

Table of Contents

- 1 Challenge
- 2 Relative stability using the Bode plot
 - 2.1 Gain Margin
 - 2.1.1 Why does this work?
 - 2.2 Gain Margin: Example
 - 2.3 Phase Margin
 - 2.3.1 Why?
 - 2.4 Phase Margin: Example
 - 2.4.1 OK but my root locus gain is not 1... now what?
- 3 Using gain and phase margin to assess the robustness of an existing control system design
- 4 Rules of thumb for gain and phase margins in a control system design
- 5 Exercises: Due FRIDAY, November 20 before class
- 6 Problem 1
 - 6.1 Problem 2
 - 6.2 Problem 3

Challenge

In today's challenge, we would like to use what we have learned about Bode plots in the last few assignments to help us calculate the *relative stability* of a control system design. The concept of *relative stability* is not actually totally new to us!

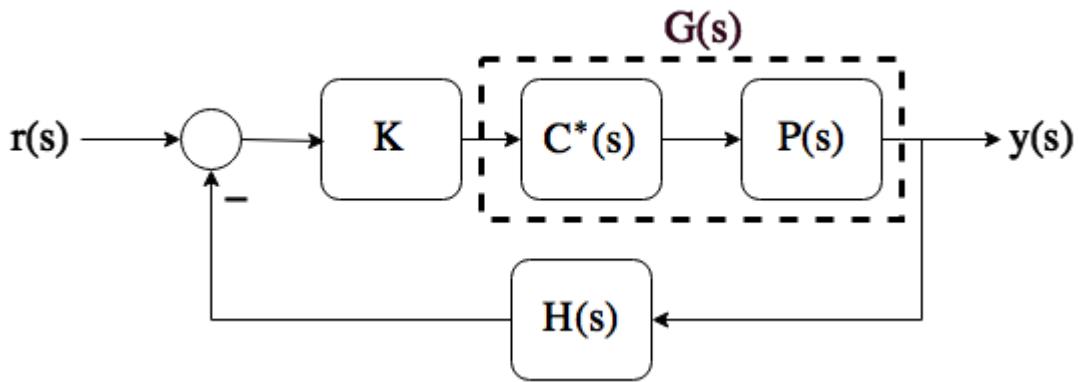
We have learned through our study of the root locus that a stable plant transfer function can become unstable under the direction of a feedback controller if the overall gain K is too high--we called the maximum possible gain allowable for the system to remain stable the *critical gain*. It is also known by another name: the "gain margin." This is one useful measure of relative stability.

However, you have probably found in your Zumo projects that unmodeled dynamics in our plant transfer function can change the landscape of the root locus, and cause instability *even when our model did not predict it*.

To understand how to predict how risky our control system design might be in the face of unmodeled plant ($P(s)$) or sensor ($H(s)$) dynamics, we can use another measure of relative stability called the "phase margin." Both the gain margin and the phase margin are easy to read off of a plant transfer function's Bode plot. To learn how, read on!

Relative stability using the Bode plot

Over the last few weeks, we have talked extensively about a feedback control system in the following, standard form:



This system has the following closed-loop transfer function:

$$G_{cl}(s) = \frac{y(s)}{r(s)} = \frac{KC^*P(s)}{1+KG(s)H(s)}$$

This system's characteristic equation is $1+KG(s)H(s)=0$, and this equation is what dictates the system's eigenvalues (or poles). We have studied how varying the "loop gain" K changes the location of the system's poles by using the **root locus**. One of the things we calculated as we sketched the root locus for a system was the "critical gain" that would make the system go unstable. This critical gain is often thought of as a measure of "relative stability," or how close a system is to being unstable.

This process was somewhat involved, because we had to solve the characteristic equation's real and imaginary parts to find both the "crossing frequency" ω_c and the "critical gain" K_{crit} . It turns out that the Bode plot for the "open loop transfer function" $G(s)H(s)$ allows us to find the same quantity with much less effort. In particular, the Bode plot is also used to find the **gain margin** (which is the same quantity as the "critical gain") and **phase margin** for a system under proportional feedback control. Just as before, if we have a system under PD, PI, or PID control, we can set $G(s)=C^*P(s)$ to look at the effect of varying only one gain on the system's poles and relative stability.

Gain Margin

The gain margin for an "open loop transfer function" is defined as the *maximum value of the loop gain K* that is allowable before the system will go unstable.

To calculate the "gain margin" (critical gain) using the frequency response and/or Bode plot:

1. Calculate the crossing frequency ω_c at which $\phi = \angle(G(j\omega)H(j\omega)) = -180^\circ$ using the phase equation $\phi(\omega)$ or by looking at the Bode phase plot.
2. Find the "open loop transfer function's magnitude $|G(j\omega)H(j\omega)|$ at this frequency. The **gain margin** or **critical gain** of the system $G(j\omega)H(j\omega)$ is $GM = |G(j\omega)H(j\omega)|_{\omega_c}^{-1}$.
3. If you are looking at a Bode plot, and find the crossing frequency ω_c visually, you will see the gain at this frequency in decibels as $G_{dB} = 20\log_{10}|GH|$. The gain margin can be calculated by finding the gain margin in decibels as: $GM_{dB} = 0 - |G_{dB}|_{\omega_c}$. Then, the gain margin GM is found by converting GM_{dB} from decibels into a raw magnitude ratio.

If the system never reaches $\phi=-180^\circ$, it has "infinite gain margin." This is something we could confirm using the root locus, but the Bode plot (really, the frequency response equations) makes finding out whether (and when) a system under feedback control might become unstable fairly quick.

Why does this work?

This is kind of strange to think about, because we have been looking at Bode plots as ways to understand a system's response to steady-state sinusoidal inputs... Stability margins actually have **nothing** to do with the type of input the system receives. So why are the frequency response equations useful here?

We can reason through why the procedure above for finding the gain margin works by remembering that the characteristic equation for our system, $1+KGH=0$, can be written as:

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

This means that if a point is on the root locus, the "open loop transfer function" $G(s)H(s)$ will have an angle of $\pm 180^\circ$, or in other words, the angle criterion of the root locus will be satisfied. This means that when we substituted $s=j\omega_c$ into our characteristic equation to find where the locus crossed the imaginary axis, we were unwittingly finding out where the angle of $G(j\omega)H(j\omega)$ was itself equal to 180° . At this frequency, solving the magnitude criterion for the root locus would give us the gain required to cause closed loop system instability. Because we are already substituting $s=j\omega$ in the calculation of GH 's frequency response, the gain margin procedure is simply a matter of following the steps above.

Gain Margin: Example

Find the gain margin for an "open loop" transfer function $G(s)H(s) = \frac{1}{(s+1)(s+2)(s+3)}$.

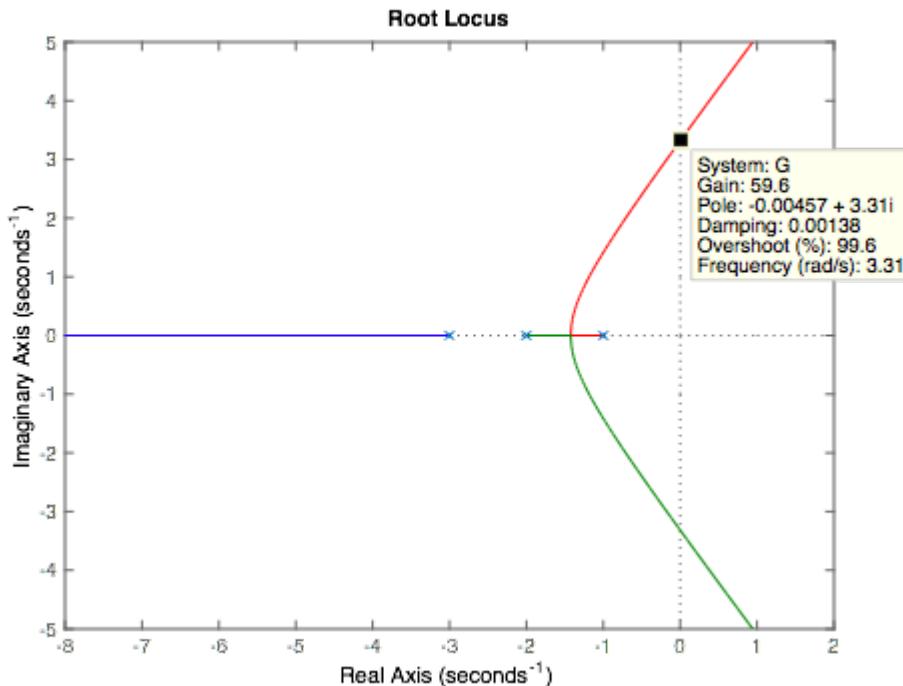
First, we write the magnitude and phase equations for $G(s)H(s)$.

$$G(j\omega)H(j\omega) = \frac{1}{(j\omega)^3 + 6(j\omega)^2 + 11(j\omega) + 6} = \frac{1}{(-\omega^3 + 11\omega) + j(6\omega^2 + 11\omega)} \quad \text{phi} = -\tan^{-1}\left(\frac{-\omega^3 + 11\omega}{6\omega^2 + 11\omega}\right) \quad |GH| = \sqrt{(-\omega^3 + 11\omega)^2 + (6\omega^2 + 11\omega)^2}$$

Looking at the phase equation phi , we can see that since $\tan 180^\circ = 0$, we need to make the numerator of the fraction inside the inverse tangent function equal to zero. Therefore, we must find where $-\omega^3 + 11\omega = 0$. This equation is satisfied when $\omega=0$ and when $\omega=\sqrt[3]{11}=3.32\text{rad/s}$. Because the phase angle is actually 0° when $\omega=0$, we know that this is not a valid solution (the denominator of the fraction must be a negative real number for $\text{phi}=180^\circ$). Therefore, we know that calculating the gain margin means calculating $|GH|$ when $\omega=3.32\text{rad/s}$. Using the magnitude equation above, we find that:

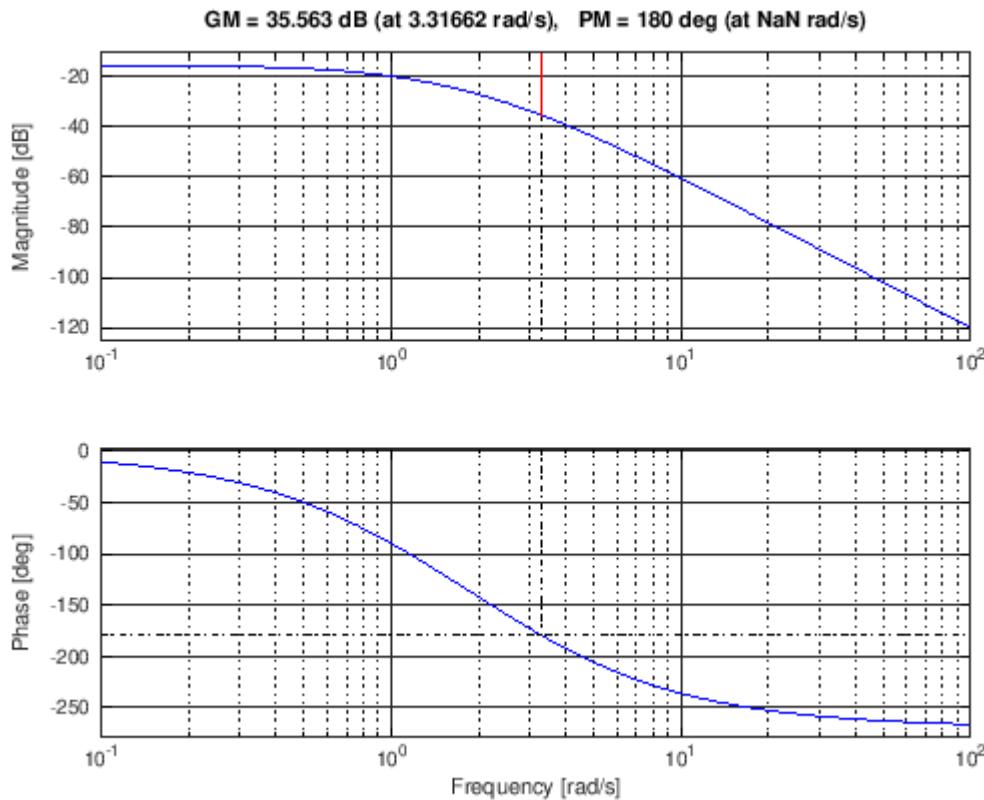
$$|GH|_{\omega=3.32} = GM^{-1} = .0166$$

This means that the system's gain margin is $GM = 60.24$. So we could put the system $G(s)H(s)$ under proportional control with a gain of up to 60.24 before the system would go unstable. This is easily confirmed using the root locus functionality in MATLAB as shown:



It is also common to find the gain margin using a Bode plot directly. MATLAB's "margin" command does this automatically, as shown below:

```
In [1]: clear all
s = tf('s');
GH = 1/((s+1)*(s+2)*(s+3));
margin(GH)
```



Notice the dotted lines that indicate where the phase plot crosses $\phi = -180^\circ$. Then, notice the red line that shows the system's magnitude ratio in decibels at this frequency. The plot title gives us precise numbers: The gain margin of this system in decibels is $GM_{\text{dB}} = 35.563$. To find the gain margin as a simple number, we convert:

$$GM = 10^{\frac{GM_{\text{dB}}}{20}} = 10^{\frac{35.563}{20}} = 59.998$$

This gives us the same result as before, to a reasonable degree of round-off error. Pretty cool!

You'll also notice that the "margin" command gives us a value for something called the "phase margin," which is another measure of a system's relative stability. We'll discuss phase margin next.

Phase Margin

The phase margin for an "open loop transfer function" is defined as the amount of *phase angle* that could be added or subtracted to a system under *unity feedback* (this means to close the loop around GH with $K=1$) before the system becomes unstable. To find the system's phase margin:

1. Find the frequency ω_0 where the magnitude ratio of $G(j\omega)H(j\omega)=1$, or where $G_{dB} = 0$ on the Bode magnitude plot.
2. The phase margin $PM = 180^\circ + \phi(\omega_0)$.

Why?

We can explore why the math behind the phase margin works out using the tools that we have. Remembering that when computing the system's frequency response we are substituting $s=j\omega$, we know that the closed loop characteristic equation can be written as:

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

The phase margin is defined for the case when $K=1$, which makes the characteristic equation:

$$-1 = G(s)H(s)$$

so if $s=j\omega_0$ is a solution to this form of the characteristic equation (that is, if the system is on the verge of becoming unstable with $K=1$), $G(j\omega_0)H(j\omega_0)$ must have a total angle of $\pm 180^\circ$. Thus, you can think of the phase margin as an "angle deficiency" of sorts that tells you how much angle must be added or subtracted to $\angle(G(j\omega)H(j\omega))$ in order to satisfy the angle criterion (for a point on the s-plane that borders on unstable!!).

Unlike satisfying the angle criterion for a controller design, here we are assuming that we know that at some $s = j\omega_0$, which represents the system's (marginally stable) eigenvalue at a loop gain $K=1$, the system would be on the verge of instability... so satisfying the angle criterion is a bad thing! But it gives us a relative measure of "how stable" our system is for a feedback system with unity gain.

So how, you might ask, could you "add phase" or "subtract phase" to a system $G(s)H(s)$? The most common reason for adding or subtracting phase angle from a system is accidental.

Modeling errors can often manifest as phase changes. Remember that when you evaluate the angle $\angle(GH)$, an extra pole will add negative angle (making the phase margin smaller)... the closer the "extra pole" is to the origin, the smaller the phase margin will be. This goes hand in hand with the concept of "dominant poles" we discussed earlier in the course! If an unmodeled (ignored) pole of $G(s)H(s)$ was very fast, it would not detract much from the phase margin.

Phase Margin: Example

The system in our first example actually has an infinite phase margin, because the system never has a magnitude ratio or \$1\$, or \$0dB\$. Another way to think about this is that for our original system under feedback with \$K=1\$, no pole at any (stable) location in the s-plane would change the phase of GH enough to cause instability. Let's amend the example we used before so that we get a non-infinite phase margin. Let's say that our "open loop transfer function" is now:

$$G(s)H(s) = \frac{30}{(s+1)(s+2)(s+3)}$$

This means that the frequency response equation is given by:

$$G(j\omega)H(j\omega) = \frac{30}{(j\omega)^3 + 6(j\omega)^2 + 11(j\omega) + 6} = \frac{1}{(-\omega^3 + 11\omega)j + 6(1 - \omega^2)}$$

Looking at the magnitude equation, we are looking for a frequency \$\omega=\omega_0\$ that causes the magnitude ratio to be \$1\$.

$$|GH| = \sqrt{(-\omega_0^3 + 11\omega_0)^2 + (6(1 - \omega_0^2))^2} = 1$$

Solving for \$\omega_0\$ here gives:

$$30^2 = (-\omega_0^3 + 11\omega_0)^2 + (6(1 - \omega_0^2))^2$$

Solving this equation via a calculator with a symbolic algebra system or with MATLAB's symbolic toolbox yields 6 solutions, but only one of them is both real and positive. Therefore, \$\omega_0=2.3486\$ rad/s, which is the frequency we will use to determine our phase margin. Substituting \$\omega=\omega_0\$ into the phase equation for \$G(s)H(s)\$, we get:

$$\phi = -\tan^{-1}\left(\frac{-\omega_0^3 + 11\omega_0}{6(1 - \omega_0^2)}\right) = -154.5758^\circ$$

This gives us a phase margin of:

$$PM = 180^\circ + \phi(\omega_0) = 25.42^\circ$$

This means that with unity feedback on the "open loop" transfer function \$GH\$, we could have ignored a pole that contributed up to \$25.42^\circ\$ of negative phase to the overall transfer function angle \$\angle(GH)|_{s=j\omega_0}\$ without making the system unstable, *assuming the system's magnitude \$|G(s)H(s)|\$ does not change when this pole is added*. For our example system, such a pole would lie at \$s\approx-5\$. This is shown in the MATLAB cell below, where we compare the original plant under feedback with \$K=1\$ to the plant with an unmodeled pole around \$s=-5\$. Take note that this pole is really, really close to the other three! In other words, the separation principle would **not** have applied here... This system \$G(s)H(s) = \frac{30}{(s+1)(s+2)(s+3)}\$ is therefore very "safe" in terms of phase margin.

```

In ... clear all
s = tf('s');
GH = 30.00/((s+1)*(s+2)*(s+3));

K = 1;

Gcl = minreal(GH/(1+GH*K));
[num,den] = tfdata(Gcl,'v');
poles_without_unmodeled_dynamics = roots(den)
%phase margin is 25.43 degrees... let's add a pole that subtracts 25 deg
%height is 2.35j
%tan(extra_phase) = 2.35j/L so L = 2.35/tand(25)
%then "extra pole" is at s=-L. This extra pole must also NOT change
%the absolute value of GH at the crossover frequency, or else it is changing
%does not allow us to see the effects of unmodeled dynamics changing PHA

%to ensure this, we can state that the unmodeled dynamics will have the -A/(s+L) with L our unmodeled pole. We can find this using trigonometry!
L = 2.34856/tand(25)
%and at s = 2.346j, or the omega_naught of the original system, we must have
%therefore, we can state that A = abs(s+L) = abs(2.346j+L)
A = abs(2.34856+L)
%now, we can construct a transfer function representing our unmodeled dynamics
Punmodeled = A/(s+L);%ALMOST!!
%now, we can calculate the closed loop TF with the unmodeled dynamics
GH_unstable = GH*Punmodeled;
Gcl_unstable = minreal(GH_unstable/(1+K*GH_unstable));
%now let's look at the eigenvalues of this "almost unstable" system with
[num,den] = tfdata(Gcl_unstable,'v');
poles_unstable = roots(den)
%finally, let's compare the step responses of the system with and without
step(Gcl,Gcl_unstable)
xlim([0 10])
ylim([-5 5])

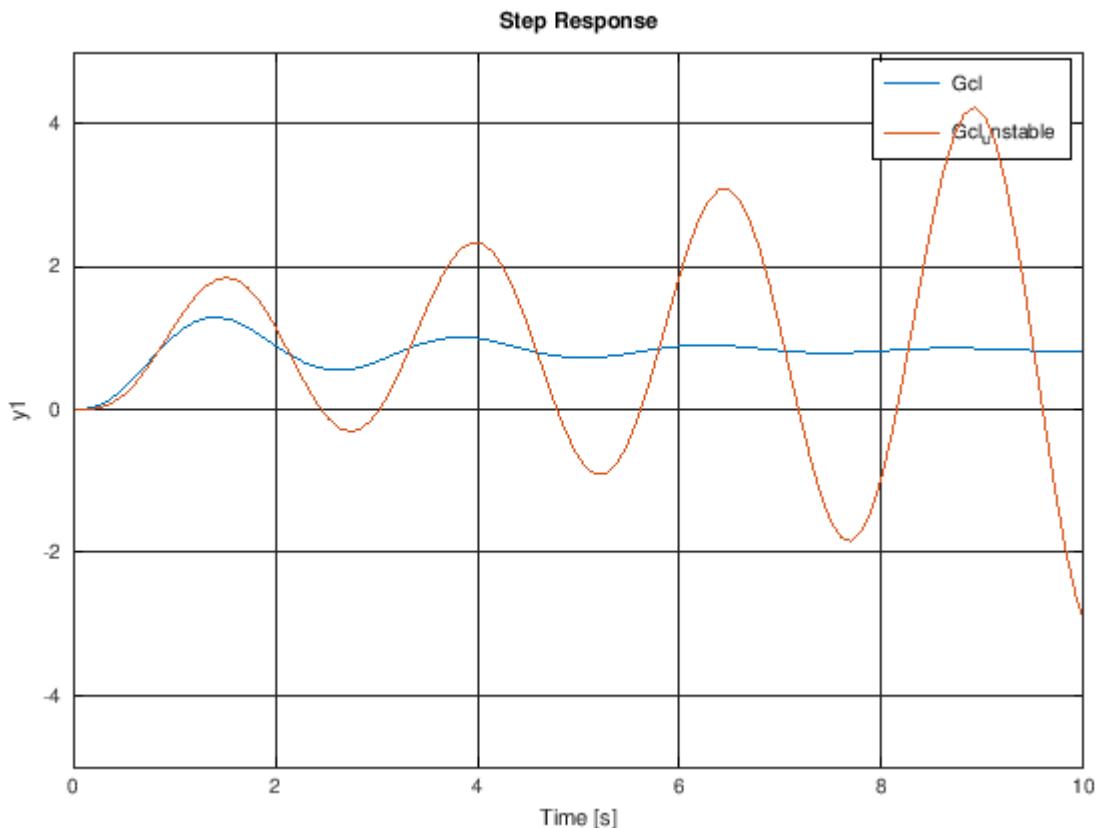
```

```

poles_without_unmodeled_dynamics =
-5.21447 + 0.00000i
-0.39277 + 2.59800i
-0.39277 - 2.59800i

L = 5.0365
A = 7.3851
poles_unstable =
-5.6865 + 2.4842i
-5.6865 - 2.4842i
0.1682 + 2.5514i
0.1682 - 2.5514i

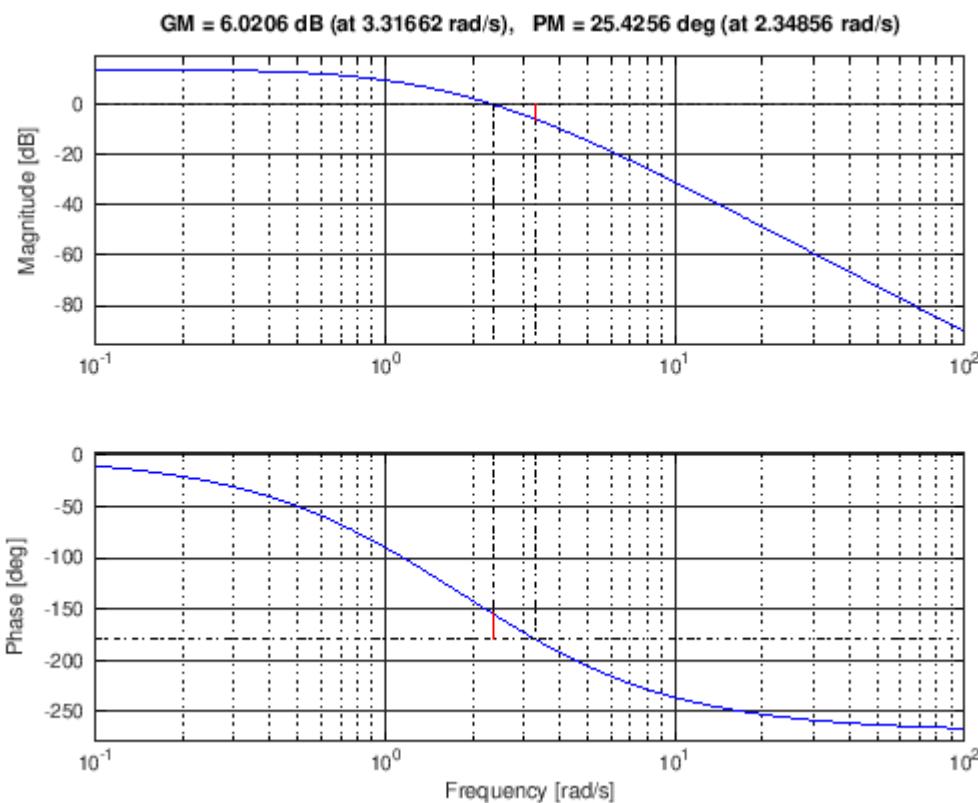
```



Notice that after the unmodeled dynamics (pole at $s=-5.04$) is added, the closed loop system becomes very oscillatory, with a pair of complex conjugate poles almost directly on the imaginary axis (but on the RIGHT hand side). This system is now unstable with $K=1$. Why isn't the eigenvalue directly at $s = 2.34856j$ as it should be? Round-off error!

As before, MATLAB/Octave will provide you with phase and gain margins directly using the "margin" command, which basically finds the margins visually using the Bode plot itself. To find the phase margin using the Bode plot, you simply find the frequency ω_0 where $G_{dB}=0$ for the "open loop transfer function" $G(s)H(s)$, compute the phase $\phi(\omega_0)$ by dropping this line down to the phase plot, and then compute the phase margin $PM=180^\circ + \phi(\omega_0)$.

```
In [3]: clear all
s = tf('s');
GH = 30.00/((s+1)*(s+2)*(s+3));
margin(GH)
```



OK but my root locus gain is not 1... now what?

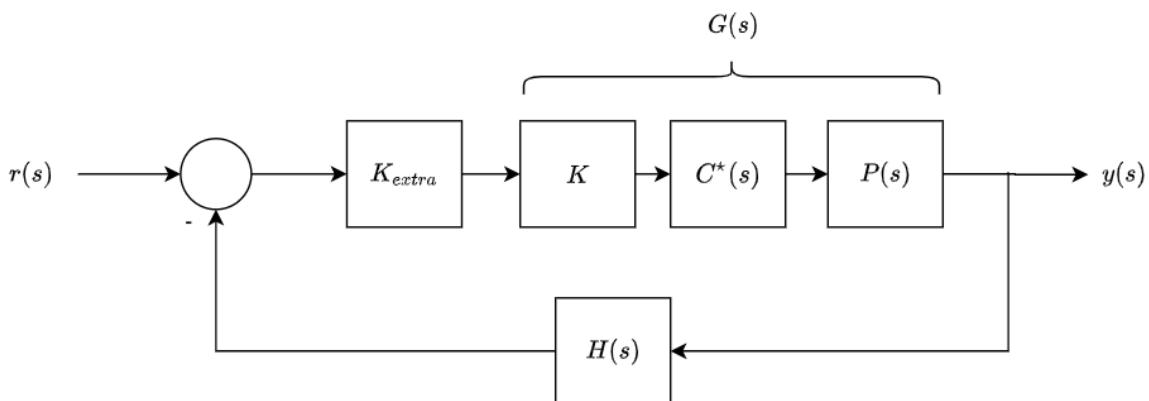
If you wish to find the "phase margin" for *your control system*, which has some root locus gain \$K\$ that is not \$1\$, you can roll this \$K\$ value directly into the phase margin's \$GH\$ transfer function, and then calculate the phase margin as usual, but you will be finding the phase margin for the transfer function \$KG(s)H(s)\$ rather than just \$G(s)H(s)\$. This is because you could write the equation: \$-1 = KG(s)H(s)\$ and use the phase margin procedure outlined above, "tricking" the procedure into thinking you are using an "overall gain" of 1. This idea is explored in more detail in the next section.

Using gain and phase margin to assess the robustness of an existing control system design

Sometimes, we would like to understand how much "extra" gain or "extra" phase our system could handle *after we have designed a controller*.

The concepts of gain margin and phase margin are not necessarily set up for this. The gain margin's original formulation only gives us the maximum root locus gain K our system's open loop transfer function $G(s)H(s)$ can handle, and the phase margin assumes $K=1$. This means that these measures of relative stability can be computed before a control design is complete.

However, we can extend these concepts to understand how close we are to trouble **after we have already made a pass through the design process**. To do this, we can look at a slightly modified version of our canonical block diagram, which includes some "extra" gain that might represent error in our design or in our plant model:



This block diagram has the closed loop transfer function:

$$\frac{y(s)}{r(s)} = \frac{K_{extra}KC^*P(s)}{1 + K_{extra}KC^*P(s)H(s)}$$

Which has the characteristic equation:

$$1 + K_{extra}KC^*P(s)H(s) = 0$$

This can be re-written as:

$$\frac{-1}{K_{extra}} = G(s)H(s) = KC^*P(s)H(s)$$

Here, K is our original, design root locus gain. K_{extra} represents some kind of scaling change in the forward path, either from a mistake in our calculation of $P(s)$ or $H(s)$, or because we may have changed our root locus gain K to be something other than what we originally chose in our design.

This approach allows us to find the gain margin that represents the "extra" root locus gain that could be added to the system before it goes unstable. Similarly, this relieves us of the strange

convention that phase margin is always calculated at $K=1$ by allowing us to set $K_{\text{extra}} = 1$, so that we can see how sensitive our design is to errors in the phase of $G(s)H(s)$ using our design's (non-unity) root locus gain K .

Rules of thumb for gain and phase margins in a control system design

So how much gain and phase margin should we shoot for in our designs?

[Kulakowski, Gardner, and Shearer](#) report in Chapter 13 of their book "Dynamic Modeling and Control of Engineering Systems" that **minimum** "safe" gain margins are in the range of 1.2-1.5 ($1.6\text{dB}-3.5\text{dB}$), and **minimum** "safe" phase margins are in the neighborhood of $30^{\circ}\text{-}45^{\circ}$. If we calculate gain and phase margins using the modified formulation above, the gain margin rule says we should aim to be able to increase our design root locus gain by between 20% and 50% without risk of instability. We should also shoot for a phase margin that allows for an error in phase that contributes -30 degrees to $G(s)H(s)$ at $s=j\omega_0$.

These rules of thumb are fairly widely accepted in the control systems community, so it's something to keep in mind if you're designing a control system: check the gain and phase margins of your "open loop transfer function" $G(s)H(s)=K C^{\star}(s) P(s) H(s)$ to find out how much "extra" gain and phase your system could handle, and use caution if your margins fall outside of these recommended values. Infinite gain and phase margins are, of course, acceptable!

Exercises: Due FRIDAY, November 20 before class

Answer the following questions:

Problem 1

Find the gain and phase margins for the following system under proportional control by hand. Show any and all hand calculations, and then use MATLAB/Octave to check using the "margin" command. You may print or screenshot a Bode plot to aid you in your "hand calculations" if you like.

$$P(s) = \frac{1000}{(s+10)(s^2+2s+10)}$$

STUDENT ANSWER: 10 POINTS POSSIBLE

Week 14 Monday

Problem 1

$$P(s) = \frac{1000}{(s+10)(s^2+2s+10)}$$

finding gain margin...

$$GH(j\omega) = \frac{1000}{((j\omega)+10)((j\omega)^2+2(j\omega)+10)} = \frac{1000}{(-\omega^3+30\omega)j+(-12\omega^2+100)}$$

$$\phi = -\tan^{-1}\left(\frac{-\omega^3+30\omega}{-12\omega^2+100}\right) \rightarrow -\omega^3+30\omega=0 \rightarrow \omega = 0, \sqrt{30}, \sqrt{30}$$

$|GH|$ when $\omega = \sqrt{30}$ rad/s...

$$|GH|_{\omega=\sqrt{30}} = GM^{-1} = 3.8462 \rightarrow GM = 0.26$$

$$GM = 10 \frac{GM_{dB}}{20} \rightarrow GM_{dB} = -11.7 dB$$

finding phase margin... (for K=1)

$$|GH| = \frac{1000}{\sqrt{(-\omega_0^3+30\omega_0)^2 + (-12\omega_0^2+100)^2}} = 1$$

$$1000^2 = (-\omega_0^3+30\omega_0)^2 + (-12\omega_0^2+100)^2 \rightarrow \omega_0 = 9.05 \text{ rad/s}$$

$$\phi = -\tan^{-1}\left(\frac{-\omega_0^3+30\omega_0}{-12\omega_0^2+100}\right) = 151.98^\circ$$

$$PM = 180 + \phi = 180 + (151.98^\circ) \Rightarrow PM = 331.98^\circ$$

STUDENT ANSWER: 10 POINTS POSSIBLE

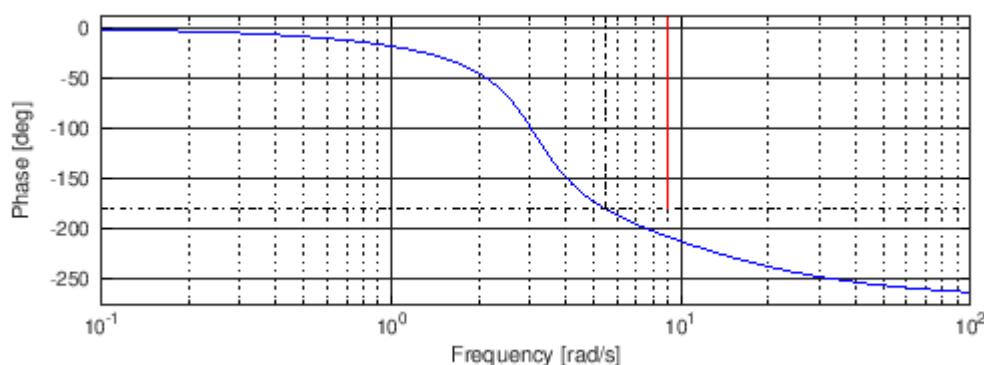
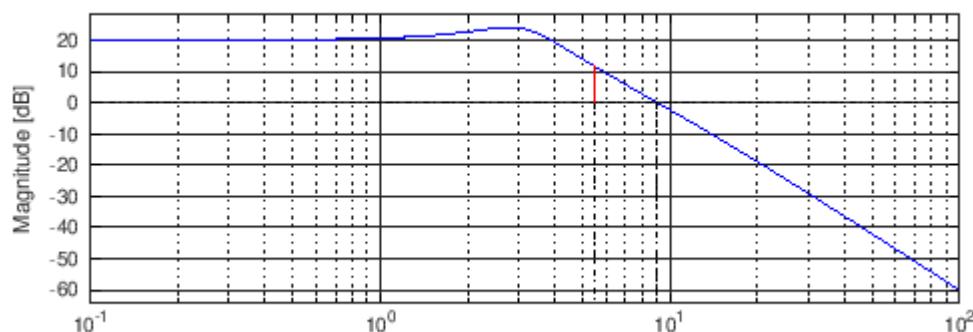
```
In [21]: % gain margin calculations...
w = sqrt(30);
GH_mag = abs(1000/((w*1j+10)*((w*1j)^2+2*w*1j+10)));
GM = 1/GH_mag

% phase margin calculations...
w0 = 9.04996;
phi = -atan2d(-w0^3+30*w0,-12*w0^2+100)
PM = 180+phi

% checking answers...
s = tf('s');
GH = 1000/((s+10)*(s^2+2*s+10));
margin(GH)

GM = 0.26000
phi = 151.98
PM = 331.98
```

GM = -11.7005 dB (at 5.47723 rad/s), PM = 331.985 deg (at 9.04996 rad/s)



Problem 2

Find the gain and phase margins for the same plant transfer function above, but under proportional-integral control with a control zero at $s=-20$. Show any and all hand calculations, and then use MATLAB/Octave to check using the "margin" command. You may print or screenshot a Bode plot to aid you in your "hand calculations" if you like.

STUDENT ANSWER: 10 POINTS POSSIBLE

Problem 2

under PI control...

$$P(s) = \frac{1000(s+20)}{s(s+10)(s^2+2s+10)}$$

finding gain margin...

$$GH(j\omega) = \frac{1000 (j\omega + 20)}{(j\omega)((j\omega) + 10)((j\omega)^2 + 2(j\omega) + 10)} \cdot \frac{(j\omega - 20)}{(j\omega - 20)}$$
$$= \frac{1000 (-\omega^2 - 400)}{(\omega^5 + 210\omega^3 - 2000\omega)j + (-8\omega^4 + 500\omega^2)}$$

$$\phi = -\tan^{-1} \left(\frac{\omega^5 + 210\omega^3 - 2000\omega}{-8\omega^4 + 500\omega^2} \right)$$

$$\rightarrow \omega^5 + 210\omega^3 - 2000\omega = 0 \rightarrow \omega = 0, -3.02, 3.02$$

$|GH|$ when $\omega = 3.02$ rad/s ...

$$|GH|_{\omega=3.02} = GM^{-1} = 104.98 \rightarrow GM = 0.00953$$

$$GM = 10^{\frac{GM_{dB}}{20}} \rightarrow GM_{dB} = -40.4 \text{ dB}$$

finding Phasemargin ...

$$|GH| = \frac{\sqrt{(-1000\omega_0^2 - 400000)^2}}{\sqrt{(\omega_0^5 + 210\omega_0^3 - 2000\omega_0)^2 + (-8\omega_0^4 + 500\omega_0^2)^2}} = 1$$

$$(-1000\omega_0^2 - 400,000)^2 = (\omega_0^5 + 210\omega_0^3 - 2000\omega_0)^2 + (-8\omega_0^4 + 500\omega_0^2)^2$$

$$\rightarrow \omega_0 = 11.69 \text{ rad/s}$$

$$\phi = \frac{1}{2} \text{num} - \frac{1}{2} \text{den} = \tan^{-1} \left(\frac{0}{10/10} \right) - \tan^{-1} \left(\frac{\omega_0^5 + 210\omega_0^3 - 2000\omega_0}{-8\omega_0^4 + 500\omega_0^2} \right)$$

$$= 81.3^\circ$$

$$PM = 180^\circ + \phi = 180^\circ + 81.3^\circ \rightarrow PM = 261.3^\circ$$

STUDENT ANSWER: 10 POINTS POSSIBLE

In [22]: % gain margin calculations...

```
w = 3.02111;
GH_mag = abs((1000*(w*1j+20))/((w*1j)*(w*1j+10)*((w*1j)^2+2*w*1j+10));
GM = 1/GH_mag
```

% phase margin calculations...

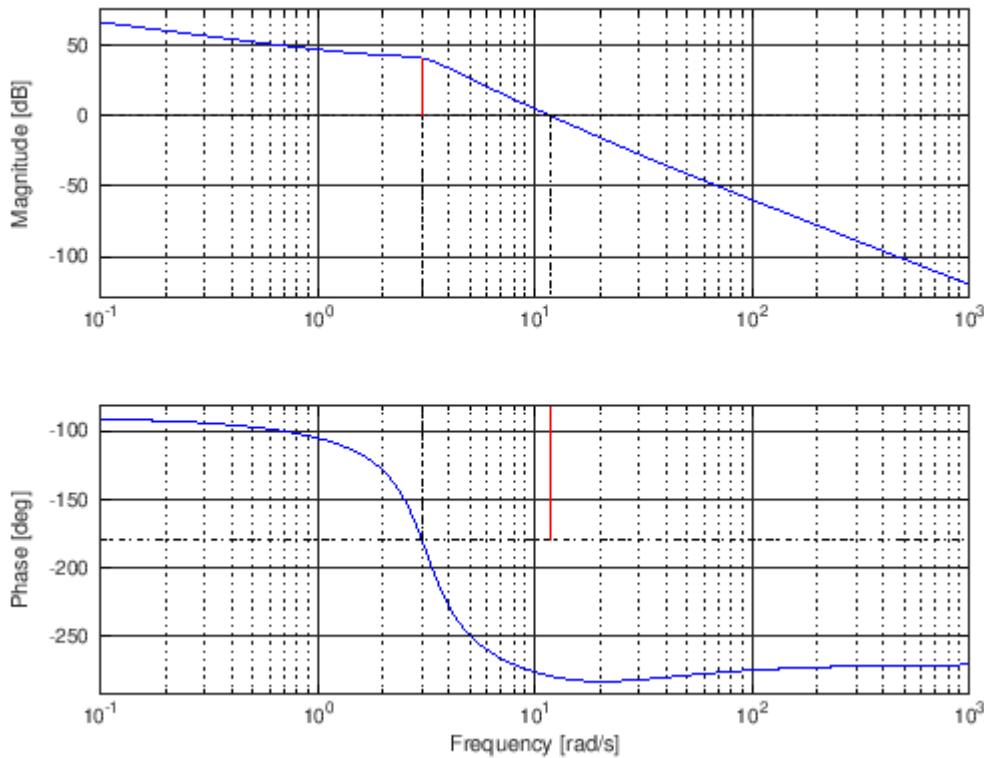
```
w0 = 11.6906;
phi = atan2d(0,1000*(-w0^2-400))-atan2d(w0^5+210*w0^3-2000*w0,-8*w0^4);
PM = 180+phi
```

% checking answers...

```
s = tf('s');
GH = (1000*(s+20))/(s*(s+10)*(s^2+2*s+10));
margin(GH)
```

GM = 0.0095255
PM = 261.31

GM = -40.4223 dB (at 3.02111 rad/s), PM = 261.309 deg (at 11.6906 rad/s)



Problem 3

Briefly discuss the qualitative differences between these two control strategies for this plant (P vs. PI), using *everything* you know about controller design. Comment on the relative stability, achievable eigenvalues, maximum root locus gain to maintain a stable system, steady state error, etc. Show any work that you use to support your analysis.

STUDENT ANSWER: 10 POINTS POSSIBLE

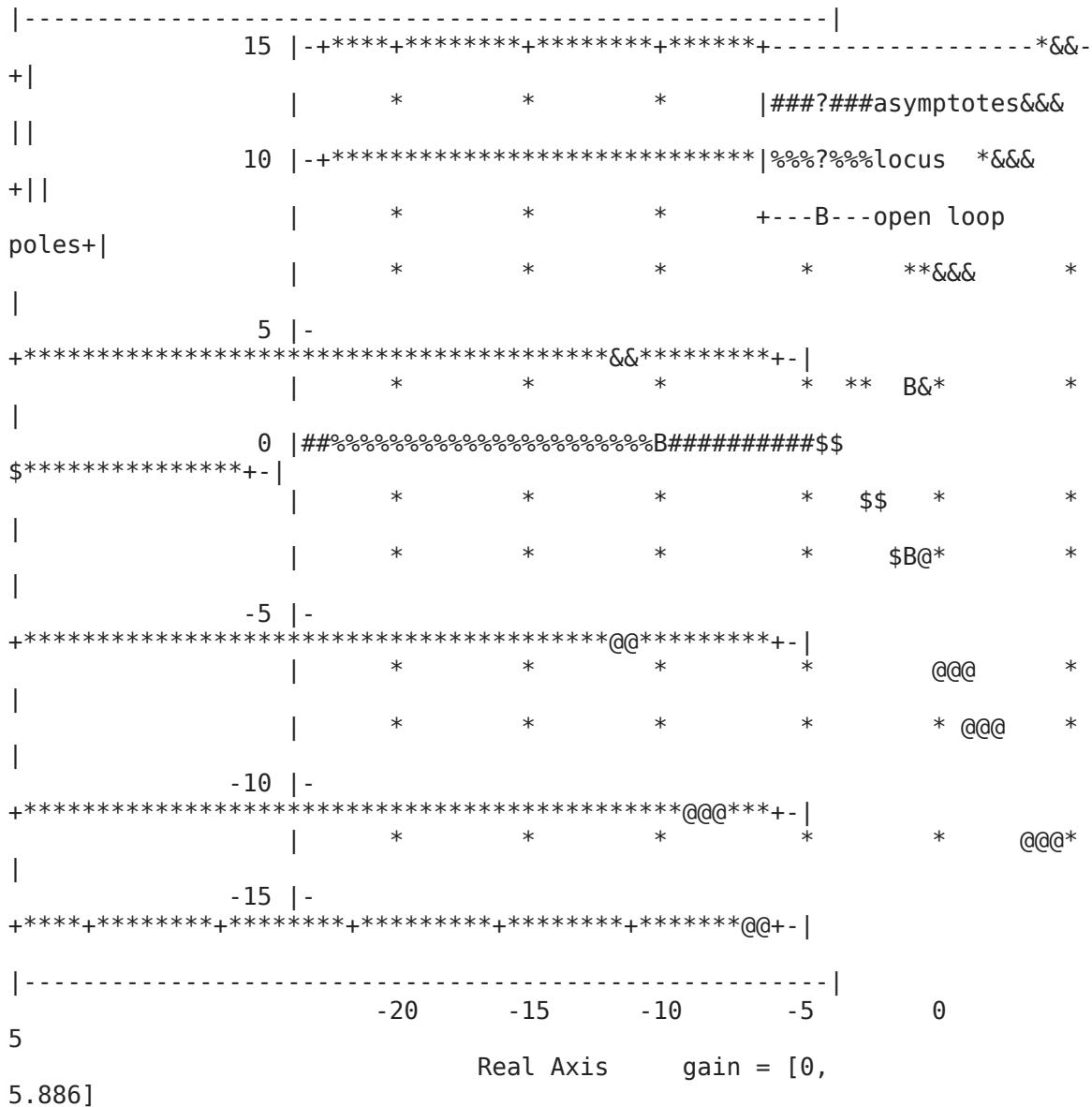
The gain and phase margin is decreased moving from P to PI control. The zero at -20, is to the left of all open loop poles (and larger in magnitude than the sum of all the poles), thus the place where the asymptotes meet is going to be to the right of the y-axis (in the unstable region). Therefore, the branches originating at the two complex open loop poles, will be bent inwards towards one another, thus they will cross the y-axis into the instability sooner than with only proportional control. In addition, from the angle deficiency method, we know that PI-control causes the root locus to bend to the right. Therefore, it makes sense that the under PI control, this system has smaller margins before going unstable as the root locus is being bent more to the right, in other words, bent as to go unstable quicker.

10/10

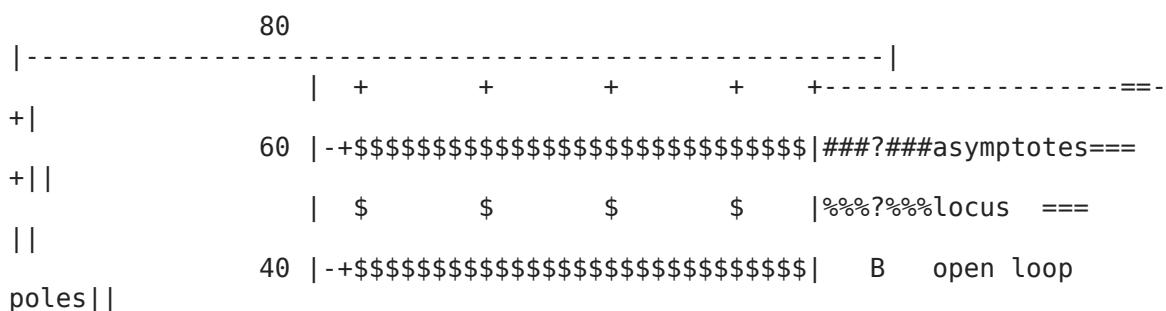
```
In [24]: GH_P = 1000/((s+10)*(s^2+2*s+10));
figure()
rlocus(GH_P)

figure()
GH_PI = (1000*(s+20))/(s*(s+10)*(s^2+2*s+10));
rlocus(GH_PI)
```

Root Locus of
GHP



Root Locus of
GHPI



Bode plot showing magnitude and phase margins versus frequency.

Magnitude Plot (Top):

Frequency (rad/s)	Magnitude (dB)
-80	-100
-60	-100
-40	-100
-20	-100
0	0
20	615.923
40	615.923

Phase Margin Plot (Bottom):

Frequency (rad/s)	Phase Margin (degrees)
-80	-61.5
-60	-61.5
-40	-61.5
-20	-61.5
0	-61.5
20	-61.5
40	-61.5

