AAE 364: Control Systems Analysis

HW12: Bode, Nyquist Plot & Practice

Dr. Sun

School of Aeronautical and Astronautical

Purdue University

Tomoki Koike

Friday May 1st, 2020

B–7–14. Consider a unity-feedback control system with the following open-loop transfer function:

$$G(s) = \frac{s^2 + 2s + 1}{s^3 + 0.2s^2 + s + 1}$$

Draw a Nyquist plot of $G(s)$ and examine the stability of the closed-loop system.

Poles & Zeros

i	Poles, P_i	Zeros, Z_i
1	0.2623+1.1451j	-1
2	0.2623-1.1451j	-1
3	-0.7246	

Corner Frequencies & Limits

$$\omega_1 = \|P_3\| = 0.7246$$

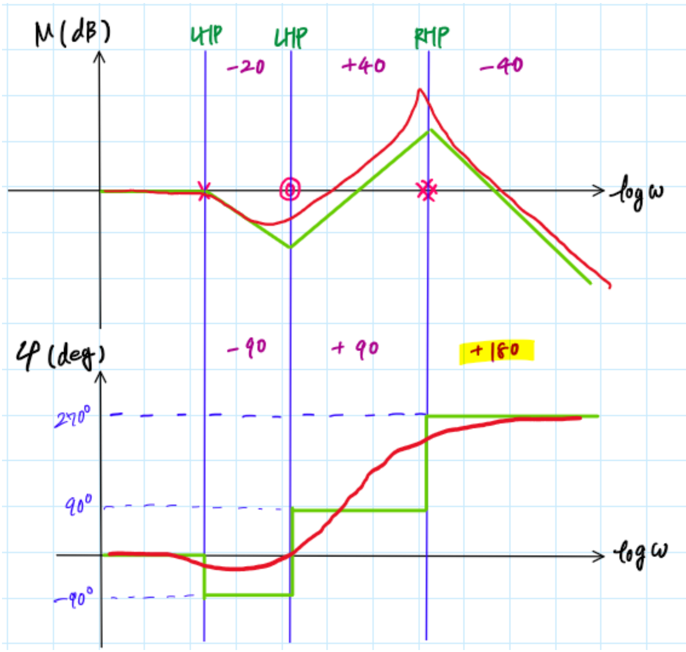
$$\omega_2 = \|Z_1\| = \|Z_2\| = 1$$

$$\omega_3 = \|P_1\| = \|P_2\| = \sqrt{0.2623^2 + 1.1451^2} = 1.1748$$

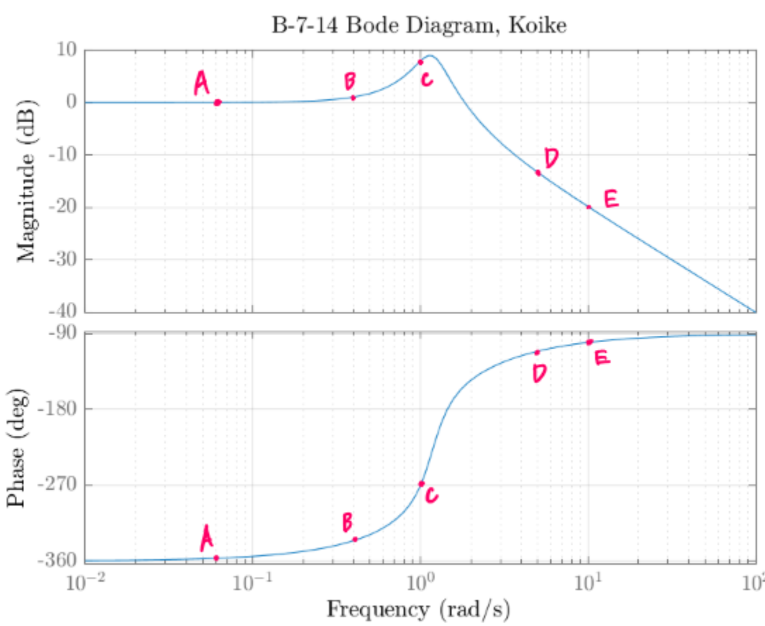
$$\lim_{\omega\to 0} G(j\omega) = \lim_{\omega\to 0} \frac{-\omega^2 + 2j\omega + 1}{-j\omega^3 - 0.2\omega^2 + j\omega + 1} = 1\angle 0^\circ$$

$$\lim_{\omega\to\infty} G(j\omega) = \lim_{\omega\to\infty} \frac{-\omega^2}{-j\omega^3} = \lim_{\omega\to\infty} -\frac{1}{\omega}j = 0\angle -90^\circ$$

Bode Plot Sketch



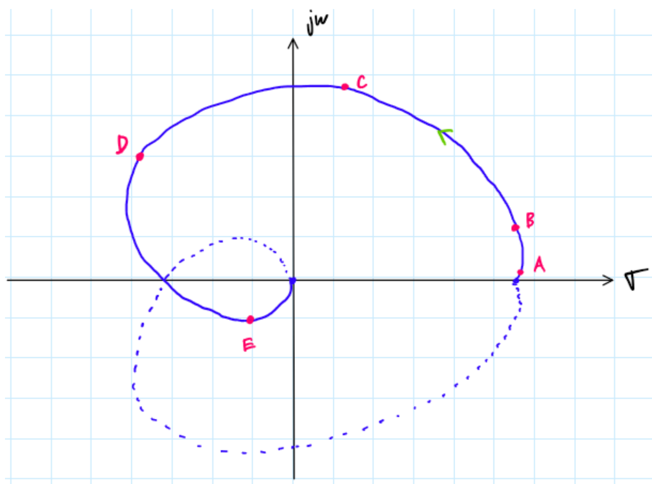
Bode Plot (MATLAB)



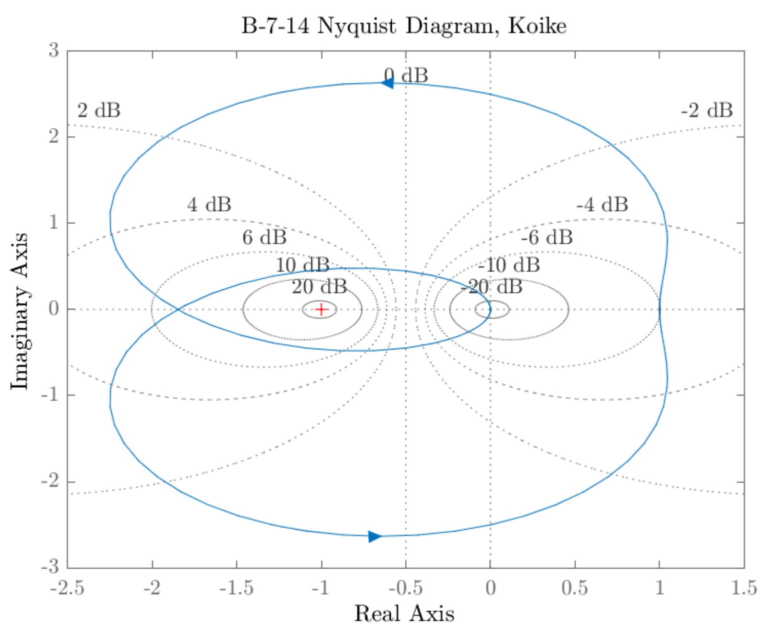
Sample Points from Bode Plot

Point	ω [rad/s]	$\angle G$ [deg]	$-20\ log_{10}\lvert G\rvert$ [dB]	$\lvert G\rvert$
A	0.05	-357.1318766	-0.015231146	1.001755089
B	0.4	-335.5394486	-1.07759798	1.132087249
C	1	-269.9999453	-7.958797179	2.499999138
D	4	-115.9725883	10.95987844	0.283143162
E	10	-100.3217069	19.82787424	0.102001437

Nyquist Plot Sketch



Nyquist Plot (MATLAB)



Nyquist Stability Analysis

Calculate the intersection of Nyquist Plot and σ-axis on the very left.

$$G(j\omega) = \frac{-\omega^2 + 2j\omega + 1}{-j\omega^3 - 0.2\omega^2 + j\omega + 1}$$

$$G(j\omega) = \frac{(1-\omega^2) + 2j\omega}{(1-0.2\omega^2) + (\omega - \omega^3)j}$$

$$G(j\omega) = \frac{[(1-\omega^2) + 2j\omega][(1-0.2\omega^2) - (\omega - \omega^3)j]}{(1-0.2\omega^2)^2 + (\omega - \omega^3)^2}$$

then the numerator equals

$$(1-\omega^2)(1-0.2\omega^2) - (1-\omega^2)(\omega-\omega^3)j + 2j\omega(1-0.2\omega^2) - 2j\omega(\omega-\omega^3)j$$

$$\Rightarrow 1 - 0.2\omega^2 - \omega^2 + 0.2\omega^4 - (\omega - \omega^3 - \omega^3 + \omega^5)j + (2\omega - 0.4\omega^3)j + 2\omega^2 - 2\omega^4$$

$$\Rightarrow (-1.8\omega^4 + 0.8\omega^2 + 1) + (-\omega^5 + 1.6\omega^3 + \omega)j$$

Now the Imaginary part equals 0

$$\omega(-\omega^4 + 1.6\omega^2 + 1) = 0$$

then

$$\omega = 0, \pm 1.4424$$

plug this into $G(j\omega)$ and we obtain

$$\boxed{\sigma = -1.8508} < -1$$

P: the number of OL poles in RHP
N: the number of clockwise encirclements about -1
Z: the number of CL poles in RHP $\Leftrightarrow$ Z = P + N

P	N	Z
2	2	4

The system is unstable

B–7–15. Consider the unity-feedback system with the following $G(s)$:

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

Suppose that we choose the Nyquist path as shown in Figure 7–156. Draw the corresponding $G(j\omega)$ locus in the $G(s)$ plane. Using the Nyquist stability criterion, determine the stability of the system.

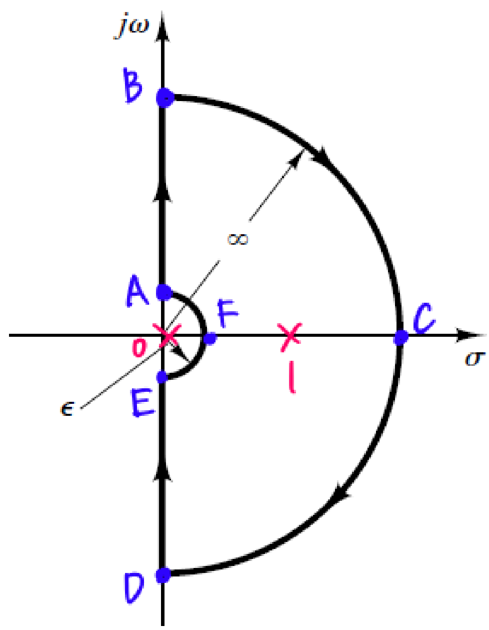


Figure 7–156
Nyquist path.

① $\overline{AB}$: $s = j\omega$, $\omega : \varepsilon \rightarrow \infty$

② $\widehat{BCD}$: $s = \infty$

③ $\overline{DE}$: $s = j\omega$, $\omega : -\infty \rightarrow \varepsilon$

④ $\widehat{EFA}$: $s = \varepsilon e^{j\theta}$, $\theta : -90° \rightarrow 0° \rightarrow 90°$

① $\omega = \varepsilon$, $G(j\omega) = \dfrac{1}{j\varepsilon(j\varepsilon - 1)} \approx -\dfrac{1}{j\varepsilon} = \dfrac{1}{\varepsilon}j = \infty\angle 90°$ (A′)

$\omega = \infty$, $G(j\omega) = \dfrac{1}{j\omega(j\omega - 1)} = -\dfrac{1}{\omega^2} \underline{\angle \lim_{\omega\to\infty}(90° + \arctan\omega)}$

$= 0 \angle 180°$ (B′)

② $s = \infty \Rightarrow G(s) = 0$ (B′) (C′) (D′)

③ $s = j\omega$, $\omega : -\infty \rightarrow \varepsilon$ $\overline{D'E'} = \mathrm{conj}(\overline{A'B'})$

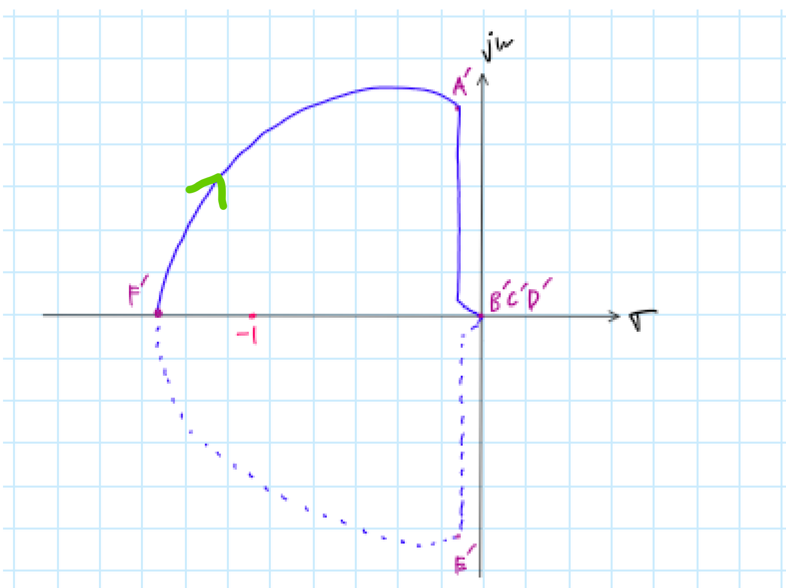
④ $s = \varepsilon e^{j\theta}$, $\theta : -90° \rightarrow 0° \rightarrow 90°$

$G(s) = \dfrac{1}{\varepsilon e^{j\theta}(\varepsilon e^{j\theta} - 1)} \approx \dfrac{1}{-\varepsilon e^{j\theta}} = -\varepsilon e^{-j\theta}$

$= \varepsilon e^{j(180° - \theta)}$

$\underline{\angle G(s)} = 270° \rightarrow 180° \rightarrow 90°$

Nyquist Plot Sketch



Nyquist Stability Analysis

P: the number of OL poles in RHP
N: the number of clockwise encirclements about -1
Z: the number of CL poles in RHP $\Leftrightarrow$ Z = P + N

P	**N**	**Z**
1	1	2

The system is unstable

B–7–16. Consider the closed-loop system shown in Figure 7–157. $G(s)$ has no poles in the right-half s plane.

If the Nyquist plot of $G(s)$ is as shown in Figure 7–158(a), is this system stable?

If the Nyquist plot is as shown in Figure 7–158(b), is this system stable?

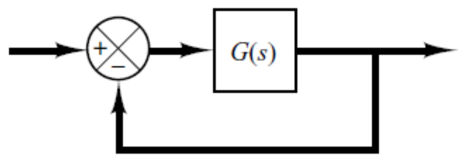
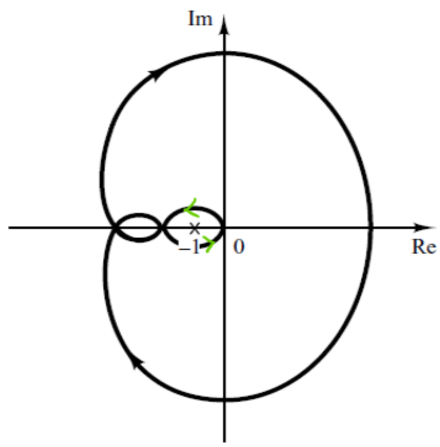


Figure 7–157
Closed-loop system.



(a)

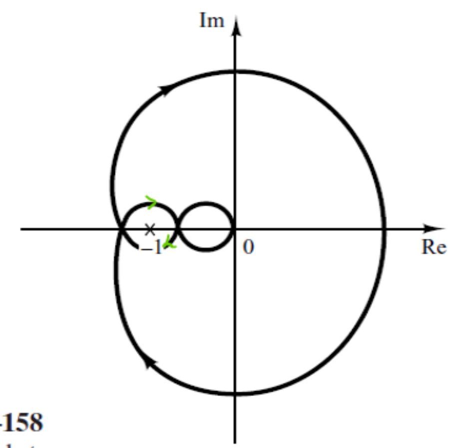


Figure 7–158
Nyquist plots.

(b)

7-158(a)

There is one clockwise encirclement outside and one counter-clockwise encirclement inside. Thus, the net clockwise encirclement becomes 0.

P: the number of OL poles in RHP
N: the number of clockwise encirclements about -1
Z: the number of CL poles in RHP ⟺ Z = P + N

P	**N**	**Z**
0	0	0

The system is stable

7-158(b)

There is one clockwise encirclement outside and another clockwise encirclement inside. Thus, the net clockwise encirclement becomes 2.

P: the number of OL poles in RHP
N: the number of clockwise encirclements about -1
Z: the number of CL poles in RHP ⟺ Z = P + N

P	**N**	**Z**
0	2	2

The system is unstable

B–7–19. Consider a negative-feedback system with the following open-loop transfer function:

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

Plot the Nyquist diagram of $G(s)$. If the system were a positive-feedback one with the same open-loop transfer function $G(s)$, what would the Nyquist diagram look like?

Poles & Zeros

i	Poles, P_i	Zeros, Z_i
1	0	
2	-1	
3	-2	

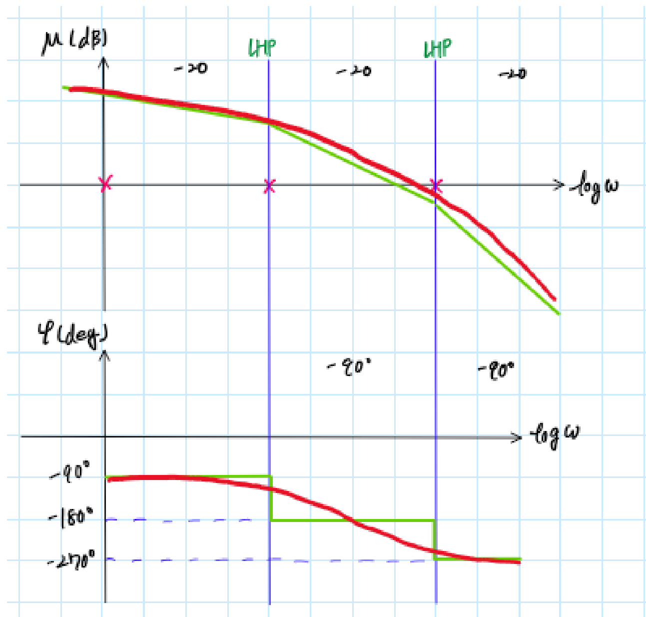
Corner Frequencies & Limits

$$\omega_1 = \|P_1\| = 0$$
$$\omega_2 = \|P_2\| = 1$$
$$\omega_3 = \|P_3\| = 2$$

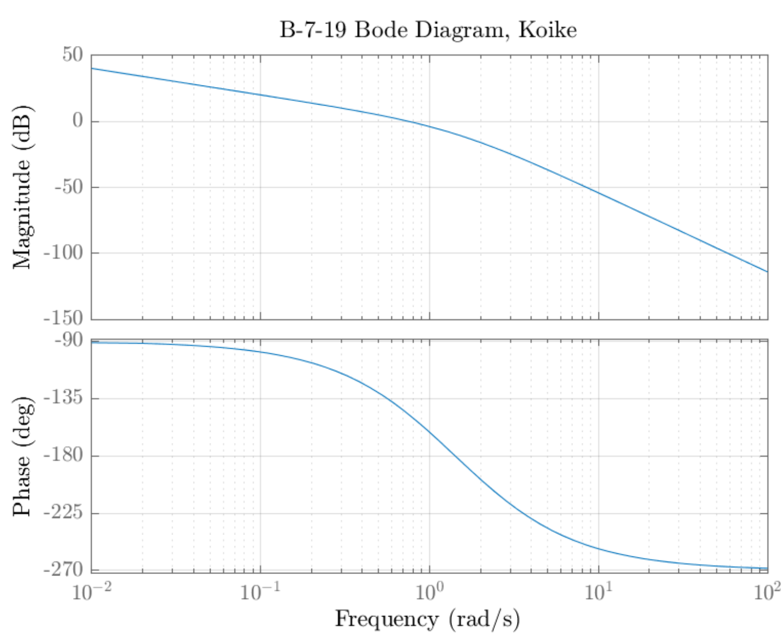
$$\lim_{\omega \to 0} G(j\omega)$$
$$= \lim_{\omega \to 0} \left\| \frac{2}{j\omega(j\omega+1)(j\omega+2)} \right\| \lim_{\omega \to 0} \left(-90^\circ - \arctan\omega - \arctan\frac{\omega}{2}\right)$$
$$= \infty \angle -90^\circ$$

$$\lim_{\omega \to \infty} G(j\omega)$$
$$= \lim_{\omega \to \infty} \left\| \frac{2}{j\omega(j\omega+1)(j\omega+2)} \right\| \lim_{\omega \to \infty} \left(-90^\circ - \arctan\omega - \arctan\frac{\omega}{2}\right)$$
$$= \lim_{\omega \to \infty} \left\| \frac{2}{j\omega^3} \right\| \angle(-90^\circ - 90^\circ - 90^\circ)$$
$$= 0 \angle -270^\circ$$

Bode Plot Sketch



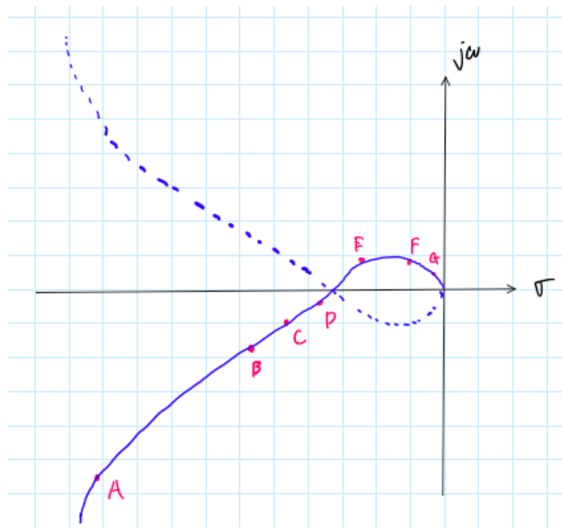
Bode Plot (MATLAB)



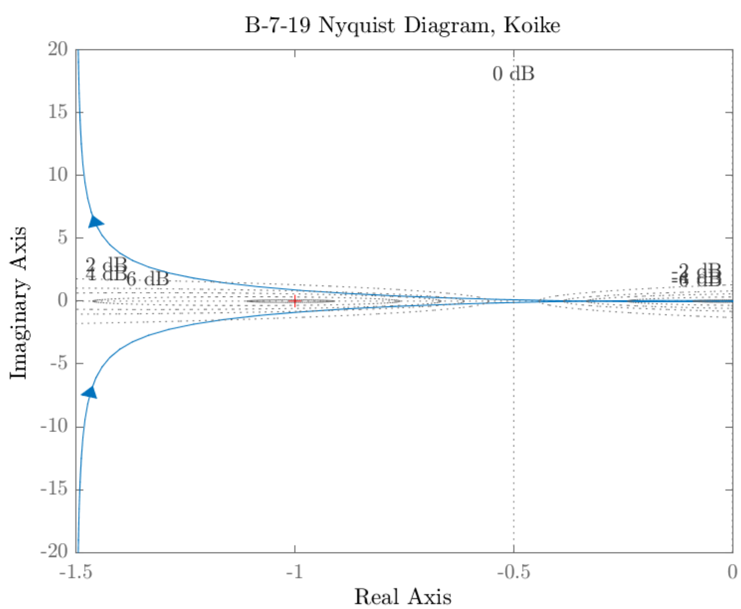
Sample Points from Bode Plot

Point	ω [rad/s]	$\angle$G [deg]	$-20\ log_{10}\lvert G\rvert$ [dB]	|G|
A	0.05	-94.29450141	-26.00704399	19.96881068
B	0.2	-107.0205255	-13.76585437	4.878571998
C	0.7	-144.2820646	-0.864315168	1.104627265
D	1	-161.5650472	3.979396614	0.632455785
E	4	-229.3986988	31.33538003	0.027116335
F	10	-252.9794718	54.19293923	0.00195143
G	15	-258.5912795	64.6406538	0.000586094

Nyquist Plot Sketch



Nyquist Plot (MATLAB)

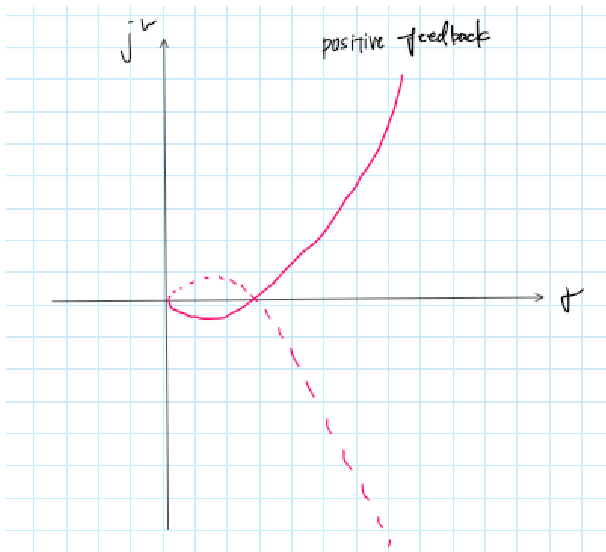


Discussion

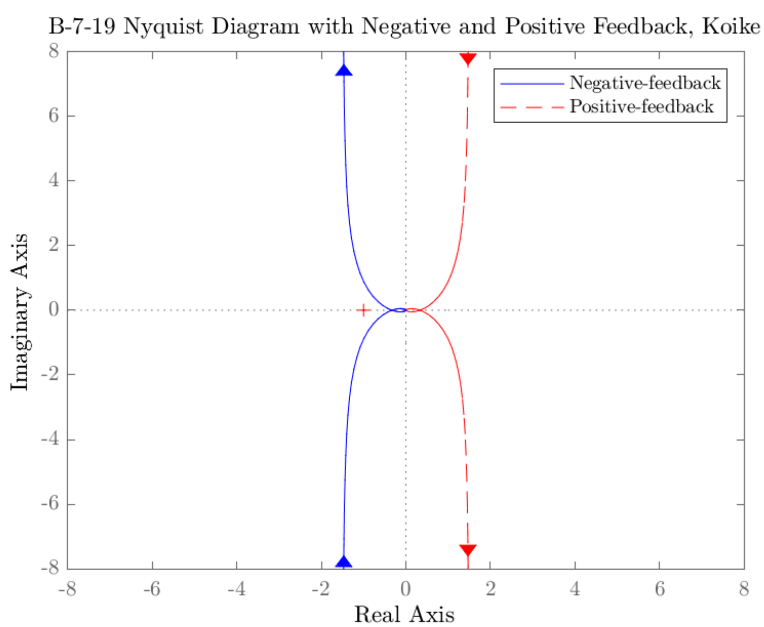
If the system were to be a positive feedback system it would change the Nyquist plot by rotating the plot 180°. This is because the Nyquist stability criterion is changed to assess the encirclements to be based on the point [1, 0] and not [-1, 0] because the characteristic equation is changed to the following

$$CE:\ 1 + G(s)H(s)$$

Nyquist Plot Sketch (Positive Feedback)



Nyquist Plot (MATLAB) (Positive Feedback)



B–7–24. Consider the system shown in Figure 7–161. Draw a Bode diagram of the open-loop transfer function $G(s)$. Determine the phase margin and gain margin.

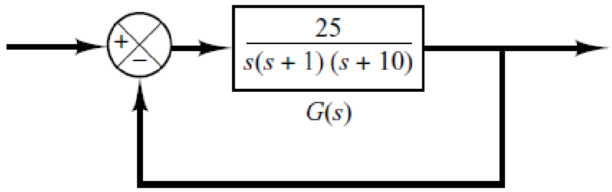


Figure 7–161
Control system.

Poles & Zeros

i	Poles, P_i	Zeros, Z_i
1	0	
2	-1	
3	-10	

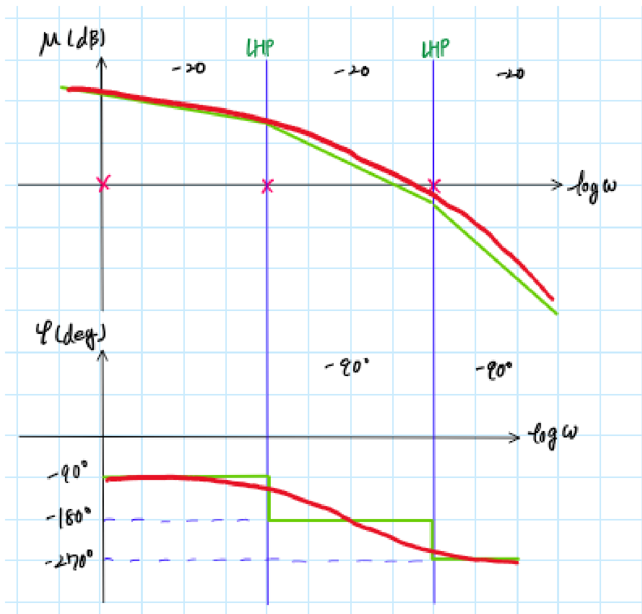
Corner Frequencies & Limits

$$\omega_1 = \|P_1\| = 0$$
$$\omega_2 = \|P_2\| = 1$$
$$\omega_3 = \|P_3\| = 10$$

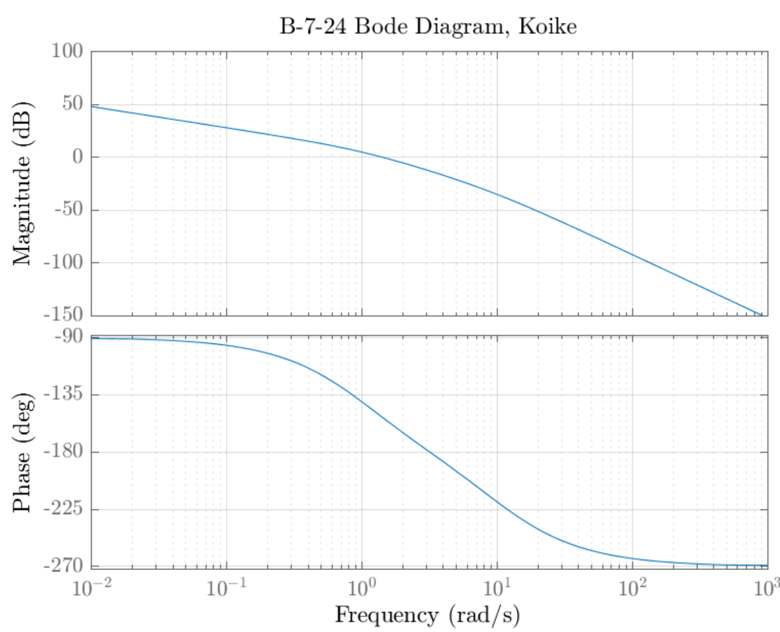
$$\lim_{\omega \to 0} G(j\omega)$$
$$= \lim_{\omega \to 0} \left\| \frac{25}{j\omega(j\omega+1)(j\omega+10)} \right\| \lim_{\omega \to 0} \left(-90^\circ - \arctan\omega - \arctan\frac{\omega}{10}\right)$$
$$= \infty \angle -90^\circ$$

$$\lim_{\omega \to \infty} G(j\omega)$$
$$= \lim_{\omega \to \infty} \left\| \frac{25}{j\omega(j\omega+1)(j\omega+10)} \right\| \lim_{\omega \to \infty} \left(-90^\circ - \arctan\omega - \arctan\frac{\omega}{10}\right)$$
$$= 0 \angle -270^\circ$$

Bode Plot Sketch



Bode Plot (MATLAB)



Phase Margin

$$\|G(j\omega)\| = 1$$

$$\iff \quad 25 = \|j\omega\| \, \|j\omega+1\| \, \|j\omega+10\|$$

$$625 = (10\omega-\omega^3)^2 + 121\omega^4$$

$$\omega^6 - 20\omega^4 + 121\omega^4 + 100\omega^2 - 625 = 0$$

$$\omega^6 + 101\omega^4 + 100\omega^2 - 625 = 0$$

The only real and positive solution is

$$\boxed{\omega_{gc} = 1.4230}$$

Then using MATLAB (code in Appendix)

$$\varphi = \arg[G(j\omega_{gc})] = -153.0027^\circ$$

Thus, phase margin becomes

$$\gamma = 180^\circ + \varphi$$

$$\gamma = 180^\circ - 153.0027^\circ$$

$$\boxed{\gamma = 26.9973^\circ}$$

Gain Margin

$$G(j\omega) = \frac{25}{j\omega(j\omega+1)(j\omega+10)} = \frac{25}{-j\omega^3 - 11\omega^2 + 10j\omega}$$

$$= \frac{25}{den}(\overline{den})$$

$$num \rightarrow 25j\omega^3 - 275\omega^2 - 250j\omega$$

the imaginary part $= 0$

$$25\omega^3 - 250\omega = 0$$

$$\Rightarrow \omega = \pm 3.1623, \ 0$$

$$\therefore \omega_{pc} = 3.1623$$

then using MATLAB (code in Appendix)

$$\boxed{K_g = \frac{1}{\|G(j\omega_{pc})\|} = \frac{1}{0.2273} = 4.4000}$$

The gain margin in dB is

$$K_g = -20\log(4.4000)$$

$$\boxed{K_g = 12.8691 \quad dB}$$

B–7–26. Consider a unity-feedback control system with the open-loop transfer function

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

Determine the value of the gain K such that the phase margin is 50°. What is the gain margin with this gain K?

Phase Margin

$$G(j\omega) = \frac{K}{j\omega[(j\omega)^2 + j\omega + 4]} = \frac{K}{j\omega(-\omega^2 + j\omega + 4)}$$

$$= \frac{K}{-j\omega^3 - \omega^2 + 4j\omega} = \frac{K}{-\omega^2 + (4\omega - \omega^3)j}$$

$$= \frac{-K\omega^2 - K(4\omega - \omega^3)j}{[-\omega^2 + (4\omega - \omega^3)j][-\omega^2 - (4\omega - \omega^3)j]}$$

If $\|G(j\omega)\| = 1$

$$K^2 = \omega^4 + (4\omega - \omega^3)^2$$

$$K^2 = \omega^4 + \omega^6 - 8\omega^4 + 16\omega^2$$

$$0 = \omega^6 - 7\omega^4 + 16\omega^2 - K^2 \quad \cdots ①$$

Since phase margin is 50°

$$\arg[G(j\omega_{gc})] = -\arg(j\omega_{gc}) - \arg(-\omega_{gc}^2 + j\omega_{gc} + 4)$$

$$-90° - \arctan\left(\frac{\omega_{gc}}{4 - \omega_{gc}^2}\right) = 50° - 180°$$

$$-\arctan\left(\frac{\omega_{gc}}{4 - \omega_{gc}^2}\right) = -40°$$

$$\frac{\omega_{gc}}{4 - \omega_{gc}^2} = \tan(40°)$$

$\omega_{gc} = -2.6828,\ 1.4910$

since $\omega_{gc} > 0$

$$\boxed{\omega_{gc} = 1.4910}$$

Plug this into ①

$$\boxed{K = 3.4585}$$

Gain Margin

For the gain margin

$$G(j\omega) = \frac{-K\omega^2 - K(4\omega-\omega^3)j}{[-\omega^2+(4\omega-\omega^3)j][-\omega^2-(4\omega-\omega^3)j]}$$

$$\Rightarrow \quad 4\omega - \omega^3 = 0 \quad \Rightarrow \quad \omega(2-\omega)(2+\omega) = 0$$

$$\omega_{pc} = 2$$

Using MATLAB we compute

$$G(j2) = -0.8646$$

Then

$$\boxed{\begin{aligned} K_g &= \frac{1}{|-0.8646|} = 1.1566 \\ K_g &= 20\log(1.1566) = 1.2634\ \text{dB} \end{aligned}}$$

Problem 2

Figure 1 shows a Bode diagram of a transfer function $G(s)$ which is minimum phase. Determine this transfer function.

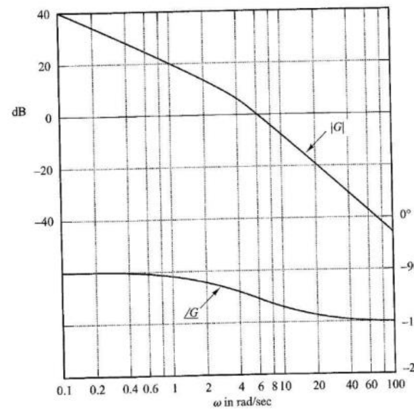
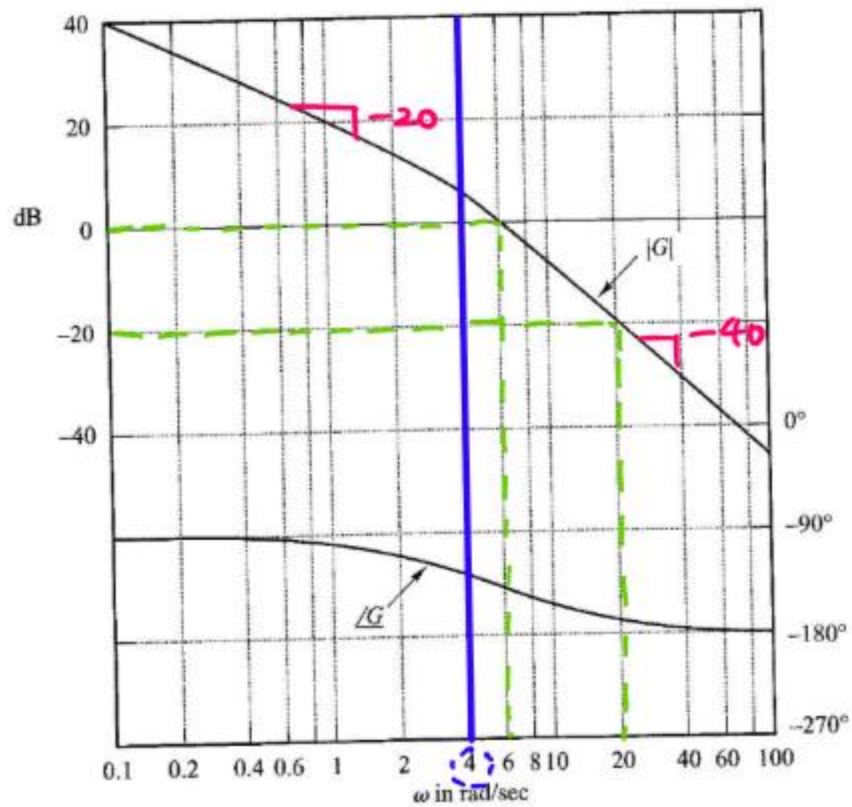


Figure 1: Bode diagram of a transfer function $G(s)$.



Since at $\omega \to 0$

$$\lim_{\omega \to 0} G(j\omega) = \infty \angle -90^\circ$$

This is a type 1 system.
From the Bode Plot we know that there is a pole at -4
and no zero since this is minimum phase.

Thus,

$$\boxed{G(s) = \frac{K}{s(s+4)} \qquad (K > 0)}$$

When $\omega = 6$ rad/s $\|G(j\omega)\| = 0$ dB

$$\|G(j\omega)\| = \frac{K}{6\,\|4 + 6j\|} = 1$$

$$\frac{K}{6\sqrt{16+36}} = 1$$

$$K = 12\sqrt{13}$$

$$K = 43.27$$

$$\boxed{G(s) = \frac{43.27}{s(s+4)}}$$

Problem 3: Aircraft Example

The following figure shows the coordinate axes and forces acting on the aircraft in the longitudinal plane of motion. Assuming that the aircraft is cruising at constant velocity and altitude.

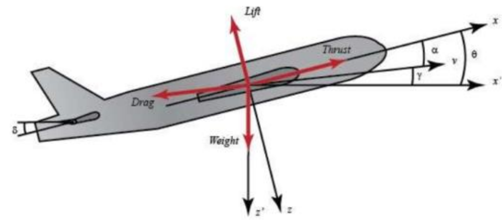


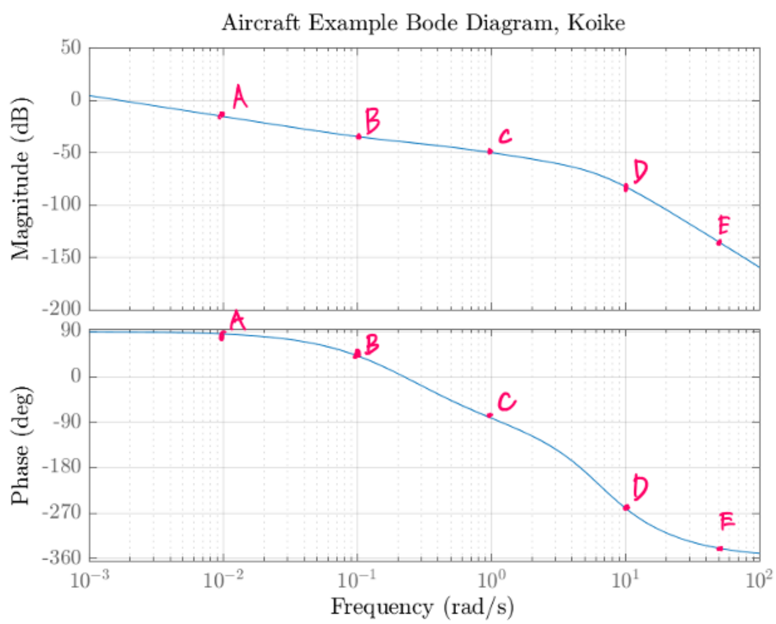
Figure 2: Forces acting on an aircraft in the Longitudinal plane.

Draw the Nyquist plot of the following $G(s)$:

1. $G(s)$ representing the aircraft altitude response output to the elevator deflection input:

$$G(s) = \frac{H(s)}{\Delta(s)} = \frac{1.1057s - 0.1900}{s^5 + 17.95s^4 + 123.3s^3 + 366.3s^2 + 112.2s}$$

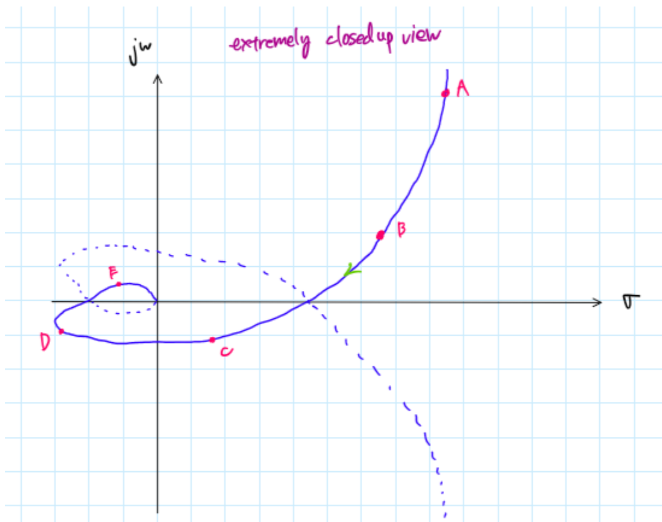
Bode Plot (From HW11)



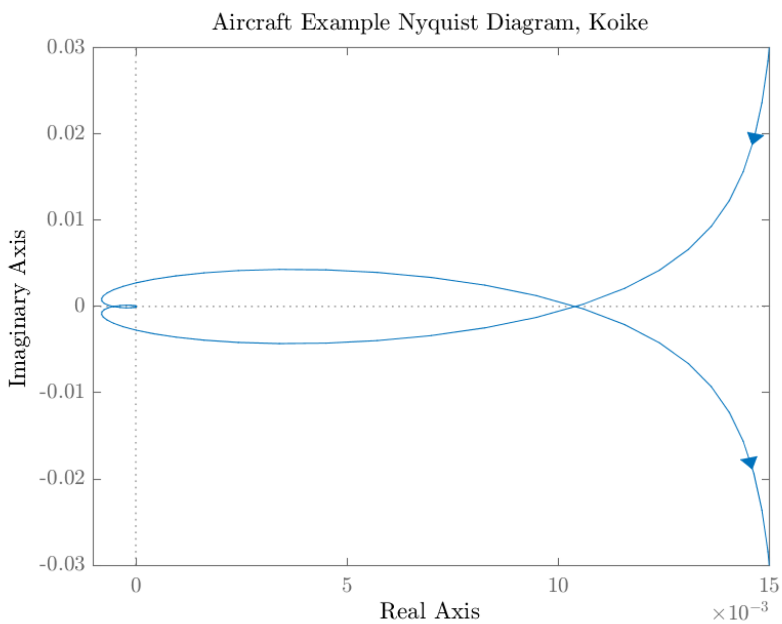
Sample Points from Bode Plot

Point	ω [rad/s]	$\angle G$ [deg]	$-20\ log_{10}\lvert G\rvert$ [dB]	$\lvert G\rvert$
A	0.01	84.79937283	15.41377481	0.169555257
B	0.1	41.54326143	34.51085838	0.018812958
C	1	-81.91045187	49.8448367	0.003219276
D	10	-260.191817	82.32819474	7.64875E-05
E	50	-339.2704051	135.1757373	1.74266E-07

Nyquist Plot Sketch



Nyquist Plot (MATLAB)



Problem 4: Spacecraft

Consider the plant $G(s)$ representing the spacecraft attitude dynamics shown in Figure 3:

$$G(s) = \frac{\theta(s)}{T_c(s)} = \frac{0.036(s+25)}{s^2(s^2+0.04s+1)} \tag{1}$$

Draw the Nyquist plot of $G(s)$.

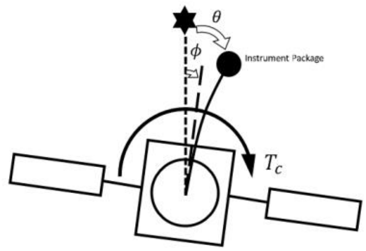
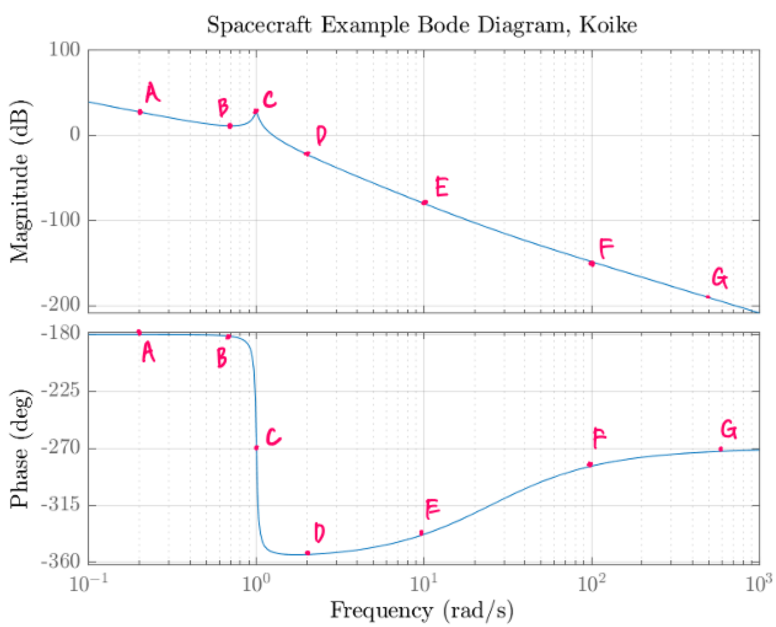


Figure 3: Two-body Model of Satellite

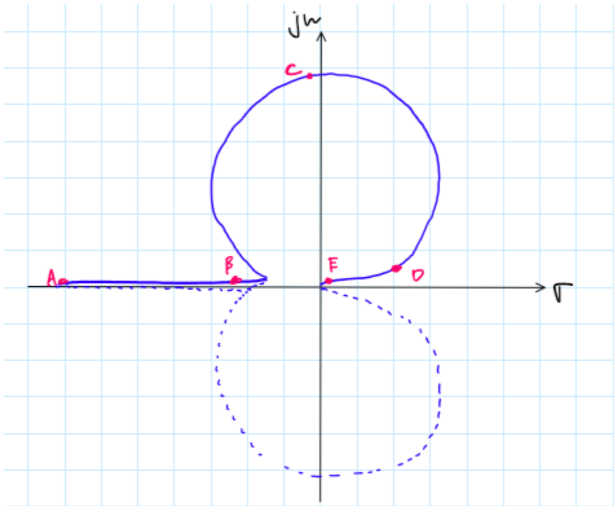
Bode Plot (From HW11)



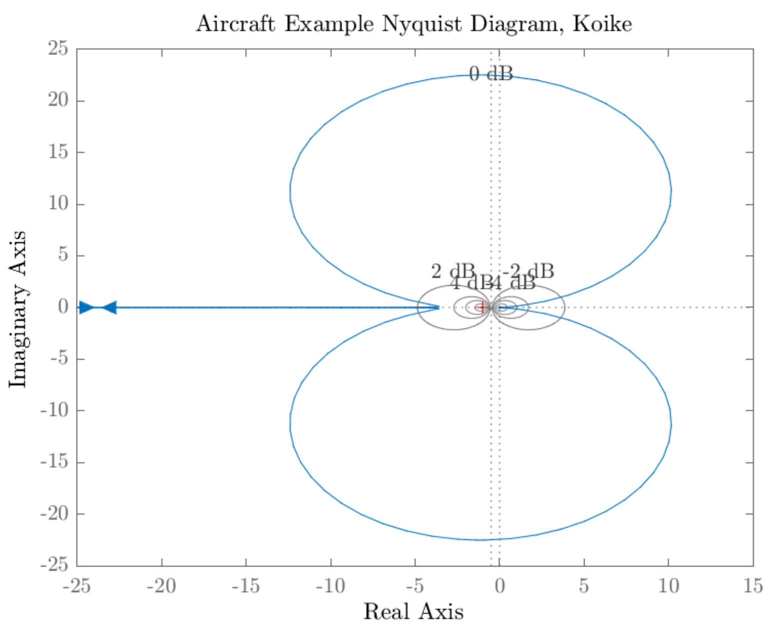
Sample Points from Bode Plot

Point	ω [rad/s]	$\angle G$ [deg]	$-20\ log_{10}\lvert G\rvert$ [dB]	$\lvert G\rvert$
A	0.2	-180.0190973	-27.39820206	23.4374362
B	0.7	-181.538633	-11.11985768	3.597434405
C	1	-267.7093899	-27.05059318	22.51799191
D	2	-353.8985533	22.47415567	0.07521288
E	10	-337.9670937	80.18334471	9.79113E-05
F	100	-284.0133229	148.6097927	3.71117E-08
G	600	-272.3821243	195.5554676	1.66812E-10

Nyquist Plot Sketch



Nyquist Plot (MATLAB)



Appendix

MATLAB CODE

AAE 364 HW11

```
clear all; close all; clc;
fdir = 'C:\Users\Tomo\Desktop\studies\2020-Spring\AAE364\matlab\matlab_output\hw12';
set(groot, 'defaulttextinterpreter','latex');
set(groot, 'defaultAxesTickLabelInterpreter','latex');
set(groot, 'defaultLegendInterpreter','latex');

% Bode plot options
opts_bd = bodeoptions('cstprefs');
opts_bd.Title.Interpreter = "latex";
opts_bd.XLabel.Interpreter = "Latex";
opts_bd.YLabel.Interpreter = "Latex";
opts_bd.Grid = 'on';

% Nyquist plot options
opts_nq = nyquistoptions("cstprefs");
opts_nq.Title.Interpreter = 'latex';
opts_nq.XLabel.Interpreter = "Latex";
opts_nq.YLabel.Interpreter = "Latex";
opts_nq.Grid = 'on';
```

B-7-14

```
% Define transfer function
num = [1 2 1];
den = [1 0.2 1 1];
G = tf(num,den);

% Find zeros, poles, and corner frequencies
zrs = roots(num);
pls = roots(den);
cornFreq = corner_freq(num,den);

% Bode Plot
fig = figure("Renderer","painters");
    opts_bd.Title.String = "B-7-14 Bode Diagram, Koike";
    bd = bodeplot(G,opts_bd);
    opt = getoptions(bd);
saveas(fig,fullfile(fdir,"B-7-14_bode.png"));

% Sample out points from Bode plot
res = bode_sample_points(G,[0.05,0.4,1,4,10]);
writetable(res,fullfile(fdir,"B-7-14.xls"),"WriteMode","overwrite");

% Nyquist Plot
fig = figure("Renderer","painters");
    opts_nq.Title.String = "B-7-14 Nyquist Diagram, Koike";
    nyquistplot(G,opts_nq);
saveas(fig,fullfile(fdir,"B-7-14_nyquist.png"));
```

```
% Calculate the intersection with the real axis
[Wpc,intrsct_Re] = phase_crossover(num,den);
```

B-7-19

```
% Define transfer function
num = [0 2];
den = conv([1 0],[1 1]); den = conv(den,[1 2]);
G = tf(num,den);

% Find zeros, poles, and corner frequencies
zrs = roots(num);
pls = roots(den);
cornFreq = corner_freq(num,den);

% Bode Plot
fig = figure("Renderer","painters");
    opts_bd.Title.String = "B-7-19 Bode Diagram, Koike";
    bd = bodeplot(G,opts_bd);
    opt = getoptions(bd);
saveas(fig,fullfile(fdir,"B-7-19_bode.png"));

% Sample out points from Bode plot
res = bode_sample_points(G,[0.05,0.2,0.7,1,4,10,15]);
writetable(res,fullfile(fdir,"B-7-19.xls"),"WriteMode","overwritesheet");

% Nyquist Plot
fig = figure("Renderer","painters");
    opts_nq.Title.String = "B-7-19 Nyquist Diagram, Koike";
    nyquistplot(G,opts_nq);
saveas(fig,fullfile(fdir,"B-7-19_nyquist.png"));

% Nyquist Plot 2
G1 = zpk([],[0, -1, -2],2); % Negative Feedback
G2 = zpk([],[0, -1, -2],-2); % Positive Feedback
fig = figure("Renderer","painters");
    hold on; grid on;
    opts_nq.Title.String = "B-7-19 Nyquist Diagram with Negative and Positive
Feedback, Koike";
    nyquistplot(G1,opts_nq, 'blue-')
    nyquistplot(G2,opts_nq, 'red--')
    legend('Negative-feedback', 'Positive-feedback')
    axis equal; xlim([-8,8]);ylim([-8,8]);
saveas(fig,fullfile(fdir,"B-7-19_nyquist2.png"));
```

B-7-24

```
% Define transfer function
num = [0 25];
den = conv([1 0],[1 1]); den = conv(den,[1 10]);
G = tf(num,den);

% Find zeros, poles, and corner frequencies
zrs = roots(num);
```

```
pls = roots(den);
cornFreq = corner_freq(num,den);

% Bode Plot
fig = figure("Renderer","painters");
    opts_bd.Title.String = "B-7-24 Bode Diagram, Koike";
    bd = bodeplot(G,opts_bd);
    opt = getoptions(bd);
saveas(fig,fullfile(fdir,"B-7-24_bode.png"));

% Phase Margin
[Wgc,phi] = gain_crossover(num,den);
PM = 180 + phi;

% Gain Margin
[Wpc,Kg_inv] = phase_crossover(num,den);
GM = 1/abs(Kg_inv);
GM_dB = 20*log10(1/abs(Kg_inv));  % [dB]

% Validate with builtin function
[Gm_v,Pm_v,Wgc_v,Wpc_v] = margin(G);
fprintf("The actual values for relative stability analysis.");
fprintf("Gain Margin: %.4f at gain crossover frequency of %.4f",Gm_v,Wgc_v);
fprintf("Phase Margin: %.4f at phase crossover frequency of %.4f\n",Pm_v,Wpc_v);
```

B-7-26

```
% From phase margin
syms w
eqn = w == (4 - w^2)*tand(40);
Wgc = double(solve(eqn,w));
Wgc = Wgc(Wgc>0);
K = sqrt(Wgc^6 - 7*Wgc^4 + 16*Wgc^2);

% Define transfer function
num = [0 K];
den = conv([1 0],[1 1 4]);
G = tf(num,den);

% Gain margin
[Wpc,Kg_inv] = phase_crossover(num,den);
GM = 1/abs(Kg_inv);
GM_dB = 20*log10(GM); % [dB]
```

P3 Aircraft Example

```
% Define transfer function
num = [1.1057 -0.19];
den = [1 17.95 123.3 366.3 112.2 0];
G = tf(num,den);

pls = roots(den)
zrs = roots(num)
cornFreq = corner_freq(num,den)

fig = figure("Renderer","painters");
    opts_bd.Title.String = "Aircraft Example Bode Diagram, Koike";
```

```
    bodeplot(G,opts_bd);
saveas(fig,fullfile(fdir,"P3_bode.png"));
```

```
% Sample out points from Bode plot
res = bode_sample_points(G,[0.01,0.1,1,10,50]);
writetable(res,fullfile(fdir,"P3.xls"),"WriteMode","overwritesheet");

% Nyquist Plot
fig = figure("Renderer","painters");
    opts_nq.Title.String = "Aircraft Example Nyquist Diagram, Koike";
    nyquistplot(G,opts_nq);
    xlim([-0.001 0.015])
saveas(fig,fullfile(fdir,"P3_nyquist.png"));
```

P4 Spacecraft Example

```
num = 0.036*[1 25];
den = [1 0.04 1 0 0];

pls = roots(den);
zrs = roots(num);
cornFreq = corner_freq(num,den);

G = tf(num,den);
fig = figure("Renderer","painters");
    opts_bd.Title.String = "Spacecraft Example Bode Diagram, Koike";
    bodeplot(G,opts_bd);
saveas(fig,fullfile(fdir,"P4_bode.png"));
```

```
% Sample out points from Bode plot
res = bode_sample_points(G,[0.2,0.7,1,2,10,100,600]);
writetable(res,fullfile(fdir,"P4.xls"),"WriteMode","overwritesheet");

% Nyquist Plot
fig = figure("Renderer","painters");
    opts_nq.Title.String = "Aircraft Example Nyquist Diagram, Koike";
    nyquistplot(G,opts_nq);
saveas(fig,fullfile(fdir,"P4_nyquist.png"));
```

```
function w_i = corner_freq(num,den)
    %{
      Function:    corner_freq()
      Author:      Tomoki Koike
      Description: Computes the corner frequencies for a Bode Plot.
      >>Inputs
          num: the numerator of the open-loop transfer function
          den: the denominator of the open-loop transfer function
      Outputs<<
          w_i: the table with the corner frequencies for poles and zeros
    %}

    pls = roots(den);
    zrs = roots(num);
    cornP = unique(abs(pls));
    cornZ = unique(abs(zrs));
    if length(cornP) > length(cornZ)
```

```
        cornZ = [cornZ; NaN([(length(cornP) - length(cornZ)), 1])];
    else
        cornP = [cornP; NaN([(length(cornZ) - length(cornP)), 1])];
    end
    w_i = array2table([cornP, cornZ],"VariableNames",{'Poles','Zeros'});
end
```

```
function res_T = bode_sample_points(sys,samp)
    %{
      Function:    bode_sample_points()
      Author:      Tomoki Koike
      Description: Finds magnitude and phase points corresponding to the
                   sample points provided as the user input.
      >>Inputs
          sys:   the system/transfer function
          samp:  the sample frequency points
      Outputs<<
          res_T: the table with all the results: points, frequencies, phase
                 angles in degrees, magnitudes in dB, and magnitudes.
    %}

    % Call the bode function to obtain required data points
    omg = logspace(-2,3,1000000);
    [mag,phase,w] = bode(sys,omg);
    mag = mag(:); phase = phase(:);

    % Find corresponding values for sample frequencies using 1D data
    % interpolation
    mag_pt = interp1(w,mag,samp);
    phase_pt = interp1(w,phase,samp);
    magdB_pt = -20*log10(mag_pt);

    % check if samp vector is row and if it is a row vector transpose to make it a
    % column vector
    if isrow(samp)
        samp = samp.';
    end

    % Construct array with results
    arr = [samp, phase_pt.', magdB_pt.', mag_pt.'];
    res_T = array2table(arr,"VariableNames",{'Frequencies','Phase','MagdB','Mag'});
end
```

```
function [Wpc,res] = phase_crossover(num,den)
    %{
      Function:    phase_crossover()
      Author:      Tomoki Koike
      Description: Computes the intersection of the real axis and the
                   Nyquist plot.
      >>Inputs
          num: the numerator of the open-loop transfer function
          den: the denominator of the open-loop transfer function
      Outputs<<
          Wpc: phase crossover frequency
          res: intersections/real number of the phase crossover point
```

```
    %}

    % Get the length of each numerator and denominator
    num_len = length(num);
    den_len = length(den);

    % Preset a array with the order of magnitudes (i.e. s^3, s^2, s^1, s^0)
    % corresponding to the numerator and denominator
    O_num = (num_len-1):-1:0;
    O_den = (den_len-1):-1:0;

    % Define a system equation to find phase crossover
    w = sym('w');
    assume(w,'real');
    N = dot(num,(w*(1j)).^O_num);
    D = dot(den,(w*(1j)).^O_den);
    fprintf('The denominator factored out.\n'); disp(vpa(D,6));
    NUM = N*conj(D);
    fprintf('The numerator combined with denominator.\n'); disp(vpa(NUM,6));
    NUM_im = imag(NUM);
    fprintf('The imaginary part you need to equate to 0.\n'); disp(vpa(NUM_im,6));
    eqn = NUM_im == 0;
    Wpc = double(solve(eqn,w));
    fprintf('Omegas that make the imaginary part equal 0 (phase crossover
frequency).\n'); disp(Wpc);
    G = N/D;
    Wpc = Wpc(Wpc~=0 & Wpc>0);  % Take out 0 and negative values
    res = double(subs(G,w,Wpc));
end

function [Wgc,phi] = gain_crossover(num,den)
    %{
      Function:    gain_crossover()
      Author:      Tomoki Koike
      Description: Computes the intersection of the real axis and the
                   Nyquist plot.
      >>Inputs
          num: the numerator of the open-loop transfer function
          den: the denominator of the open-loop transfer function
      Outputs<<
          res: gain crossover frequency
          phi: angle corresponding to the gain crossover frequency (deg)
    %}

    % Get the length of each numerator and denominator
    num_len = length(num);
    den_len = length(den);

    % Preset a array with the order of magnitudes (i.e. s^3, s^2, s^1, s^0)
    % corresponding to the numerator and denominator
    O_num = (num_len-1):-1:0;
    O_den = (den_len-1):-1:0;

    % Define a system equation to find the gain crossover frequency
    w = sym('w');
```

```
    assume(w,'real');
    N = dot(num,(w*(1j)).^O_num);
    D = dot(den,(w*(1j)).^O_den);
    eqn = (abs(N))^2 == (abs(D))^2;
    fprintf('Equation of |G(jw)| = 1.\n'); disp(eqn);
    Wgc = double(solve(eqn,w));
    % Filter result to be one value
    Wgc = unique(abs(Wgc));

    % Compute corresponding angle from the found frequency
    G = tf(num,den);
    s = Wgc*1i;
    phi = rad2deg(angle(evalfr(G,s)));
end
```