Frequency Response Analysis

Lecture 7

Outline

- Introduction
- Bode Plot
- Polar Plot
- Log-magnitude-vs-phase plots
- Nyquist Stability Criterion
- Relative Stability
- Frequency response based techniques for controller design
    - Lead Compensator Design
    - Lag Compensator Design

Introduction

- Frequency response means steady-state response of a system to a sinusoidal input.
- In frequency-response methods, we vary the frequency of the input signal over a certain range and study the resulting response.
- Nyquist Stability Criterion enables us to investigate both the absolute stability and relative stabilities of linear closed-loop systems from a knowledge of their open-loop frequency response characteristics.

Steady-State Output to Sinusoidal Inputs

- Consider an LTI system,
$$\frac{Y(s)}{X(s)} = G(s)$$

- For a sinusoidal input $x(t) = X \sin \omega t$

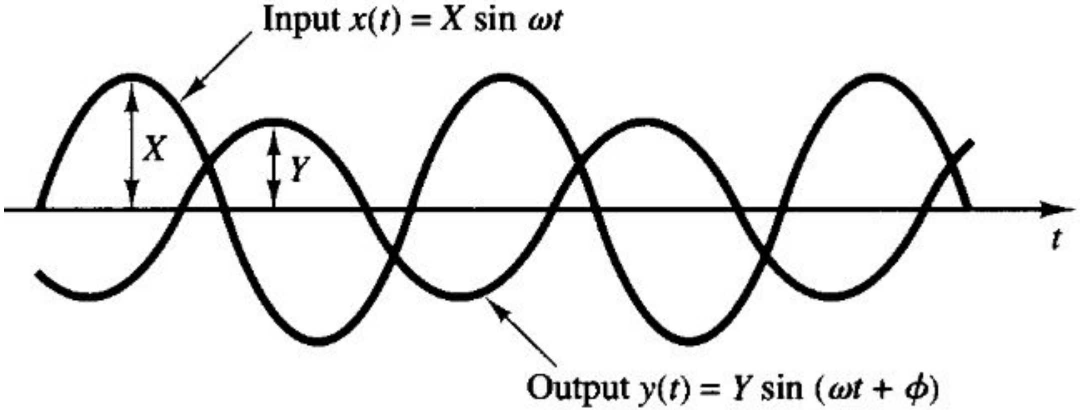
- System output is given by $y(t) = Y \sin(\omega t + \phi)$
  Where
$$Y = X\left|G(j\omega)\right|$$
$$\phi = \angle G(j\omega) = \tan^{-1}\left[\frac{\text{imaginary part of } G(j\omega)}{\text{real part of } G(j\omega)}\right]$$

Note that for sinusoidal inputs

$$|G(j\omega)| = \left|\frac{Y(j\omega)}{X(j\omega)}\right| = \text{amplitude ratio of the output sinusoid to the input sinusoid}$$

$$\angle G(j\omega) = \angle \frac{Y(j\omega)}{X(j\omega)} = \text{phase shift of the output sinusoid with respect to the input sinusoid}$$

$$\frac{Y(j\omega)}{X(j\omega)} = G(j\omega)$$



Frequency Response characteristics of a systems can be analyzed by using the following graphical forms:

- Bode plot
- Nyquist Plot / Polar Plot
- Log-magnitude vs phase plot

Bode Plot

- A Bode diagram consists of two graphs:
  - One is a plot of logarithm of magnitude versus frequency
  - Other one is a plot of phase angle versus frequency

- The logarithmic magnitude of G(jw) is given by $20\log|G(j\omega)|$ and its unit is decibel (dB). The base of logarithm is 10.
- The phase angle is in degrees.
- The main advantage of Bode diagram is that multiplication of magnitudes can be converted into addition.
- Secondly, it is easy to hand-draw approximations of Bode plot using asymptotic approximations.

Drawing Bode-plot by Hand

- The basic factors that frequently occur in $G(j\omega)H(j\omega)$ are of following types:

  1. Gain K
  2. Integral and derivative factors $(j\omega)^{\mp 1}$
  3. First-order factors $(1 + j\omega T)^{\mp 1}$
  4. Quadratic factors $[1 + 2\zeta(j\omega/\omega_n) + (j\omega/\omega_n)^2]^{\mp 1}$

- By knowing the logarithmic plots for these basic factors, it is possible to construct the composite logarithmic plot for any general $G(j\omega)H(j\omega)$

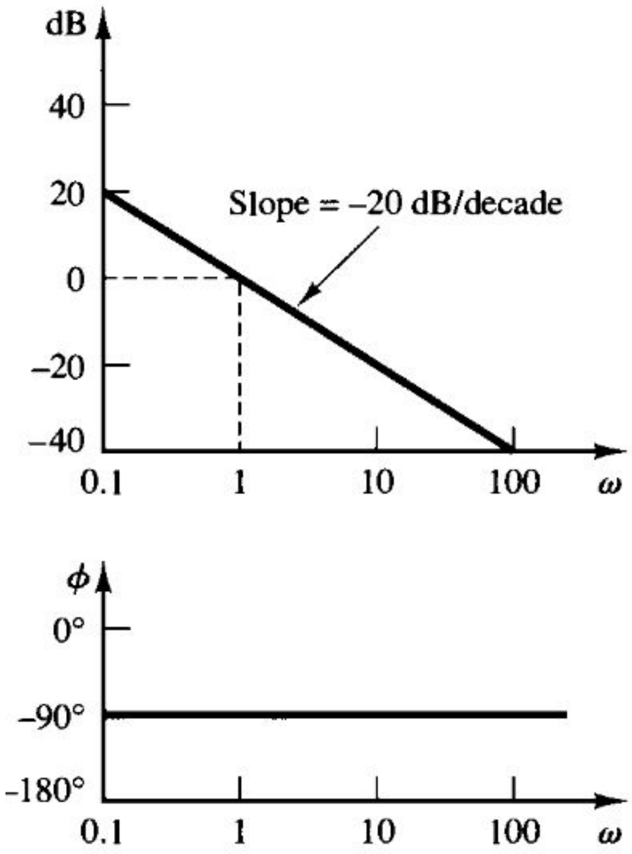
- **Constant gain K** - The logarithmic plot is a constant line of magnitude of 20 log K decibels.

- **Integral and derivative factors** $(j\omega)^{\pm}$

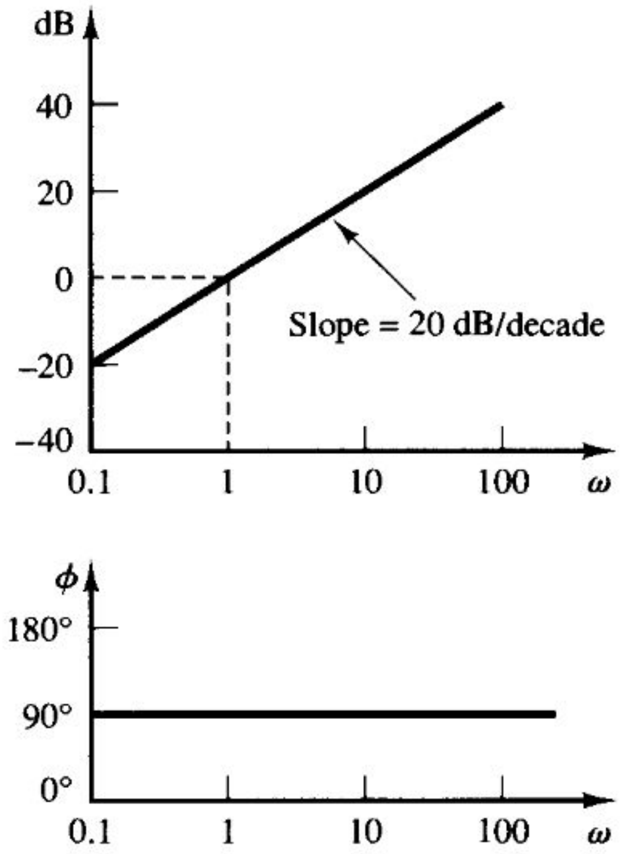
$$20 \log \left| \frac{1}{j\omega} \right| = -20 \log \omega \text{ dB} \qquad \angle\phi = -90^\circ$$

$$(-20 \log 10\omega) \text{ dB} = (-20 \log \omega - 20) \text{ dB}$$

$$20 \log |j\omega| = 20 \log \omega \text{ dB} \qquad \angle\phi = +90^\circ$$



Bode diagram of
$G(j\omega) = 1/j\omega$



Bode diagram of
$G(j\omega) = j\omega$

First Order Terms $(1 + j\omega T)^{\pm}$

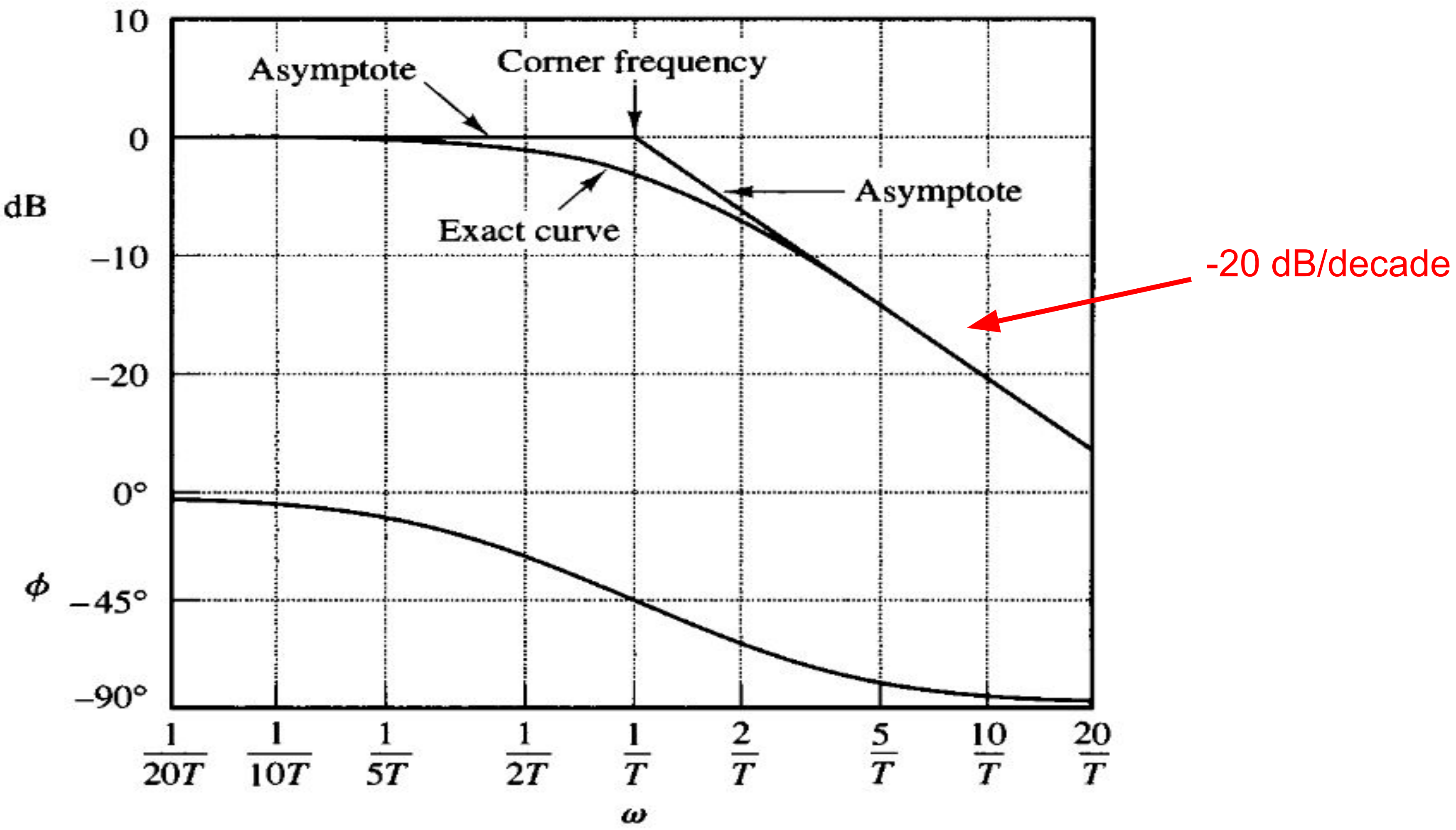
$$20 \log \left| \frac{1}{1 + j\omega T} \right| = -20 \log \sqrt{1 + \omega^2 T^2} \text{ dB}$$

For $\omega << \frac{1}{T}$, $-20 \log \sqrt{1 + \omega^2 T^2} \doteq -20 \log 1 = 0 \text{ dB}$

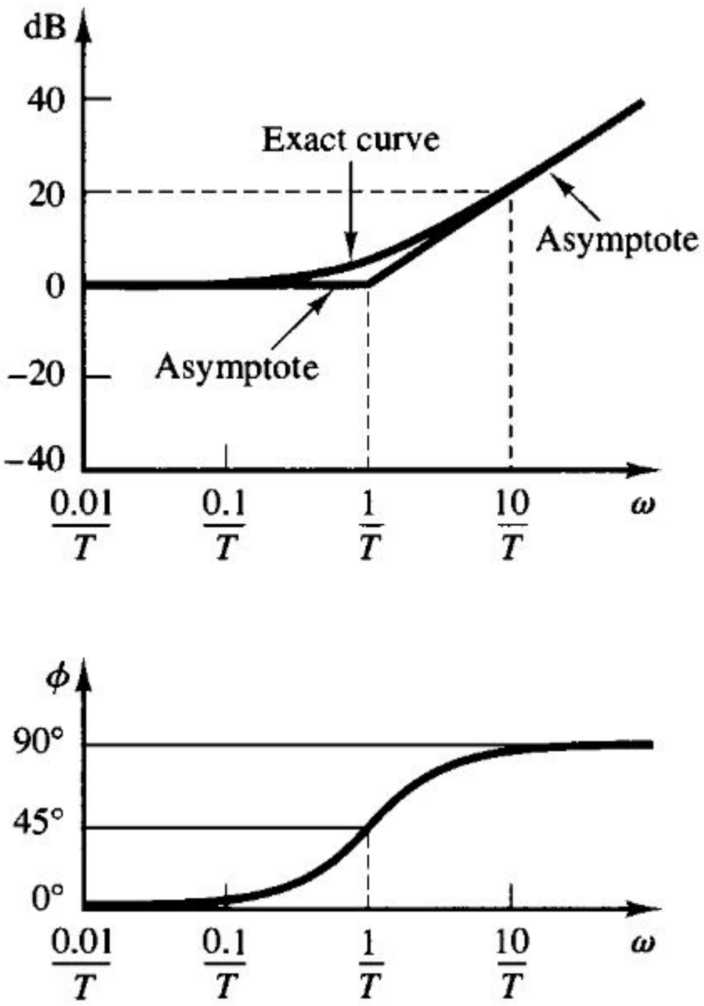
For $\omega >> \frac{1}{T}$, $-20 \log \sqrt{1 + \omega^2 T^2} \doteq -20 \log \omega T \text{ dB}$

$\phi = -\tan^{-1} \omega T$ Corner Frequency: $\omega = \frac{1}{T}$

$$\phi = 0, -45^\circ, -90^\circ; \text{ with } \omega = 0, \frac{1}{T}, \infty$$



10
0
dB
−10
−20
0°
ϕ
−45°
−90°
Asymptote
Corner frequency
Asymptote
Exact curve
-20 dB/decade
1/20T
1/10T
1/5T
1/2T
1/T
2/T
5/T
10/T
20/T
ω



$$20 \log \left|1 + j\omega T\right| = -20 \log \left|\frac{1}{1 + j\omega T}\right|$$

$$\angle 1 + j\omega T = \tan^{-1} \omega T = -\angle \frac{1}{1 + j\omega T}$$

Quadratic Factors: $\left[1 + 2\zeta(\frac{j\omega}{\omega_n}) + \left(\frac{j\omega}{\omega_n}\right)^2\right]^{\pm}$

$$20 \log \left| \frac{1}{1 + 2\zeta\left(j\frac{\omega}{\omega_n}\right) + \left(j\frac{\omega}{\omega_n}\right)^2} \right| = -20 \log \sqrt{\left(1 - \frac{\omega^2}{\omega_n^2}\right)^2 + \left(2\zeta\frac{\omega}{\omega_n}\right)^2}$$

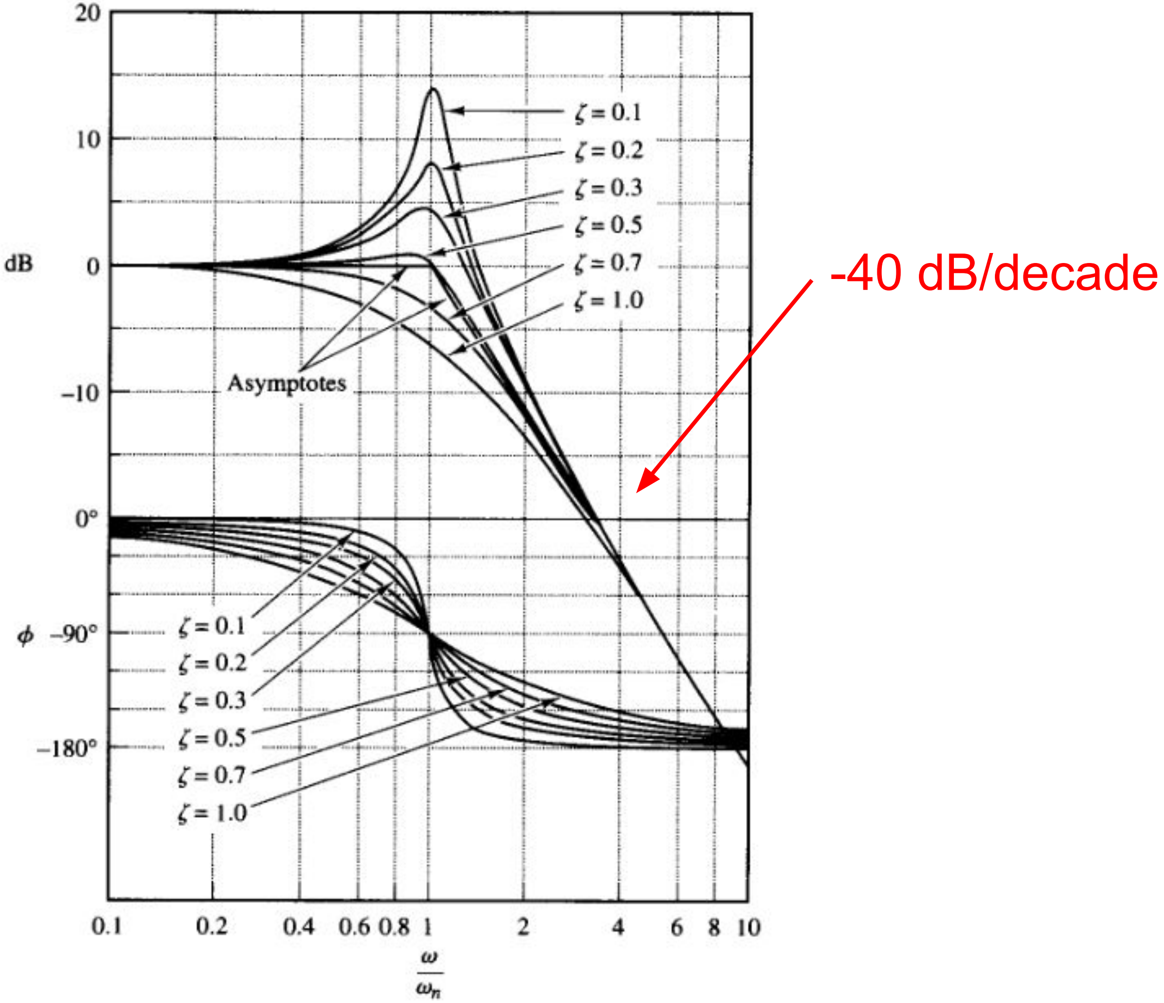
For $\omega << \omega_n$, $\quad -20 \log 1 = 0 \text{ dB}$

For $\omega >> \omega_n$, $\quad -20 \log \frac{\omega^2}{\omega_n^2} = -40 \log \frac{\omega}{\omega_n} \text{ dB}$

$$\phi = \angle \frac{1}{1 + 2\zeta\left(j\frac{\omega}{\omega_n}\right) + \left(j\frac{\omega}{\omega_n}\right)^2} = -\tan^{-1}\left[\frac{2\zeta\frac{\omega}{\omega_n}}{1 - \left(\frac{\omega}{\omega_n}\right)^2}\right]$$

w	0	w_n	\infinity
Phi	0	-90	-180

Corner Frequency: $\omega = \omega_n$



20
10
dB
0
−10
0°
ϕ −90°
−180°
ζ = 0.1
ζ = 0.2
ζ = 0.3
ζ = 0.5
ζ = 0.7
ζ = 1.0
Asymptotes
0.1
0.2
0.4
0.6
0.8
1
2
4
6
8
10
$\frac{\omega}{\omega_n}$
-40 dB/decade

Resonant Frequency and Resonant Peak

The frequency at which the magnitude reaches a peak value is called a resonant frequency and the corresponding magnitude is called resonant peak value.

$$G(j\omega) = \frac{1}{1 + 2\zeta\left(j\frac{\omega}{\omega_n}\right) + \left(j\frac{\omega}{\omega_n}\right)^2}$$

$$|G(j\omega)| = \frac{1}{\sqrt{\left(1 - \frac{\omega^2}{\omega_n^2}\right)^2 + \left(2\zeta\frac{\omega}{\omega_n}\right)^2}}$$

Magnitude will be maximum, when the denominator becomes minimum:

$$g(\omega) = \left(1 - \frac{\omega^2}{\omega_n^2}\right)^2 + \left(2\zeta\frac{\omega}{\omega_n}\right)^2$$

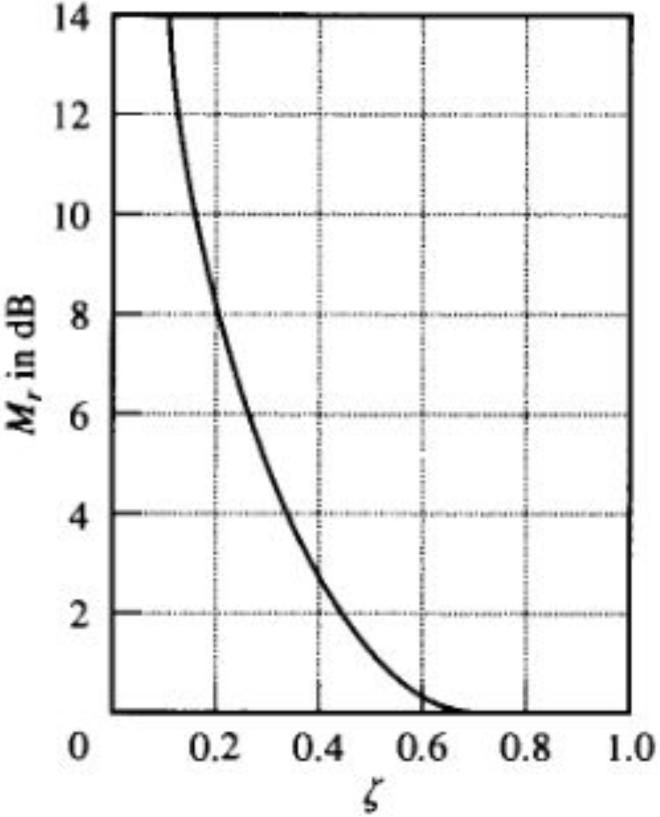
This gives:

$$\omega_r = \omega_n\sqrt{1 - 2\zeta^2}, \qquad \text{for } 0 \le \zeta \le 0.707$$

$$M_r = |G(j\omega)|_{\max} = |G(j\omega_r)| = \frac{1}{2\zeta\sqrt{1 - \zeta^2}} \qquad 0 \le \zeta \le 0.707,$$

For $\zeta > 0.707$, $\qquad M_r = 1$

$$\angle G(j\omega_r) = -\tan^{-1}\frac{\sqrt{1 - 2\zeta^2}}{\zeta} = -90^\circ + \sin^{-1}\frac{\zeta}{\sqrt{1 - \zeta^2}}$$



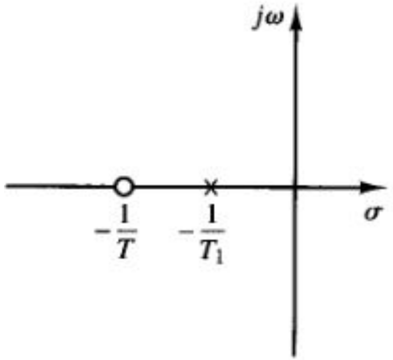
- Resonant Peak increases with decreasing damping ratio.
- As $\zeta \to 0, M_r \to \infty$

$$\text{At } \zeta = 0, \omega_r = \omega_n$$

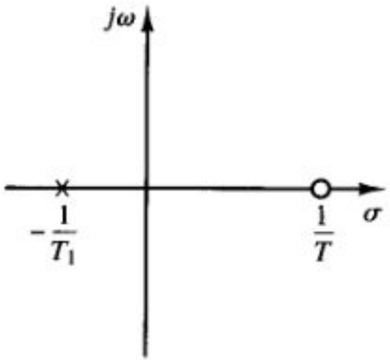
Minimum and Non-minimum phase Systems

- Transfer functions having neither poles nor zeros in the right half of the s-plane are called ***minimum-phase systems***.
- Transfer functions having one or more poles or zeros on the right-half of s-plane are called non-minimum phase systems.
- For systems with identical magnitude characteristics, the range in phase angle of the minimum-phase transfer function is minimum among all such systems.
- For minimum-phase systems, the transfer function can be uniquely derived from the magnitude curve alone. This is not so in case of non-minimum phase systems.
- For minimum phase systems, magnitude and phase angle characteristics are uniquely related.

$$G_1(j\omega) = \frac{1 + j\omega T}{1 + j\omega T_1}, \qquad G_2(j\omega) = \frac{1 - j\omega T}{1 + j\omega T_1} \qquad 0 < T < T_1$$

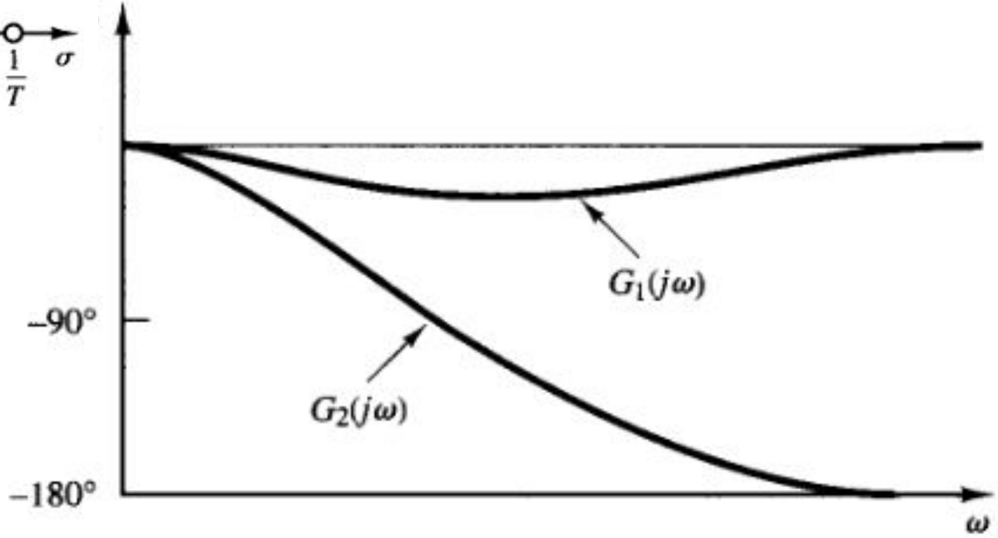


$G_1(s) = \frac{1 + Ts}{1 + T_1 s}$



$G_2(s) = \frac{1 - Ts}{1 + T_1 s}$

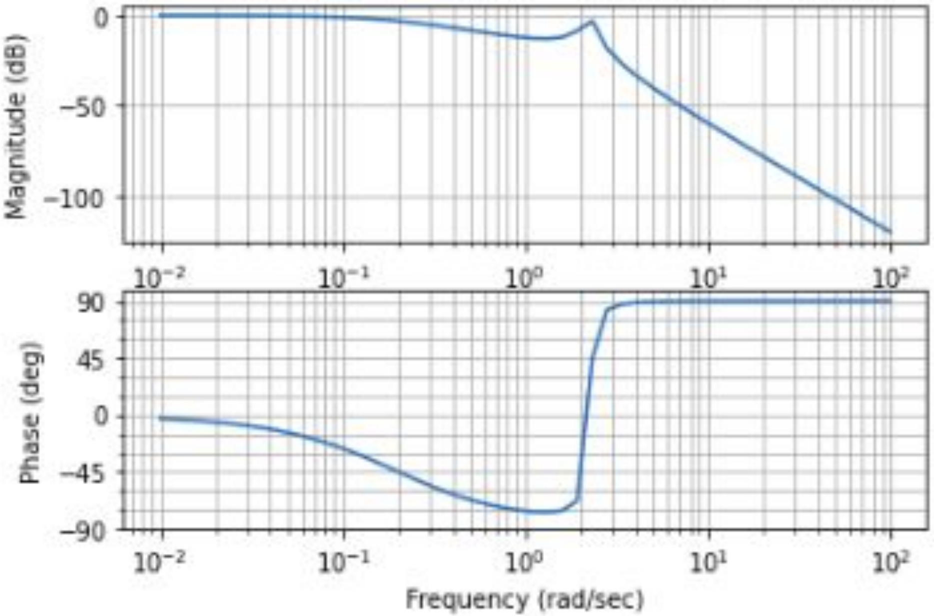
Magnitude is always 1 for both the systems.



Phase variation is minimum in minimum phase systems.

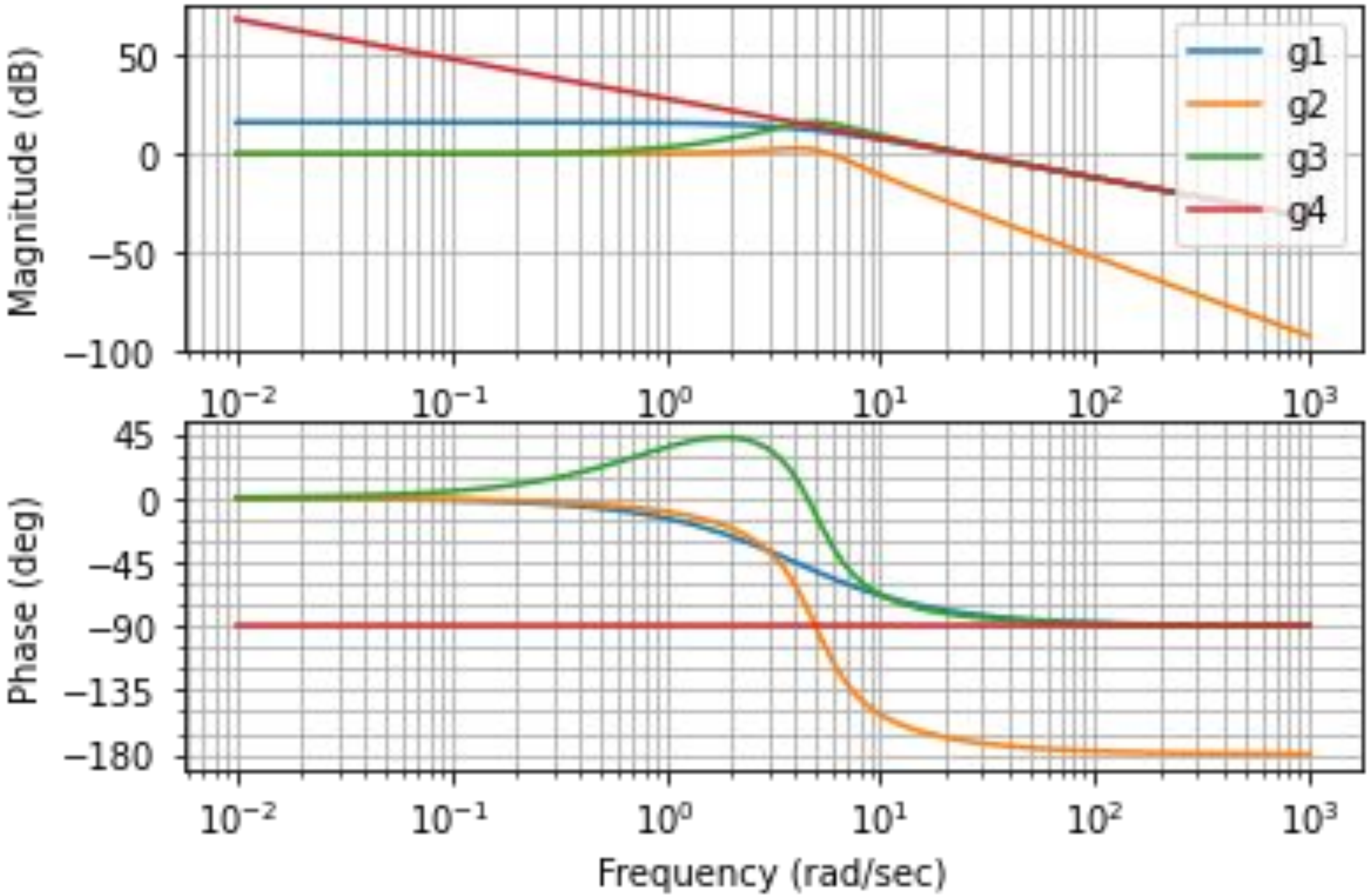
Example 1

```python
from control import *

g = tf([1], [1, 0, 5, 1])

mag, ph, w = bode_plot(g, dB=True)
```



Bode Plot for Multiple Systems

```python
import numpy as np
import matplotlib.pyplot as plt
from control import *

g1 = tf([25],[1, 4])
g2 = tf([25], [1, 4, 25])
g3 = tf([25, 25], [1, 4, 25])
g4 = tf([25], [1,0])
w = np.logspace(-2,3,100)
m,p,w = bode([g1,g2, g3, g4], w, dB=True)
fig = plt.gcf()
fig.axes[0].legend(['g1', 'g2', 'g3', 'g4'],
                   loc='best')
```

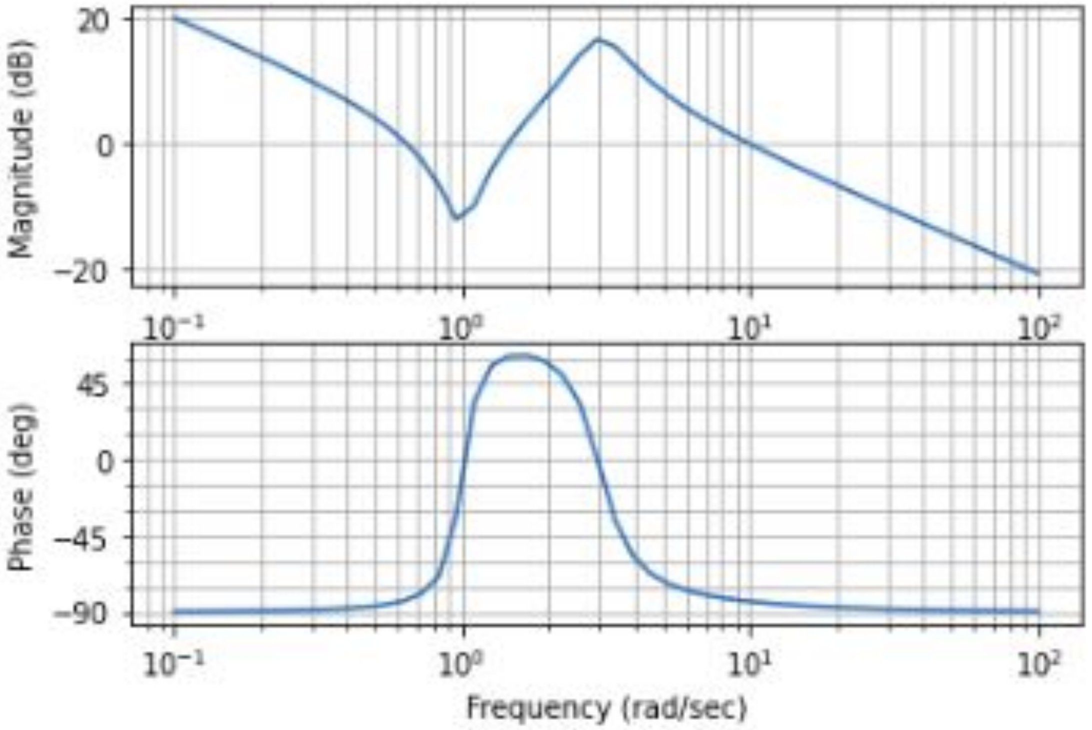


Magnitude (dB)
50
0
−50
−100
g1
g2
g3
g4
10^{-2}
10^{-1}
10^{0}
10^{1}
10^{2}
10^{3}
Phase (deg)
45
0
−45
−90
−135
−180
Frequency (rad/sec)

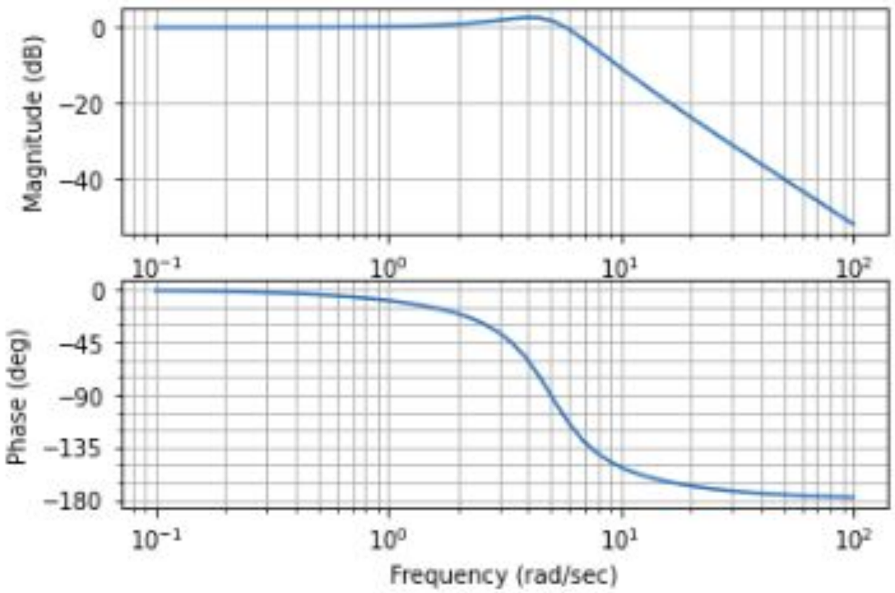
Example 2:

Example 3

```
from control import *
g = tf([9, 1.8, 9], [1, 1.2, 9, 0])
m,p,w = bode(g, dB=True)
```



```python
from control import *
A = [[0,1],[-25,-4]]
B = [[0],[25]]
C = [1, 0]
D = [0]
sys = ss(A,B,C,D)
m,p,w = bode(sys, dB=True)
```

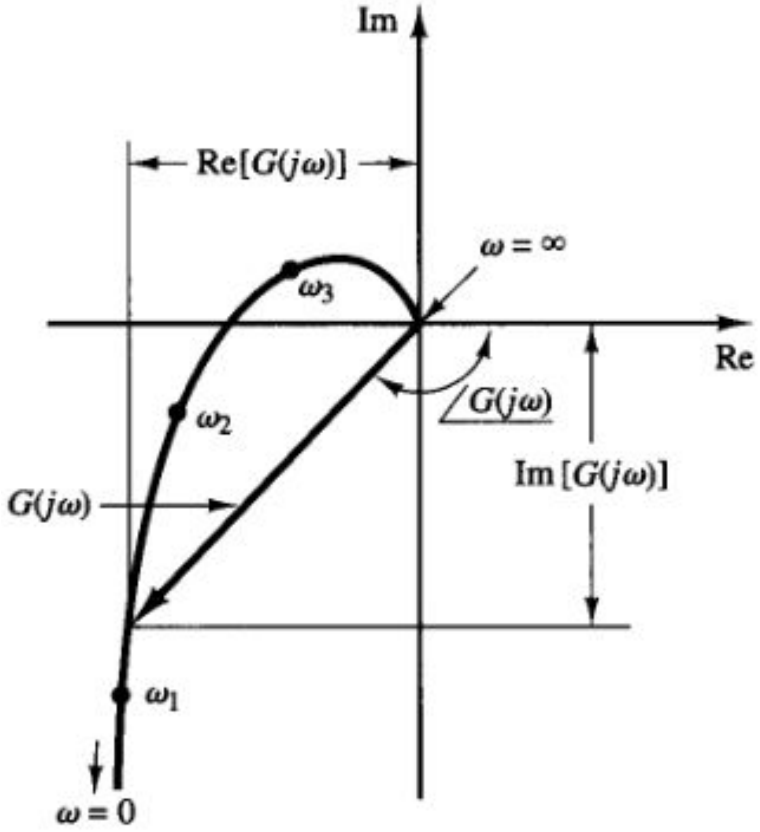


Example 4:

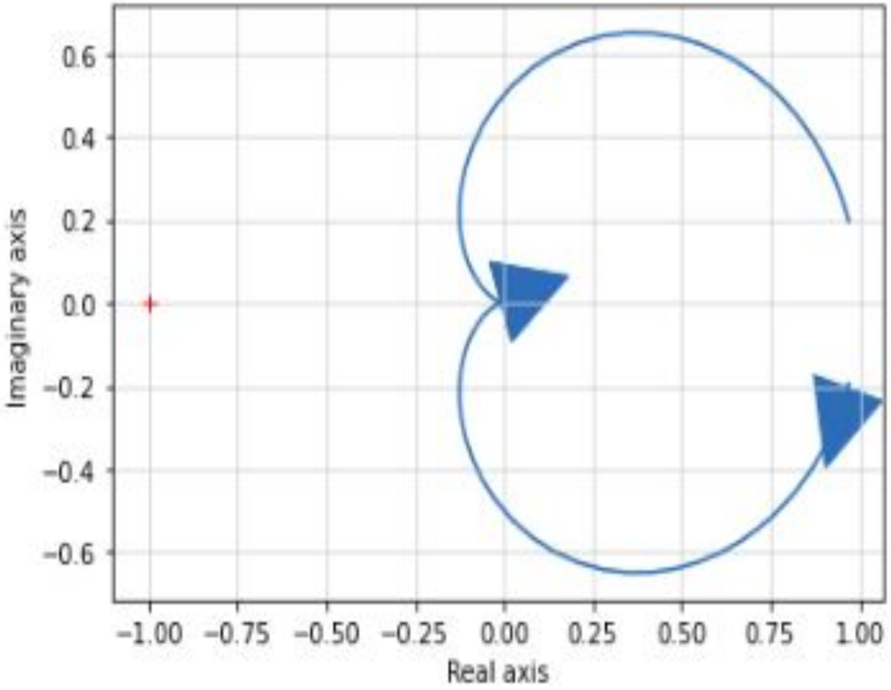
System can be specified in state-space model as well.

Polar Plots / Nyquist Plots

- The polar plot of a transfer function G(jw) is a plot of magnitude |G(jw)| versus phase angle of G(jw) on polar coordinates as frequency w is varied from 0 to infinity.
- The polar plot is the locus of vector $|G(j\omega)|\angle G(j\omega)$ as w is varied from 0 to infinity.
- Positive phase angle is measured counterclockwise from the positive real axis.
- Polar plot is also known as Nyquist plot.

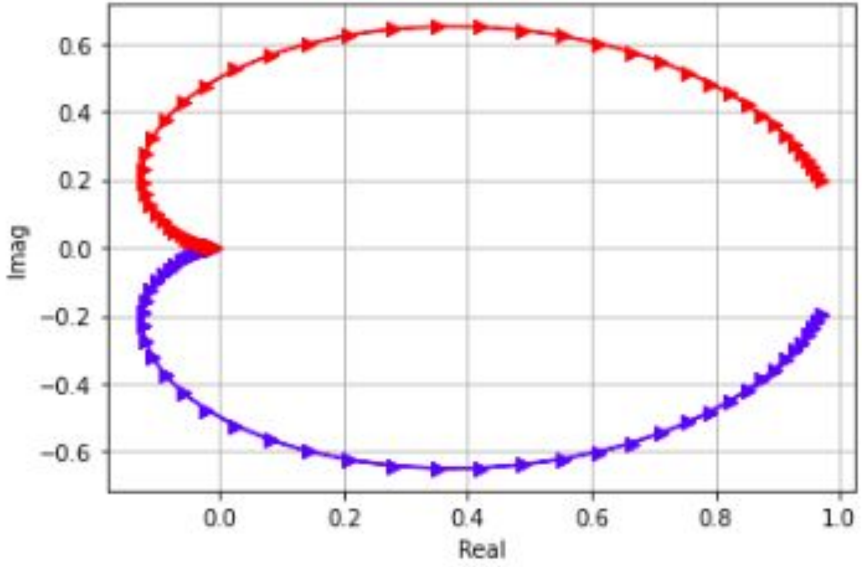


```python
from control import *
g = tf([1], [1,2,1])
real,imag,w = nyquist_plot(g)
```

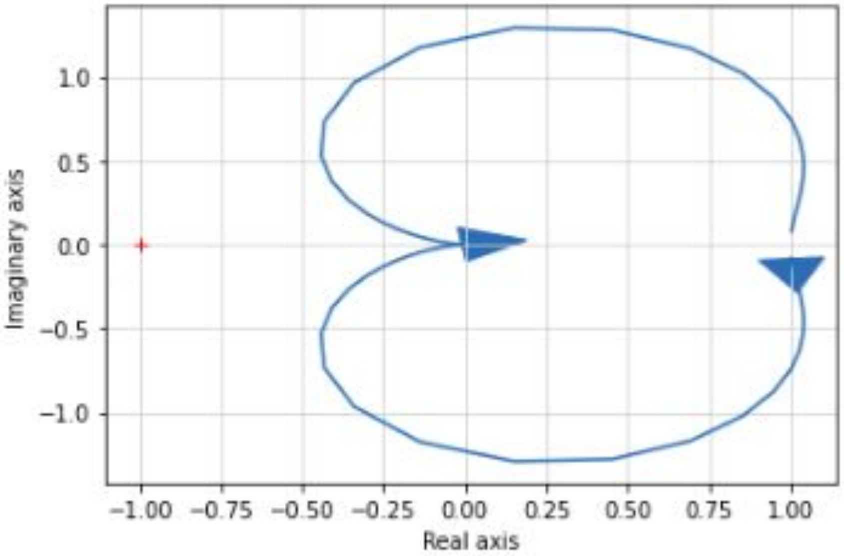



```python
import matplotlib.pyplot as plt
plt.plot(real,imag, 'b->')
plt.plot(real, -1*imag, 'r->')
plt.grid()
plt.xlabel('Real')
plt.ylabel('Imag')
```

```
Text(0, 0.5, 'Imag')
```

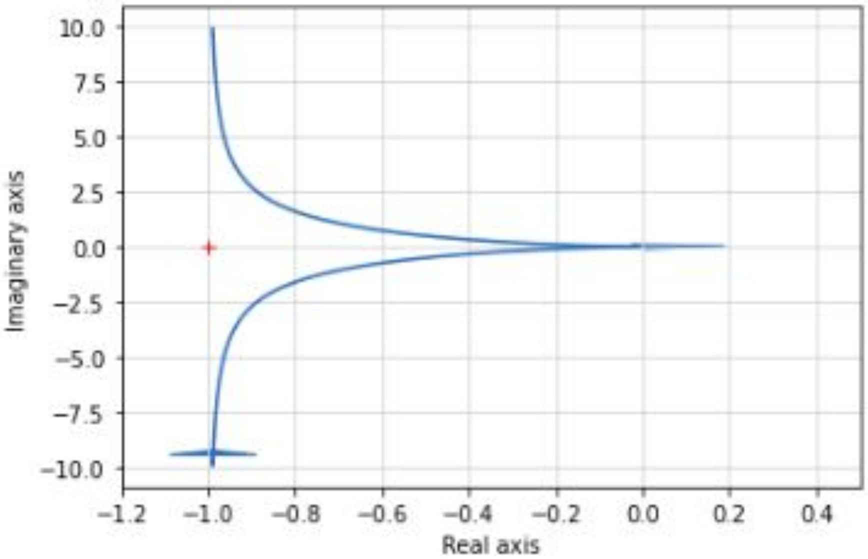


```python
from control import *
g = tf([1],[1, 0.8, 1])
r,i,w = nyquist(g)
```



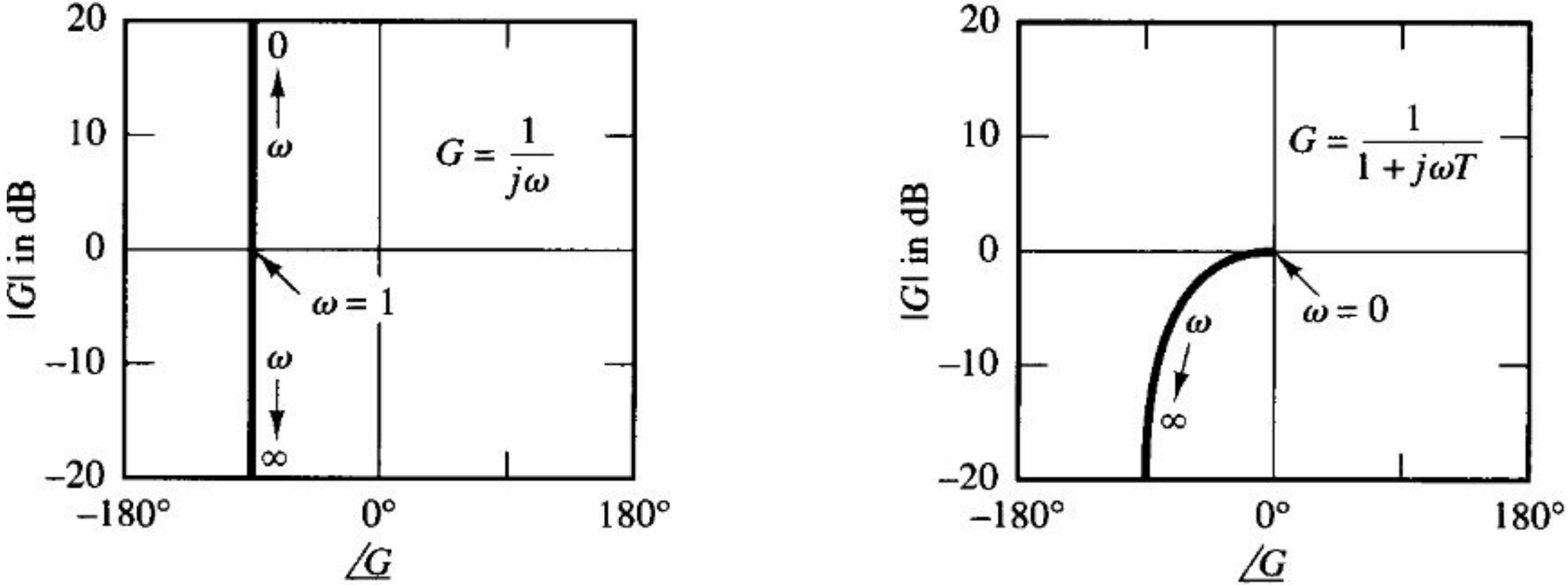

```python
from control import *
g = tf([1],[1,1,0])
r,i,w = nyquist(g)
plt.xlim((-1.2, 0.5))
```

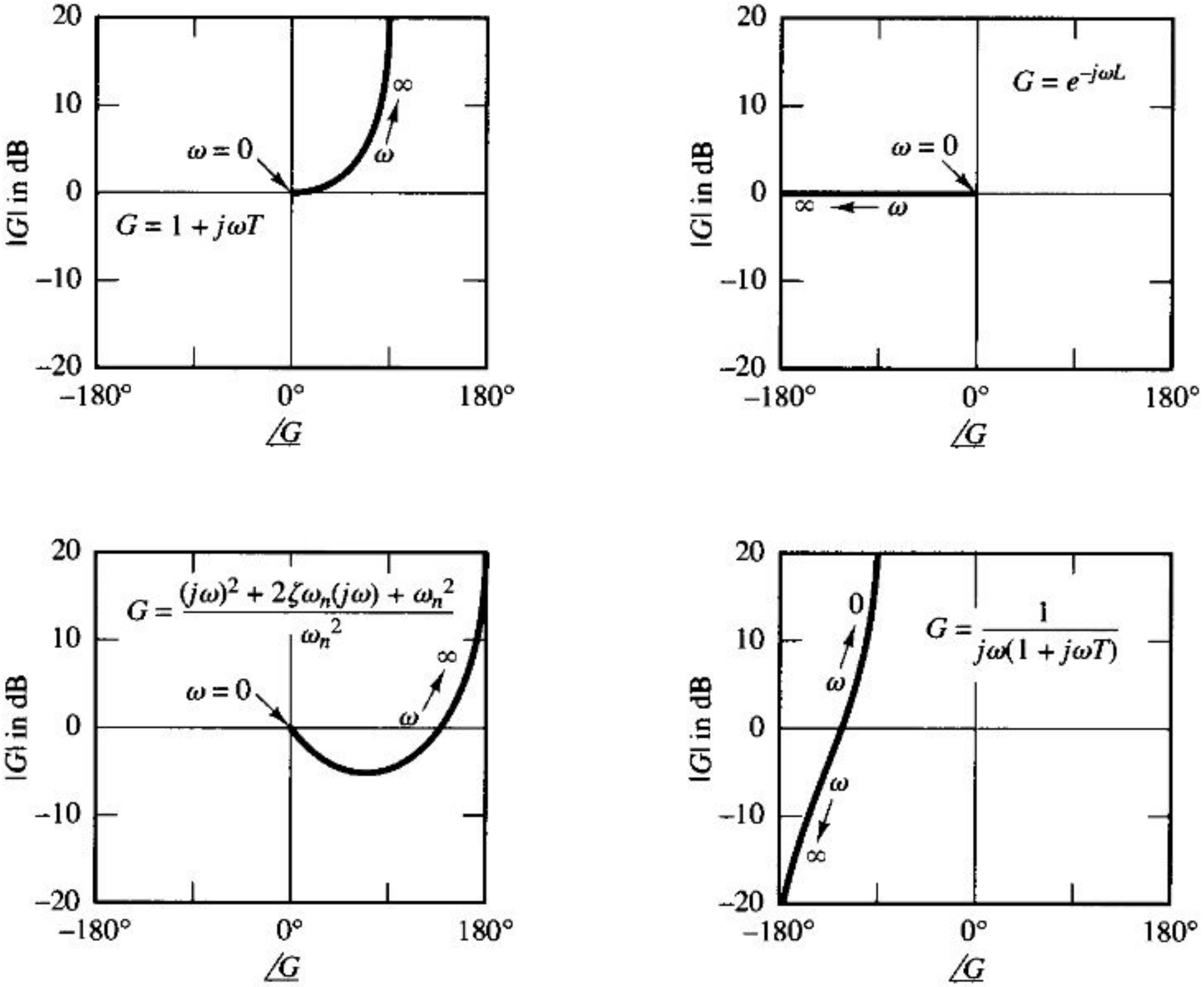
```
(-1.2, 0.5)
```



Log Magnitude Vs Phase Plot

The log magnitude in decibel is plotted against phase angle for a given range of frequencies.





$G = 1 + j\omega T$
$\omega = 0$
ω
∞
$G = e^{-j\omega L}$
$\omega = 0$
∞
ω
$G = \frac{(j\omega)^2 + 2\zeta\omega_n(j\omega) + \omega_n^2}{\omega_n^2}$
$\omega = 0$
ω
∞
$G = \frac{1}{j\omega(1 + j\omega T)}$
0
ω
ω
∞
|G| in dB
20
10
0
−10
−20
−180°
0°
180°
$\angle G$

Mapping Theorem

- Let ***F(s)*** be a ratio of two polynomials in ***s***.
- Let ***P*** be the number of poles and ***Z*** be the number of zeros of ***F(s)*** that lie inside some closed contour in s plane, with multiplicity of poles and zeros accounted for.
- Let this contour be such that it does not pass through any poles or zeros of ***F(s)***.
- This closed contour in the s-plane is then mapped into the ***F(s)*** plane as a closed curve.
- The total number ***N*** of clockwise encirclement of origin of the ***F(s)*** plane, as a representative point s traces out the entire contour in the clockwise direction, is equal to ***Z-P***.

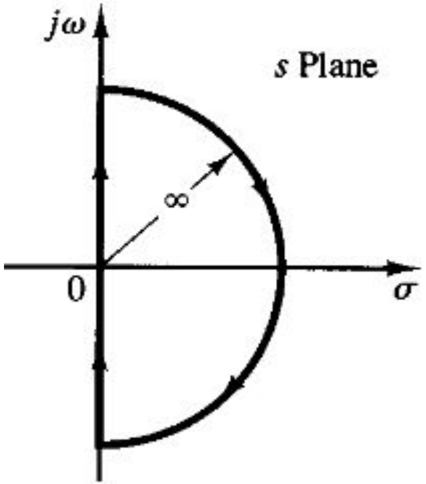
- Let ***F(s) = 1 + G(s)H(s)***

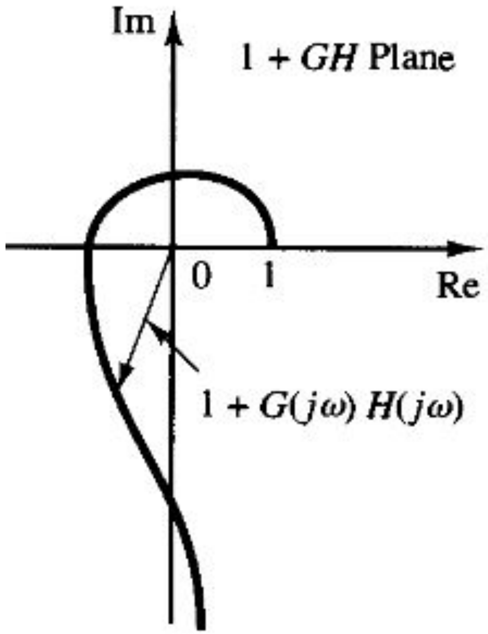
- ***Z*** = No. of Zeros of F(s) = No. of ***closed-loop poles*** (roots of characteristic equation: 1+GH(s) = 0
- **P** = No. of poles of F(s) = No. of ***open-loop poles*** of GH(s).

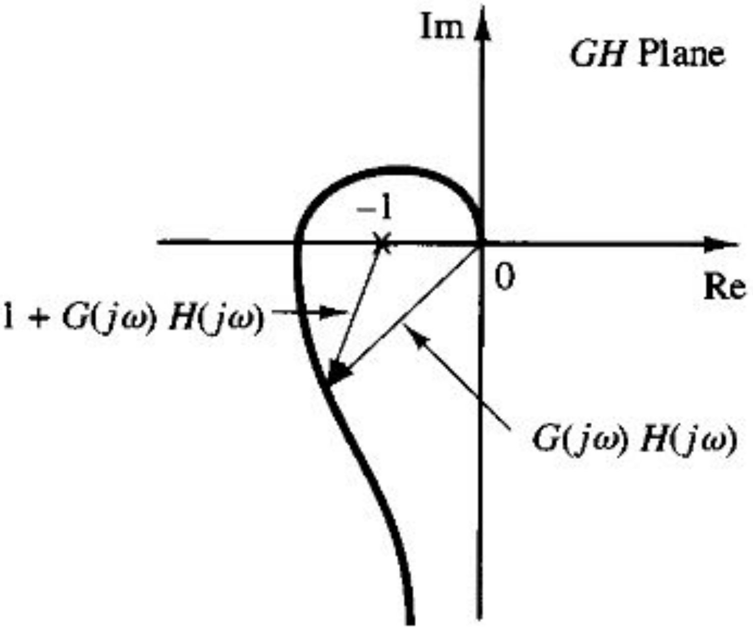
- If the closed contour in s-plane encloses the entire right-half of s-plane, then
  Z = P + N
  Where N is the number of clockwise encirclement of origin of 1+GH(s) plane, which is the same as the number of clockwise encirclement of the point -1+j0 in the GH plane.

- For stability, ***Z = 0*** and hence, ***N = -P*** or the GH plot encircles the -1+j0 point P times in the counterclockwise direction.

- Assuming that $\lim_{s \to \infty} [1 + G(s)H(s)] = \text{constant}$, the encirclement of origin of F(s) can be analyzed by considering only the jw axis where w varies from -infty to + infty



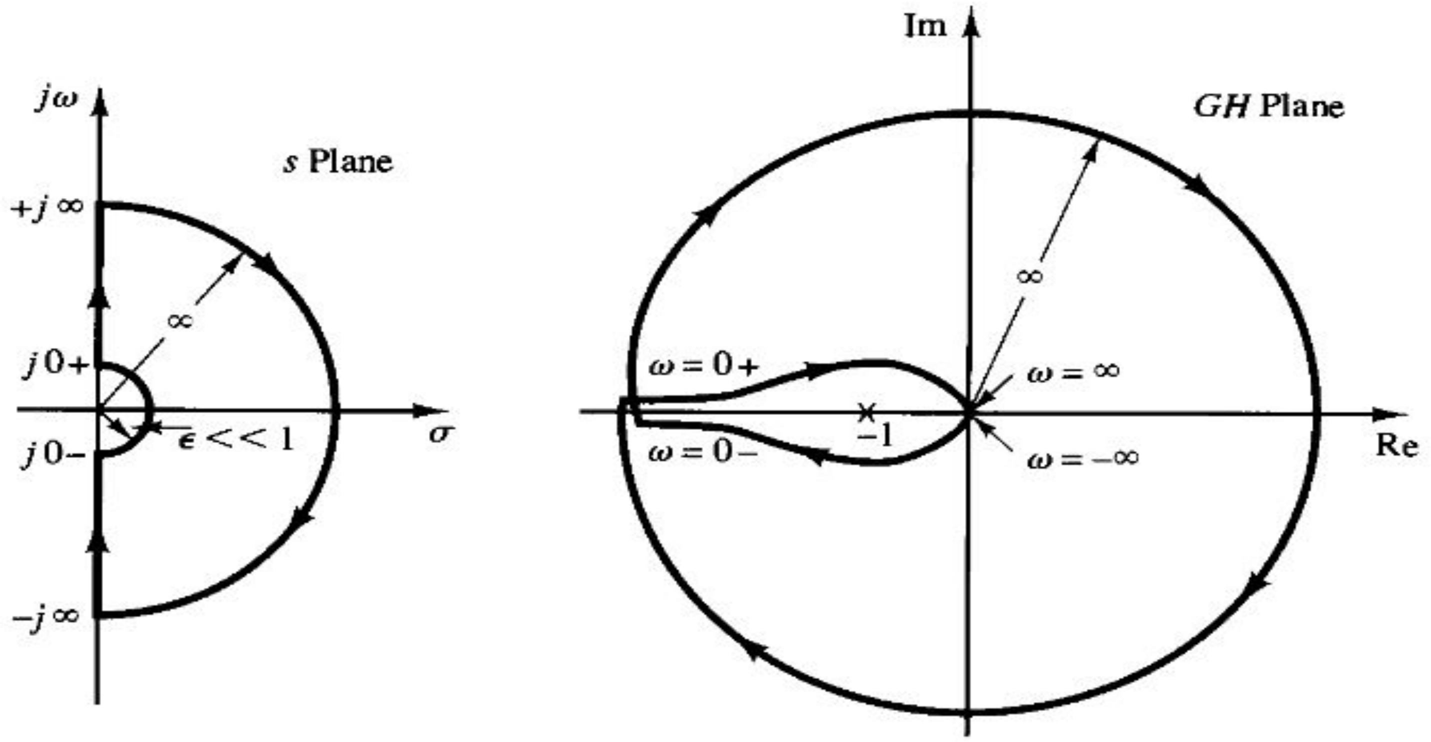




It is assumed that s-plane contour does not pass through the poles or zeros of F(s) = 1 + GH(s).

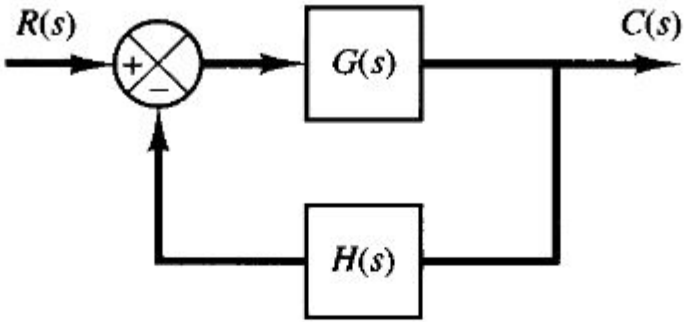
$$\lim_{s\to\infty} [1 + G(s)H(s)] = \text{constant}$$

GH(jw) and GH(-jw) are symmetric about the real-axis.



If there are poles or zeros on the jw axis, a small detour is taken around these singularities to form the s-contour, also known as nyquist contours.

Nyquist Stability Criterion



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

- For stability, the roots of characteristic equation 1+G(s)H(s)=0 must lie in the left-half s-plane.

- NSC relates the open-loop frequency response GH(jw) to the number of zeros and poles of 1+G(s)H(s) that lie in the right-half of s-plane.

- The absolute stability of closed-loop system can be determined graphically from open-loop frequency response curves without finding the closed-loop poles (for which computers were needed).

- If the open-loop transfer function G(s)H(s) has k poles in the right half of the s-plane, then for stability the G(jw)H(jw) locus, as w varies from 0 to infinity (as s traces the nyquist path in s-plane), must encircle the -1 + j0 point k times in the counterclockwise direction.

$$Z = N + P$$

where Z = number of zeros of $1 + G(s)H(s)$ in the right-half s plane
N = number of clockwise encirclements of the $-1 + j0$ point
P = number of poles of $G(s)H(s)$ in the right-half s plane

- For the stability of closed-loop system, Z = 0, therefore N = -P.
- If G(s)H(s) does not have any poles in the right-half of s-plane, Z = N. Hence for stability, there should not be any encirclement of the point -1+j0

Example

$$G(s)H(s) = \frac{K(s+3)}{s(s-1)}$$

For $g_1, K = 1$

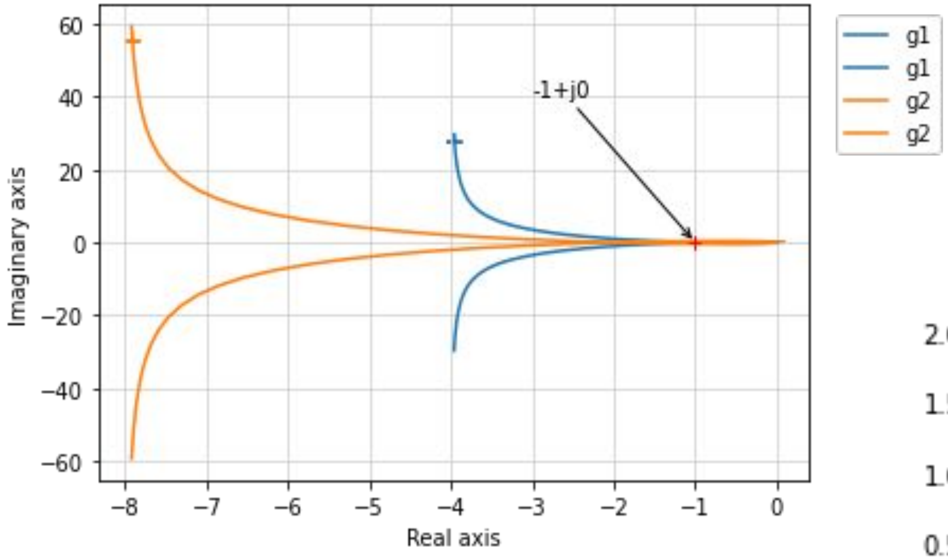
For $g_2, K = 2$

$$P = 1$$

For the closed-loop system to be stable, the following Nyquist condition must be satisfied.

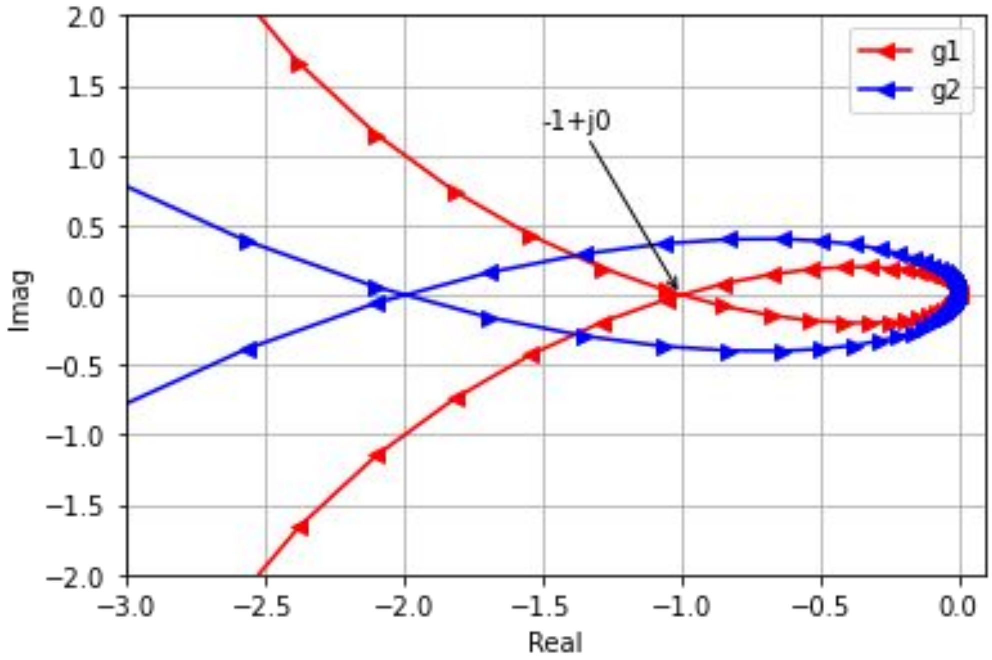
$$N = -P$$

In other words, Nyquist plot must encircle the point -1+j0 counterclockwise P times.



When k = 1, Nyquist plot passes through the critical point and hence, the CL system is unstable.

When k = 2, Nyquist plot (shown in blue) encircles the critical point once in counterclockwise direction once, thereby satisfying the Nyquist stability criterion. So, the CL system is stable.



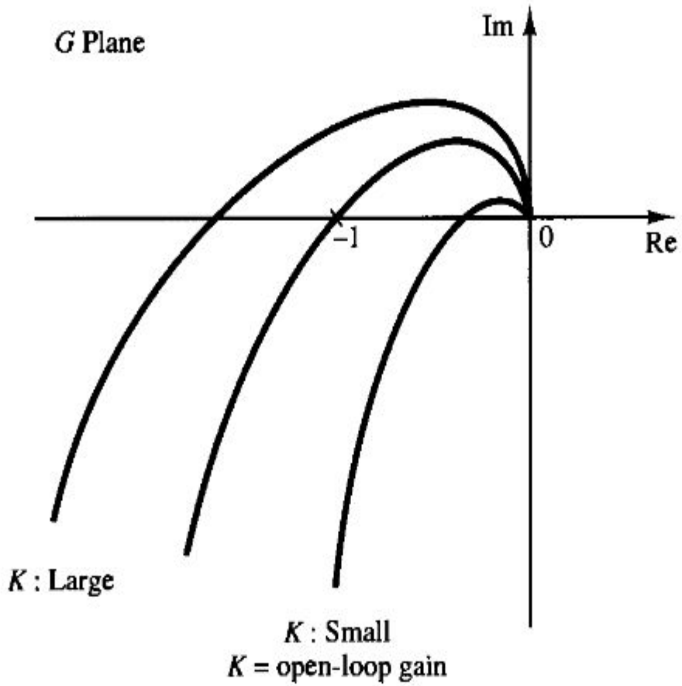
Relative Stability

- In designing controllers, it is necessary to have absolute stability for the closed-loop system.

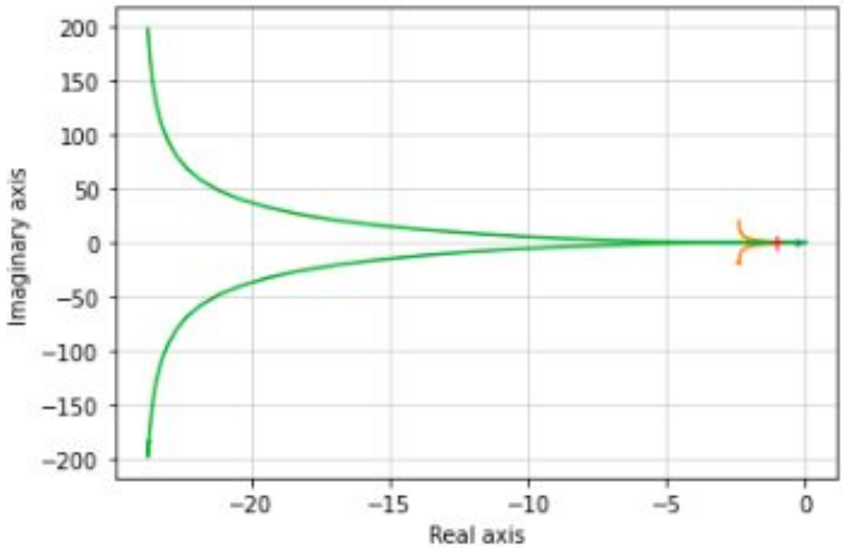
- In addition, it is important to have adequate relative stability.

- Nyquist plot can be used to measure the degree of stability for a stable system.

- The closeness of G(jw) plot to -1+j0 point can be used as a measure of relative stability.

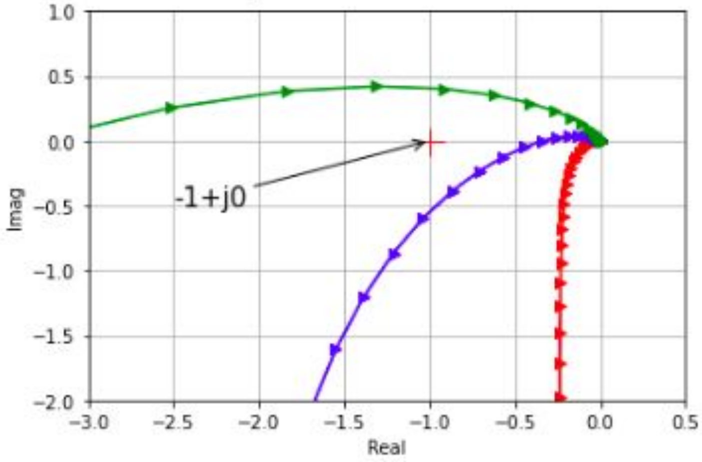


```python
from control import *
import matplotlib.pyplot as plt

g1 = tf([1],[1,6,5,0])
g2 = tf([10],[1,6,5,0])
g3 = tf([100], [1,6,5,0])
r1,i1,w1 = nyquist(g1)
r2,i2,w2 = nyquist(g2)
r3,i3,w3 = nyquist(g3)
```




```python
plt.plot(r1,i1,'r->')
plt.plot(r2,i2,'b->')
plt.plot(r3,i3, 'g->')
plt.plot(-1,0, 'r+', markersize=15)
#plt.text(-1.2,0.6, '-1+j0', fontsize=15)
plt.annotate('-1+j0', xy=(-1,0), xytext=(-2.5,-0.5),
             fontsize=15,
             arrowprops=dict(arrowstyle="->",
              connectionstyle="arc3" ) )
plt.ylim((-2,1))
plt.xlim((-3,0.5))
plt.grid()
plt.xlabel('Real')
plt.ylabel('Imag')
```

```
Text(0, 0.5, 'Imag')
```



Phase Margin & Gain Margin

- **Phase Margin:** The amount of additional phase lag at gain crossover frequency required to bring the system to the verge of instability. It is given by

$$\gamma = 180° + \phi \qquad \phi = \angle GH(j\omega_g)$$

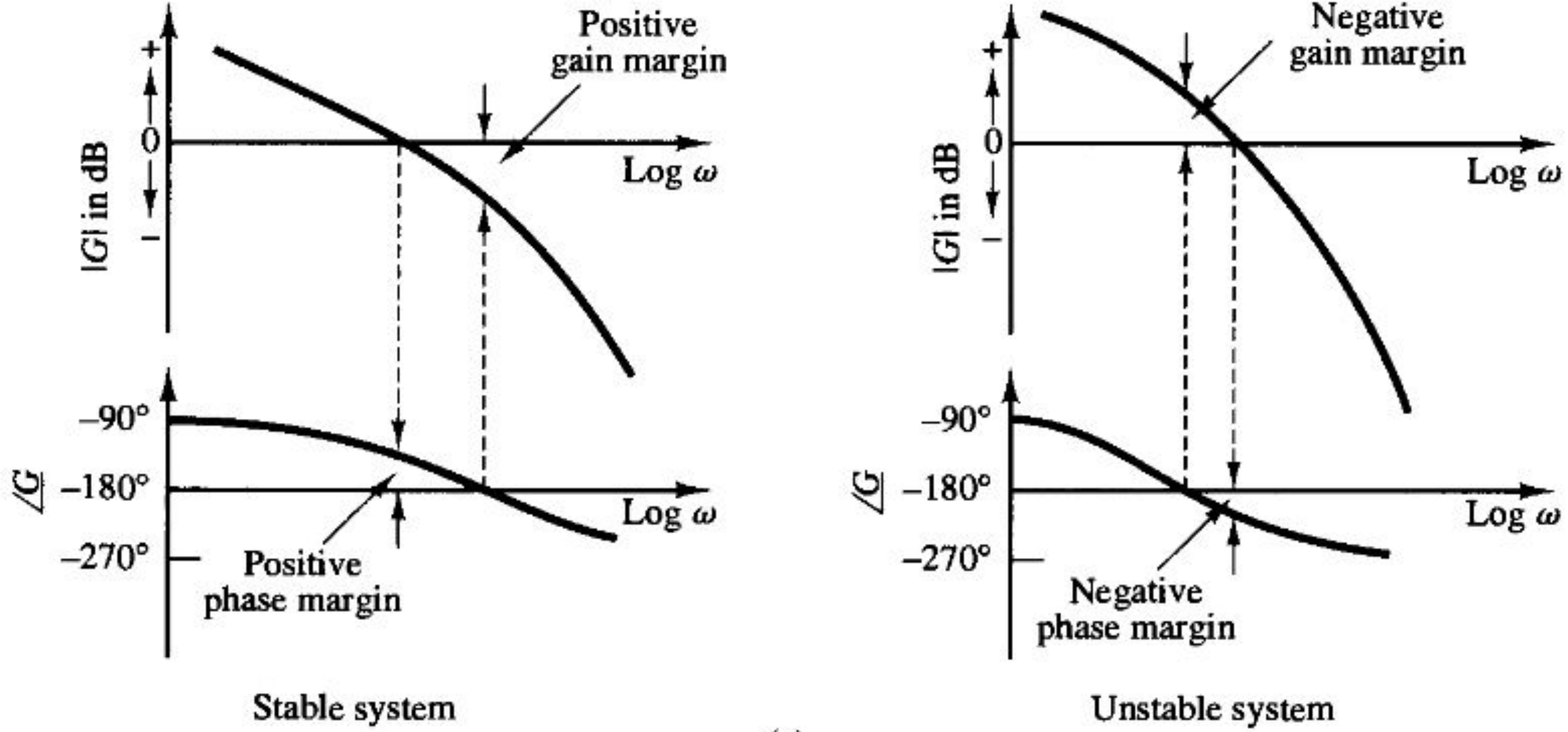
- The **gain crossover frequency** (w_g) is the frequency at which the magnitude of open-loop transfer function is unity, i.e., **|GH(jw)| = 1**.

- **Gain Margin** is the reciprocal of |G(jw)| at the frequency at which the phase angle is -180 degrees. This frequency is known as the **phase cross over frequency** (w_p).

$$K_g = \frac{1}{|GH(j\omega_p)|}$$

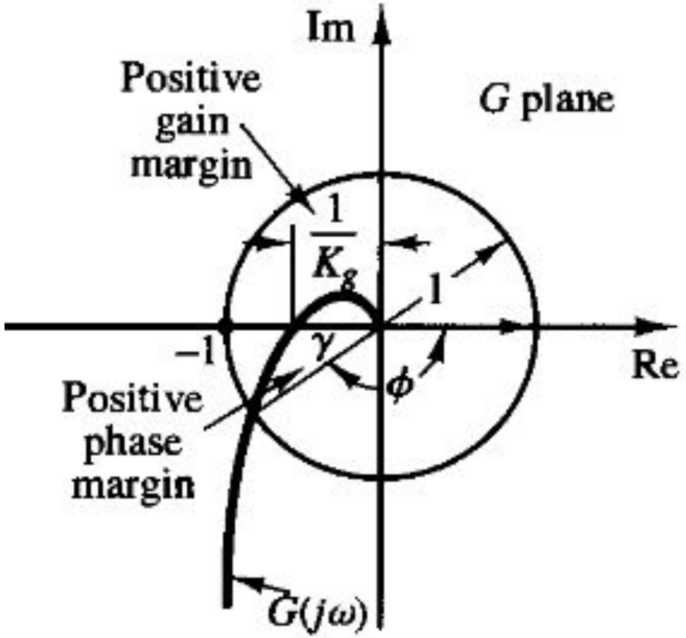
$$K_g \text{ dB} = 20 \log K_g = -20 \log |GH(j\omega_p)|$$

- For a stable minimum phase system, the phase margin is positive.
- For a stable minimum phase system, the gain margin indicates how much the gain can be increased before the system becomes unstable.
- For an unstable system, the gain margin indicates how much the gain must be decreased ro make the system stable.
- In logarithmic plots, the critical point in the complex plane correspond to 0 dB and -180 degree lines.
- For a stable non-minimum phase system, the phase margin and gain margin will be negative.
- PM and GM should be used together to determine the relative stability of a system.
- For a minimum-phase system to be stable, both GM and PM should be positive.

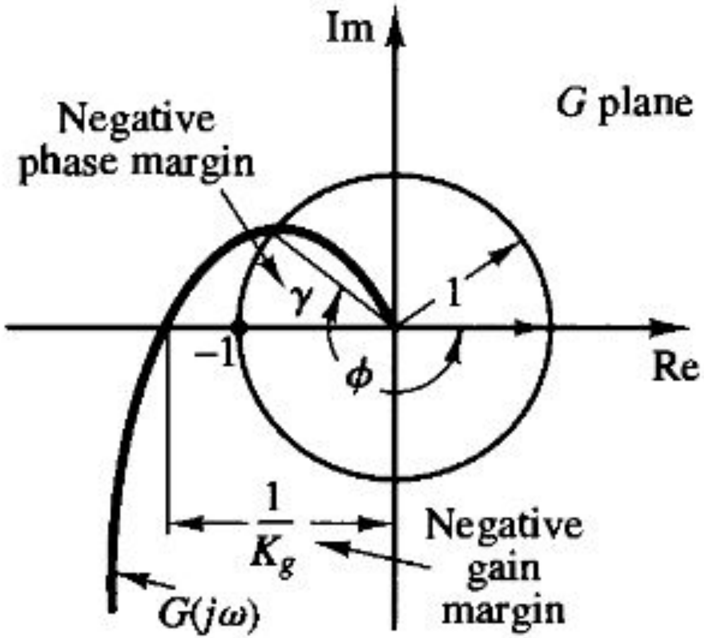


Positive
gain margin
|G| in dB
+
0
−
Log ω
−90°
∠G
−180°
−270°
Log ω
Positive
phase margin
Stable system
Negative
gain margin
|G| in dB
+
0
−
Log ω
−90°
∠G
−180°
−270°
Log ω
Negative
phase margin
Unstable system

(a)



Stable system

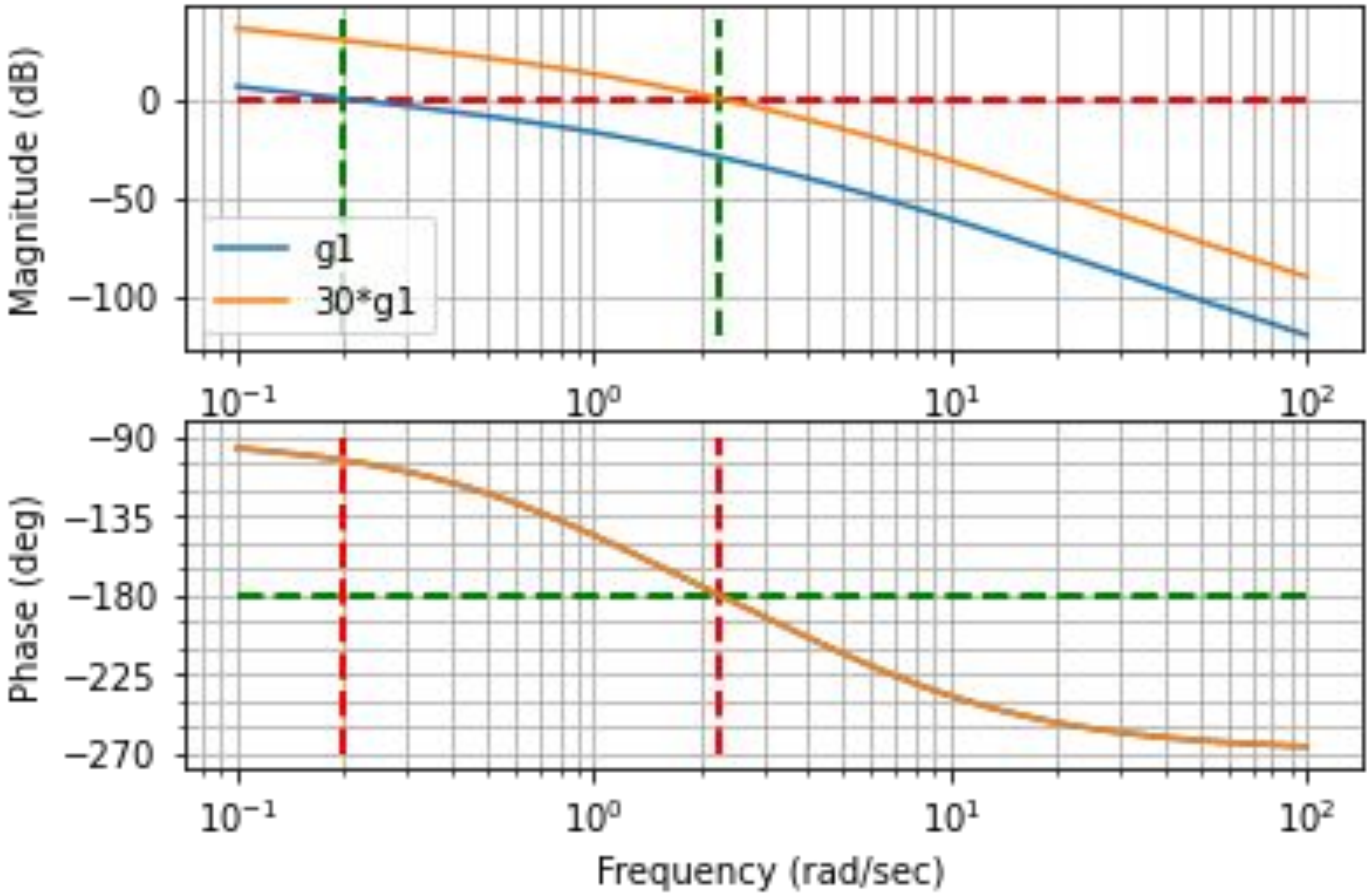


Unstable system

(b)

Example: Find stability margins for the system $G(s) = \dfrac{K}{s(s+1)(s+5)}$

```python
from control import *
import matplotlib.pyplot as plt
g1 = tf([1], [1, 6, 5, 0])
g2 = tf([30], [1,6,5,0])
m,p,w = bode([g1,g2], dB=True)
#gm,pm,sm,wg,wp,ws = stability_margins(g)
gm, pm, wp, wg = margin(g)
print('GM (dB): {:.2f}\nPM (Deg): {:.2f}\nWg (rad/s): {:.2f}\nWp (rad/s): {:.2f}'\
      .format(gm, pm, wg, wp))
fig = plt.gcf()
fig.axes[0].hlines(0,0.1,100, colors='r', linestyle='dashed', linewidth=2)
fig.axes[0].vlines(wg, -120,40, color='g', linestyle='dashed', linewidth=2)
fig.axes[0].vlines(wp, -120,40, color='g', linestyle='dashed', linewidth=2)

fig.axes[1].hlines(-180,0.1,100, colors='g', linestyle='dashed', linewidth=2)
fig.axes[1].vlines(wg, -270,-90, color='r', linestyle='dashed', linewidth=2)
fig.axes[1].vlines(wp, -270,-90, color='r', linestyle='dashed', linewidth=2)
fig.axes[0].legend(['g1', '30*g1'], loc='lower left')
```



GM (dB): 30.00
PM (Deg): 76.66
Wg (rad/s): 2.24
Wp (rad/s): 0.20

Control System Design by Frequency Response

- Lead Compensation
- Lag Compensation

Lead Compensator

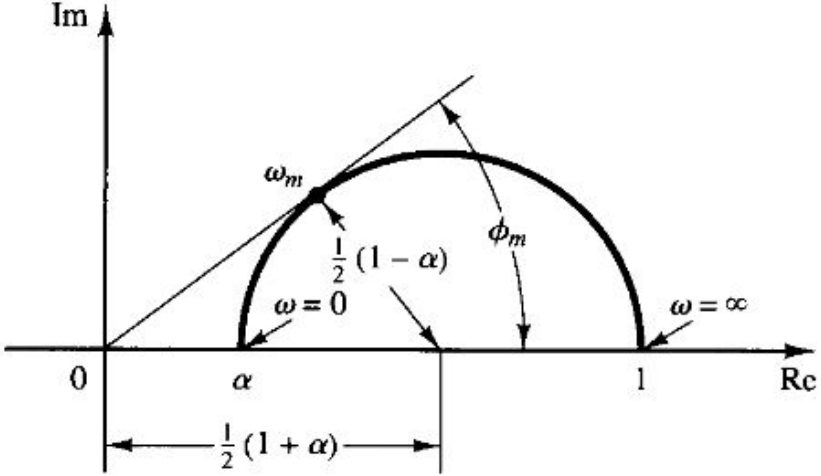
- Lead compensator contributes a positive phase to the open-loop system:

$$G_c(s) = K_c \alpha \frac{(sT+1)}{(\alpha T s+1)};\ 0 < \alpha < 1$$

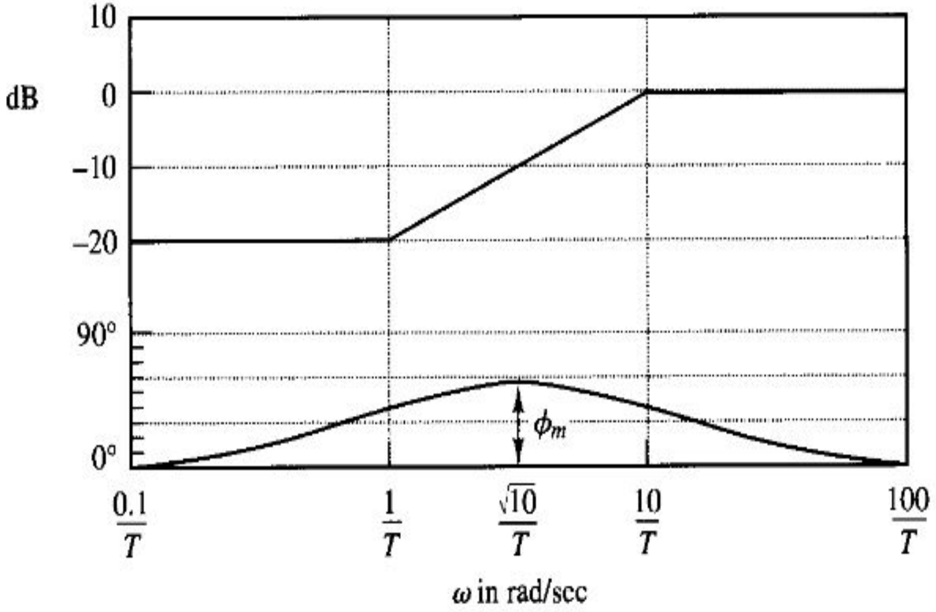
- The maximum phase lead contributed by this compensator is given by

$$\sin \phi_m = \frac{1-\alpha}{1+\alpha};\ \text{at}\ \omega = \omega_m = \frac{1}{T\sqrt{\alpha}}$$

- Maximum phase lead occurs at the geometric mean of two corner frequencies.



Lead Compensator acts as a high-pass filter.



Lead Compensation Example:

$$G(s) = \frac{4}{s(s+2)}$$

Design a lead compensator so that $K_v = 20$, phase margin is at least 50° and gain margin is at least 10 dB.

Lead compensator is given by

$$G_c(s) = \frac{K_c\alpha(Ts+1)}{(\alpha Ts+1)} = \frac{K(Ts+1)}{(\alpha Ts+1)}$$

- Step 1: Compute gain K from steady-state error requirement:

$$K_v = \lim_{s\to 0} sG_c(s)G(s) = 20$$

  This gives $K = 10$.

- Step 2: Draw the bode plot for system $KG(s)$. Find PM
- Step 3: The required lead for meeting PM requirement is around $50 - PM = 33 + 5 = 38^\circ$.
- Step 4: Compute α using the following formula:

$$\sin\phi_m = \frac{1-\alpha}{1+\alpha}$$

  This gives $\alpha = 0.24$

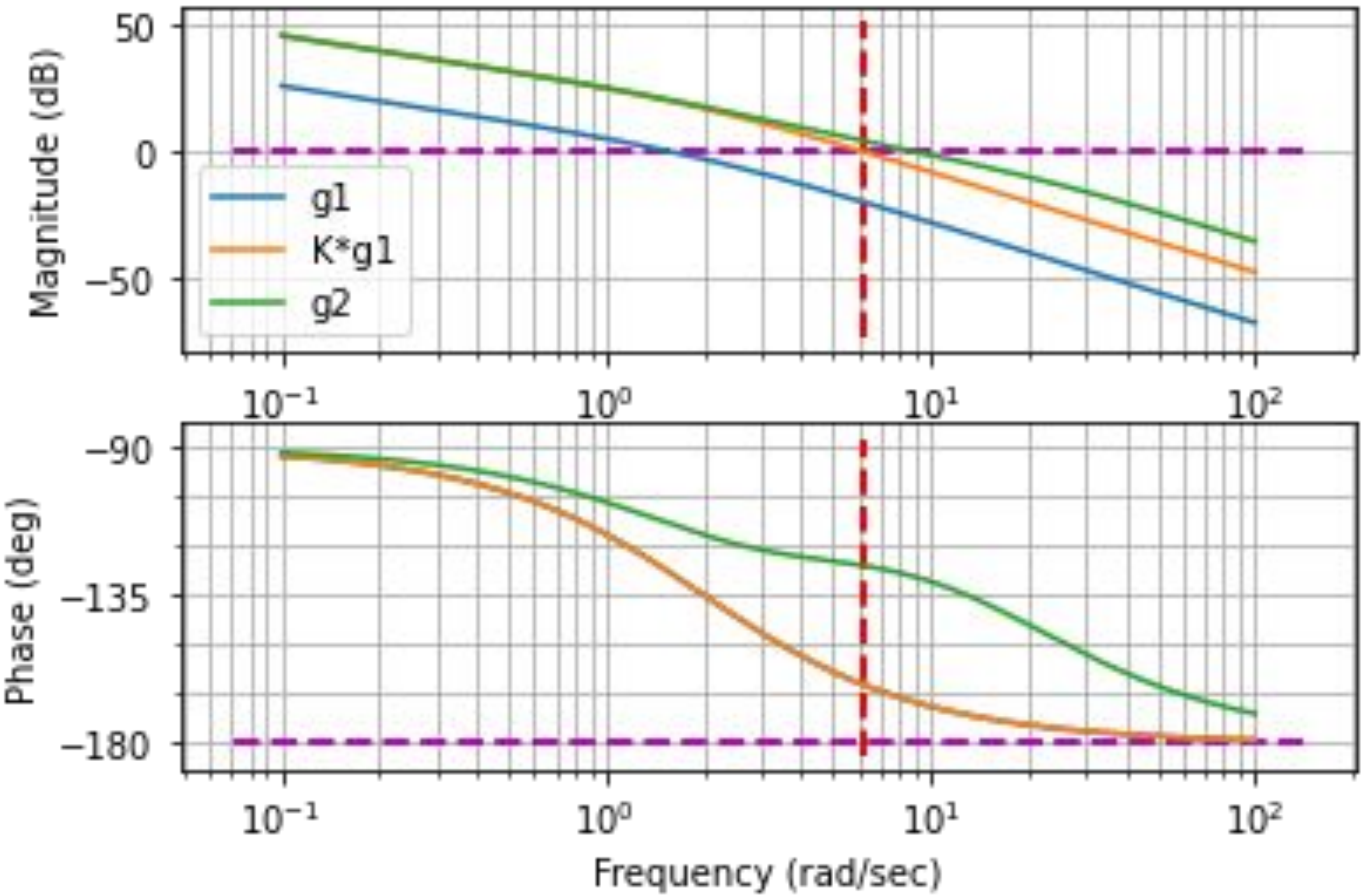
- Step 5: Maximum phase ϕ_m occurs at the geometric mean of two corner frequency which gives $\omega_m = \frac{1}{T\sqrt{\alpha}}$.

  So, choose $\omega_m = 9$ rad/s. This gives $T = 0.227$. Hence, the final compensator is given by

$$G_c(s) = \frac{41.7(s+4.41)}{(s+18.5)}$$

```python
num1 = [4]
den1 = [1, 2, 0]
# Find K to meet steady-state requirement (Kv)
K = 10
g1 = tf(num1, den1)
g11 = tf(K*np.asarray(num1), den1)
#Controller
c1 = tf(41.7*np.asarray([1, 4.41]), [1, 18.5])
# compensated Open-loop system: Gc(s)G(s)
g2 = series(c1,g1)
w = np.logspace(-1, 2, 1000)
# Bode Plot
m1,p1,w1 = bode([g1,g11, g2], w, dB=True)
#stability margins
gm, pm, wp, wg = margin(g11)
gm2, pm2, wp2, wg2 = margin(g2)

print('Stability Margins for uncompensated system (K*g1):')
print('GM (dB): {:.2f}\nPM (Deg): {:.2f}\nWg (rad/s): {:.2f}\nWp (rad/s): {:.2f}'\
      .format(gm, pm, wg, wp))
print('Stability Margins for compensated system (g2):')
print('GM (dB): {:.2f}\nPM (Deg): {:.2f}\nWg (rad/s): {:.2f}\nWp (rad/s): {:.2f}'\
      .format(gm2, pm2, wg2, wp2))

fig = plt.gcf()
xmin, xmax = plt.xlim()
ymin1, ymax1 = fig.axes[0].get_ylim()
ymin2, ymax2 = fig.axes[1].get_ylim()
fig.axes[0].hlines(0, xmin, xmax, colors='m', linestyle='dashed', linewidth=2)
fig.axes[0].vlines(wg, ymin1, ymax1, color='r', linestyle='dashed', linewidth=2)
fig.axes[0].vlines(wp, ymin1, ymax1, color='r', linestyle='dashed', linewidth=2)
fig.axes[1].hlines(-180, xmin, xmax, colors='m', linestyle='dashed', linewidth=2)
fig.axes[1].vlines(wg, ymin2, ymax2, color='r', linestyle='dashed', linewidth=2)
fig.axes[1].vlines(wp, ymin2, ymax2, color='r', linestyle='dashed', linewidth=2)
fig.axes[0].legend(['g1','g11','g2'], loc='lower left')
```

```
Stability Margins for uncompensated system (K*g1):
GM (dB): inf
PM (Deg): 17.96
Wg (rad/s): 6.17
Wp (rad/s): nan
Stability Margins for compensated system (g2):
GM (dB): inf
PM (Deg): 50.67
Wg (rad/s): 8.86
Wp (rad/s): nan
```



50
0
−50
Magnitude (dB)
g1
K*g1
g2
10^{-1}
10^{0}
10^{1}
10^{2}
−90
−135
−180
Phase (deg)
Frequency (rad/sec)

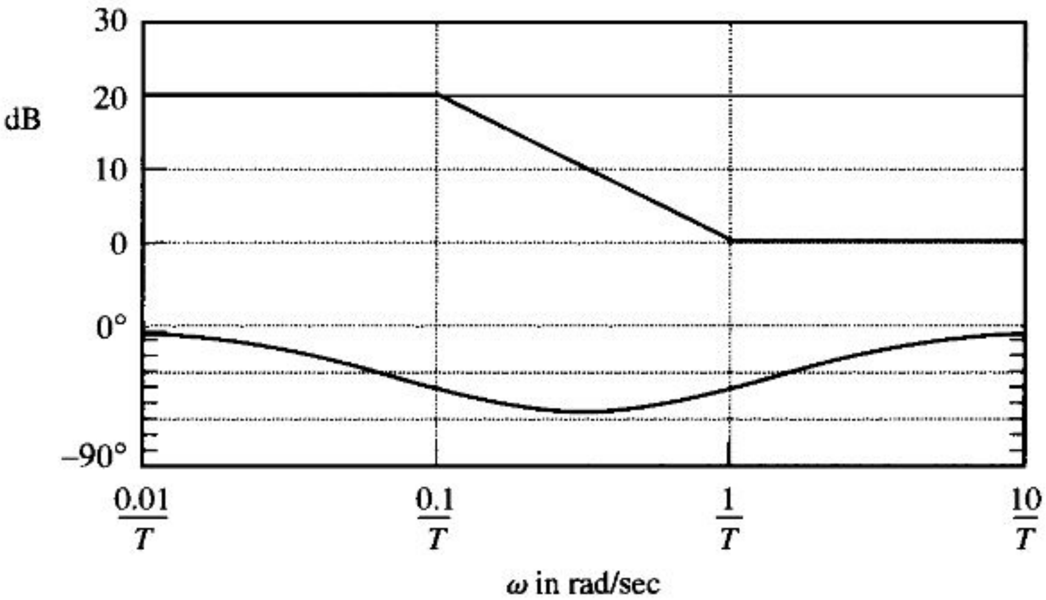
Lag Compensator

$$G_c(s) = K_c\beta \frac{Ts+1}{\beta Ts+1} = K_c \frac{s+\frac{1}{T}}{s+\frac{1}{\beta T}} \qquad (\beta > 1)$$

Lag compensator acts as a low pass filter that attenuates the signal at higher frequencies.

Max phase lag is obtained between the two corner frequencies:

$$\omega_1 = \frac{1}{T}, \omega_2 = \frac{1}{\beta T}$$



Consider the open-loop system: $G(s) = \frac{1}{s(s+1)(0.5s+1)}$. Design a lag compensator so that the static velocity error constant is 5, the PM is at least 40° and GM is at least 10 dB.

Solution: The compensator is given by $G_c(s) = K_c\beta\frac{Ts+1}{\beta Ts+1} = K\frac{Ts+1}{\beta Ts+1}$, where $K = \beta K_c$. Need to find K_c, β and T.

1. Compute K from steady-state error condition: $K_v = \lim_{s\to 0} sG_c(s)G(s) = 5$. This gives $K = 5$.
2. Draw the bode plot of system: $G_1 = KG(s)$. PM is -13°, so the system is unstable.
3. Find the frequency where PM of $KG = 40^\circ$. This is around $\omega = 0.7rad/s$. Choose the zero of compensator at least 1 octave to 1 decade below this frequency. Let's choose zero at $w_z = \frac{1}{T} = 0.1$ rad/s. This gives $T = 10$.
4. Since this corner frequency is close to the new gain cross_over frequency, add a correction of about 12° to the required PM which is now about $40 + 12 = 52^\circ$. This occurs at $\omega_1 = 0.5$ rad/s. The gain at this frequency is around 20 dB.
5. For ω_1 to be the new gain crossover frequency, the compensator should provide an additional attenuation of -20 dB to bring the composite gain to 0 dB after compensation at this frequency. so $20\log\frac{1}{\beta} = -20$. This gives $\beta = 10$.

$$
\begin{aligned}
20\log|G_c(j\omega_1)|.|KG(j\omega_1)| &= 0 \text{ dB}\\
20\log|G_c(j\omega_1)| + 20 \text{ dB} &= 0 \text{ dB}\\
20\log\frac{\sqrt{\omega^2T^2+1}}{\sqrt{\beta^2\omega^2T^2+1}} &= -20 \text{ dB}\\
20\log\frac{1}{\beta} &= -20 \text{ dB } (\because \omega T >> 1)
\end{aligned}
$$

Lag Compensator Design Example

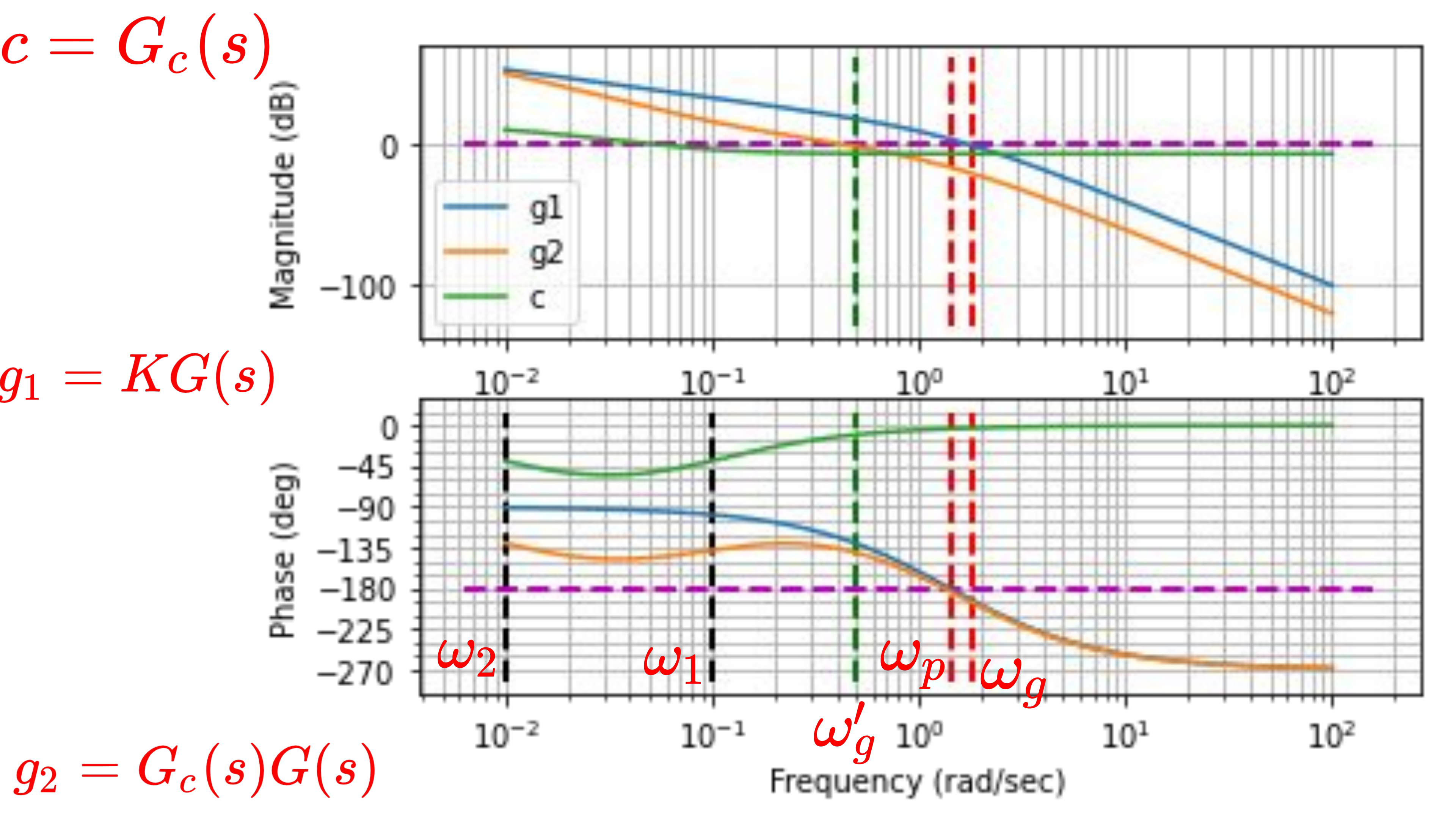
6. The pole of compensator is at $\omega_p = \frac{1}{\beta T} = 0.01$ rad/s.
7. Finally, $K_c = \frac{K}{\beta} = \frac{5}{10} = 0.5$

So our compensator is $G_c(s) = \frac{5(10s+1)}{(100s+1)}$.

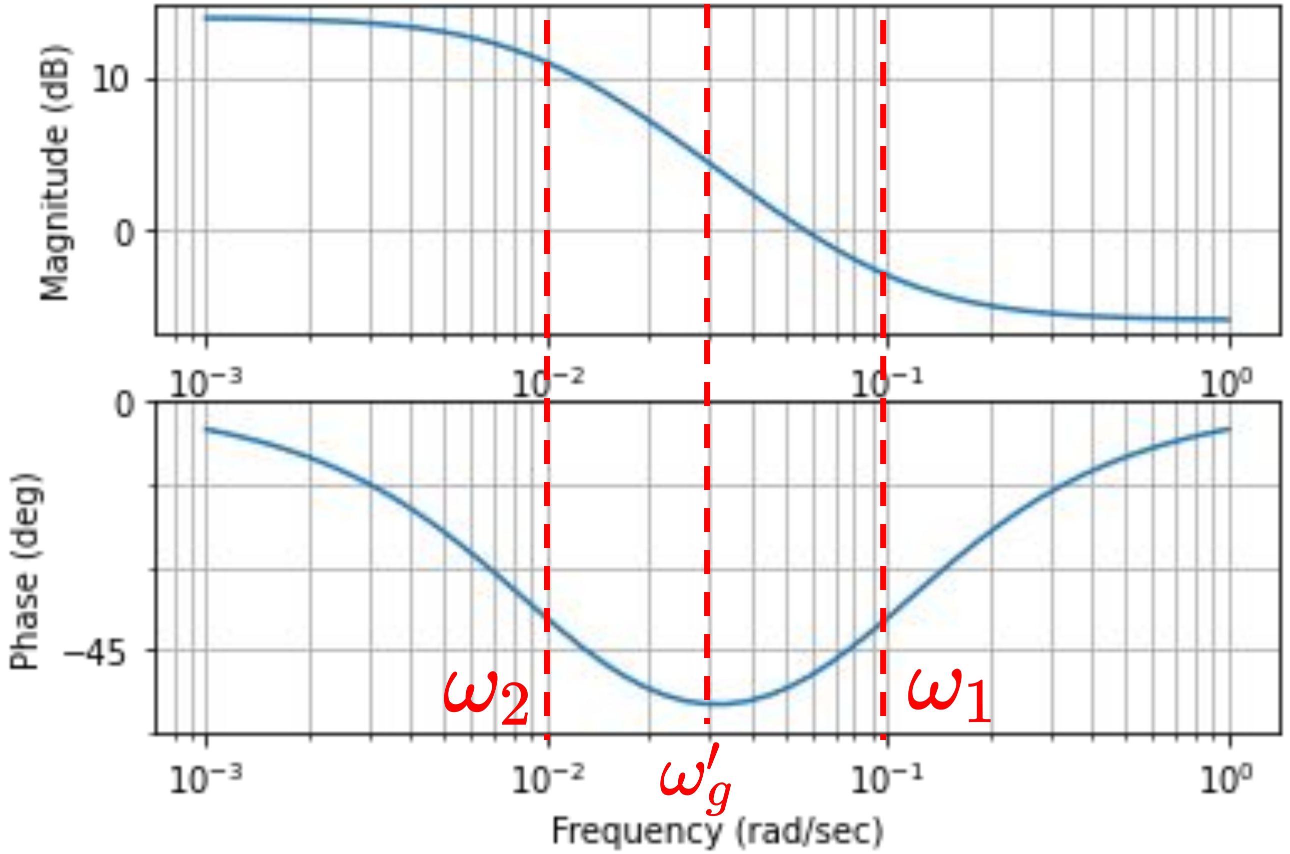
```python
num1 = [1]
den1 = [0.5, 1.5, 1, 0]

K = 5  # obtained from steady-state error condition
g = tf(num1, den1)
g1 = tf(K*np.asarray(num1), den1)

# Compensator
num2 = [10,1]
den2 = [100, 1]
c = tf(K*np.asarray(num2), den2)

# Compensated System
g2 = series(c, g)

w = np.logspace(-2,2,1000)
m1, p1, w1 = bode([g1,g2,c], w, dB=True)

gm, pm, wp, wg = margin(g1)
gm2, pm2, wp2, wg2 = margin(g2)
```

```
Stability Margins for uncompensated system (g1):
GM (dB): 0.60
PM (Deg): -13.00
Wg (rad/s): 1.80
Wp (rad/s): 1.41
Stability Margins for compensated system (g2):
GM (dB): 5.19
PM (Deg): 41.61
Wg (rad/s): 0.45
Wp (rad/s): 1.32
```

```python
fig = plt.gcf()
xmin, xmax = plt.xlim()
ymin1, ymax1 = fig.axes[0].get_ylim()
ymin2, ymax2 = fig.axes[1].get_ylim()
fig.axes[0].hlines(0, xmin, xmax, colors='m', linestyle='dashed', linewidth=2)
fig.axes[0].vlines(wg, ymin1, ymax1, color='r', linestyle='dashed', linewidth=2)
fig.axes[0].vlines(wp, ymin1, ymax1, color='r', linestyle='dashed', linewidth=2)

fig.axes[1].hlines(-180, xmin, xmax, colors='m', linestyle='dashed', linewidth=2)
fig.axes[1].vlines(wg, ymin2, ymax2, color='r', linestyle='dashed', linewidth=2)
fig.axes[1].vlines(wp, ymin2, ymax2, color='r', linestyle='dashed', linewidth=2)
fig.axes[0].legend(['g1', 'g2', 'c'], loc='lower left')

wg_new = 0.5 # new gain crossover freq
w1 = 0.1  # corner frequency due to zero
w2 = 0.01 # corner freqency due to pole
fig.axes[0].vlines(wg_new, ymin1, ymax1, color='g', linestyle='dashed', linewidth=2)
fig.axes[1].vlines(wg_new, ymin2, ymax2, color='g', linestyle='dashed', linewidth=2)
fig.axes[1].vlines(w1, ymin2, ymax2, color='k', linestyle='dashed', linewidth=2)
fig.axes[1].vlines(w2, ymin2, ymax2, color='k', linestyle='dashed', linewidth=2)
```

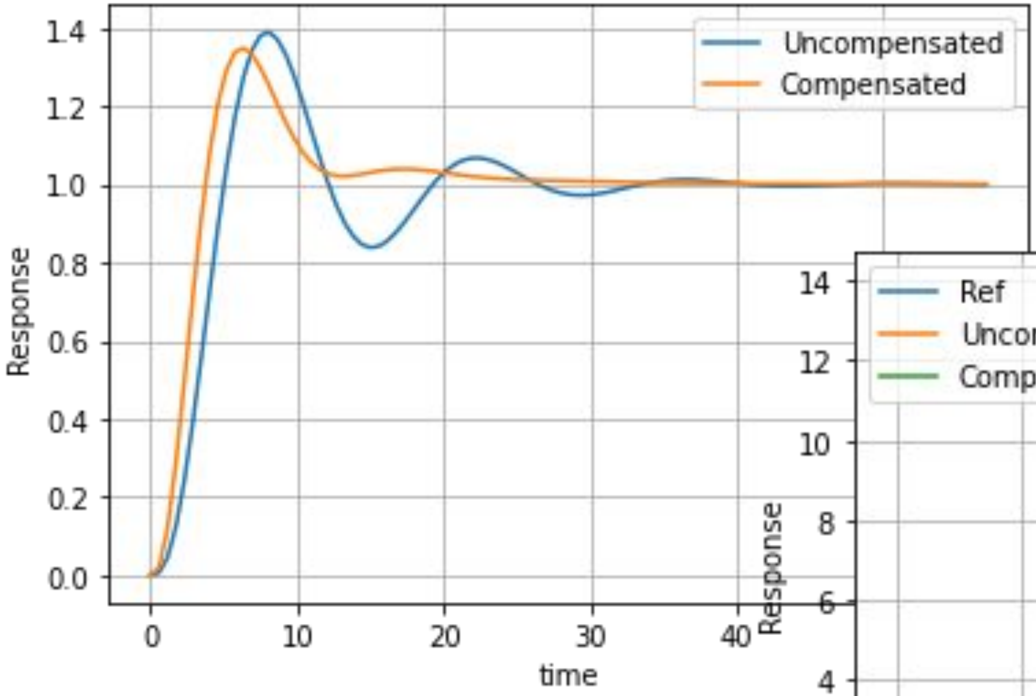


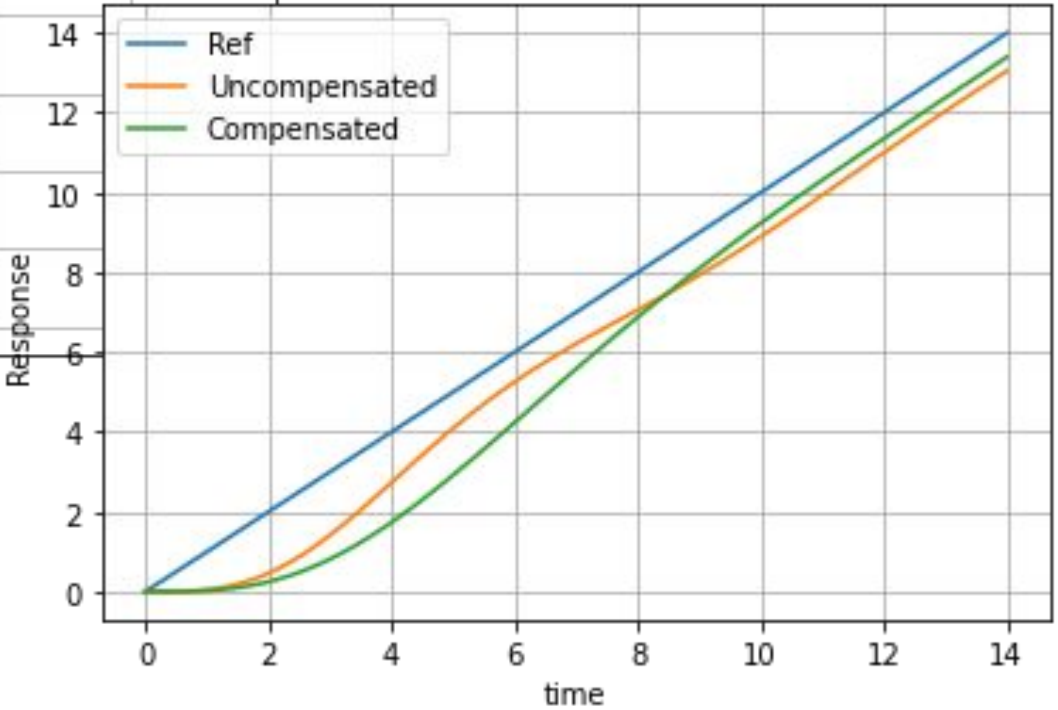
$c = G_c(s)$
$g_1 = KG(s)$
$g_2 = G_c(s)G(s)$
Magnitude (dB)
0
−100
g1
g2
c
10^{-2}
10^{-1}
10^{0}
10^{1}
10^{2}
Phase (deg)
0
−45
−90
−135
−180
−225
−270
ω_2
ω_1
ω_p
ω_g
ω_g'
Frequency (rad/sec)



Bode Plot of the lag compensator:

$$G_c(s) = \frac{5(10s+1)}{(100s+1)}$$


Uncompensated
Compensated
Response
time
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
0
10
20
30
40


Ref
Uncompensated
Compensated
Response
time
14
12
10
8
6
4
2
0
0
2
4
6
8
10
12
14

Summary

- Frequency response analysis concerns itself with the steady-state behaviour of a system subjected to sinusoidal inputs.

- Various graphical techniques are used for analyzing frequency response behaviour - Bode Plot, Polar Plots and Log-magnitude-vs-phase plots.

- Polar plots / Nyquist plot can be used to analyze both the absolute stability as well as relative stability of closed loop systems.

- Gain margins and phase margins are two popular measures for evaluating relative stability of control systems.

- We demonstrated few examples of control design based on frequency response methods.