Active Vibration Control: Design of Controller using Transform Method

Bishakh Bhattacharya

Dept of Mechanical Eng

Smart Materials & Systems Laboratory

Jargons!

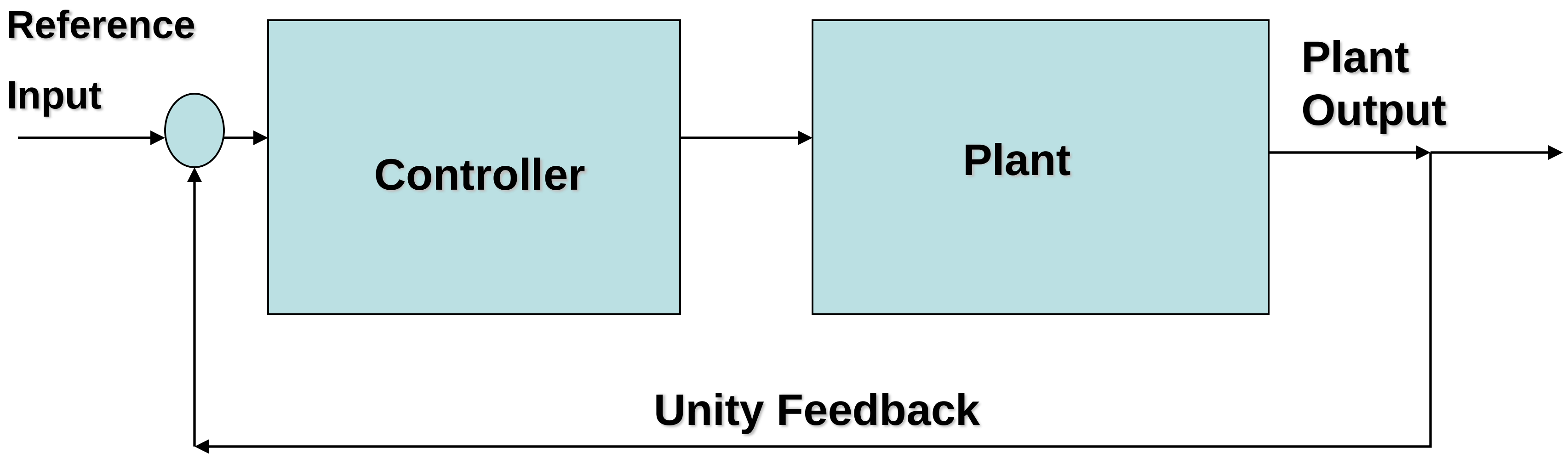
- Cascade – If object A is cascaded over object B $\longrightarrow$
A is falling or hanging over B. We imply: A and B are in the forward path, output of A is input of B

- Compensator – A controller that is used to compensate the unwanted dynamics of a Plant/Process or System

- Lead Controller - The transfer function of this type of controller introduces a dominant zero – thereby creating phase lead in the transfer function – eqv. To PD control
- Lag Controller - Reverse – here it introduces a dominant pole – and hence phase lag in the transfer function – eqv. To PI
- Lead-Lag – eqv. To PID

- Notch The frequency response shows as if a notch has been cut in an otherwise flat response.

Cascaded Controller



Lag Compensator, Lead Compensator, Lead-Lag Compensator, Notch Filter

Transient response of a second order system

A Second order system may be represented as follows:

$$Y(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} R(s)$$

Where, R(s) is the excitation signal and the second order plant is defined by it's natural frequency ω_n and damping ratio ζ. The response of the system in time domain may be written as:

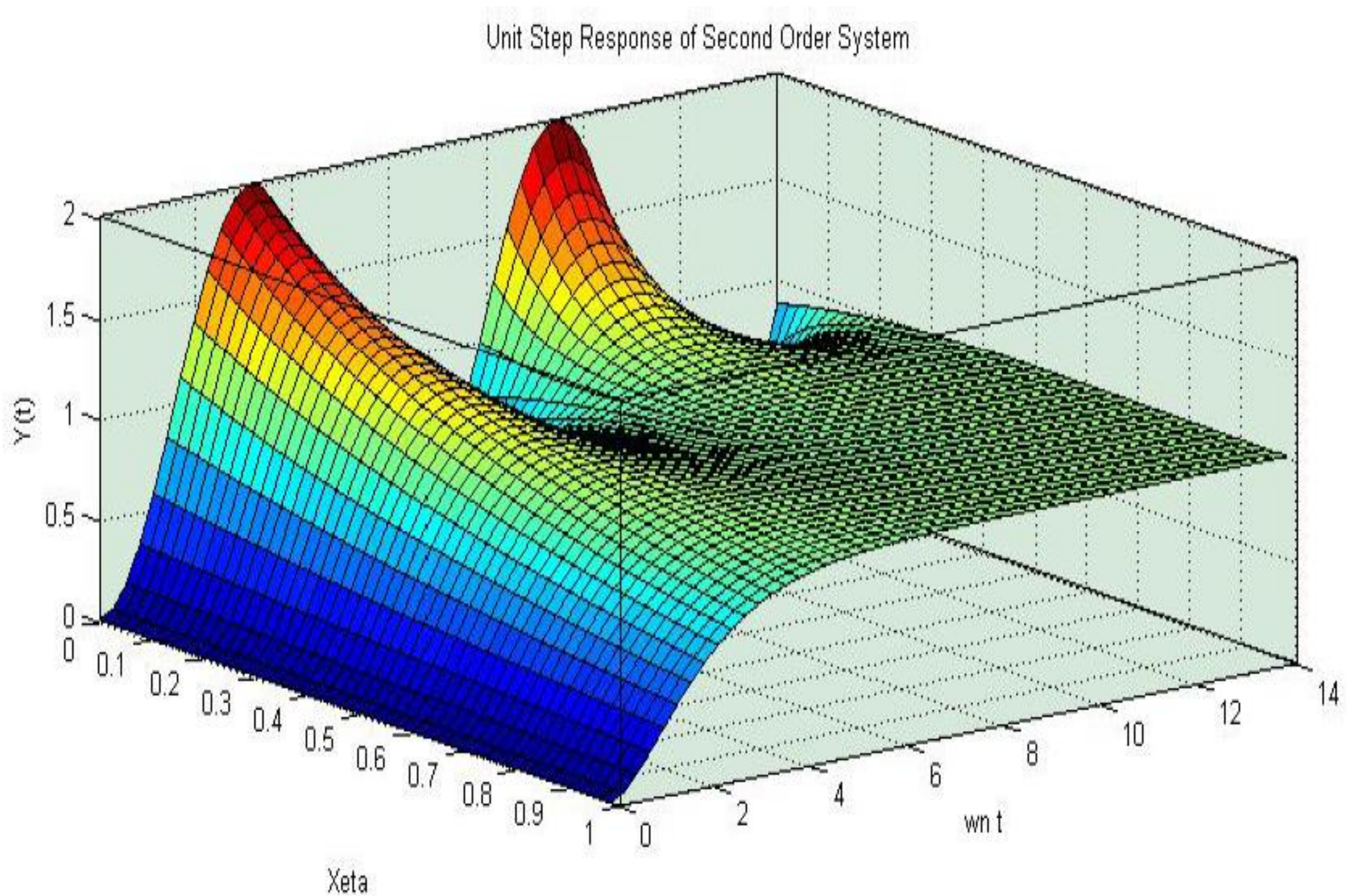
$$y(t) = 1 - \frac{1}{\beta} e^{\zeta\omega_n t} \sin(\omega_n \beta t + \theta)$$

$$\beta = \sqrt{(1 - \zeta^2)}$$

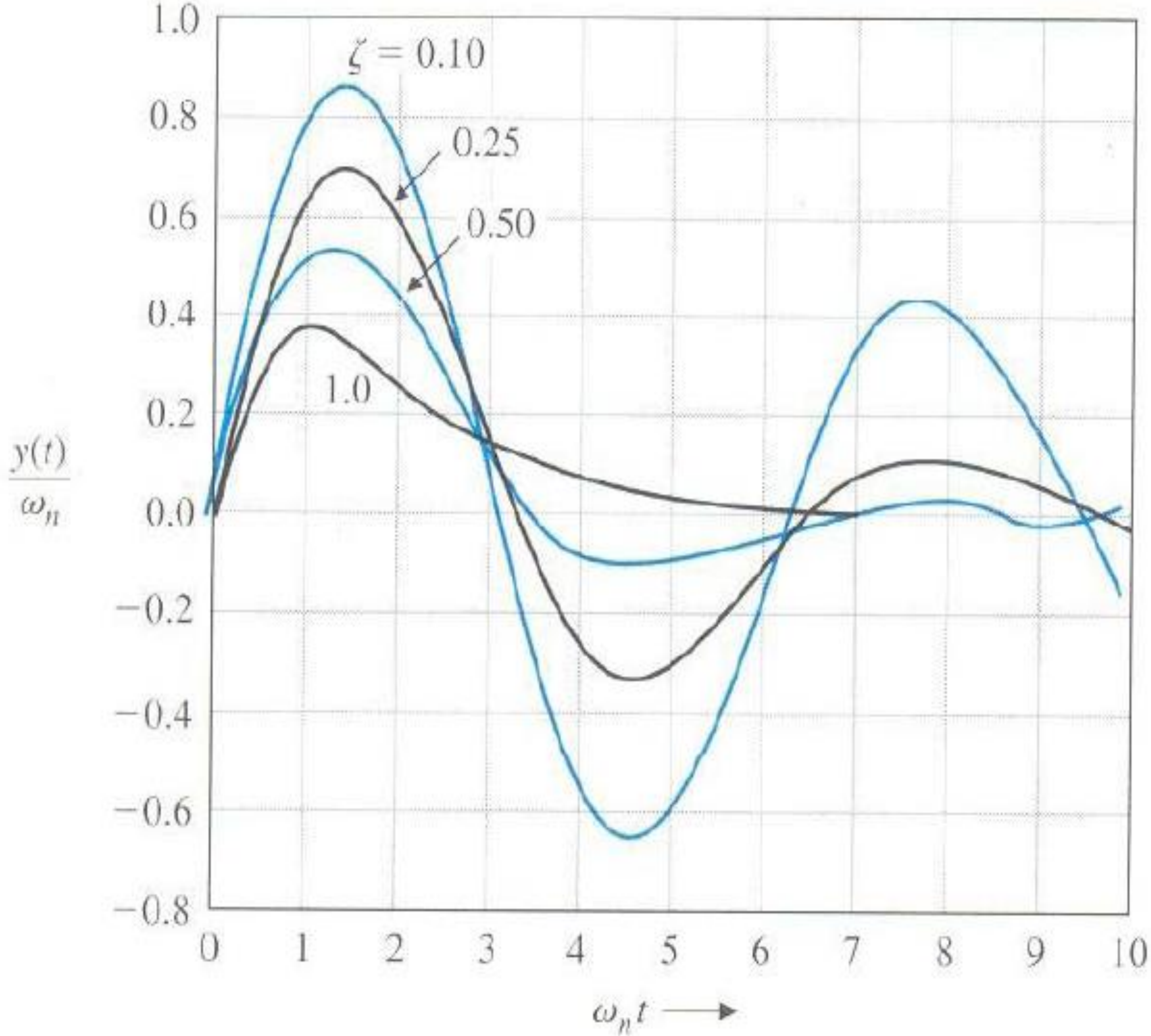
$$\theta = \cos^{-1} \zeta$$

$$\zeta - 0 \text{to} 1$$

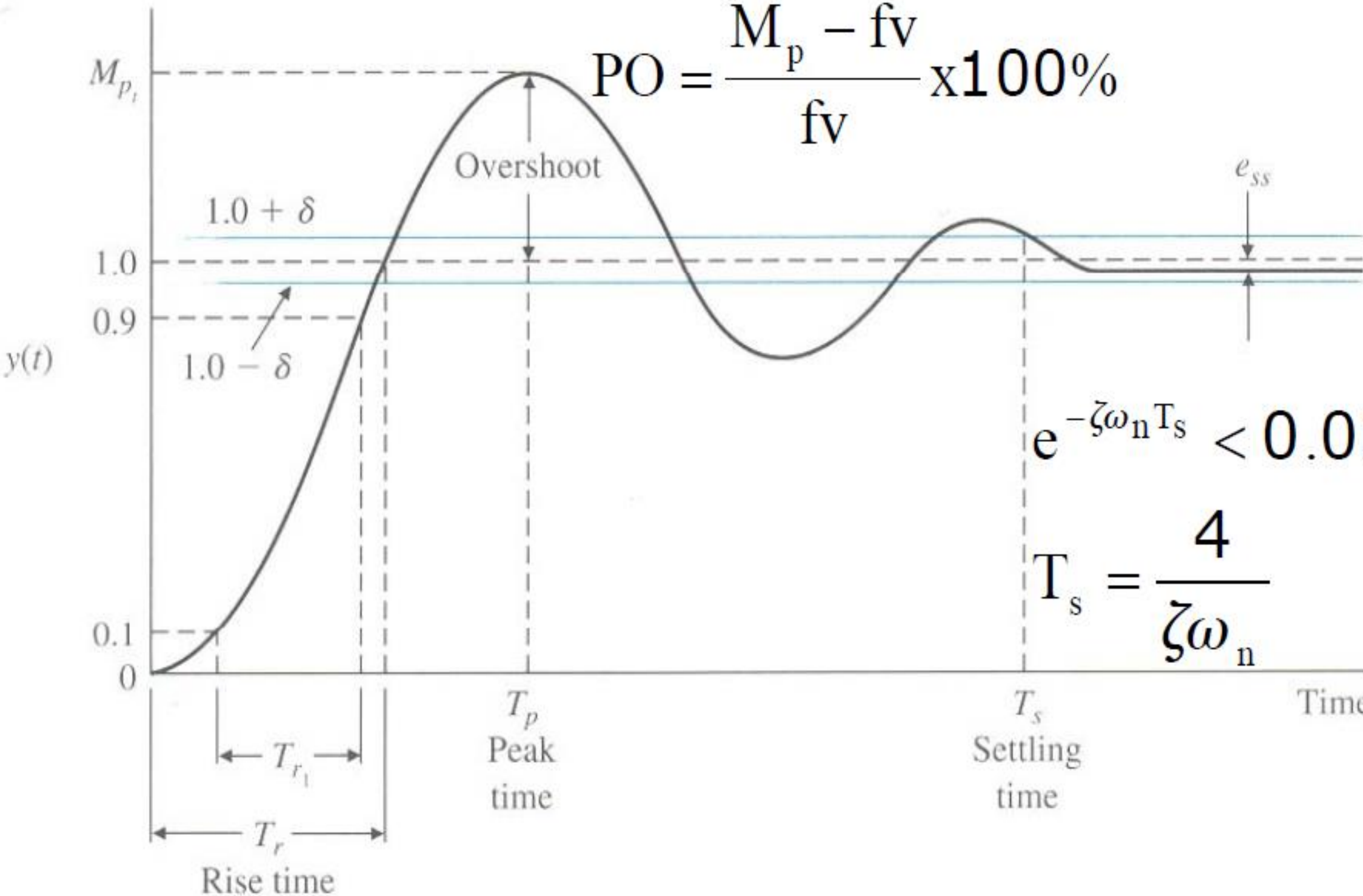
A 3-D representation of unit step response



Unit Impulse Response



Unit step response



$$PO = \frac{M_p - fv}{fv} x100\%$$

$$e^{-\zeta\omega_n T_s} < 0.02$$

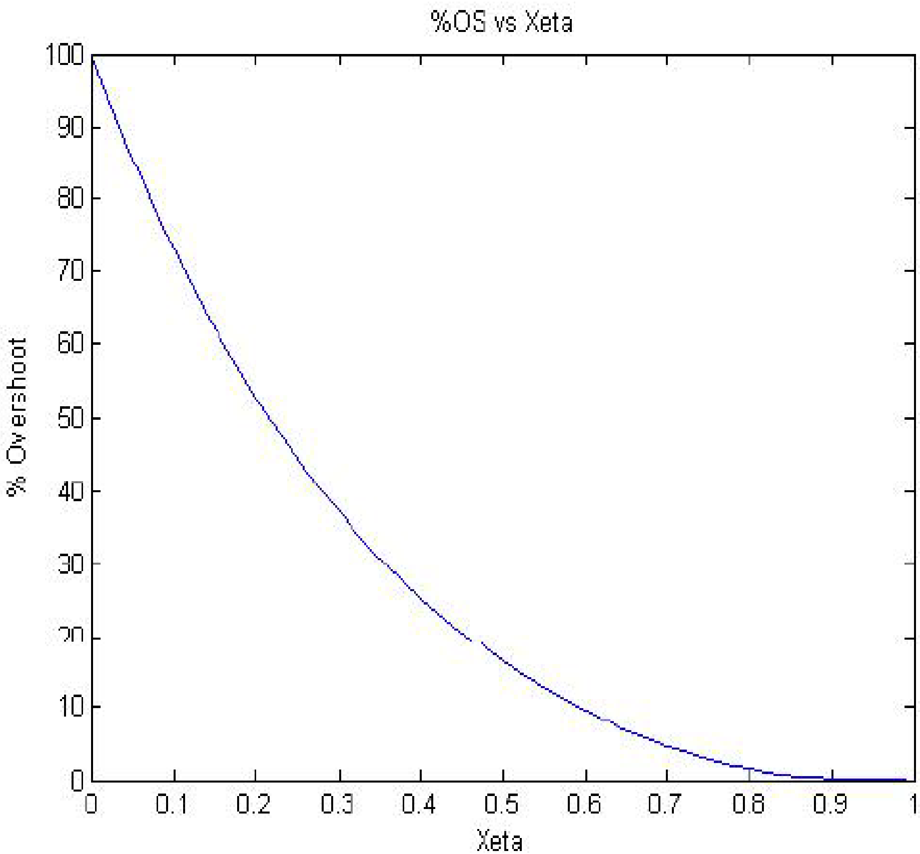
$$T_s = \frac{4}{\zeta\omega_n}$$

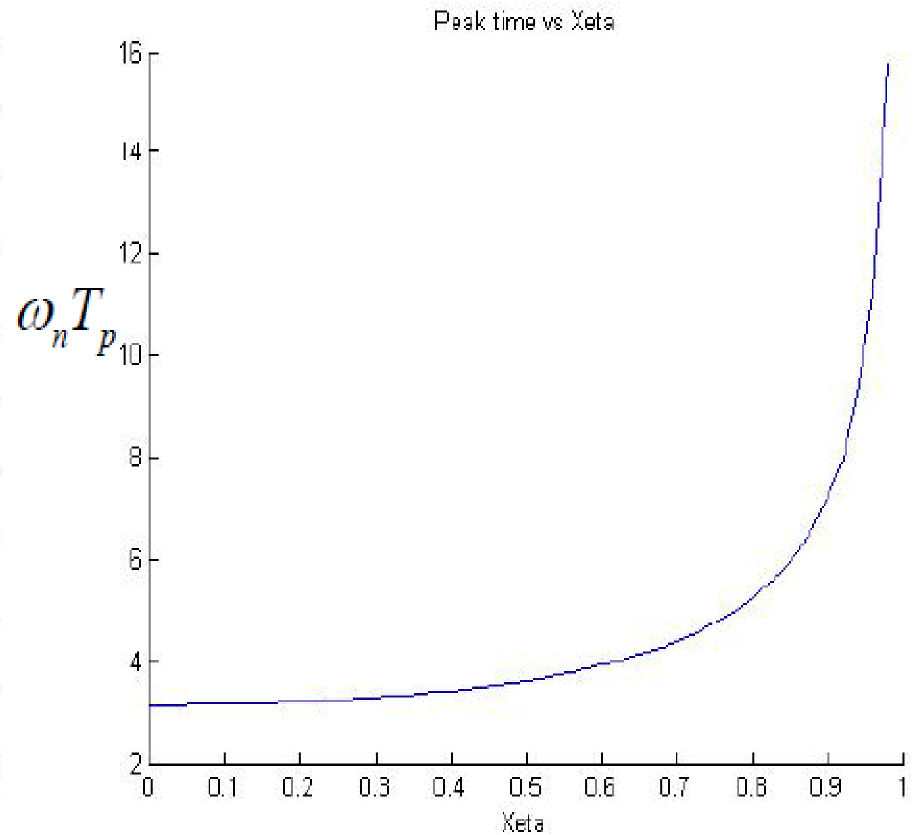
PO: A Measure of Closeness of response

$$PO = 100e^{-\zeta\pi/(\sqrt{(1-\zeta^2)})}$$

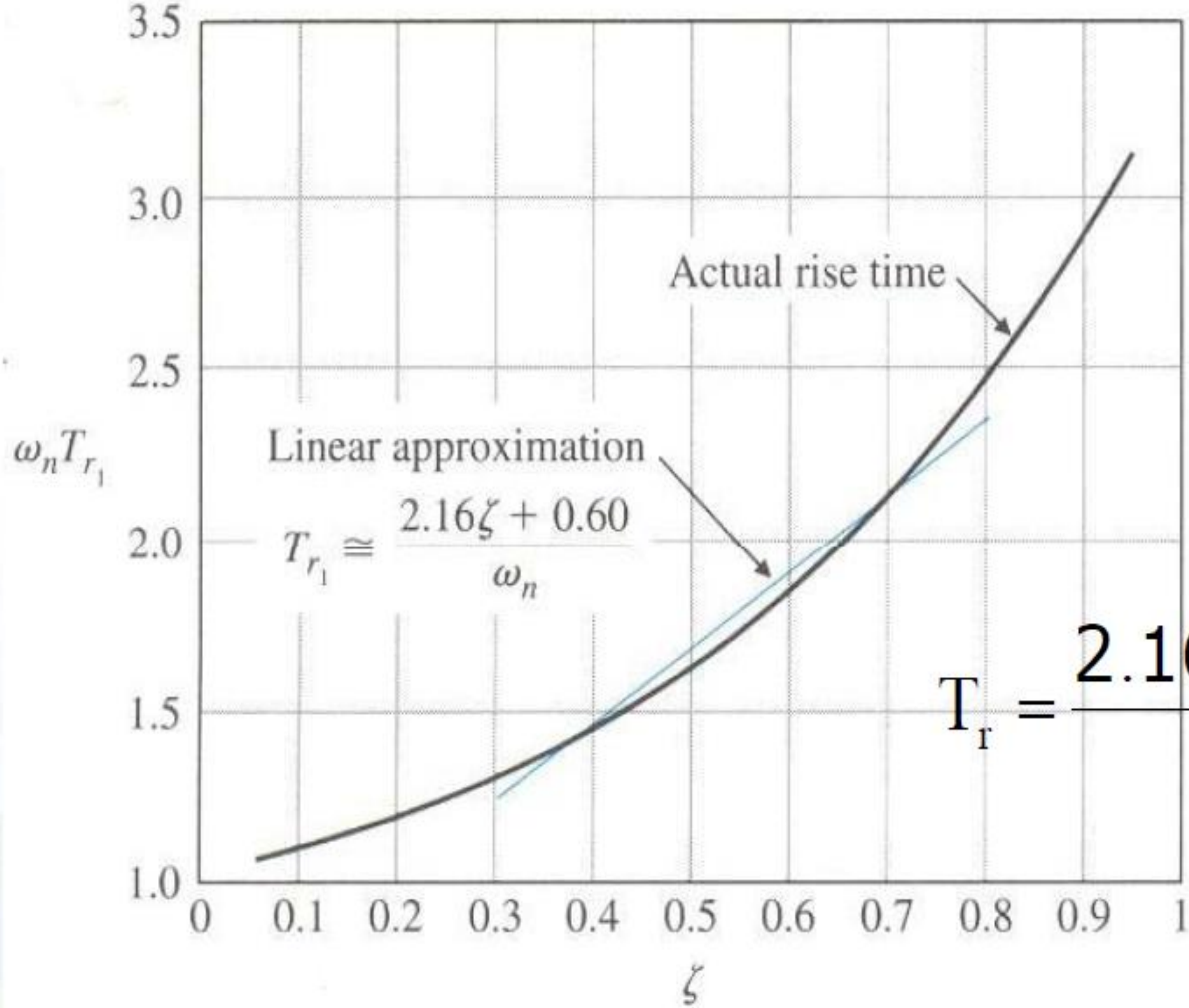
Peak Time: A Measure of swiftness

$$T_p = \frac{\pi}{\omega_n\sqrt{(1-\zeta^2)}}$$





Normalized Rise-time



$$T_r = \frac{2.16\zeta + 0.60}{\omega_n}$$

Steady state error - Introduction

- Steady state error refers to the long-term behavior of a dynamic system.
- The Type of a system is significant to predict the nature of this error.
- A system having no pole at the origin is referred as Type-0 system.
- Thus, Type-1, refers to one pole at the origin and so on.
- It will be shown in this lecture that, it is the type of a system which can directly determine whether a particular command will be followed by a system or not.
- We will consider three common commands: namely, step, ramp and parabolic ramp and find out the steady state response/error of a system to follow these commands.
- A closed loop control system shows remarkable performance in reducing the steady state error of a system.

Steady state error of a system

- Error in a system: E(s) = U(s) /(1+G(s))

$$\lim_{t\to\infty} e(t) = e_{ss} = \lim_{s\to 0} \frac{sU(s)}{1+G(s)}$$

- For a step input

$$e_{ss} = \frac{A}{1+G(0)}$$

- Plant Transfer function G(s) is defined as

$$G(s) = \frac{K\prod_{i=1}^{M}(s+z_i)}{s^k \prod_{j=1}^{N}(s+p_j)}$$

Error Constants

- **Position error constant** $$K_p = \lim_{s \to 0} G(s)$$

- **Steady state error of a step input of magnitude A is** $e_{ss} = A/(1+K_p)$

- **Steady state error will be zero for system with type greater than or equal to 1**

- **For ramp input** $$e_{ss} = \lim_{s \to 0} \frac{A}{sG(s)}$$

- **Define velocity constant as** $$K_v = \lim_{s \to 0} sG(s)$$

- **Hence steady state error is** A/K_v

- **Error will be zero for k greater than or equal to 2**

Summary of Steady State Errors

Type	Step (A/s)	Ramp (A/s^2)	Parabolic Ramp (A/s^3)
0	$E_{ss}= A/(1+K_p)$	Inf	Inf
1	0	A/K_v	Inf
2	0	0	A/K_a

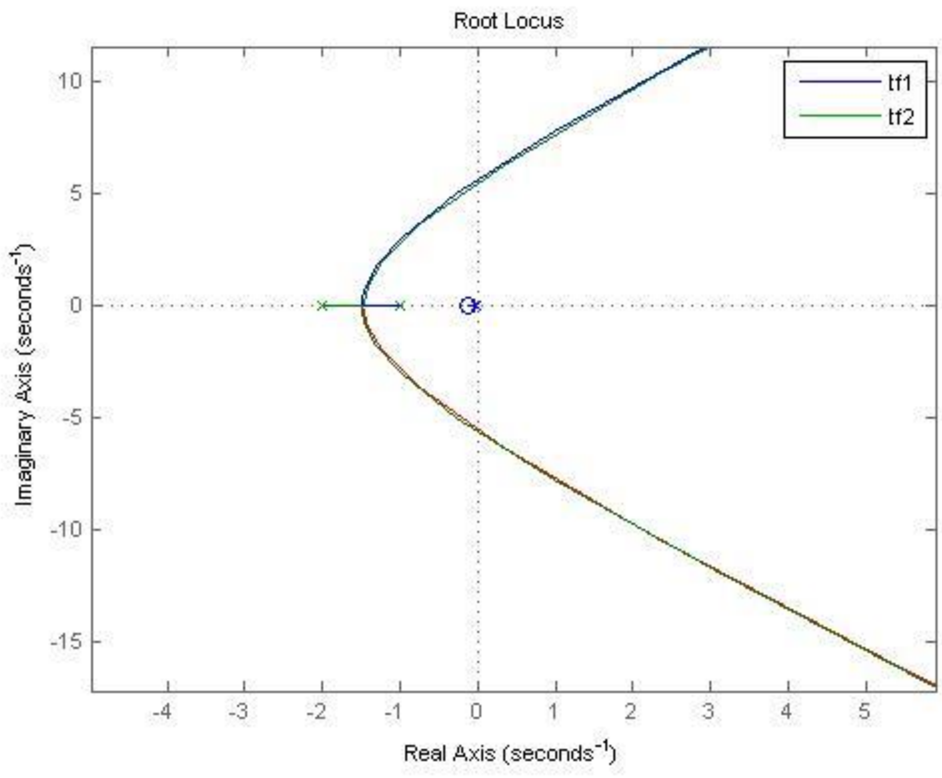
Lag Filter/Compensation

- Introduces a set of Pole and Zero
- Improves Steady State Performance if the Pole is very near to the origin and the zero is little away in the left side from the pole.
- K_0 (uncompensated) = Kz1z2…/$p_1 p_2$….
- K (compensated) = $K_o\ z_c/p_c$

Design of a Lag Compensator

- Consider a Plant with open loop poles at -1,-2 and -10, Find K_p, Improve steady state error by a factor of 10 and get the closed loop poles. Damping ratio should be around 0.174.
- K_p = 8.23 (Dominant Pair Gain/$p_1p_2p_3$)
- e_α= 1/1+K_p=.108
- Desired e_α =(1/10) e_α =.0108, K_p= 91.59
- z_c/p_c = 91.59/8.23 = 11.13, Use p_c=0.01, z_c = 0.111

Root Loci for the two systems

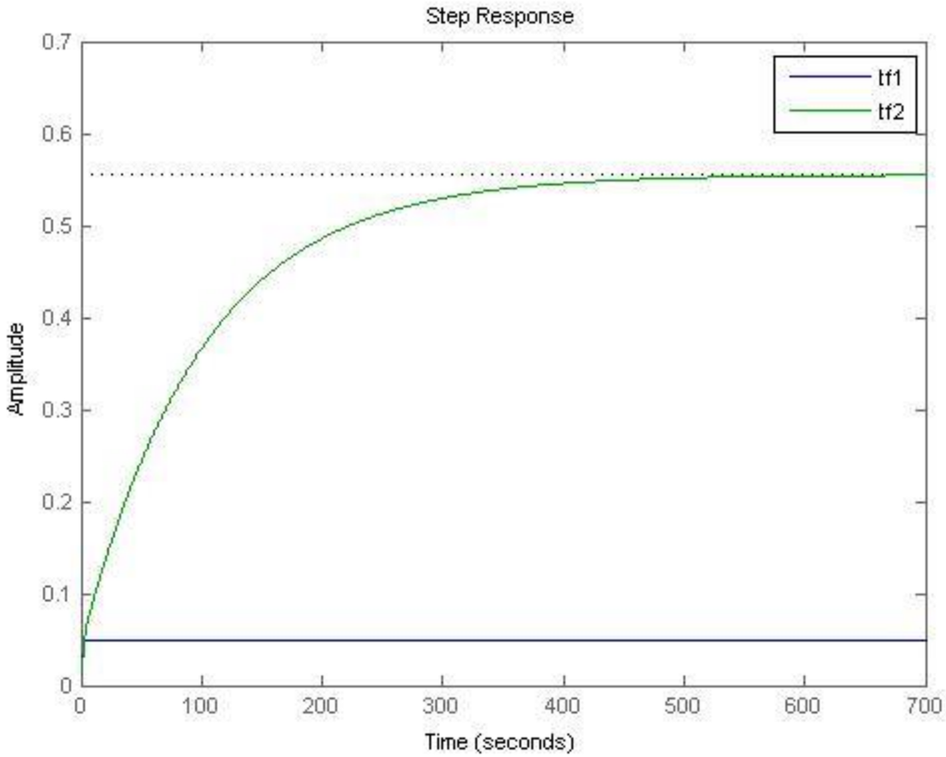


$$tf_1 = \frac{1}{(s+1)(s+2)(s+10)}$$

$$tf_2 = \frac{(s+0.11)}{(s+0.01)(s+1)(s+2)(s+10)}$$

There's almost no effect of compensator on the transient behavior and stability of the system

Steady State Response of the two systems



$$tf_1 = \frac{1}{(s+1)(s+2)(s+10)}$$

$$tf_2 = \frac{(s+0.11)}{(s+0.01)(s+1)(s+2)(s+10)}$$

You may have observed that the steady state response has increased more than five times.

Application of Lag Controller

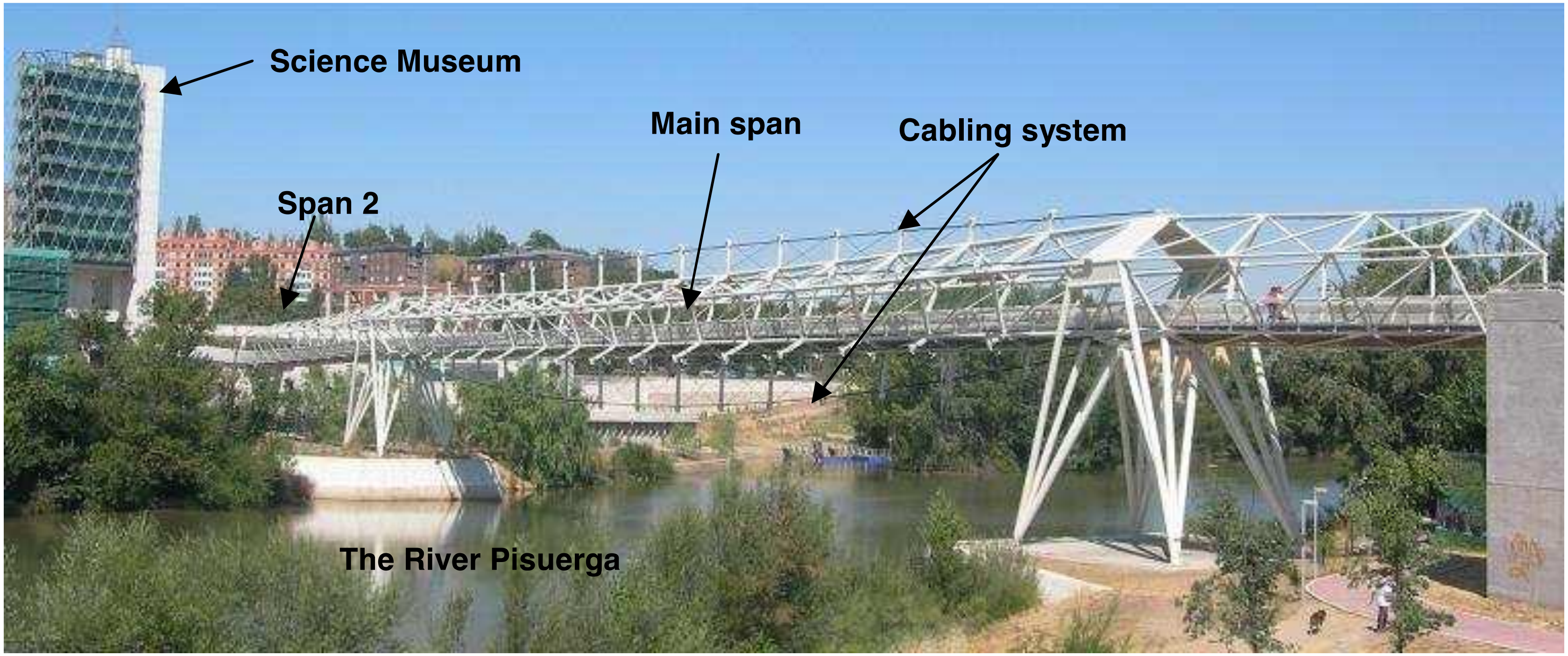


Figure 1. General view of the structure

Vibration Control of the Central Span

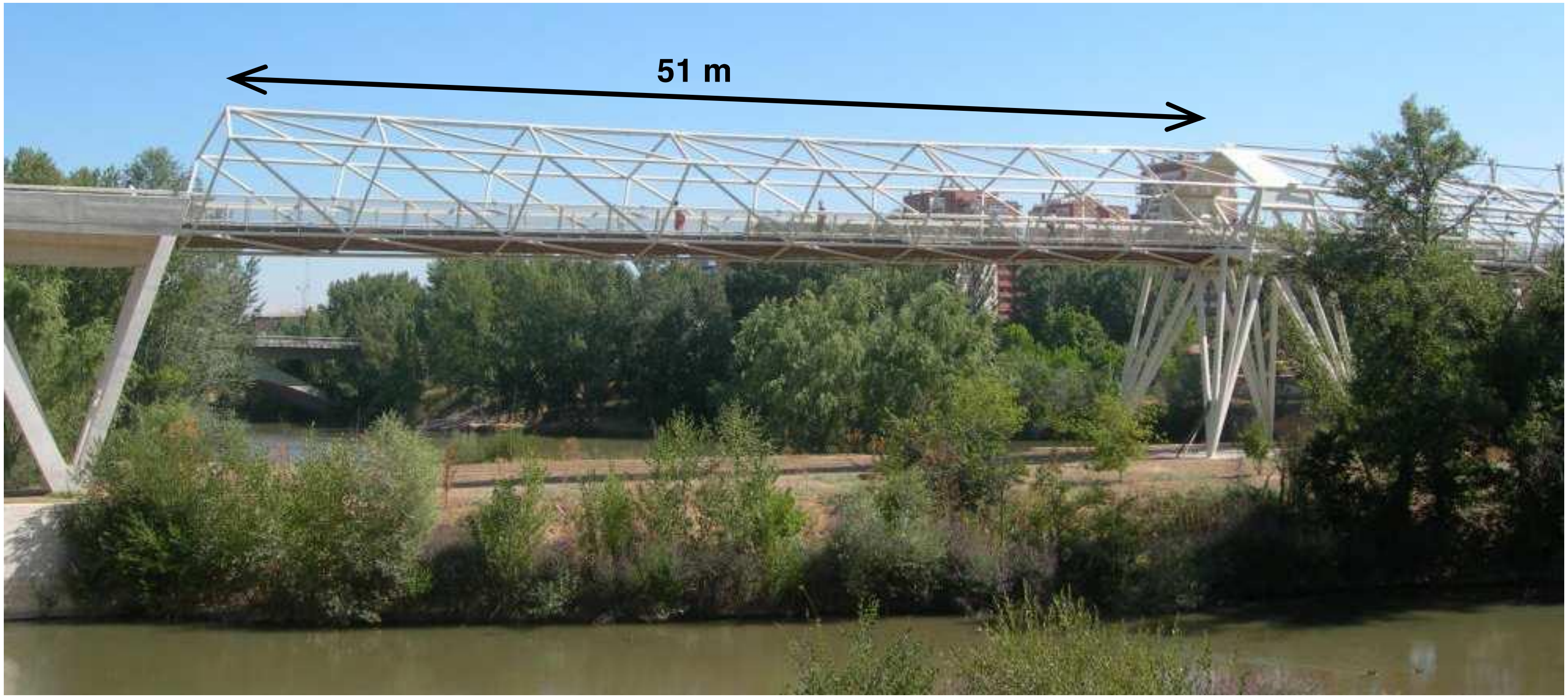
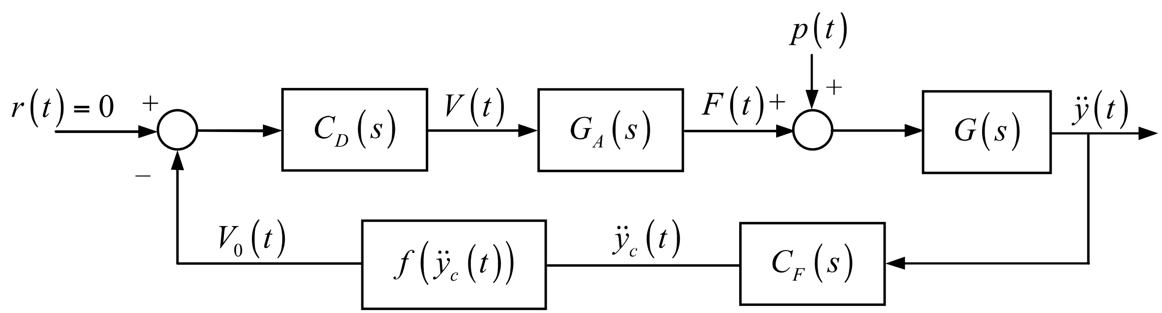


Figure 2. View of Span 2

Three vibration modes were identified in the frequency range 1–15 Hz

$$G(s)=\sum_{i=1}^{3}\frac{\alpha_i s^2}{s^2+2\zeta_i\omega_i s+\omega_i^2}=\frac{7.13\cdot 10^{-5}s^2}{s^2+0.264s+483.6}+\frac{4.54\cdot 10^{-6}s^2}{s^2+0.279s+2162}+\frac{5.85\cdot 10^{-5}s^2}{s^2+0.591s+3488},$$

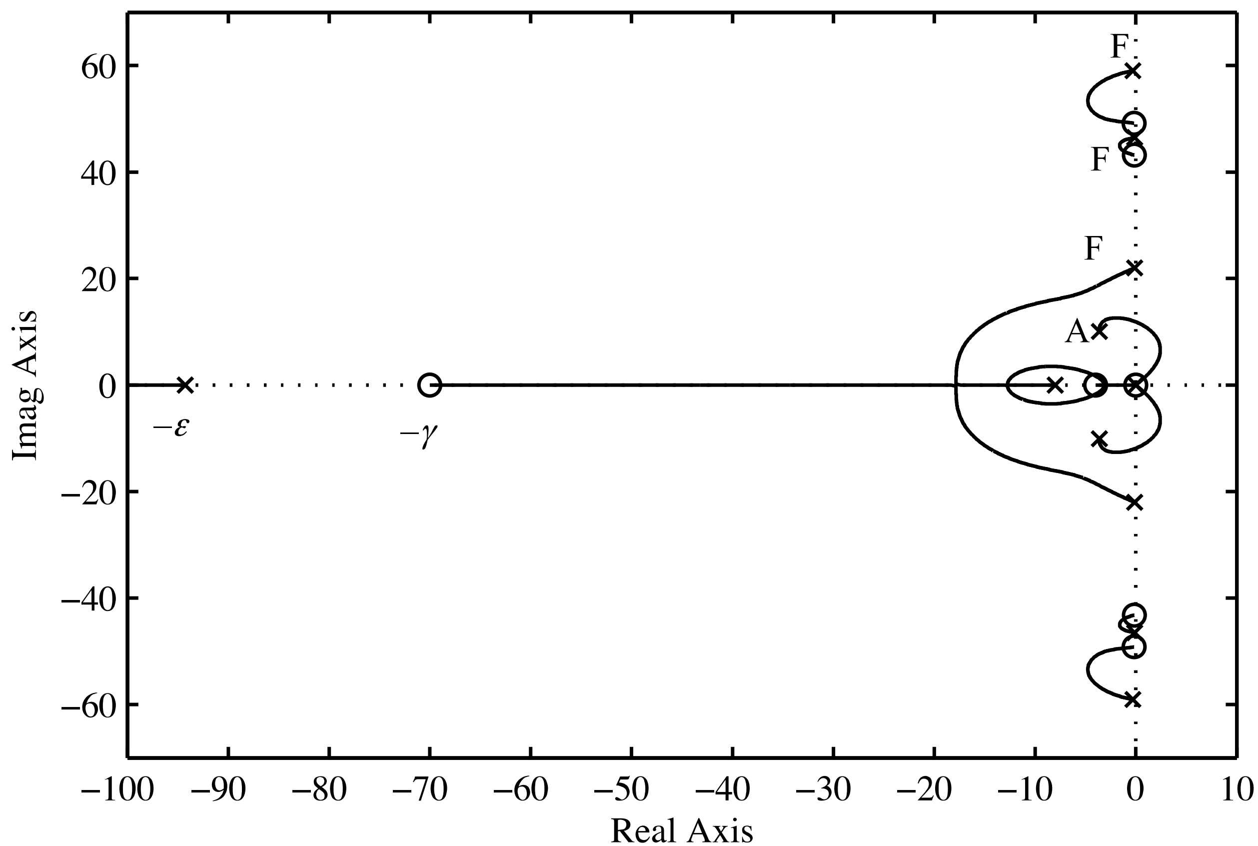
Lag Compensator in the feedback loop



$r(t)$	Reference command	$\ddot{y}(t)$:	Acceleration response
$V(t)$:	Control voltage	$\ddot{y}_c(t)$:	Compensated acceleration
$F(t)$:	Actuator force	$V_0(t)$:	Initial control voltage
$p(t)$:	Plant disturbance	$f(\ddot{y}_c)$:	Nonlinear element
$C_D(s)$:	Transfer function of the direct compensator		
$G_A(s)$:	Transfer function of the proof-mass actuator		
$G(s)$:	Transfer function of the floor structure		
$C_F(s)$:	Transfer function of the feedback compensator		

Figure 4. General control scheme

Root Locus Diagram



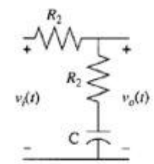
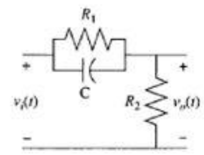
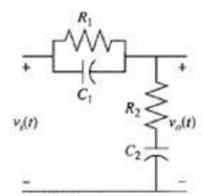
Root locus of the total transfer function G_T. (×) pole; (o) zero; (F) footbridge; (A) actuator

Practical Realization of a Lag Compensator

The figure below shows the practical realization of a Lag Compensator with the help of resistors and capacitor. In terms of mechanical elements we can realize the same by using dashpot and springs. Just replace t the resistors by dashpots and capacitor by spring.

Passive-Circuit Realization

- Lag, lead, and lag-lead compensators can also be implemented with passive networks (Table 2) .

Function	Network	Transfer function, $\frac{V_o(s)}{V_i(s)}$
Lag compensation		$\frac{R_2}{R_1+R_2}\frac{s+\frac{1}{R_2C}}{s+\frac{1}{(R_1+R_2)C}}$
Lead compensation		$\frac{s+\frac{1}{R_1C}}{s+\frac{1}{R_1C}+\frac{1}{R_2C}}$
Lag-lead compensation		$\frac{\left(s+\frac{1}{R_1C_1}\right)\left(s+\frac{1}{R_2C_2}\right)}{s^2+\left(\frac{1}{R_1C_1}+\frac{1}{R_2C_2}+\frac{1}{R_2C_1}\right)s+\frac{1}{R_1R_2C_1C_2}}$

What is a Lead Compensator?

- Lead Compensator is similar in structure as a Lag Compensator, the transfer function could be written as:

$$K_{lead}(s) = \frac{K_c(s+a)}{s+b}$$

- However, in this case the zero is dominating over the pole and hence *b>a*.

- Lead Compensator improves the stability of a system by shifting the root locus towards the left of the origin. Thus while a lag compensator improves the steady state response sacrificing the stability, lead improves the stability of the system.

Design of a Lead filter

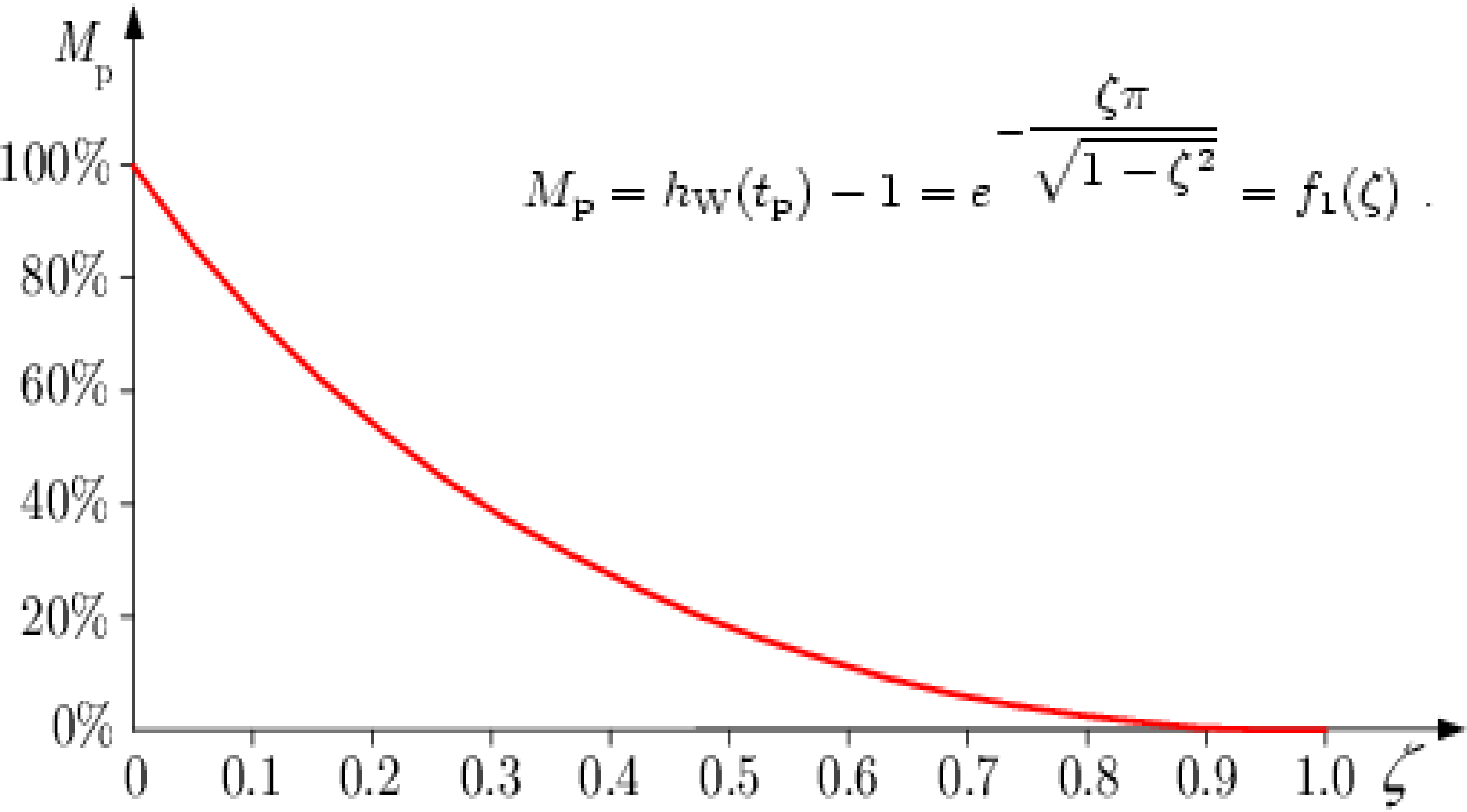
- Consider a Plant with open loop poles at 0,-4 and

-6, even if this is a 3rd order system, the third pole is quite away.

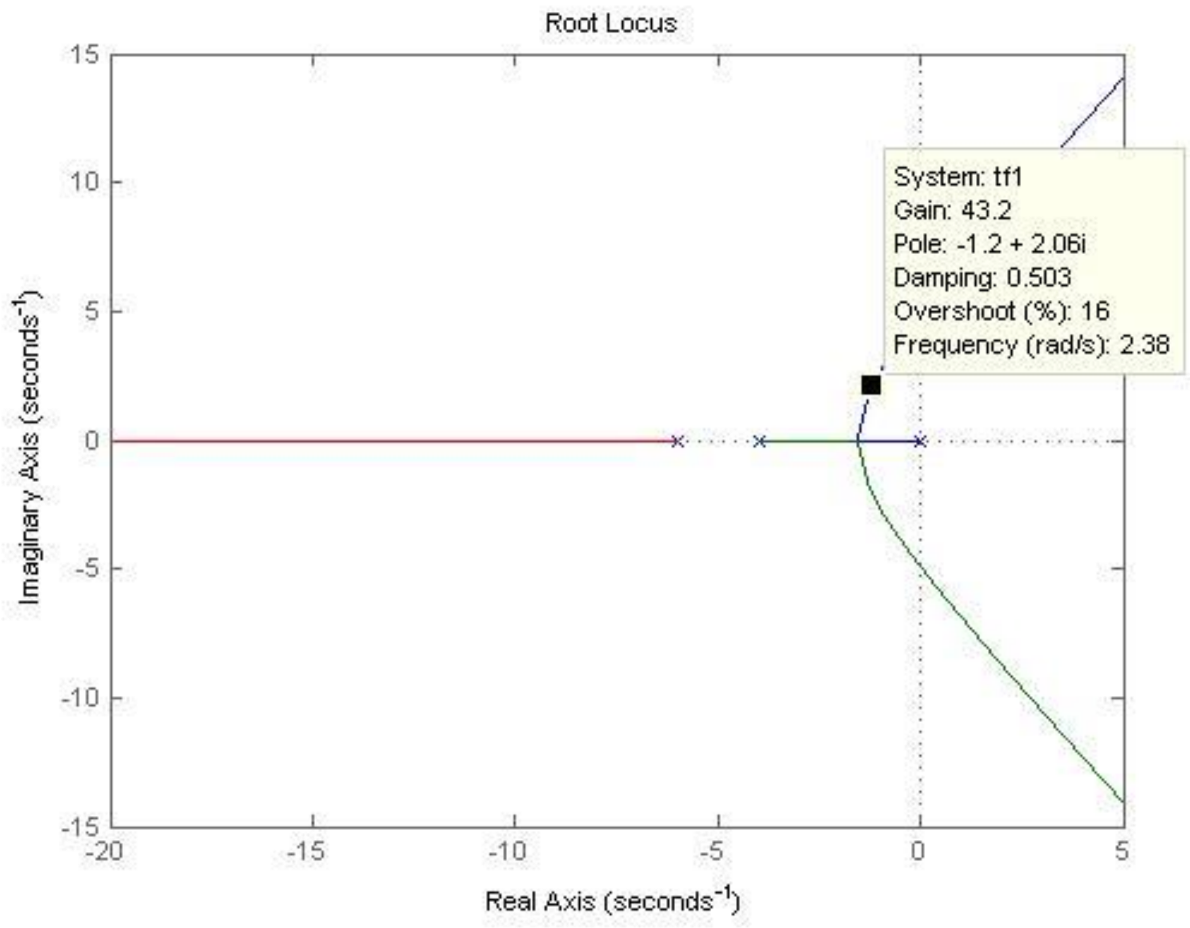
Objective: Get 16% overshoot with three-fold reduction in settling time.

- ζ=.504 (for 16% OS) (see the chart in the next page)
- Poles at -1.20 +- j2.06, K=43.20
- The uncompensated system is shown in the next slide.

Damping Factor Vs. Overshoot



Design of a Lead filter



$$tf_1 = \frac{1}{s(s+4)(s+6)}$$

Design of a Lead filter

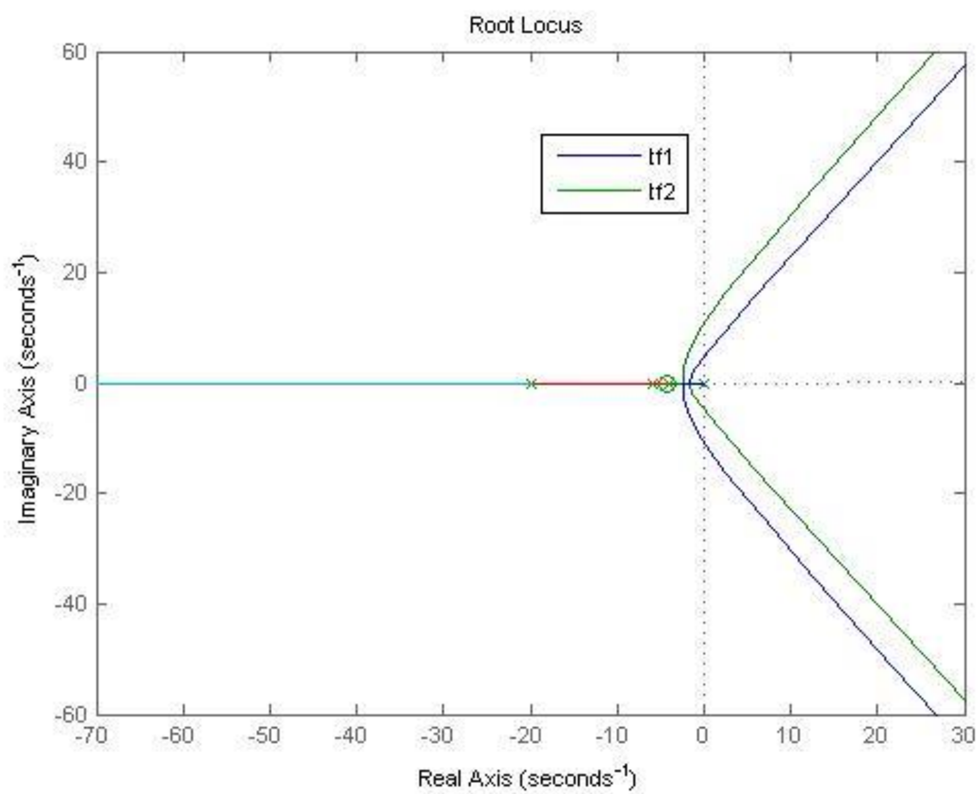
- As per design specification: at the current location

$T_s = 4/\zeta\omega = 4/1.20 = 3.33$

- Recheck it as a 2nd order system
- For 3-fold reduction, desired T_s = 1.107 s, corresponding $\sigma = \zeta\omega = 4/T_s = 3.613$
- With same damping, ω_d = 3.613 tan(180-120.26) = 6.193
- Hence, the desired poles are at -3.613 +- j6.193
- Plot and get the phase at the point as -275.6^o
- Hence, angle of the compensator zero is 95.6^o, Use graphical plot to get the real part of zero.
- The compensator zero could be obtained as -4.22
- The compensator pole may be chosen at -20.

Root Loci of the Compensated and uncompensated system

- The plot clearly shows that the introduction of the lead compensator has shifted the plot towards left and increased the stability of the system.



PD Vs Lead Compensator

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

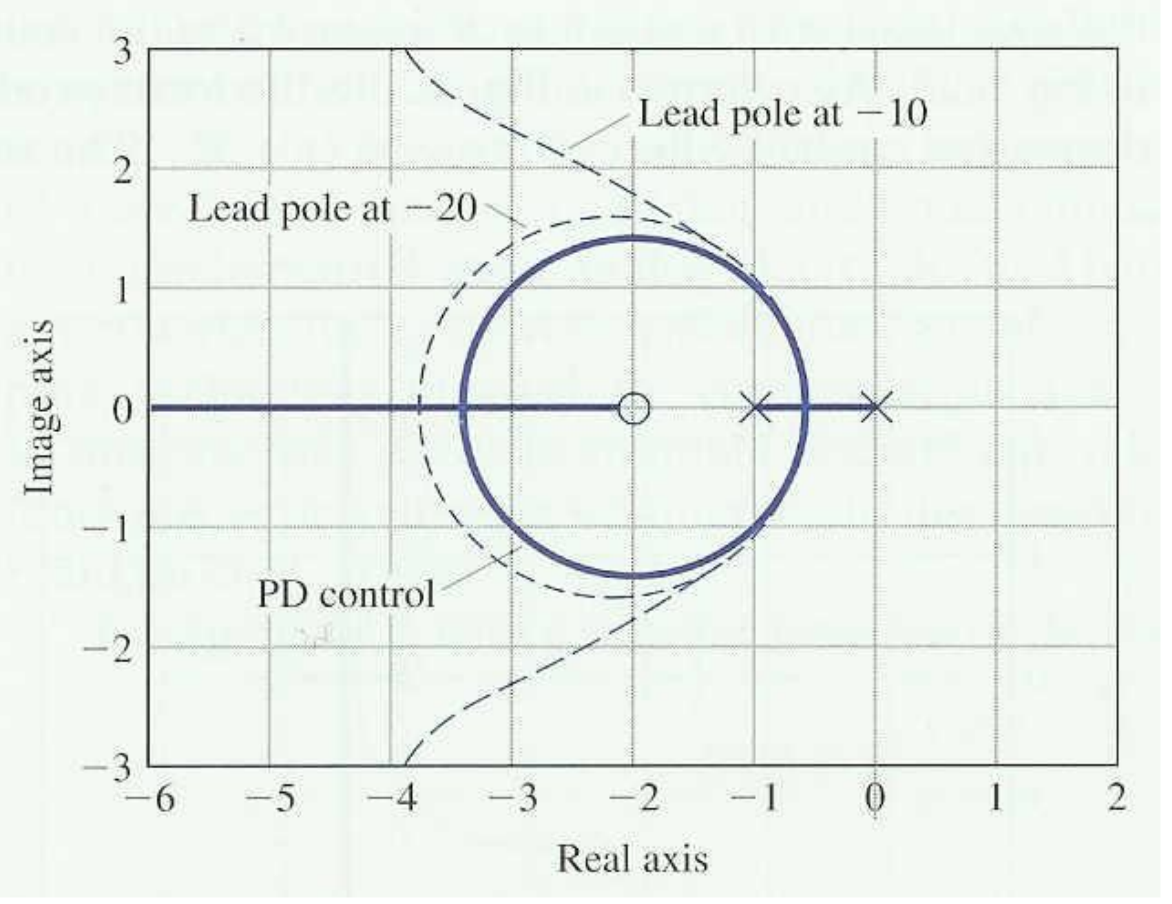
PD - control

$$D(s) = K(s+2)$$

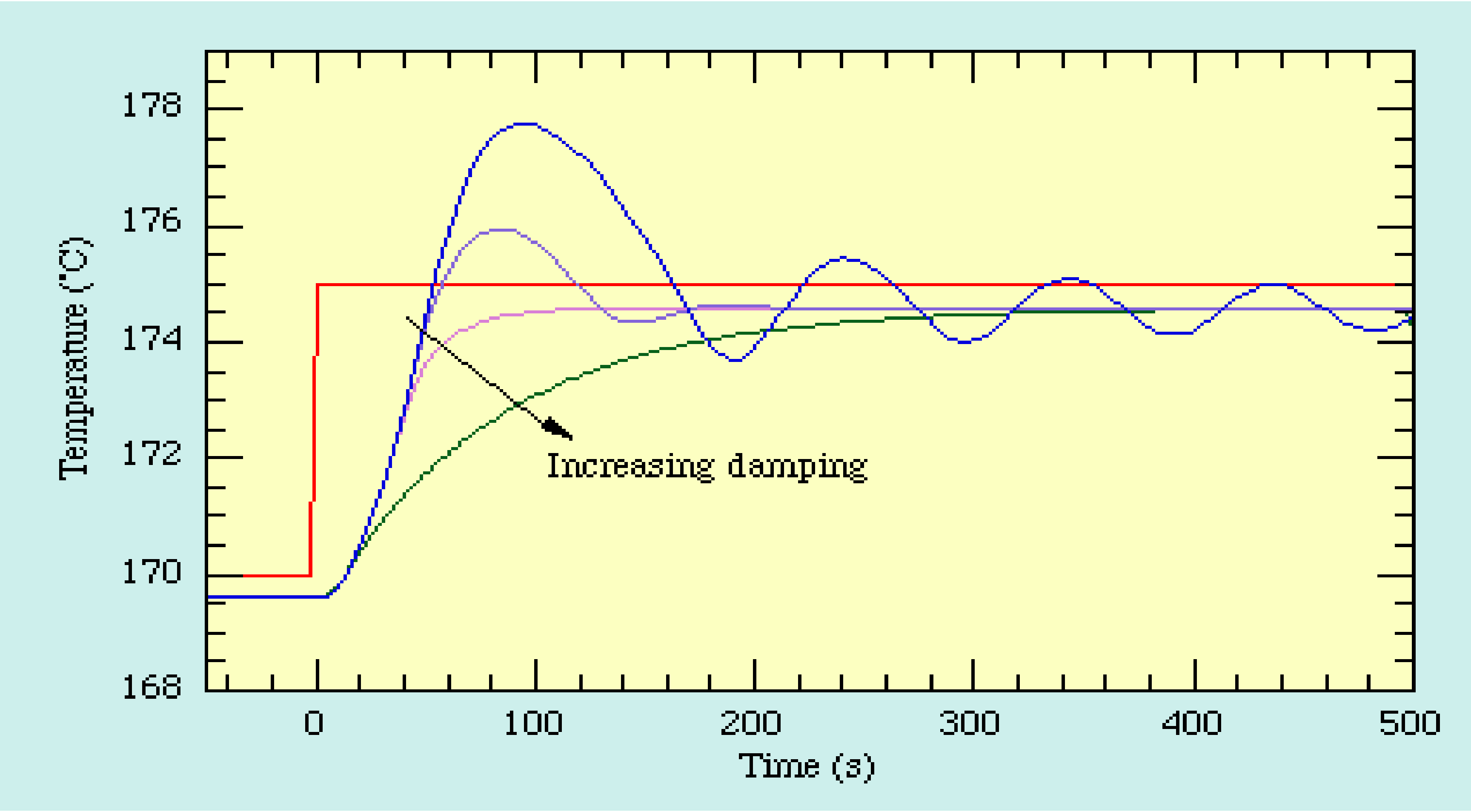
Lead compensati on

$$D(s) = K\frac{s+2}{s+p},$$

$$p = \begin{cases} 5z = 10 \\ 10z = 20 \end{cases}$$



PD Control



178
176
174
172
170
168
Temperature (°C)
0
100
200
300
400
500
Time (s)
Increasing damping

Design of a Lead – Lag Compensator

- Evaluate the performance of uncompensated system and get the desired pole location.
- Design the Lead Compensator to meet the transient response
- Simulate and Redesign to meet the performance
- Evaluate steady state performance and obtain the lag compensator
- Simulate and Iterate to check all the performances

What is PID Control?

- **Dynamic Systems are often controlled with the help of a three term compensator known as PID; P – stands for Proportional Control, I stands for Integral Control and D stands for Derivative Control.**
- **Consider a closed loop system with unity feedback as follows:**

- **The plant *P*, is controlled by a control input *u(t)*, which can be expressed as follows:**

$$u(t) = K_p e(t) + K_i \int e(t)\,dt + K_d \dot{e}(t)$$

Frequency Domain Representation of Controller

- **The control input could be represented in frequency domain as follows:**

$$U(s) = [K_p + \frac{K_i}{s} + K_d s]E(s)$$

- **The closed loop transfer function could be written as follows:**

$$\frac{Y(s)}{R(s)} = \frac{C(s)P(s)}{1 + C(s)P(s)}$$

- **This tells us that the poles of the closed loop transfer functions are actually the zeros of *1+C(s)P(s)*. By considering the three terms as three parameters, you can study the effect of changing each one of these parameters on the root locus.**

Advantage of Different Parameters of PID Controller

Parameters	Advantage	Limitation
K_p	Adjustment of Controller output	May cause instability
K_i	Produces zero steady state error	Slow dynamic Response and Instability
K_d	Provides rapid system response	Sensitive to Noise and non-zero offset

Application of PI Controller on a First Order System

- Consider a first order plant as follows:

$$P(s) = \frac{K}{1+\tau s}$$

- If we apply a PI controller then *C(s)* becomes:

$$C(s) = K_p + \frac{K_i}{s} = K_p \frac{s + \dfrac{K_i}{K_p}}{s}$$

- The closed loop transfer function may be written as:

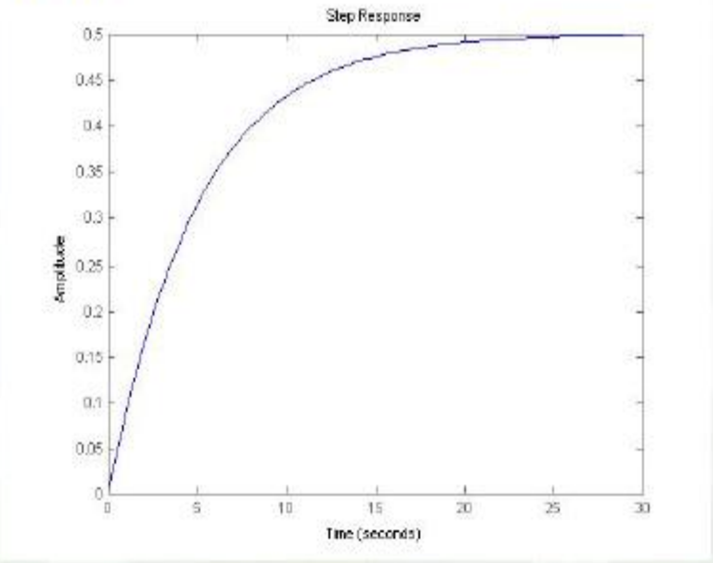
$$\frac{Y(s)}{R(s)} = (KK_p)\frac{s + K_i / K_p}{\tau s^2 + (1 + K K_p)s + KK_i}$$

Numerical Simulation of the system

- Let us consider the first order system with K=1 and τ = 10 s, hence the open loop system transfer function may be written as

$$T(s) = \frac{1}{1+10s}$$

- Let us look at the unit step response of this system – the controller has miserably failed to follow!



First Order System - Numerical Simulation

- **Now let us consider a PI controller with the following parameters:**

$$K = 1$$

$$K_p = 1$$

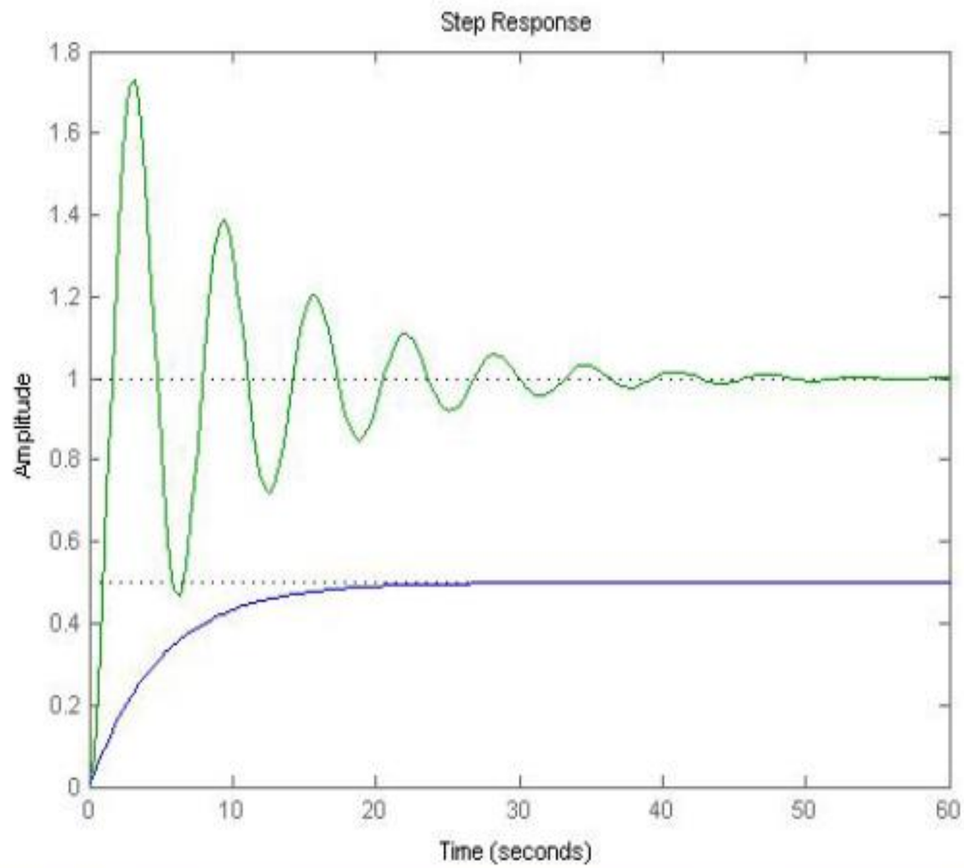
$$K_i = 10$$

- **The new closed loop transfer function may be written as:**

$$T = \frac{s + 10}{10s^2 + 2s + 10}$$

Response of the new system

- The unit step response of the new system is shown below vis a vis the old system:



Application of PD Controller on a First Order System

- Consider a first order plant as follows:

$$P(s) = \frac{K}{1 + \tau s}$$

- If we apply a PD controller then *C(s)* becomes:

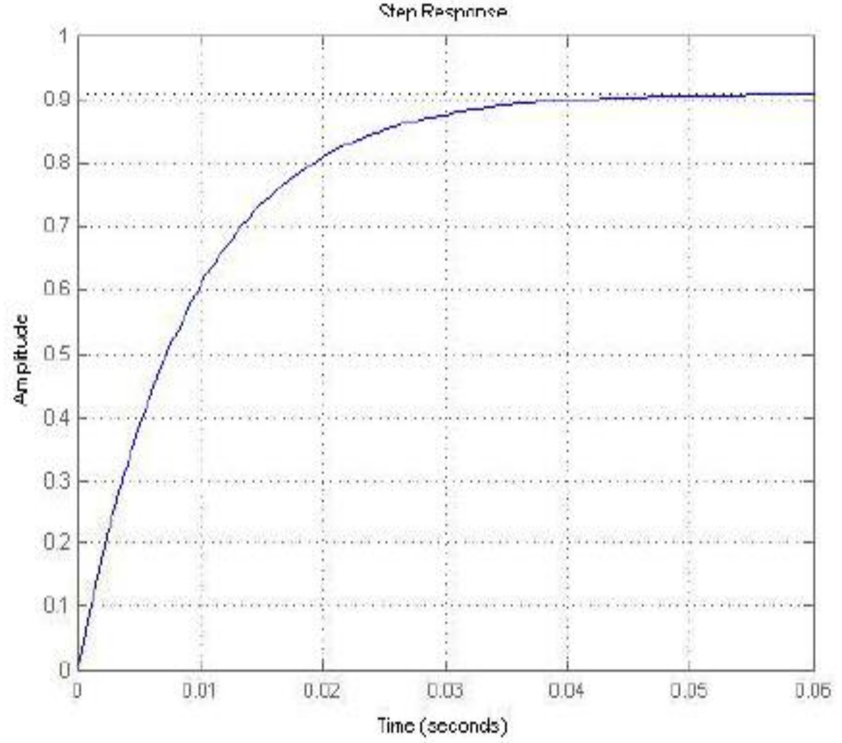
$$C(s) = K_p + K_d s = K_d\,(s + K_p / K_d)$$

- The closed loop transfer function may be written as:

$$\frac{Y(s)}{R(s)} = \frac{KK_d s + KK_p}{(\tau + KK_d)\,s \;\; + (1 + K\,K_p)}$$

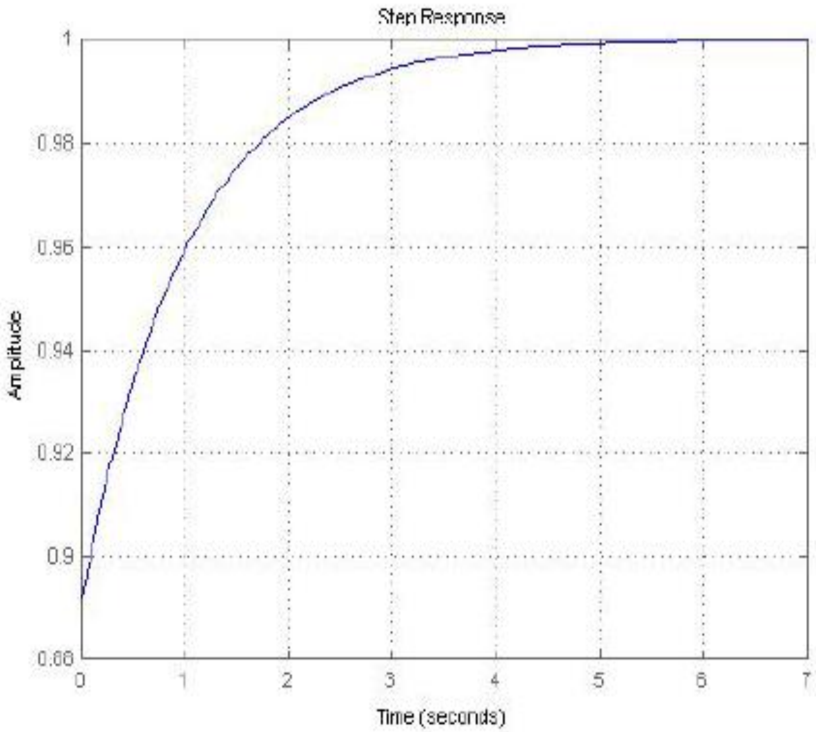
A numerical example

- Let us consider a numerical example where K=10 and $\tau = 0.1$. The response of the system without any compensator is as follows: Note that it took about 0.05 seconds for the system to reach the steady state which is 0.9.



A Modified Response

- If you now add a compensator with $K_p = 9$ and $K_d = 1$, you can see the change in step response as follows. You may observe that the same value (0.9) is obtained in less than 0.2 seconds.



How do we tune a PID Controller?

- In the last lecture, we have shown how by choosing the three different tuning constants of the PID compensator you can control the nature of the response. However, the question remains how can we obtain the actual values of these constants. Is it only through trial and error or are there any rules to obtain them?

- Sometimes in Industry people try to tune compensators intuitively. For example, we do know that the derivative constant helps to remove sluggish response of a system or the integral constant helps to remove offset errors. However, quite often it is found that these constants are interrelated. For example, increasing integral constant may help to improve the steady state response but it may increase the system overshoot.

- A popular rule for tuning which is being used since last century is known as Zeigler-Nichols rule. This was developed way back in 1942 and is still popular today.

Zeigler-Nichols Rule

- There are two sets of rules. These rules are based on transient response of the system. The system dynamics may or may not be known to us. Let us now consider the control output *U(s)* to be defined in terms of the tuning constants as follows:

$$U(s) = K_p [1 + \frac{1}{T_i s} + T_d s] E(s)$$

- The first rule is for systems for which the exact dynamics is unknown.
- Using only a proportional controller first increase the gain so much that the response of the system starts to show oscillatory behavior. The corresponding gain could be termed as K_0 and the corresponding time period as T_0.

- Now the constants are determined by using the following table-

The Tuning Table for PID Controller

Type of controller	K_p	T_i	T_d
P	$0.5K_0$	∞	0
PI	$0.45K_0$	$1/1.2T_0$	0
PID	$0.6K_0$	$0.5T_0$	$0.125T_0$

The transfer function corresponding to the PID controller may be written as

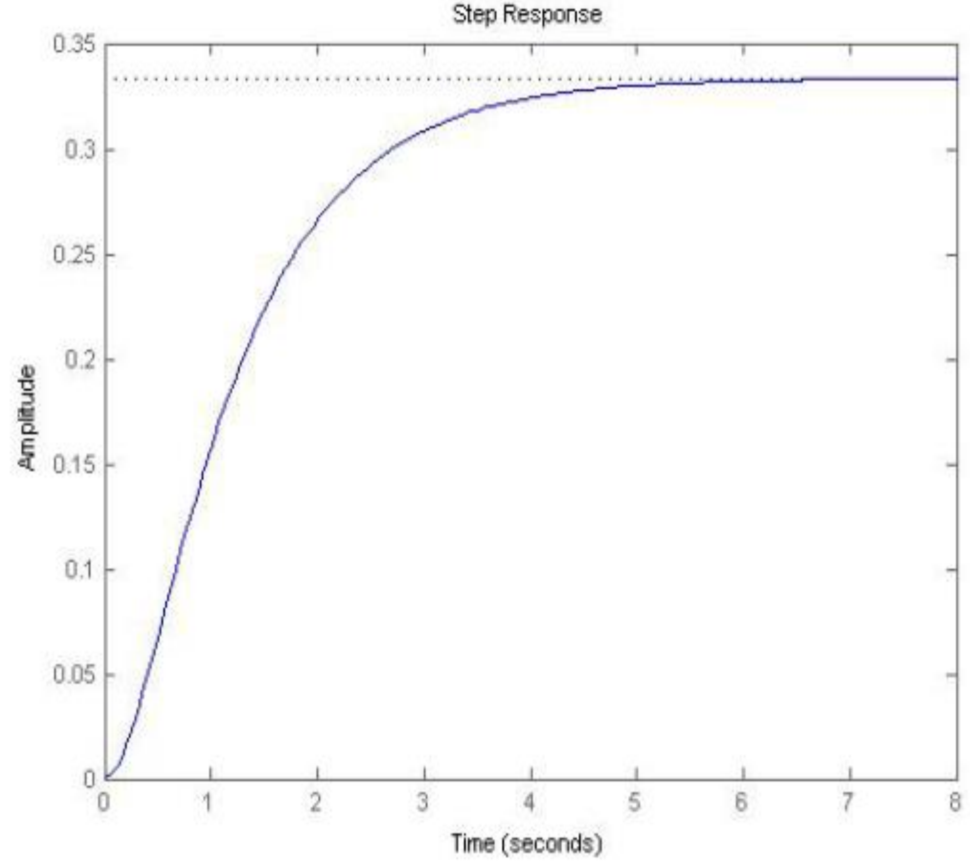
$$T(s) = 0.07K_0T_0\frac{(s+4/T_0)^2}{s}$$

Zeigler-Nichol's tuning for Known Systems

- Now consider a dynamic system for which the system parameters are known and hence mathematical modeling is available.
- Consider the system where $T_i = \infty$ and $T_d = 0$. Now for such a system, find out by using Routh's stability analysis the critical gain K_0 for which the roots cross the $j\omega$.
- Find out the frequency and the time period of the oscillation. Then, use the table mentioned earlier to determine the constants.
- These should be used as the starting point for tuning. Obtain the step response of the system and check whether the overshoot is less than 25%. If not, you should fine tune the system by moving the double zeros introduced by the PID controller.

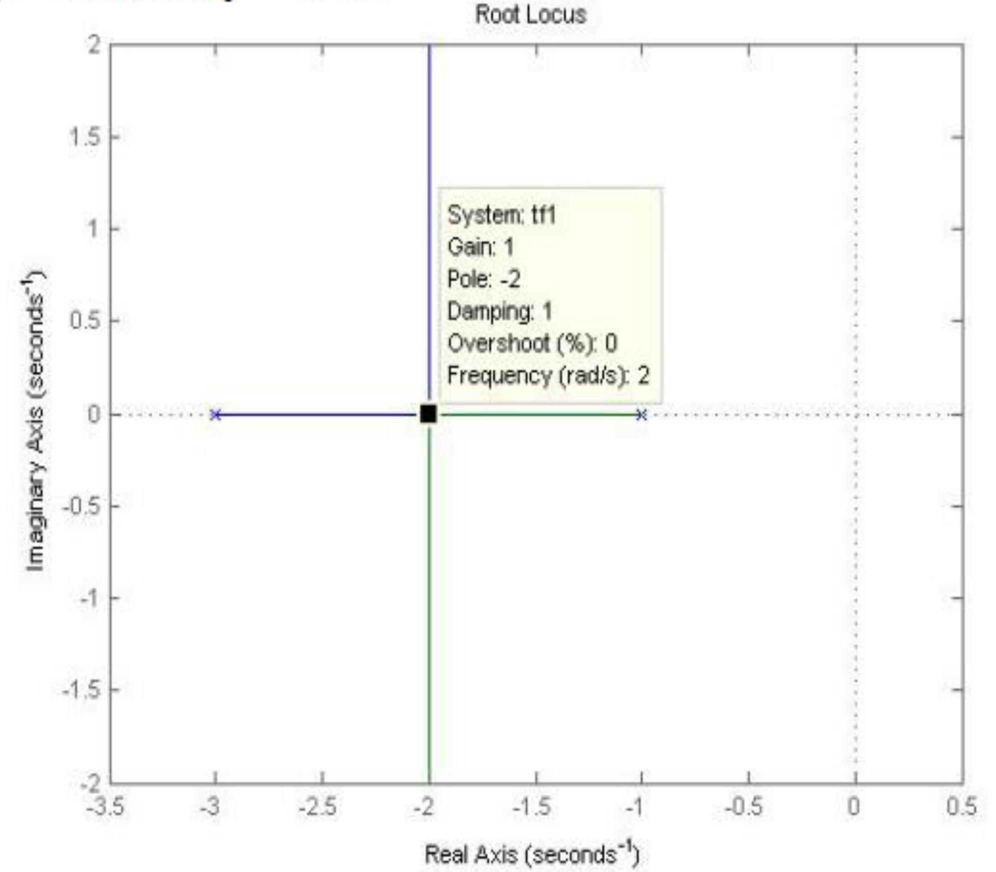
Simulation Example

Let us consider a 2nd-order system having two real poles at -1 and -3. The step response of the over-damped system is shown below:



Next step to find the Critical Gain:

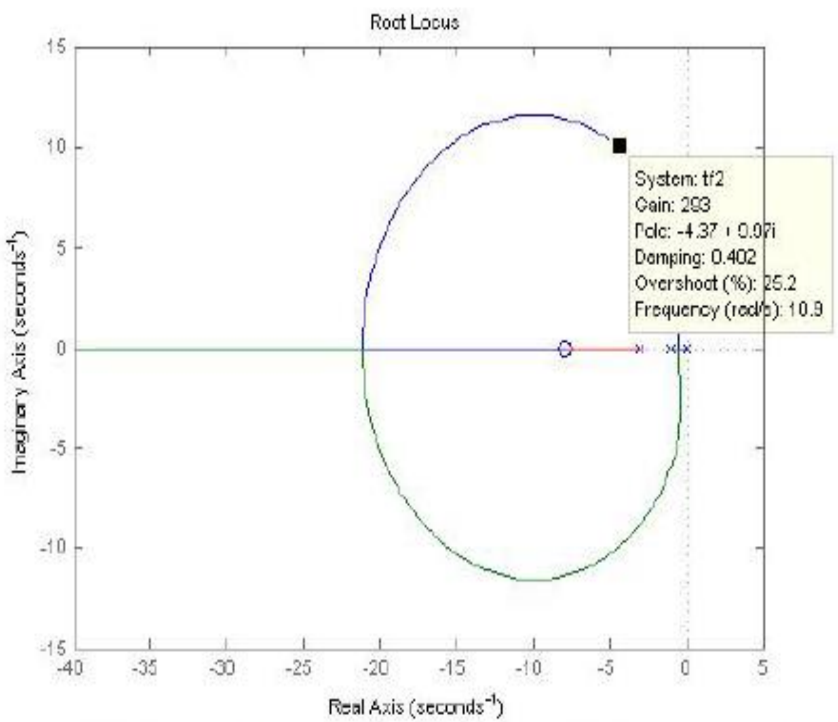
- Let us look at the root-locus of the system. At unit gain, the system will be just critically damped beyond which it will start to oscillate at 2 rad/s frequency. Hence, $K_0 = 1$ and $T_0 = 0.5$.



PID Controller following Z-N Rule:

- Following the **Z-N Table**, the controller transfer function may be written as:

$$T(s) = 0.035\frac{(s+8)^2}{s}$$



The resultant system is found to have increased robustness and at gain 233, the overshoot is about 25% which is quite acceptable.

Notch compensation

- Example system: (1/s(s+1))
  - Consider a system with a lead-lag controller

$$D(s) = 127 \frac{s+5.4}{s+20} \frac{s+0.03}{s+0.01}$$

  - Now, suppose the real system has an un-damped oscillation at about 50 rad/sec.
  - So, we include this oscillation in the model

$$G(s) = \frac{1}{s(s+1)} \frac{2500}{(s^2+s+2500)}$$

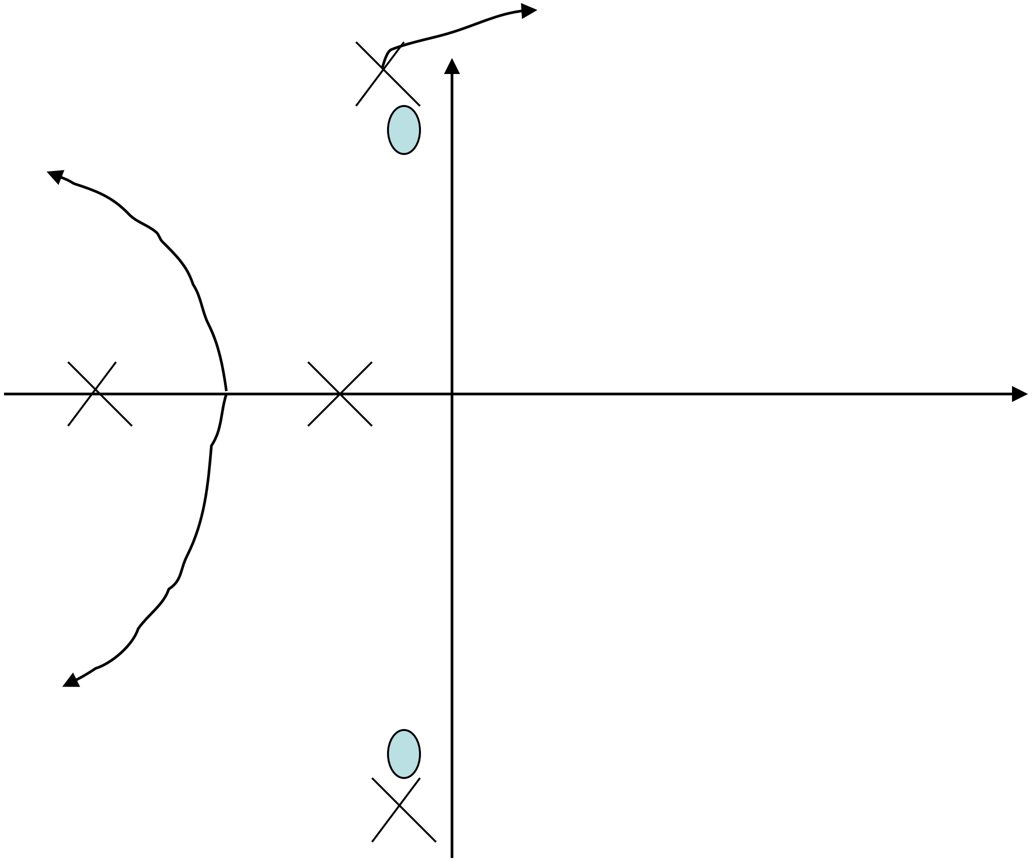
  - Can we use the original controller ?

Notch compensation

- Aim: Remove or dampen the oscillations
- Possibilities
  - Gain stabilization
    - Reduce the gain at high frequencies
    - Thus, insert poles above the bandwidth but below the oscillation frequency – might not be feasible
  - Phase stabilization (*notch compensation*)
    - A zero near the oscillation frequency
    - A zero increases the phase
    - Possible transfer function

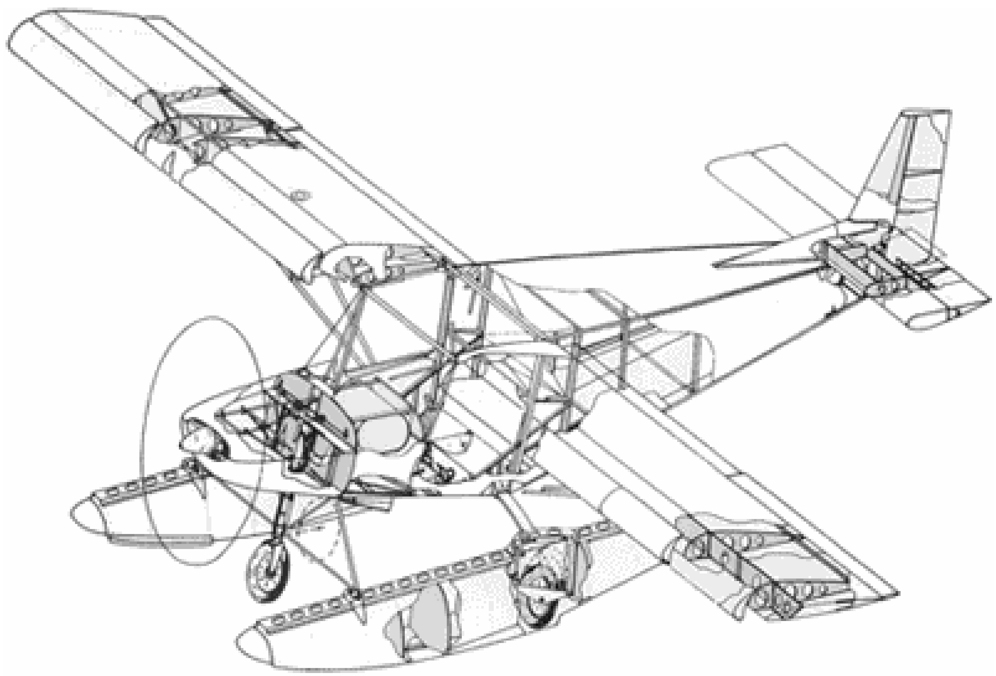
$$D_n(s) = \frac{s^2 + 2\zeta\omega_0 s + \omega_0^2}{(s+\omega_0)^2}$$

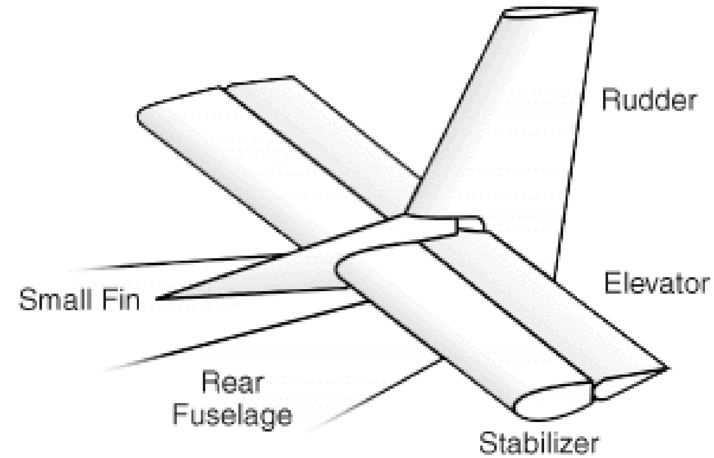
Design of a notch filter



Use zeroes close to low damping ratio poles

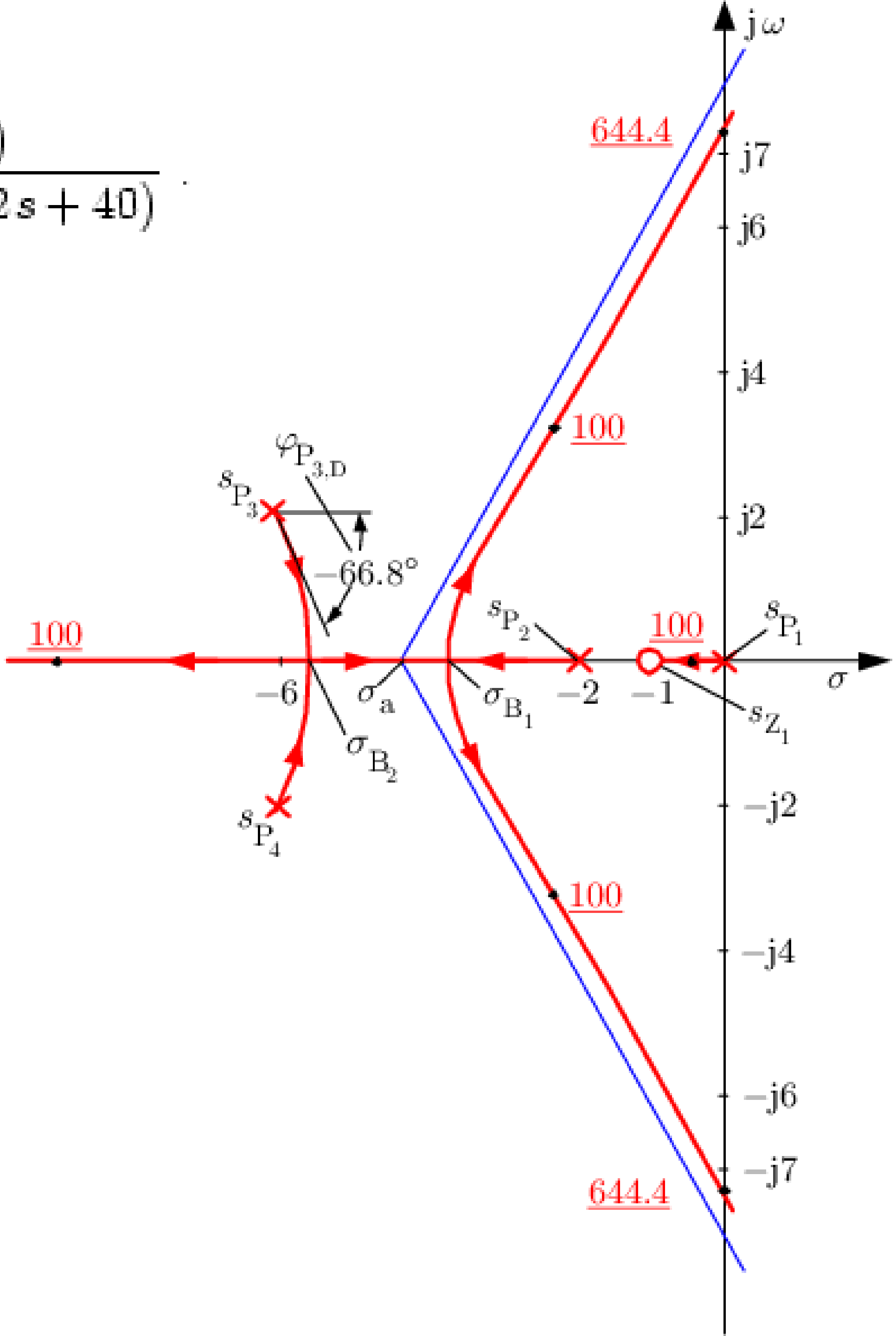
$$G_0(s) = \frac{k_0(s+1)}{s(s+2)(s^2+12s+40)}\ .$$





For the above transfer function (typical of an air-craft elevation control), design a Lead-Lag filter for 5 times reduction in steady state error and 3 times reduction in settling time. Assume

$$G_0(s) = \frac{k_0(s+1)}{s(s+2)(s^2+12s+40)}\ .$$



- Use of MATLAB for ROOT LOCUS DESIGN

```
num=[1.151 0.1774];
den=[1 0.739 0.921 0];
Wn=0.9; zeta=0.52;
rlocus (num,den)
sgrid (zeta,Wn)
axis ([-1 0 -2.5 2.5])
[K, poles]=rlocfind
(num,den) de=0.2;
[numc,denc]=cloop
(K*num,den,-1);
step (de*numc,denc)
```

Time Delay

  - Time delay always reduces the stability of a system !
  - Important to be able to analyze its effect
  - In the s-domain a time delay is given by e^{-ls}
  - Most applications contain delays (sampled systems)
- Root locus analysis
  - The original method does only handle polynomials
- Solutions
  - Approximation (Padé) of e^{-ls}
  - Modifying the root locus method (direct application)

Time Delay

First approximat ion

(1,1) Padé approximan t

$$e^{-s} \approx \frac{b_0 s + b_1}{a_0 s + 1}$$

McLauren series

$$e^{-s} = 1 - s + \frac{s^2}{2} + \frac{s^3}{3!} + \frac{s^4}{4!} + \dots$$

$$\frac{b_0 s + b_1}{a_0 s + 1} = b_1 + (b_0 - a_0 b_1)s$$

$$- a_0(b_0 - a_0 b_1)s^2 + a_0^2(b_0 - a_0 b_1)s^3$$

$$b_1 = 1$$

$$b_0 - a_0 b_1 = -1$$

$$- a_0(b_0 - a_0 b_1) = \frac{1}{2}$$

$$a_0^2(b_0 - a_0 b_1) = -\frac{1}{6}$$

$\Downarrow$ with $s \equiv T_d s$

$$e^{-T_d s} \cong \frac{1 - (T_d s / 2)}{1 + (T_d s / 2)}$$

Time Delay

Direct approach (exact calculatio n)

Process

$$G(s) = e^{-T_d s} G_0(s)$$

Notice, as $s = \sigma + j\omega$

$$\angle e^{-T_d s} = \angle(e^{-T_d \sigma} e^{-jT_d \omega}) = -T_d \omega$$

Modified root locus condition

$$\angle D(s)G(s) = \angle\big(D(s)G_0(s)\big) + \angle\left(e^{-T_d s}\right) = 180^\circ$$

$$\Downarrow$$

$$\underline{\angle D(s)G(s) = 180^\circ + T_d \omega}$$

However, Matlab does not support this approach...