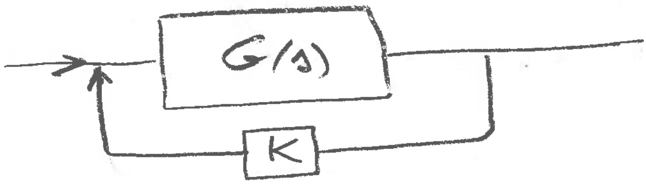
1 SFB

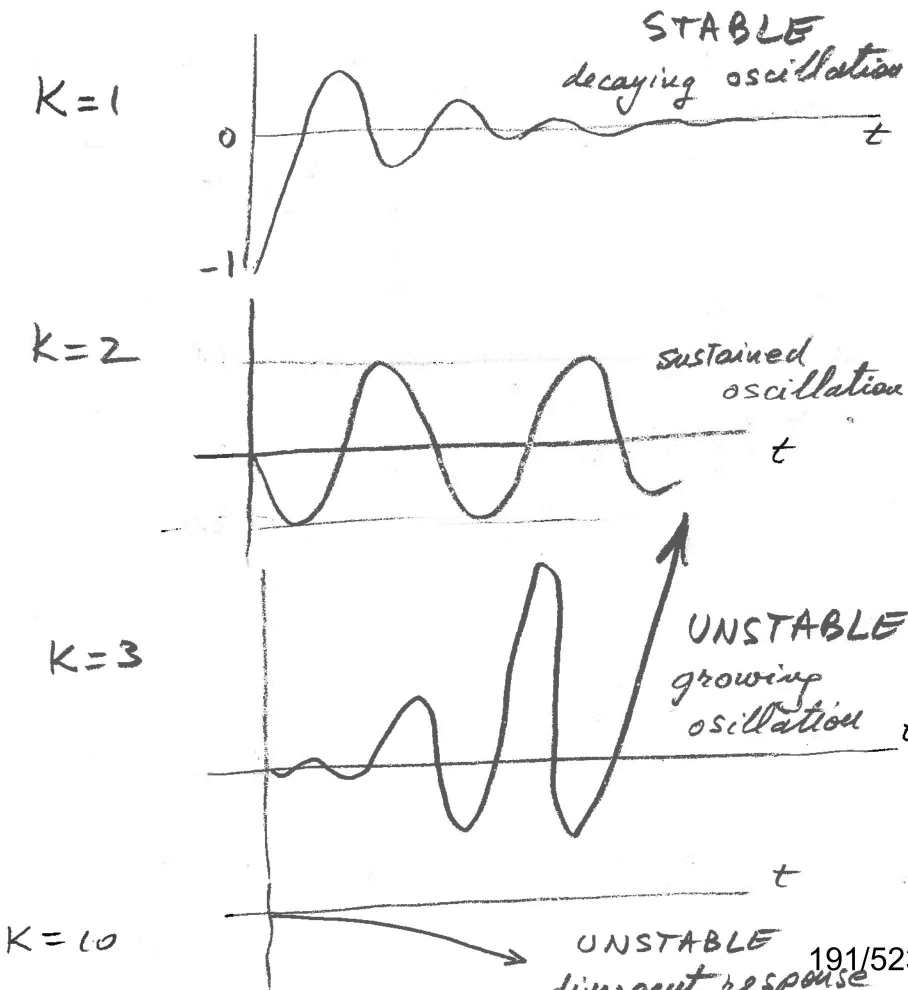
STABILITY OF FEEDBACK SYSTEMS

EXAMPLE

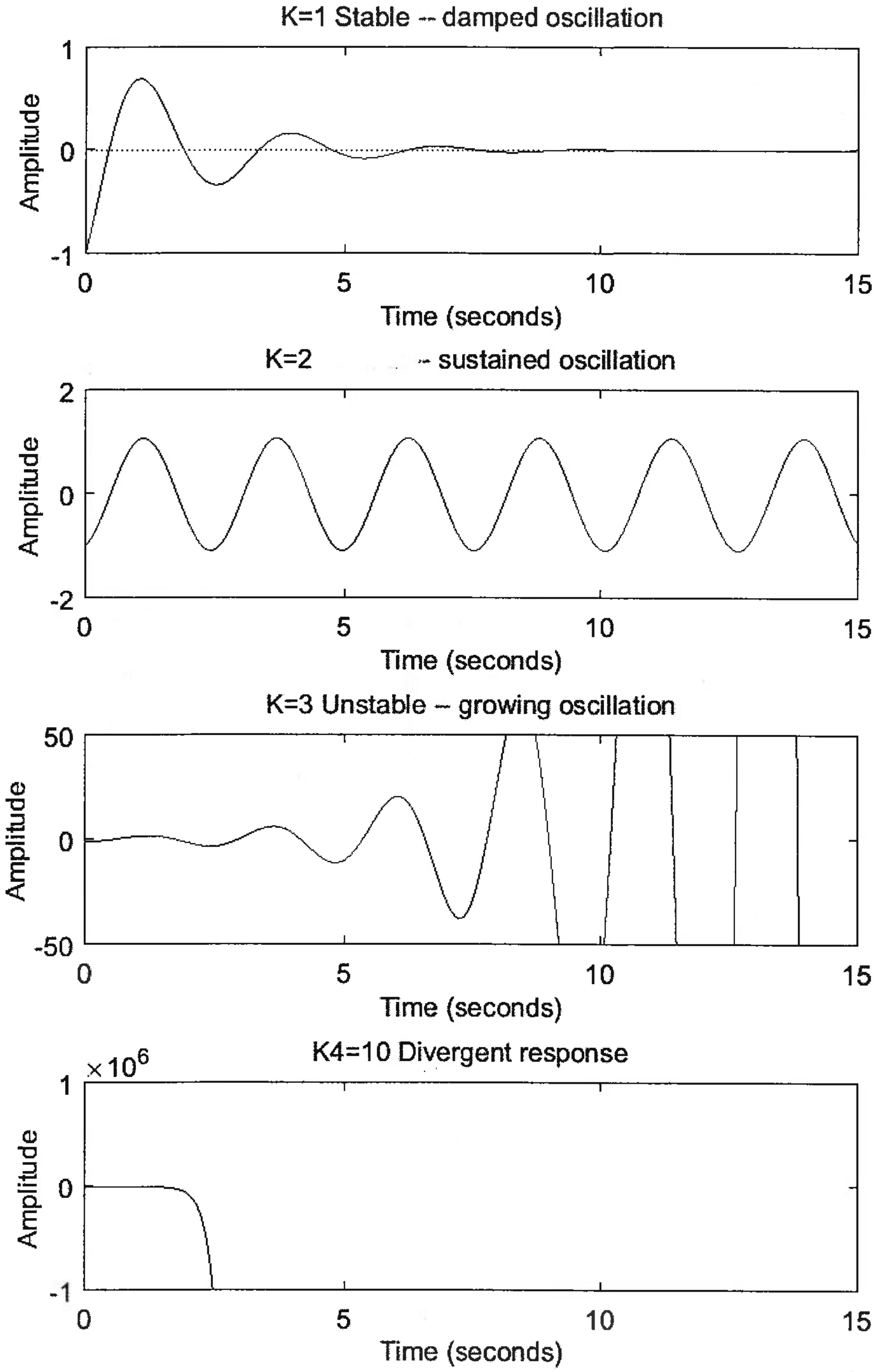
Run MATLAB code FB_stability_ex1



$$G(s) = \frac{1-s}{s^2+2s+4}$$



2
SFB


K=1 Stable -- damped oscillation
Amplitude
1
0
-1
0
5
10
15
Time (seconds)
K=2 -- sustained oscillation
Amplitude
2
0
-2
0
5
10
15
Time (seconds)
K=3 Unstable -- growing oscillation
Amplitude
50
0
-50
0
5
10
15
Time (seconds)
K4=10 Divergent response
$\times 10^6$
Amplitude
1
0
-1
0
5
10
15
Time (seconds)

3 SFB

Explanation

$$G_{CL} = \frac{G}{1+GK} = \frac{\dfrac{1-s}{s^2+2s+4}}{1+K\dfrac{1-s}{s^2+2s+4}}$$

$$= \frac{\dfrac{1-s}{s^2+2s+4}}{s^2+2s+4+K(1-s)} = \frac{1-s}{s^2+(2-K)s+(K+4)}$$

Impulse response: $X(s) = G_{CL}(s) = \dfrac{1-s}{s^2+(2-K)s+(K+4)}$

Response type depends on characteristic eq.

$$s^2+(2-K)s+(K+4) = 0$$ characteristic eqn.

$K=1$ $\quad s^2+s+5=0$

$p_{1,2} = \dfrac{-1\pm\sqrt{1-4\times5}}{2} = -\dfrac{1}{2} \pm i\dfrac{\sqrt{19}}{2}$ LHS decaying osc. STABLE

$K=2$ $\quad s^2+6=0$

$p_{1,2} = \pm i\sqrt{6}$ imag. axis sustained oscillation

$K=3$ $\quad s^2-s+7=0$

$p_{1,2} = \dfrac{1\pm\sqrt{1-28}}{2} = \dfrac{1}{2} \pm i\dfrac{\sqrt{27}}{2}$ RHS growing osc. UNSTABLE

$K=10$ $\quad s^2-8s+14=0$

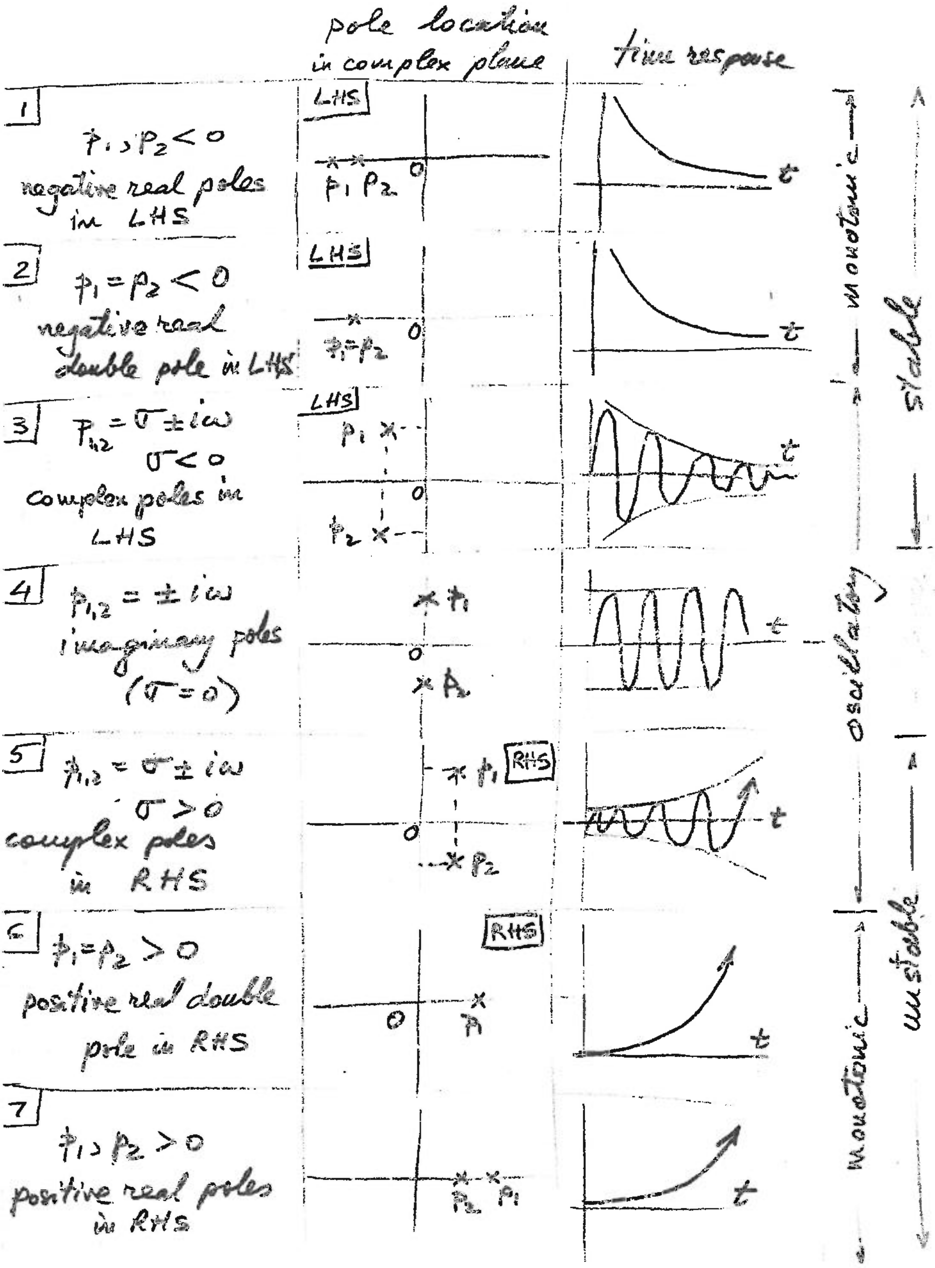
$p_{1,2} = 4\pm\sqrt{4^2-14} = 4\pm\sqrt{2} \begin{cases} 5.41 \\ 2.59 \end{cases}$ RHS UNSTABLE non-oscillatory

4

```
K1 =
     1

G1CL =

    -s + 1
  -----------
  s^2 + s + 5

Continuous-time transfer function.

p1 =
  -0.5000 + 2.1794i
  -0.5000 - 2.1794i
wn1, rad/sec | zeta1 =
    2.2361    0.2236
    2.2361    0.2236
=========================
K2 =
     2

G2CL =

  -s + 1
  -------
  s^2 + 6

Continuous-time transfer function.

p2 =
   0.0000 + 2.4495i
   0.0000 - 2.4495i
wn2, rad/sec | zeta2 =
    2.4495         0
    2.4495         0
```

```
K3 =
     3

G3CL =

    -s + 1
  -----------
  s^2 - s + 7

Continuous-time transfer function.

p3 =
   0.5000 + 2.5981i
   0.5000 - 2.5981i
wn3, rad/sec | zeta3 =
    2.6458   -0.1890
    2.6458   -0.1890
=========================
K4 =
    10

G4CL =

      -s + 1
  --------------
  s^2 - 8 s + 14

Continuous-time transfer function.

p4 =
    5.4142
    2.5858
wn4, rad/sec | zeta4 =
    5.4142   -1.0000
    2.5858   -1.0000
```

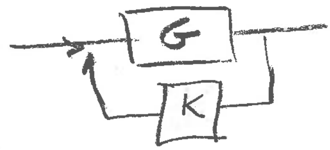
Recall stability analysis as function of pole location:



20140305

Root Locus

Given $G(s)$



Trace the poles of $G_{CL}(s)$ as K increases

$$G(s) = \frac{B(s)}{A(s)} = \frac{num(s)}{den(s)}$$

$$G_{CL} = \frac{G}{1+KG} = \frac{B(s)}{A(s)+KB(s)}$$

The root locus method looks at the root of the denominator of the CL system G_{CL}

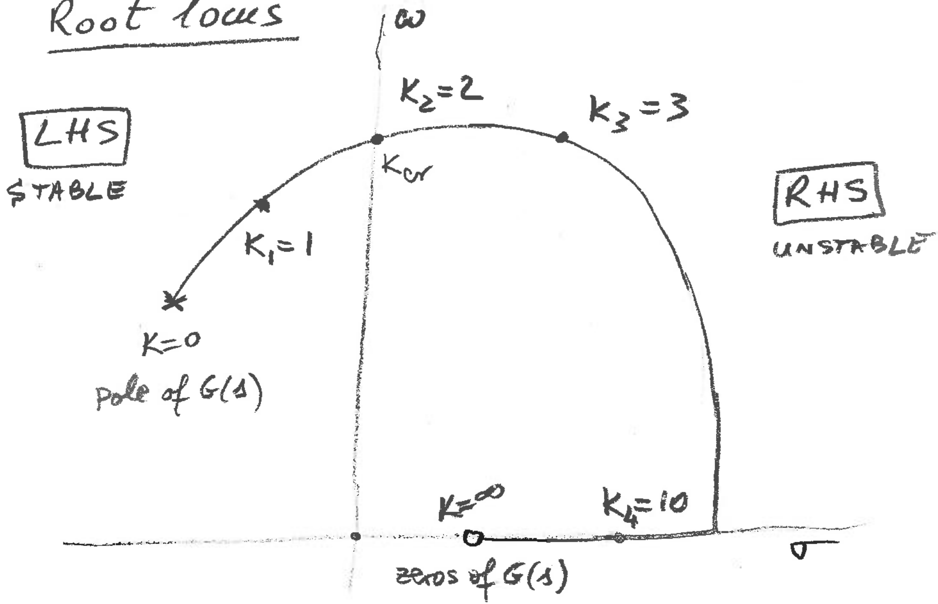
$$A(s) + KB(s) = 0$$

The roots are traced in the s-plane for $K = [0, \infty)$

For $K=0$, the poles of G_{CL} are the same as the poles of G.

1
SFB

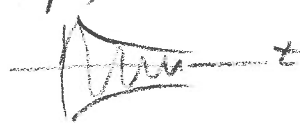
Root locus



$$p_{1,2} = \sigma \pm i\omega_d = -\zeta\omega_n \pm i\omega_d$$

$$x(t) = C e^{-\zeta\omega_n t} \sin(\omega_d t + \varphi)$$

K_1 : $\sigma < 0 \longrightarrow \zeta > 0$



K_2 : $\sigma = 0 \longrightarrow \zeta = 0$
K_{cr}

K_3 : $\sigma > 0 \longrightarrow \zeta < 0$

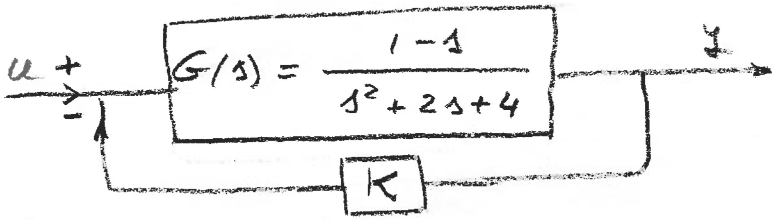
K_4 : $\omega = 0$

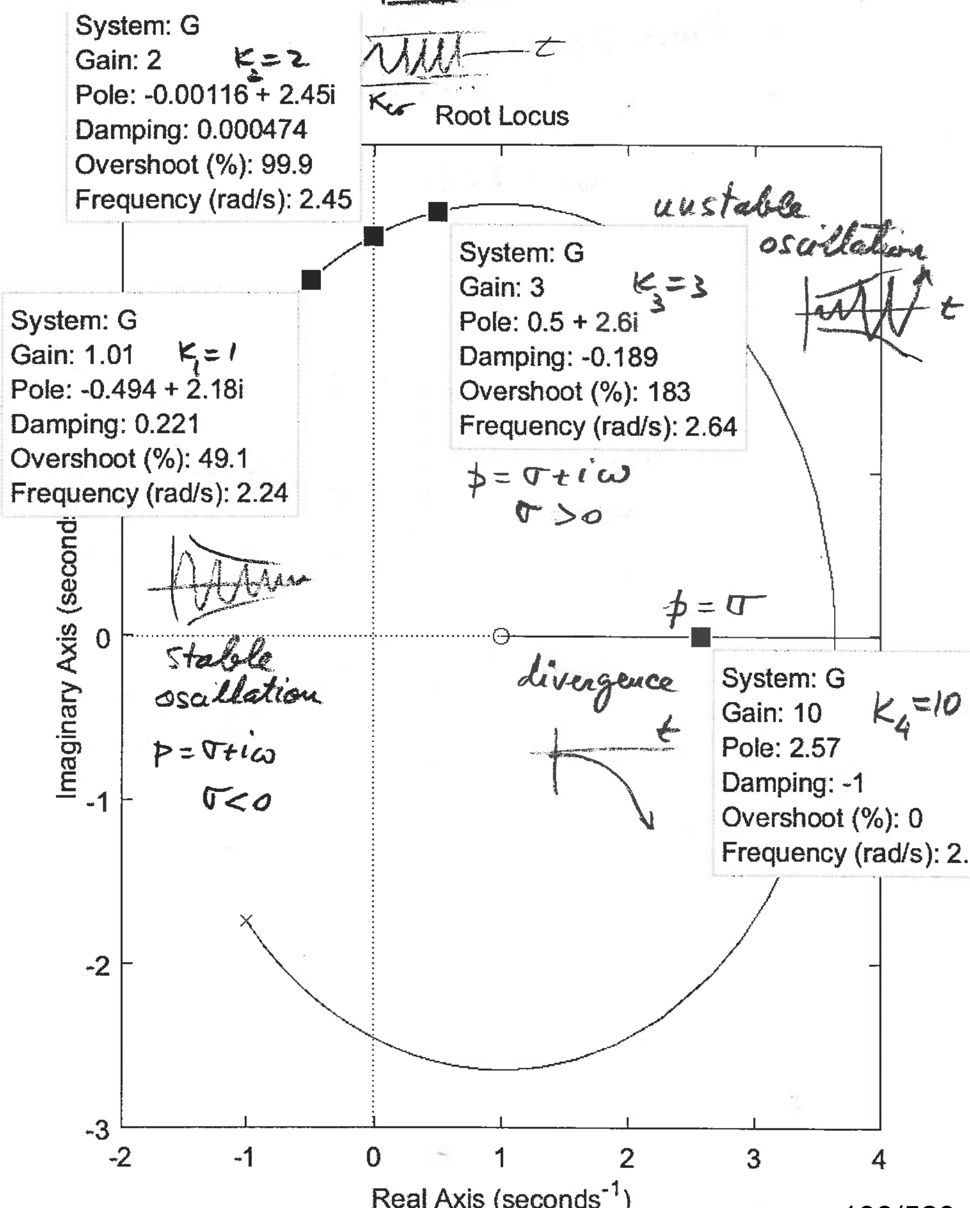
$p_1 = \sigma_1$, $p_2 = \sigma_2 > 0$

$$x(t) = C_1 e^{p_1 t} + C_2 e^{p_2 t}$$

8 SFB

MATLAB rlocus function





7
sFB

rlocus

Evans root locus

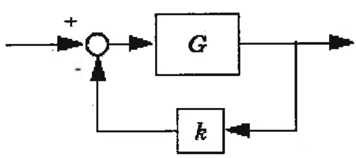
Syntax

```
rlocus
rlocus(sys)
rlocus(sys1,sys2,...)
```

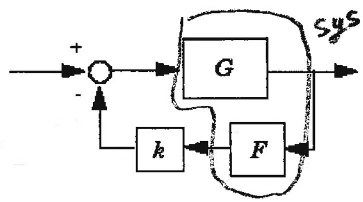
Description

`rlocus` computes the Evans root locus of a SISO open-loop model. The root locus gives the closed-loop pole trajectories as a function of the feedback gain k (assuming negative feedback). Root loci are used to study the effects of varying feedback gains on closed-loop pole locations. In turn, these locations provide indirect information on the time and frequency responses.

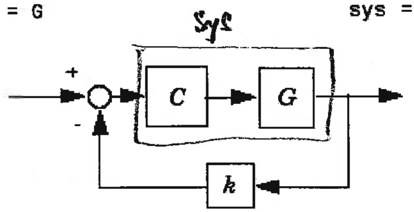
`rlocus(sys)` calculates and plots the root locus of the open-loop SISO model `sys`. This function can be applied to any of the following *negative* feedback loops by setting `sys` appropriately.



sys = G



sys = F * G



sys = G * C

If `sys` has transfer function

$$h(s) = \frac{n(s)}{d(s)}$$

the closed-loop poles are the roots of

$$d(s) + k\ n(s) = 0$$

`rlocus` adaptively selects a set of positive gains k to produce a smooth plot. Alternatively,

10
FB

```
rlocus(sys,k)
```

uses the user-specified vector `k` of gains to plot the root locus.

`rlocus(sys1,sys2,...)` draws the root loci of multiple LTI models `sys1, sys2,...` on a single plot. You can specify a color, line style, and marker for each model, as in

```
rlocus(sys1,'r',sys2,'y:',sys3,'gx').
```

When invoked with output arguments,

```
[r,k] = rlocus(sys)
r = rlocus(sys,k)
```

return the vector `k` of selected gains and the complex root locations `r` for these gains. The matrix `r` has `length(k)` columns and its `j`th column lists the closed-loop roots for the gain `k(j)`.

Remarks

You can change the properties of your plot, for example the units. For information on the ways to change properties of your plots, see Ways to Customize Plots.

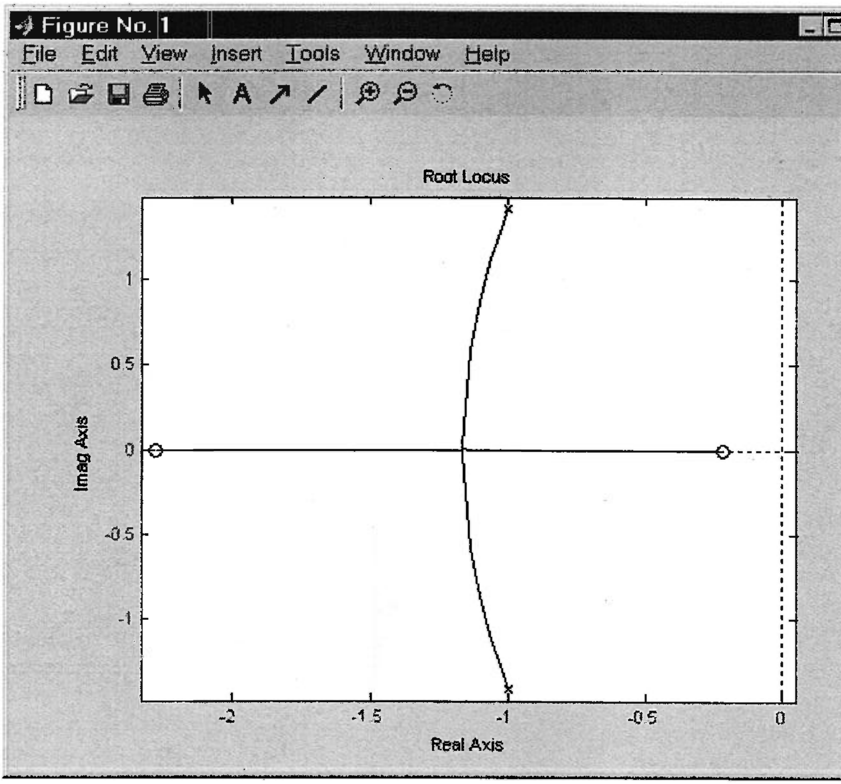
Example

Find and plot the root-locus of the following system.

$$h(s) = \frac{2s^2 + 5s + 1}{s^2 + 2s + 3}$$

```
h = tf([2 5 1],[1 2 3]);
rlocus(h)
```

11
5FB



You can use the right-click menu for rlocus to add grid lines, zoom in or out, and invoke the Property Editor to customize the plot. Also, click anywhere on the curve to activate a data marker that displays the gain value, pole, damping, overshoot, and frequency at the selected point.

See Also

pole, pzmap

12
SFB

ROOT LOCUS METHOD

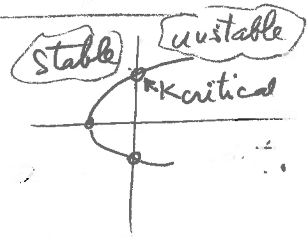
$G(s) = \frac{B(s)}{A(s)} = \frac{num}{den}$

$$G_{CL} = \frac{KG}{1+KG}$$

$$G(s) = \frac{(s+z_1)(s+z_2)+ \cdots (s+z_m)}{(s+p_1)(s+p_2)+ \cdots (s+p_n)} = \frac{num}{den}$$

$-z_1, \ldots, -z_m$ zeros of open loop transfer function

$-p_1, \ldots, -p_n$ poles ——"——



<u>charact. eq^n</u>: $1 + KG = 0$.

$$A(s) + K B(s) = 0$$

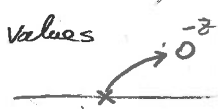
$$(s+p_1)(s+p_2) \cdots (s+p_n) + K(s+z_1) \cdots (s+z_m) = 0.$$

Solution of this equation gives the CL poles
$s = \sigma + i\omega$

Root locus: trajectory of these poles as
$K = 0 \longrightarrow \infty$

rlocus (num, den) ⟶ automatic K generation
$K \in [0, \infty)$

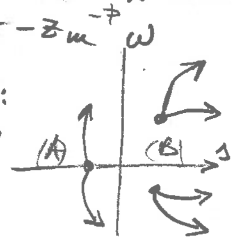
rlocus (num, den, K) ⟶ must give K values



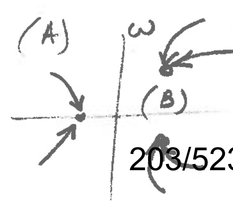
<u>$K=0$:</u> roots are the OL poles: $-p_1, -p_2, \cdots -p_n$

$K=\infty$: roots are the OL zeros: $-z_1, -z_2 \cdots -z_m$

<u>Breakaway</u>: multiple roots branch out:
A) from real poles on real axis
(B) from conjugate poles



<u>Break in</u>: branches coalesce into multiple roots
(A) into real poles
(B) into conjugate poles

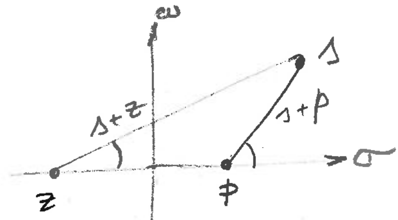


13
SFB

Angle condition

$$1+KG=0$$

$$KG=-1$$

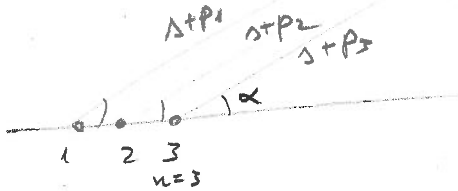


$$\angle KG = \angle -1 = \pm 180^\circ(2K+1),\ K=0,1,\ldots$$

$$\angle KG = \angle K\frac{(s+z_1)(s+z_2)\cdots}{(s+p_1)(s+p_2)\cdots}$$

$$= \angle s+z_1 + \angle s+z_2 + \cdots$$
$$- \angle s+p_1 - \angle s+p_2 - \angle s+p_3 \cdots = \pm 180^\circ(2k+1)$$

Asymptotes $s \to \infty$



- angles of asymptotes

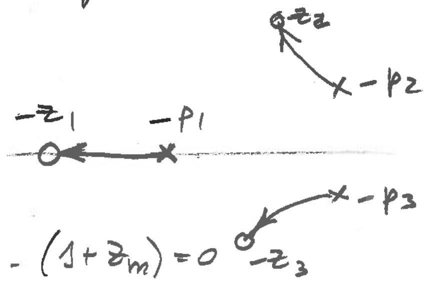
$$\angle asymp = \frac{\pm 180^\circ(2k+1)}{3}$$
$$= 60^\circ;\ 180^\circ,\ -60^\circ$$

- axis crossing of asymptotes

14
SFB $(s+p_1)(s+p_2)\cdots + K(s+z_1)(s+z_2)\cdots = 0$

$K=0 \longrightarrow (s+p_1)(s+p_2)\cdots(s+p_n)=0$

roots are the OL poles



$K \to \infty$: $K(s+z_1)(s+z_2)\cdots(s+z_m)=0$

roots are the OL zeros