

MIT OpenCourseWare
<http://ocw.mit.edu>

2.004 Dynamics and Control II
Spring 2008

For information about citing these materials or our Terms of Use, visit: <http://ocw.mit.edu/terms>.

Lecture 29¹

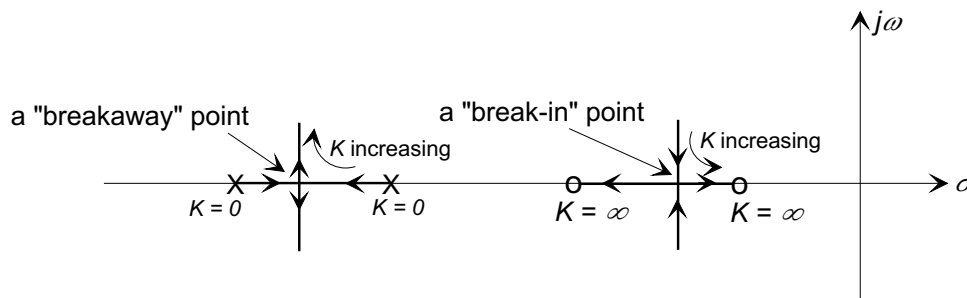
Reading:

- Nise: Chapter 8

1 Root Locus Refinement

The complete set of sketching rules contains additional methods to make a sketched plot more accurate. While these were useful in the days before ubiquitous computation, today with the existence of tools such as MATLAB makes these graphical refinements somewhat unnecessary. We therefore just mention them here and refer you to Nise, Section 8.5, for more detail.

1.1 Real-Axis Breakaway and Break-In Points



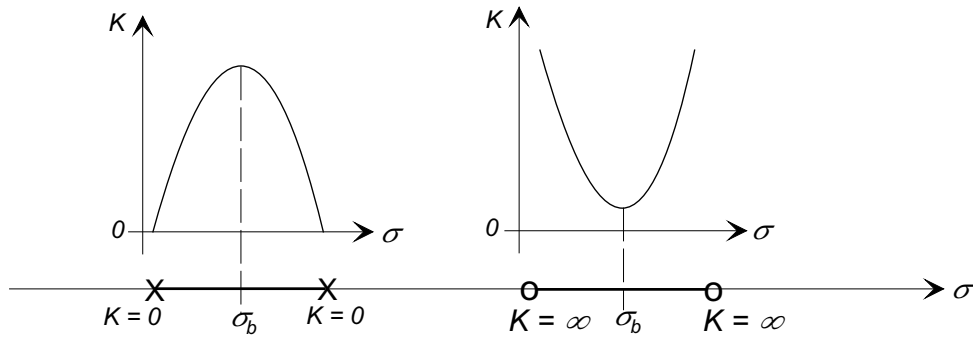
A *breakaway point* is the point on a real axis segment of the root locus between two real poles where the two real closed-loop poles meet and diverge to become complex conjugates. Similarly, a *break-in point* is the point on a real axis segment of the root locus between two real zeros where two real closed-loop complex conjugate zeros meet and diverge to become real.

Because the closed-loop poles originate from open-loop poles (when $K = 0$), a breakaway point will correspond to the point of *maximum* K along the real-axis segment. Similarly, a break-in point will correspond to the point of *minimum* K on the real axis segment between the two zeros.

The closed-loop characteristic equations is $1 + KG(s) = 0$, so that along the root locus segments on the real axis ($s = \sigma$)

$$K = -\frac{1}{G(\sigma)} = -\frac{D(\sigma)}{N(\sigma)}$$

¹copyright © D.Rowell 2008



The breakaway/break-in points (maximum/minimum points) will therefore occur where

$$\frac{dK}{d\sigma} = -\frac{d}{dK} \left(\frac{D(\sigma)}{N(\sigma)} \right) = 0$$

or when

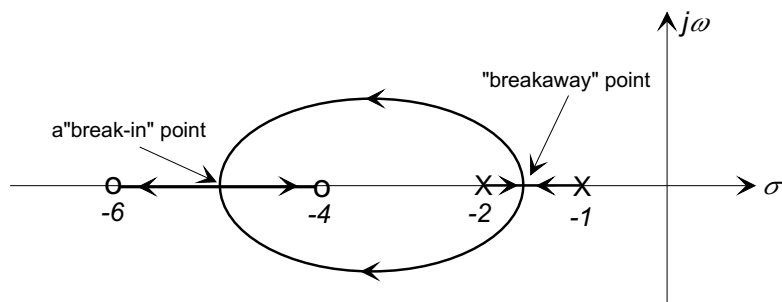
$$N(\sigma)D'(\sigma) - N'(\sigma)D(\sigma) = 0.$$

■ Example 1

Find the real axis breakaway/break-in points for the closed-loop system with

$$G(s) = \frac{s^2 + 10s + 24}{s^2 + 3s + 2} = \frac{(s + 6)(s + 4)}{(s + 1)(s + 2)}.$$

The root locus has two real-axis segments, between the pole pair and between the zero pair. There will therefore be a breakaway point and a break-in point.



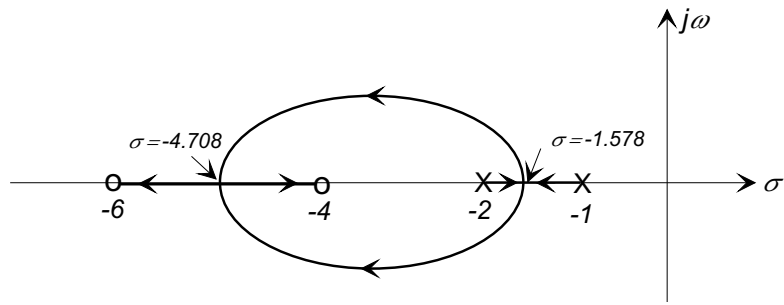
The breakaway/break-in points will be contained in the roots of

$$N(\sigma)D'(\sigma) - N'(\sigma)D(\sigma) = 0.$$

or

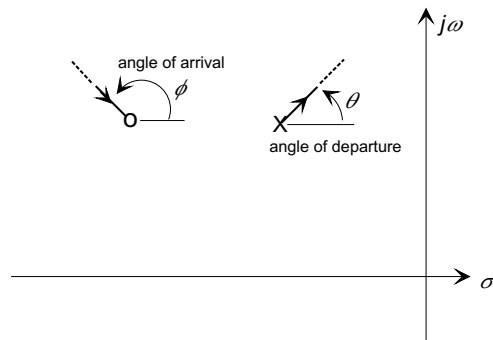
$$(\sigma^2 + 10\sigma + 24)(2\sigma + 3) - (\sigma^2 + 3\sigma + 2)(2\sigma + 10) = 7\sigma^2 + 44\sigma + 52 = 0$$

giving $\sigma_{1,2} = -4.708, -1.578$, as shown below:

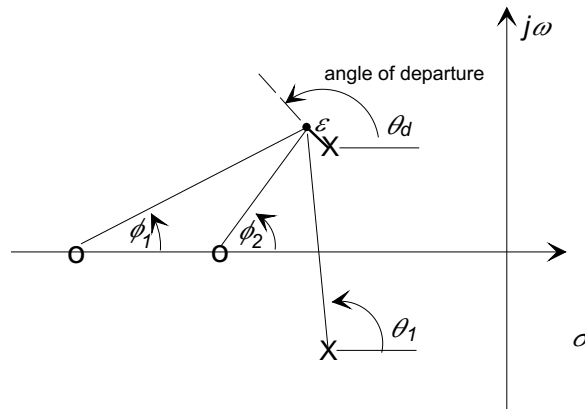


1.2 Angle of Arrival and Departure from Zeros and Poles

Further refinement of the Root Locus may be made by computing the angle at which the branches of the locus depart from the open-loop poles, and arrive at the open-loop zeros.



Consider a point a small distance ϵ from a pole:



The angle condition at the point requires

$$\sum \text{angles from the zeros} - \sum \text{angles from the poles} = (2k + 1)\pi$$

or

$$\phi_1 + \phi_2 - \theta_1 - \theta_d = (2k + 1)\pi$$

Let $k = 0$ and let $\epsilon \rightarrow 0$, then

$$\theta_d = \phi_1 + \phi_2 - \theta_1 - \pi$$

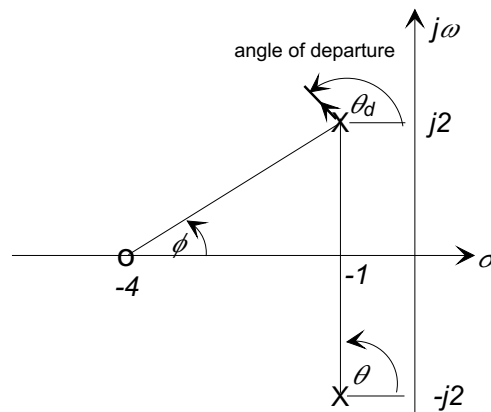
where the angles are measured to the pole itself.

A similar argument defines the angle of arrival at a complex zero.

■ Example 2

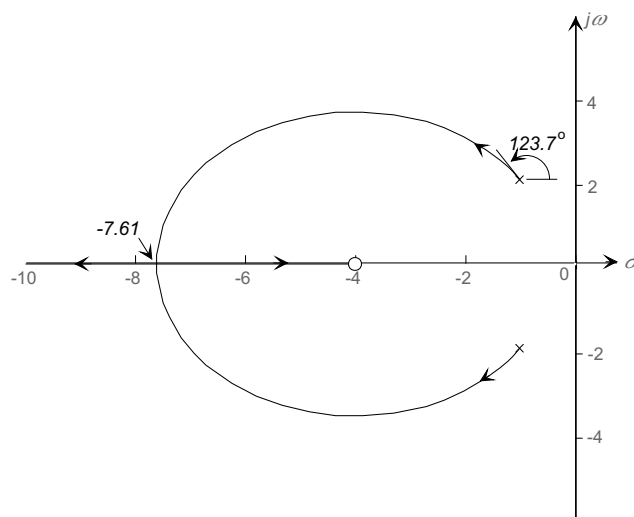
Find the angle of departure at the pole $p = -1 + j2$ for the closed-loop system where

$$G(s) = \frac{s + 4}{s^2 + 2s + 5}$$



In the above figure $\theta = 90^\circ$, $\phi = \arctan(2/3) = 33.7^\circ$. The angle of departure is therefore

$$\begin{aligned} \theta_d &= \phi - \theta - \pi \\ &= 33.7^\circ - 90^\circ - 180^\circ \\ &= -236.31^\circ = 123.69^\circ \end{aligned}$$



1.3 Summary of Root Locus Sketching Rules

Definitions	<ul style="list-style-type: none"> • The open-loop transfer function is $KG_c(s)G_pH(s)$ which can be rewritten as $KN(s)/D(s)$. • $N(s)$, the numerator, is an mth order polynomial; $D(s)$ is nth order. • $G(s)$ has zeros at z_i, ($i = 1 \dots m$); and poles at p_i ($i = 1 \dots n$).
Symmetry	The locus is symmetric about real axis (i.e., complex poles appear as conjugate pairs).
Number of branches	There are n branches of the locus, one for each pole of the closed-loop transfer function.
Start and end points	The locus starts (when $K = 0$) at poles of the open-loop transfer function, and ends (when $K = \infty$) at the zeros. Note: there are $n - m$ zeros of the open-loop transfer function as $ s \rightarrow \infty$.
Locus on real axis	The locus exists on the real axis <i>to the left of an odd number of poles and zeros</i> .
Asymptotes as $s \rightarrow \infty$	If $n > m$ there are $n - m$ asymptotes of the root locus that intersect the real axis at $\sigma_a = (\sum p_i - \sum z_i)/(n - m)$, and radiate out with angles $\theta_k = (2k + 1)\pi/(n - m)$, for $k = 0 \dots (n - m - 1)$.

Refinement of the Root Locus:

Breakaway and break-in points on the real axis	There are breakaway or in points of the locus on the real axis wherever $N(s)D'(s) - N'(s)D(s) = 0$.
Angle of departure from a complex pole	<p>The angle of departure from pole p_j is</p> $\theta_{d,p_j} = \pm 180^\circ + \sum_{i=1}^m \angle(p_j - z_i) - \sum_{i=1, i \neq j}^n \angle(p_j - p_i)$
Angle of arrival at complex zero	<p>The angle of arrival at zero z_j is</p> $\phi_{a,z_j} = \pm 180^\circ + \sum_{i=1, i \neq j}^m \angle(z_j - z_i) - \sum_{i=1}^n \angle(z_j - p_i)$
Imaginary axis crossings (stability limits)	Use Routh-Hurwitz to determine where the locus crosses the imaginary axis, or assume a form for the closed-loop char. eqn. and solve for the coefficients
Determine the poles for a given gain K	Substitute the value of K into $D(s) + KN(s) = 0$ and find roots of characteristic equation. (This may require a computer)
Determine K for a given pole location	Use the magnitude condition with $s = \sigma + j\omega$, ie $K = -D(s)/N(s)$. (If s is not <i>exactly</i> on the locus, K may be complex, but the imaginary part should be small. Take the real part of K for your answer.)

2 MATLAB Root Locus Functions

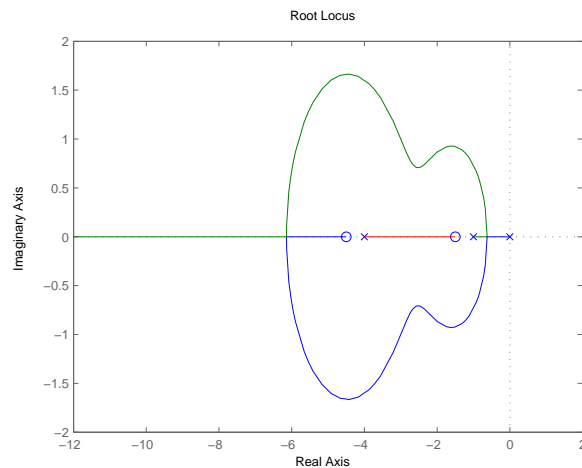
2.1 MATLAB Language Functions

rlocus() The function `rlocus(sys)` produces a root locus plot for the system object `sys`.

For example

```
sys = zpk([-1.5 -4.5], [0 -1 -4], 1)
```

```
rlocus(sys)
```

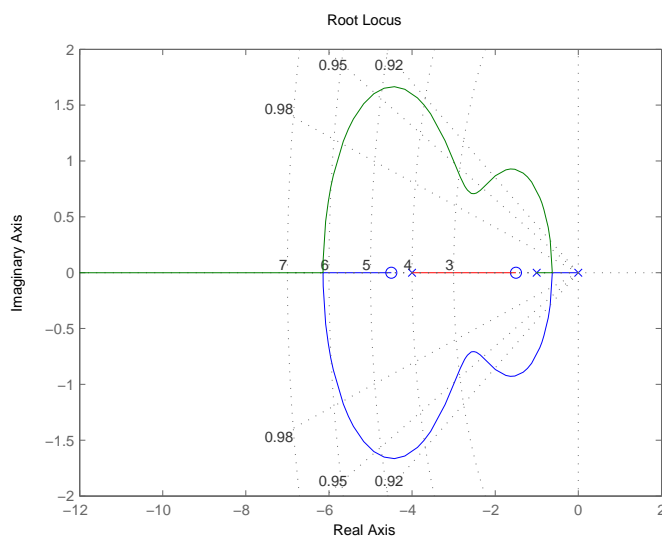


produces the plot

sgrid() The function `sgrid` without any arguments generates a grid over an existing continuous s-plane root locus or pole-zero plot. Lines of constant damping ratio (ζ) and undamped natural frequency (ω_n) are drawn.

If invoked with a pair of arguments, `sgrid(zeta, wn)`, lines of constant damping ratio for the values given in the vectors `zeta` and `wn` will be plotted. For example

`sgrid([0.92 0.95 0.98], [3 4 5 6 7])` superimposed on the above example produces the plot

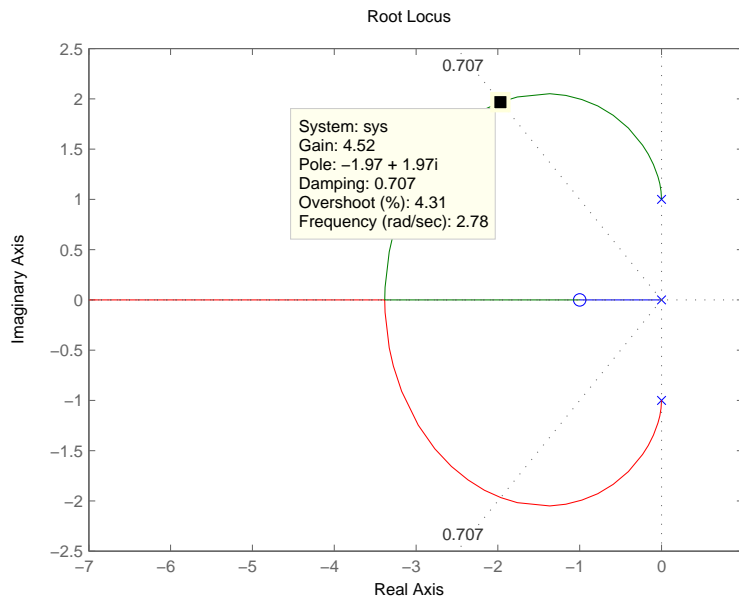


■ Example 3

Use MATLAB to find the gain K to achieve a closed-loop damping ratio of $\zeta = 0.707$ for the open-loop system

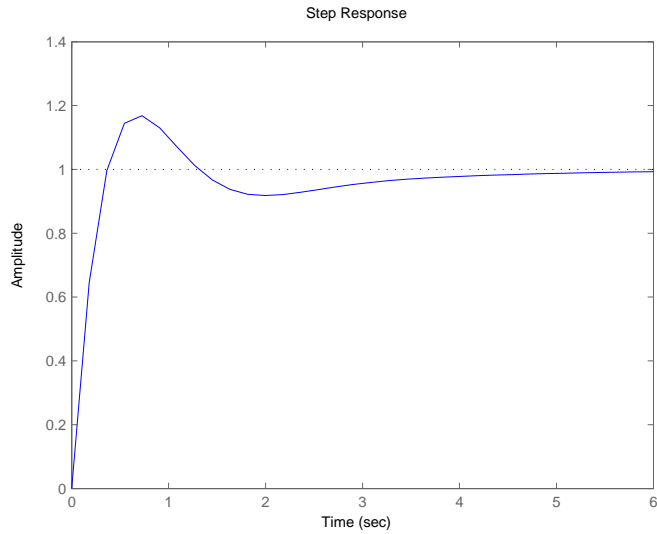
$$G(s) = \frac{(s + 1)^2}{s(s^2 + 1)}$$

The commands `sys = tf([1 2 1],[1 0 1 0])`
`sgrid(0.707, 0)`
produce the plot



With the cursor, the gain is found to be $K = 4.52$. The step response of the closed-loop system is found with the following commands:

```
closed_loop = feedback (4.52*sys, 1)  
step(closed_loop)
```

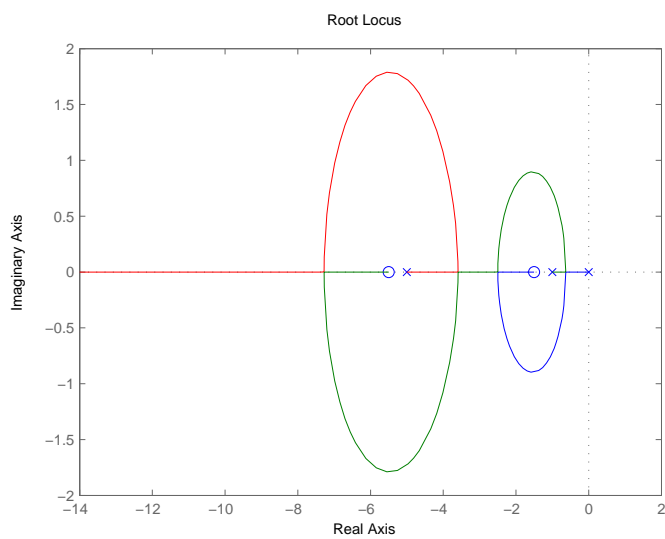
We note that the overshoot is greater than the 5% predicted by the poles with a damping ratio of $\zeta = 0.707$. This is because the closed-loop system retains the open-loop zeros, and these zeros accentuate the overshoot.

■ Example 4

Plot the root locus for

$$G(s) = \frac{(s + 1.5)(s + 5.5)}{s(s + 1)(s + 5)}$$

```
sys = zpk([-1.5 -5.5], [0 -1 -5], 1)
rlocus(sys)
```



2.2 RLTOOL, an Interactive Root Locus Design Tool

RLTOOL is a variant of SISOTOOL, which is an interactive GUI based control system design tool in MATLAB. It will be introduced through a classroom demo.

- The plant $G_p(s)$ may be imported as a system object from the MATLAB workspace.
- The compensator/controller may be defined interactively by placing poles and zeros directly on the root locus plot.
- These poles may be moved around by dragging on the plot.
- Response curves (step, impulse, etc) may be displayed as the gain is changed.