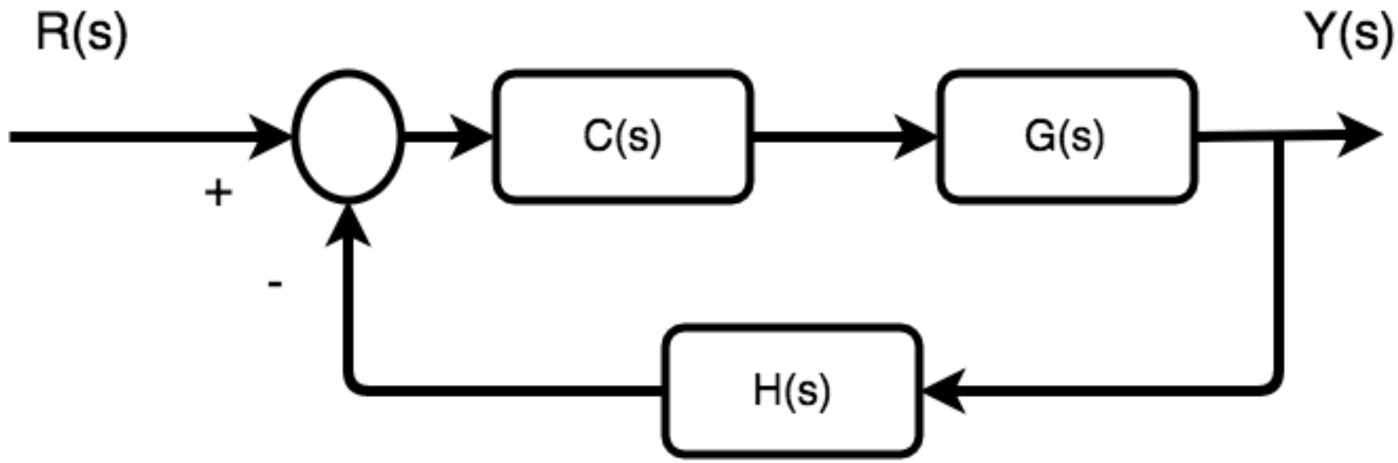
ROCO218: Control Engineering
Dr Ian Howard

Lecture 10

Root locus

Meaning of Root Locus

Consider the simple controller system where C(s) is a controller



Often need to design controller C(s) to improve system performance/stability

Poles of system need to be stable and located appropriately!

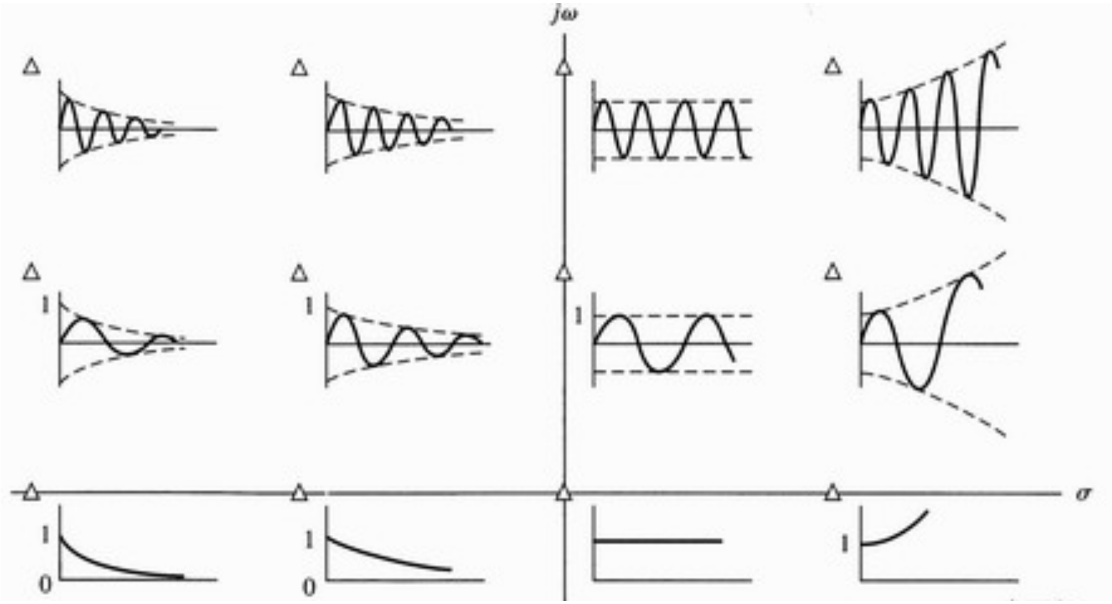
The closed-loop transfer function of the basic feedback system above is:

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

The characteristic equation, whose roots are the poles of this transfer function, is therefore:

$$1 + C(s)G(s)H(s) = 0$$

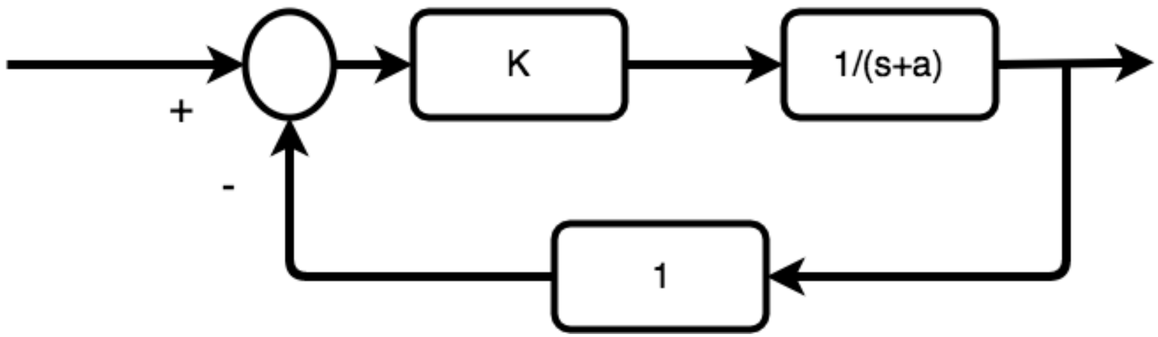
Remember: S-plane and stability



- For stable systems we want poles of the transfer function of our feedback system located in the left of the s-plane
- Also might have other – e.g. time/settle time requirements - for the system
- Root locus is a graphical presentation of the closed-loop poles as a system parameter (normally a forward gain K) is varied.
- Root locus is a powerful method of analysis and design for stability and transient response.
- A simple set of rules allow the loci to be sketched (will mention first 2 rules only as the do by hand approach is dated)

Calculating root locus: Example1

For the simple proportional controller system with forward path gain K



- This system has the open-loop transfer function

$$P(s)=\frac{K}{(s+a)}$$

- This leads to the closed-loop transfer function

$$CL(s)=\frac{\frac{K}{(s+a)}}{\left(1+\frac{K}{(s+a)}\right)}=\frac{K}{(s+a+K)}$$

Adjustable gain K will therefore affect location of the system poles

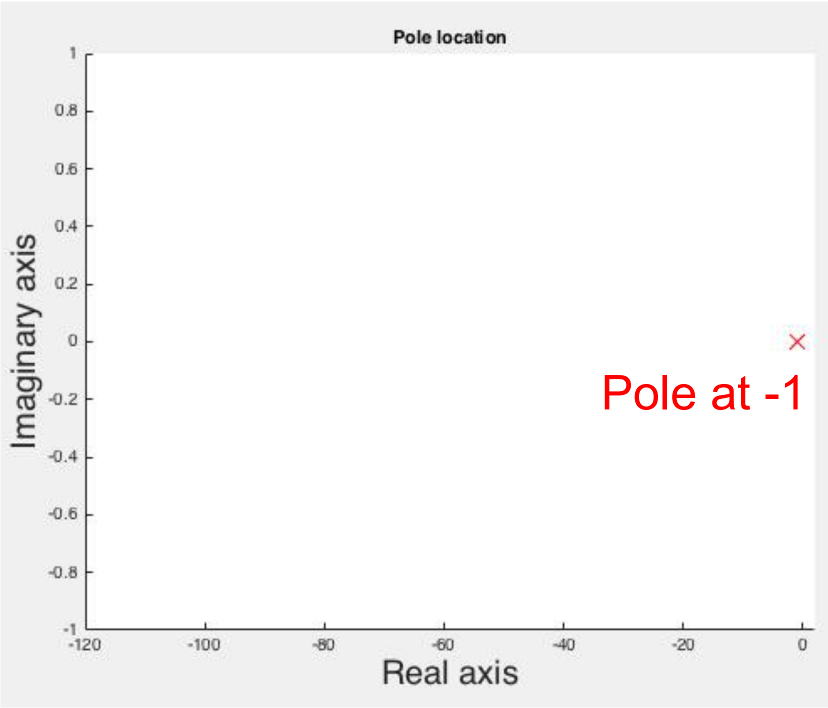
Poles occur when (s+a+K)=0 so therefore s = -(a+K)
There are no zeros for this transfer function

DIY Matlab plot of poles for a range of gain K

- By common convention
- Poles are marked on a graph with an 'X'
- Zeros are marked with an 'O'

$$CL(s) = \frac{1}{(s+a)}$$

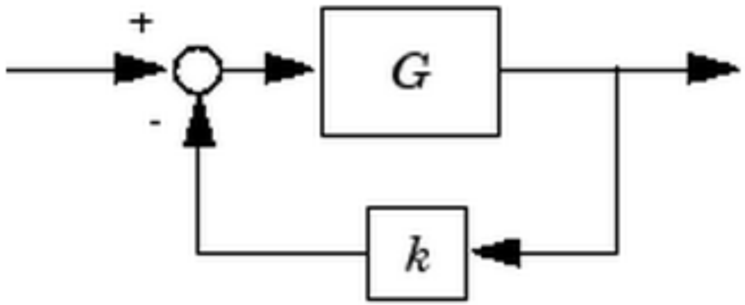
- Root-locus found by plotting poles of closed-loop transfer function for K starting at zero and going to to large value
- Here plotted here for range K = 1 to 100 in video




```matlab
% proportional gain
a=1;
% compute location of pole for samples of gain K
% for 20 points between 0 and 100
points=20
Kp = linspace(0, 100, points);
poleXY = [-(a+Kp); zeros(1,points);];

% Prepare the new file.
vidObj = VideoWriter('rootLocus1Pole.avi');
open(vidObj);
% plot the pole
figure
hold on
title('Pole location');
h=xlabel('Real axis')
set(h,'FontSize', 20);
h=ylabel('Imaginary axis')
set(h,'FontSize', 20);
%set(gca,'nextplot','replacechildren');
axis([-120 2 -1 1])
for idx = 1:points
    h=plot(poleXY(1,idx),poleXY(2,idx), 'Xr');
    set(h,'MarkerSize', 10);
    % Write each frame to the file.
    currFrame = getframe(gcf);
    writeVideo(vidObj,currFrame);
end
legend('Pole');
% Close the file.
close(vidObj);
```

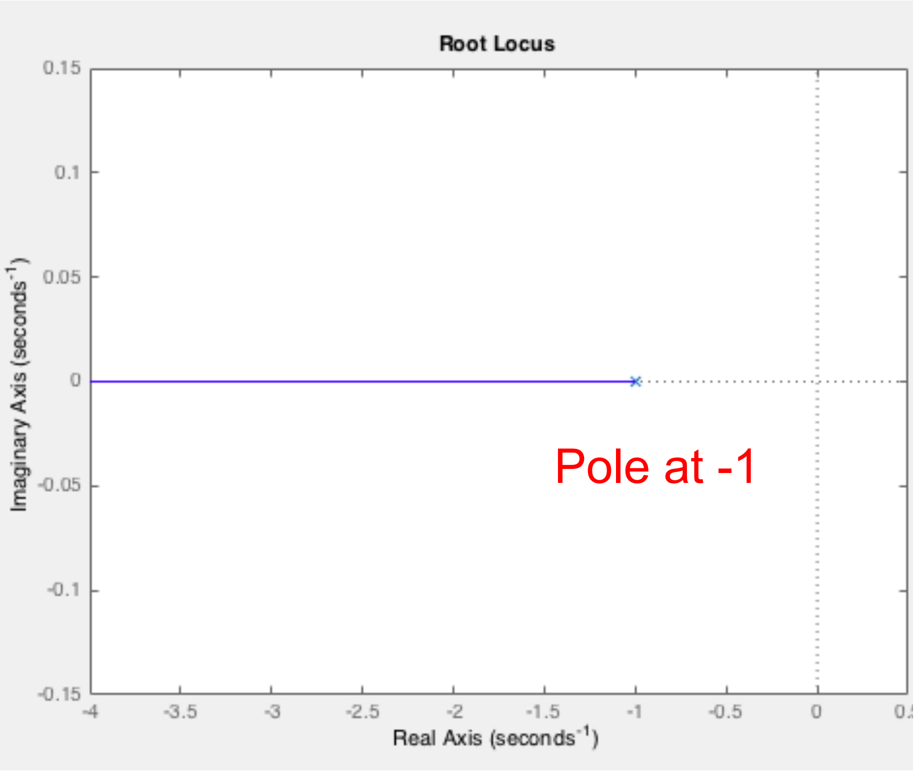
Using Matlab rlocus function

- Plotting a root locus with Matlab is easy using rlocus, rlocfind, rltool
- NB: In general root locus is hard to determine by hand



Matlab puts gain in feedback path and not in the forward path

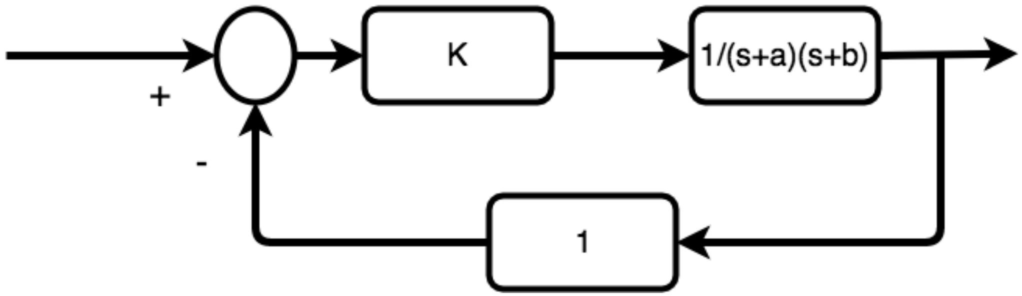
$$CL(s) = \frac{1}{(s+a)}$$




```
% first system function for s term
s = tf('s');
% build open loop system transfer function
a=1;
sys = 1/(s+a);
% calculate and plot the root locus of
% the open-loop SISO model
figure
rlocus(sys)
```

Calculating root locus: Example 2

For the proportional controller system



- This system has the open-loop transfer function

$$P(s) = \frac{K}{(s+a)(s+b)}$$

Remember: For a quadratic equation:

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}.$$

- The closed-loop transfer function is therefore

$$CL(s) = \frac{\dfrac{K}{(s+a)(s+b)}}{\left(1 + \dfrac{K}{(s+a)(s+b)}\right)} = \frac{K}{((s+a)(s+b)+K)} = \frac{K}{\left(s^2 + (a+b)s + (ab+K)\right)}$$

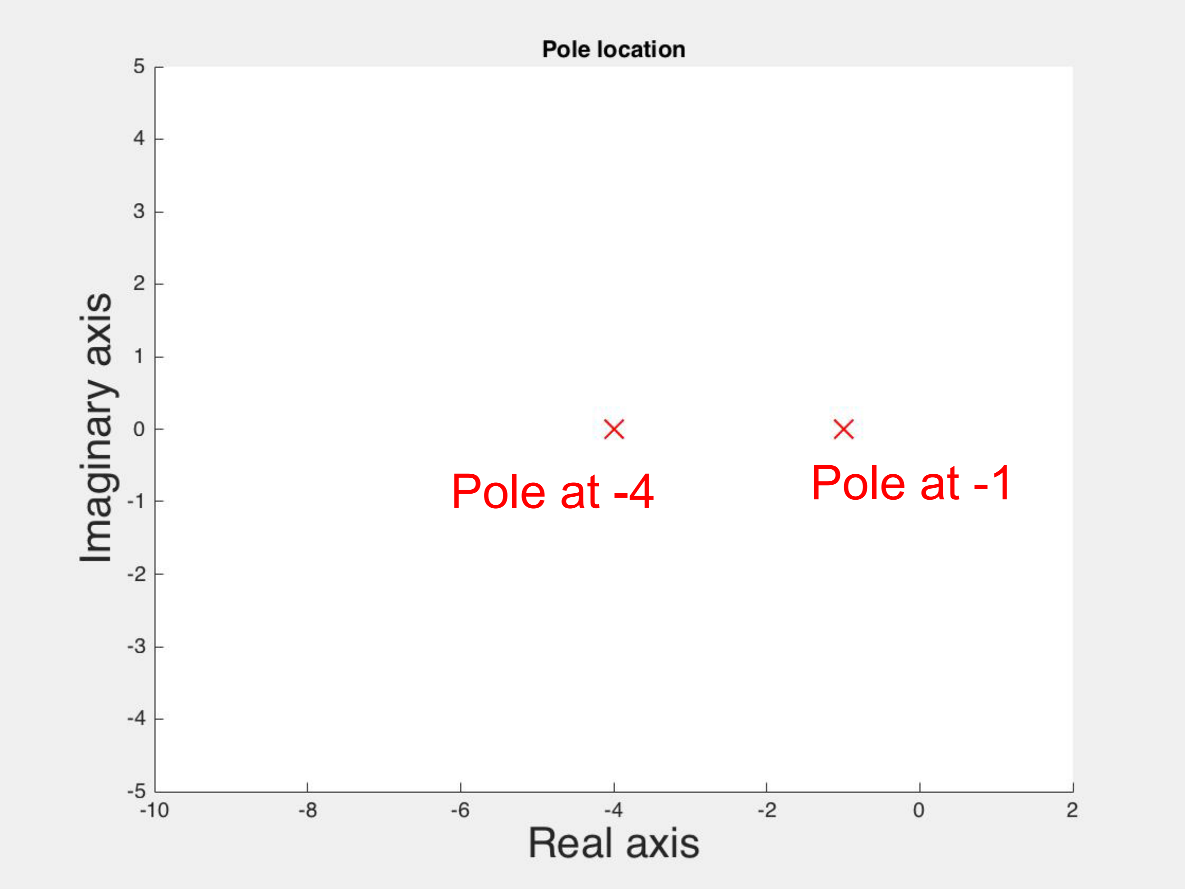
- Therefore location of poles give by $s = \dfrac{-(a+b) \pm \sqrt{(a+b)^2 - 4(ab+K)}}{2}$

- Thus some values of K will lead to complex solutions
- There are no zeros for this transfer function

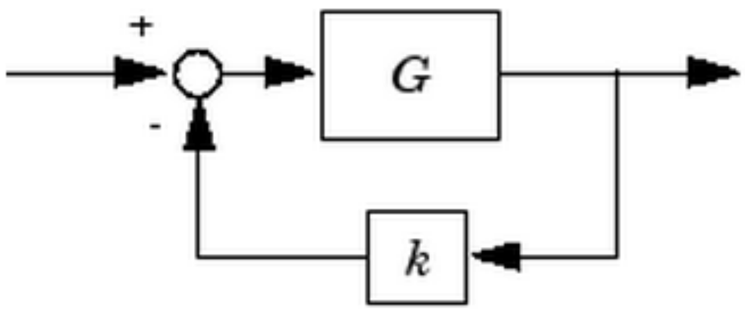
DIY Matlab plot poles for a range of K values

```
% points between 0 and 100
points=100;
Kp = linspace(0, 30, points);
a=1;
b=4;
% Prepare the new file.
vidObj = VideoWriter('rootLocus2Pole.avi');
open(vidObj);

% plot the pole
figure
hold on
title('Pole location');
h=xlabel('Real axis')
set(h,'FontSize', 20);
h=ylabel('Imaginary axis')
set(h,'FontSize', 20);
%set(gca,'nextplot','replacechildren');
axis([-10 2 -5 5])
for idx = 1:points
    pole =  (-(a+b)+ sqrt( (a+b)^2 - 4*(a*b+Kp(i
    poleC1 = complex(pole);
    pole =  (-(a+b)- sqrt((a+b)^2 - 4*(a*b+Kp(ic
    poleC2 = complex(pole);
    h=plot(poleC1, 'Xr');
    set(h,'MarkerSize', 10);
    h=plot(poleC2, 'Xr');
    set(h,'MarkerSize', 10);

    % Write each frame to the file.
    currFrame = getframe(gcf);
    writeVideo(vidObj,currFrame);
end
legend('Pole');
% Close the file.
close(vidObj);
```

$$P(s) = \frac{K}{(s+a)(s+b)}$$



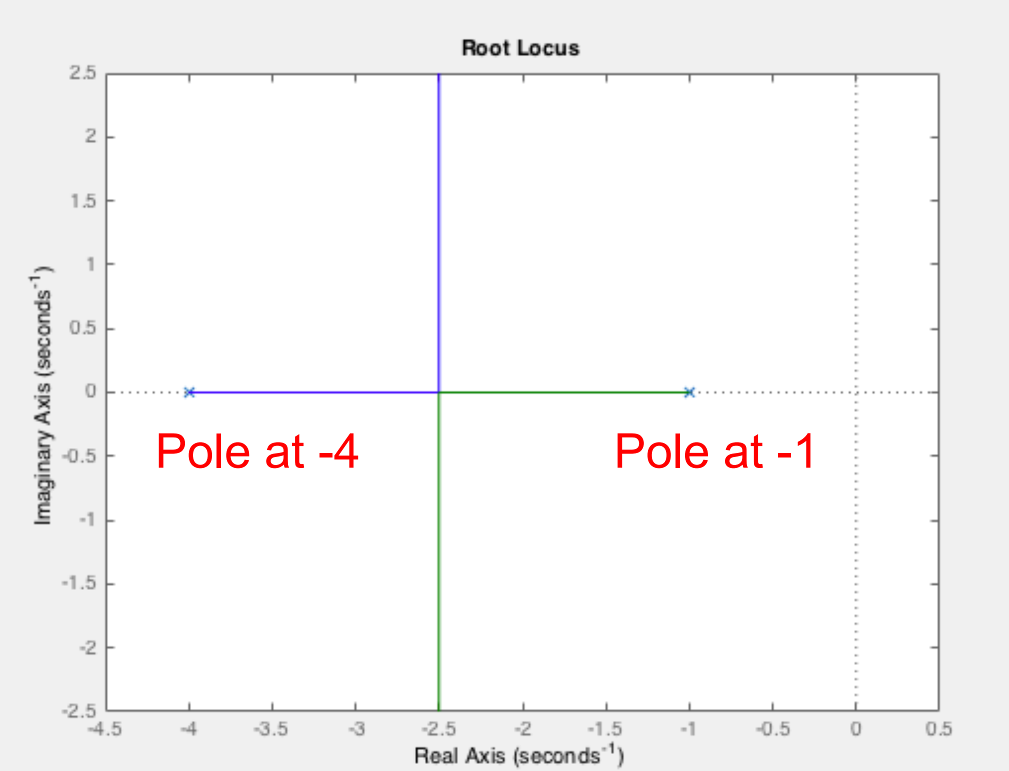
Using Matlab rlocus function



Again Matlab puts gain in feedback path and not in the forward path

$$P(s) = \frac{K}{(s+a)(s+b)}$$

```
% first system function for s term
s = tf('s');

% proportional gain init
k=1;
a=1;
b=4;

% build open loop system transfer function
sys = k/((s+a)*(s+b));

% calculate and plot the root locus of
% the open-loop  model
figure
rlocus(sys);
```



ROCO218: Control Engineering
Dr Ian Howard

Lecture 10

Two Root Locus sketching rules

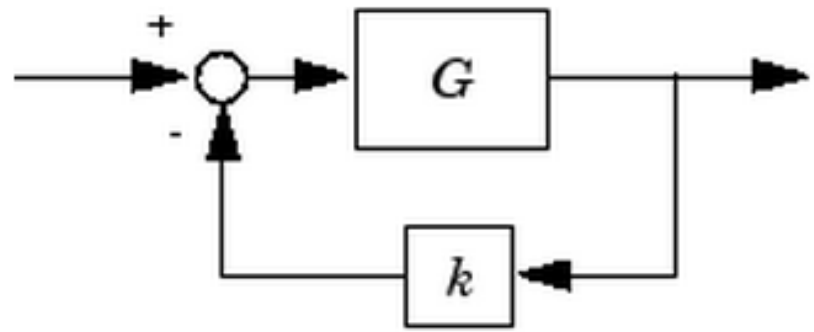
Root Locus sketching rules

- Sketching involves using a few mathematical properties of the transfer function to draw the main form of the root locus
- Matlab can do this but by hand sketch still can give/demand insights of the important features
- General rules for the construction of a root locus were developed by Evans.
- Explicit solutions are difficult to obtain for higher-order system
- However, in control design we are also interested in how to modify the dynamic response so that a system can meet the specifications for good control performance.
- Only consider hand determination of start and end points of plots here

Root Locus sketching rules

- Give the feedback system with open loop transfer function G(s)

$$CL(s)=\frac{G(s)}{\left(1+kG(s)\right)}$$



- Poles given when

$$1+kG(s)=0$$

- We can express the root locus problem in several equivalent ways
- Writing the transfer function as

$$G(s)=\frac{N(s)}{D(s)}$$

Numerator term

Denominator term

- Therefore we can write

$$1+k\frac{N(s)}{D(s)}=0$$

$$\Rightarrow D(s)+kN(s)=0$$

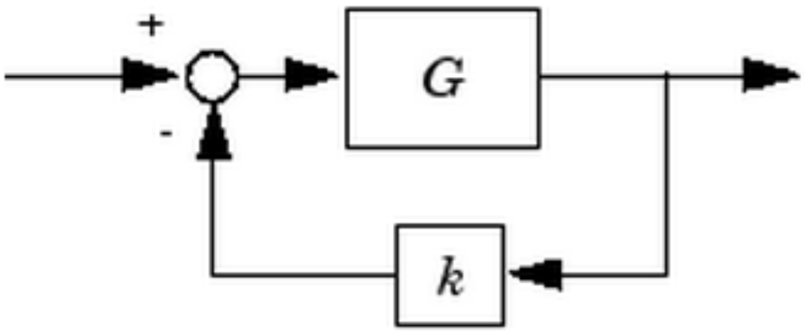
The equations above are sometimes referred to as "*the root locus form of a characteristic equation*."

The root locus is the set of values of *s* for which the above equations hold for some positive real value of *k*.

Root Locus sketch: Starting point open loop pole

- From the expression

$$D(s) + kN(s) = 0$$



- It can been seen that when k=0 we have

$$D(s) = 0$$

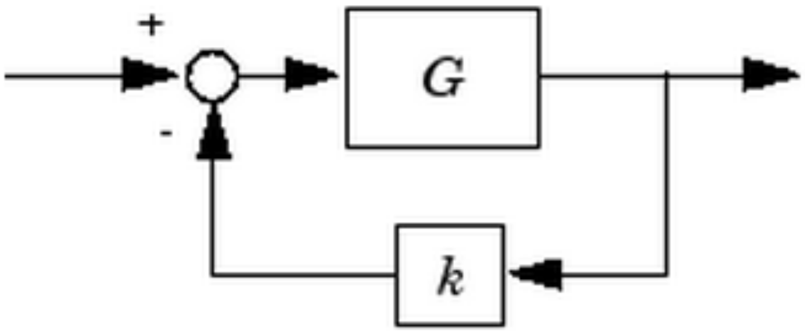
- So roots of the closed loop system are then the poles of G(s)

- **Rule 1: Starting Point (*k=0*)**
- The root locus starts at open loop poles.

Root Locus sketch: Ending point open loop zero

- From the expression

$$1 + k\frac{N(s)}{D(s)} = 0$$



- Dividing through by k

$$\Rightarrow \frac{1}{k} + \frac{N(s)}{D(s)} = 0$$

- It can been seen that when k ➔ ∞ the 1st term goes to zero so we have

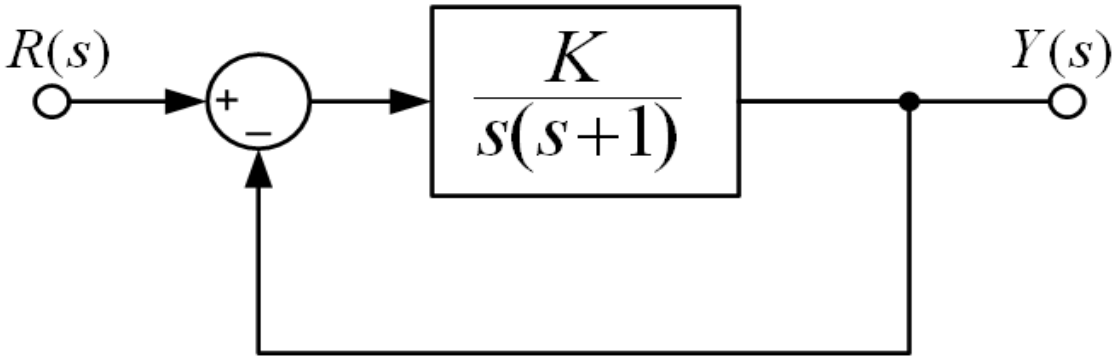
$$\frac{N(s)}{D(s)} = 0$$

- This expression has roots when N(s) = 0
- So the roots are the open loop zeros of G(s)

- **Rule 2: Ending Point (*k* ➔ ∞)**
- The root locus ends at open loop zeros.

Plotting a Root Locus by-hand example

Examine the following closed-loop system, with unity negative feedback.



The closed-loop transfer function is given as:

$$\frac{Y(s)}{R(s)} = \frac{K}{\underbrace{s^2 + s + K}}$$

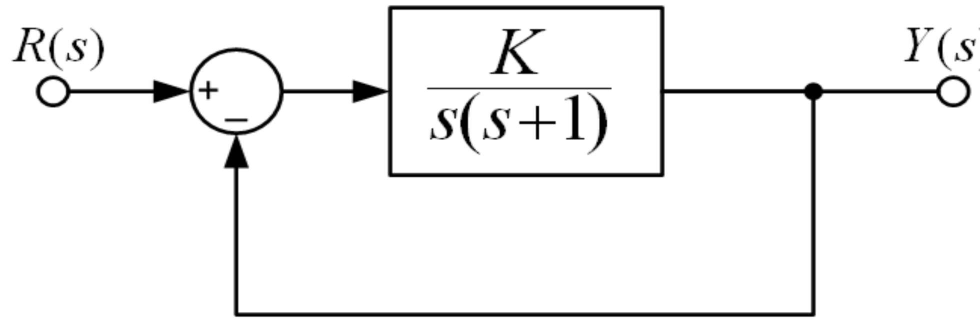
The roots of the characteristic equation are:

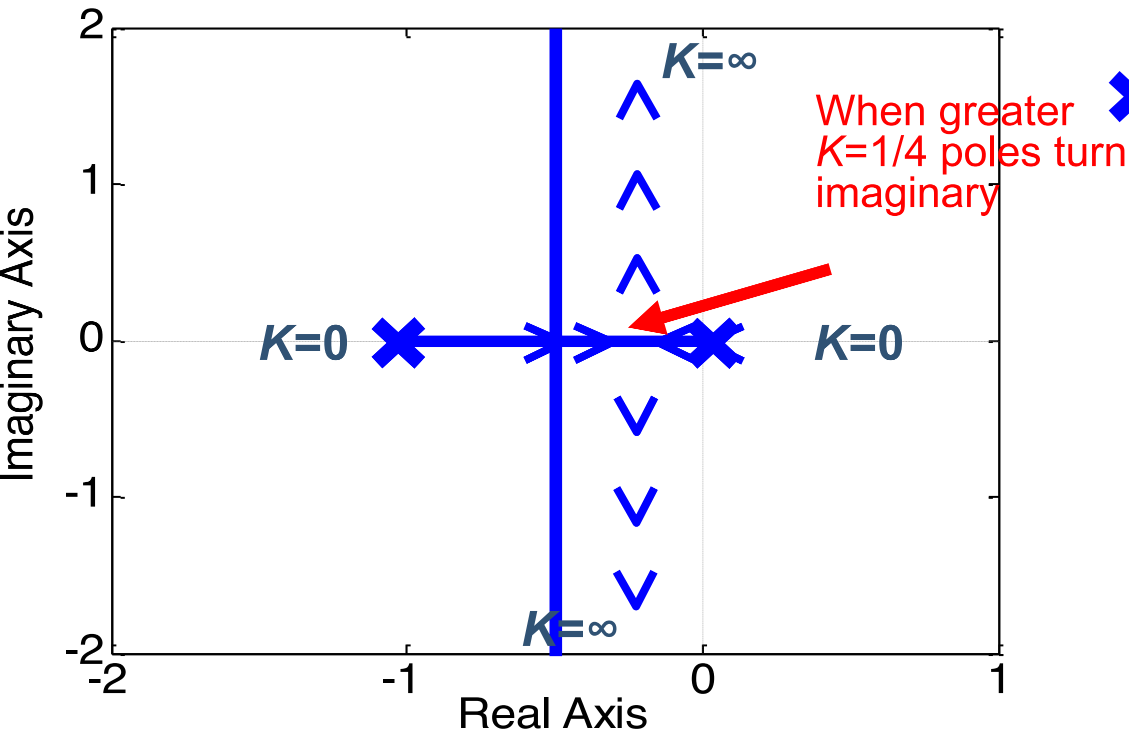
$$s_{1,2} = \frac{-1 \pm \sqrt{1-4K}}{2}$$

The characteristic equation - which is the denominator of the closed-loop transfer function

Plotting a Root Locus by-hand example

$$s_{1,2} = \begin{cases} -\frac{1}{2} \pm \frac{\sqrt{1-4K}}{2}, & 0 \le K \le \frac{1}{4} \\ -\frac{1}{2} \pm j\frac{\sqrt{4K-1}}{2}, & K > \frac{1}{4} \end{cases}$$





✖: Poles of open-loop transfer function

Open loop poles at 0 and -1

In this case no zeros so paths end up at +/- infinity on imaginary axis

Example: Examining some root locus plots

- Consider the open loop transfer function

$$H(s) = \frac{(s+7)}{s(s+5)(s+15)(s+20)}$$

- By inspection of the open-loop transfer function we see that when the gain k = 0 there are 4 closed-loop poles at:
- s = 0, s = -5, s = -15 & s = -20

- Similarly there is a closed-loop zeros at
- s = -7
- Since there must be 3 zeros in total the other two zeros must be at s = ∞

Examining some root locus plots

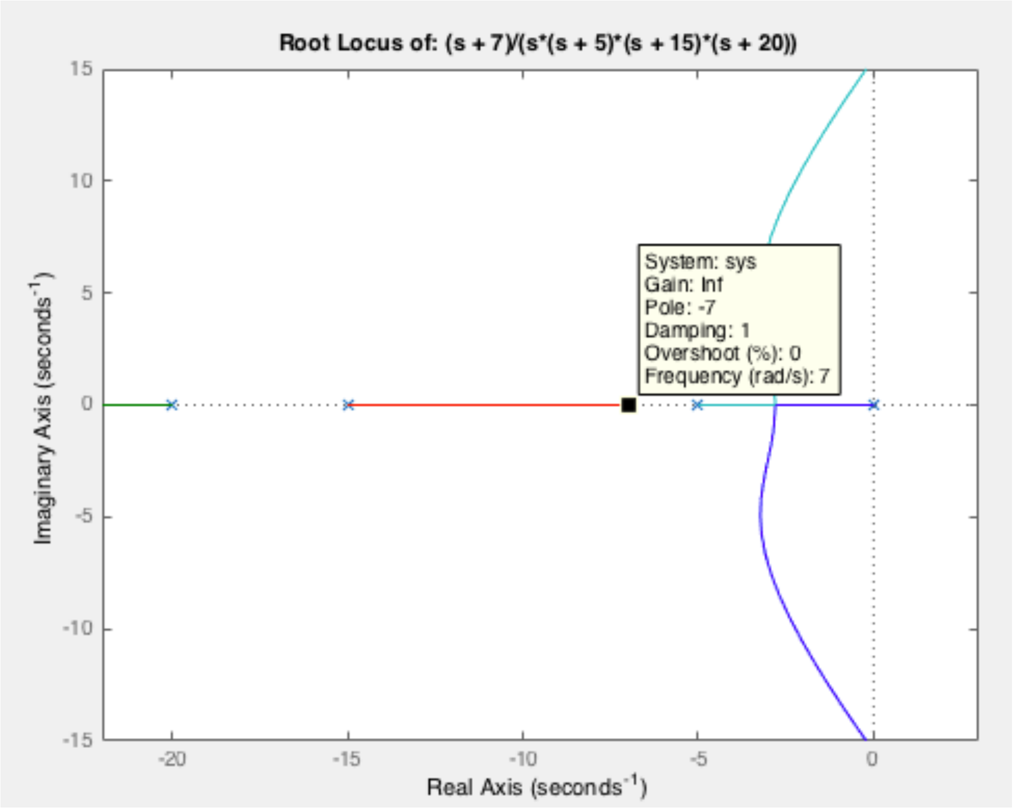
- Consider the open loop transfer function

$$H(s)=\frac{(s+7)}{s(s+5)(s+15)(s+20)}$$

- Now look at root locus plot using Matlab

```
% build Matlab open loop transfer fucntion
s = tf('s');
sys = (s + 7)/(s*(s + 5)*(s + 15)*(s + 20));

% Matlab root locus function
rlocus(sys)

% limt range of plot
axis([-22 3 -15 15])

% put on a title
title('Root Locus of: (s + 7)/(s*(s + 5)*(s + 15)*(s + 20))');
```

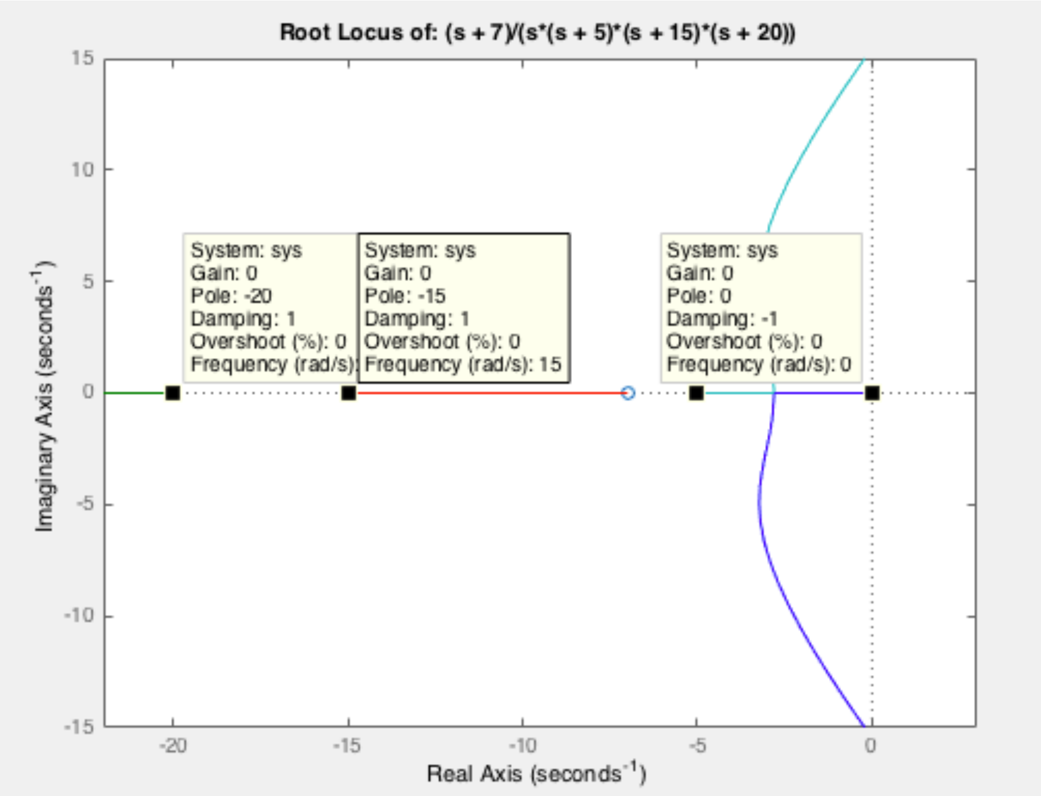
- This leads to the following plot



Examining some root locus plots

$$H(s) = \frac{(s+7)}{s(s+5)(s+15)(s+20)}$$

- As expected poles on the root locus plot when the gain is zero start at the open loop poles of the system:
- s = 0
- s = -5
- s = -15
- s = -20

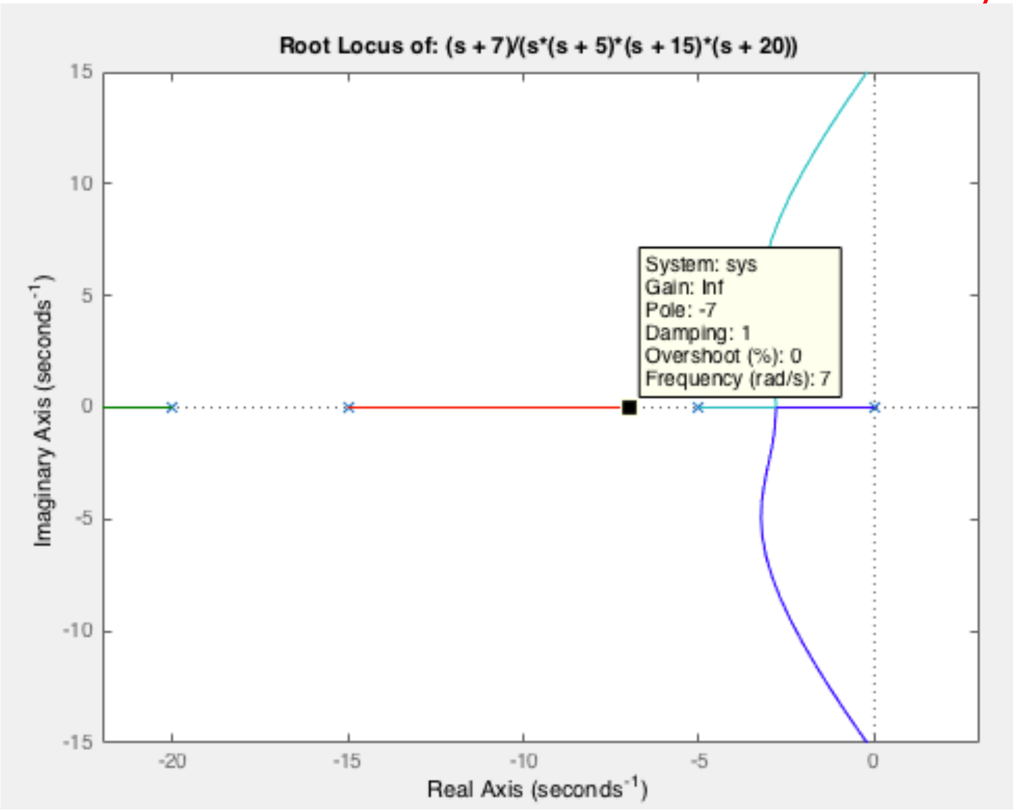


Examining some root locus plots

- As expected a pole on the root locus plot when the gain is zero ends at the open loop zero of the system:
- s = -7
- And two go off to infinity

$$H(s)=\frac{(s+7)}{s(s+5)(s+15)(s+20)}$$

Off to infinity

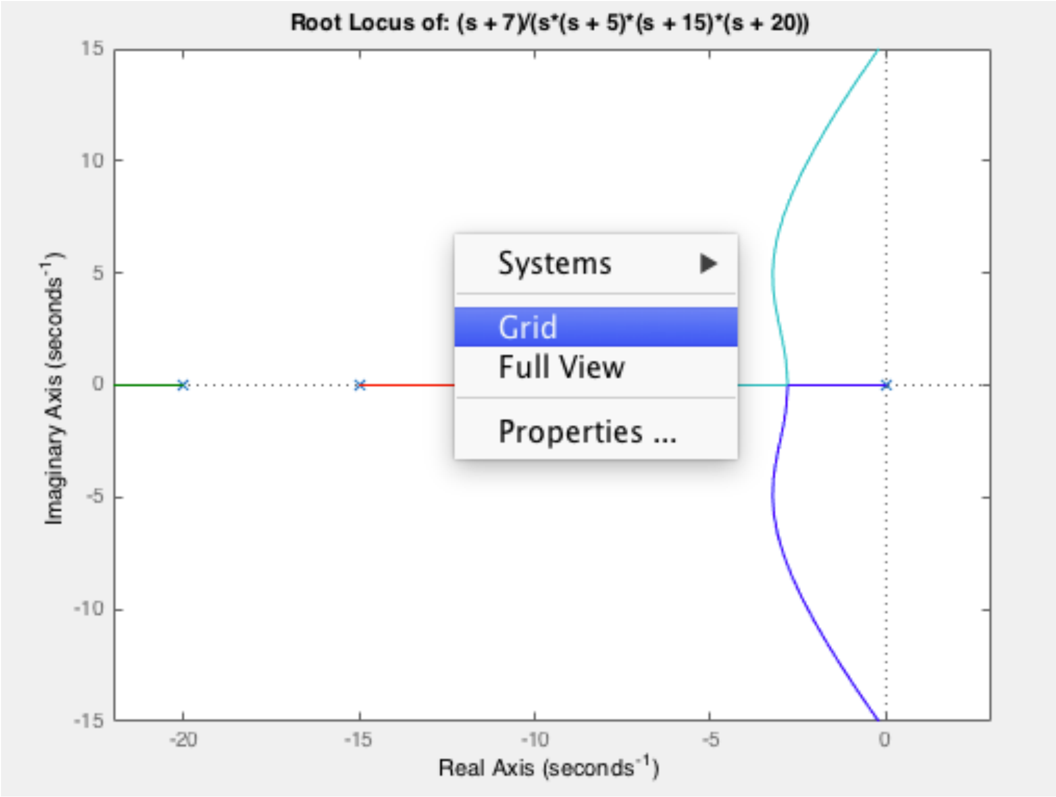


Off to infinity

Examining some root locus plots

- We can switch on a grid to examine damping

$$H(s) = \frac{(s+7)}{s(s+5)(s+15)(s+20)}$$

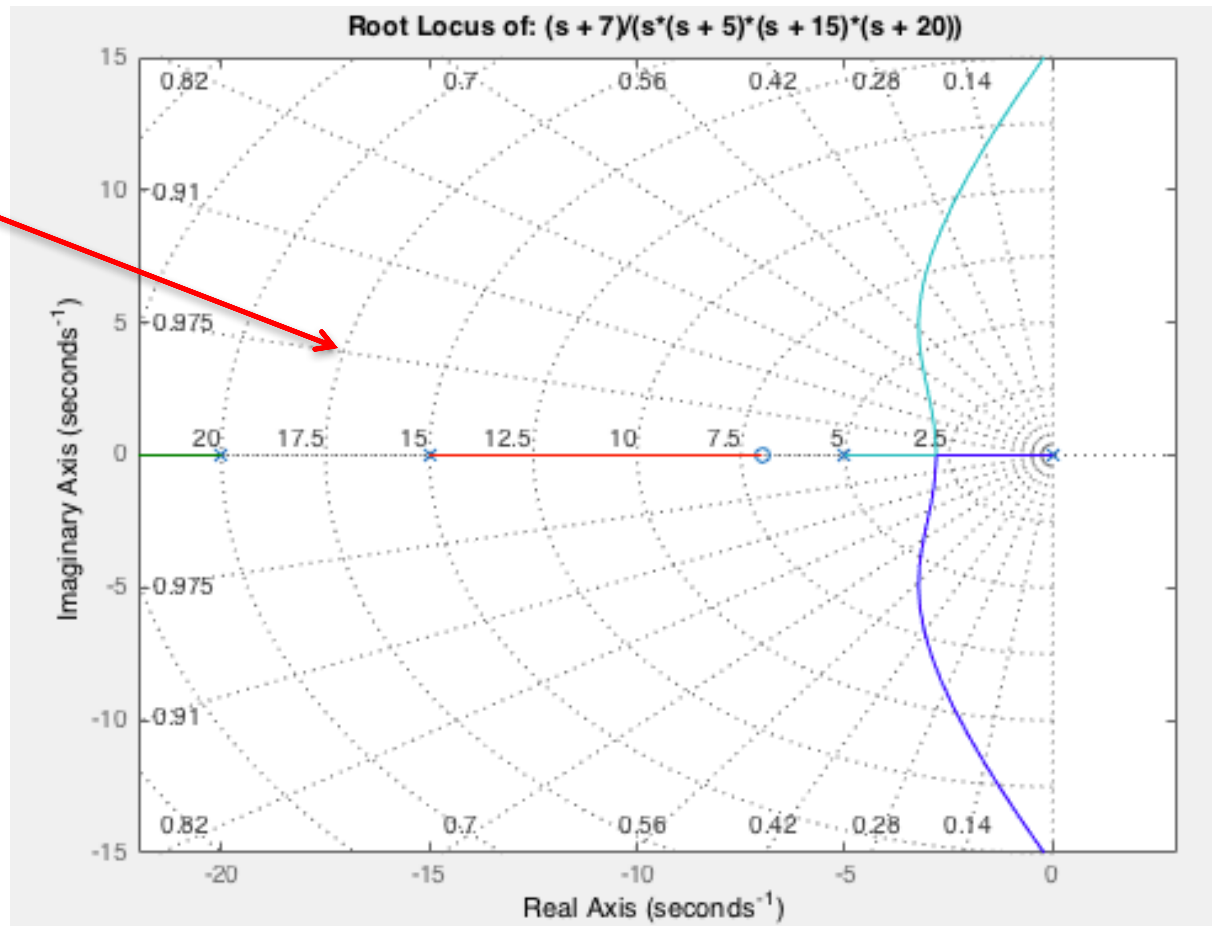


Examining some root locus plots

$$H(s)=\frac{(s+7)}{s(s+5)(s+15)(s+20)}$$

This gives lines from origin that represent constant damping ratio

For systems that approximate 2nd order systems, damping ratio and natural frequency can be directly read from root-locus plots



ROCO218: Control Engineering

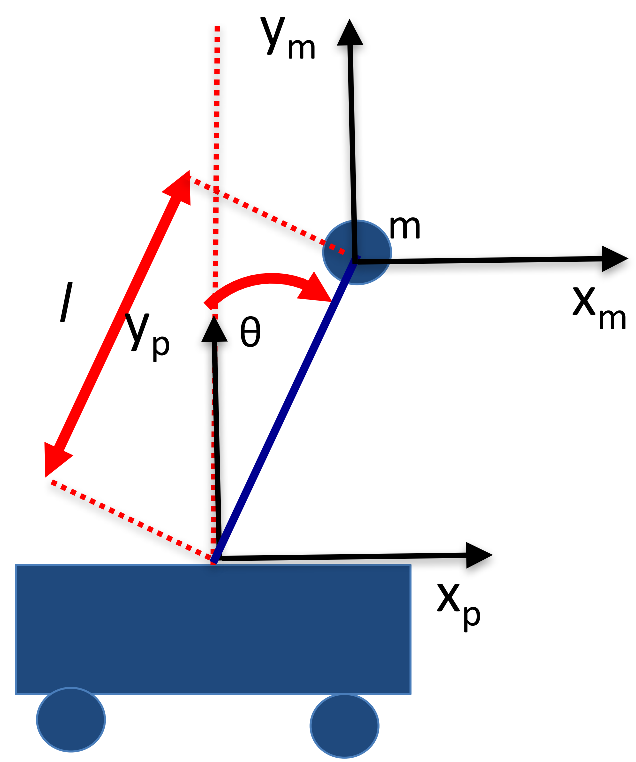
Lecture 10

Root locus of inverted pendulum

Rod inverted pendulum

For an inverted rod pendulum where

- The angle to the vertical is denoted by θ
- The coefficient of viciously is denoted by μ
- The mass of the pendulum is denoted by m
- The moment of inertial of the rod about the center of mass is denoted by I
- The length to the centre of mass is denoted by *l*
- The velocity of the pivot is given by V_p



- The linearized equation of motion is given by

$$\left(I + ml^2\right)\frac{d^2\theta}{dt^2} + \mu\frac{d\theta}{dt} = mgl\theta + ml\frac{d^2x_p}{dt^2}$$

Taking Laplace transforms

From dynamic equation

$$\left(I+ml^2\right)\frac{d^2\theta}{dt^2}+\mu\frac{d\theta}{dt}=mgl\theta+ml\frac{d^2x_p}{dt^2}$$

Take Laplace transforms assuming zero initial conditions.

$$\Rightarrow\left(I+ml^2\right)s^2\Phi(s)+\mu s\Phi(s)=mgl\Phi(s)+mls^2X_p(s)$$

$$\Rightarrow\left(\left(I+ml^2\right)s^2+\mu s-m\,\mathrm{lg}\right)\Phi(s)=mls^2X_p(s)$$

$$\Rightarrow X_p(s)=\frac{\left(\left(I+ml^2\right)s^2+\mu s-m\,\mathrm{lg}\right)\Phi(s)}{mls^2}$$

Dividing through by ml

$$\Rightarrow X_p(s)=\frac{\left(\left(\frac{I}{ml}+l\right)s^2+\frac{\mu s}{ml}-\mathrm{g}\right)\Phi(s)}{s^2}$$

Deriving the transfer function

Writing velocity as $V_p(s) = sX_p(s)$

$$\Rightarrow V_p(s) = s\frac{\left(\left(\frac{I}{ml}+l\right)s^2+\frac{\mu s}{ml}-\mathrm{g}\right)\Phi(s)}{s^2} = \frac{\left(\left(\frac{I}{ml}+l\right)s^2+\frac{\mu s}{ml}-\mathrm{g}\right)\Phi(s)}{s}$$

So the transfer function for output angle for velocity input is given by

$$\Rightarrow \frac{\Phi(s)}{V_p(s)} = \frac{s}{\left(\left(\frac{I}{ml}+l\right)s^2+\frac{\mu s}{ml}-\mathrm{g}\right)}$$

$$\boxed{= \frac{sml}{\left(\left(I+ml^2\right)s^2+\mu s - ml\,\mathrm{g}\right)}}$$

Interpreting the transfer function

We now re-write transfer function with a unity term in front of the highest power of s

$$\frac{\Phi(s)}{V_p(s)} = \frac{sml}{\left(\left(I+ml^2\right)s^2 + \mu s - ml\,\mathrm{g}\right)} = \frac{sml \big/ \left(I+ml^2\right)}{\left(s^2 + s\left(\mu \big/ \left(I+ml^2\right)\right) - \left(ml\,\mathrm{g} \big/ \left(I+ml^2\right)\right)\right)}$$

Comparing with canonical form we can find natural frequency by inspection

$$\frac{sk}{[s^2 + 2\xi w_n s - w_n^2]} \quad \Rightarrow \omega_n = \sqrt{ml\,\mathrm{g} \big/ \left(I+ml^2\right)}$$

NB: We could also find damping ratio too

Since $2\pi f_n = \omega_n$

$$\Rightarrow 2\pi f_n = \sqrt{ml\,\mathrm{g} \big/ \left(I+ml^2\right)} \quad \Rightarrow f_n = \frac{1}{2\pi}\sqrt{ml\,\mathrm{g} \big/ \left(I+ml^2\right)}$$

Point mass condition

From expression for natural frequency

$$f_n = \frac{1}{2\pi}\sqrt{\frac{ml\,g}{\left(I + ml^2\right)}}$$

So for point mass there is no MoI about the centre of mass

$$I = m.0 = 0$$

$$\Rightarrow f_n = \frac{1}{2\pi}\sqrt{\frac{ml\,g}{\left(0 + ml^2\right)}}$$

$$\Rightarrow f_n = \frac{1}{2\pi}\sqrt{\frac{g}{l}}$$

Same result as for a simple point mass pendulum!

Rod pendulum condition

From expression for natural frequency

$$f_n = \frac{1}{2\pi}\sqrt{mlg \big/ \left(I + ml^2\right)}$$

For rod about its CoG with rod length 2l, since l is pivot distance to CoG

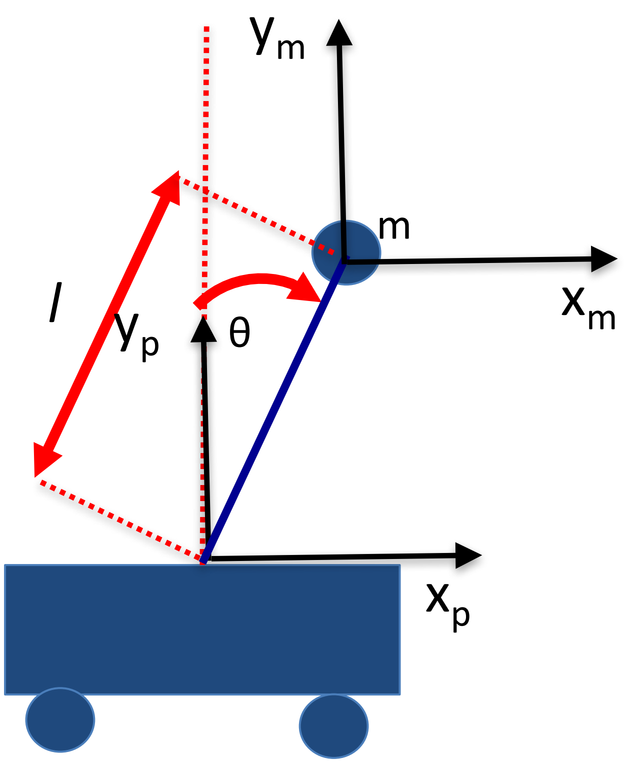
$$I = m\frac{(2l)^2}{12} \quad = \frac{4l^2}{12}$$

$$\Rightarrow f_n = \frac{1}{2\pi}\sqrt{mlg \Big/ \left(\left(4ml^2 \big/ 12\right) + ml^2\right)} = \frac{1}{2\pi}\sqrt{mlg \Big/ \left(16ml^2 \big/ 12\right)} = \frac{1}{2\pi}\sqrt{3g \big/ 4l}$$

$$\Rightarrow f_n = \frac{1}{2\pi}\sqrt{3g \big/ 2l_{full}}$$

Where l_{full} is the full length of the rod (= 2x distance to CoG)

Evaluation with pararmeter values

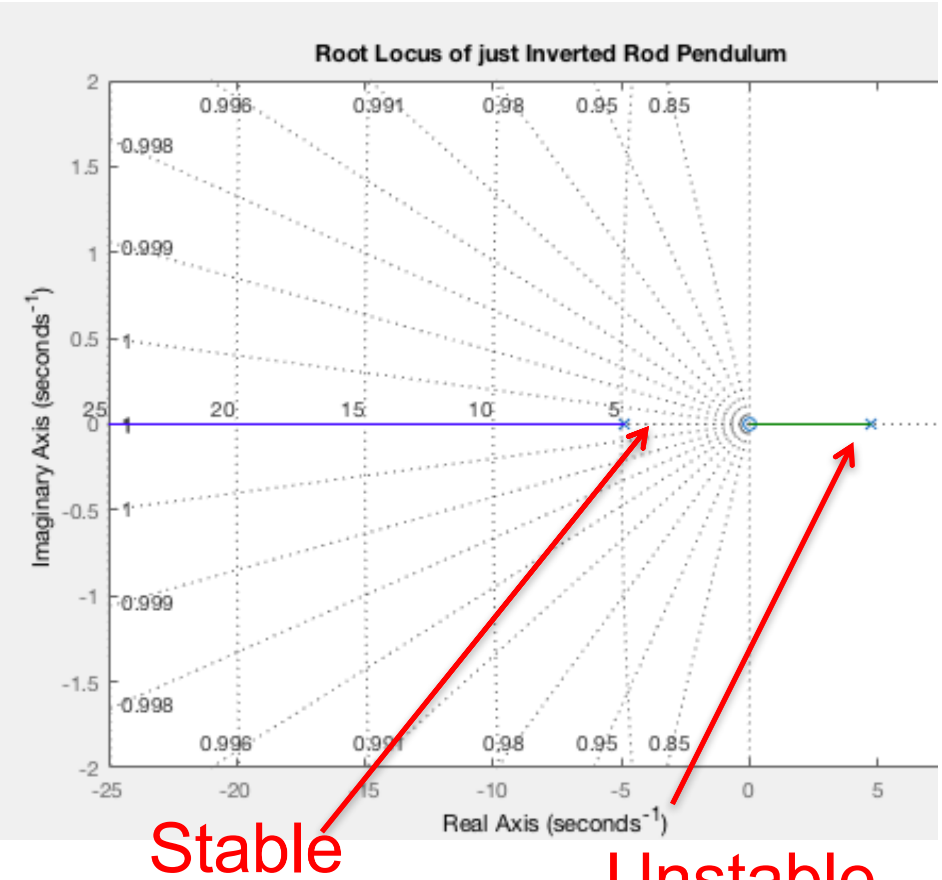


We will now consider the following values:

- Length of pendulum rod 0.64 m
- Mass of pendulum rod 0.314 kg
- Coefficient of viscous friction: 0.005

Matlab root locus plant only analysis

We can now perform Matlab root locus analysis of the uncontrolled pendulum

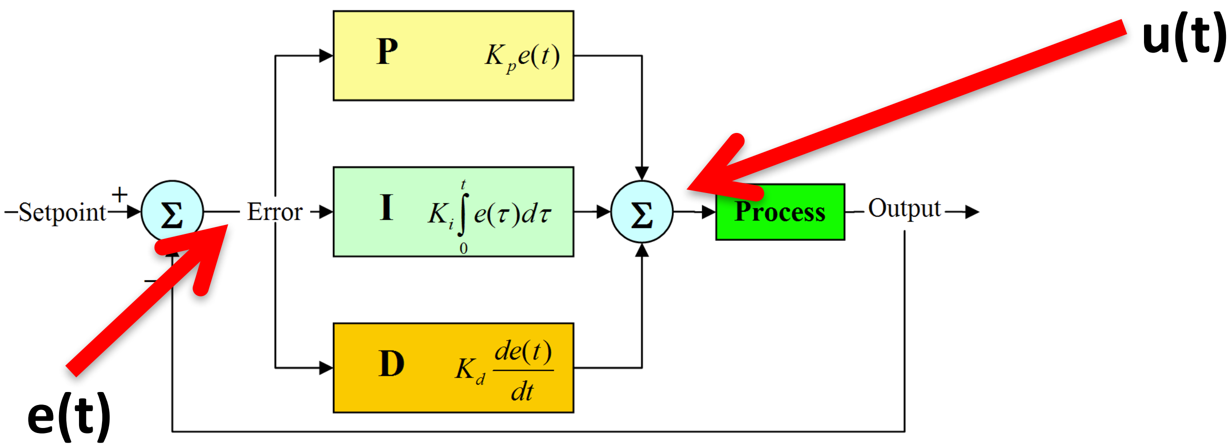



```
% basic position controlled pendulum
l = 0.64;           % 64cm rod
l2 =l/2;            % rod mid point
g = 9.81;           % acc due to gravity
u = 0.005;          % add some damping
m = 0.314;          % 0.314Kg pendulum mass
I = (m * l^2)/12;% pendulum inertia of rod
                    % about its midpoint

% build angle/velocity transfer function
s = tf('s');
n = ( s * m * l2 );
d = ( s^2 * (I+m * l2 ^ 2) + u * s - m * l2 * g );
sys =  n / d;
disp(sys)

figure
hold on
rlocus(sys)
title('Root Locus of just Inverted Rod Pendulum ');
```

```
sys =

            0.1005 s
  ------------------------------
  0.04287 s^2 + 0.005 s - 0.9857
```

With only proportional gain the pendulum system is unstable and cannot be controlled!

Remember: PID differential equation



- Relationship between input error e(t) and output control signal u(t) is thus captured by the differential equation:

$$u(t) = K_p e(t) + K_i \int e(t)\,dt + K_d \frac{de}{dt}$$

Remember: Transfer function of PID controller

- Differential equation that described PID controller

$$u(t) = K_p e(t) + K_i \int e(t)\,dt + K_d \frac{de}{dt}$$

- Taking Laplace transformations and rearranging gives the transfer function for the PID controller:

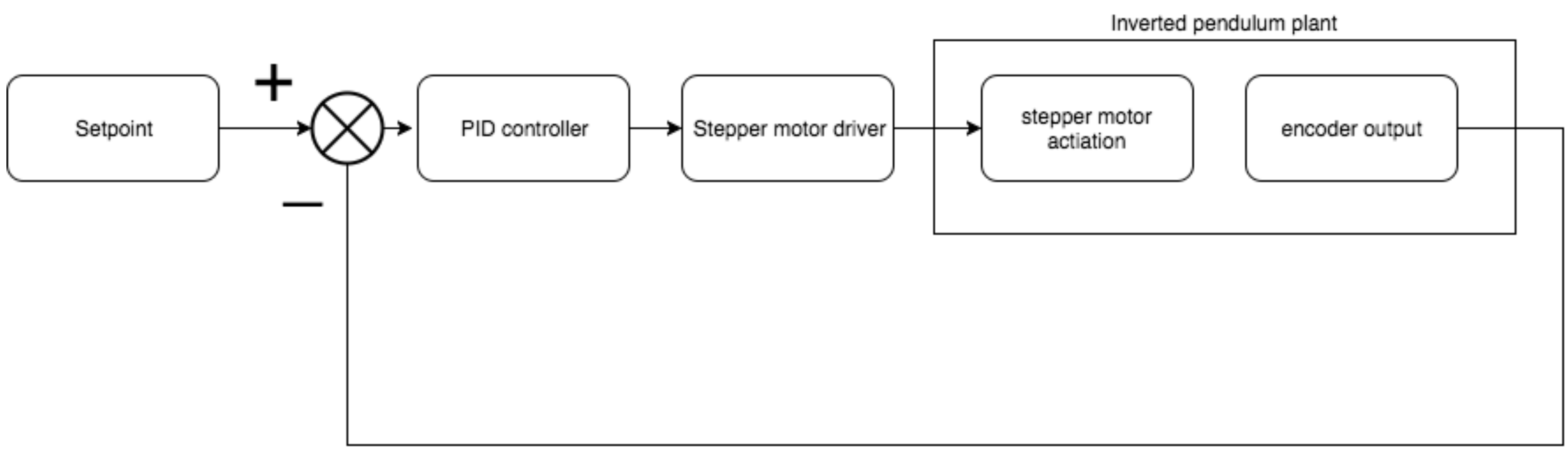
$$U(s) = K_p E(s) + K_i \frac{E(s)}{s} + K_d s E(s)$$

$$\Rightarrow U(s) = E(s)\left(K_p + \frac{K_i}{s} + K_d s\right)$$

$$\Rightarrow \frac{U(s)}{E(s)} = K_p + \frac{K_i}{s} + K_d s$$

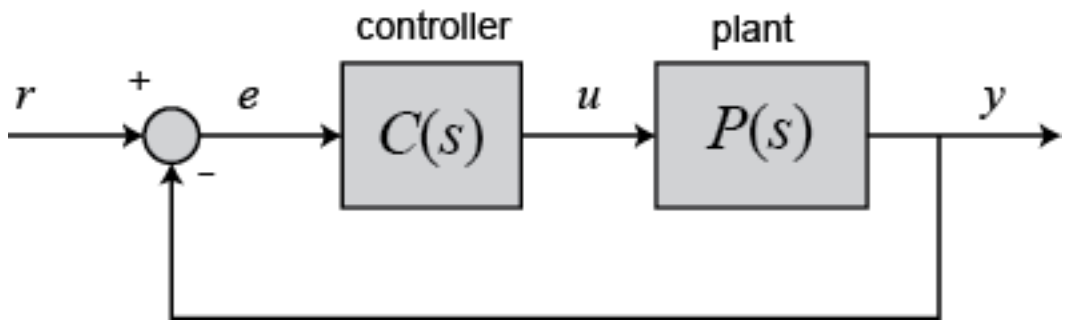
PID control of an inverted pendulum

- Want to use feedback with PID controller to stabilize pendulum
- Overall system flow graph is as follows



- The open loop forward path is therefore due to the serial combination of the PID controller and the inverted pendulum plant

PID control of an inverted pendulum



PID controller has the transfer function

$$C(s) = K_p + \frac{K_i}{s} + K_d s$$

Inverted pendulum plant has the transfer function

$$P(s) = \frac{sml}{\left(\left(I + ml^2\right)s^2 + \mu s - ml\,\mathrm{g}\right)}$$

Product of these series elements gives

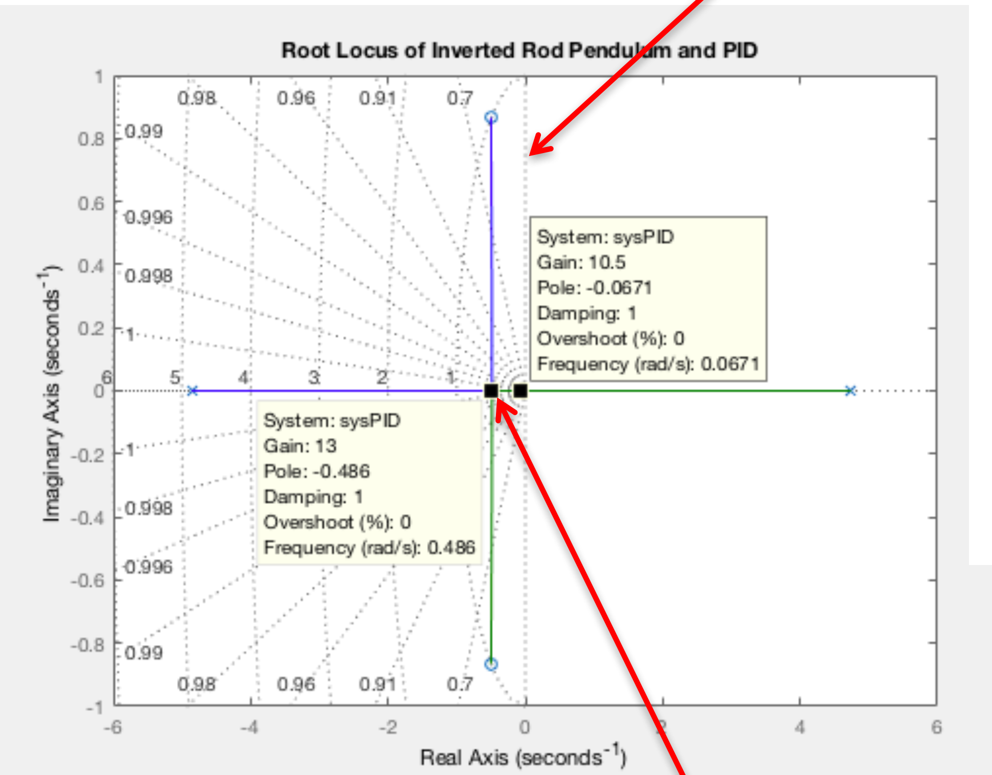
$$C(s)P(s) = \left(K_p + \frac{K_i}{s} + K_d s\right)\frac{sml}{\left(\left(I + ml^2\right)s^2 + \mu s - ml\,\mathrm{g}\right)}$$

To design a PID controller we need to find suitable K_p, K_i and K_d gains

Matlab root locus with PID controller

We can now perform Matlab root locus analysis of the PID controlled pendulum

Pole becomes stable as locus crosses imaginary axis



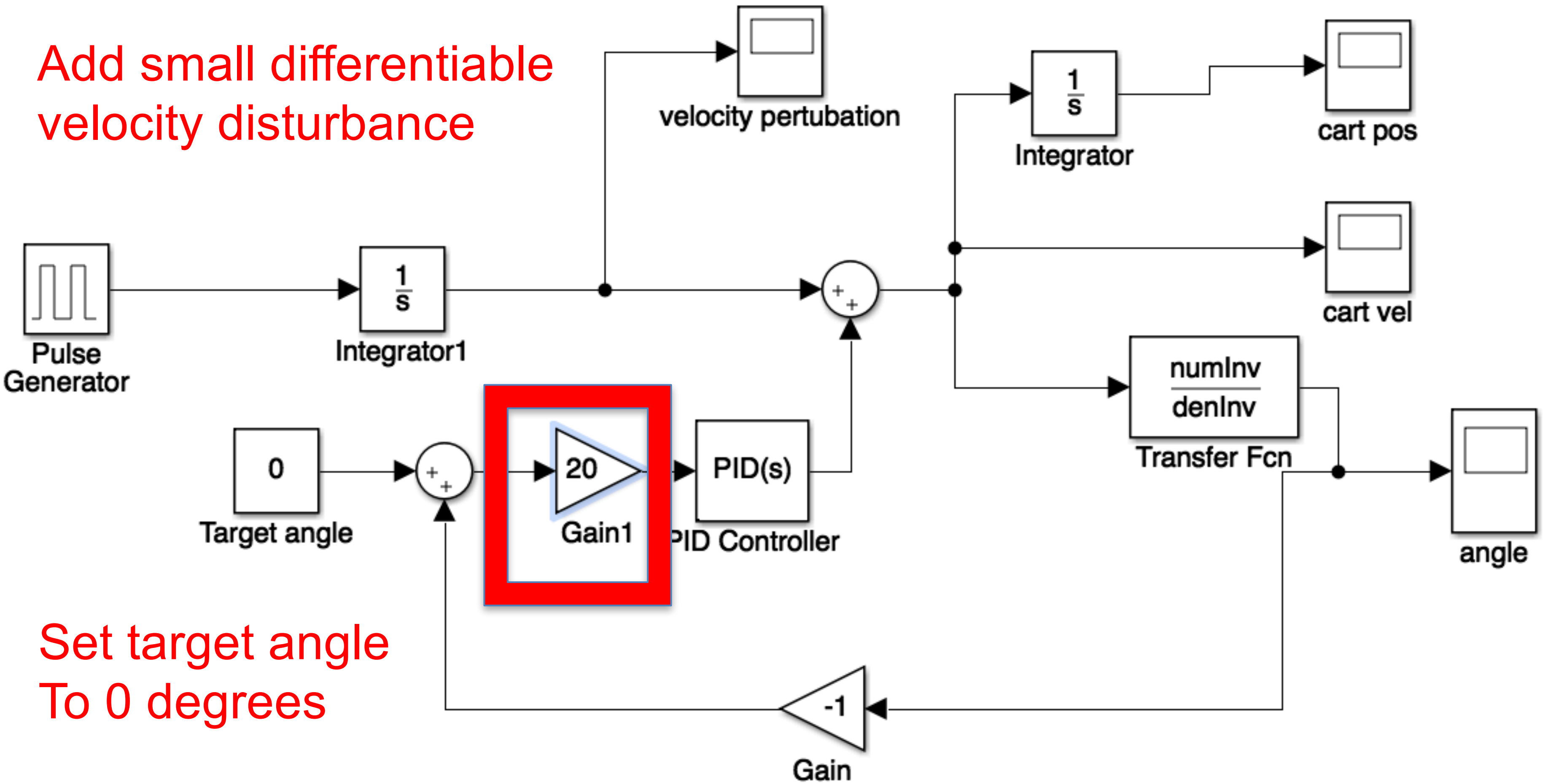
Here gain is about 13 after which complex poles are formed

```
% build angle/velocity transfer function
s = tf('s');
n = ( s * m * l2 );
d = ( s^2 * (I+m * l2 ^ 2) + u * s - m * l2 * g );
sys =  n / d;

% simple unity gain PID controller
PID =  (s + 1 + 1/s);

% plot rlocus for  open loop function with PID
figure
sysPID = sys * PID;
sysPID
rlocus(sysPID);
title('Root Locus of Inverted Rod Pendulum and PID ');

numInv = sysPID.Numerator{1};
denInv = sysPID.Denominator{1};
```

```
sysPID =

  0.1005 s^3 + 0.1005 s^2 + 0.1005 s
  ----------------------------------
  0.04287 s^3 + 0.005 s^2 - 0.9857 s
```

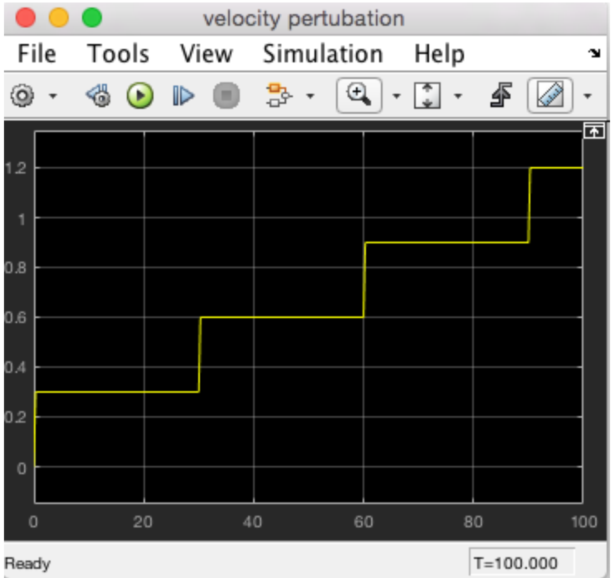
Simulink: PID with stable gain

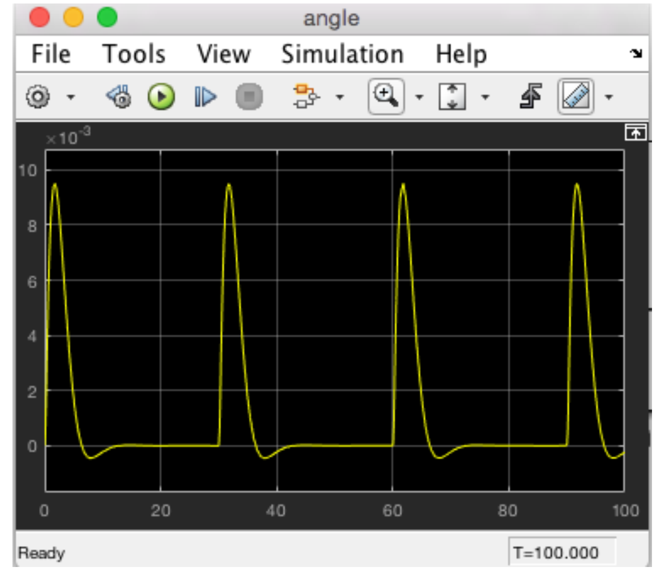
Add small differentiable velocity disturbance

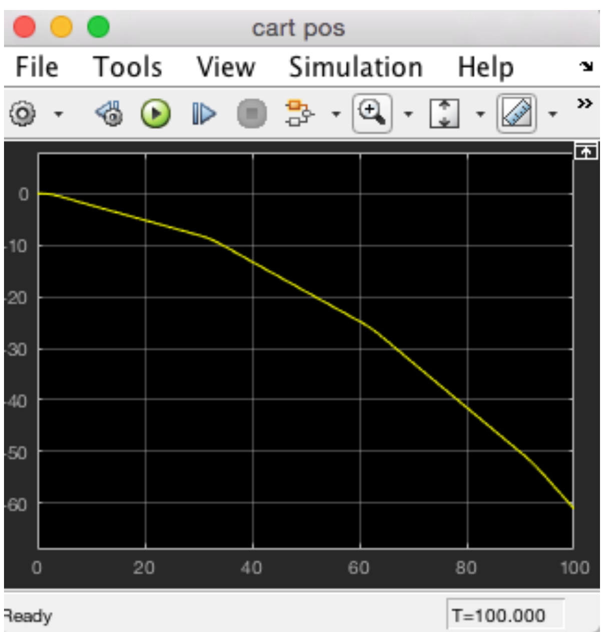


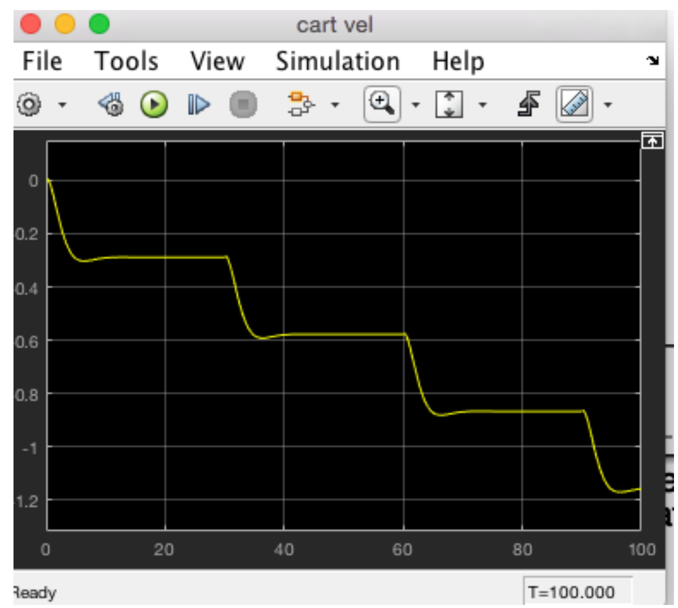
Set target angle
To 0 degrees

Signals: stable gain = 20



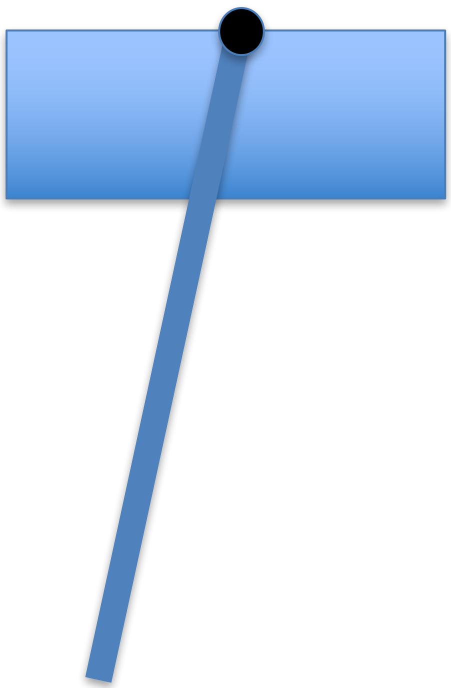






Normal hanging-down rod pendulum

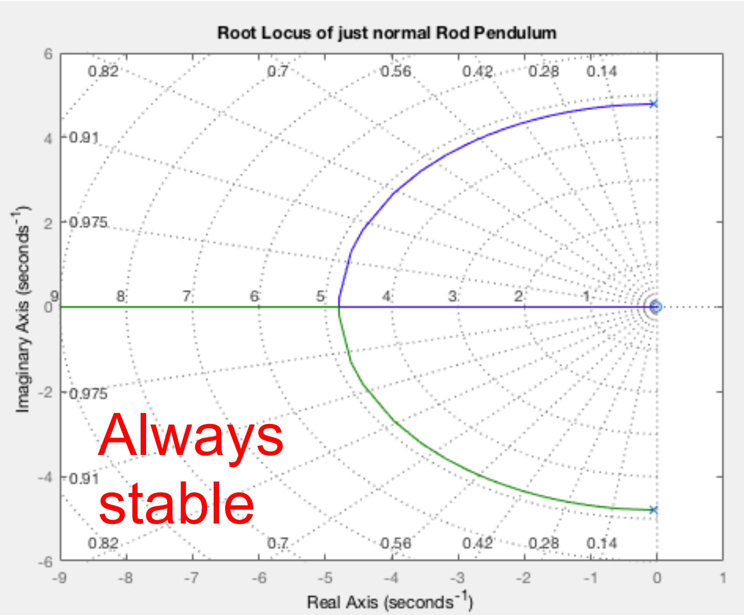
This time effect of gravity assists movement to stable equilibrium



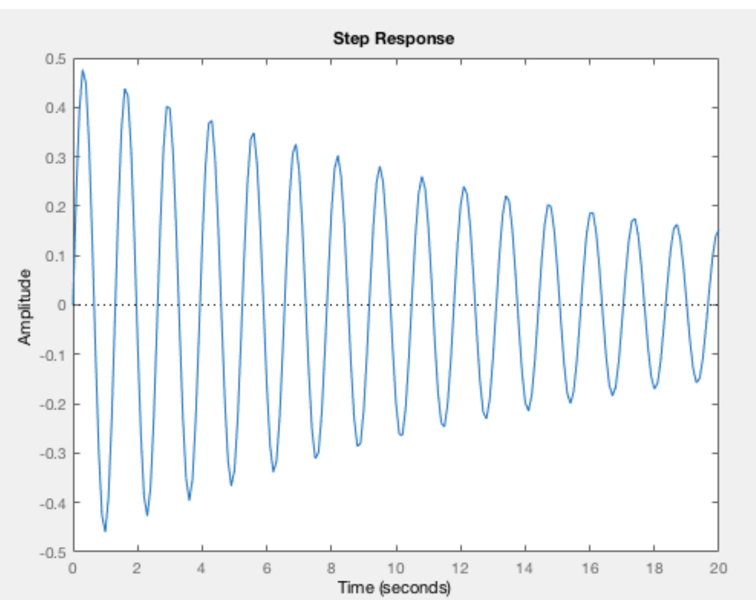
Notice change of sign of gravity term

$$\Rightarrow \frac{V_p(s)}{\Phi(s)} = \frac{sml}{\left(\left(I + ml^2\right)s^2 + \mu s \boxed{+ ml\,g}\right)}$$

Matlab root locus normal rod pendulum




```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% if not inverted then gravity restore rather than disrups
% so change sign of gravity term
% build angle/velocity transfer function
s = tf('s');
n = ( s * m * l2 );
d = ( s^2 * (I+m * l2 ^ 2) + u * s + m * l2 * g );
sys =  n / d;
sys
figure
hold on
rlocus(sys)
title('Root Locus of just normal Rod Pendulum ');

figure
title('Open loop step response normal pendulum')
step(sys, 0:0.1:20)
stepinfo(sys)
```

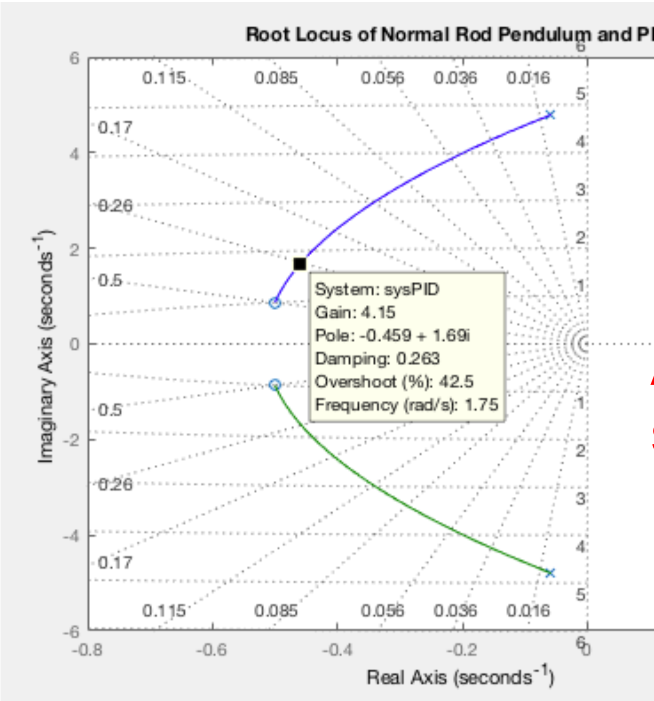


Open loop response is a damped sinusoid

```
        RiseTime: 0
    SettlingTime: 67.1770
     SettlingMin: -0.4616
     SettlingMax: 0.4796
       Overshoot: Inf
      Undershoot: Inf
            Peak: 0.4796
        PeakTime: 0.3276
```

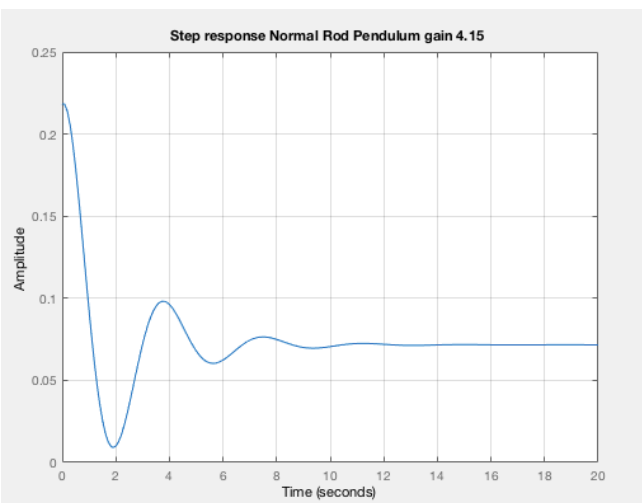
```
sys =

               0.1005 s
  ----------------------------
  0.04287 s^2 + 0.005 s + 0.9857
```

Matlab PID root locus normal rod pendulum



Always stable

```
% if not inverted then gravity restore rather than disrups
% so change sign of gravity term
% build angle/velocity transfer function
s = tf('s');
n = ( s * m * l2 );
d = ( s^2 * (I+m * l2 ^ 2) + u * s + m * l2 * g );
sys =  n / d;

% simple unity gain PID controller
PID =  (s + 1 + 1/s);

% plot rlocus for full open loop function
figure
sysPID = sys * PID;
sysPID
rlocus(sysPID);
title('Root Locus of Normal Rod Pendulum and PID ');

numNor = sys.Numerator{1};
denNor = sys.Denominator{1};
```

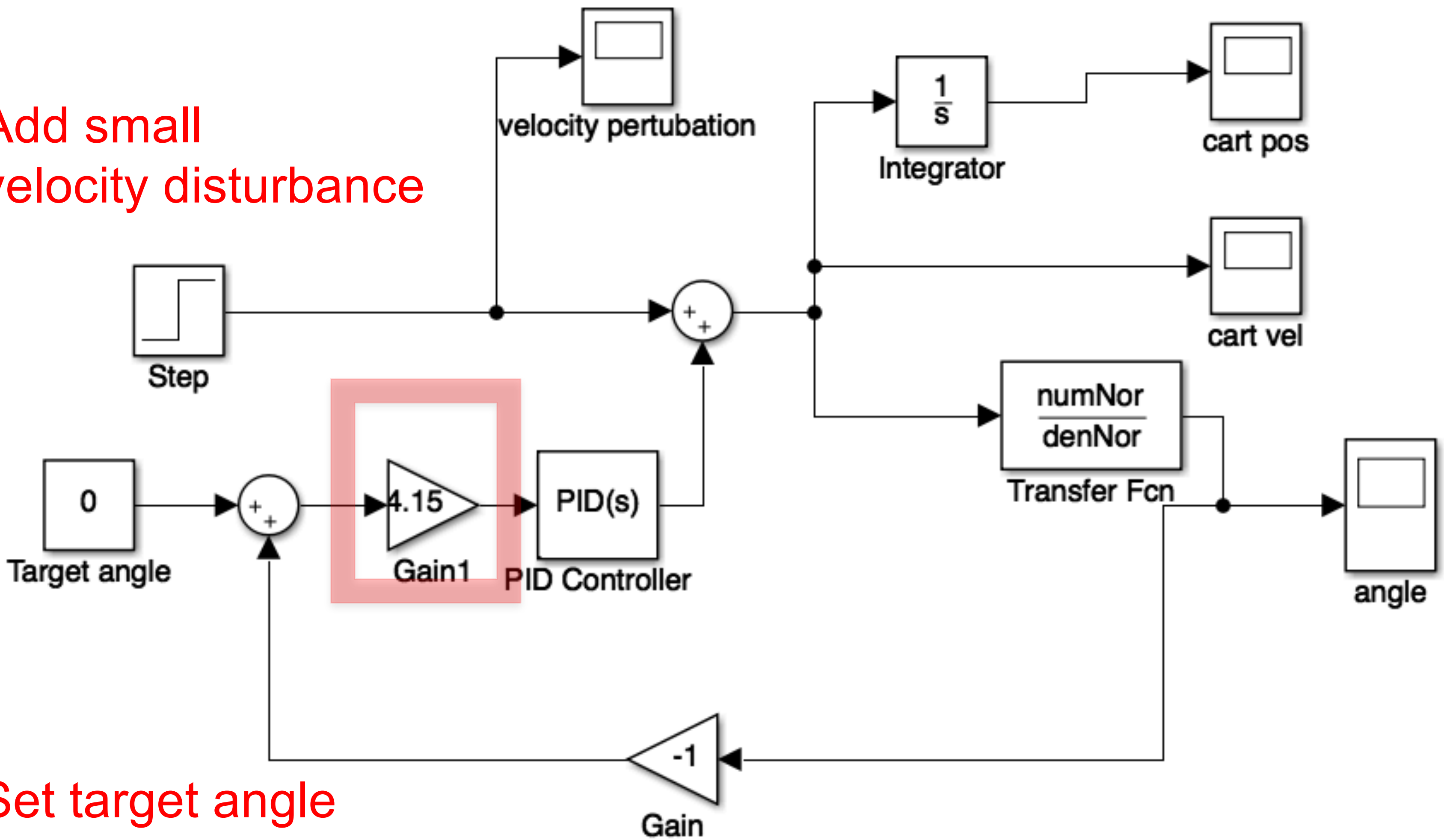


Step response

```
sysPID =

  0.1005 s^3 + 0.1005 s^2 + 0.1005 s
  ----------------------------------
  0.04287 s^3 + 0.005 s^2 + 0.9857 s
```

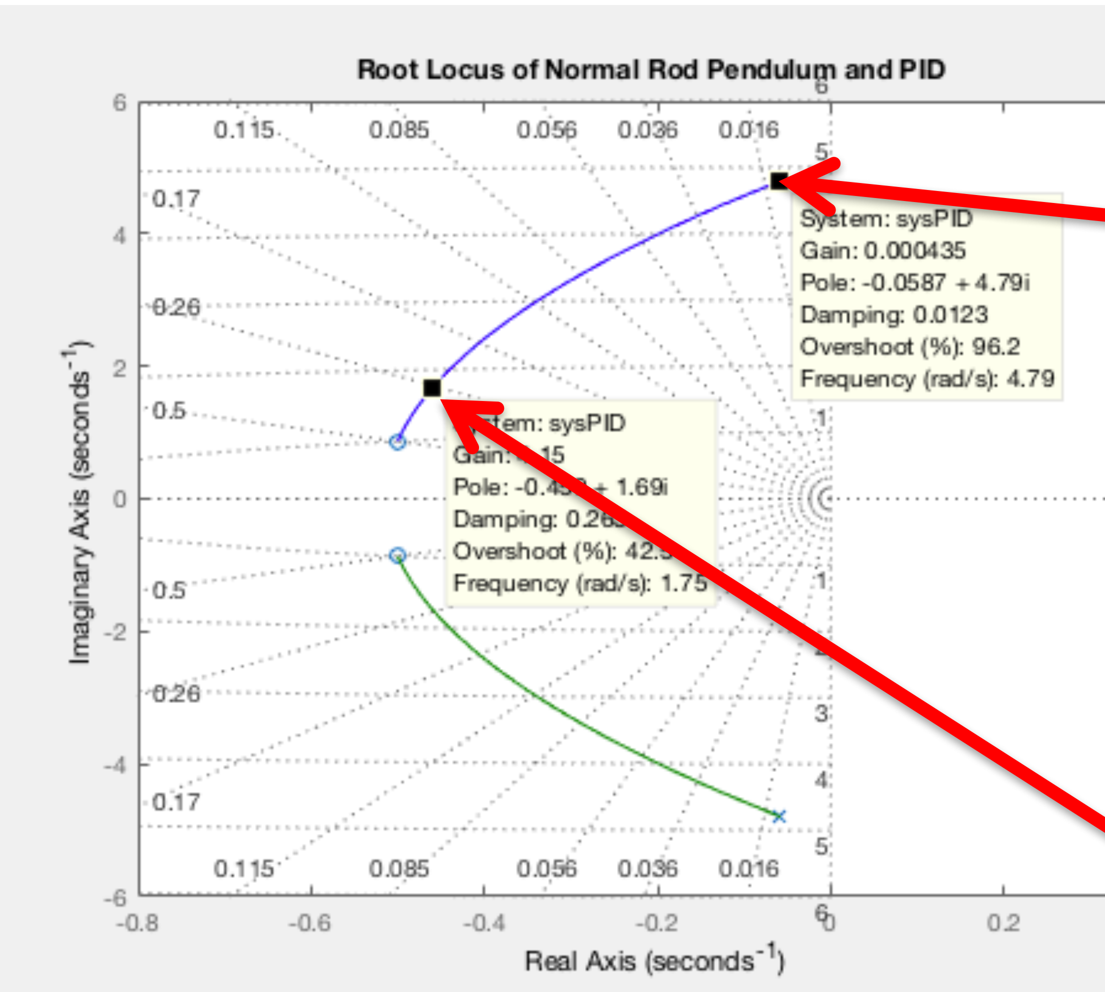
Simulink: PID with 4.15 gain

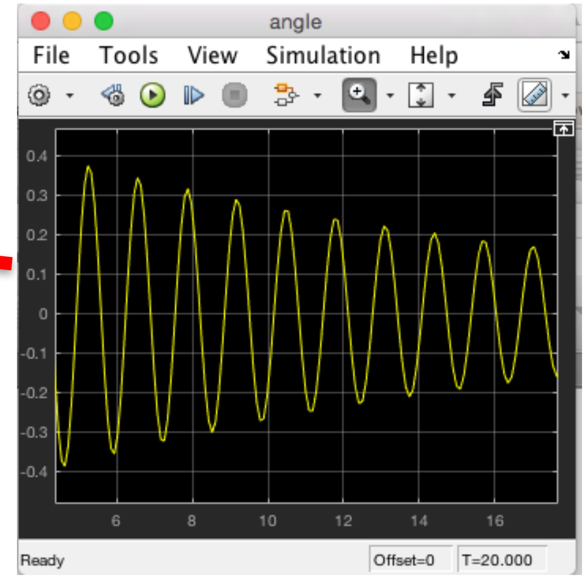
Add small
velocity disturbance

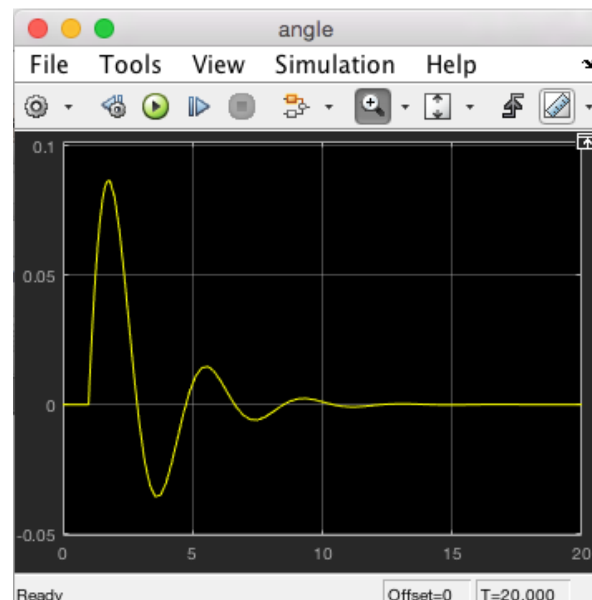


Set target angle
To 0 degrees

Normal rod pendulum angle







Interlude

10 minute break

ROCO218: Control Engineering
Dr Ian Howard

Lecture 10

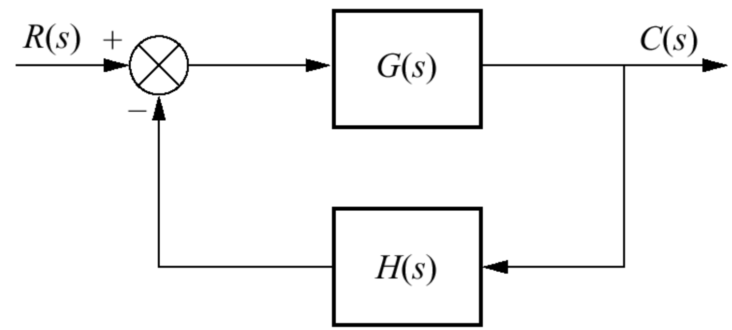
Nyquist stability criterion

Nyquist Criterion background concepts

- Bode stability criterion is based on Bode diagram of open-loop system
- Used to determine the stability of closed-loop system
- Nyquist stability criterion is based on the Nyquist diagram of the open-loop system
- Also used to determine the stability of closed-loop system
- Both determine the stability of closed-loop system on the basis of the frequency response of the open-loop system

Nyquist Criterion background concepts

Transfer function

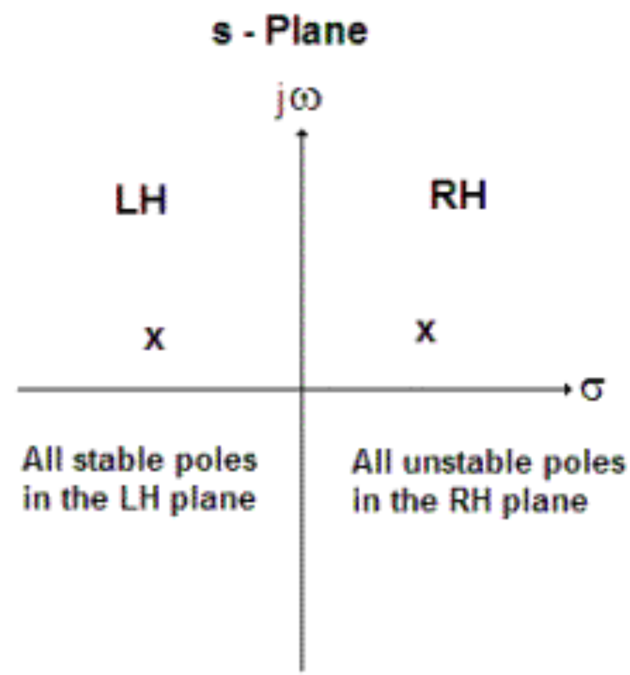


$$T(s) = \frac{G(s)}{1 + G(s)H(s)}$$

- The transfer function T(s) has closed-loop pole on the right half of the s-plane, the system is unstable

- To test for stability we need to check whether the characteristic equation has poles outside the left-half-plane

- If there are no closed-loop poles on the right half of the s-plane, the system is stable



Nyquist plot

- A Nyquist diagram is a plot of the frequency response on an Argand diagram

- No detailed rules for sketching Nyquist diagrams

- Suffices to determine the asymptotic behavior as angular frequency $\omega \rightarrow 0$ and $\omega \rightarrow \infty$

- Nyquist's Stability Theorem allows us to deduce closed-loop properties, the location of poles of the transfer function

$$T(s) = \frac{G(s)}{1 + \boxed{G(s)H(s)}}$$

- By consideration of the open loop frequency response of the return function

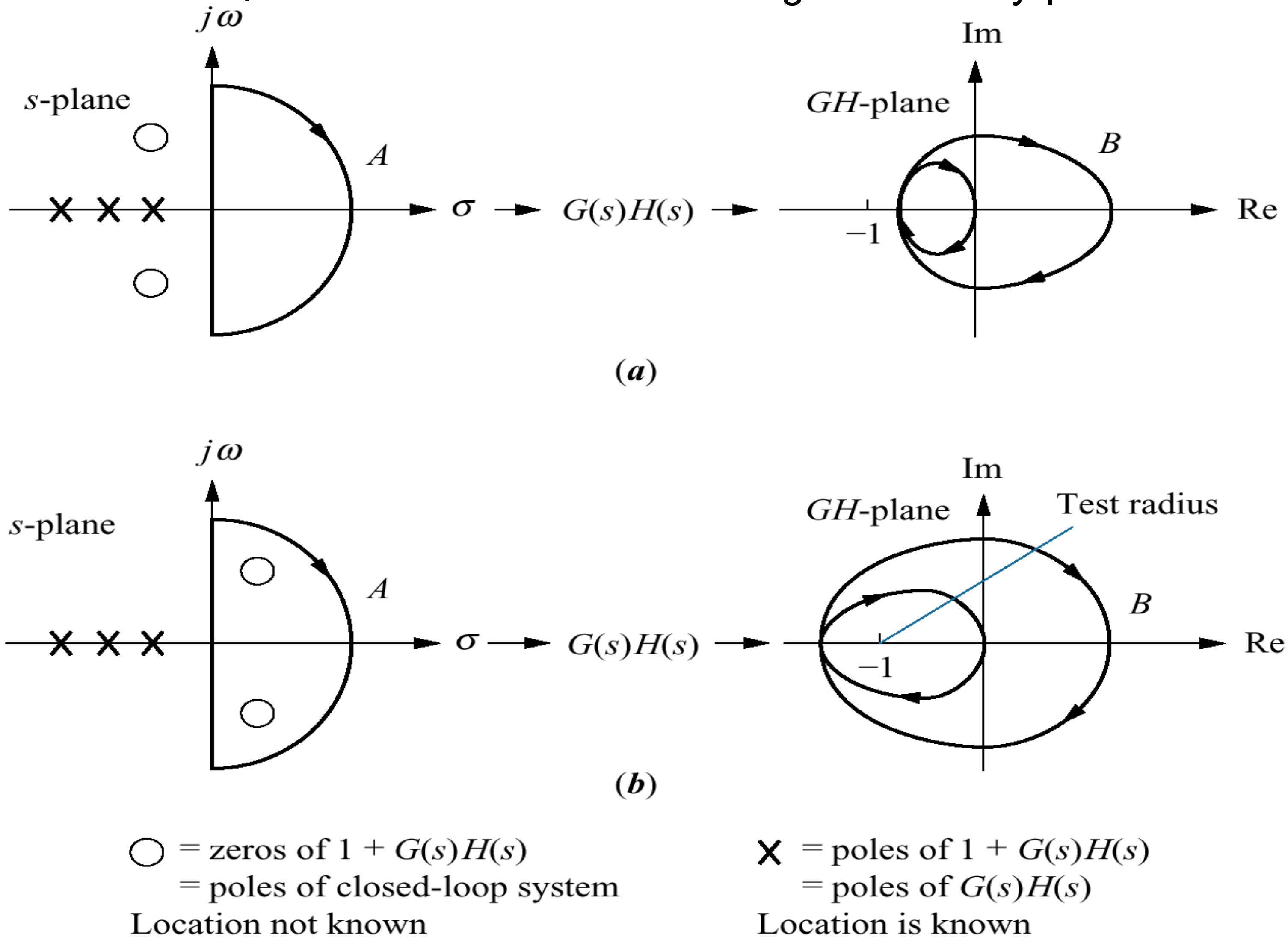
$$L(j\omega) = G(j\omega)H(j\omega)$$

- Main idea is we use –ve feedback to reduce error
- But when fed-back 180° out of phase it becomes +ve feedback!
- If gain >1 at this point the the system will become unstable!

Mapping between s-plane and F(s)-plane

Follow this s-plane contour

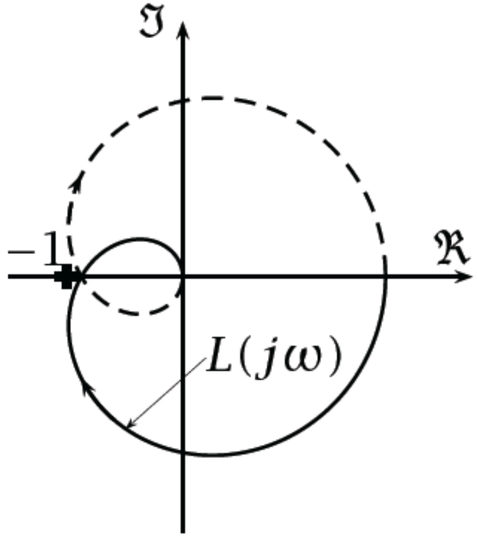
This generates Nyquist contour



Using Nyquist plot in a nutshell

$$L(j\omega) = G(j\omega)H(j\omega)$$

- The closed-loop system is stable if the Nyquist diagram of the return ratio doesn't enclose the point "−1"

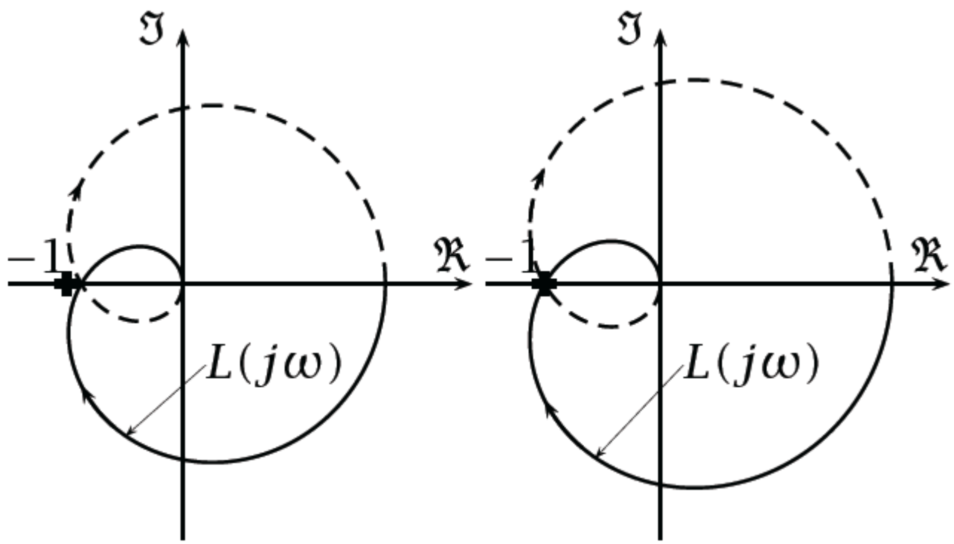


$\Longrightarrow \dfrac{L(s)}{1+L(s)}$ is asymptotically stable

Using Nyquist plot in a nutshell

$$L(j\omega) = G(j\omega)H(j\omega)$$

- Closed-loop system is stable if the Nyquist diagram of the return ratio doesn’t enclose the point “−1”



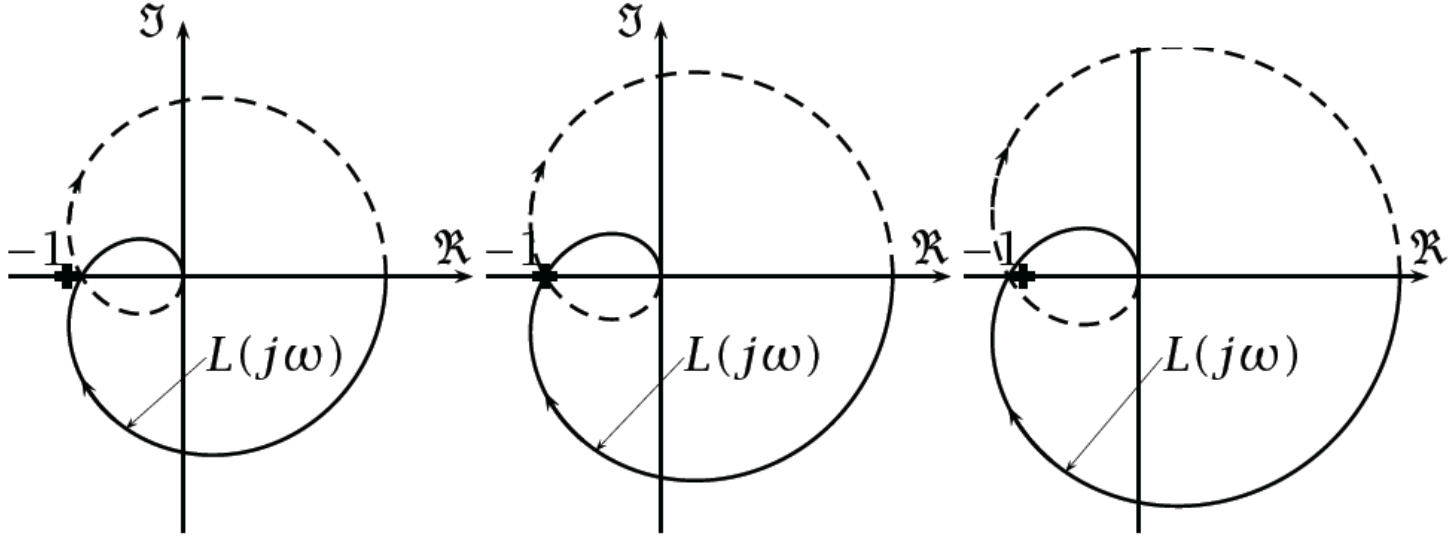
$\Longrightarrow \dfrac{L(s)}{1+L(s)}$ is asymptotically stable

$\Longrightarrow \dfrac{L(s)}{1+L(s)}$ is marginally stable

Using Nyquist plot in a nutshell

$$L(j\omega) = G(j\omega)H(j\omega)$$

- Closed-loop system is stable if the Nyquist diagram of the return ratio doesn't enclose the point "−1"



$\Longrightarrow \dfrac{L(s)}{1+L(s)}$ is asymptotically stable

$\Longrightarrow \dfrac{L(s)}{1+L(s)}$ is marginally stable

$\Longrightarrow \dfrac{L(s)}{1+L(s)}$ is unstable

Encirclement of -1

The reason the closed-loop system is stable if the Nyquist diagram of the return ratio

$$L(j\omega) = G(j\omega)H(j\omega)$$

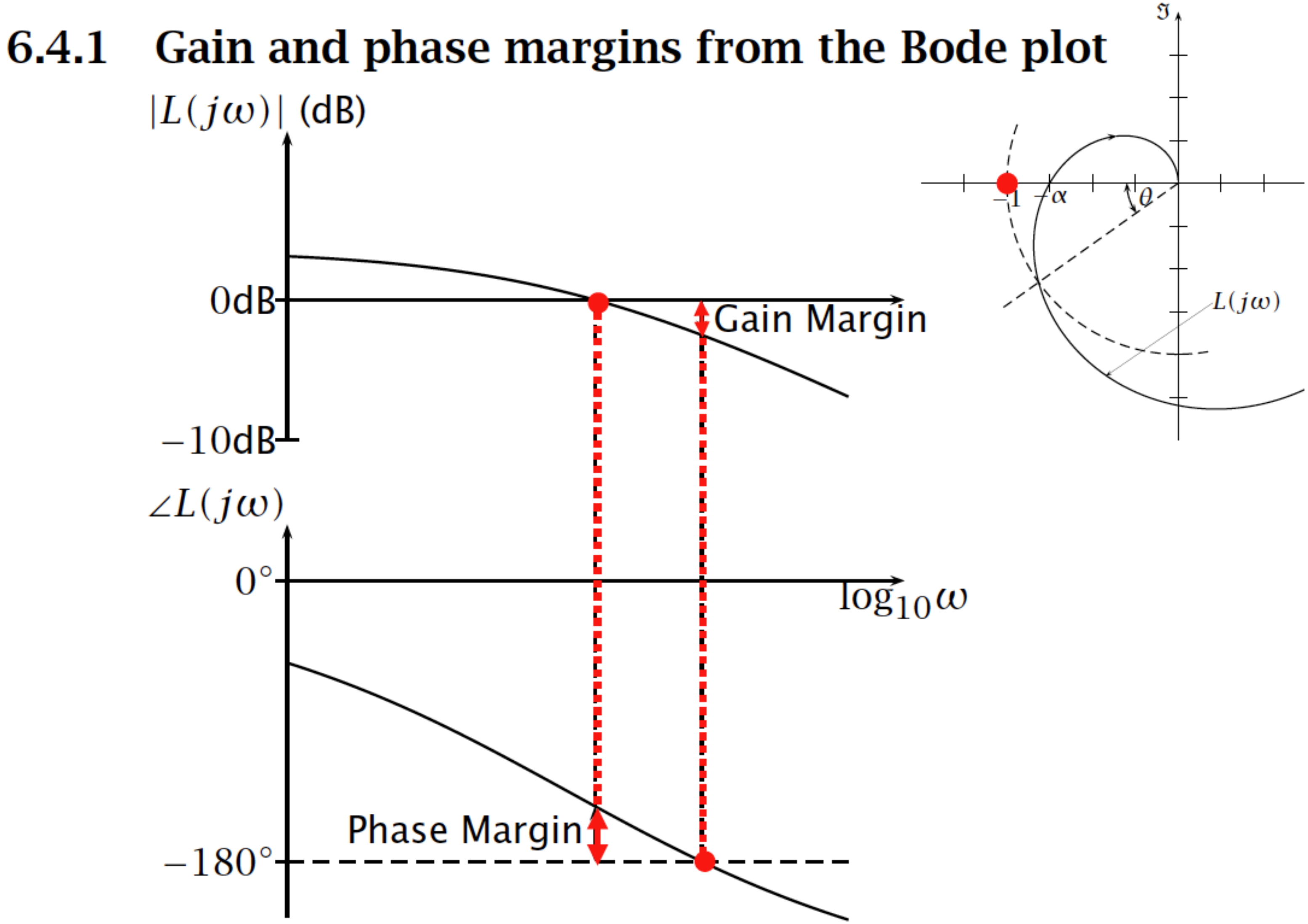
doesn't enclose the point "−1" is as follows:

If the Nyquist locus passes through the point "−1" then G(s)H(s) = 1

$$T(s) = \frac{G(s)}{1+G(s)H(s)} = \frac{G(s)}{1+L(s)}$$

That is we have a +ve real pole at 1 so the system is unstable
The closed-loop frequency response T(s) becomes infinite at that frequency
There is thus a sustained oscillation of the feedback system even when there is no external input

Compare Nyquist and Bode margin plots

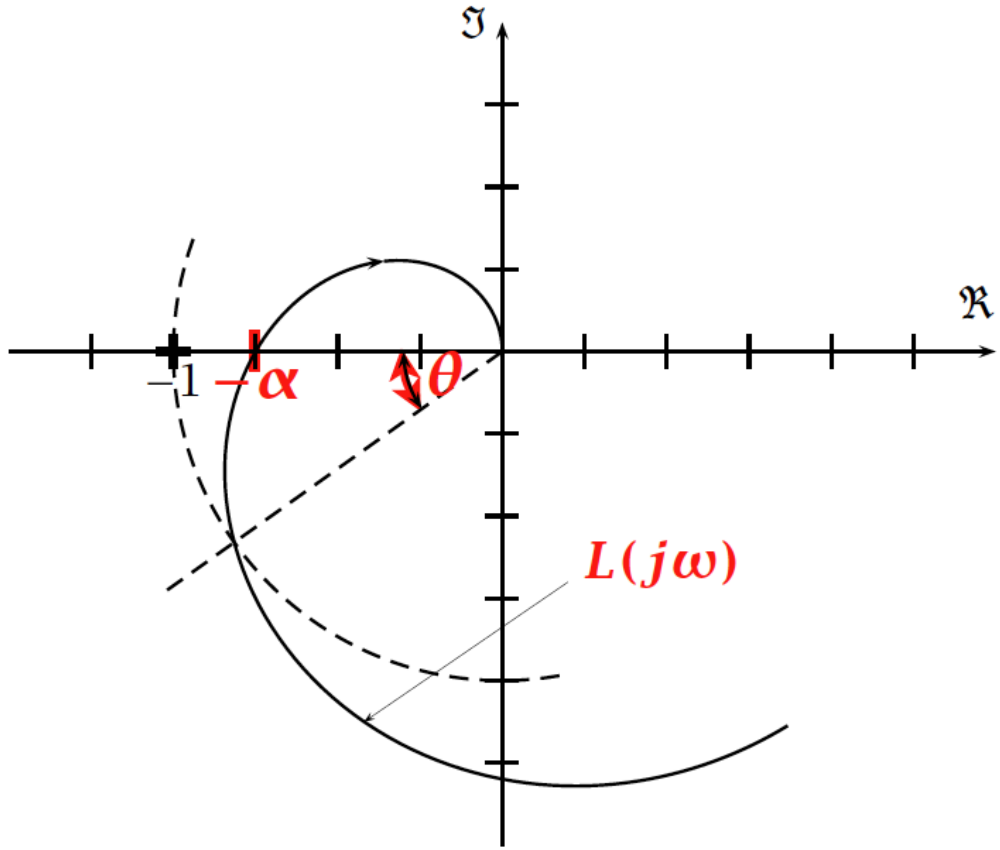


Gain Margin $= 20\log_{10} 4/3 = 2.5$dB. Phase Margin $= 35°$ (as before)

- We can read off gain and phase margins from a Bode plot

Nyquist plot gain and phase margin

- Similarly we can read off gain and phase margins from a Nyquist plot



$$\text{Gain Margin} = \frac{1}{\alpha} \quad \text{Phase Margin} = \theta$$

In this example we have $\theta = 35°$ and $-\alpha = -0.75$. Hence

Phase Margin $= 35°$ and Gain Margin $= 1/0.75 = 4/3$

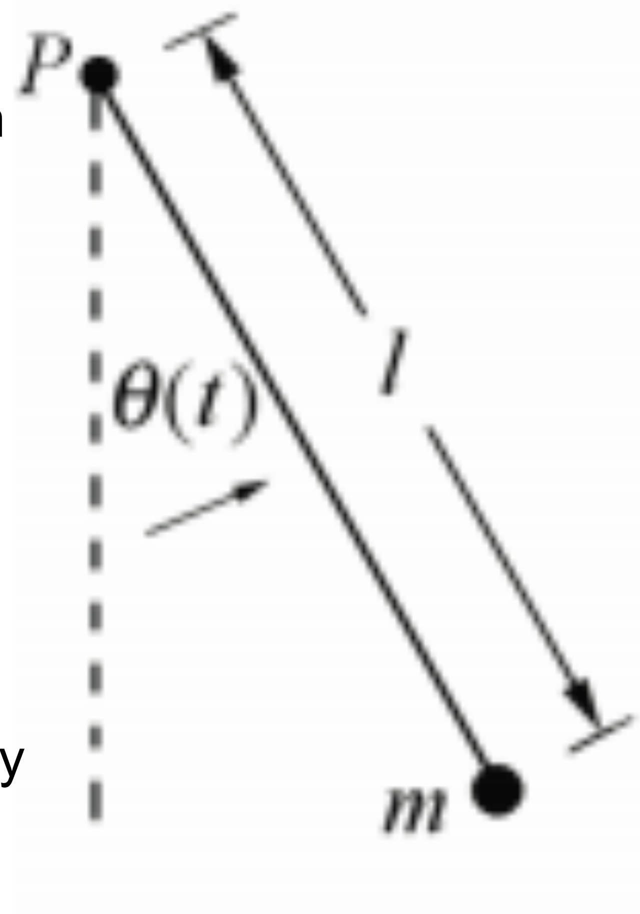
ROCO218: Control Engineering
Dr Ian Howard

Lecture 10

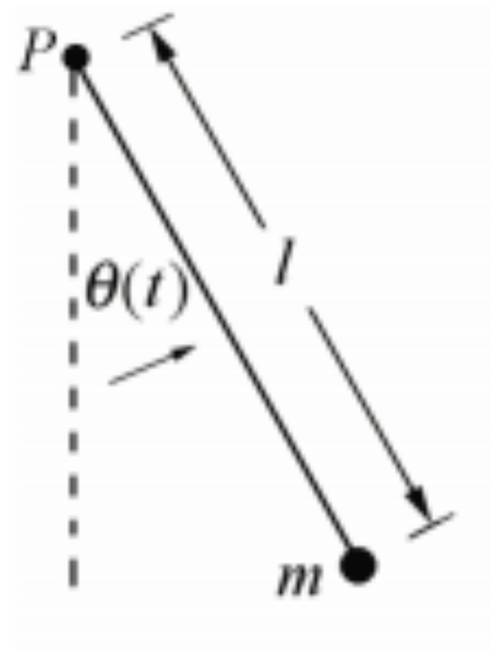
Pendulum analysis using KE and PE

Fixed point mass pendulum

- Consider the conservation of energy within a swinging pendulum
- In the absence of any losses use to friction, when the system is initially pushed from equilibrium it will oscillate
- It will swing from side to side
- It will reach zero velocity at the top of each swing
- It will reach peak velocity at its lowest position
- There will be an exchange between kinetic energy (KE) and potential energy (PE)
- And the total energy will therefore be a constant.
- We shall represent total energy by K



Fixed point mass pendulum



- Total energy within the pendulum will be given by

$$KE + PE = K$$

- The kinetic energy term KE is given by

$$KE = \frac{1}{2} I\omega^2$$

- Where I is the moment of inertia of the point mass and ω is angular velocity
- For a point mass

$$I = ml^2$$

$$\Rightarrow KE = \frac{1}{2} ml^2\omega^2$$

- The potential energy term PE is given by

$$PE = mgh$$

- Where m is the point mass mass, h is its height and g is the acceleration due to gravity
- Taking the origin at the pivot then the height h is given by $h = -l\cos(\theta)$

$$\Rightarrow PE = -mgl\cos(\theta)$$

Fixed point mass pendulum

- Consideration of total energy in the pendulum results in the equation

$$\frac{1}{2}ml^2\omega^2 - mgl\cos(\theta) = K$$

  - Differentiating both sized to get rid of constant

$$\frac{d}{dt}\left(\frac{1}{2}ml^2\omega^2 - mgl\cos(\theta)\right) = \frac{d}{dt}(K)$$

$$\Rightarrow \frac{1}{2}ml^2\frac{d}{dt}(\omega^2) - mgl\frac{d}{dt}(\cos(\theta)) = 0$$

- Now

$$\omega = \dot{\theta}$$

$$\Rightarrow \frac{1}{2}ml^2\frac{d}{dt}(\dot{\theta}^2) - mgl\frac{d}{dt}(\cos(\theta)) = 0$$

Chain rule for single variable differentiation

- Consider the one variable equation corresponding to a function of a function

$$z = f(g(x))$$

- Writing

$$y = g(x)$$

- Chain rule gives

$$\frac{dz}{dx} = \frac{dz}{dy} . \frac{dy}{dx}$$

- For example if

$$z = e^{x^2}$$

- In this case let

$$y = x^2 \qquad \Rightarrow \frac{dy}{dx} = 2x$$

$$z = e^y \qquad \Rightarrow \frac{dz}{dy} = e^y$$

$$\Rightarrow \frac{dz}{dx} = e^y .2x \qquad \boxed{= 2xe^{x^2}}$$

Fixed point mass pendulum

- Consider the term

$$\frac{d}{dt}\left(\dot{\theta}^2\right)$$

- So using chain rule writing $\quad y = \dot{\theta}^2 \quad \Rightarrow \dfrac{dy}{dt} = \dot{\theta}^2$

Letting

$$y = u^2 \qquad \Rightarrow \frac{dy}{du} = 2u \qquad = 2\dot{\theta}$$

$$u = \dot{\theta} \qquad \Rightarrow \frac{du}{dt} = \ddot{\theta}$$

- Therefore

$$\Rightarrow \frac{dy}{dt} = \frac{dy}{du} . \frac{du}{dt} \qquad = 2\dot{\theta}\ddot{\theta}$$

Fixed point mass pendulum

- Consider the term

$$\frac{d}{dt}\left(\cos(\theta)\right)$$

- So using chain rule writing $\quad y = \cos(\theta) \quad \Rightarrow \frac{dy}{dt} = \frac{d}{dt}\left(\cos(\theta)\right)$

Letting

$$y = \cos(u) \qquad \Rightarrow \frac{dy}{du} = -\sin(u)$$

$$u = \theta \qquad \Rightarrow \frac{du}{dt} = \dot{\theta}$$

- Therefore

$$\Rightarrow \frac{dy}{dt} = \frac{dy}{du}.\frac{du}{dt} \qquad = -\dot{\theta} sin(\theta)$$

Fixed point mass pendulum

- Therefore the expression

$$\frac{1}{2}ml^2\frac{d}{dt}\left(\dot{\theta}^2\right)-mgl\frac{d}{dt}(\cos(\theta))=0$$

$$\Rightarrow\frac{1}{2}ml^2 2\dot{\theta}\ddot{\theta}+mgl\dot{\theta}sin(\theta)=0$$

$$\Rightarrow\dot{\theta}\left(ml^2\ddot{\theta}+mglsin(\theta)\right)=0$$

- Trivial solution is then when pendulum stationary

$$\Rightarrow\dot{\theta}=0$$

- Simplifying expression for non-trivia solution

$$\Rightarrow l\ddot{\theta}=-gsin(\theta)$$

$$\Rightarrow\ddot{\theta}=-\frac{g}{l}\sin(\theta)$$

ROCO218: Control Engineering
Dr Ian Howard

Lecture 10

Topics we haven't covered

Additional topics in control engineering

- System analysis using the Euler–Lagrange equation
    - Deriving system differential equations by consideration of energy
- Dealing with time delays
    - Smith predictor
- Digital sampled-time systems
    - Z-transform rather than Laplace transform
    - Discrete rather than continuous state space models
- Dealing with controllable and unobservable systems
    - Use Kalman decomposition to find stabilizable and detectable parts of the systems
- The Kalman filter for optimal state estimation
    - Stochastic approach whereas Luenberger is deterministic
- System identification
    - Learning the plant kinematics and dynamics and optimal control gains and strategies from experience
- Plus lots more…….