = 0$, sketch the root locus with varying K_D.

$$1 + K_D\left[\frac{s}{s(s+10)(s+5.66)+K_P}\right] = 0.$$



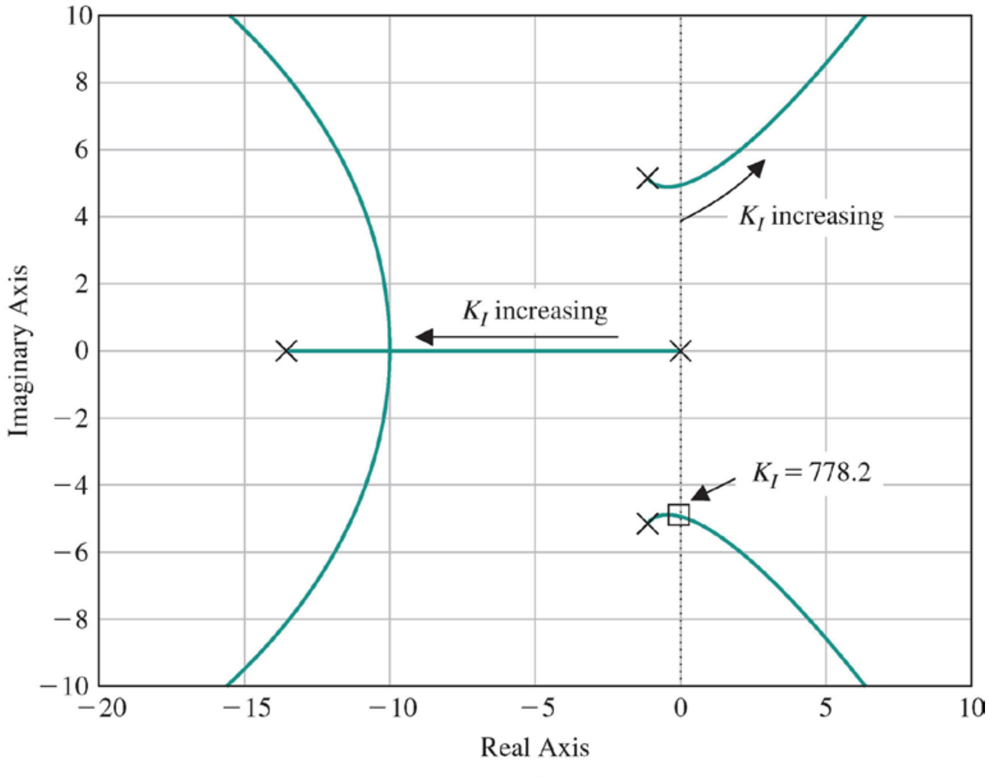
- As K_D increase, associated damping ratio increases and thereby decreases P.O.; the complex poles to the left also increases $\zeta\omega_n$, thereby reducing the settling time;
- Until $K_D = 75$, when $K_D > 75$, the real root begins dominant the response.

Percent overshoot and settling time with $K_P = 370$, $K_I = 0$, and $5 \leq K_D < 75$.



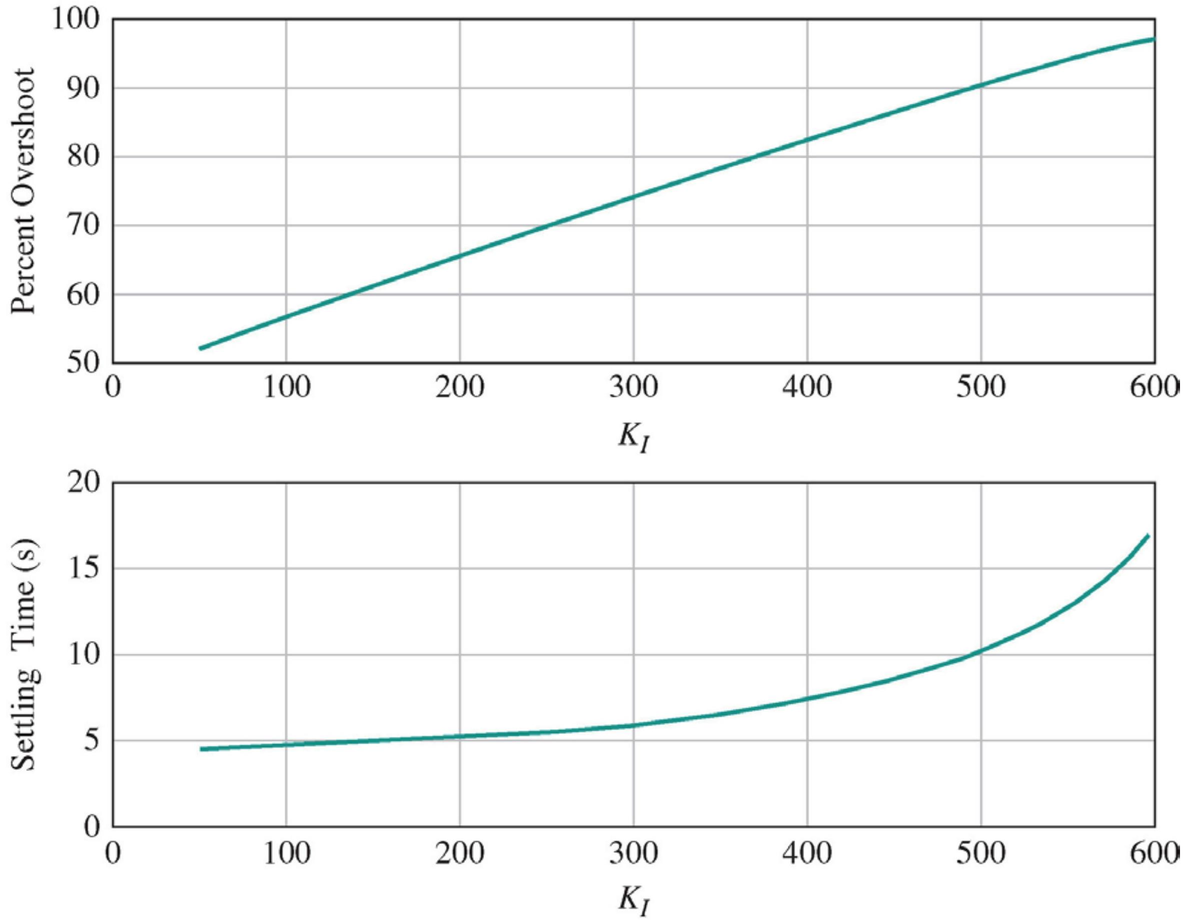
- Set $K_P = 370$ and $K_D = 0$, sketch the root locus with varying K_I.

$$1 + K_I\left[\frac{1}{s\left(s(s+10)(s+5.66)+K_P\right)}\right] = 0.$$

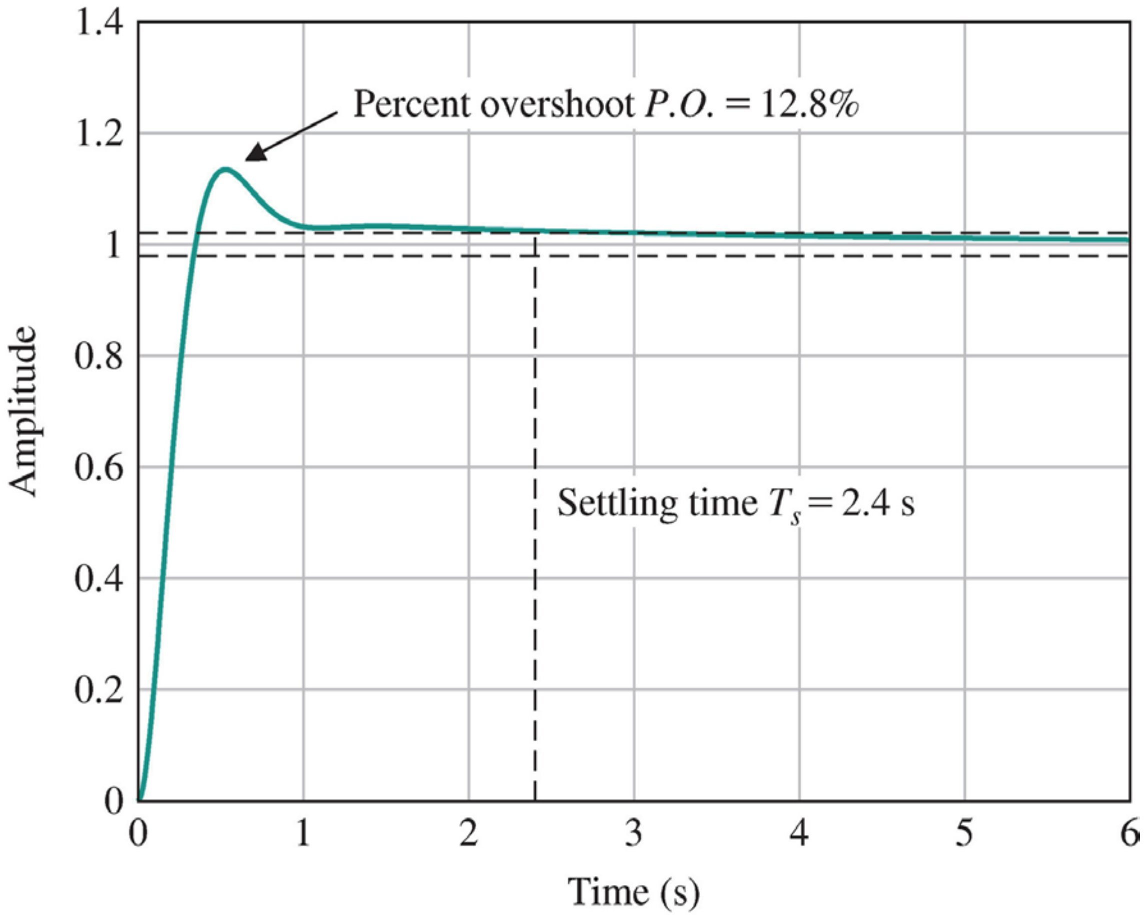


- As K_I increase, the complex pair poles move right, this increases the P.O. and increases the settling time;
- In fact, when $K_I = 778.2$, the system will become marginally stable.

Percent overshoot and settling time with $K_P = 370$, $K_D = 0$, and $50 \le K_I < 600$.



Finally, we choose $K_P = 370, K_D = 60, K_I = 100$, the step response of $P.O. = 12.8\%$ and $T_s = 2.4s$ is achieved.



Ziegler-Nichols PID Tuning Method

- Two important PID controller gain tuning methods were published in 1942 by John G. Ziegler and Nathaniel B. Nichols **intend to achieve a fast closed-loop step response without excessive oscillations and excellent disturbance rejection.** The two approaches are classified under the general heading of Ziegler-Nichols tuning methods.

- The Ziegler-Nichols tuning methods are based on assumed forms of the models of the process, but the models do not have to be precisely known. This makes the tuning approach very practical in process control applications.

- It is suggested to consider the Ziegler-Nichols rules to obtain initial controller designs followed by design iteration and refinement.

- Remember that the Ziegler-Nichols rules will not work with all plants or processes.

The Closed-loop Ziegler-Nichols Tuning Method considers the closed-loop system response to a step input (or step disturbance) with the PID controller in the loop:

1. Set $K_I = 0$ and $K_D = 0$;
2. Increase K_P (in simulation or on the actual system) until the closed-loop system reaches the boundary of instability. The gain on the border of instability, denoted by $\boldsymbol{K_U}$, is called the **ultimate gain**. The period of the sustained oscillations, denoted by $\boldsymbol{T_U}$, is called the **ultimate period**.
3. Once K_U and P_U are determined, the PID gains are computed using the following relationship.

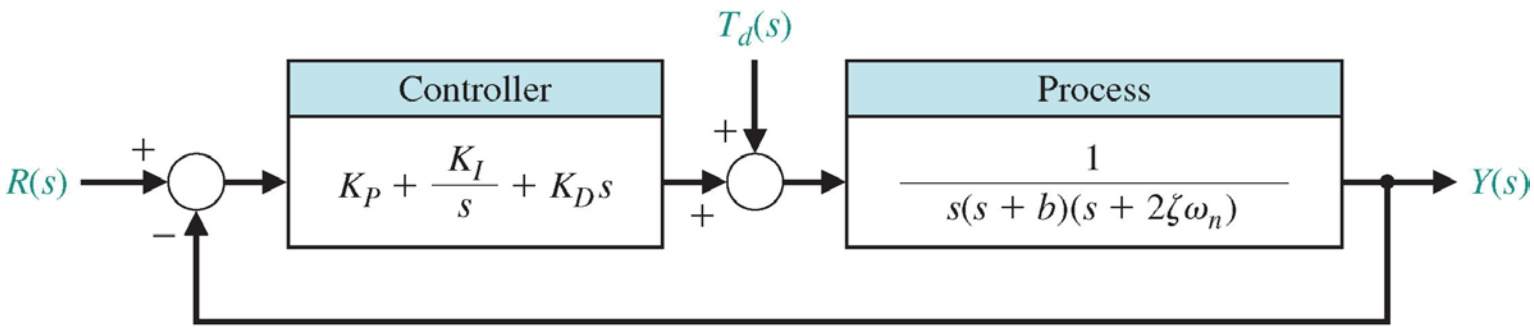
Table 7.7 Ziegler-Nichols PID Tuning Using Ultimate Gain, K_U, and Oscillation Period, P_U

Ziegler-Nichols PID Controller Gain Tuning Using Closed-loop Concepts

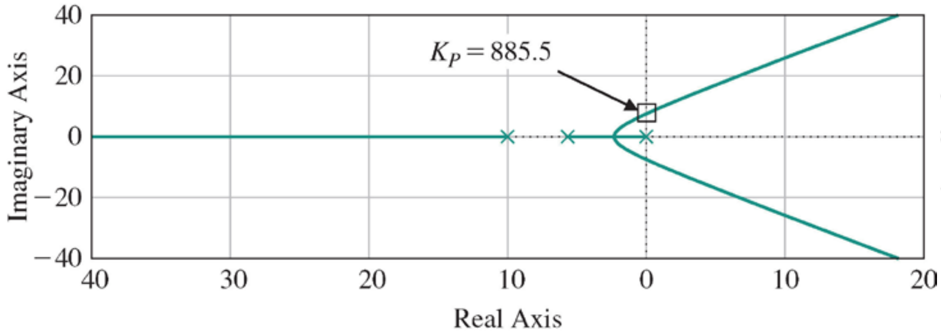
Controller Type	K_P	K_I	K_D
Proportional (P) $G_c(s) = K_P$	$0.5K_U$	–	–
Proportional-plus-integral (PI) $G_c(s) = K_P + \frac{K_I}{s}$	$0.45K_U$	$\frac{0.54K_U}{T_U}$	–
Proportional-plus-integral-plus-derivative (PID) $G_c(s) = K_P + \frac{K_I}{s} + K_D s$	$0.6K_U$	$\frac{1.2K_U}{T_U}$	$\frac{0.6K_U T_U}{8}$

Example 20.2

Re-consider the previous example.

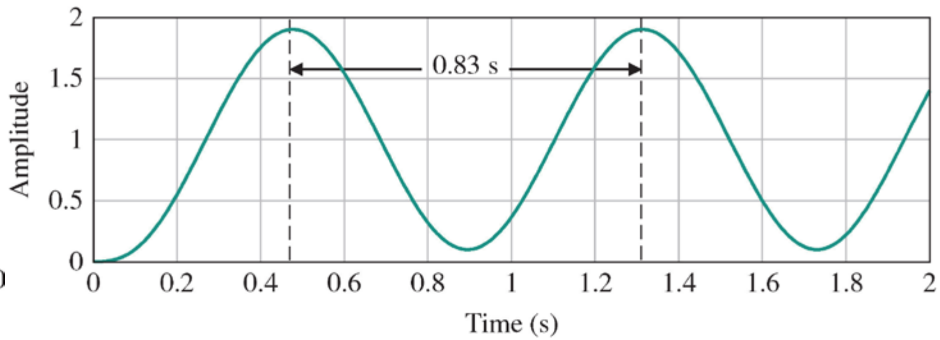


Root Locus with varying K_P
($K_D = K_I = 0$)



$$K_U = 885.5$$

Step Response

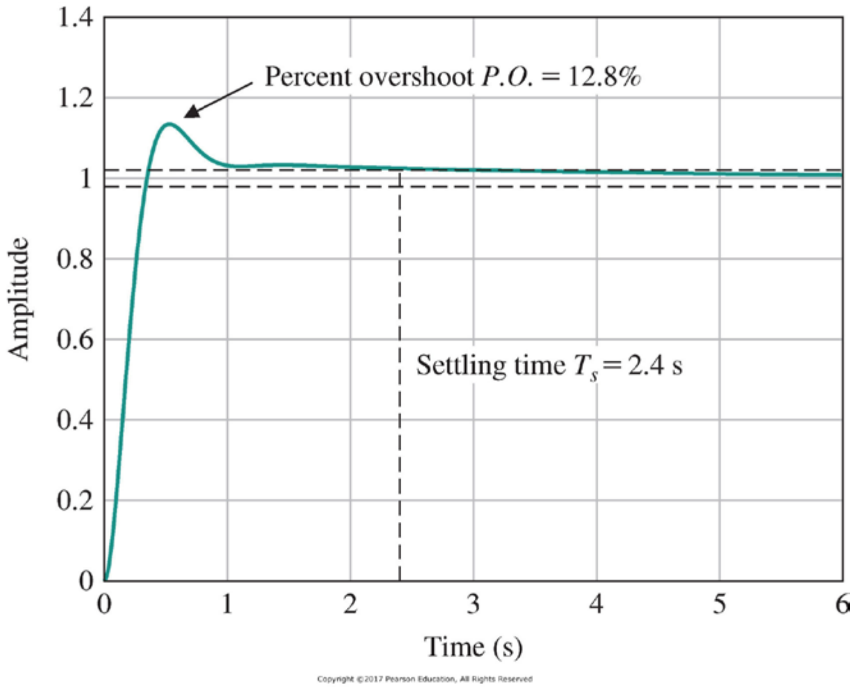


$$T_U = 0.83s$$

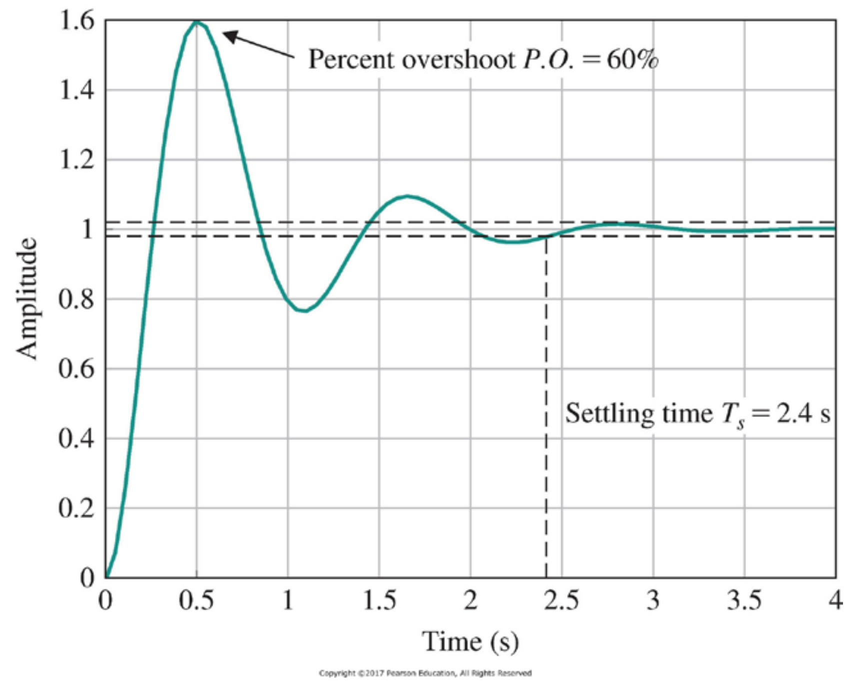
By using the Ziegler-Nichols formulas we obtain

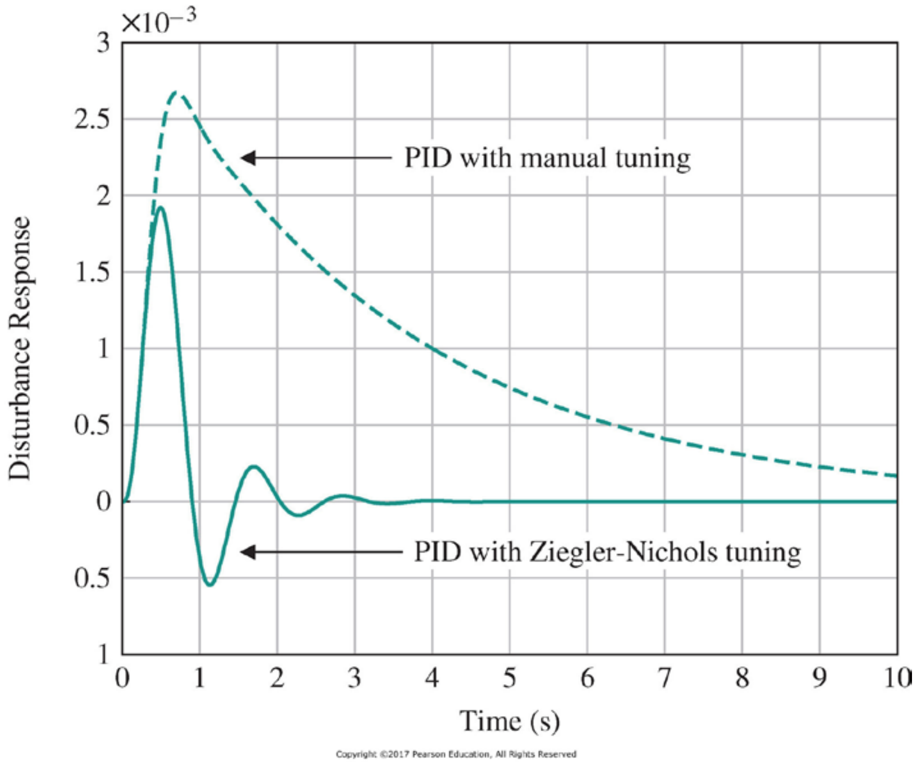
$$K_P = 0.6K_U = 531.3, \qquad K_I = \frac{1.2K_U}{T_U} = 1280.2, \qquad \text{and } K_D = \frac{0.6K_U T_U}{8} = 55.1$$

Time response for Manual Tuning



Time response for the Ziegler–Nichols Tuning



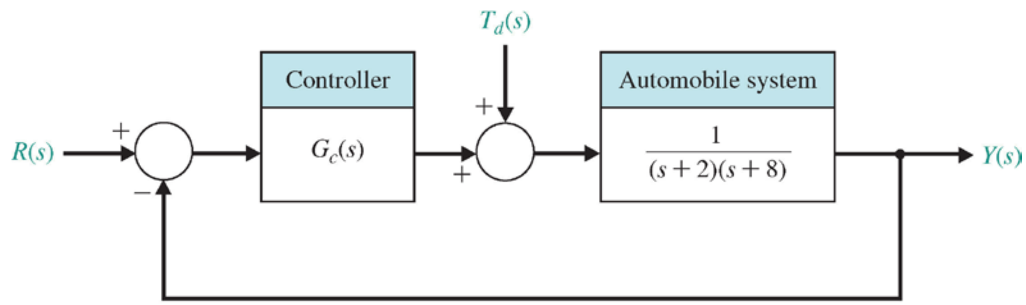


Disturbance response for the Ziegler–Nichols PID tuning versus the manual tuning.

- We see that the step disturbance performance of the Ziegler-Nichols PID controller is indeed better than the manually tuned controller;
- While Ziegler-Nichols approach provides a structured procedure for obtaining the PID controller gains, the appropriateness of the Ziegler-Nichols tuning depends on the requirements of the problem under investigation.

Design Example 20.3: Automobile Velocity Control

A velocity control system for maintain the velocity between the two automobiles are shown as follows:



Control Goal:

- Maintain the prescribed velocity between the two vehicles, and maneuver the active vehicle as command.

Variable to Be Controlled:

- The relative velocity between vehicles, denoted by $y(t)$.

Design specifications:

- DS1. Zero steady-state error to a step input.
- DS2. Steady-state error due to a ramp input of $e_{ss} \leq 25\%$ of the input magnitude.
- DS3. Percent overshoot of $P.O. \leq 5\%$ to a step input.
- DS4. Settling time of $T_s \leq 1.5\ s$ to a step input (using 2% criterion).

Solutions:

Step 1.

$G(s)$ is a type 0 system. To guarantee a zero steady-state error to a step input (DS1), the controller needs to increase the system type by at least 1. That is, a controller with one **integrator**.

We consider the **PI** controller $G_c = K_P + \frac{K_I}{s} = K_P \frac{s+\frac{K_I}{K_P}}{s}$.

To meet DS2, it requires that

$$e_{ss} = \lim_{s\to 0} s \cdot \frac{1}{1+G_c(s)G(s)} \cdot \frac{1}{s^2} = \lim_{s\to 0} \frac{(s+2)(s+8)}{K_P(s+\frac{K_I}{K_P})} = \frac{16}{K_I} \leq 25\%$$

Therefore, the integral gain must satisfy

$$K_I \geq 64$$

Step 2.

Consider values of K_P and K_I to make the system stable. The closed-loop characteristic equation is

$$\Delta(s) = s^3 + 10s^2 + (16 + K_P)s + K_I$$

The Routh array is

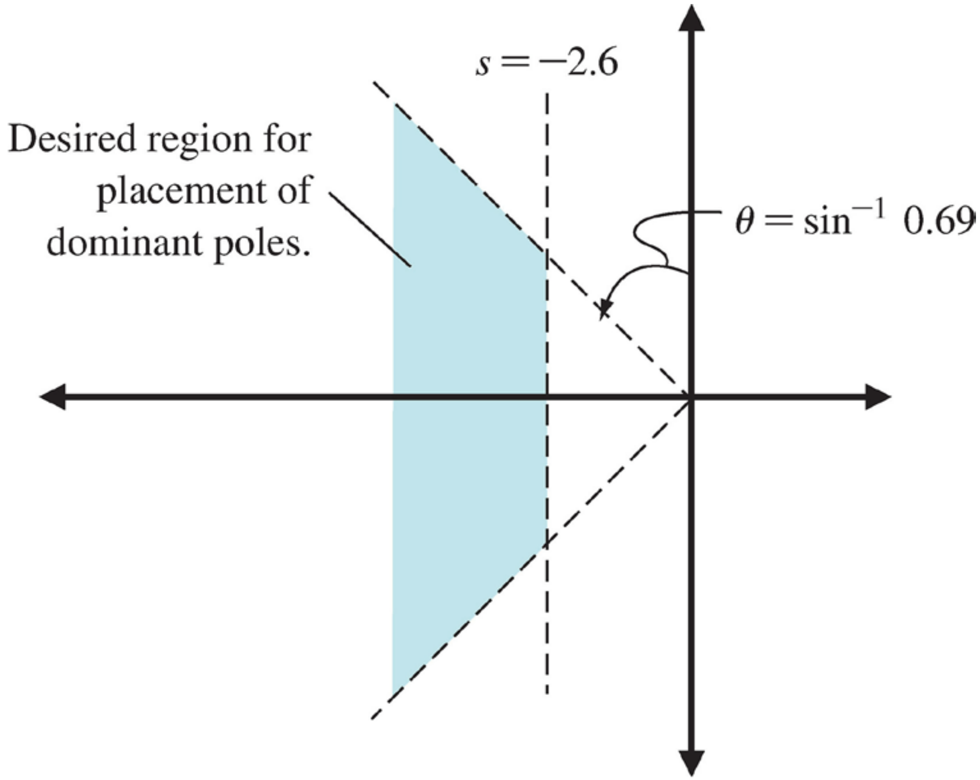
$$\begin{array}{c|cc}
s^3 & 1 & 16 + K_P \\
s^2 & 10 & K_I \\
s & \dfrac{10(K_P + 16) - K_I}{10} & 0 \\
1 & K_I &
\end{array}$$

Therefore, we need

$$K_I > 0 \quad \text{and} \quad K_P > \frac{K_I}{10} - 16$$

Step 3.

Consider DS3 and DS4, we can obtain the desired region in the s-plane for locating the dominant system poles.



Step 4.

Consider the root locus. Here we want to have the dominant poles to the left of the $s = -2.6$ line. We know from our experience sketching the root locus that since we have three poles (at $s = 0, s = -2\ and\ s = -8$) and one zero (at $s = -K_I/K_P$), we expect two branches of the loci to go to infinity along two asymptotes at $\emptyset = \pm 90^o$ centered at

$$\sigma_A = \frac{\sum(-p_i) - \sum(-z_i)}{n - M}$$

where $n = 3$ and $M = 1$. In this case

$$\sigma_A = \frac{-2 - 8 - (-\frac{K_I}{K_P})}{2} = -5 + \frac{1}{2}\frac{K_I}{K_P}$$

We want to have $\sigma_A < -2.6$ so that the two branches will bend into the desired regions.

$$\sigma_A = -5 + \frac{1}{2}\frac{K_I}{K_P} < -2.6 \quad \longrightarrow \quad \frac{K_I}{K_P} < 4.7$$

Step 5.

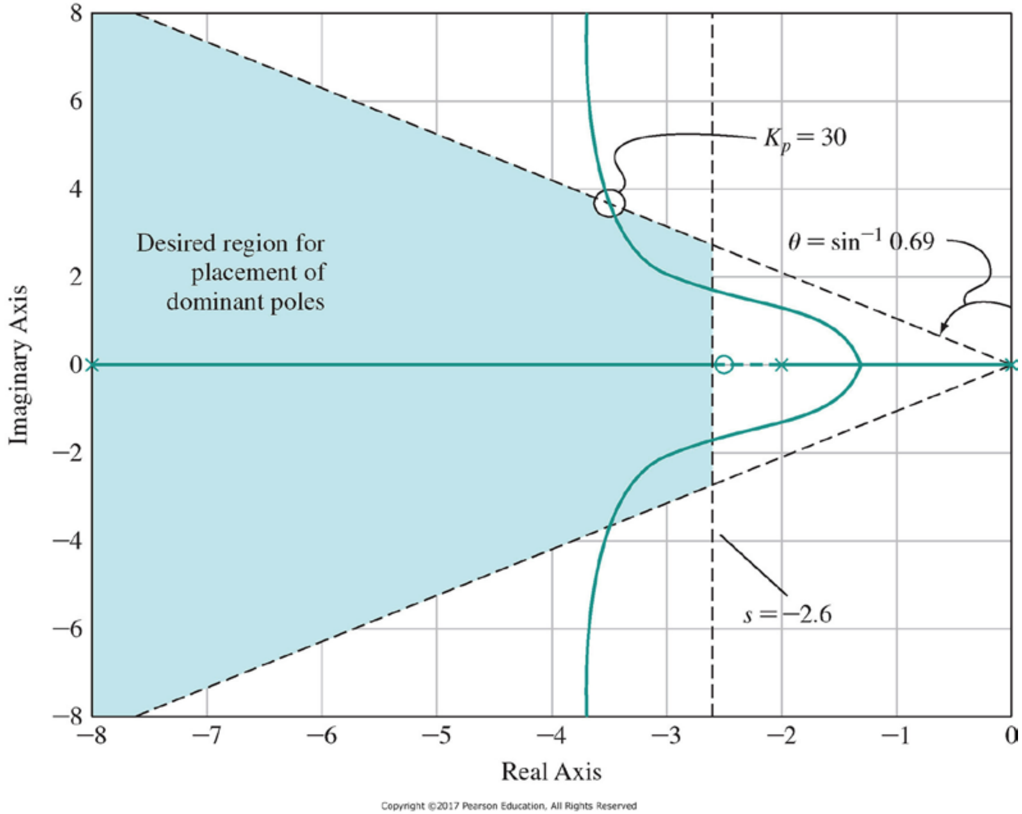
As a first design, we can select K_P and K_I such that

$$K_I \geq 64, \qquad K_P > \frac{K_I}{10} - 16, \qquad \text{and} \qquad \frac{K_I}{K_P} < 4.7$$

Suppose we choose $\frac{K_I}{K_P} = 2.5$. Then the closed-loop characteristic equation is

$$1 + K_P \frac{s + 2.5}{s(s + 2)(s + 8)} = 0.$$

Sketch the root locus for the above equation.



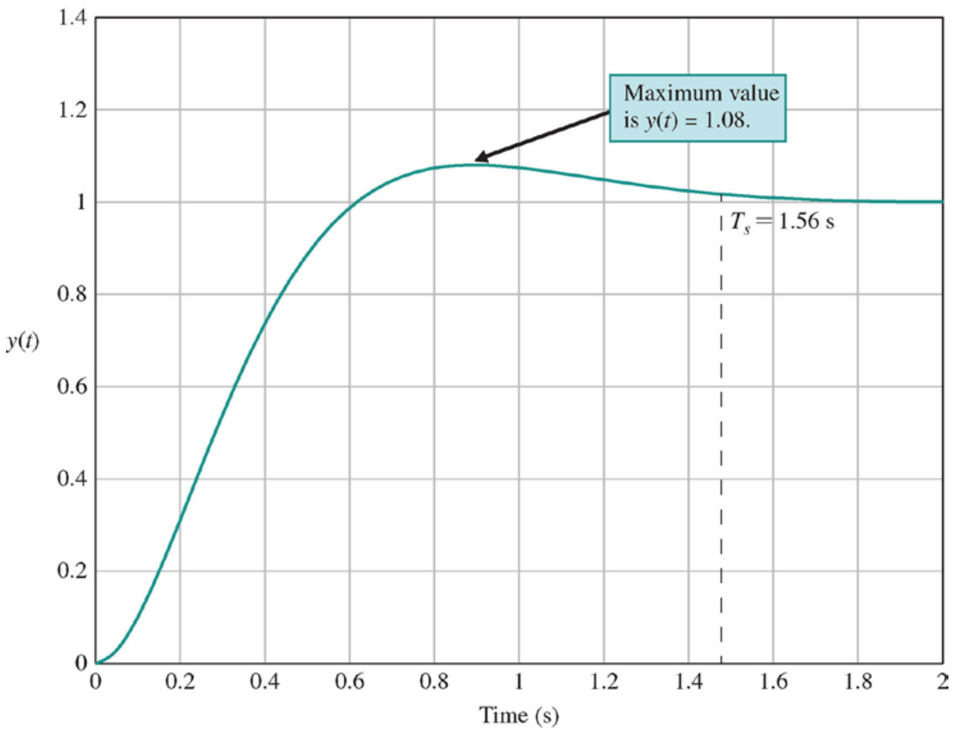
- We need to select $K_P < 30$ to meet the DS3 ($P.O. \leq 5\%$, i.e. $\zeta < 0.69$) requirement.

We can select $K_P = 26$ as a first try. Since we have $\frac{K_I}{K_P} = 2.5$, then $K_I = 65$. This satisfies the DS2 since $K_I > 64$. The resulting PID controller is

$$\boxed{G_c = 26 + \frac{65}{s}}$$

Step 6.

Check the step response.



- $P.O. = 8\%$, $T_s = 1.56s$. DS3 and DS4 are not precisely satisfied; but the controller represents a very good first design.
- We can iterate the refine it.

Thank You !