# Chapter 7

# Servo Controller Design

Servo controllers are a type of feedback controllers where the reference input is a differentiable time-varying function, and the response follows the reference input quickly and accurately. Servo controllers are used in mechatronic and robotic applications because of its fast tracking performance. In order to achieve servo control capability, the control system should be designed in frequency domain where reference input is represented by its frequency components (Fourier transform) and the control system as a low pass filter. The response is contributed by the low pass filtered frequency components of the reference input. If the control system has a sufficient pass band so that most of the reference frequency components pass through without attenuation, then the response will be a close representation of the reference input. Feedback controller design in frequency domain was pioneered by Bode [8].

## 7.1 Frequency Response

Frequency response of a plant is illustrated in Figure 7.1. The plant is represented by a series of blocks $G(s) = G_1(s)G_2(s)...G_n(s)$, and the reference input is selected as a single frequency signal $r(t) = A \sin \omega t$. As this signal passes through the plant, each block changes its magnitude and phase. For example, block $G_1(s)$ changes the amplitude of the signal by introducing a gain $M_1(j\omega) = |G_1(j\omega)|$ and also adds phase $\phi_1(j\omega) = \angle G_1(j\omega)$. After passing through all of the blocks the response appears as $y(t) = AM \sin(\omega t + \phi)$, where $M(j\omega) = |G(j\omega)|$ is the overall gain, and $\phi(j\omega) = \angle G(j\omega)$is the overall phase addition. Both gain and phase are frequency dependent. Gains of successive blocks multiply whereas phase of successive bocks add together. Therefore, the overall system gain is given by

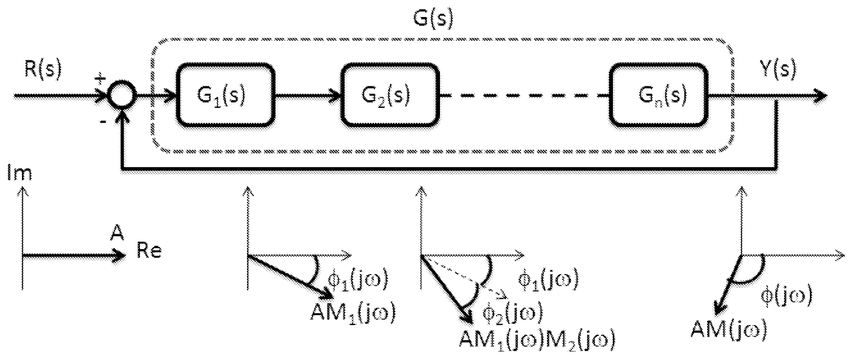



Figure 7.1: Magnitude and phase change as frequency $\omega$ passes through the system

$$\begin{aligned} M(j\omega) &= \Pi_1^n M_i(j\omega) \\ &= \Pi_1^n |G_i(j\omega)| \quad (7.1) \\ M(j\omega)\text{dB} &= \Sigma_1^n |G_i(j\omega)|\text{dB} \quad (7.2) \end{aligned}$$

where $|G_i(j\omega)|\text{dB} = 20\log|G_i(j\omega)|\text{dB}$. The net phase is given by

$$\begin{aligned} \phi(j\omega) &= \Sigma_1^n \phi_i(j\omega) \\ &= \Sigma_1^n \angle G_i(j\omega) \quad (7.3) \end{aligned}$$

### 7.1.1 Gain and Phase of Common Blocks

#### Zeros and Poles at Origin

For a pole at origin there is $s$ term in the denominator, and for a zero at origin there is a $s$ term in the numerator. The existence of an $n$ number of poles or zeros at origin is represented by $s^{\pm n}$, where $+n$ is for zeros and $-n$ is for poles. Therefore, the overall gain of $n$ number of zeros or poles is

$$\begin{aligned} |G_i(j\omega)| &= \omega^{\pm n} \\ |G_i(j\omega)|dB &= \pm n 20\log\omega\text{dB} \quad (7.4) \end{aligned}$$

And, the overall phase addition by an $n$ number of zeros or poles is

$$\begin{aligned} \angle G_i(j\omega) &= \pm n \tan^{-1}\left(\frac{\omega}{0}\right) \\ &= \pm 90 n^0 \end{aligned} \tag{7.5}$$

Following MatLab code draws the gain and phase plots (Bode plots) for a zero at origin as illustrated in Fig.7.2(a), and for a pole at origin as illustrated in Fig.7.2(b). For a zero at origin, the gain increases by +20dB, that is by 10 times when frequency increases by a factor of 10. For a pole at origin, gain reduces by the same proportion. This change of 20dB change per 10 times change in frequency is termed as 20dB/decade.

```
sys=tf([1 0],[1]); % for a zero at origin
%sys=tf([1],[1 0]); % for a pole at origin
bode(sys); grid on;
```

### First Order Block

First order block is a zero or a pole located at $-a$, and modeled by $G_i(s) = (s+a)^{\pm 1}$, where + sign represents a first order zero and $-1$ represents a first order pole. The gain of a first order pole or zero for frequency $\omega$ is given by

$$\begin{aligned} |G_i(j\omega)| &= |(j\omega + a)|^{\pm 1} \\ |G_i(j\omega)|\text{dB} &= \pm 20 \log(\sqrt{\omega^2 + a^2})\text{dB} \end{aligned} \tag{7.6}$$

And the phase addition for frequency $\omega$ is given by

$$\angle G_i(j\omega) = \pm \tan^{-1}\left(\frac{\omega}{a}\right) \tag{7.7}$$

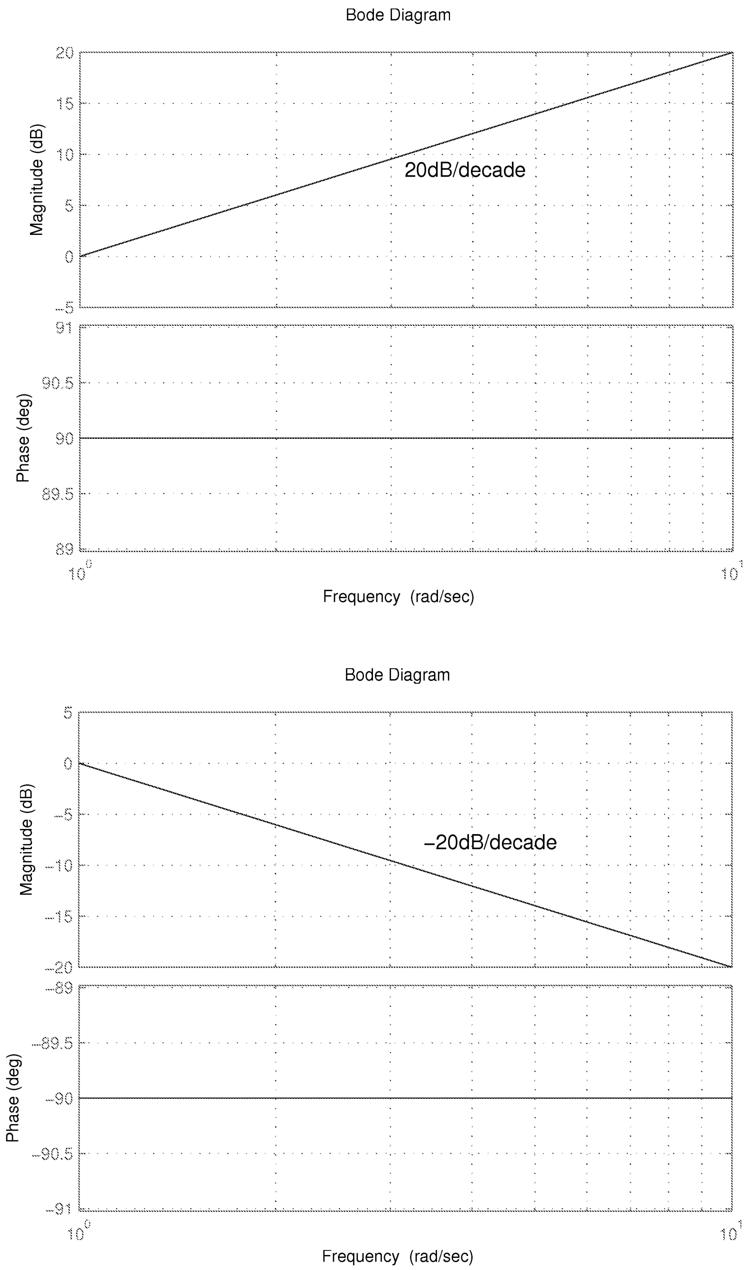
From (7.6) the gain at 0 and $a$, the location of the zero or pole can be determined as follows.

$$\begin{aligned} |G_i(j0)| &= \pm 20 \log \sqrt{0^2 + a^2} \\ &= \pm 20 \log a\text{dB} \end{aligned} \tag{7.8}$$

$$\begin{aligned} |G_i(ja)| &= \pm 20 \log \sqrt{z^2 + a^2} = \pm 20 \log \sqrt{(2a^2)}\text{dB} \\ &= \pm(20 \log a + 20 \log \sqrt{2})\text{dB} \\ &= \pm 20 \log a\text{dB} \pm 3\text{dB} \end{aligned} \tag{7.9}$$

From (7.8) and (7.9), the gain change when frequency changes from 0rad/s to $a$rad/s is 3dB. For a zero it is an amplification of signal, whereas for a pole it is an attenuation. Figure 7.3 shows gain and phase lag for a zero at $s = -10$ and the same for a pole at the same location. Following Matlab code draws gain and phase plots of a first order block.

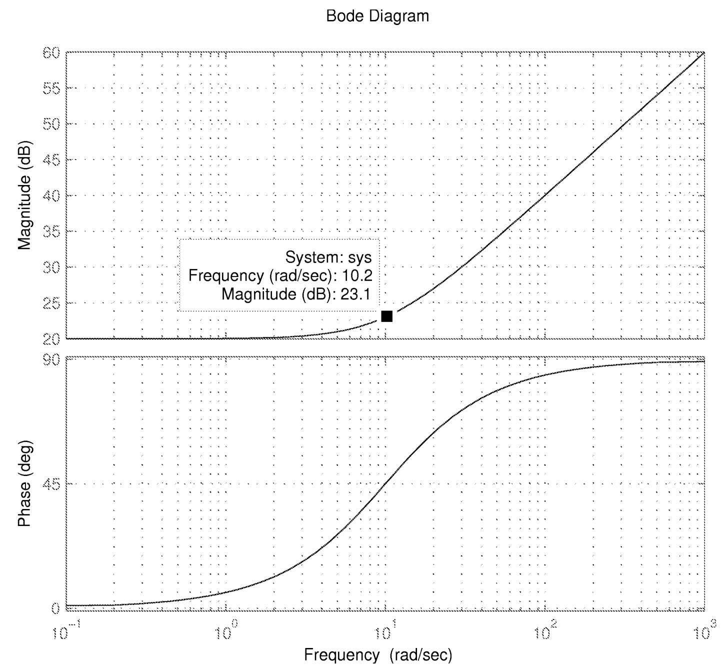
```
sys=tf([1 10],[1]); % for a zero at -10
%sys=tf([1],[1 10]); % for a pole at -10
bode(sys); grid on;
```



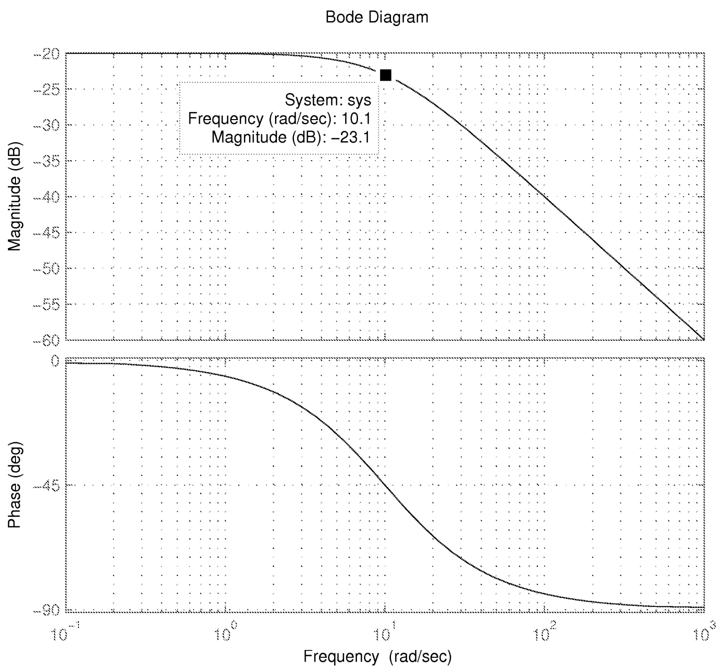

(a)

(b)

Figure 7.2: Gain and phase plots (a) for a zero at origin, and (b) for a pole at origin




(a)




(b)

Figure 7.3: Gain and phase plots of a first order block (a) for a zero $s = -10$, and (b) for a pole at $s = -10$

### Second Order Blocks

The generic second order block is represented by $G(s) = (s^2+2\zeta\omega_n s+\omega_n^2)^{\pm 1}$, where $+1$ represents a conjugate pair of zeros, and $-1$ represents a conjugate pair of poles. The gain of this block for a frequency $\omega$ is given by

$$\begin{aligned} |G_i(j\omega)| &= |s^2+2\zeta\omega_n s+\omega_n^2|^{\pm 1} \\ &= |(\omega_n^2-\omega^2)+j2\zeta\omega\omega_n|^{\pm 1} \\ &= \sqrt{(\omega_n^2-\omega^2)^2+(2\zeta\omega\omega_n)^2} \\ |G_i(j\omega)|\text{dB} &= \pm 20\log\sqrt{(\omega_n^2-\omega^2)^2+(2\zeta\omega\omega_n)^2}\text{dB} \end{aligned} \tag{7.10}$$

And, the phase addition is given by

$$\angle G_i(j\omega) = \pm\tan^{-1}\left(\frac{2\zeta\omega\omega_n}{\omega_n^2-\omega^2}\right) \tag{7.11}$$

From (7.10), the gain for 0 rad/s and that for $\omega_n$ rad/s can be calculated as follows.

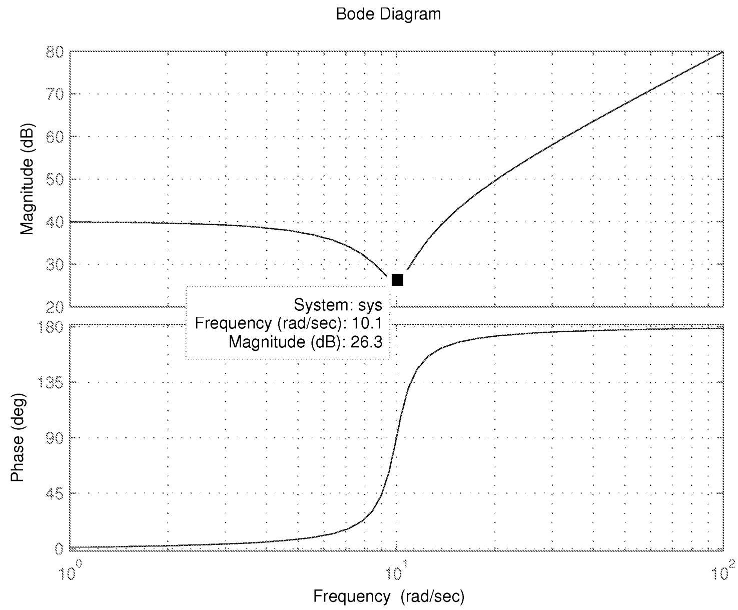
$$\begin{aligned} |G_i(j0)|\text{dB} &= \pm 20\log\sqrt{\omega_n^4+0^2}\text{dB} \\ &= \pm 40\log\omega_n\text{dB} \end{aligned} \tag{7.12}$$

$$\begin{aligned} |G_i(j\omega_n)|\text{dB} &= \pm 20\log\sqrt{0^2+(2\zeta\omega_n^2)^2}\text{dB} \\ &= \pm 20\log 2\zeta\omega_n^2\text{dB} \\ &= \pm(40\log\omega_n+20\log 2\zeta)\text{dB} \\ &= \pm 40\log\omega_n\text{dB}\pm 20\log 2\zeta\text{dB} \end{aligned} \tag{7.13}$$

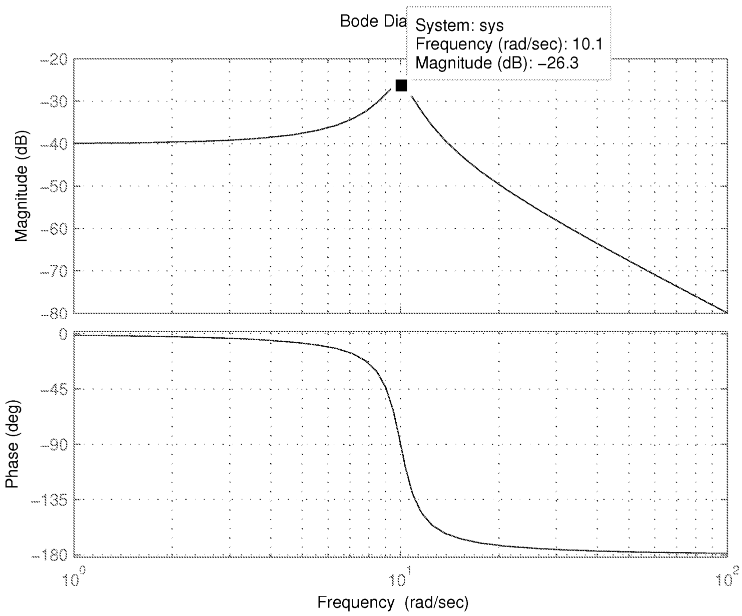
From (7.12) and (7.13), the gain change as frequency increases from 0rad/s to $\omega_n$rad/s is $20\log 2\zeta$dB. This gain is an amplification for a conjugate pair of poles, whereas it is an attenuation for a pair of conjugate poles. Following MatLab code draws the gain and phase plots for a conjugate pair of zeros or poles.

```
% Bode plots of a second order block
zeta=0.1;
wn=10;
% sys=tf([1 2*zeta*wn wn*wn],[1]); % for a pair of zeros
sys=tf([1],[1 2*zeta*wn*wn]); % for a pair of poles
bode(sys); grid on;
```

The frequency responses drawn by the above MatLab code are illustrated in Figure 7.4, which shows the gain and phase addition for the second order block $(s^2 + 2s + 100)^{\pm 1}$, where $\omega_n = 10$rad/s and $\zeta = 0.1$. The gain change when frequency increases to $\omega = 10$rad/s is $20 \log 2 \times 0.1 \approx 14$dB.




(a)




(b)

Figure 7.4: Gain and phase plots (a) for a conjugate pair of zeros $(s^2 + 2s + 100)$, and (b) for conjugate pair of poles $(s^2 + 2s + 100)^{-1}$

## 7.2 Phase Margin and Gain Margin

The total phase change that any frequency undergoes within the forward path of the plant should not be 180. The phase change of 180 is the same as inverting the sinusoidal signal, and when this inverted signal is negatively fed back, it turns out to a positive feedback. However, if the gain at 180 is less than 1 the signal will die out in magnitude in successive feedback. Therefore, system will maintain stability with respect to that frequency. The additional gain that frequency can accept before it becomes unstable is called as gain margin.

When the overall gain of any frequency is unity, the phase change should not be 180 to maintain stability. The additional phase away from 180 of the frequency, which has unity gain is called the phase margin. Figure 7.5 illustrates the gain margin and phase margin of a plant. The plant has unity gain (0dB) at 0.414 rad/s, and at that frequency the total phase change is 28.2 away from 180 unstable limit. Similarly, the plant shows 180 phase change at 0.671rad/s, and at this frequency the gain is lower than unity so that it can accept an additional gain up to 8.53 before stability is affected.

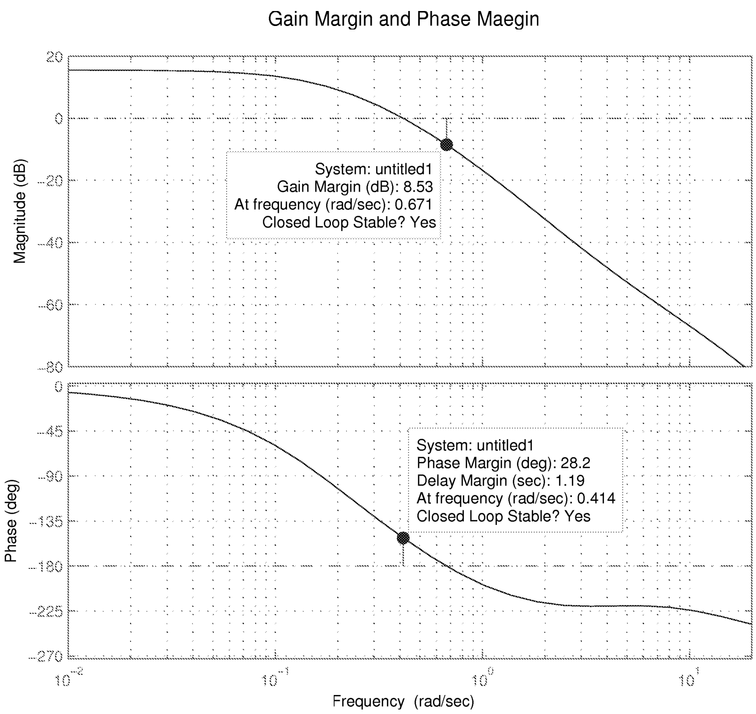



Figure 7.5: Gain margin and phase margin

## 7.3 Example: Servo Controller Design

A plant is modeled by the transfer function

$$G(s) = \frac{3(s+3)}{(s+4)(s^2+3s+20)} \tag{7.14}$$

1. Determine frequency response of the plant in terms of gain and phase plots
2. Design a compensator to increase bandwidth to 20rad/s and overall phase margin to $45^o$
3. Design a compensator to maintain the unit step steady state error within 0.01

**Design Procedure**
The following MatLab code will draw gain and phase plots as shown in Fig.7.6. The code also calculates the gain and phase at 20[rad/s].

```
numG=3*[1 3];
denG=conv([1 4],[1 3 20]);
G=tf(numG,denG);
bode(G); grid on;
[gainG,phaseG]=bode(G,20) % gain and phase at 20[rad/s]
```

**Bandwidth Adjustment**
From the gain response of Fig. 7.6, the gain at 20 rad/s is -42.2dB. The gain response can be lifted up by introducing a forward gain $K$ so that the gain at 20 rad/s become 0dB. Therefore, the value for $K$ can be calculated as follows.

$$\begin{aligned} K|G(j\omega)|_{\omega=20} &= 1 \\ K\left|\frac{3(j20+3)}{(j20+4)(-20^2+3\times j20+20)}\right| &= 1 \\ K\times 0.0077 &= 1 \\ K &= 129.3 \end{aligned}$$

The value of $K$ can also be calculated using the gain response in Fig.7.6, where -42.2dB is the gain for 20[rad/s]. In order to achieve 0dB at this frequency $20\log K - 42.2dB =$ 0dB. Therefore, $K = 10^{42.2/20} \approx 129$. With

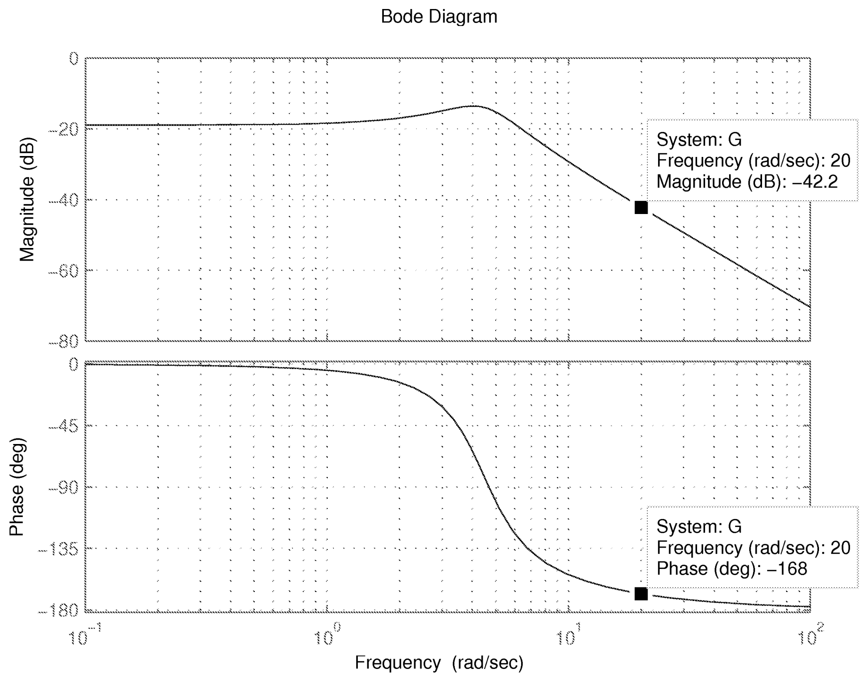



Figure 7.6: Gain and phase plots of $G(j\omega)$

$K = 129.3$. Introduction of $K$ does not have any phase contribution as it is a real quantity. The gain and phase responses after introduction of $K$ are shown in Fig.7.7. These responses are drawn by developing that MatLab code as follows.

```
numG=3*[1 3];
denG=conv([1 4],[1 3 20]);
G=tf(numG,denG);
bode(G); grid on;
[gainG,phaseG]=bode(G,20) % gain and phase at 20[rad/s]
% Bandwidth Adjustment
K=1/gainG
G1=K*G
bode(G1); grid on; hold on;
```

According to Fig.7.7, after introduction of $K = 129$, the gain response has been shifted up while phase response remains unchanged.

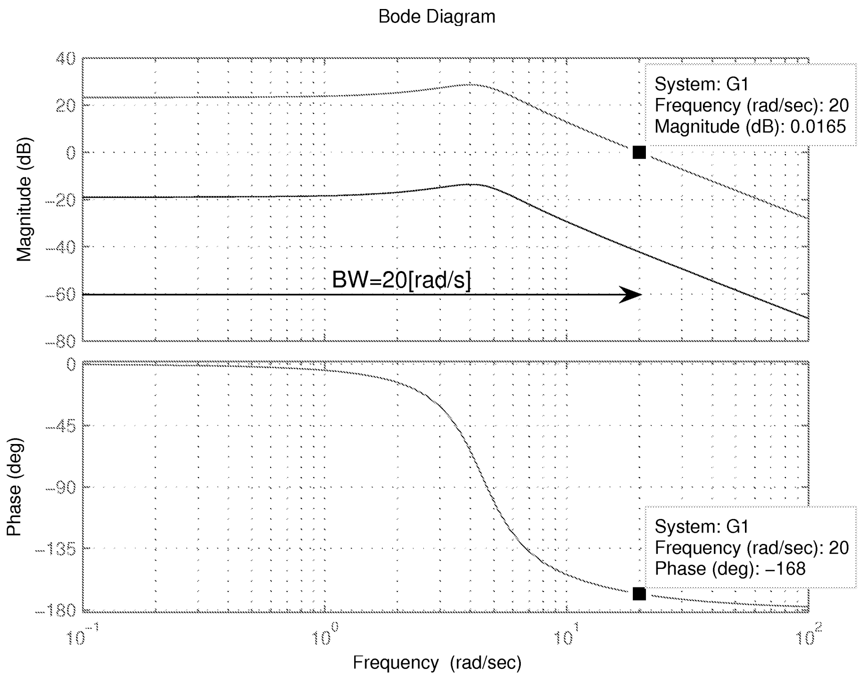



Figure 7.7: Gain and phase response of bandwidth adjusted system $KG(j\omega)$

**Phase Margin Adjustment**
The phase addition at 20[rad/s] can be calculated as follows

$$\begin{aligned}\angle G(j\omega)_{\omega=20} &= \angle(j20+3) - \angle(j20+4) - \angle(20^2 + 3\times j20 + 20)\\ &= \angle(j20+3) - \angle(j20+4) - \angle(j60 - 380)\\ &= \tan^{-1}\left(\frac{20}{3}\right) - \tan^{-1}\left(\frac{20}{4}\right) - \left[180^0 - \tan^{-1}\left(\frac{60}{380}\right)\right]\\ &= -168^0\end{aligned}$$

Phase addition can directly be read from the phase response shown in Fig.7.7, where it is indicated as -$168^0$ at 20[rad/s]. Therefore, the phase margin is only $12^0$,which is not adequate. A lead compensator is introduced in order to add phase to the plant so that phase margin can be improved to $45^0$. The required phase contribution $\phi_{le} = 33^0$ of the lead compensator at 20 rad/s is calculated as follows.

$$\angle G(j20) + \phi_{le} - PM = -180^0$$

$$\begin{aligned}\phi_{le} &= -180 + PM - (\angle G(j20)) \\ &= -180^0 + 45^0 - (-168^0) \\ &= 33^0\end{aligned}$$

The lead compensator $\frac{(s+z_{le})}{(s+p_{le})}$; $p_{le} > z_{le}$ is shown in Fig.7.8.

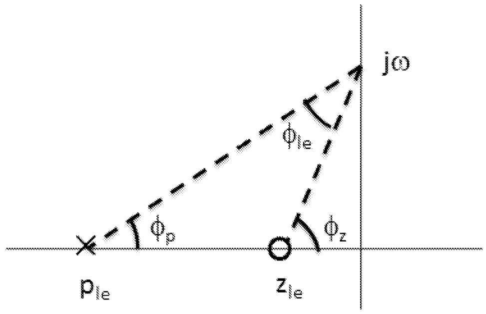



Figure 7.8: Phase lead of the lead compensator for frequency $\omega$

The phase contribution $\phi_{le}(\omega)$ of the lead compensator is

$$\begin{aligned}\phi_{le}(\omega) &= \phi_z - \phi_p \\ \phi_{le}(\omega) &= \tan^{-1}\left(\frac{\omega}{z_{le}}\right) - \tan^{-1}\left(\frac{\omega}{p_{le}}\right) \end{aligned} \tag{7.15}$$

Referring to Fig.7.8 and (7.15), the phase contribution of the lead compensator diminishes when $\omega \to \{0, \infty\}$ as shown below.

$$\begin{aligned}\lim_{\omega\to 0}\phi_{le}(\omega) &= 0^0 - 0^0 = 0 \\ \lim_{\omega\to\infty}\phi_{le}(\omega) &= 90^0 - 90^0 = 0 \end{aligned} \tag{7.16}$$

In between these two limits phase contribution increases to a maximum and then decreases. The locations of pole and zero of the lead compensator decides the frequency $\omega_{\phi_{max}}$ as shown in (7.17) at which the phase contribution is maximum.

$$\omega_{\phi_{max}} = \sqrt{z_{le}p_{le}} \tag{7.17}$$

From (7.15) and (7.17) for maximum phase contribution

$$\begin{aligned}\phi_{max} &= \tan^{-1}\left(\frac{\sqrt{z_{le}p_{le}}}{z_{le}}\right) - \tan^{-1}\left(\frac{\sqrt{z_{le}p_{le}}}{p_{le}}\right) \\ &= \tan^{-1}\left(\frac{\omega_{max}}{z_{le}}\right) - \tan^{-1}\left(\frac{z_{le}}{\omega_{max}}\right)\end{aligned}$$

In order to have maximum phase of $33^o$ at 20 rad/s

$$33^0 = \tan^{-1}\left(\frac{20}{z_{le}}\right) - \tan^{-1}\left(\frac{z_{le}}{20}\right) \tag{7.18}$$

The numerical solution of (7.18) is $z_{le} \approx 10.8$. Then, the pole of the lead compensator is $p_{le} = (20^2/z_{le}) = 37$. The required lead compensator is

$$C_{le}(s) = K_{le}\frac{(s+10.8)}{(s+37)} \tag{7.19}$$

The lead compensator should not change the unity gain at 20 rad/s, which has already been designed. Therefore, lead compensator gain at 20 rad/s should be adjusted to unity using another gain $K_{le}$ as follows.

$$\begin{aligned} |C_{le}(j20)| &= K_{le}|\frac{(s+10.8)}{(s+37)}| \\ 1 &= K_{le}\frac{\sqrt{20^2+10.8^2}}{\sqrt{20^2+37^2}} \\ 1 &= K_{le}0.54 \\ 1.85 &= K_{le} \end{aligned}$$

Figure 7.9 shows the frequency response of the lead compensator, and the controlled plant is shown in Fig.7.10. Following MatLab code draws Fig.7.11, the gain and phase responses of the controlled plant.

```
numG=3*[1 3];
denG=conv([1 4],[1 3 20]);
G=tf(numG,denG);
bode(G); grid on;
[gainG,phaseG]=bode(G,20) % gain and phase at 20[rad/s]
% Bandwidth Adjustment
K=1/gainG
G1=K*G
bode(G1); grid on; hold on;
% Lead compensator z=10.8; p=20^2/z; lead compensator pole and
zero
CLe=tf([1 z],[1 p]);
[gainLe,phaseLe]=bode(CLe,20);
KLe=1/gainLe;
G2=KLe*CLe*G1;
```

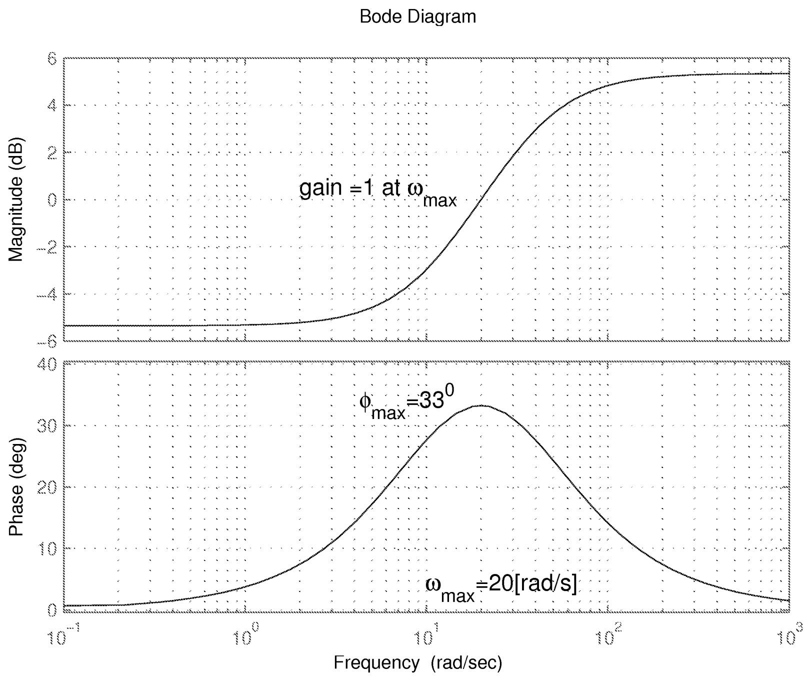



Figure 7.9: Gain and phase responses of the lead compensator

```
bode(G2) grid on;

z=10.8; p=20^2/z; lead compensator pole and zero
CLe=tf([1 z],[1 p]);
[gainLe,phaseLe]=bode(CLe,20);
KLe=1/gainLe;
G2=KLe*CLe*G1;
bode(G2) grid on;
```

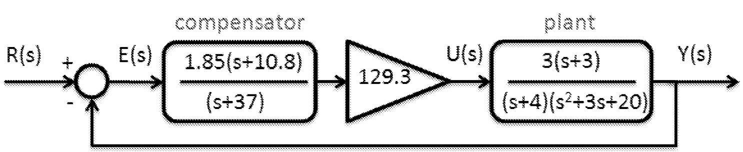



Figure 7.10: The controlled plant for 20[rad/s] bandwidth and 45$^0$ phase margin

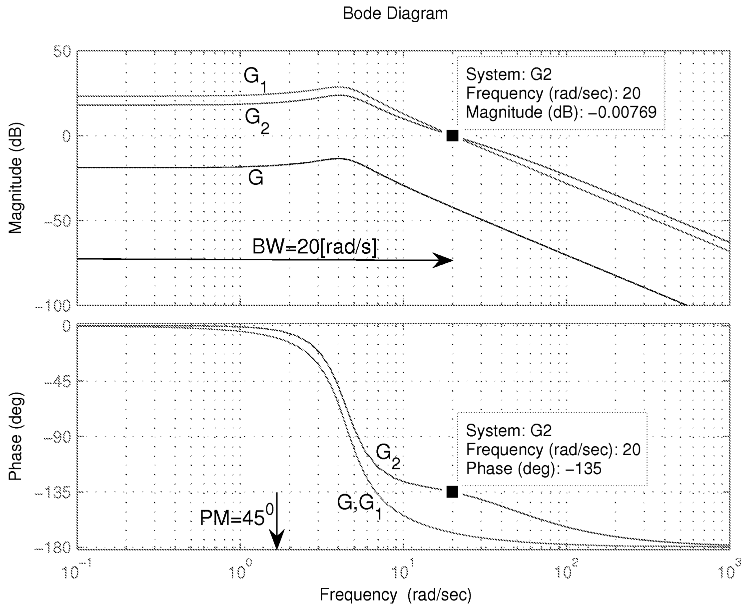



Figure 7.11: Gain and phase responses of the controlled plant for 20[rad/s] bandwidth and $45^0$ phase margin

**Steady State Error Adjustment**

The error signal of the controlled plant in Fig.7.10 is as follows.

$$\begin{aligned} E(s) &= R(s) - Y(s) \\ &= R(s) - E(s)C_{le}(s)KG(s) \\ E(s) &= \frac{1}{1 + C_{le}(s)KG(s)}R(s) \end{aligned}$$

The steady state error $e_{ss} = \lim_{t\to\infty} e(t)$. And, using final value theorem (3.42), the steady state error for unit step input $R(s) = 1/s$ can be determined as follows.

$$\begin{aligned} e_{ss} &= \lim_{s\to 0} sE(s) \\ &= \lim_{s\to 0} \frac{1}{1 + C_{le}(s)KG(s)} \\ &= \lim_{s\to 0} \frac{1}{1 + \frac{1.85(s+10.8)}{(s+37)} 129.3 \frac{(s+3)}{(s^2+3s+20)}} \\ &= \frac{1}{1 + 10.47} \\ &= 0.087 \end{aligned} \tag{7.20}$$

This error is unacceptable because it is more than the specified limit of 0.01. Therefore, a lag compensator $C_{la}(s) = \frac{s+z_{la}}{s+p_{la}}; p_{la} > z_{la}$ is introduced to improve the DCG of the plant. With the lag compensator, $E(s)$ will change as follows.

$$\begin{aligned} E(s) &= \frac{1}{1 + C_{la}(s)C_{le}(s)KG(s)} \\ &= \frac{1}{1 + \frac{(s+z_{la})}{(s+p_{la})} \frac{1.85(s+10.8)}{(s+37)} 129.3 \frac{3(s+3)}{(s^2+3s+20)}} \\ &= \frac{1}{1 + \frac{(s+z_{la})}{(s+p_{la})} 10.47} \\ &= \frac{p_{la}}{p_{la} + 10.47 z_{la}} \end{aligned}$$

In order to meet $e_s s = 0.01$,

$$\begin{aligned}
\frac{p_{la}}{p_{la}+10.47z_{la}} &= 0.01 \\
\frac{z_{la}}{p_{la}} &= \frac{0.99}{0.1047} \\
&= 9.46 \qquad (7.21)
\end{aligned}$$

This expression dictates only the proportion between pole and zero, thus, the actual locations can be freely selected based on other concerns. It is possible to have both pole and zero close to the origin (low frequency response), or farther away (high frequency response). Actual locations of them will affect frequency response only locally. In this view, both pole and zero should be located closer to the origin so that lag compensator does not cause significant changes in frequency response around the 20rad/s, the bandwidth frequency of the plant. Assuming $p_{la} = 0.1$, the zero is $z_{la} = 0.95$ from (7.21). Therefore, the lag compensator is

$$C_{la}(s) = \frac{(s+0.95)}{(s+0.1)} \qquad (7.22)$$

The frequency response of the lag compensator is shown in Fig.7.12, and the complete control system is shown in Fig. 7.13.

The following MatLab code includes all the design steps of the servo controller design. And, it draws the frequency responses of the controlled plant in each design step as shown in Fig.7.14.

```
% Bode Design
PM=45; ess=0.01;% Design specifications

numG=3*[1 3];
denG=conv([1 4],[1 3 20]);
G=tf(numG,denG);% System transfer function
bode(G); grid on; hold on;% draw Bode plots

% BW adjustment using gain
[gainG,phaseG]=bode(G,20)  % gain and phase at 20[rad/s]
K=1/gainG  % required gain to increase BW to 20[rad/s]
G1=K*G;  % BW adjusted system
bode(G1); grid on; hold on;  % draw Bode plots
```

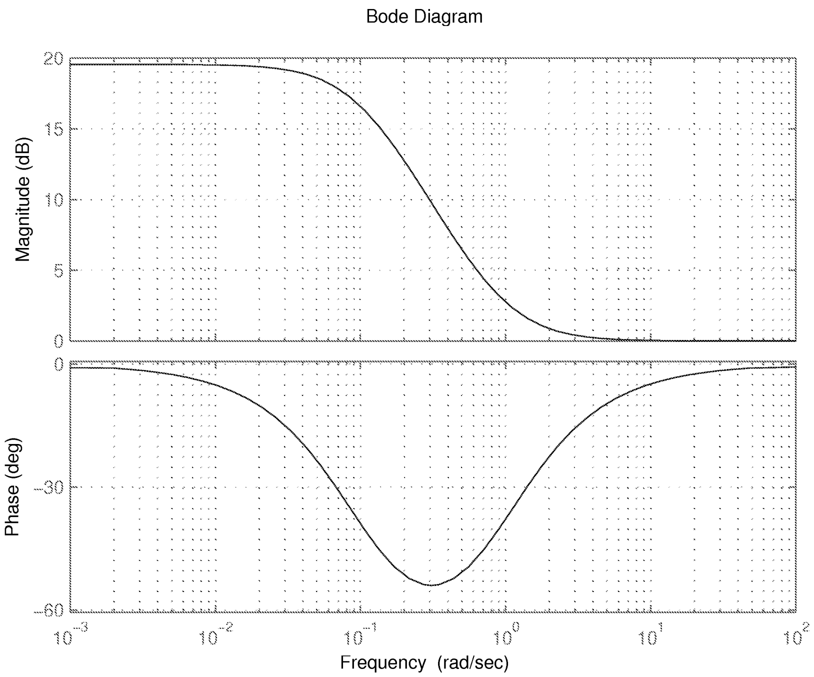



Figure 7.12: Frequency response of the lag compensator

```
% PM adjustment using a lead compensator
phi=PM-(180+phase)
% manual calculation: phi=atan(20/z)-atan(z/20) =¿ find z=10.8
z=10.8; p=(20/sqrt(z))2̂ % lead compensator pole and zero
CLe=tf([1 z],[1 p]); % lead compensator transfer function
[gainLe,phaseLe]=bode(CLe,20) % gain and phase of the lead at 20[rad/s]
KLe=1/gainLe % adjusts lead compensator gain=1 at 20 [rad/s]
G2=KLe*CLe*G1; % lead compensated system
bode(G2); grid on; hold on; % draw Bode plots
```

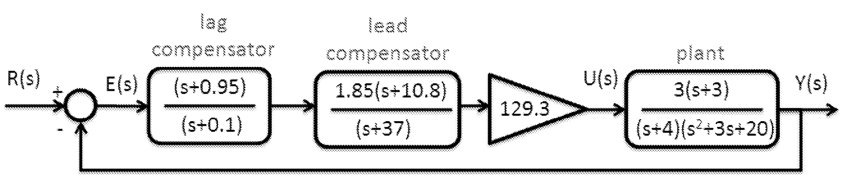



Figure 7.13: The controlled plant for 20[rad/s] bandwidth, $45^0$ phase margin, and $e_{ss} = 0.01$

```
% Steady state error adjustment by a lag compensator
% manual calculation: use lag CLa=(s+z)/(s+p)
% evaluate E=1/(1+CLaG2)
% and find z/p>9.46

p=0.1; z=9.5*p;% lag compensator pole and zero
CLa=tf([1 z],[1 p]);  % lag compensator transfer function
G3=CLa*G2;  % Complete control system
bode(G3);% draw Bode plots
```

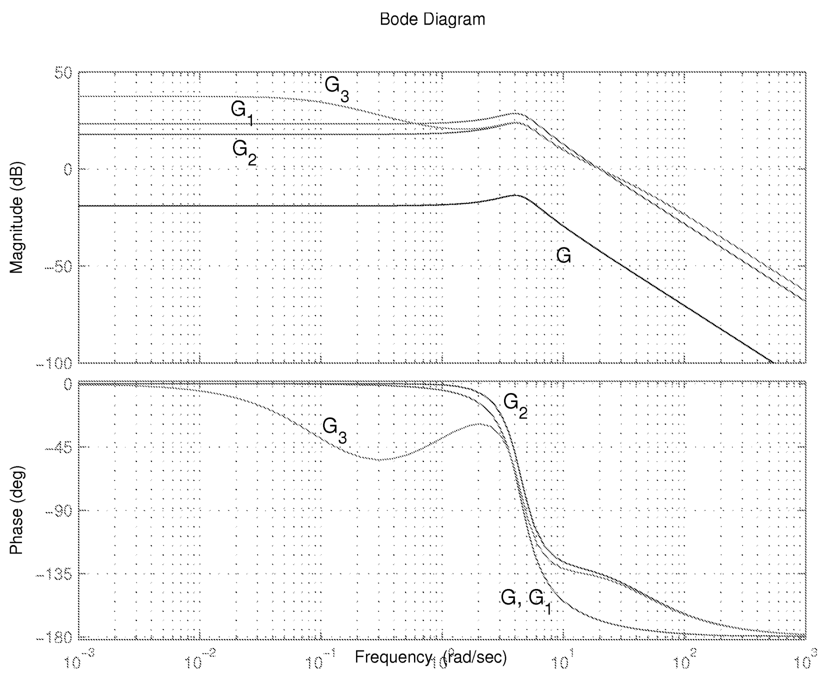



Figure 7.14: Frequency response of the controlled plant for 20[rad/s] bandwidth, $45^0$ phase margin, and $e_{ss} = 0.01$

**Simulation** The controlled plant in Fig. 7.14 can be build using `MatLab Simulink` as shown in Fig.7.15, in that three sinusoidal frequency components are used to synthesize the reference input. This Simulink plant can be run using the following `Matlab` code. This Matlab code creates the multi frequency reference input signal and excite the controlled plant with it.

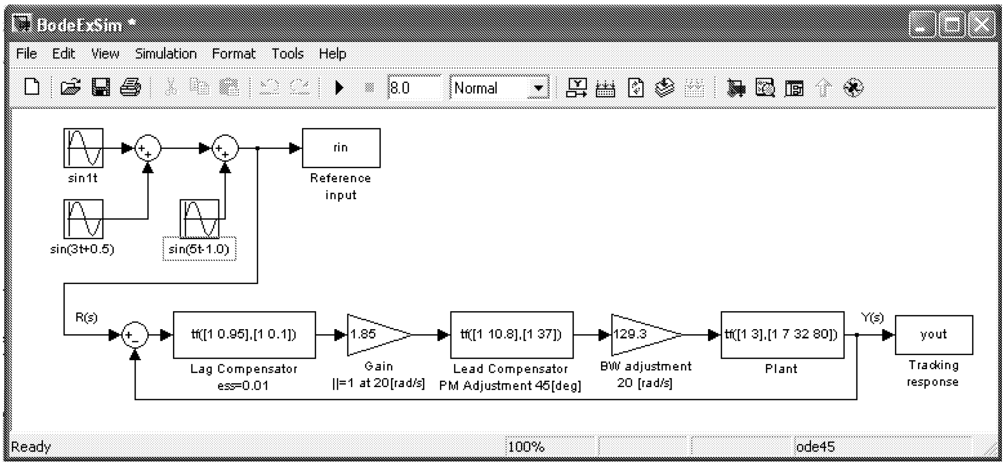



Figure 7.15: Controlled plant build in Matlab Simulink. Reference input is synthesized using three sinusoidals

```
% Reference input componants
a1=1; omega1=2; phi1=0; b1=0; % signal 1 attributes
a2=1; omega2=4; phi2=0.5; b2=0.1; % signal 2 attributes
a3=1; omega3=6; phi3=-1.0; b3=-0.2; % signal 3 attributes

sim BodeExSim; % calls simulink block

plot(tout,yout,'b',tout,rin,'r--'); % draw r(t) and y(t)
xlabel('time[s]'); ylabel('r(t) and y(t)'); grid on;
legend('y(t)','r(t)'); % add legend to graph
```

In this code BodeExSim is the simulink plant. Reference input is synthesized by adding three sinusoidals as $r(t) = \Sigma_{i=1}^{3} a_i \sin \omega_i t + \phi_i + b_i$, in that their amplitudes, frequencies, phases, and biases can be specified in the `MatLab` code. The tracking response of the plant for low a frequency reference $r(t) = [1. \sin t] + [1. \sin(3t + 0.5)] + [1. \sin(6t - 1.0)]$ is shown in Fig.7.16. In this result, the reference starts at -3.5, whereas the response start at 0. However, the response is able to closely track the reference. Very accurate tracking response cannot be expected because the controlled plant has a lower bandwidth of 20 rad/s, or 3.18Hz, and the lag compensator introduces a substantial phase distortion within this low bandwidth. More accurate tracking response can be achieved for a higher bandwidth designs.

Figure 7.17 shows the tracking response when high frequency components present in the reference input given by $r(t) = [1. \sin 8t] + [1. \sin(10t + 0.5)] +$

$[1.\sin(12t - 1.0)]$. In this result, tracking performance has been deteriorated to some extent due to frequency attenuation closer to 20 rad/s. The time delay in feedback is also more prominent. Nevertheless, if time delay between reference and response is disregarded, the response still tracks the reference somewhat satisfactorily. If the reference input consists of even higher frequencies beyond 20 rad/s, the tracking performance will deteriorate beyond acceptable level.

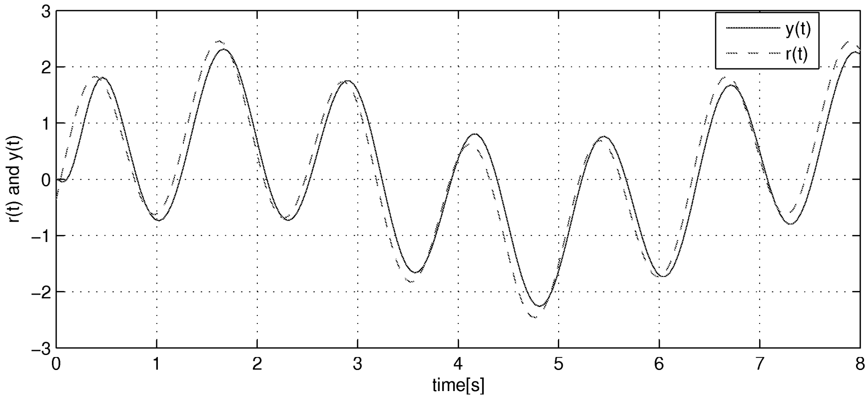



Figure 7.16: Tracking response of the controlled plant for a time-varying reference which contain 1[rad/s], 3[rad/s], and 5[rad/s] frequency components

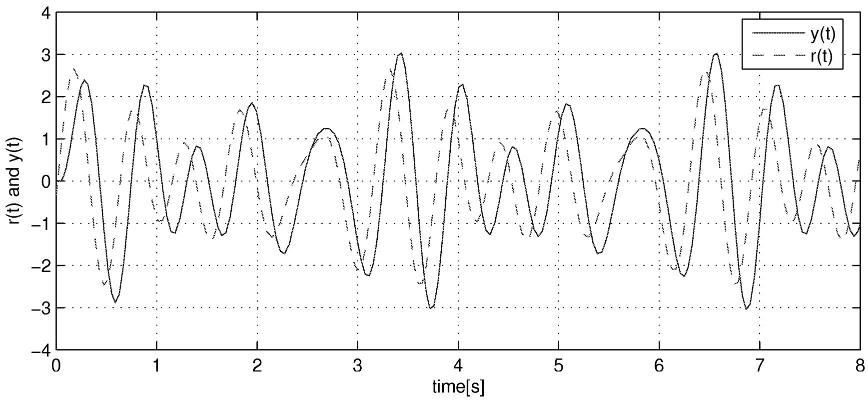



Figure 7.17: Tracking response of the servo system for a time-varying reference which contain 8[rad/s], 10[rad/s], and 12[rad/s] frequency components

A good collection of examples on servo system design using frequency response is available in [9].

## 7.4 Summary

In control application areas such as mechatronics and robotics, the actuators are given time-varying position references (e.g. knee joint motor of a bipedal walking robot). The control systems of these actuators should be designed in order to be able to track these reference inputs quickly and accurately. These controllers are known as servo controllers and they are designed in frequency domain. The technique used in servo controller design is procedure of few steps. First, system bandwidth is adjusted using a forward gain. After that, appropriate compensators are designed based on spefified performance requirements. Lead and lag compensators are usually employed in seperate frequency localities in order to shape up the overall frequency response. The entire servo controller can be designed and simulated in MatLab and SImulink. Accurate tracking response must be achieved in simulation before fabrication and implementation of the servo controllers. The actual industrial servo controller are more complex than what is presented here, and they achieve greater tracking performance by way of feedforwarding the velocity and acceleration of the reference input.