Root Locus

Dr. Mitch Pryor

Lesson Objective

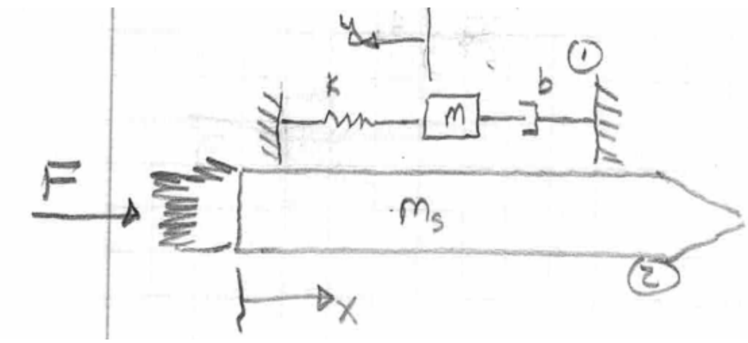
- Understand how Root Locus can be a useful tool for understanding controllers in the frequency domain (using an illustrative example)
- Find the value of a parameter of interest which makes a system go from a stable to unstable response.

Recall our awesome rocket sled…

http://www.sandia.gov/vqsec/SON-ST.html

Rocket Sled, Review

- Find the open loop transfer function T(s) for a rocket sled.



Let, $$Q(s) = -\frac{F(t)}{M_s}$$

$$m\frac{d^2(x+y)}{dt^2} + b\frac{dy}{dt} + ky = 0 \qquad (1)$$

$$M_s\frac{d^2y}{dt^2} = F(\mathrm{t}) \qquad (2)$$

Plug (2) into (1)

$$\frac{d^2y}{dt^2} + \frac{b}{m}\frac{dy}{dt} + \frac{k}{m}y = -\frac{F(t)}{M_s}$$

$$s^2Y(s) + 3sY(\mathrm{s}) + 2Y(s) = Q(s)$$

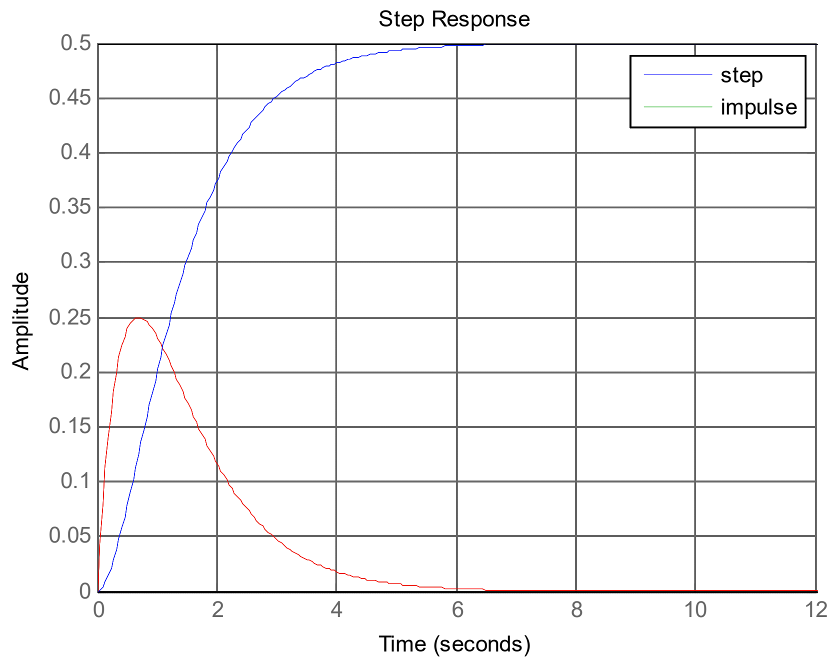
$$T(s) = \frac{output}{input} = \frac{Y(s)}{Q(s)} = \frac{1}{s^2 + 3s + 2}$$

Recall, T(s) is the ratio of the output/input for a system given all initial conditions are zero.

Rocket Sled

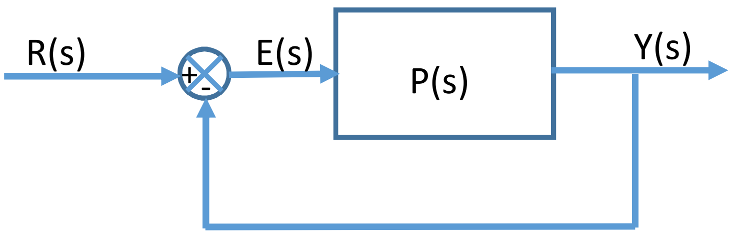
$$T(s) = \frac{1}{s^2 + 3s + 2}$$

```
num = [1]
den = [ 1 3 2 ]

sys = tf( num, den )


hold on;
impulse(sys, 'r')
step(sys, 'b')
legend('step', 'impulse');
```



Our open loop zero frequency gain is 0.5 since our reference input was 1.

Rocket Sled with Unity Feedback



$$T_{cl}(s) = \frac{Y(s)}{R(s)} = \frac{\mathrm{P}(s)}{1+\mathrm{P}(s)}$$

$$T_{cl}(s) = \frac{Y(s)}{R(s)} = \frac{\frac{1}{s^2+3s+2}}{1+\frac{1}{s^2+3s+2}}$$

$$T_{cl}(s) = \frac{Y(s)}{R(s)} = \frac{\frac{1}{s^2+3s+2}}{1+\frac{1}{s^2+3s+2}} \frac{s^2+3s+2}{s^2+3s+2}$$

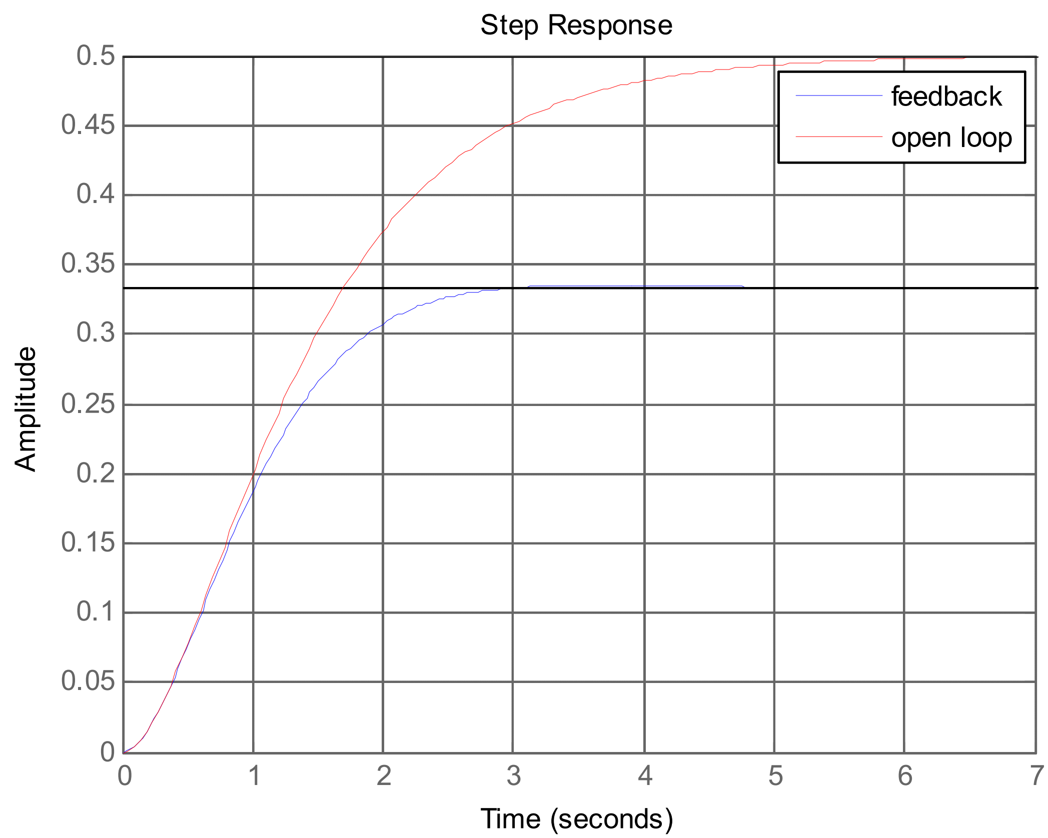
$$T_{cl}(s) = \frac{Y(s)}{R(s)} = \frac{1}{s^2+3s+2+1}$$

$$T_{cl}(s) = \frac{Y(s)}{R(s)} = \frac{1}{s^2+3s+3}$$

Rocket Sled with Feedback

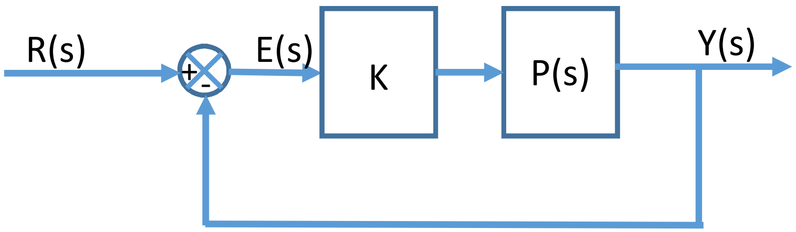
$$T_{cl}(s) = \frac{Y(s)}{R(s)} = \frac{1}{s^2 + 3s + 3}$$

```
clear all;
num = [1];
den = [ 1 3 2 ];
numfb = [ 1 ];
denfb = [ 1 3 2+1 ];

sys = tf( num, den );
sysfb = tf( numfb, denfb );


hold on; grid on;
step( sysfb, 'b' );
step( sys, 'r' );
legend('feedback', 'open loop');
```



Simple unity feedback didn't really help, and the zero frequency gain is lower.

Rocket Sled with Amplified Feedback



$$T_{cl}(s) = \frac{Y(s)}{R(s)} = \frac{K\,\mathrm{P}(s)}{1 + K\,\mathrm{P}(s)}$$

$$T_{cl}(s) = \frac{Y(s)}{R(s)} = \frac{\dfrac{K}{s^2+3s+2}}{1+\dfrac{K}{s^2+3s+2}}$$

$$T_{cl}(s) = \frac{Y(s)}{R(s)} = \frac{\dfrac{K}{s^2+3s+2}}{1+\dfrac{K}{s^2+3s+2}}\,\frac{s^2+3s+2}{s^2+3s+2}$$

$$T_{cl}(s) = \frac{Y(s)}{R(s)} = \frac{K}{s^2+3s+2+K}$$

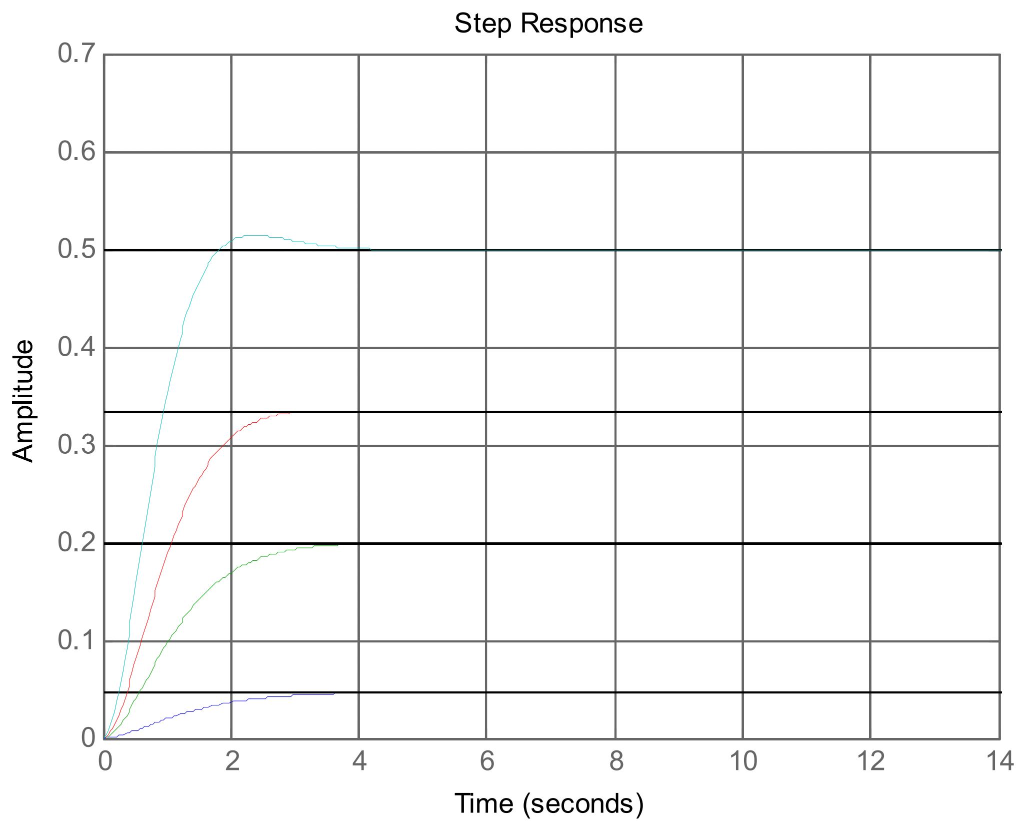
What is the zero frequency gain as the gain approaches infinity?

Are there practical limits to what the maximum gain can be?

Robot sled with amplified feedback

$$T_{cl}(s) = \frac{Y(s)}{R(s)} = \frac{K}{s^2 + 3s + 2 + K}$$

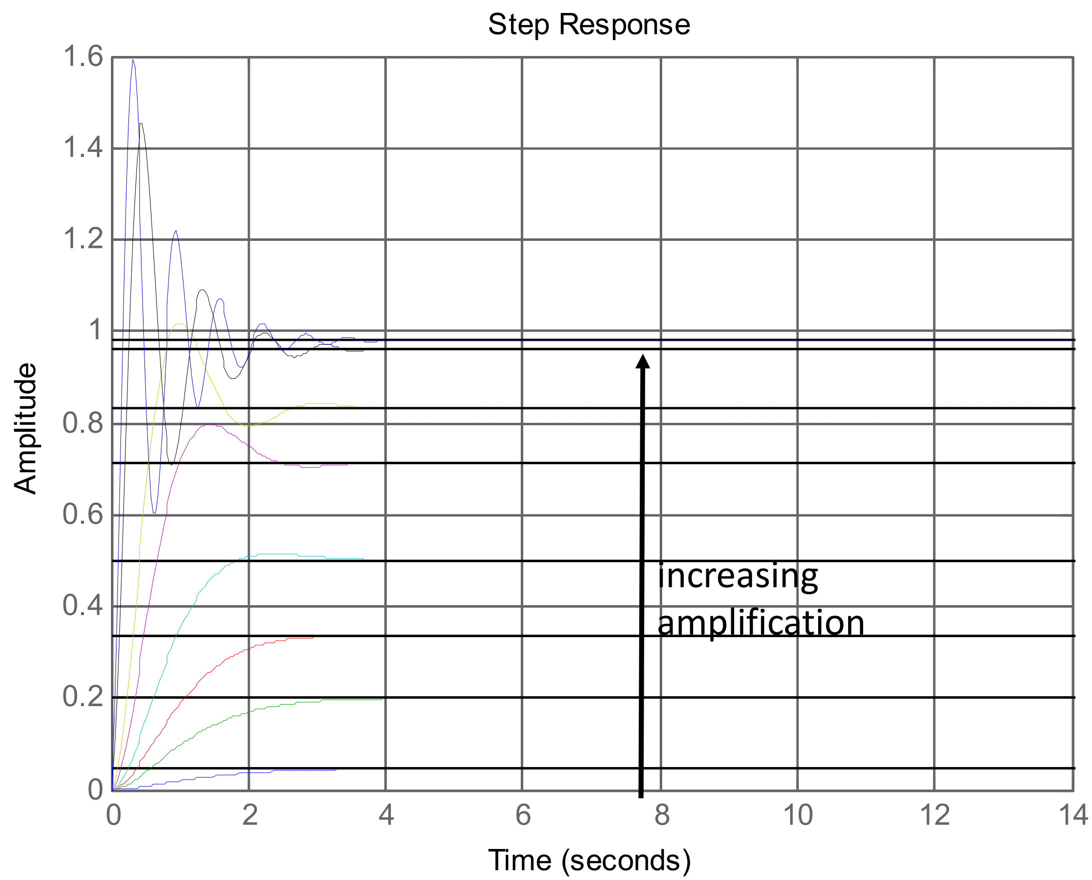
```
clear all;

K = [ .1 .5 1 2 ];

for i=1:length(K)
    numfb = [ K(i) ];
    denfb = [ 1 3 2+K(i) ];
    sysfb = tf( numfb, denfb );
    hold on;
    step( sysfb );
end
hold off; grid on;
```



Robot sled with amplified feedback

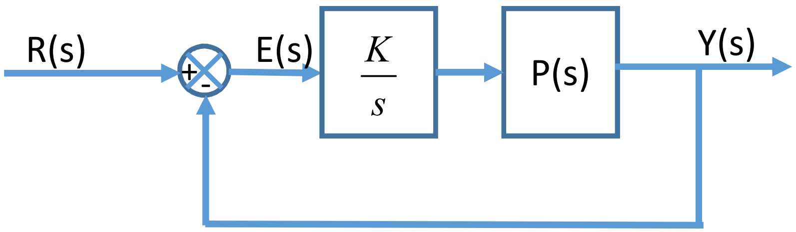
$$T_{cl}(s) = \frac{Y(s)}{R(s)} = \frac{K}{s^2 + 3s + 2 + K}$$

```
clear all;

K = [ .1 .5 1 2 5 10 50 100 ];

for i=1:length(K)
    numfb = [ K(i) ];
    denfb = [ 1 3 2+K(i) ];
    sysfb = tf( numfb, denfb );
    hold on;
    step( sysfb );
end
hold off; grid on;
```



Practically, is it possible to eliminate the steady state error?

Recall that integrators eliminate steady-state error



$$T_{cl}(s) = \frac{Y(s)}{R(s)} = \frac{\frac{K}{s}\mathrm{P}(s)}{1+\frac{K}{s}\mathrm{P}(s)}$$

$$T_{cl}(s) = \frac{Y(s)}{R(s)} = \frac{\frac{K}{s^3+3s^2+2s}}{1+\frac{K}{s^3+3s^2+2s}}$$

$$T_{cl}(s) = \frac{Y(s)}{R(s)} = \frac{\frac{K}{s^3+3s^2+2s}}{1+\frac{K}{s^3+3s^2+2s}}\,\frac{s^3+3s^2+2s}{s^3+3s^2+2s}$$

$$T_{cl}(s) = \frac{Y(s)}{R(s)} = \frac{K}{s^3+3s^2+2s+K}$$

What is the zero frequency gain as the gain approaches infinity?

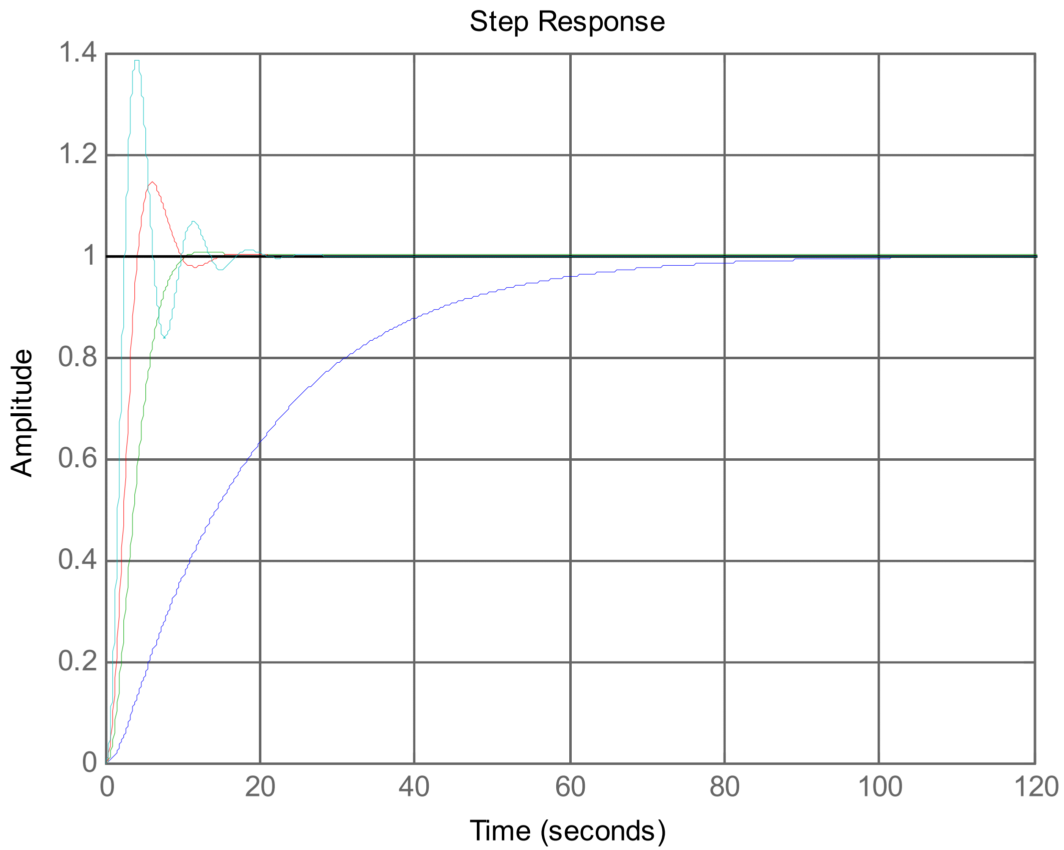
More simply, what is the zero frequency gain?

Now amplify the feedback

$$T_{OL}(s) = \frac{1}{s\left(s^2 + 3s + 2\right)}$$

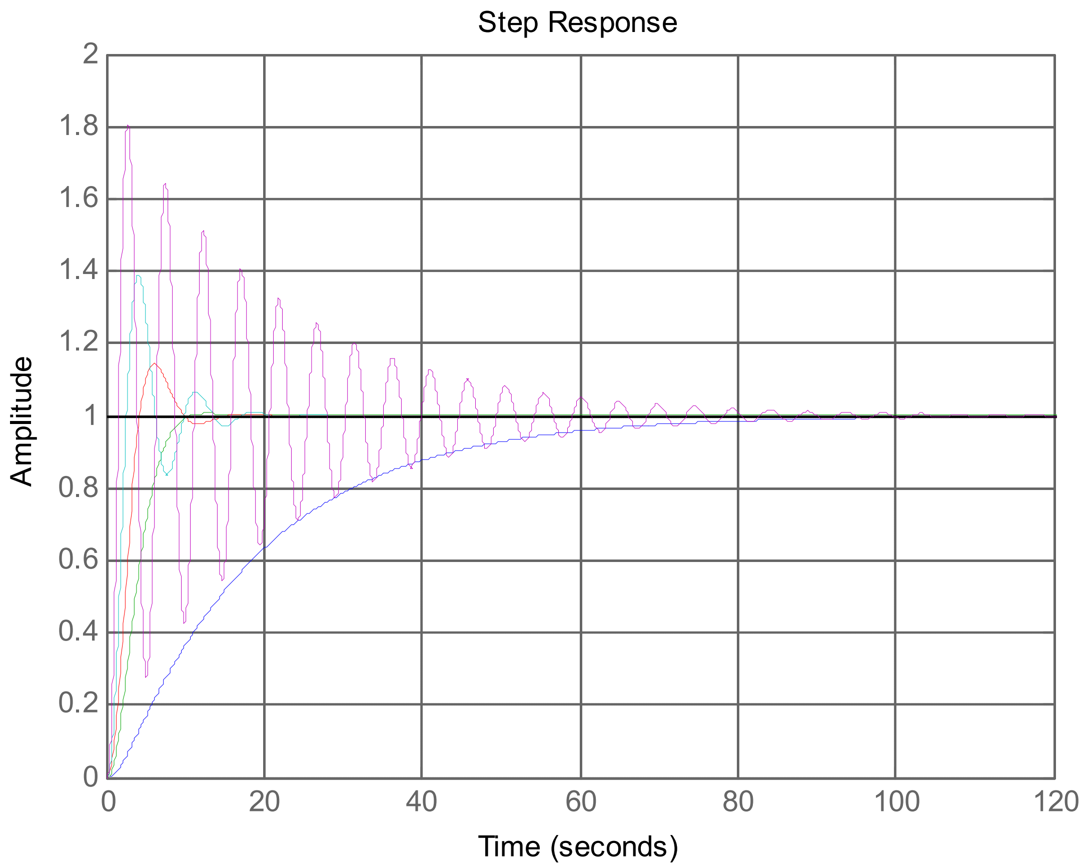
$$T_{CL}(s) = \frac{K}{s^3 + 3s^2 + 2s + K}$$

```
clear all;

K = [ .1 .5 1 2 ];

for i=1:length(K)
    numfb = [ K(i) ];
    denfb = [ 1 3 2 K(i) ];
    sysfb = tf( numfb, denfb );
    hold on;
    step( sysfb );
end
hold off; grid on;
```



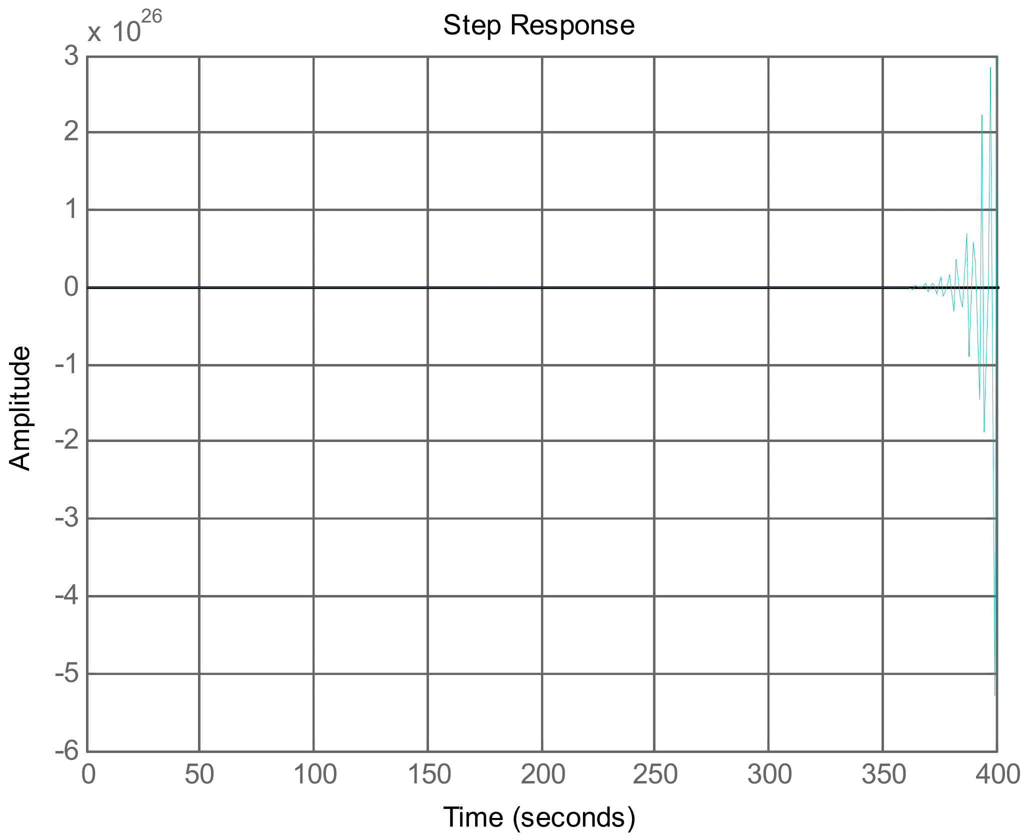
Really amplify the feedback

```
clear all;

K = [ .1 .5 1 2 5 ];

for i=1:length(K)
    numfb = [ K(i) ];
    denfb = [ 1 3 2 K(i) ];
    sysfb = tf( numfb, denfb );
    hold on;
    step( sysfb );
end
hold off; grid on;
```

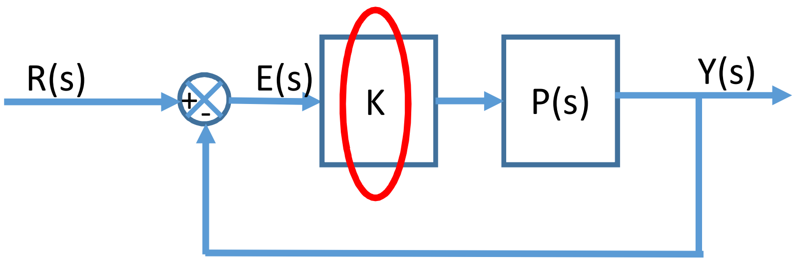


Really, REALLY amplify the feedback

```
clear all;

K = [ .1 .5 1 2 5 10 ];

for i=1:length(K)
    numfb = [ K(i) ];
    denfb = [ 1 3 2 K(i) ];
    sysfb = tf( numfb, denfb );
    hold on;
    step( sysfb );
end
hold off; grid on;
```

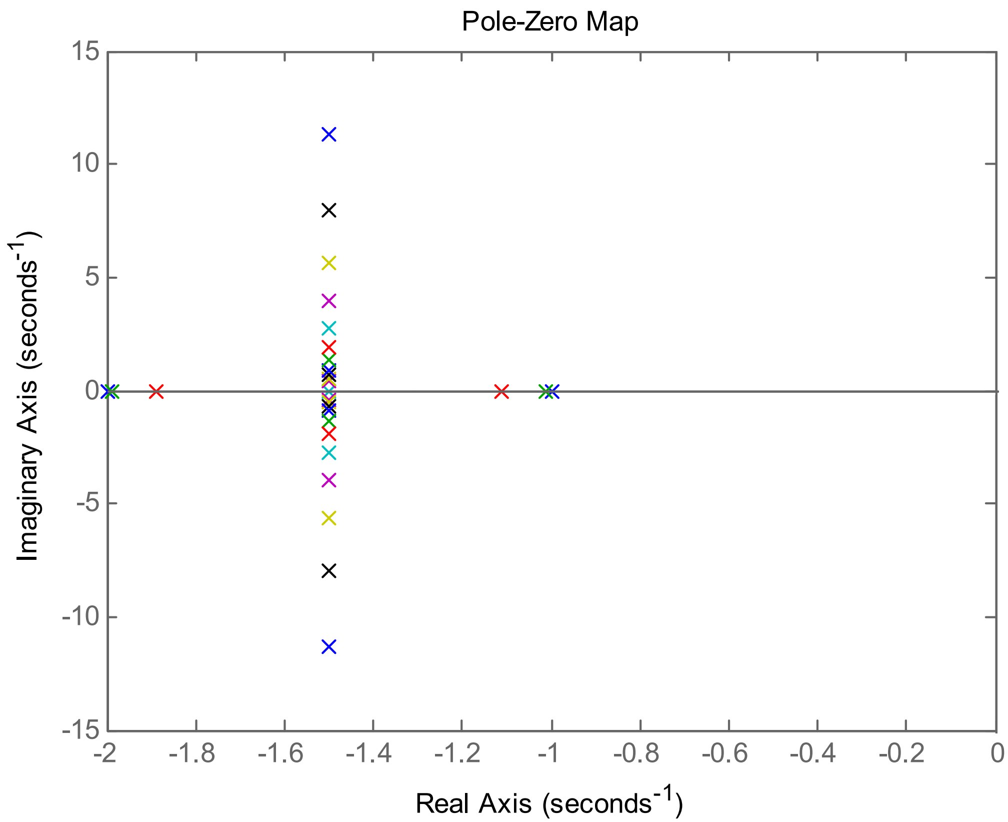


Poles for different gains

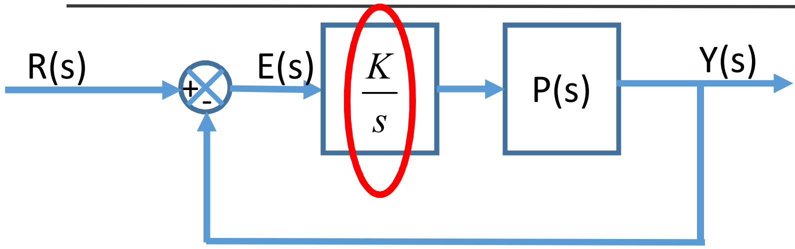


$$T_{CL}(s) = \frac{K}{s^2 + 3s + 2 + K}$$

```
clear all;

K = [ 0 .01 .1 .25 .4 .5 .75 1 2 4 8 16
32 64 128 ];

for i=1:length(K)
    num = [K(i)];
    den = [ 1 3 2+K(i) ];
    sys = tf( num, den );
    pzmap( sys );
    hold on;
    pause( 1 );
end
```

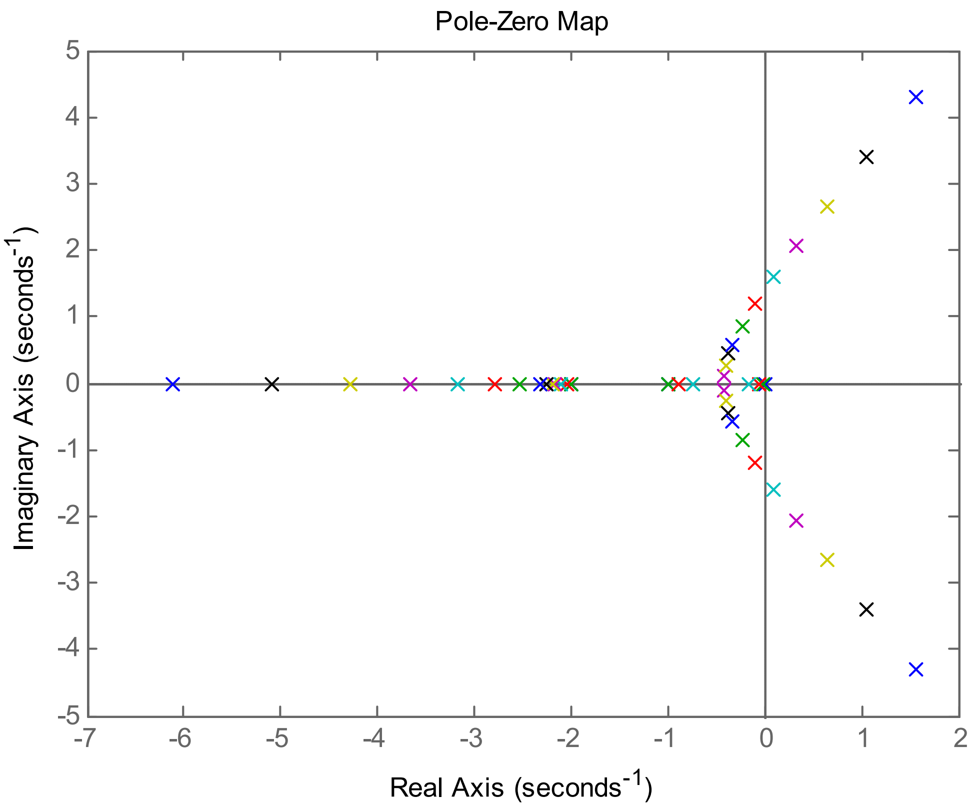


And in the system with the integrator…

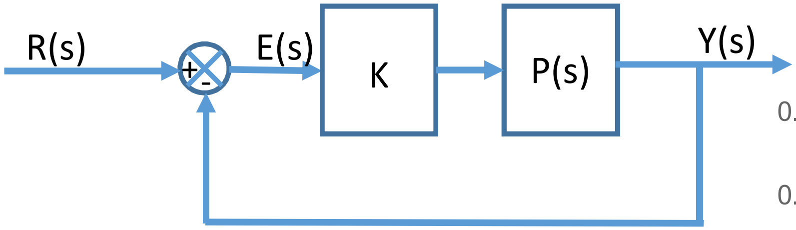


$$T_{CL}(s) = \frac{K}{s^3 + 3s^2 + 2s + K}$$

```
clear all;

K = [ 0 .01 .1 .25 .4 .5 .75 1 2 4 8 16
32 64 128 ];

for i=1:length(K)
    num = [K(i)];
    den = [ 1 3 2 K(i) ];
    sys = tf( num, den );
    pzmap( sys );
    hold on;
    pause( 1 );
end
```

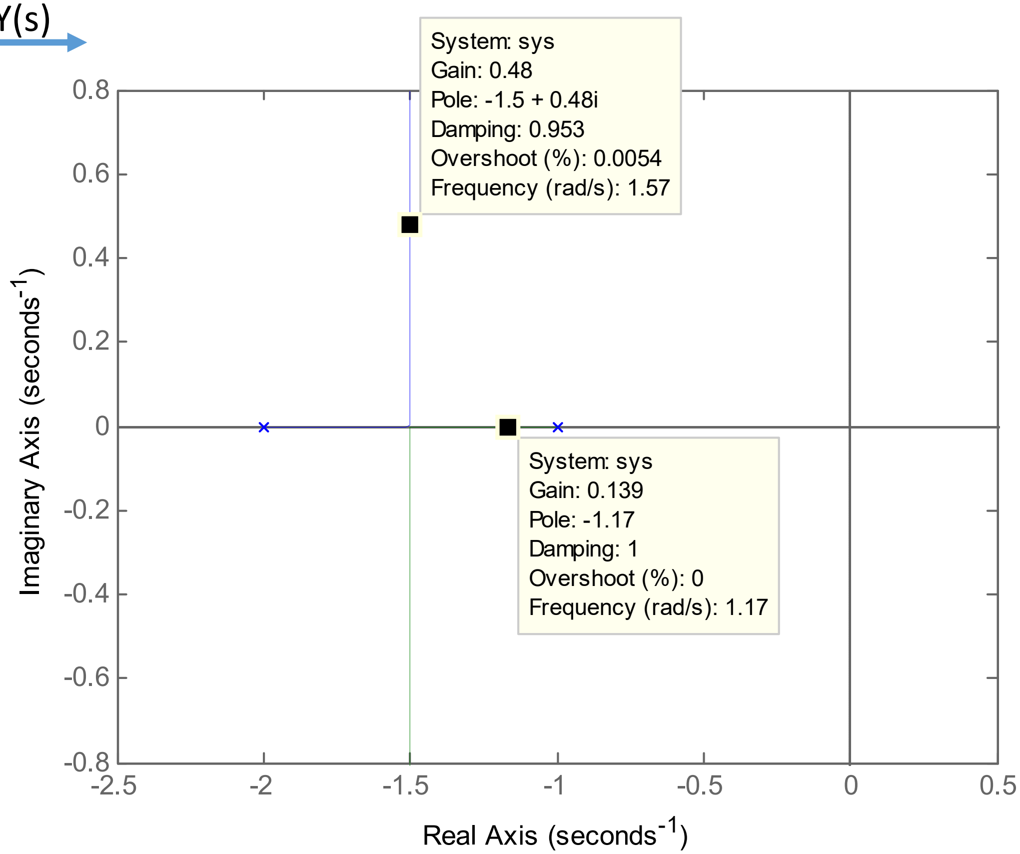


Example: Root Locus

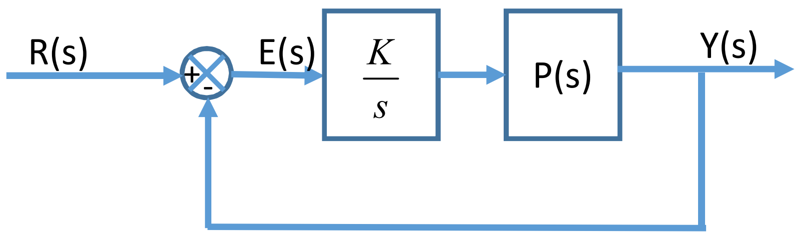


$$T_{CL}(s) = \frac{K}{s^2 + 3s + 2 + K}$$

```
clear all;

num = [1 ];
den = [ 1 3 2 ];
sys = tf( num, den );

rlocus(sys)
```

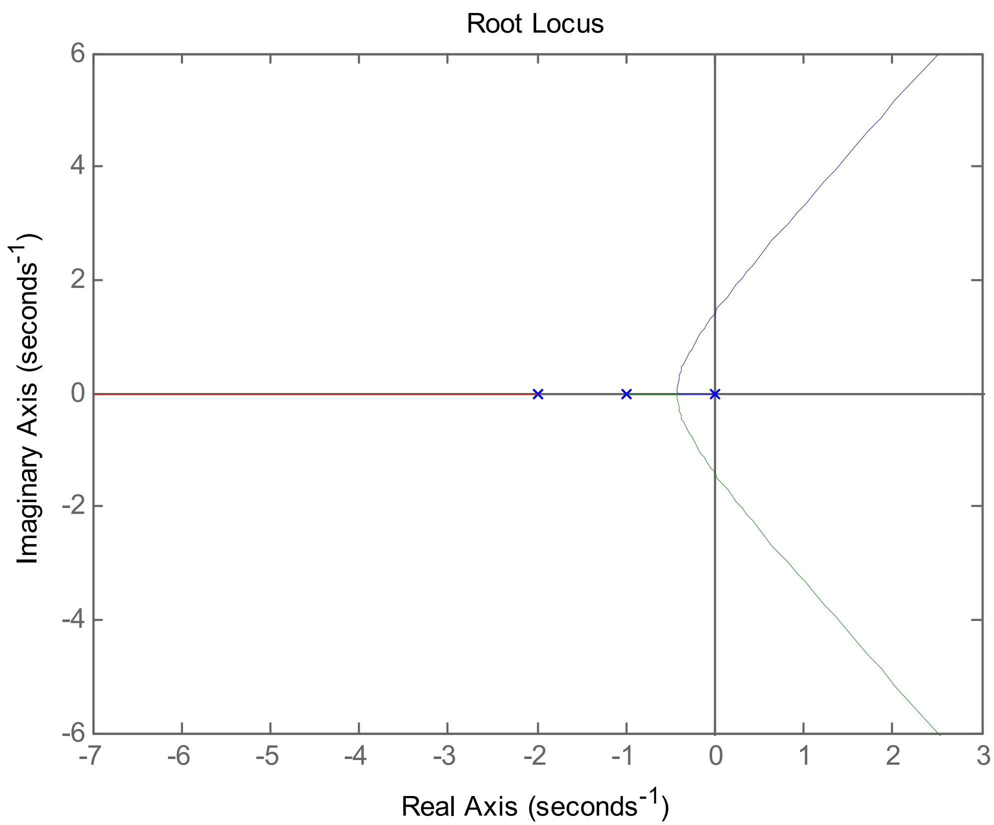


Example: Root Locus



$$T_{CL}(s) = \frac{K}{s^3 + 3s^2 + 2s + K}$$

```
clear all;

num = [1 ];
den = [ 1 3 2 0 ];
sys = tf( num, den );

rlocus(sys)
```



Root Locus relates Gains, Poles & Zeros

$$G(s) = C(sI - A)^{-1}B + D = \frac{\text{num}(s)}{\text{den}(s)} = K\frac{\text{num}'(s)}{\text{den}'(s)}$$

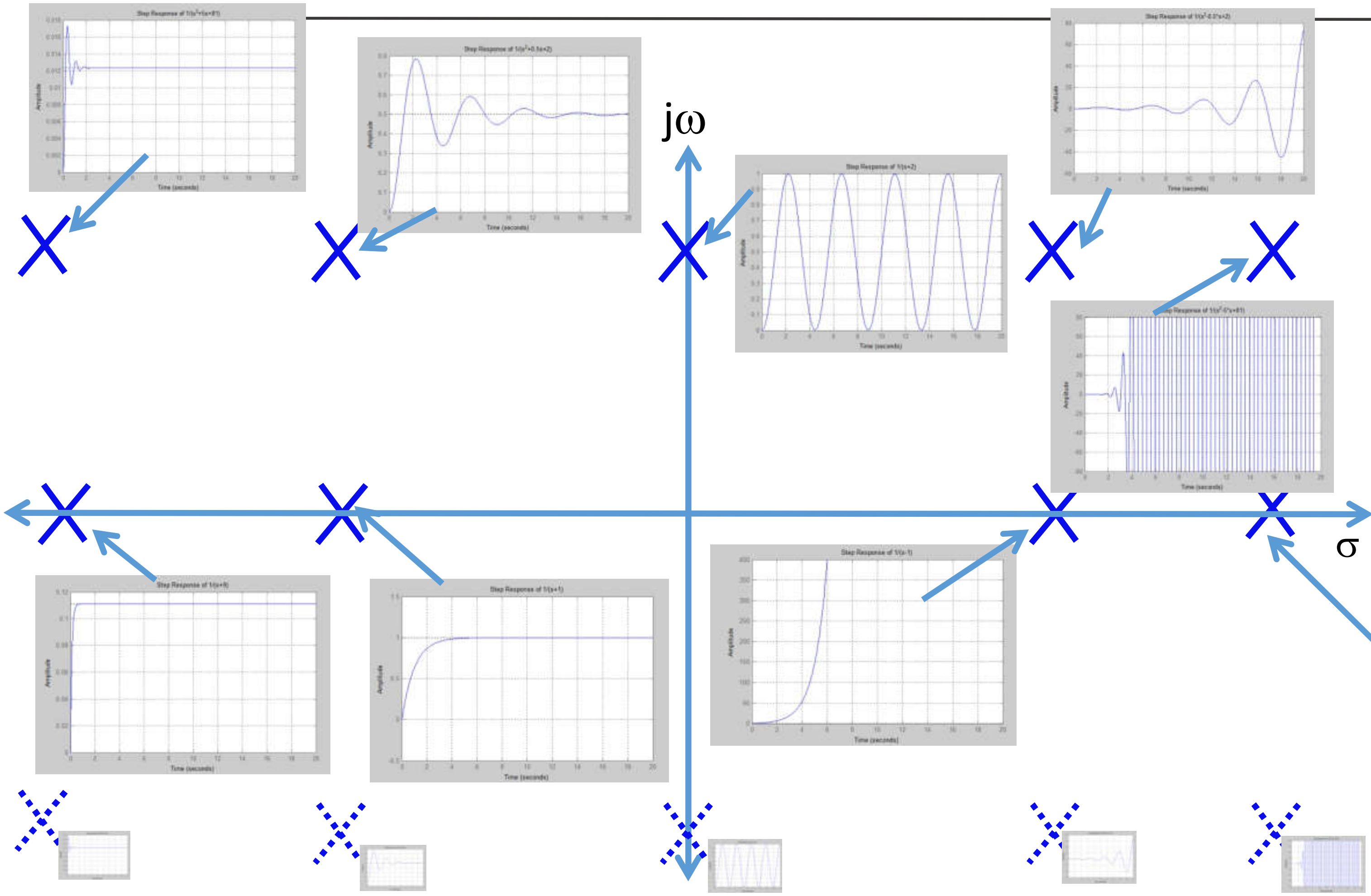
- Denominator roots are the system “poles”
  - Poles associated with the *modes* of the system
  - and the eigenvalues from state-space representation
- Numerator roots are the system “zeros”
  - Zeros counteract the effect of a pole at a given location
- The variable s is a complex number (0+jω for sinusoidal input)
- The value of G(0) is the zero frequency or steady state gain of the system

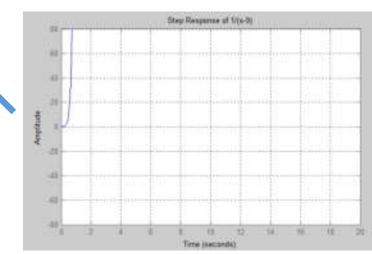
$$G(0) = D - CA^{-1}B = \frac{y_o}{u_o} = \frac{b_1 0^m + b_2 0^{m-1} + \cdots + b_m}{a_1 0^n + a_2 0^{n-1} + \cdots + a_n} = \frac{b_m}{a_n}$$

Using Root Locus

- We can use the block diagrams to design simple (and eventually complex) controllers for systems modeled in the Laplace domain.
- In many cases, the parameter we need to quantify is an amp gain that determines the response and stability of the feedback system.
  - But it can be tedious to find this gain value via trial and error.
- Root Locus is a plot of the systems transfer function on a complex plane,
  - We can manual plot the root locus by incrementally increasing the parameter of interest
  - MATLAB has several functions (rlocus, rlocfind, rltool, zpk, pzmap, pzplot, etc.) to find (or help find) the Root Locus
  - We have a step by step procedure to find it more intuitively by hand
- We can quickly examine the impact of adding zeros or poles to the system as a control strategy.

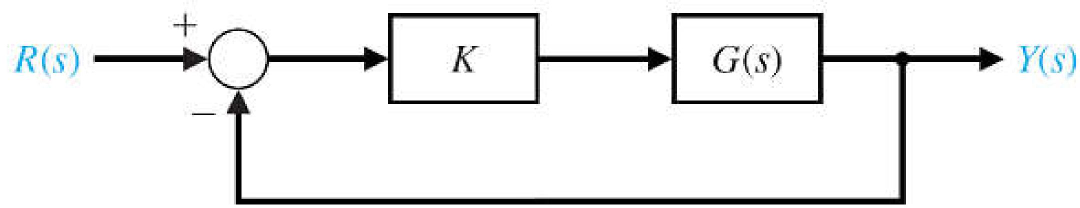
Root Locus Factor Responses





- A complete system will sum all of these effects that are present in the system's response
- The dominating effects are from poles closest to the origin

The root locus plot



- The root locus plots system response as a function of positive real values of K.
- With *unity* negative feedback, the closed loop transfer function is

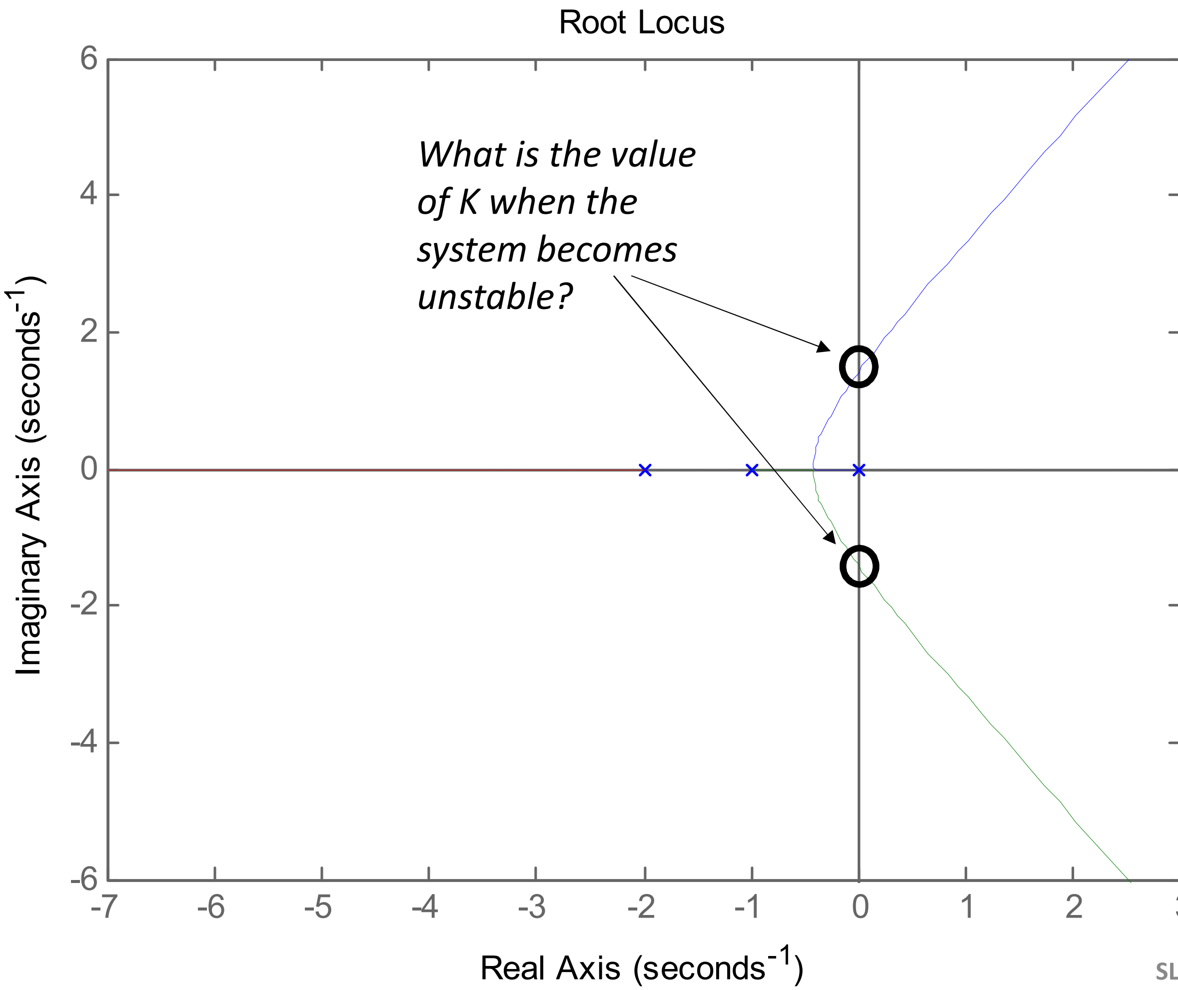
$$\frac{Output(s)}{Input(s)} = \frac{KG(s)}{1+KG(s)} = \frac{K\dfrac{N_G(s)}{D_G(s)}}{1+K\dfrac{N_G(s)}{D_G(s)}} = \frac{KN_G(s)}{D_G(s)+KN_G(s)}$$

- The characteristic equation is $\Delta(s) = 1 + KG(s) = D_G(s) + KN_G(s)$
- MATLAB makes the rlocus relatively easy to find
  - rlocus, rlocfind, rltool, zpk, pzmap, pzplot, etc.
- It is (relatively) easy *and useful* to sketch the Root Locus.

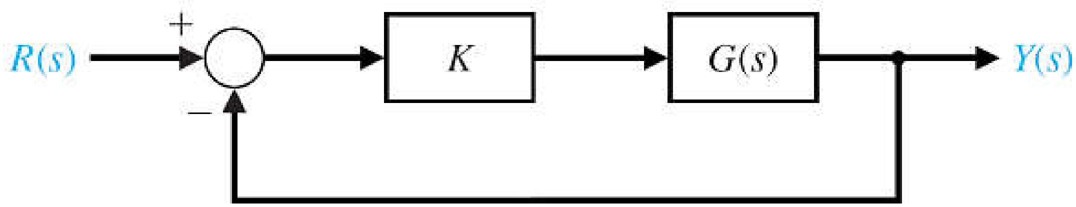
...But first...

$$T_{CL}(s) = \frac{K}{s^3 + 3s^2 + 2s + K}$$

```
clear all;

num = [1 ];
den = [ 1 3 2 0 ];
sys = tf( num, den );

rlocus(sys)
```



Necessary but NOT sufficient



$$\frac{Output(s)}{Input(s)} = \frac{KG(s)}{1+KG(s)} = \frac{K\dfrac{N_G(s)}{D_G(s)}}{1+K\dfrac{N_G(s)}{D_G(s)}} = \frac{KN_G(s)}{D_G(s)+KN_G(s)}$$

- The Characteristic Equation (CE) is $\Delta(s) = 1 + KG(s) = D_G(s) + KN_G(s)$
- It is necessary *but not sufficient for all the coefficients of the CE to be nonzero and the same sign*

<u>Definitely unstable</u>

$$\Delta(s) = s^5 + 4s^4 - s^3 + 3s^2 + s + 1$$

<u>Possibly stable</u>

$$\Delta(s) = s^5 + 4s^4 + s^3 + 3s^2 + s + 1$$

Hurwitz Criteria (necessary & sufficient)

$$\Delta(s) = a_n s^n + a_{n-1} s^{n-1} + \cdots + a_2 s^2 + a_1 s + a_0$$

Is defined using the following set of determinants:

$$\mathbf{D}_1 = \left| a_{n-1} \right|$$

$$\mathbf{D}_2 = \left\| \begin{bmatrix} a_{n-1} & a_n \\ a_{n-3} & a_{n-2} \end{bmatrix} \right\|$$

$$\mathbf{D}_3 = \left\| \begin{bmatrix} a_{n-1} & a_n & 0 \\ a_{n-3} & a_{n-2} & a_{n-1} \\ a_{n-5} & a_{n-4} & a_{n-3} \end{bmatrix} \right\|$$

Or more generally

$$\mathbf{D}_{n-1} = \left\| \begin{bmatrix} a_{n-1} & a_n & \cdots & 0 \\ a_{n-3} & a_{n-2} & \cdots & \vdots \\ \vdots & \vdots & \cdots & \vdots \\ \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & \cdots & a_0 \end{bmatrix} \right\|$$

This schema can simplify things…

$$\begin{array}{llllll} \mathbf{D_1} & a_{n-1} & a_n & 0 & 0 & 0 \\ \mathbf{D_2} & a_{n-3} & a_{n-2} & a_{n-1} & a_n & 0 \\ \mathbf{D_3} & a_{n-5} & a_{n-4} & a_{n-3} & a_{n-2} & a_{n-1} \cdots \\ \mathbf{D_4} & a_{n-7} & a_{n-6} & a_{n-5} & a_{n-4} & a_{n-3} \cdots \\ \vdots & & & & & \end{array}$$

Note: the transpose of a matrix has the same determinant. So some books/sites set up the matrices using differently, but all should give the same result.

Hurwitz Criteria, cont'd

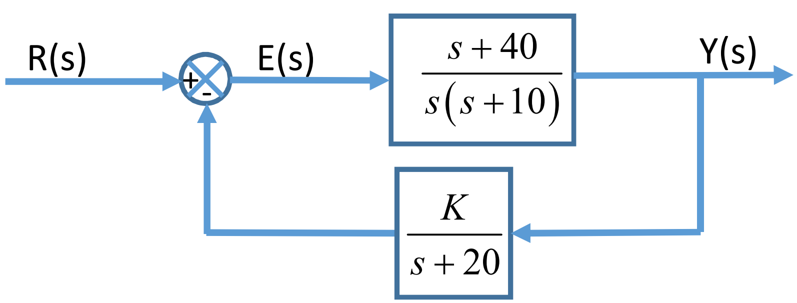
$$\Delta(s) = a_n s^n + a_{n-1} s^{n-1} + \cdots + a_2 s^2 + a_1 s + a_0$$

$$\begin{array}{llllll}
\mathbf{D}_1 & a_{n-1} & a_n & 0 & 0 & 0 \\
\mathbf{D}_2 & a_{n-3} & a_{n-2} & a_{n-1} & a_n & 0 \\
\mathbf{D}_3 & a_{n-5} & a_{n-4} & a_{n-3} & a_{n-2} & a_{n-1} \cdots \\
\mathbf{D}_4 & a_{n-7} & a_{n-6} & a_{n-5} & a_{n-4} & a_{n-3} \cdots \\
\vdots & & & & &
\end{array}$$

All Hurwitz determinants must be positive for a system to be stable. This criterion is both sufficient and necessary.

$$\mathbf{D}_j > 0 \forall j = 1, 2, \cdots n$$

Quick example:



$$T_{cl}(s) = \frac{\dfrac{s+40}{s(s+10)}}{1 + \dfrac{K(s+40)}{s(s+10)(s+20)}}$$

$$\begin{aligned}
\Delta(s) &= s(s+10)(s+20) + K(s+40) \\
&= s^3 + 30s^2 + (200 + K)s + 40K
\end{aligned}$$

Hurwitz Criteria, cont'd

$$\begin{array}{llllll}
\mathbf{D}_1 & a_{n-1} & a_n & 0 & 0 & 0 \\
\mathbf{D}_2 & a_{n-3} & a_{n-2} & a_{n-1} & a_n & 0 \\
\mathbf{D}_3 & a_{n-5} & a_{n-4} & a_{n-3} & a_{n-2} & a_{n-1} \cdots \\
\mathbf{D}_4 & a_{n-7} & a_{n-6} & a_{n-5} & a_{n-4} & a_{n-3} \cdots \\
\vdots & & & & &
\end{array}$$

$$\Delta(s) = a_n s^n + a_{n-1} s^{n-1} + \cdots + a_2 s^2 + a_1 s + a_0$$

Quick example cont'd $\Delta(s) = s^3 + 30s^2 + \left(200 + K\right)s + 40K$

Necessary but not sufficient: $200 + K > 0$

$$40K > 0$$

Necessary AND sufficient: $\mathbf{D}_1 = \left|a_{n-1}\right| = 30$ >0 so always true.

$$\mathbf{D}_2 = \left|\begin{bmatrix} a_{n-1} & a_n \\ a_{n-3} & a_{n-2} \end{bmatrix}\right| = \left|\begin{bmatrix} a_2 & a_3 \\ a_1 & a_0 \end{bmatrix}\right| = \left|\begin{bmatrix} 30 & 1 \\ 40K & 200+K \end{bmatrix}\right|$$

Therefore:

$$6000 + 30K - 40K > 0$$

$$600 > K$$

Thus the system is stable for all 0<*K*<600.

At this gain, at least one pole resides in the right hand plane of the Root Locus.

Routh Criterion

$$\Delta(s) = a_n s^n + a_{n-1}s^{n-1} + \cdots + a_2 s^2 + a_1 s + a_0$$

The Routh Schema contains *n+1* rows organized in the following manner…

$$\begin{array}{c|c|ccccc}
n & a_n & a_{n-2} & a_{n-4} & a_{n-6} & \cdots & 0 \\
n-1 & a_{n-1} & a_{n-3} & a_{n-5} & a_{n-7} & \cdots & 0 \\
\hline
n-2 & b_{n-1} & b_{n-2} & b_{n-3} & b_{n-4} & \cdots & 0 \\
n-3 & c_{n-1} & c_{n-2} & c_{n-3} & c_{n-4} & \cdots & 0 \\
\vdots & \vdots & & & & & \\
3 & d_{n-1} & d_{n-2} & 0 & & & \\
2 & e_{n-1} & e_{n-2} & 0 & & & \\
1 & f_{n-1} & & & & & \\
0 & g_{n-1} & & & & &
\end{array}$$

System is stable *if and only if* all elements in the first column of the Routh schema are positive.

Where the values of the third row and beyond are found using…

$$b_{n-1} = \frac{a_{n-1}a_{n-2} - a_n a_{n-3}}{a_{n-1}} \qquad c_{n-1} = \frac{b_{n-1}a_{n-3} - a_{n-1}b_{n-2}}{b_{n-1}} \qquad f_{n-1} = \frac{e_{n-1}d_{n-2} - d_{n-1}e_{n-2}}{e_{n-1}}$$

$$b_{n-2} = \frac{a_{n-1}a_{n-4} - a_n a_{n-5}}{a_{n-1}} \qquad c_{n-2} = \frac{b_{n-1}a_{n-5} - a_{n-1}b_{n-3}}{b_{n-1}} \qquad \ldots$$

$$b_{n-3} = \frac{a_{n-1}a_{n-6} - a_n a_{n-7}}{a_{n-1}} \qquad c_{n-3} = \frac{b_{n-1}a_{n-7} - a_{n-1}b_{n-4}}{b_{n-1}} \qquad g_{n-1} = e_{n-2}$$

$$\vdots \qquad \vdots$$

Most dynamical systems are not an extremely high order…

Routh Criterion, cont'd

Let's do a quick example

$$\Delta(s) = a_n s^n + a_{n-1} s^{n-1} + \cdots + a_2 s^2 + a_1 s + a_0$$
$$= s^3 + 2s^2 + 4s + 9$$

$$\begin{array}{c|cccccc} n & a_n & a_{n-2} & a_{n-4} & a_{n-6} & \cdots & 0 \\ n-1 & a_{n-1} & a_{n-3} & a_{n-5} & a_{n-7} & \cdots & 0 \\ \hline n-2 & b_{n-1} & b_{n-2} & b_{n-3} & b_{n-4} & \cdots & 0 \\ n-3 & c_{n-1} & c_{n-2} & c_{n-3} & c_{n-4} & \cdots & 0 \\ \vdots & \vdots & & & & & \\ 3 & d_{n-1} & d_{n-2} & 0 & & & \\ 2 & e_{n-1} & e_{n-2} & 0 & & & \\ 1 & f_{n-1} & & & & & \\ 0 & g_{n-1} & & & & & \end{array}$$

Where the values of the third row and beyond are found using…

$$\begin{array}{lcc} n=3: & 1 & 4 \\ n=2: & 2 & 9 \\ n=1: & -0.5 & 0 \\ n=0: & 9 & 0 \end{array}$$

$$b_{n-1} = \frac{a_{n-1}a_{n-2} - a_n a_{n-3}}{a_{n-1}}$$

$$b_{n-2} = \frac{a_{n-1}a_{n-4} - a_n a_{n-5}}{a_{n-1}}$$

$$b_{n-3} = \frac{a_{n-1}a_{n-6} - a_n a_{n-7}}{a_{n-1}}$$

$$\vdots$$

$$c_{n-1} = \frac{b_{n-1}a_{n-3} - a_{n-1}b_{n-2}}{b_{n-1}}$$

$$c_{n-2} = \frac{b_{n-1}a_{n-5} - a_{n-1}b_{n-3}}{b_{n-1}}$$

$$c_{n-3} = \frac{b_{n-1}a_{n-7} - a_{n-1}b_{n-4}}{b_{n-1}}$$

$$\vdots$$

Note $b_{n-2}=b_1$ is zero since there is not a_{n-4} *or* a_{n-5}

Note $c_{n-2}=c_1$ is zero since there is no b_{n-1} *or* b_{n-3}

The system is unstable since not all elements in the first column have the same sign.

Routh Criterion, cont'd

$$
\begin{array}{c|cccccc}
n & a_n & a_{n-2} & a_{n-4} & a_{n-6} & \cdots & 0 \\
n-1 & a_{n-1} & a_{n-3} & a_{n-5} & a_{n-7} & \cdots & 0 \\
\hline
n-2 & b_{n-1} & b_{n-2} & b_{n-3} & b_{n-4} & \cdots & 0 \\
n-3 & c_{n-1} & c_{n-2} & c_{n-3} & c_{n-4} & \cdots & 0 \\
\vdots & \vdots & & & & & \\
3 & d_{n-1} & d_{n-2} & 0 & & & \\
2 & e_{n-1} & e_{n-2} & 0 & & & \\
1 & f_{n-1} & & & & & \\
0 & g_{n-1} & & & & &
\end{array}
$$

Let's do an example with a parameter of interest:

$$
\begin{aligned}
\Delta(s) &= a_n s^n + a_{n-1} s^{n-1} + \cdots + a_2 s^2 + a_1 s + a_0 \\
&= s^3 + 2s^2 + (5+K)s + 3K
\end{aligned}
$$

Where the values of the third row and beyond are found using…

$$
\begin{array}{lcc}
n=3: & 1 & 5+K \\
n=2: & 2 & 3K \\
n=1: & 5-2K & 0 \\
n=0: & 3K & 0
\end{array}
$$

$$b_{n-1} = \frac{a_{n-1}a_{n-2} - a_n a_{n-3}}{a_{n-1}}$$

$$b_{n-2} = \frac{a_{n-1}a_{n-4} - a_n a_{n-5}}{a_{n-1}}$$

$$b_{n-3} = \frac{a_{n-1}a_{n-6} - a_n a_{n-7}}{a_{n-1}}$$

$$\vdots$$

$$c_{n-1} = \frac{b_{n-1}a_{n-3} - a_{n-1}b_{n-2}}{b_{n-1}}$$

$$c_{n-2} = \frac{b_{n-1}a_{n-5} - a_{n-1}b_{n-3}}{b_{n-1}}$$

$$c_{n-3} = \frac{b_{n-1}a_{n-7} - a_{n-1}b_{n-4}}{b_{n-1}}$$

$$\vdots$$

Note b_{n-2}=b_1 is zero since there is not a_{n-4} *or* a_{n-5}

Note c_{n-2} =c_1 is zero since there is no b_{n-1} *or* b_{n-3}

The system is only stable if 0<K<2.5!

Summary

- The Root locus can tell us a lot about the performance of a system over the range of a parameter of interest including a controller gain.
- A critical value is one where the system goes from stable to unstable.
  - MATLAB's rlocus(sys) function produces an interactive plot
  - The Hurwitz and Routh Criteria provide some analytical methods to determine this critical value.
    - Gonna leave the theory related to Hurwitz numbers for another course (or you can productively procrastinate with a little Googling…)
- Coming up! How to create the Root Locus by hand!
  - Learn more about the impact of poles and zeros (even from a proposed controller)