Filters in the s-Plane

NDSU ECE 376

Lecture #26

Inst: Jake Glower

Please visit Bison Academy for corresponding lecture notes, homework sets, and solutions

LaPlace Transforms

Circuits with inductors or capacitors are described by differential equaitons

$$V = L\frac{dI}{dt}$$

$$I = C\frac{dV}{dt}$$

LaPlace transforms assume

$$y = e^{st}$$

giving

$$\frac{dy}{dt} = s\, e^{st} = s\, y$$

Differential Equations and Transfer Functions

With the LaPlace assumption, you can turn differential equations into transfer functions

Example:

$$\frac{d^3y}{dt^3} + 7\frac{d^2y}{dt^2} + 9\frac{dy}{dt} + 15y = 10\frac{dx}{dt} + 3x$$

Using LaPlace notation

$$s^3Y + 7s^2Y + 9sY + 15Y = 10sX + 3X$$

or

$$\boxed{Y = \left(\frac{10s+3}{s^3+7s^2+9s+15}\right)X}$$

G(s) is called the *transfer function*

$$G(s) = \left(\frac{10s+3}{s^3+7s^2+9s+15}\right)$$

Note that this goes both ways:

Example: Find the differential equation relating X and Y

$$Y = \left(\frac{10s+3}{s^3+7s^2+9s+15}\right)X$$

Solution: Cross multiply

$$(s^3 + 7s^2 + 9s + 15)Y = (10s + 3)X$$

Replace each 's' with $\frac{d}{dt}$

$$\boxed{\frac{d^3y}{dt^3} + 7\frac{d^2y}{dt^2} + 9\frac{dy}{dt} + 15y = 10\frac{dx}{dt} + 3x}$$

Analyzing Filtes for Sinusoidal Inputs

G(s) is the gain at all frequencies.

- For a specific frequency, substitute $s \rightarrow j\omega$
- Exprss X in phasor form (real = cosine, imag = -sine)
- Output = Gain * Input

Example: Find y(t):

$$Y = \left(\frac{10s+3}{s^3+7s^2+9s+15}\right) X$$

$$x(t) = 2\cos(4t) + 3\sin(4t)$$

Solution: Express X using phasor notation

$$X = 2 - j3$$

$$s = j4$$

$$Y = \left(\frac{10s+3}{s^3+7s^2+9s+15}\right)_{s=j4} X = (-0.138 - j0.372)(2 - j3)$$

$$Y = -1.394 - j0.330$$

$$\boxed{y(t) = -1.394\cos(4t) + 0.330\sin(4t)}$$

Example 2: Multiple Inputs

- Use superposition

Example: Find y(t)

$$Y = \left(\frac{10s+3}{s^3+7s^2+9s+15}\right)X$$

$$x(t) = 3\cos(4t) + 5\sin(6t)$$

Solution: Treat this as two separate problems:

$$x_1(t) = 3\cos(4t)$$

$$x_2(t) = 5\sin(6t)$$

$x_1(t) = 3\cos(4t)$

$$Y = \left(\frac{10s+3}{s^3+7s^2+9s+15} \right)_{s=j4} (3 + j0) = 0.415 - j1.117$$

$$y_1(t) = 0.415\cos(4t) + 1.117\sin(4t)$$

$x_2(t) = 5\sin(6t)$

$$Y = \left(\frac{10s+3}{s^3+7s^2+9s+15} \right)_{s=j6} (0 - j5) = -0.642 + j0.640$$

$$y_2(t) = -0.642\cos(6t) - 0.640\sin(6t)$$

$$y(t) = y_1 + y_2$$

Filter Analysis: Bode Plots

Easy:

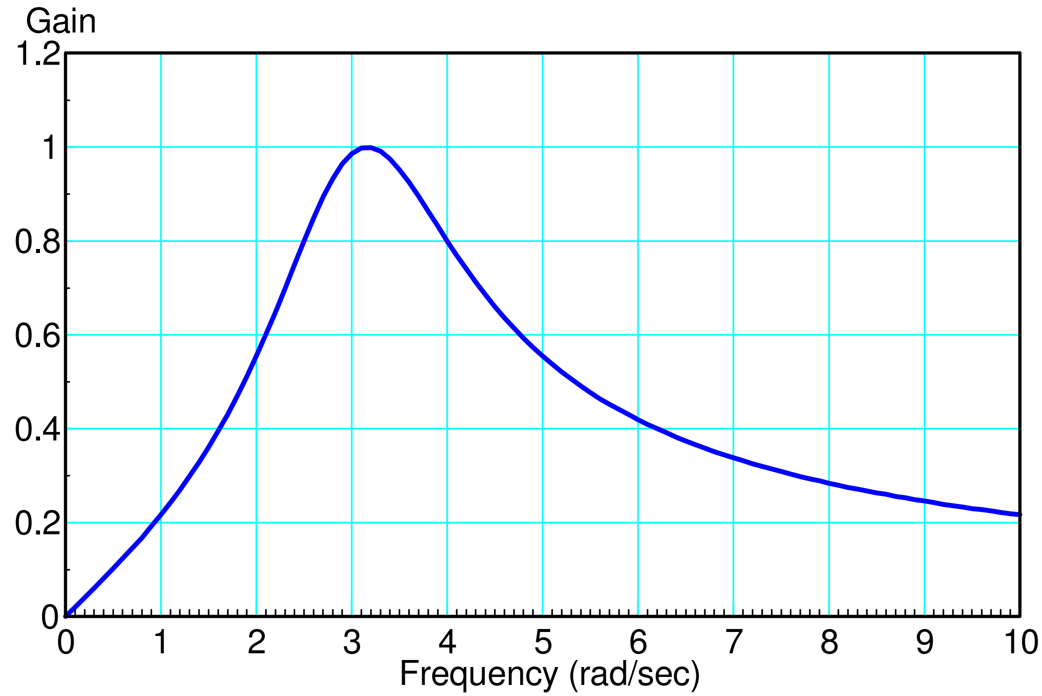
- Plug in $s = j\omega$
- Plot gain vs. frequency

Example:

$$G(s) = \left(\frac{2s}{s^2+2s+10}\right)$$

Matlab Code:

```
w = [0:0.01:10]';
s = j*w;
G = 2*s ./ (s.^2 + 2*s + 10);
plot(w,abs(G));
xlabel('Frequency (rad/sec)');
ylabel('Gain');
```



Filter Design: Poles and Zeros

In general, G(s) will have a numerator and a denominator polynomial

- The zeros are the roots of the numerator polynomial
- The poles are the roots of the denominator polynomial.

$$G(s) = k\left(\frac{z(s)}{p(s)}\right)$$

Graphically, this is

$$G(s) = k \cdot \frac{\Pi(\text{distance from the zeros to j}\omega)}{\Pi(\text{distance from the poles to j}\omega)}$$

meaning

- **Place zeros near frequencies where you want the gain to be small**
- **Place poles near frequencies where you want the gain to be large**

Types of Filters

Filters are categorized into different types:

Filter Type	Characteristic	Example
Low-Pass	Low-frequency gain is large (pass) High-frequency gain is small (reject)	$\left(\frac{10}{s+10}\right)$
High-Pass	High-frequency gain is large (pass) Low-frequency gain is smalle (reject)	$\left(\frac{10s}{s+10}\right)$
Band-Pass	High-frequency gain is small Low frequency gain is small Mid-range frequency is large	$\left(\frac{2s}{(s+1+j50)(s+1-j50)}\right)$

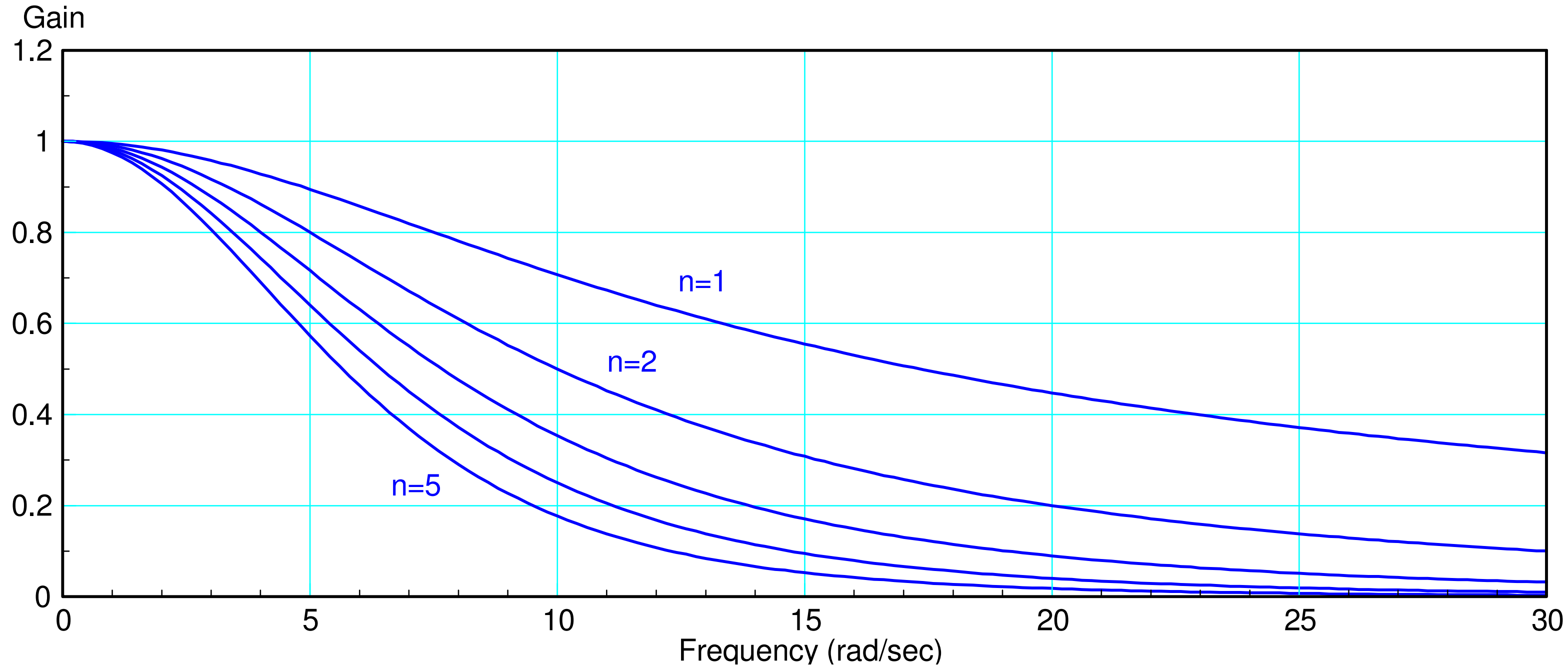
A filter's order is the number of poles the filter has. In general, the more poles a filter has, the better the filter.

RC Filter:

Closest approximation to an ideal low-pass filter with

- Gain < 1
- No zeros
- Poles can only be real

Example: $G(s) = \left(\frac{10}{s+10}\right)^n$



Butterworth Filter

Closest approximation to an ideal low-pass filter with

- Gain < 1
- No zeros
- Poles can be real or complex

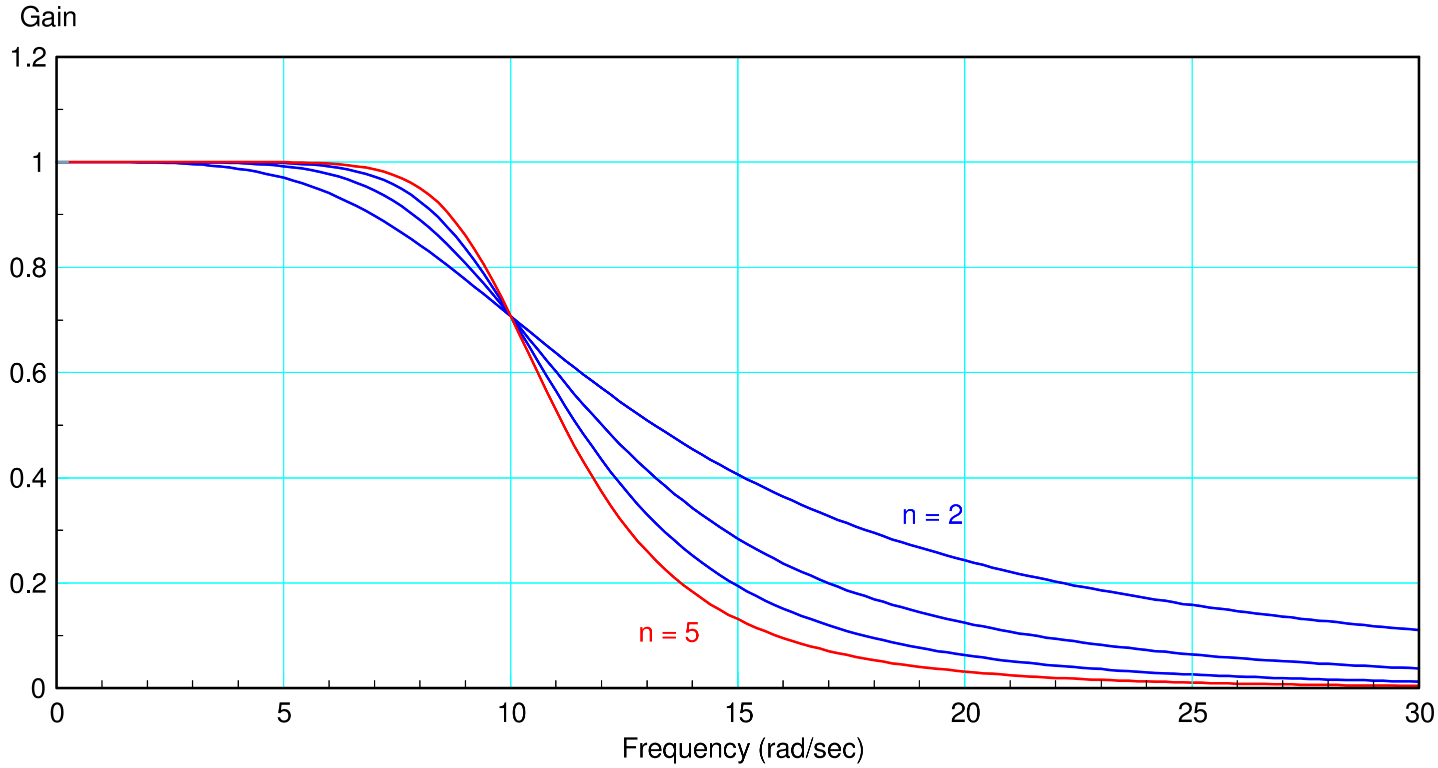
Solution: Pole Locations for Corner = 1 rad/sec

	N=2	N=3	N=4	N=5	N=6
zeros	none	none	none	none	none
poles	$-1\angle\pm 45^0$	-1 $-1\angle\pm 60^0$	$-1\angle\pm 22.5^0$ $-1\angle\pm 67.5^0$	-1 $-1\angle\pm 36^0$ $-1\angle\pm 72^0$	$-1\angle\pm 15^0$ $-1\angle\pm 45^0$ $-1\angle\pm 75^0$

Butterworth Example:

- 5th-Order Butterworth filter
- Corner = 10 rad/sec

$$G(s) = \left(\frac{10^5}{(s+10)\left(s+10\angle\pm 36^0\right)\left(s+10\angle\pm 72^0\right)} \right)$$



Chebychev Filter

Closest approximation to an ideal low-pass filter with

- Gain < 1 + ε
- No zeros
- Poles can be real or complex

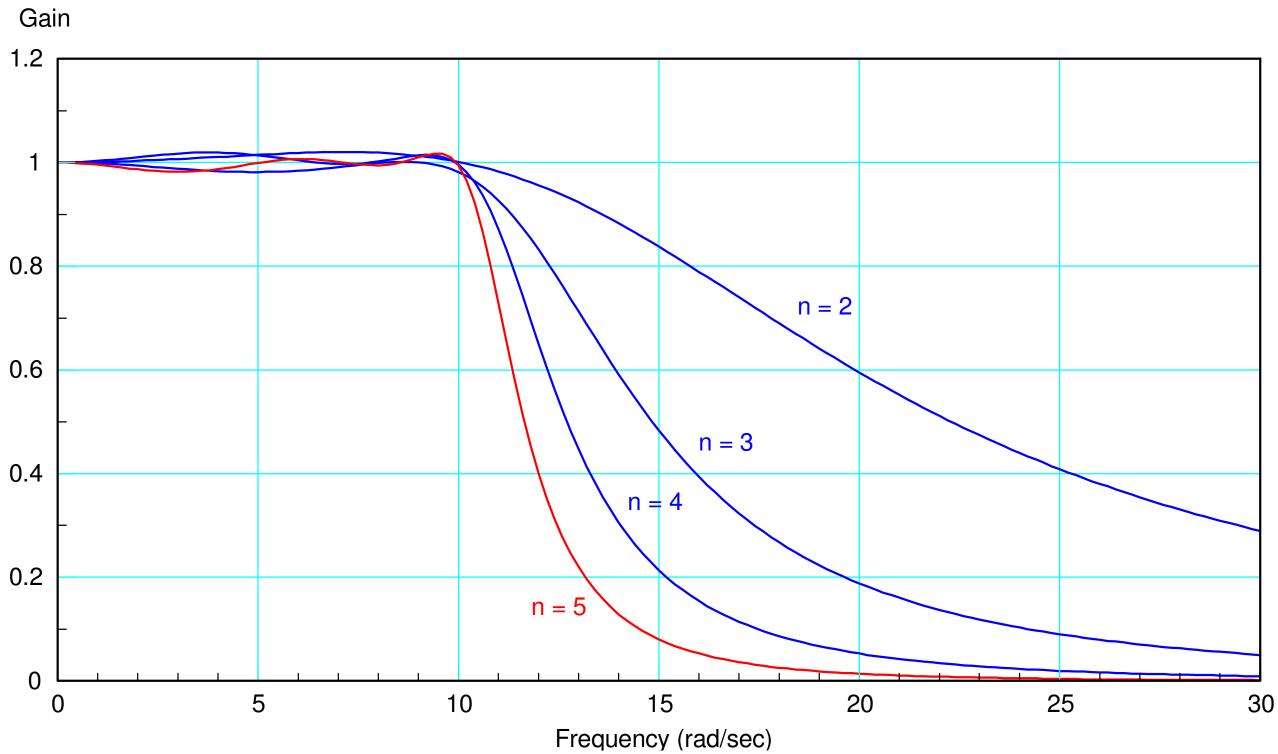
Solution: Pole location for $\varepsilon = 0.02$ and corner = 1 rad/sec

	N=2	N=3	N=4	N=5	N=6
zeros	none	none	none	none	none
poles	$-1.60\angle\pm 50.7^0$	-0.85 $-1.21\angle\pm 69.5^0$	$-0.72\angle\pm 38.5^0$ $-1.11\angle\pm 77.8^0$	-0.48 $-0.76\angle\pm 59.3^0$ $-1.06\angle\pm 82.0^0$	$-0.47\angle\pm 36.1$ $-0.81\angle\pm 69.8$ $-1.04\angle\pm 84.4$

Chebychev Example:

- 5th-Order Butterworth filter
- Corner = 10 rad/sec

$$G(s) = \left(\frac{4.8 \cdot 7.6^2 \cdot 10.6^2}{(s+4.8)\left(s+7.6\angle\pm 59.3^0\right)\left(s+10.6\angle\pm 82^0\right)} \right)$$



Filter Design with *fminsearch()*

Another way to design filters is to use the function *fminsearch*

Problem: Find { a,b,c,d,e } so that

$$G(s) = \left(\frac{a}{\left(s^2+bs+c\right)\left(s^2+ds+e\right)} \right) \approx \begin{cases} 1 & \omega < 4 \\ 0 & otherwise \end{cases}$$

```
function [ J ] = costf( z )
 a = z(1);
 b = z(2);
 c = z(3);
 d = z(4);
 e = z(5);

 w = [0:0.01:10]';
 s = j*w;
 Gideal = 1 .* (w < 4);

 G = a ./  ( (s.^2 + b*s + c) .* (s.^2 + d*s + e) );

 E = abs(Gideal) - abs(G);

 J = sum(E .^ 2);

 end
```

Minimizing the cost:

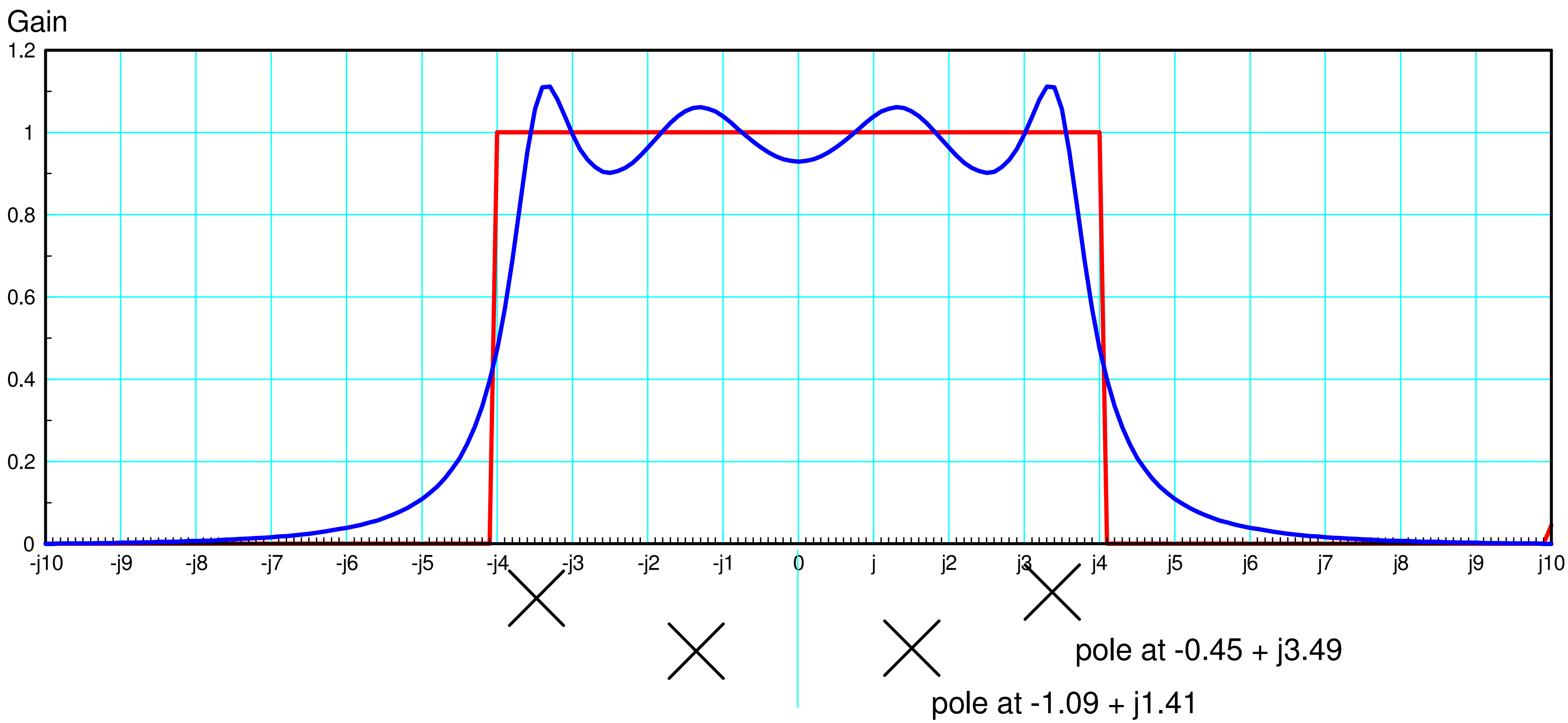
```
>> [a,b] = fminsearch('costf',10*rand(1,5))

a =   36.6716   0.8314   12.3599  2.1860   3.1799
b =   13.0720
```

meaning

$$G(s) = \left(\frac{36.67}{\left(s^2+0.8314s+12.3599\right)\left(s^2+2.1860s+3.1799\right)} \right)$$

The gain vs. frequency and pole location looks like:



Summary

Filters are circuits whose gain changes with frequency

Phasors make filter analysis easy

- Assumes sinusoidal inputs
- Requires the use of complex numbers

Filter design is a little harder

- Place poles close to frequencies you want to pass
- Place zeros close to frequencies you want to reject