Filter Approximations & Frequency Transformations

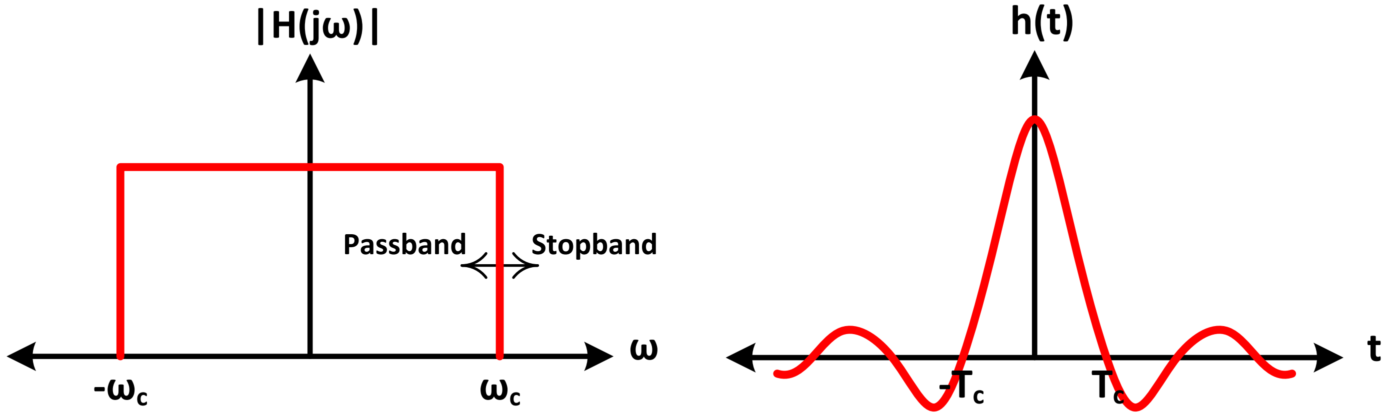
Filter Approximation Concepts

How do you translate filter specifications into a mathematical expression which can be synthesized ?

- Approximation Techniques

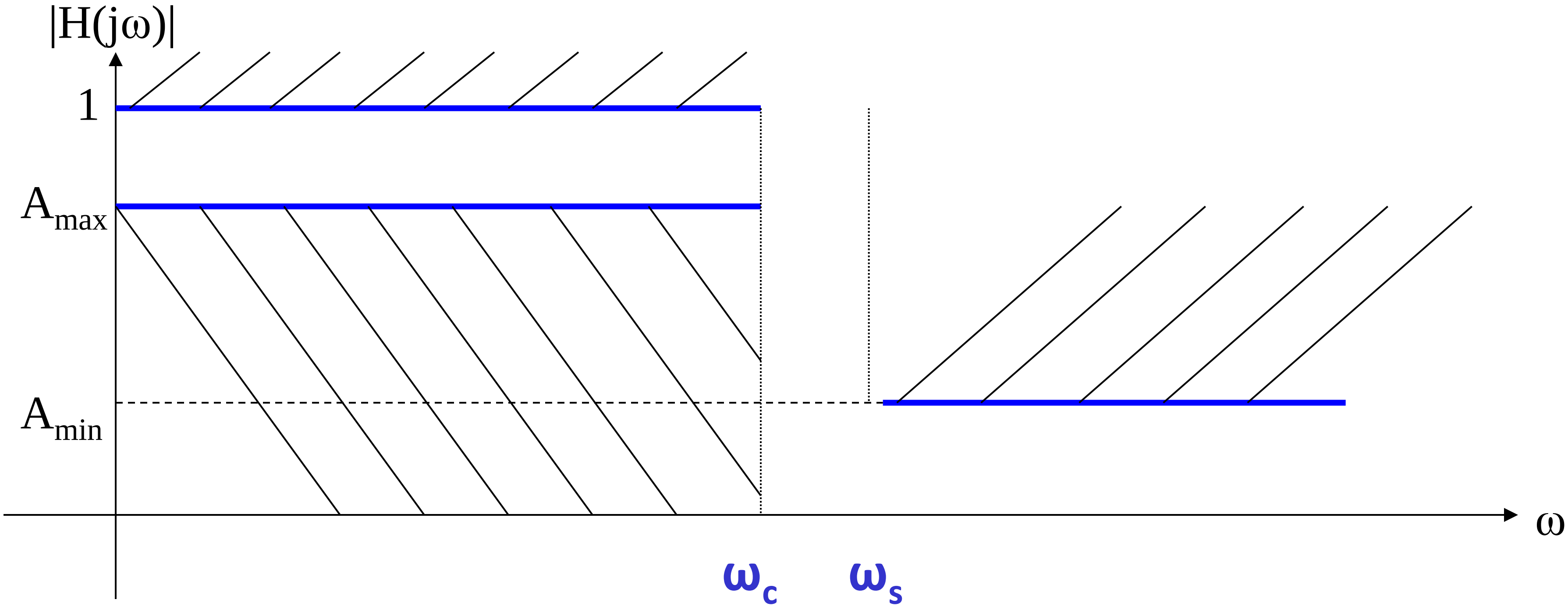
Why an ideal Brick Wall Filter can not be implemented ?

- Causality: Ideal filter is non-causal
- Rationality: No rational transfer function of finite degree (n) can have such abrupt transition



Filter Approximation Concepts

Practical Implementations are given via window specs.



$\mathbf{A_{max} = A_p}$ is the maximum attenuation in the passband

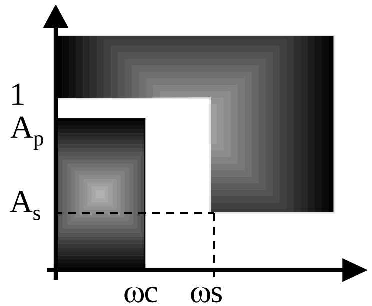
$\mathbf{A_{min} = A_s}$ is the minimum attenuation in the stopband

$\boldsymbol{\omega_s - \omega_c}$ is the Transition Width

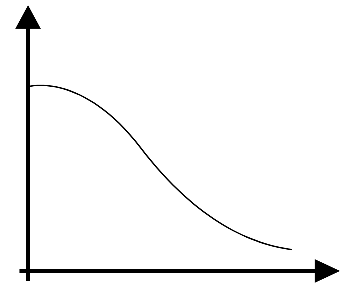
Approximation Types of Lowpass Filter

Definitions

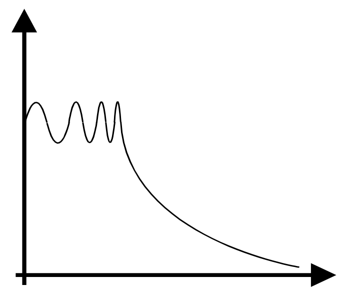
- Ripple = $1\text{-}A_p$
- Stopband attenuation $= A_s$
- Passband (cutoff frequency) $= \omega_c$
- Stopband frequency $= \omega_s$



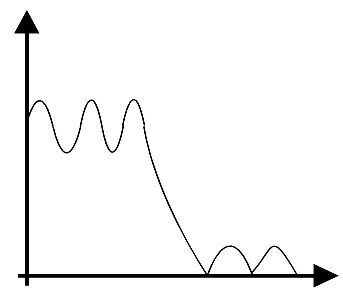
Filter specs



Maximally flat (Butterworth)



Equal-ripple (Chebyshev)



Elliptic

Approximation of the Ideal Lowpass Filter

Since the ideal LPF is unrealizable, we will accept a small error in the passband, a non-zero transition band, and a finite stopband attenuation

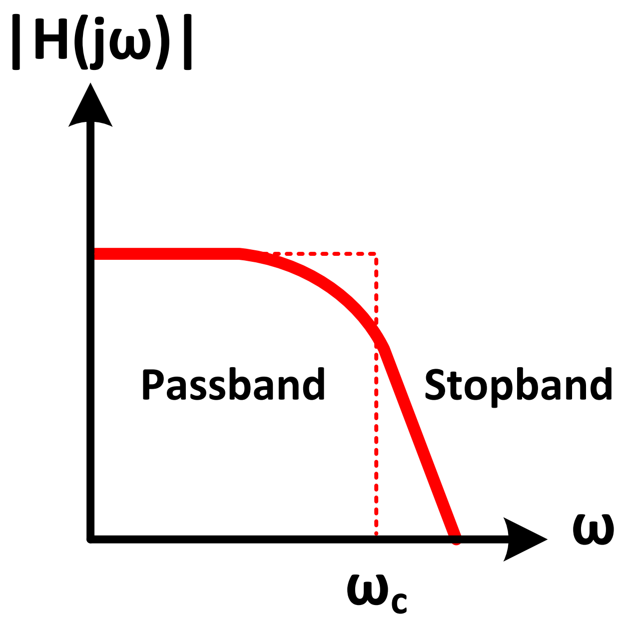
$$|H(j\omega)|^2 = \frac{1}{1 + |K(j\omega)|^2}$$

- $H(j\omega)$: filter's transfer function
- $K(j\omega)$: Characteristic function

(deviation of $|T(j\omega)|$ from unity)

For $0 \leq \omega \leq \omega_c$ $\rightarrow 0 \leq |K(j\omega)| \leq 1$

For $\omega > \omega_c$ $\rightarrow |K(j\omega)|$ increases very fast



Maximally Flat Approximation (Butterworth)

Stephen Butterworth showed in 1930 that the gain of an n^{th} order maximally flat magnitude filter is given by

$$|H(j\omega)|^2 = H(j\omega)H(-j\omega) = \frac{1}{1+\varepsilon^2\omega^{2n}}$$

$K(j\omega) \cong 0$ in the passband in a *maximally flat* sense

$$\left.\frac{d^k\left(|K(j\omega)|^2\right)}{d(\omega^2)^k}\right|_{\omega=0} = 0 \qquad \text{for } k = 1,2,\dots,2n-1$$

The corresponding pole locations (for $\varepsilon = 1$) can be determined as follows

$$|H(s)|^2 = \frac{1}{1+\left(\frac{s}{j}\right)^{2n}} = \frac{1}{1+(-1)^n s^{2n}} \rightarrow \left(s_p\right)^{2n} = -(-1)^{-n} = e^{j\pi(2k-1+n)}$$

$$s_p = e^{j\frac{\pi}{2}\left(\frac{2k-1+n}{n}\right)} \qquad k = 1,\dots,2n$$

Pole Locations: Maximally flat (ε=1)

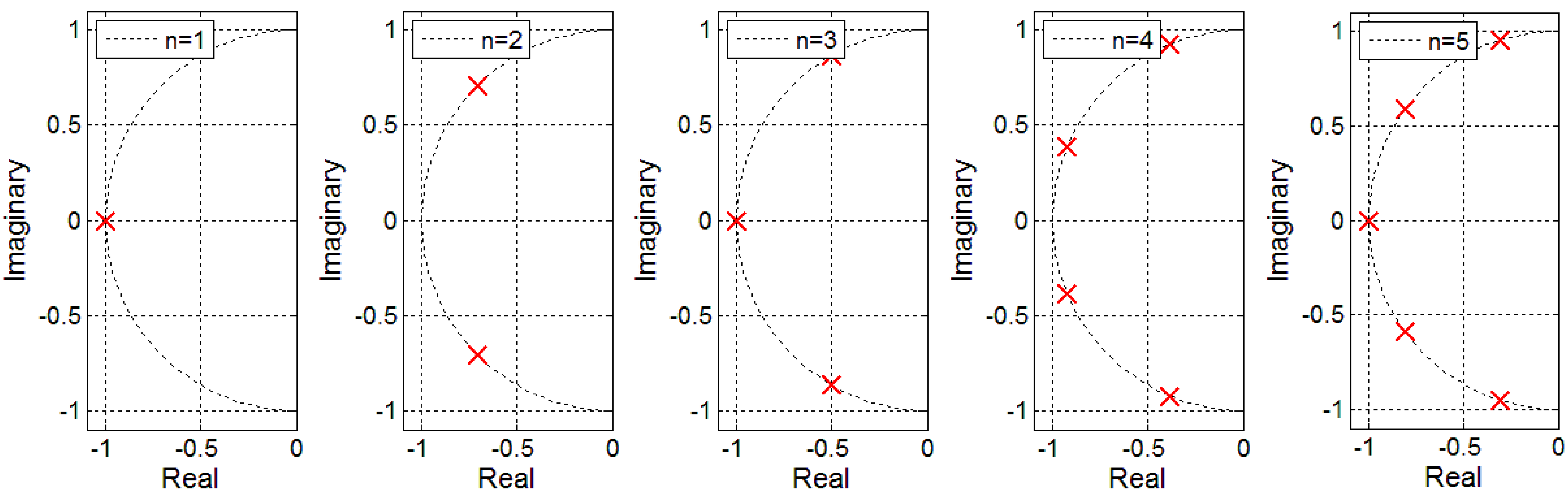
The poles are located on the unity circle at equispaced angles

$$s_p = e^{j\theta_k} \qquad \text{where} \qquad \theta_k = \frac{\pi}{2}\left(\frac{2k-1+n}{n}\right) \qquad k = 1,2,\ldots,2n$$

The real and imaginary parts are

$$Re\{S_p\} = -\sin\left(\frac{2k-1}{n}\frac{\pi}{2}\right) \qquad \text{Im}\{S_p\} = \cos\left(\frac{2k-1}{n}\frac{\pi}{2}\right)$$

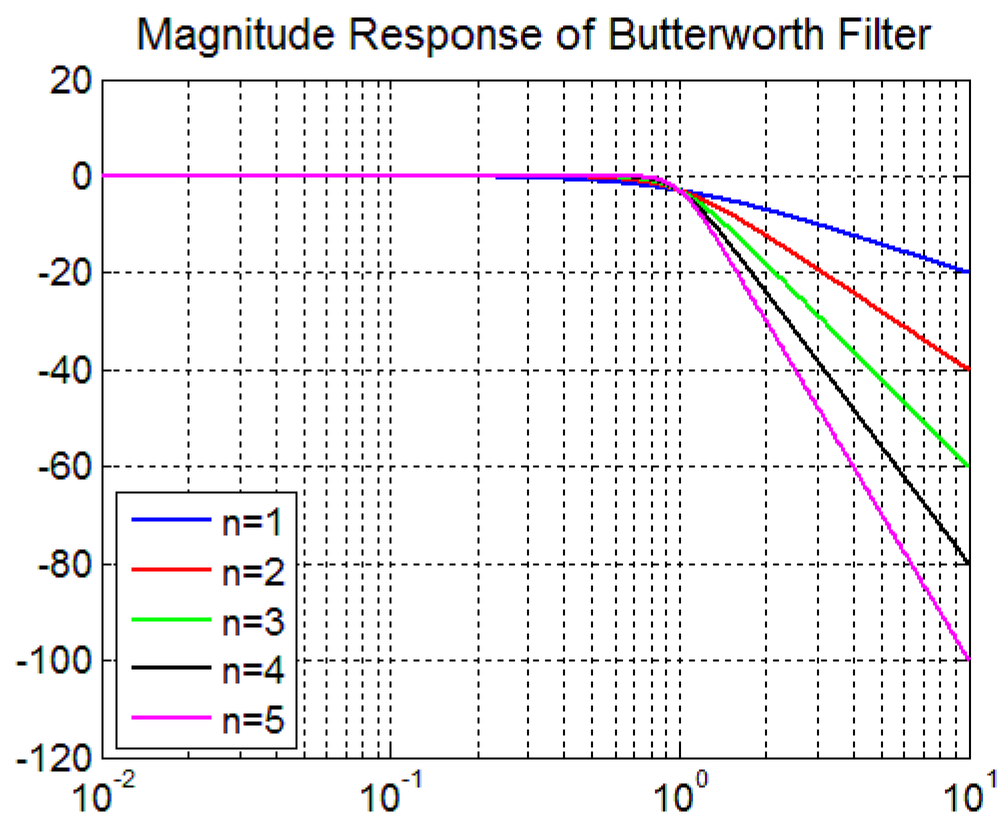
Poles in the LHP are associated with $H(s)$ and poles in the RHP are associated with $H(-s)$



θ_k	0°	±135°	±120°, 180°	±112.5°, ±157.5°	±108°, ±144°, 180°

Magnitude Response of Butterworth Filter

```
clc,clear all,close all
Wp=1;          %Passband cut-off frequency
n=4;           %Order of butterworth filter
[z,p,k]=butter(n,Wp,'s');
s=tf('s');
H=zpk(z,p,k);
w=logspace(-2,1,1000);
[mag,phase]=bode(H,w);
Hdb=20*log10(squeeze(mag));

figure(1)
semilogx(w,Hdb,colors(n),'linewidth',2),hold on;
title('Magnitude Response','fontsize',16)
set(gca,'fontsize',14);
xlabel('\omega [rad/sec]','fontsize',16);
ylabel('Magnitude [dB]','fontsize',16);
legend(['n=' num2str(n)],'Location','southwest')
grid;
```



Design Example

Design a 1-KHz maximally flat lowpass filter with:

- *Attenuation at 10 kHz ≥2000*

❑ Normalized prototype: $\omega_c = 1\ rad/s$, $\omega_s = 10\ rad/s$

$$|H(j\omega_s)|^2 = \frac{1}{1+\omega_s^{2n}} \le \left(\frac{1}{2000}\right)^2 \quad \rightarrow \quad 10^{2n} \ge 4\times 10^6 \quad \text{or} \quad n \ge 3.3$$

Choose $n = 4$

❑ Pole locations

$$s_p = e^{j\theta_k} \qquad \text{where} \qquad \theta_k = \pm 112.5°, \pm 157.5°$$

$$s_{p1,2} = -0.383 \pm j0.924$$

$$s_{p3,4} = -0.924 \pm j0.383$$

❑ Normalized transfer function

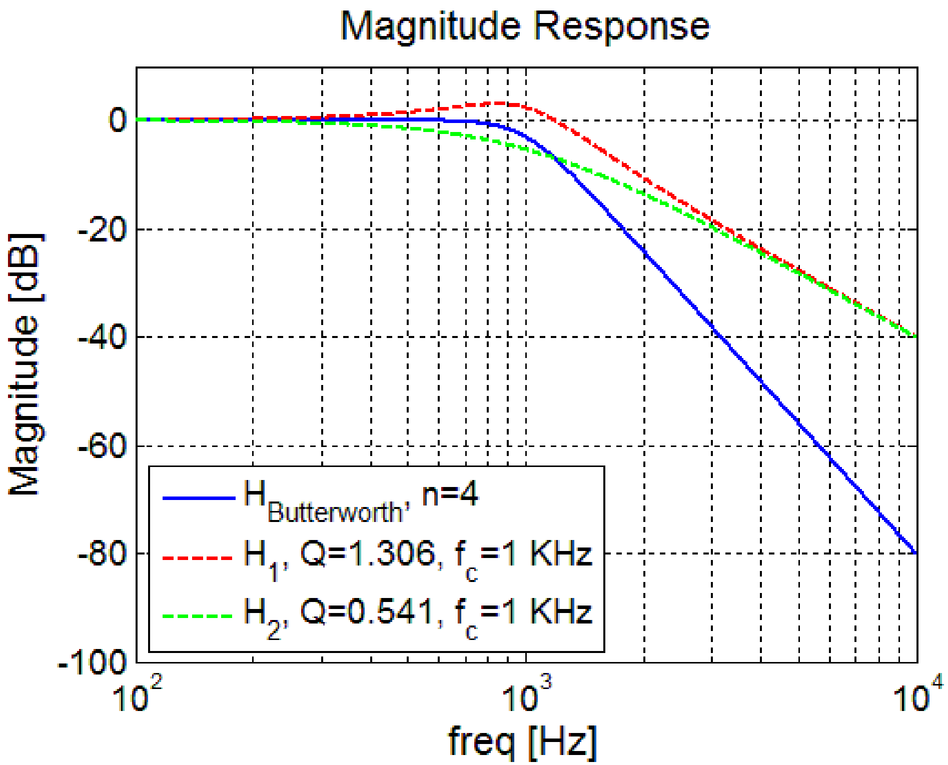
$$H(s) = \frac{1}{\underbrace{(s^2 + 0.765s + 1)}_{H_1:\ Q = 1.306,\ \omega_c = 1}\underbrace{(s^2 + 1.848s + 1)}_{H_2:\ Q = 0.541,\ \omega_c = 1}}$$

Design Example

- Denormalized transfer function $\left(s = \frac{s}{\omega_c}\right)$

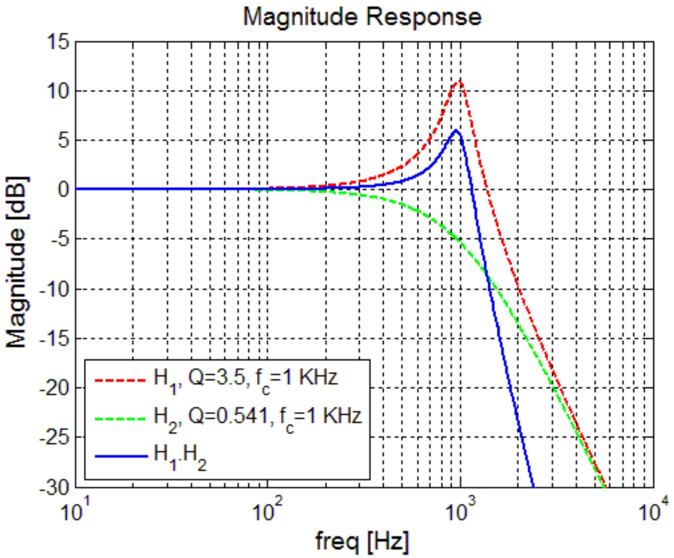
$$H(s) = \frac{1.5585 \times 10^{15}}{\underbrace{(s^2 + (4.8 \times 10^3)s + 3.95 \times 10^7)}_{H_1}\underbrace{(s^2 + (1.16 \times 10^4)s + 3.95 \times 10^7)}_{H_2}}$$

$Q = 1.306$, $\omega_c = 2\pi 10^3$ $Q = 0.541$, $\omega_c = 2\pi 10^3$

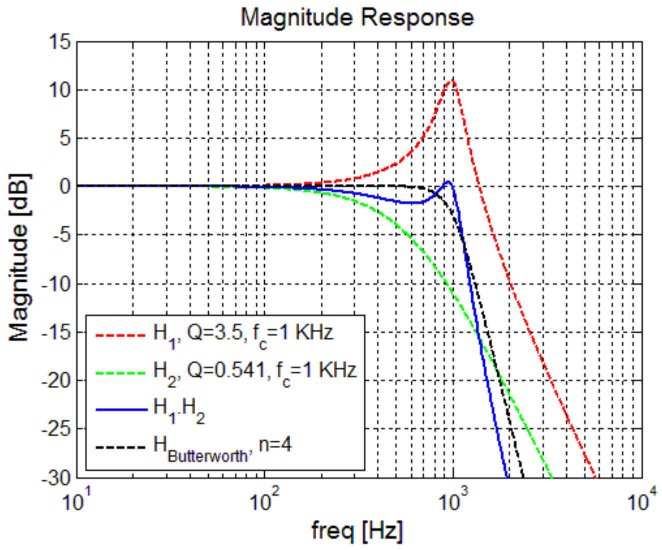


Design Example -Discussion

- We can sharpen the transition in the previous example by increasing the quality factor of one of the two cascaded filters (Q_1=3.5 instead of 1.3)



- To alleviate the peaking problem in the previous response, we can reduce ω_{c2} ($\omega'_{c2} = 0.6\omega_{c1}$)



- Passband ripples are now existing
  - They can be tolerated in some applications
- The resulting response has steeper transition than Butterworth response

Equiripple Filter Approximation (Chebyshev I)

This type has a steeper transition than Butterworth filters of the same order but at the expense of higher passband ripples

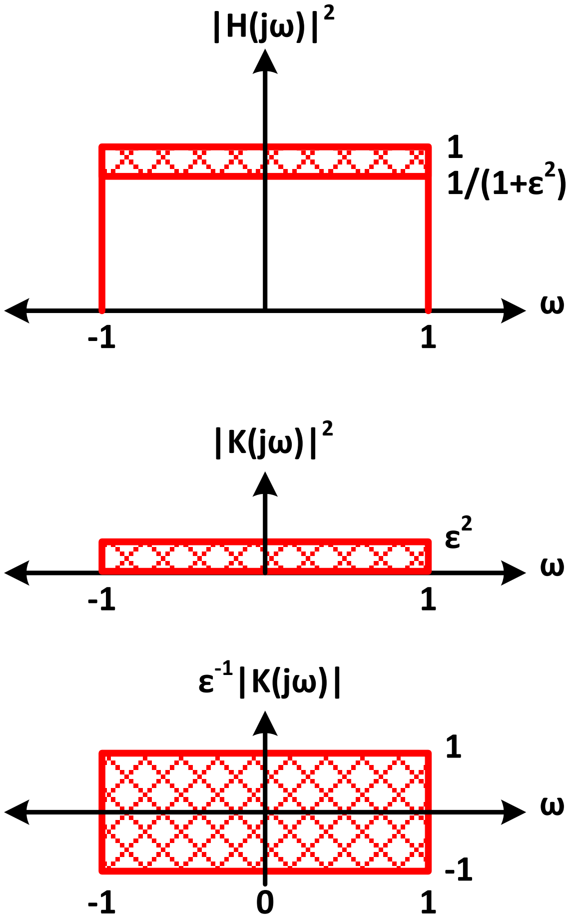
Magnitude response of this type is given by

$$|H(j\omega)|^2 = \frac{1}{1+|K(j\omega)|^2}$$

$$|K(j\omega)| = \varepsilon C_n(\omega)$$

$$C_n(\omega) = \underbrace{\cos(n\cos^{-1}\omega)}_{\text{Going back and forth in } \pm 1 \text{ range for } |\omega| \le 1} \quad |\omega| \le 1$$

C_n is called Chebyshev's polynomial



Defining the passband area of $|K(j\omega)|$

Equiripple Filter Approximation (Chebyshev I)

Chebyshev's Polynomial

$$C_n(\omega) = \cos(n\cos^{-1}\omega) = \frac{e^{jn\phi}+e^{-jn\phi}}{2}$$

For the stopband ($\omega > 1$)

$$\phi = \cos^{-1}(\omega) \text{ is complex}$$

Thus, $C_n > 1$

Since

$$\cos(n\phi) = \cosh(nj\phi)$$

and

$$j\phi = \cosh(\omega)$$

Then

$$C_n(\omega) = \cosh(n\cosh^{-1}(\omega)) \quad \text{for} \quad |\omega| > 1$$

n	C_n
0	1
1	ω
2	$2\omega^2 - 1$
3	$4\omega^3 - 3\omega$
4	$8\omega^4 - 8\omega^2 + 1$
n	$2\omega C_{n-1} - C_{n-2}$

Properties of the Chebyshev polynomials

$$C_n = \cos(n \cdot \cos^{-1}(\omega)) \le 1 \quad \omega \le 1$$

Faster response in the stopband
⬇For n>3 and ω>1

In the passband, ω<1, Cn is limited to ±1, then, the ripple is determined by ε

$$\frac{\partial}{\partial\omega} C_n^2 \qquad \begin{matrix}\text{Butterworth} \\ 2n\omega^{2n-1}\end{matrix} \qquad \begin{matrix}\text{Chebyshev} \\ \approx \left(2^3\right)^{n-1} \omega^{2n-1}\end{matrix}$$

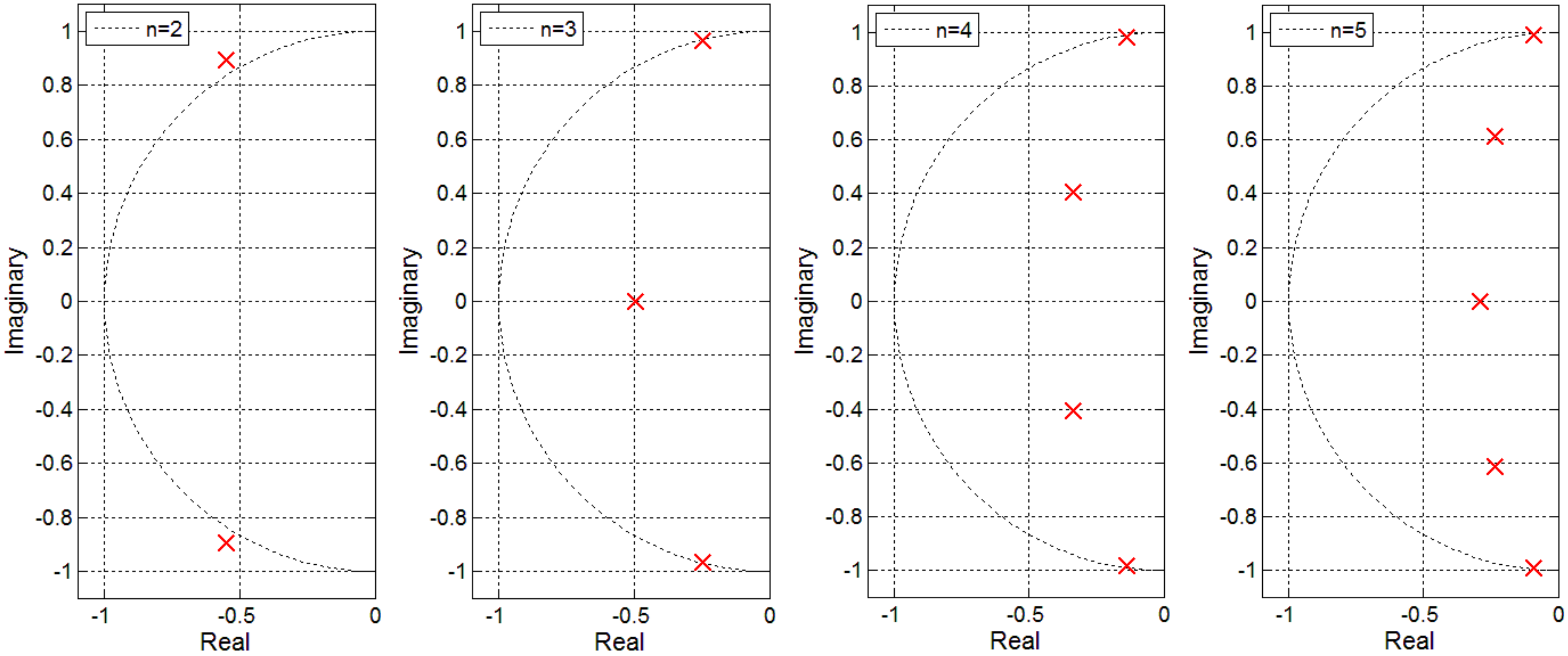
$$\left|N(j\omega)\right|^2 = \frac{1}{1+\varepsilon^2 C_n^2}$$

The -3 dB frequency can be found as:

$$\varepsilon^2 C_n^2(\omega_{-3dB}) = 1 \qquad \omega > 1$$

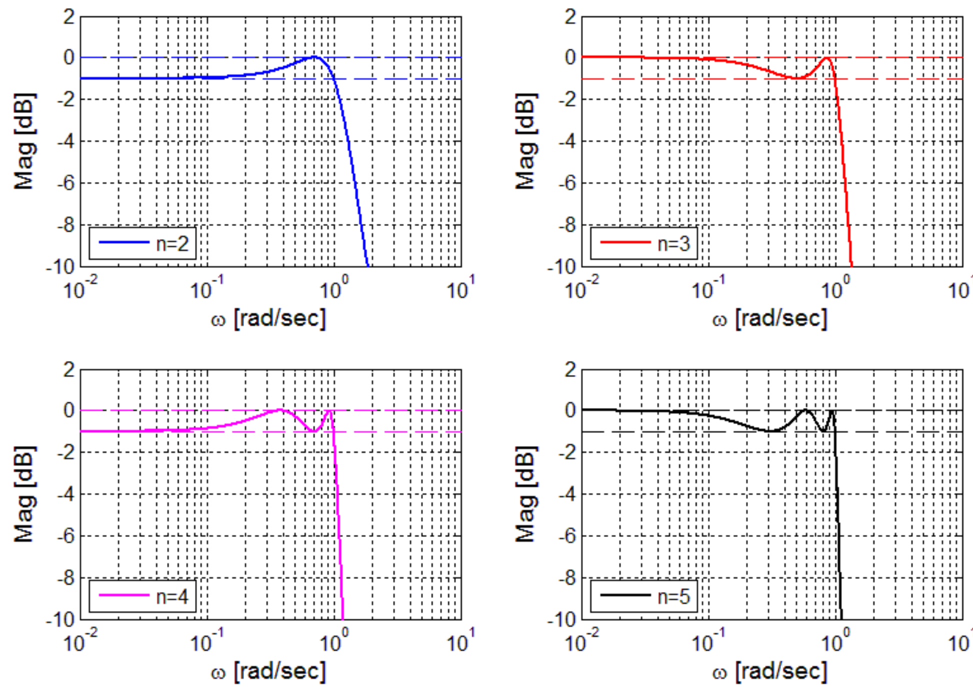
$$\omega_{-3dB} = \cosh\left(\frac{1}{n}\cosh^{-1}\frac{1}{\varepsilon}\right) \qquad \omega > 1$$

Pole Locations (Chebyshev Type I)



Magnitude Response (Chebyshev Type I)

```
clc,clear all,close all
Wp=1;                   %Edge of Passband
n=4;                    %Order of Chebyshev's filter
R=1;                    %Passband Ripples in dB
[z,p,k]=cheby1(n,R,Wp,'s');
H=zpk(z,p,k);
w=logspace(-2,1,1000);
[mag,phase]=bode(H,w);
Hdb=20*log10(squeeze(mag));

figure(1)
semilogx(w,Hdb,'linewidth',2);
hold on
plot([w(1) w(end)],[-1 -1],'--b','linewidth',1)
title('Magnitude Response','fontsize',16)
set(gca,'fontsize',14);
xlabel('\omega [rad/sec]','fontsize',16);
ylabel('Magnitude [dB]','fontsize',16);
legend('H_{cheby1}, n=4','Location','southwest')
axis([1e-2,1e1,-30,2])
grid;
```



Comparison of Design Steps for Maximally Flat and Chebyshev Cases

Step	Maximally Flat	Chebyshev
Find n	$n = \frac{\log[(10^{0.1\alpha_{min}} - 1)/(10^{0.1\alpha_{max}} - 1)]}{2\log(\omega_s)}$ Round up to an integer	$n = \frac{\cosh^{-1}[(10^{0.1\alpha_{min}} - 1)/(10^{0.1\alpha_{max}} - 1)]^{0.5}}{2\cosh^{-1}(\omega_s)}$ Round up to an integer
Find ε	$(10^{0.1\alpha_{max}} - 1)^{1/2}$	$(10^{0.1\alpha_{max}} - 1)^{1/2}$
Find pole locations	If n is odd $\theta_k = 0°, \pm k\frac{180°}{n}$ If n is even $\theta_k = \pm k\frac{180°}{2n}$ Radius $= \Omega_0 = \varepsilon^{-1/n}\omega_p$ $-\sigma_k = \Omega_0 \cos(\theta_k)$ $\pm\omega_k = \Omega_0 \sin(\theta_k)$	Find θ_k as in Butterworth case Find $\alpha = \frac{1}{n}\sinh^{-1}\left(\frac{1}{\varepsilon}\right)$ Then, $-\sigma_k = \sin(\theta_k)\sinh(\alpha)$ $\pm\omega_k = \cos(\theta_k)\cosh(\alpha)$

Inverse Chebyshev Approximation (Chebyshev Type II)

This type has

- Steeper transition compared to Butterworth filters (but not as steep as type I)
- No passband ripples
- Equal ripples in the stopband

Magnitude response of this type is given by

$$|H(j\omega)|^2 = \frac{1}{1+|K(j\omega)|^2}$$

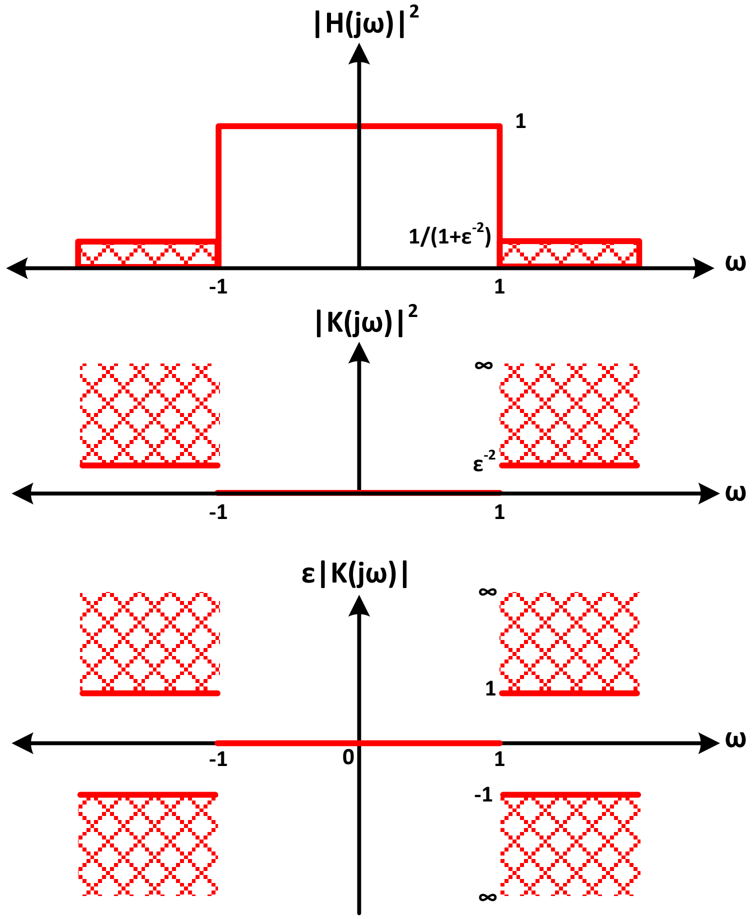
$$|K(j\omega)| = \frac{1}{\varepsilon C_n\left(\frac{1}{\omega}\right)}$$

$$C_n\left(\frac{1}{\omega}\right) = \cos\left(n\cos^{-1}\frac{1}{\omega}\right) \qquad \left|\frac{1}{\omega}\right| \leq 1$$

$$|\omega| \geq 1$$

Going back and forth in ± 1 range for $|\omega| \geq 1$

C_n is Chebyshev's polynomial



Defining the stopband area of $|K(j\omega)|$

Inverse Chebyshev Approximation (Chebyshev Type II)

For the passband ($\omega < 1$)

$$C_n\left(\frac{1}{\omega}\right) = \cosh\left(n \cosh^{-1}\left(\frac{1}{\omega}\right)\right) \qquad \text{for } \omega < 1$$

$$\cong 2^{n-1}\left(\frac{1}{\omega}\right)^n \qquad \text{for } \omega \ll 1$$

Attenuation α

$$\alpha = 10\log\left(1 + \frac{1}{\varepsilon^2 C_n^2\left(\frac{1}{\omega}\right)}\right)\ dB$$

$$\alpha_{max} = 10\log\left(1 + \frac{1}{\varepsilon^2 C_n^2\left(\frac{1}{\omega_p}\right)}\right) \qquad \alpha_{min} = 10\log\left(1 + \frac{1}{\varepsilon^2}\right)$$

To find the required order for a certain filtering template

$$C_n^2\left(\frac{1}{\omega_p}\right) = \cosh\left(n \cosh^{-1}\left(\frac{1}{\omega_p}\right)\right) = \left[\frac{10^{\alpha_{min}/10}-1}{10^{\alpha_{max}/10}-1}\right]^{1/2}$$

$$n = \frac{\cosh^{-1}\left[\left(10^{\alpha_{min}/10} - 1\right)/\left(10^{\alpha_{max}/10} - 1\right)\right]^{1/2}}{\cosh^{-1}\left(\frac{1}{\omega_p}\right)}$$

Pole/zero Locations (Inverse Chebyshev)

Pole/zero locations

$$|H(j\omega)|^2 = \frac{1}{1+\varepsilon^{-2}C_n^{-2}\left(\frac{1}{\omega}\right)} = \frac{\varepsilon^2 C_n^2\left(1/\omega\right)}{1+\varepsilon^2 C_n^2\left(1/\omega\right)}$$

We have imaginary zeros at $\pm\omega_{z,k}$ where

$$C_n^2\left(\frac{1}{\omega_{z,k}}\right) = 0$$

$$\omega_{z,k} = \sec\left(\frac{k\pi}{2n}\right) \quad , k = 1,3,5,\ldots,n$$

If $s_k = \sigma_k + j\omega_k$ are the poles of Chebyshev filter

Then,

$$p_k = \alpha_k + j\beta_k = \frac{1}{s_k}$$ are the poles of inverse Chebyshev filter

Magnitude and quality factor of imaginary poles

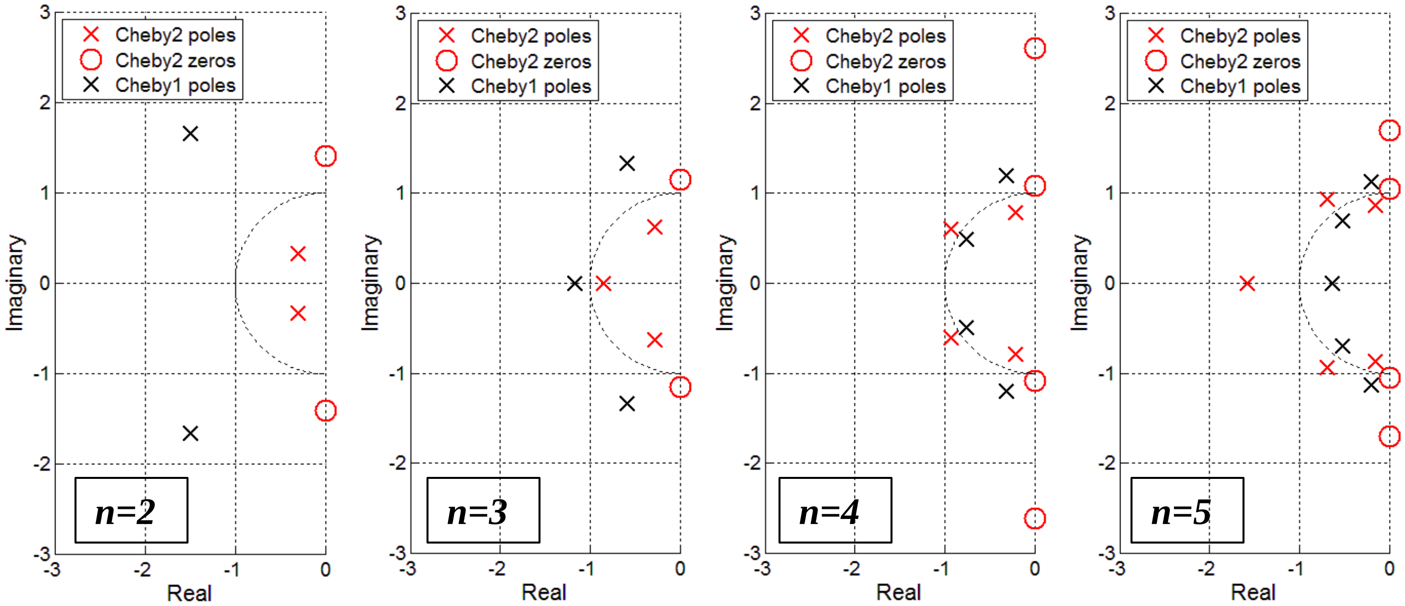
$$|p_k| = \frac{1}{|s_k|} \qquad Q_{iCheb} = Q_{Cheb}$$

Pole/zero Locations (Inverse Chebyshev)

Poles of Chebyshev and inverse Chebyshev filters are reciprocal

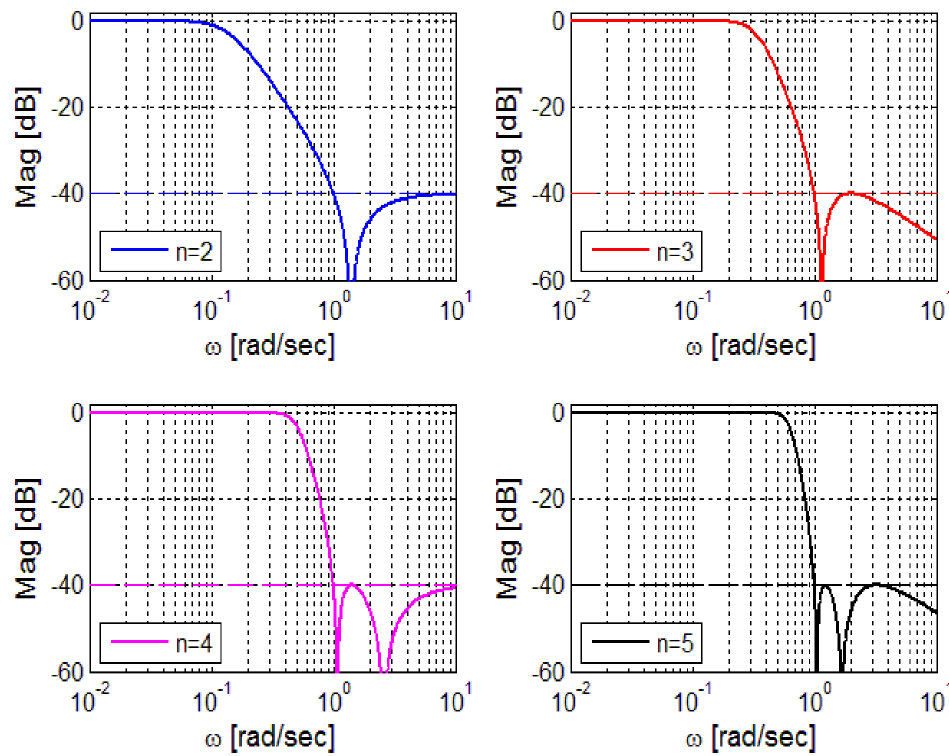
Since the poles are on the radial line, they have the same pole Q

Imaginary zeros creates nulls in the stopband



Magnitude Response (Inverse Chebyshev)

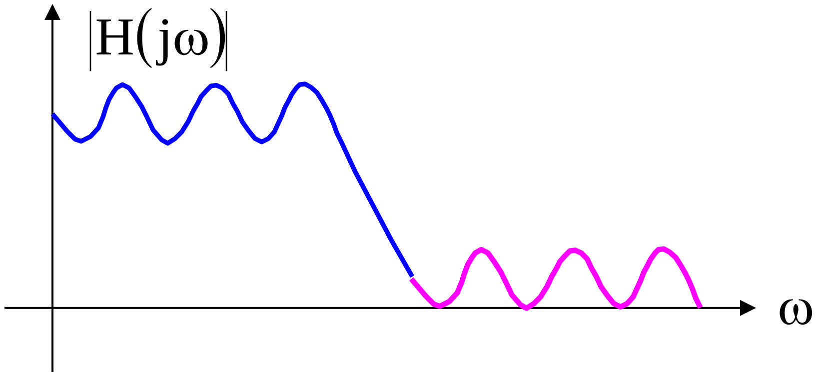
```matlab
clc,clear all,close all
Ws=1;                    %Edge of Stopband
n=4;                     %Order of Inverse Chebyshev filter
R=40;                    %Stopand Ripples
                         %(Minimum stopband attenuation in dB)

[z,p,k]=cheby2(n,R,Ws,'s');
H=zpk(z,p,k);
w=logspace(-2,1,1000);
[mag,phase]=bode(H,w);
Hdb=20*log10(squeeze(mag));

figure(1)
semilogx(w,Hdb,'b','linewidth',2);
hold on
plot([w(1) w(end)],[-R -R],'--b','linewidth',1)
axis([1e-2,1e1,-60,2])
legend(['n=' num2str(n)],'Location','southwest')
set(gca,'fontsize',12);
xlabel('\omega [rad/sec]','fontsize',14);
ylabel('Mag [dB]','fontsize',14);
grid;
```

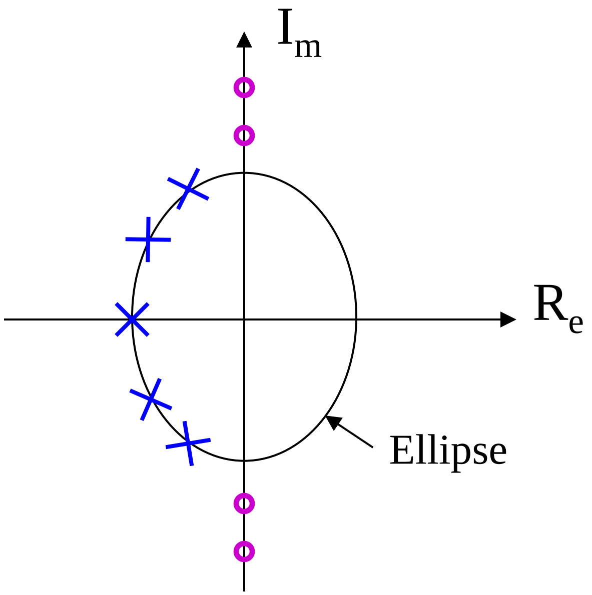


Elliptic Filter Approximation

Elliptic filter

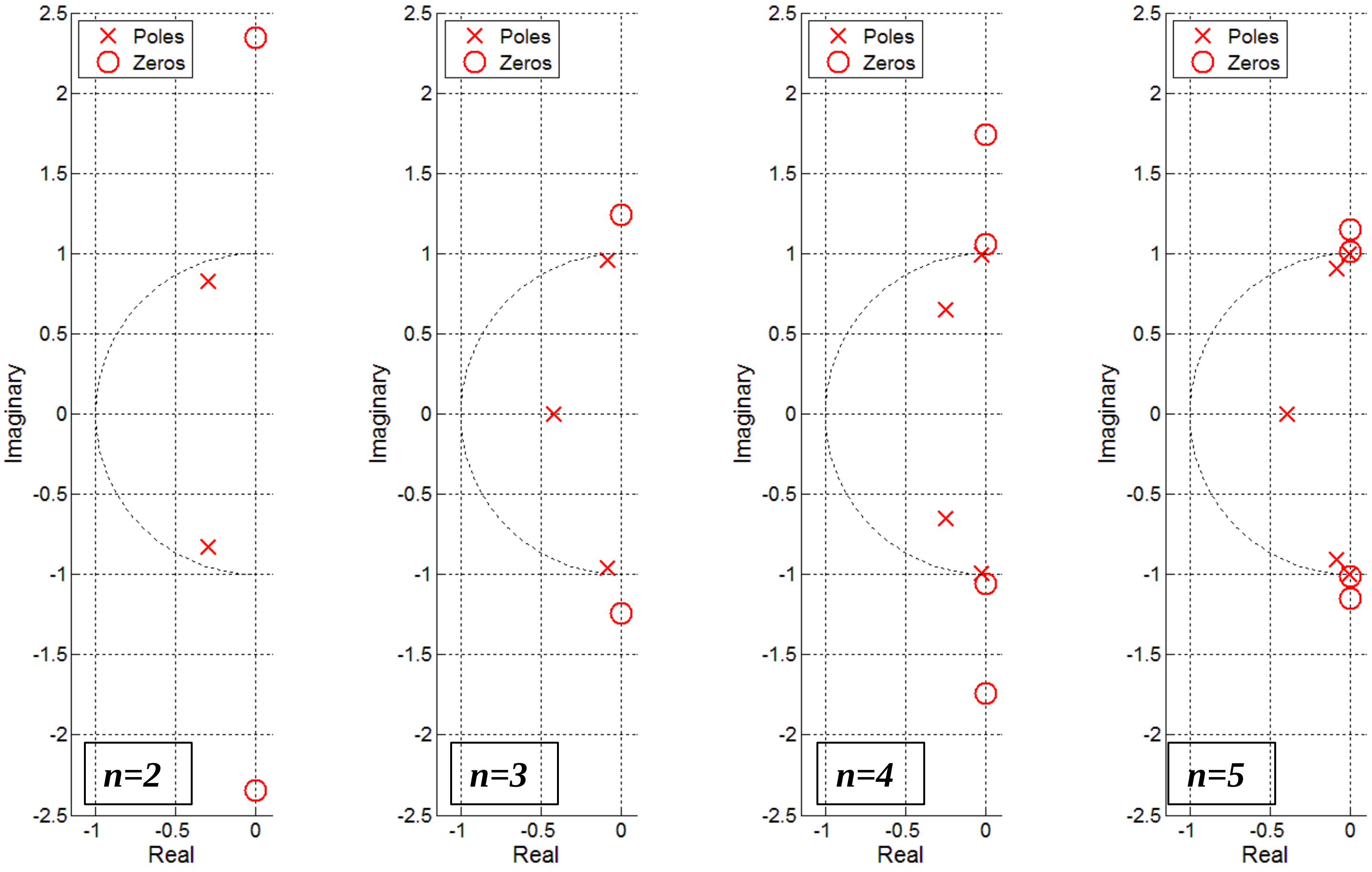
- Equal ripple passband and stopband
- Nulls in the stopband
- Sharpest transition band compared to same-order Butterworth and Chebyshev (Type I and II)





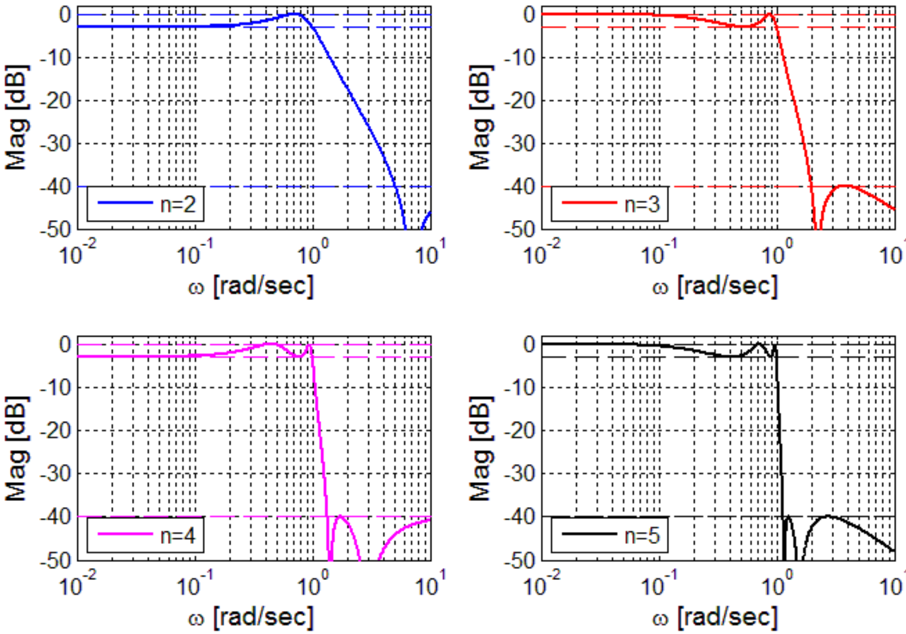
Pole/zero Locations (Elliptic)

Imaginary zeros creates nulls in the stopband



Magnitude Response (Elliptic)

```
clc,clear all,close all
Wp=1;                    %Edge of Passband
n=4;                     %Order of Elliptical filter
Rp=3;                    %Passband Ripple in dB
Rs=40;                   %Stopband Ripple in dB
                         %(Minimum stopband attenuation in dB)

[z,p,k]=ellip(n,Rp,Rs,Wp,'s');
H=zpk(z,p,k);
w=logspace(-2,1,1000);
[mag,phase]=bode(H,w);
Hdb=20*log10(squeeze(mag));

figure(1)
semilogx(w,Hdb,'b','linewidth',2);
hold on
plot([w(1) w(end)],[0 0],'--b','linewidth',1)
plot([w(1) w(end)],[-Rp -Rp],'--b','linewidth',1)
plot([w(1) w(end)],[-Rs -Rs],'--b','linewidth',1)
axis([1e-2,1e1,-60,2])
legend(['n=' num2str(n)],'Location','southwest')
set(gca,'fontsize',12);
xlabel('\omega [rad/sec]','fontsize',14);
ylabel('Mag [dB]','fontsize',14);
grid;
```



Design Example

Design a lowpass filter with:

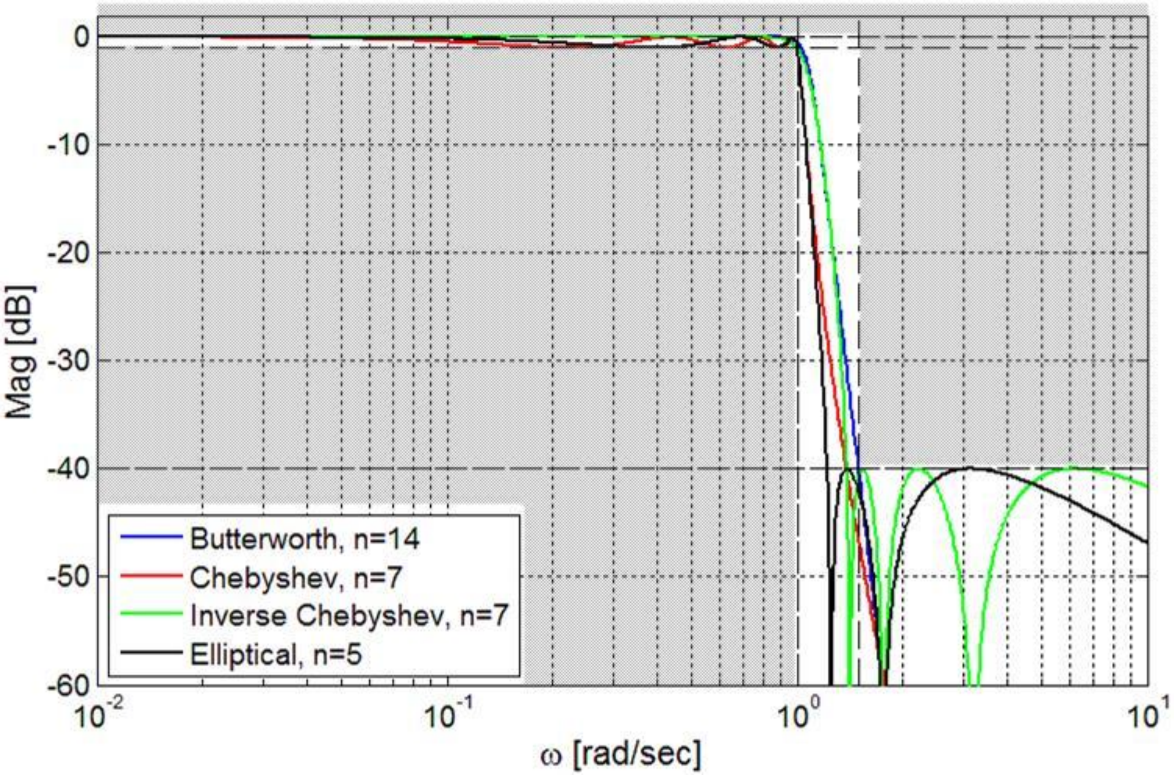
- $\omega_p = 1$ $\qquad R_p = 1\ dB$
- $\omega_s = 1.5$ $\qquad R_s = 40\ dB$

```
clc,clear all,close all
Wp=1;                   %Edge of Passband
Ws=1.5;                 %Edge of Stopband
Rp=1;                   %Passband Ripple in dB
Rs=40;                  %Stopband Ripple in dB
                        %(Minimum stopband attenuation in dB)

w=logspace(-2,1,1000);  %Plotting frequency range

%Butterworth Filter Design
[NB,WB]=buttord(Wp,Ws,Rp,Rs,'s');
[z,p,k]=butter(NB,WB,'s');
Hbut=zpk(z,p,k);
[mag,phase]=bode(Hbut,w);
Hbut_db=20*log10(squeeze(mag));

%Chebyshev I Design
[Nch1,Wchp]=cheb1ord(Wp,Ws,Rp,Rs,'s');
[z,p,k]=cheby1(Nch1,Rp,Wchp,'s');
Hch1=zpk(z,p,k);
[mag,phase]=bode(Hch1,w);
Hch1_db=20*log10(squeeze(mag));

%Chebyshev II Design
[Nch2,Wchs]=cheb2ord(Wp,Ws,Rp,Rs,'s');
[z,p,k]=cheby2(Nch2,Rs,Wchs,'s');
Hch2=zpk(z,p,k);
[mag,phase]=bode(Hch2,w);
Hch2_db=20*log10(squeeze(mag));

%Elliptical Design
[Nel,Wel]=ellipord(Wp,Ws,Rp,Rs,'s');
[z,p,k]=ellip(Nel,Rp,Rs,Wel,'s');
Hel=zpk(z,p,k);
[mag,phase]=bode(Hel,w);
Hel_db=20*log10(squeeze(mag));
```

Matlab function *buttord*, *cheb1ord*, *cheb2ord*, and *ellipord* are used to find the least order filters that meet the given specs.

Design Example

Magnitude response

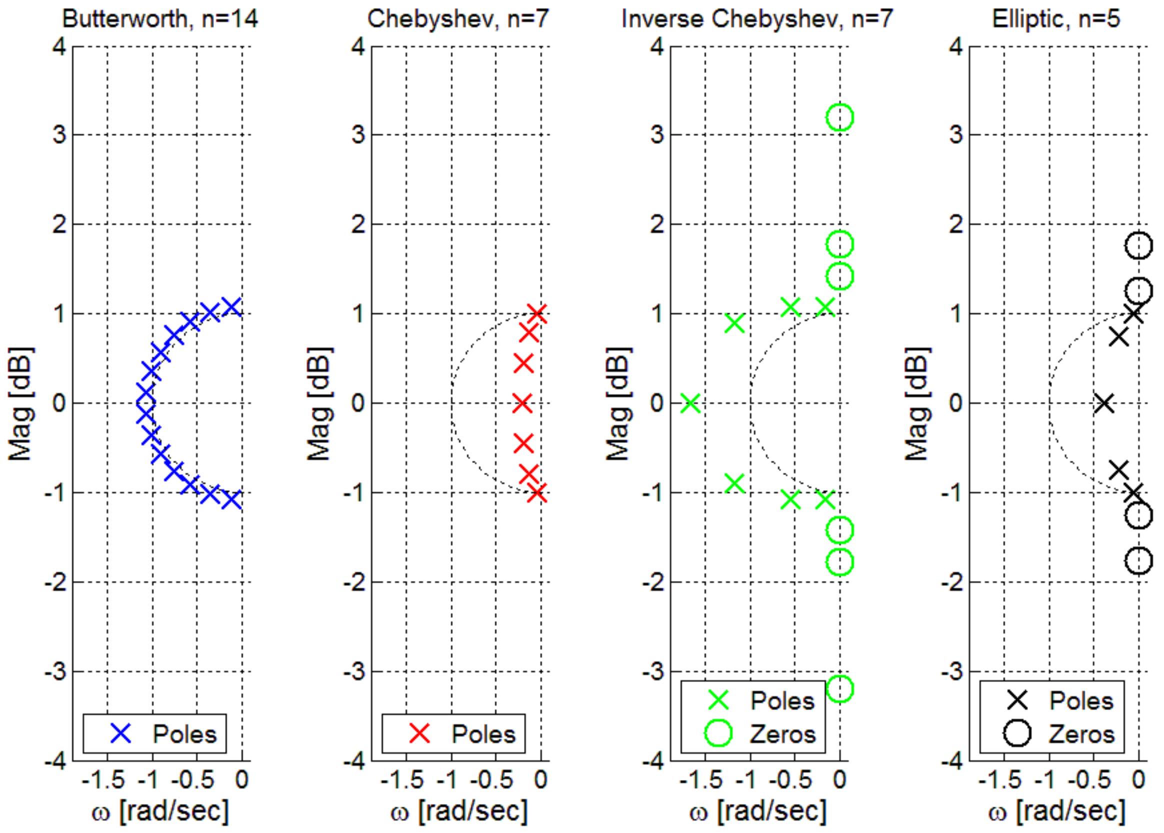


Filter approximation meeting the same specification yield

Order (Butterworth)> Order (Chebyshev)> Order (Elliptic)

Design Example

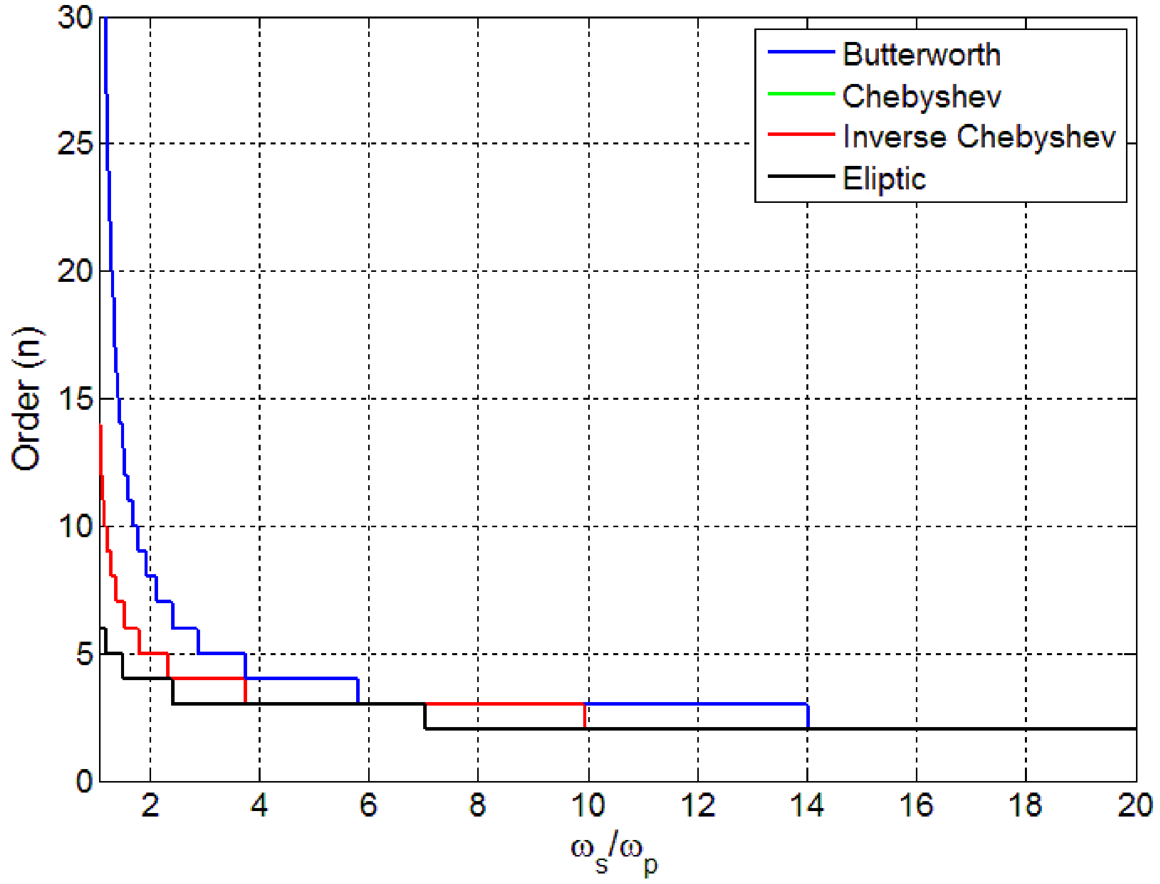
Pole/zero locations



Note that Chebyshev and Eliptic approximations needs ***high-Q*** poles

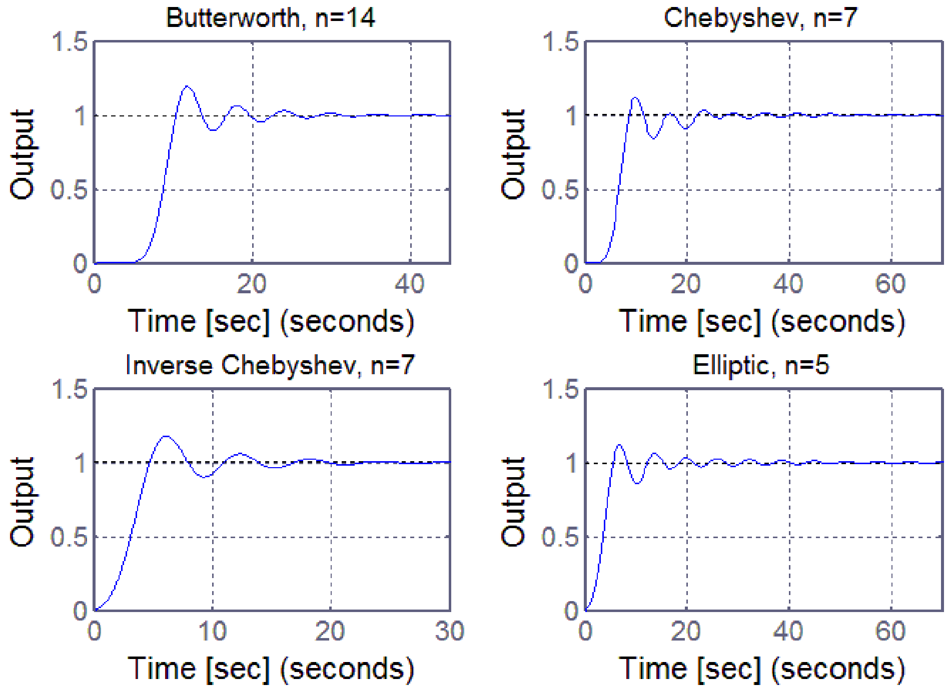
Design example

Filter order for $R_p = 1$ and $R_s = 40$



Eliptic filter always yields the least order. *Is it always the best choice ?*

Step response of the design example



- Inverse Chebyshev filter has the least overshoot and ringing
- Ringing and overshoots can be problematic in some applications
- The pulse deformation is due to the fact that the filter introduces different time delay to the different frequency components (Phase distortion)

Phase Distortion

- Consider a filter with a transfer function

$$H(j\omega) = |H(j\omega)|e^{j\phi(\omega)}$$

- Let us apply two sine waves at different frequencies

$$v_{in}(t) = A_1 \sin(\omega_1 t) + A_2 \sin(\omega_2 t)$$

- The filter output is

$$v_{out}(t) = A_1|H(j\omega_1)| \sin\left(\omega_1\left(t + \frac{\phi_1}{\omega_1}\right)\right) + A_2|H(j\omega_2)| \sin\left(\omega_2\left(t + \frac{\phi_2}{\omega_2}\right)\right)$$

- Assuming that the difference between $|H(j\omega_1)|$ and $|H(j\omega_2)|$ is small, the shape of the time domain output signal will be preserved if the two signals are delays by the same amount of time

$$\frac{\phi(\omega_1)}{\omega_1} = \frac{\phi(\omega_2)}{\omega_2}$$

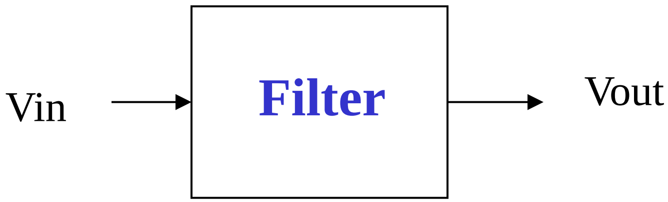
- This condition is satisfied for

$$\phi(\omega) = t_0\omega \qquad t_0 = constant$$

- A filter with this characteristic is called “linear phase”

Linear Phase Filters

- For this type of filters: The magnitude of the signals is scaled equally, and they are delayed by the same amount of time



$$v_{out}(t) = Kv_{in}(t - t_0)$$

$$V_{out}(s) = V_{in}(s)\left[Ke^{-j\omega(t_0)}\right]$$

- The filter transfer function is

$$H(s) = Ke^{-j\omega t_0}$$

$$|H(j\omega)| = K \qquad \phi(j\omega) = t_0\omega$$

- In this types of filters the phase delay $\tau_{PD} = -\frac{\phi(j\omega)}{\omega}$, and the group delay $\tau_{GD} = -\frac{d\phi(j\omega)}{\omega}$ are constant and equal

Linear Phase Filter Approximation

- For a typical lowpass filter

$$H(s) = \frac{K}{1 + a_1 s + a_2 s^2 + \cdots} = \frac{K}{[1 - a_2\omega^2 + \cdots] + j\omega[a_1 - a_3\omega^3 + \cdots]}$$

Thus the phase shift is given by

$$\phi(\omega) = \arg\big(H(j\omega)\big) = -\tan^{-1}\left(\frac{a_1 - a_3\omega^3 + \cdots}{1 - a_2\omega^2 + \cdots}\right)$$

- Using power-series expansion $\tan^{-1}(x) = x - \frac{x^3}{3} + \frac{x^5}{5} - \cdots$

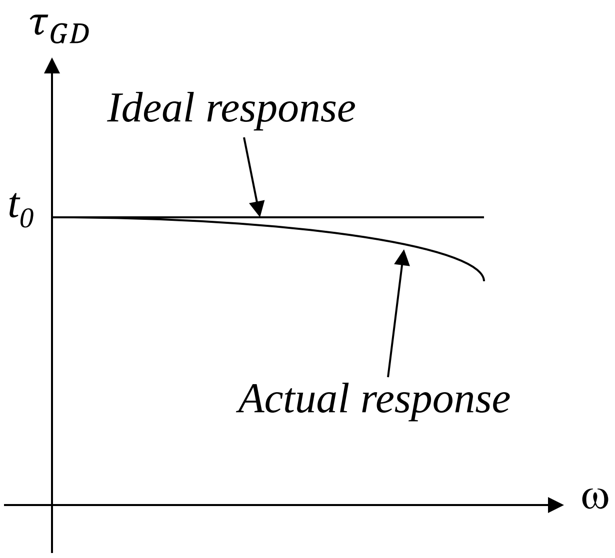
The condition for linear phase is satisfied if

$$\frac{\partial}{\partial\omega}\tan^{-1}(x) = \frac{\partial}{\partial\omega}\left(x - \frac{x^3}{3} + \frac{x^5}{5} - \cdots\right) = constant$$

These are called Bessel polynomials, and the resulting networks are called Thomson filters

Linear Phase Filter Approximation

- Typically the phase behavior of these filters shows some deviation

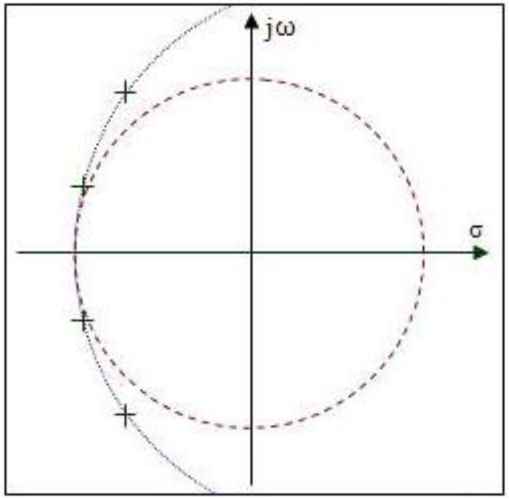


- Errors are measured in time
- τ_{GD} is the Group Delay

$$\tau_{GD} = -\frac{\partial \phi}{\partial \omega}$$

Bessel (Thomson) Filter Approximation

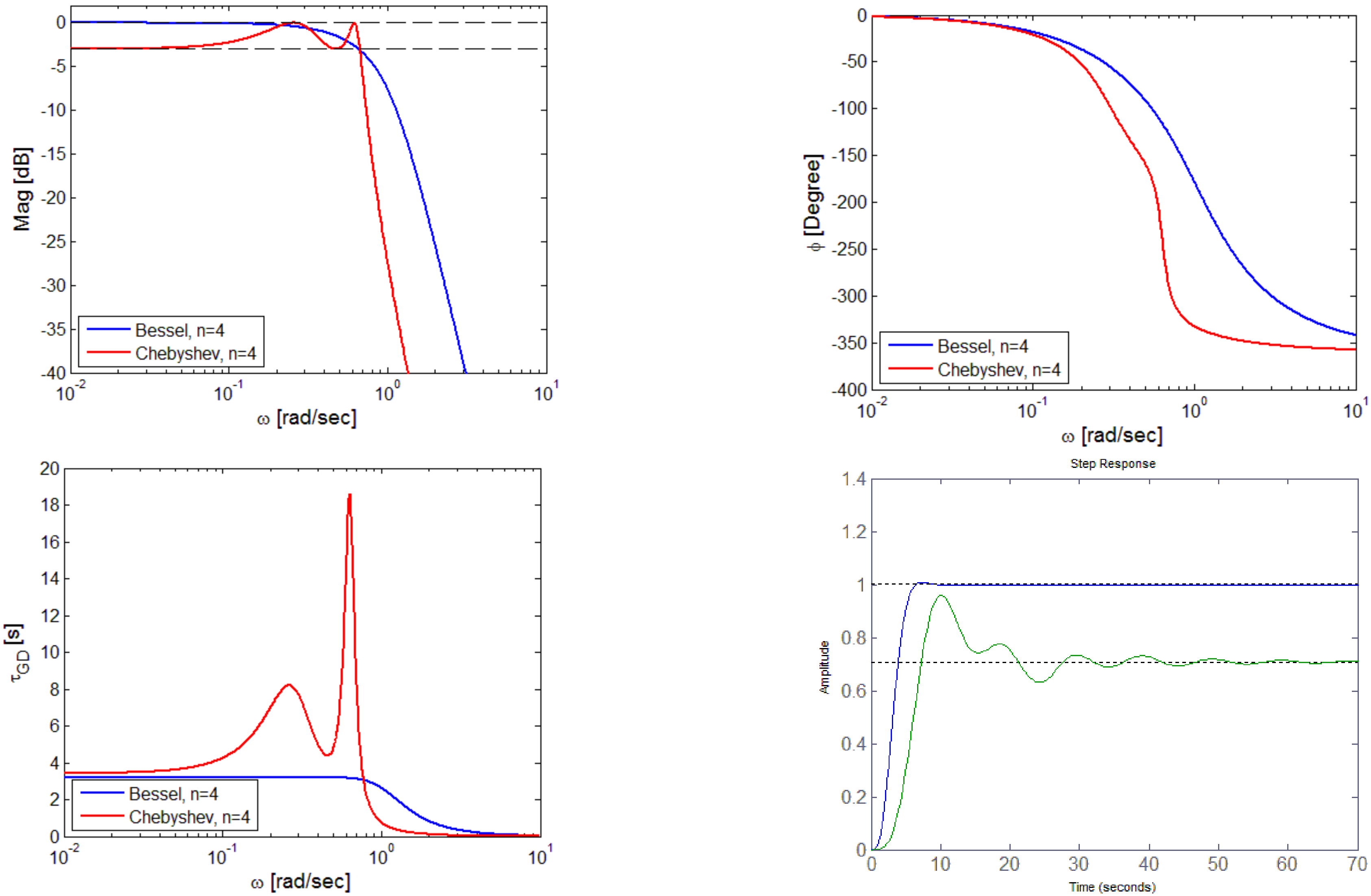
- All poles
- Poles are relatively low Q
- Maximally flat group delay (Maximally linear phase response)
- Poor stopband attenuation



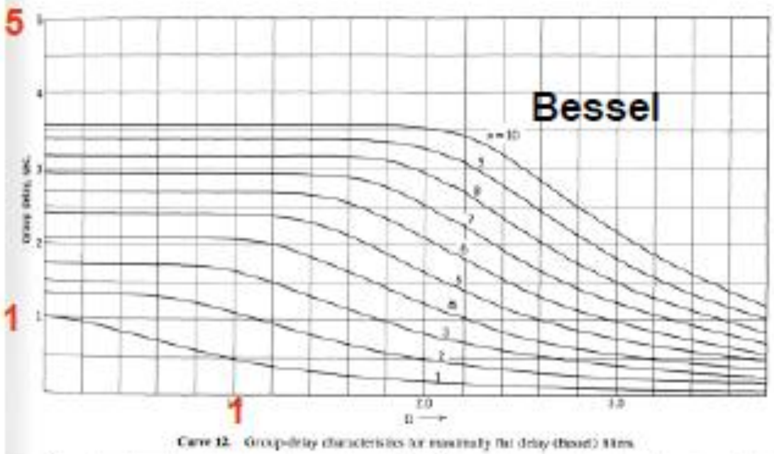
Order (N)	Re Part (-σ)	Im Part (±jω)
1	1.0000	
2	1.1030	0.6368
3	1.0509 1.3270	1.0025
4	1.3596 0.9877	0.4071 1.2476
5	1.3851 0.9606 1.5069	0.7201 1.4756

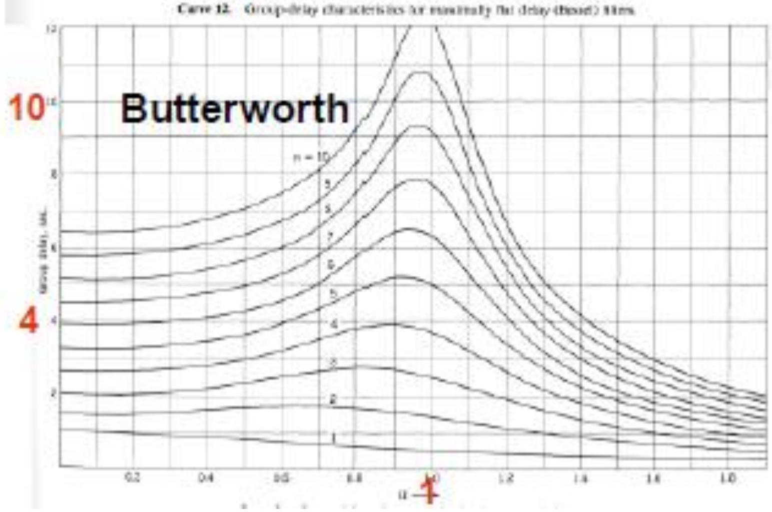
http://www.rfcafe.com/references/electrical/bessel-poles.htm

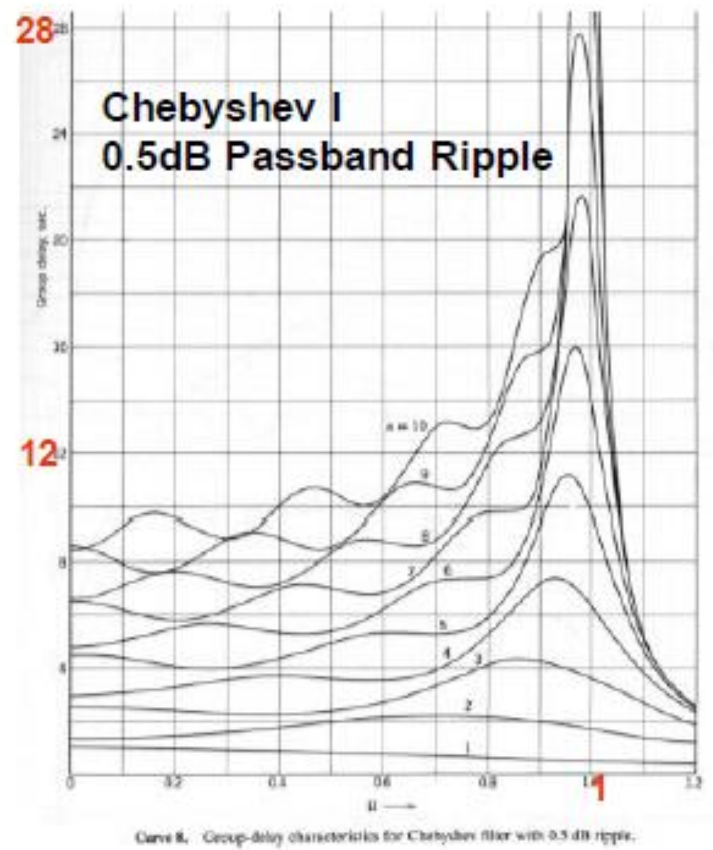
Bessel Filter Approximation



Comparison of various LPF Group delay







Ref: A. Zverev, Handbook of filter synthesis, Wiley, 1967.

Filter Design Conventional Procedure

- **Transform your filter specs into a normalized LPF**

- **Filter order, zeros, poles and/or values for the passive elements can be obtained from tables or from a software package like FIESTA or Matlab**

- **If you use biquadratic sections, you need poles and zeros matching**

- **For ladder filters, the networks can be obtained from tables**

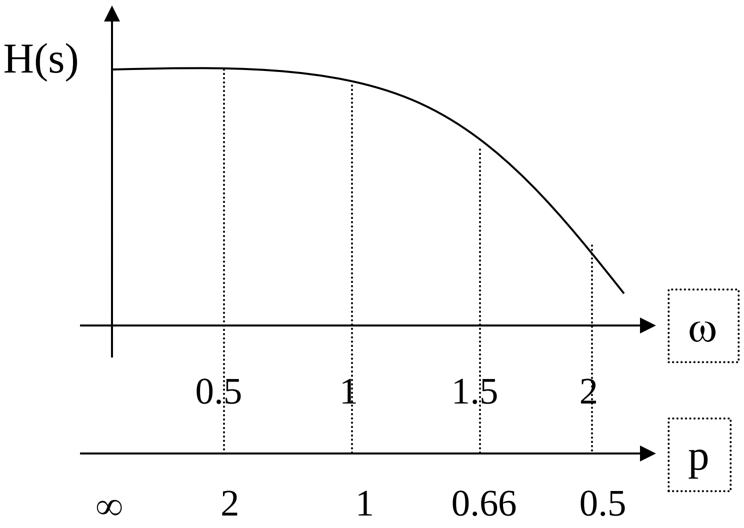
- **Transform the normalized transfer function to your filter by using**
  - **Filter transformation (LP to BP, HP, BR)**
  - **Frequency transformation**
  - **Impedance denormalization**

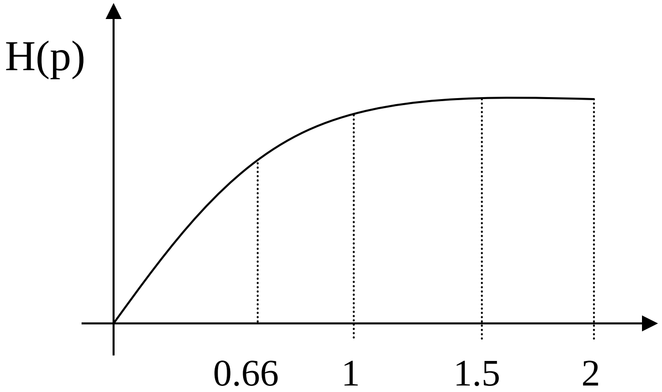
- **You obtain the transfer function or your passive network**

Frequency Transformations

➤Lowpass to Highpass

$$s \Rightarrow \frac{1}{p} \quad \text{then} \quad H_{lp}(s) = \frac{H_0}{\sum_{i=0}^{n} a_i s^i} \quad \Rightarrow \quad H_{hp}(p) = \frac{H_0}{\sum_{i=0}^{n} a_i \left(\frac{1}{p}\right)^i} = \frac{H_0 p^n}{\sum_{i=0}^{n} a_i p^{n-i}}$$





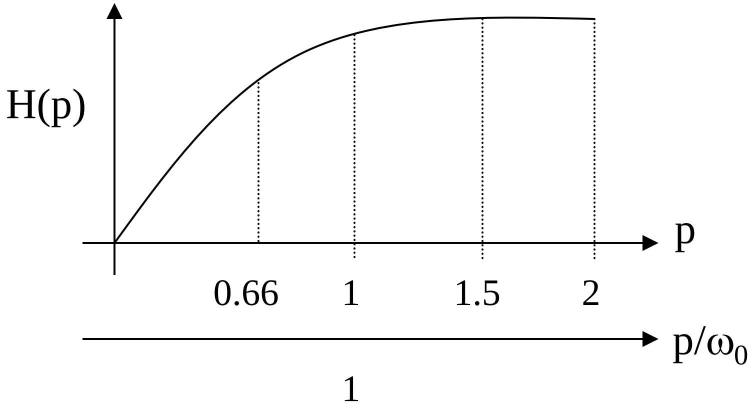
✓ Lowpass to Highpass

$$H_{lp}(s) = \frac{H_0}{\sum_{i=0}^{n} a_i s^i} \Leftrightarrow H_{hp}(p) = \frac{H_0 p^n}{\sum_{i=0}^{n} a_i p^{n-i}}$$

✓N zeros at ∞ are translated to zero
✓Poles are not the same!!!
✓The main characteristics of the lowpass filter are maintained
✓for a highpass filter with cutoff frequency at ω_{0}, then

$$s \Rightarrow \frac{\omega_0}{p}$$

This transformation scheme translates ω=1 to $p=\omega_0$



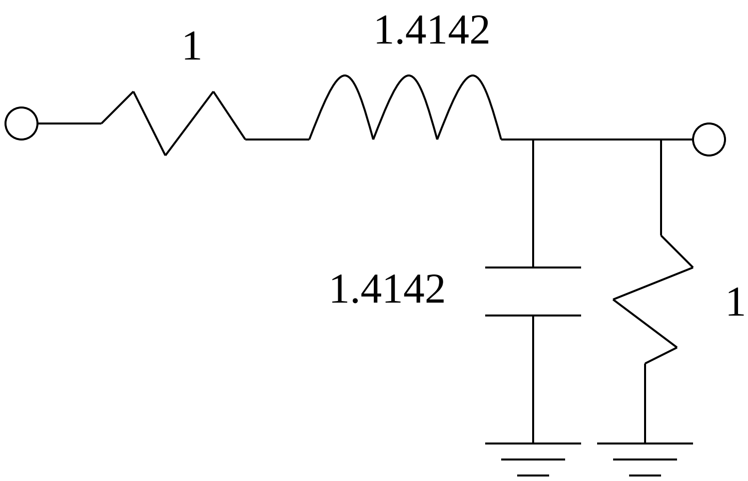
Frequency Transformations

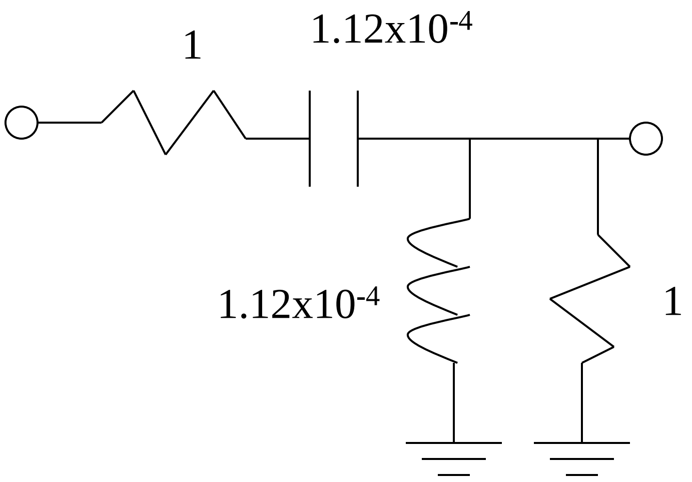
- The Lowpass to Highpass transformation can also be applied to the elements

Element	impedance	transformed
Resistor	R	R
inductor	sL	$\frac{\omega_0 L}{p}$
capacitor	$\frac{1}{sC}$	$\frac{p}{\omega_0 C}$

- The resistors are not affected
- L is transformed in a capacitor $C_{eq}=1/\omega_0 L$
- C is transformed in an inductor $L_{eq}=1/\omega_0 C$

Example: Design a 1KHz HP-filter from a LP prototype.





Frequency Transformations

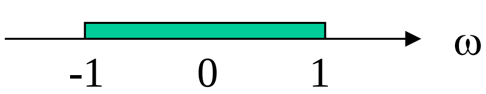
➢Lowpass to Bandpass transformation

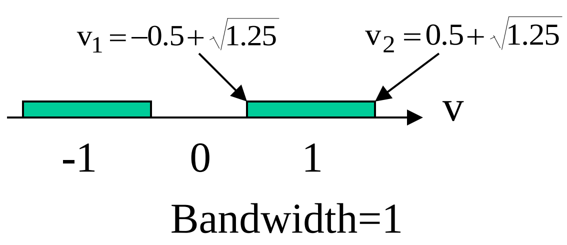
$$s \Rightarrow \frac{p^2+1}{p} \qquad \text{then} \quad H_{lp}(s) = \frac{H_0}{\sum_{i=0}^{n} a_i s^i} \quad \Rightarrow \quad H_{bp}(p) = \frac{H_0 p^n}{\sum_{i=0}^{2n} b_i (p)^i}$$

- n zeros at ω=0 and n zeros at ∞
- even number of poles
- The bandwidth of the BP is equal to the bandwidth of the LP

- In the p-domain

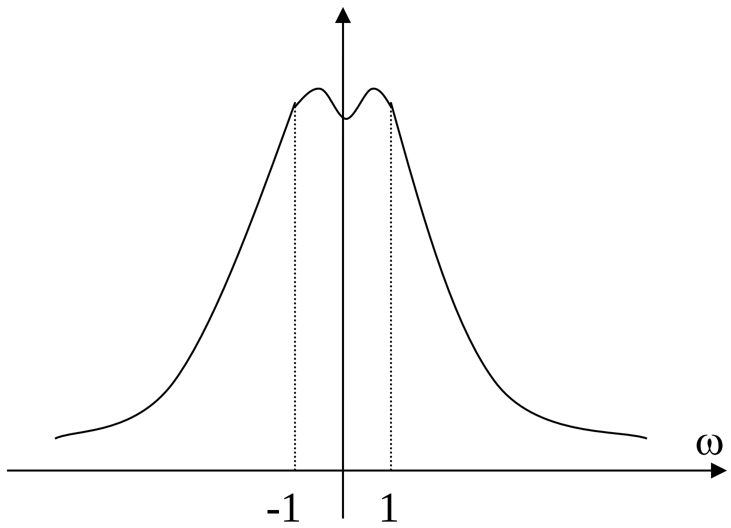
$$p = jv = \frac{s}{2} \pm \sqrt{\left(\frac{s}{2}\right)^2 - 1} \quad \text{or} \quad v = \frac{\omega}{2} \pm \sqrt{\left(\frac{\omega}{2}\right)^2 + 1}$$



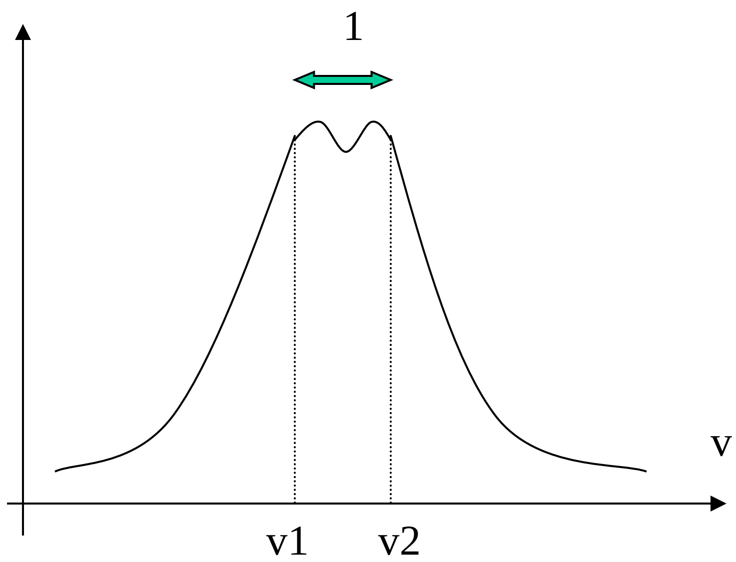


Frequency Transformations

- Note that $v_2 - v_1 = 1$ and $v_2 \cdot v_1 = 1$



Lowpass prototype



Bandpass filter

Frequency Transformations

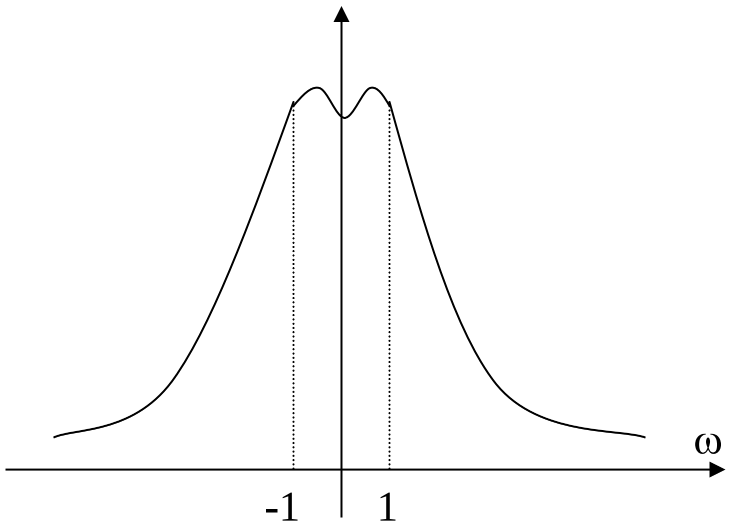
- General transformation $s = \frac{1}{BW}\left(\frac{p^2 + \omega_0{}^2}{p}\right)$

$$v = \frac{BW\cdot\omega}{2} \pm \sqrt{\left(\frac{BW\cdot\omega}{2}\right)^2 + \omega_0{}^2}$$

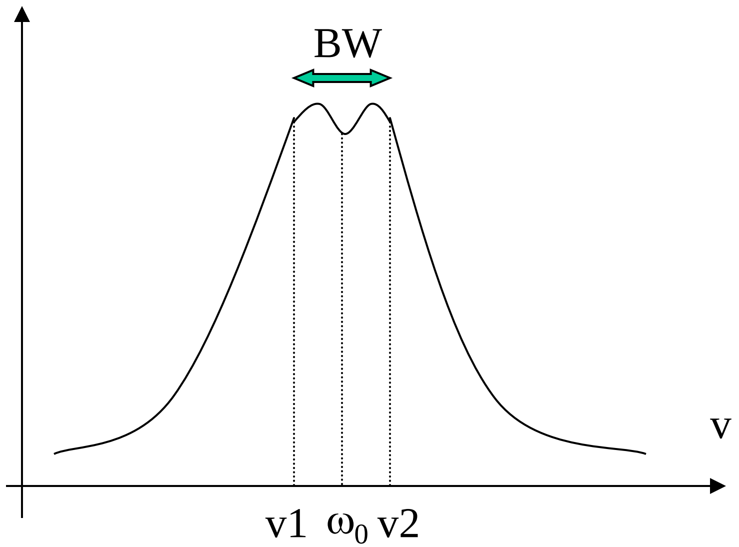
$$-1 \Rightarrow v_1 = -\frac{BW}{2} + \sqrt{\left(\frac{BW}{2}\right)^2 + \omega_0{}^2}$$

$$0 \Rightarrow \omega_0$$

$$1 \Rightarrow v_2 = +\frac{BW}{2} + \sqrt{\left(\frac{BW}{2}\right)^2 + \omega_0{}^2}$$



Lowpass prototype

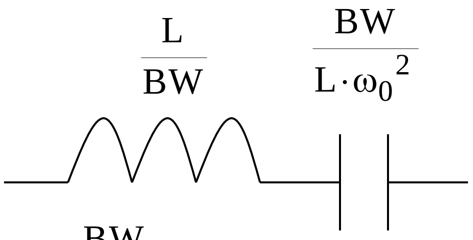


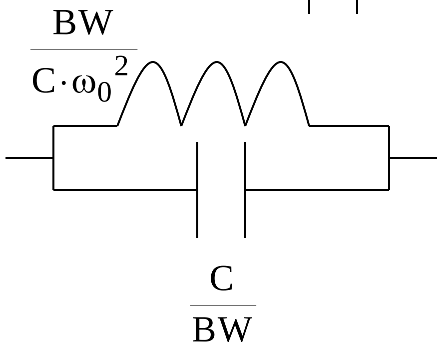
Bandpass filter

Frequency Transformations

- The Lowpass to Bandpass transformation can also be applied to the elements

Element	impedance	transformed
Resistor	R	R
inductor	sL	$\frac{L}{BW}p+\frac{\omega_0^2 L}{BW}\frac{1}{p}$
capacitor	$\frac{1}{sC}$	$\frac{1}{\frac{C}{BW}p+\frac{\omega_0^2 C}{BW}\frac{1}{p}}$

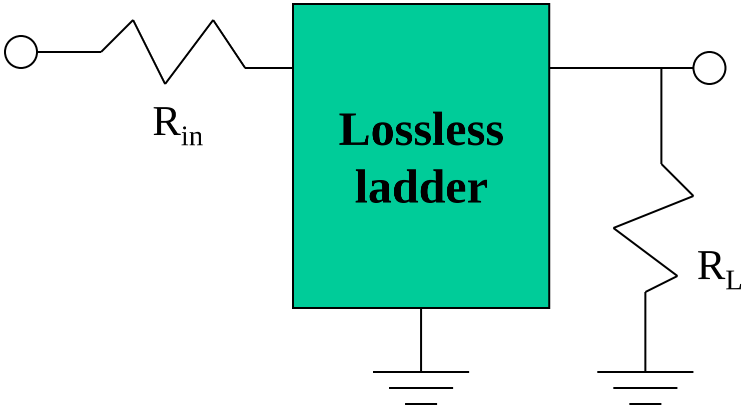




- **Note that for $\omega=\omega_0$**
- **for the inductor Zeq=0**
- **for the capacitor Yeq=0 (Zeq=∞)**

Frequency Transformations

- **In general, for double-resistance terminated ladder filters**



➢ **around $\omega=\omega_0$**

$$H(s)\Big|_{\omega=\omega_0} = \frac{R_L}{R_{in}+R_L} = \frac{1}{1+\dfrac{R_L}{R_{in}}}$$

➢ **In the passband, the transfer function can be very well controlled**

➢**Low-sensitivy**

Frequency Transformations

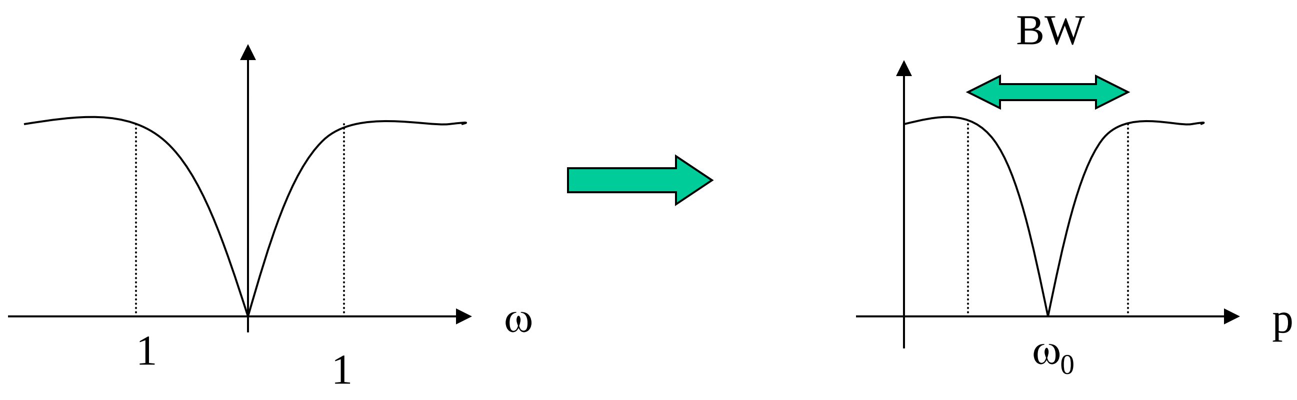
- Lowpass to Bandreject transformation

$$s \Rightarrow \frac{1}{\frac{1}{BW}\left(\frac{p^2+\omega_0{}^2}{p}\right)}$$

- Lowpass to Highpass transformation (notch at $\omega=0$)

- Shifting the frequency to ω_0 and adjusting the bandwidth to BW

. Bandpass transformation!!!



Transformation Methods

- Transformation methods have been developed where a low pass filter can be converted to another type of filter by simply transforming the complex variable s.

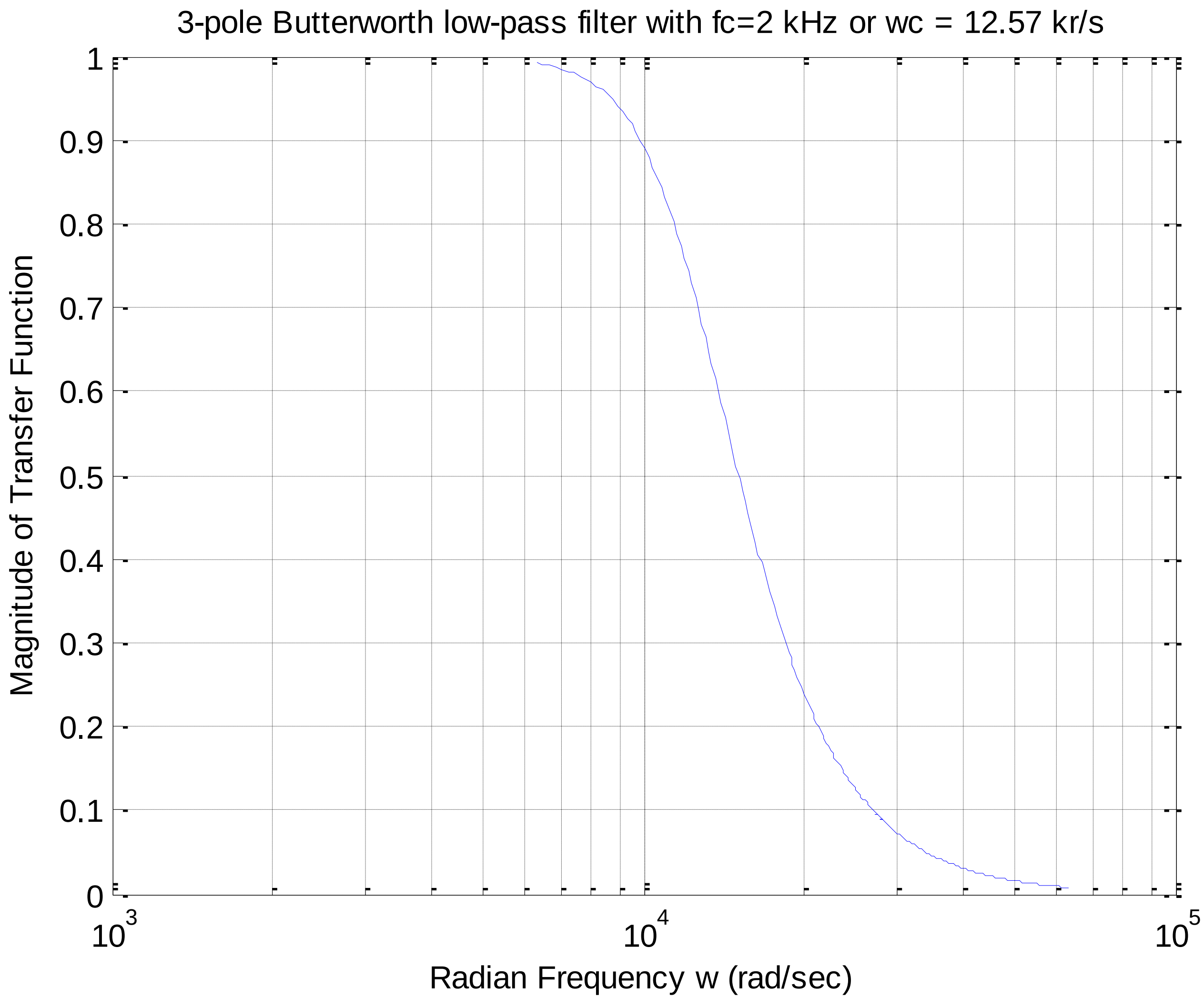
- Matlab lp2lp, lp2hp, lp2bp, and lp2bs functions can be used to transform a low pass filter with normalized cutoff frequency, to another low-pass filter with any other specified frequency, or to a high pass filter, or to a band-pass filter, or to a band elimination filter, respectively.

LPF with normalized cutoff frequency, to another LPF with any other specified frequency

- Use the MATLAB **buttap** and **lp2lp** functions to find the transfer function of a third-order Butterworth low-pass filter with cutoff frequency fc=2kHz.

```
% Design 3 pole Butterworth low-pass filter (wcn=1 rad/s)
[z,p,k]=buttap(3);
[b,a]=zp2tf(z,p,k);          % Compute num, den coefficients of this filter
(wcn=1rad/s)
f=1000:1500/50:10000;   % Define frequency range to plot
w=2*pi*f;                   % Convert to rads/sec
fc=2000;                    % Define actual cutoff frequency at 2 KHz
wc=2*pi*fc;                 % Convert desired cutoff frequency to rads/sec
[bn,an]=lp2lp(b,a,wc);    % Compute num, den of filter with fc = 2 kHz
Gsn=freqs(bn,an,w);       % Compute transfer function of filter with fc = 2 kHz
semilogx(w,abs(Gsn));
grid;
xlabel('Radian Frequency w (rad/sec)')
ylabel('Magnitude of Transfer Function')
title('3-pole Butterworth low-pass filter with fc=2 kHz or wc = 12.57 kr/s')
```

LPF with normalized cutoff frequency, to another LPF with any other specified frequency

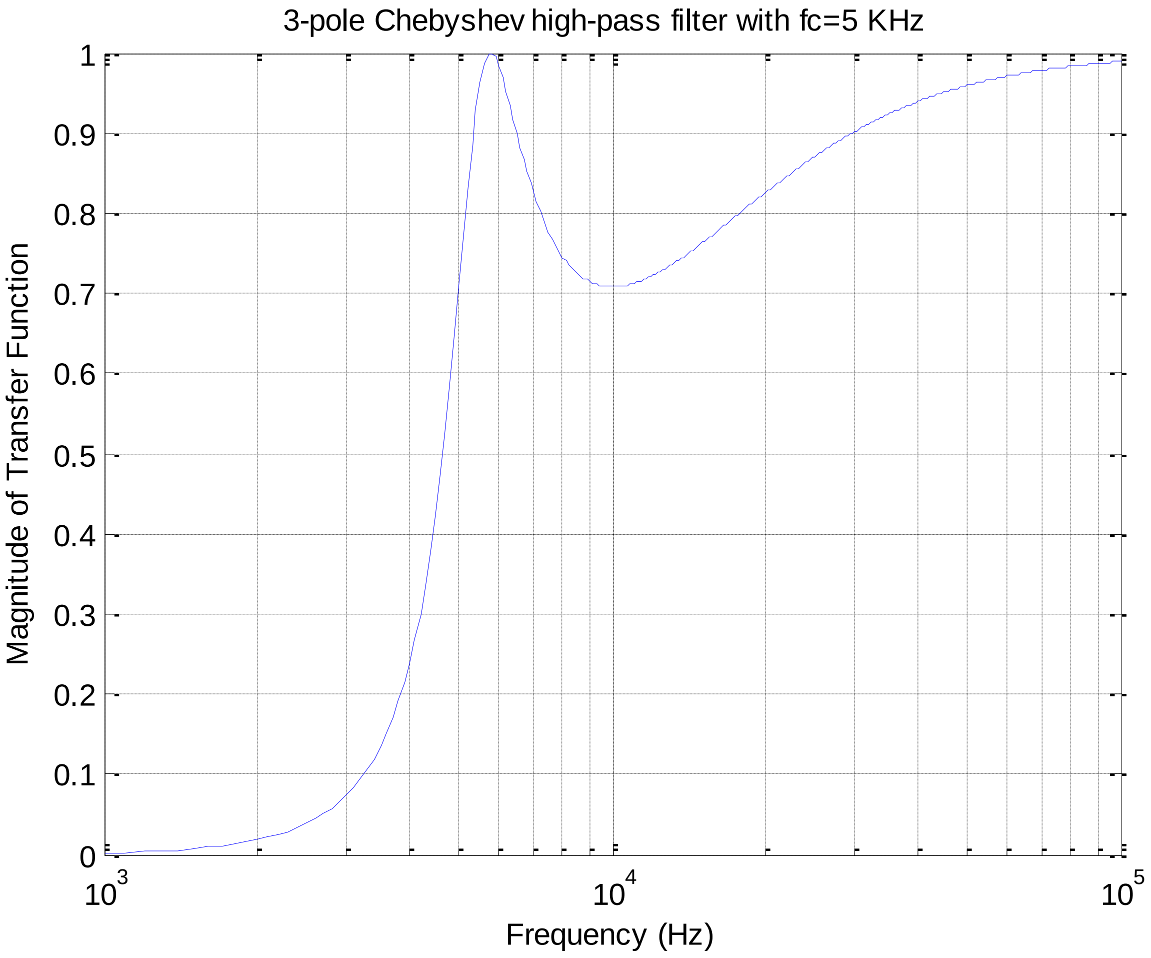


High-Pass Filter

- Use the MATLAB commands **cheb1ap** and **lp2hp** to find the transfer function of a 3-pole Chebyshev high-pass analog filter with cutoff frequency fc = 5KHz.

```
% Design 3 pole Type 1 Chebyshev low-pass filter, wcn=1 rad/s
[z,p,k]=cheb1ap(3,3);
[b,a]=zp2tf(z,p,k);              % Compute num, den coef. with wcn=1 rad/s
f=1000:100:100000;          % Define frequency range to plot
fc=5000;                              % Define actual cutoff frequency at 5 KHz
wc=2*pi*fc;                          % Convert desired cutoff frequency to rads/sec
[bn,an]=lp2hp(b,a,wc);       % Compute num, den of high-pass filter with fc =5KHz
Gsn=freqs(bn,an,2*pi*f);   % Compute and plot transfer function of filter with fc = 5 KHz
semilogx(f,abs(Gsn));
grid;
xlabel('Frequency (Hz)');
ylabel('Magnitude of Transfer Function')
title('3-pole Type 1 Chebyshev high-pass filter with fc=5 KHz ')
```

High-Pass Filter

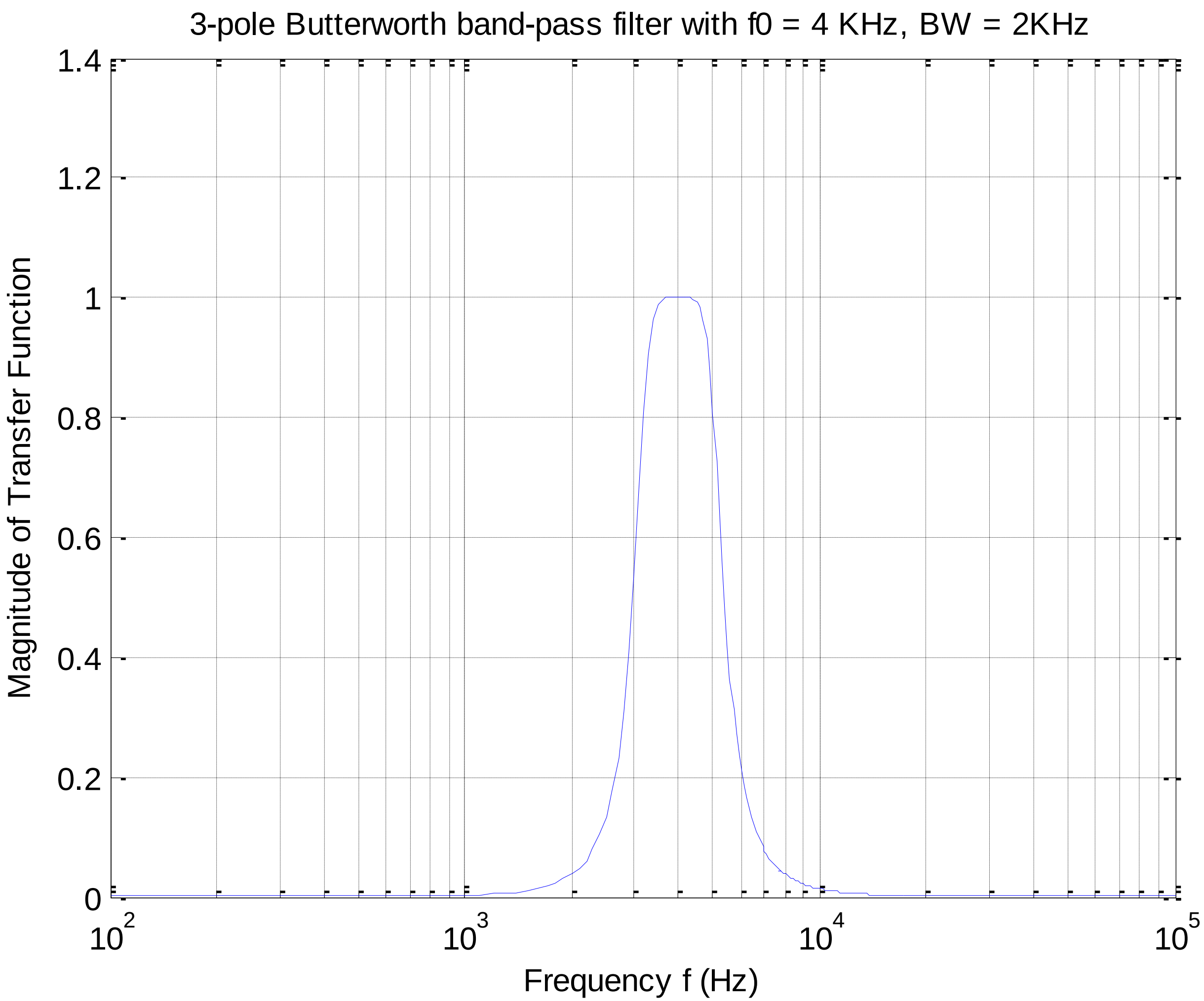


Band-Pass Filter

- Use the MATLAB functions **buttap** and **lp2bp** to find the transfer function of a 3-pole Butterworth analog band-pass filter with the pass band frequency centered at fo = 4kHz , and bandwidth BW =2KHz.

```
[z,p,k]=buttap(3);      % Design 3 pole Butterworth low-pass filter with wcn=1 rad/s
[b,a]=zp2tf(z,p,k);     % Compute numerator and denominator coefficients for wcn=1 rad/s
f=100:100:100000;       % Define frequency range to plot
f0=4000;                % Define centered frequency at 4 KHz
W0=2*pi*f0;             % Convert desired centered frequency to rads/s
fbw=2000;               % Define bandwidth
Bw=2*pi*fbw;            % Convert desired bandwidth to rads/s
[bn,an]=lp2bp(b,a,W0,Bw);       % Compute num, den of band-pass filter
% Compute and plot the magnitude of the transfer function of the band-pass filter
Gsn=freqs(bn,an,2*pi*f);
semilogx(f,abs(Gsn));
grid;
xlabel('Frequency f (Hz)');
ylabel('Magnitude of Transfer Function');
title('3-pole Butterworth band-pass filter with f0 = 4 KHz, BW = 2KHz')
```

Band-Pass Filter

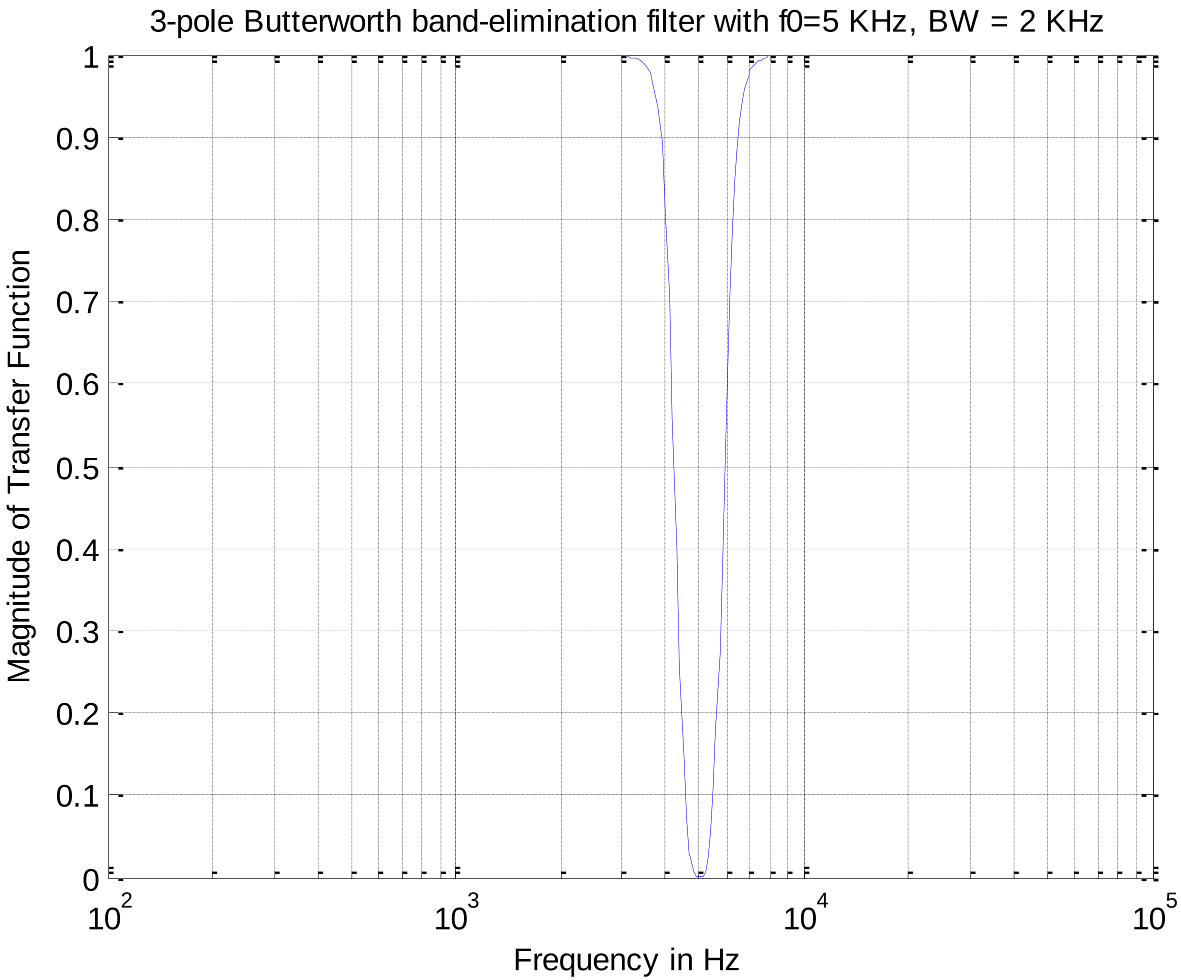


Band-Elimination (band-stop) Filter

- Use the MATLAB functions **buttap** and **lp2bs** to find the transfer function of a 3-pole Butterworth band-elimination (band-stop) filter with the stop band frequency centered at fo = 5 kHz , and bandwidth BW = 2kHz.

```
[z,p,k]=buttap(3);              % Design 3-pole Butterworth low-pass filter, wcn = 1 r/s
[b,a]=zp2tf(z,p,k);             % Compute num, den coefficients of this filter, wcn=1 r/s
f=100:100:100000;               % Define frequency range to plot
f0=5000;                        % Define centered frequency at 5 kHz
W0=2*pi*f0;                     % Convert centered frequency to r/s
fbw=2000;                       % Define bandwidth
Bw=2*pi*fbw;                    % Convert bandwidth to r/s
% Compute numerator and denominator coefficients of desired band stop filter
[bn,an]=lp2bs(b,a,W0,Bw);
% Compute and plot magnitude of the transfer function of the band stop filter
Gsn=freqs(bn,an,2*pi*f);
semilogx(f,abs(Gsn));
grid;
xlabel('Frequency in Hz'); ylabel('Magnitude of Transfer Function');
title('3-pole Butterworth band-elimination filter with f0=5 KHz, BW = 2 KHz')
```

Band-Elimination (band-stop) Filter



How to find the minimum order to meet the filter specifications ?

The following functions in Matlab can help you to find the minimum order required to meet the filter specifications:

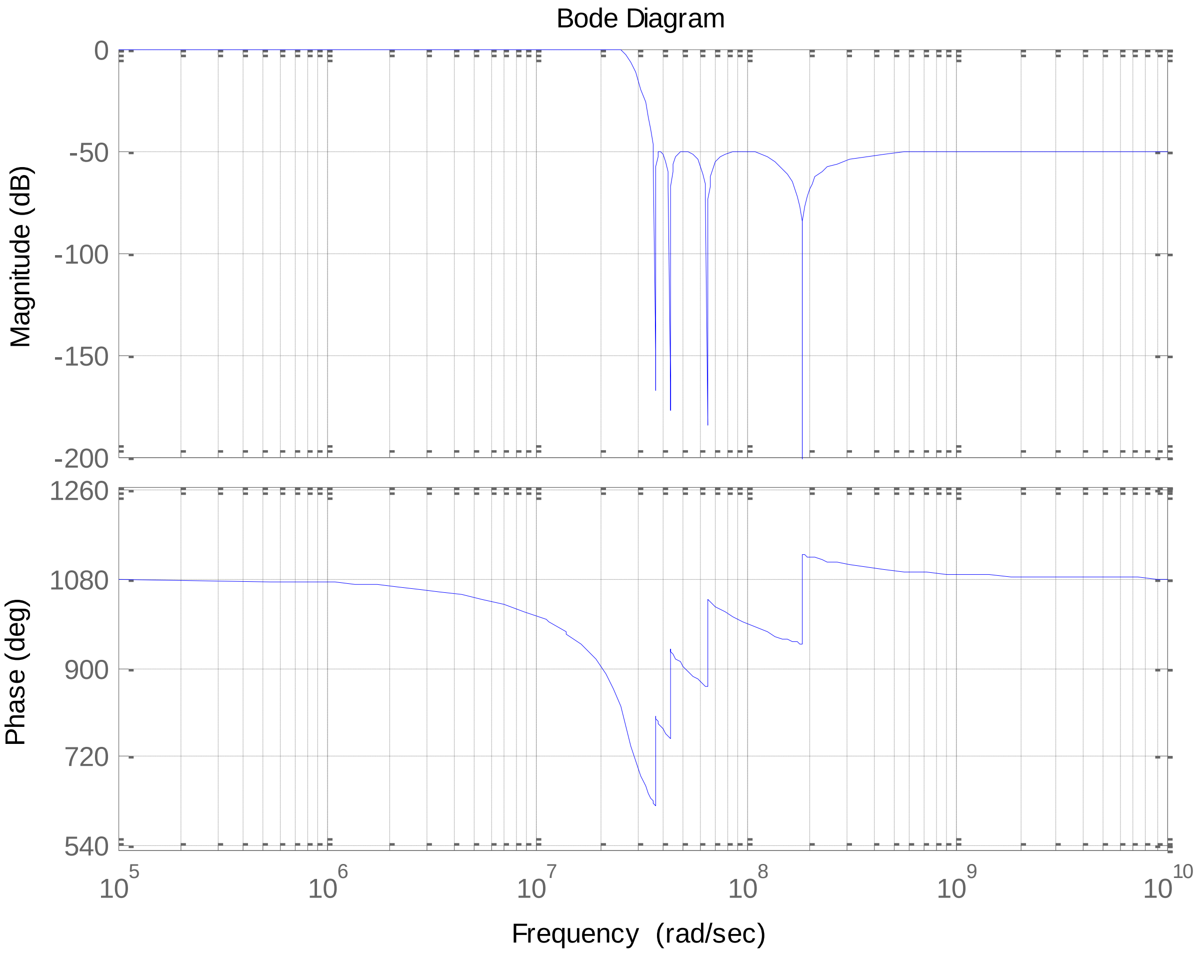
- Buttord for butterworth
- Cheb1ord for chebyshev
- Ellipord for elliptic
- Cheb2ord for inverse chebyshev

Calculating the order and cutoff frequency of a inverse chebyshev filter

- Design a 4MHz Inverse Chebyshev approximation with Ap gain at passband corner. The stop band is 5.75MHz with -50dB gain at stop band.

```
clear all;
Fp = 4e6; Wp=2*pi*Fp;
Fs=1.4375*Fp; Ws=2*pi*Fs;
Fplot = 20*Fs;
f = 1e6:Fplot/2e3:Fplot ;
w = 2*pi*f;
Ap = 1;
As = 50;
% Cheb2ord helps you find the order and wn (n and Wn) that
%you can pass to cheby2 command.
[n, Wn] =   cheb2ord(Wp, Ws, Ap, As, 's');
[z, p, k]  = cheby2(n, As, Wn, 'low', 's');
[num, den] = cheby2(n, As, Wn, 'low', 's');
bode(num, den)
```

Bode Plot


Bode Diagram
Magnitude (dB)
0
-50
-100
-150
-200
Phase (deg)
1260
1080
900
720
540
10^5
10^6
10^7
10^8
10^9
10^{10}
Frequency (rad/sec)

References

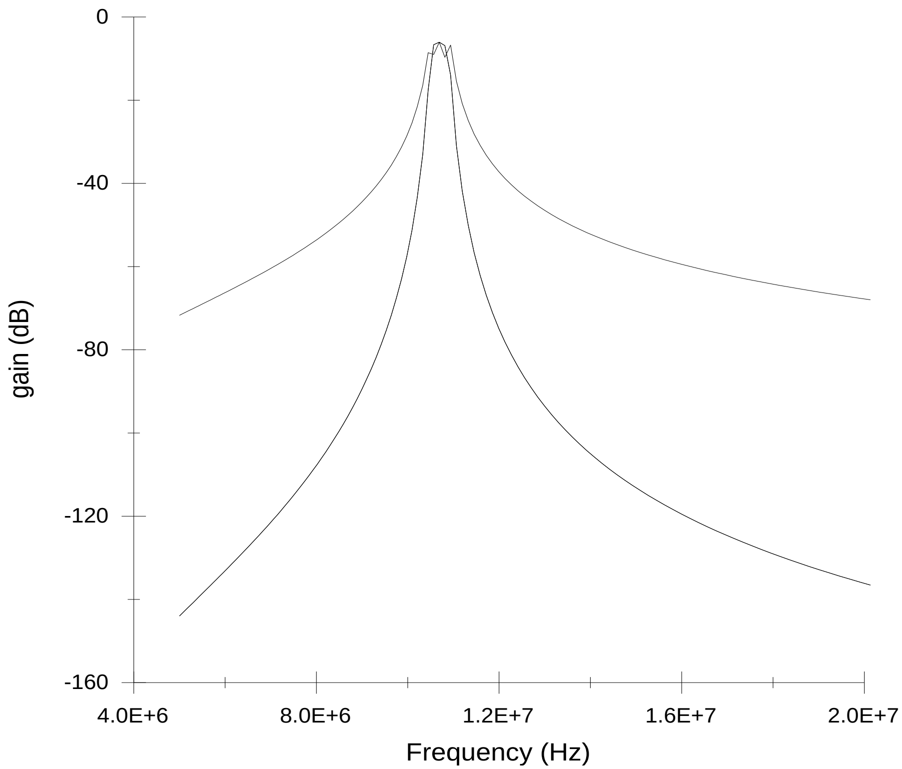
[1] S. T. Karris, “Signals and Systems with Matlab Computing and Simulink Modeling,” Fifth Edition. Orchard Publications

[2] Matlab Help Files

Ladder Filters

- The ladder filter realization can be found in tables and/or can be obtained from FIESTA

- The elements must be transformed according to the frequency and impedance normalizations



Sensitivity

- Definition $S_x^y = \frac{x}{y}\frac{\partial y}{\partial x}$ Y= transfer function and x = variable or element

Some properties:

$$S_x^{ky} = S_{kx}^{y} = S_x^y \qquad S_{kx}^{x} = S_x^x = 1$$

$$S_x^{1/y} = S_{1/x}^{y} = -S_x^y \qquad S_x^{y^n} = nS_x^y$$

$$S_{x^n}^{y} = \frac{1}{n}S_x^y \qquad S_x^y = S_{x_2}^{y} S_x^{x_2}$$

$$S_x^{\prod_{i=1}^{n} y_i} = \sum_{i=1}^{n} S_x^{y_i} \qquad S_x^{\sum_{i=1}^{n} y_i} = \frac{\sum_{i=1}^{n} y_i S_x^{y_i}}{\sum_{i=1}^{n} y_i}$$

For a typical H(s)

$$S_{a_j}^{H(s)=\frac{H_0}{\sum a_i (j\omega)^i}} = -\frac{\sum a_i (j\omega)^i S_{a_j}^{a_i (j\omega)^i}}{\sum a_i (j\omega)^i} = -\frac{a_j (j\omega)^j}{\sum a_i (j\omega)^i}$$

Sensitivity

- Sensitivity is a **measure of the change in the performance of the system** due to a **change in the nominal value of a certain element.**

$$S_x^y = \frac{x}{y}\frac{\partial y}{\partial x} \quad \Rightarrow \quad S_x^y \cong \frac{x}{y}\frac{\Delta y}{\Delta x} \quad \text{or} \quad \frac{\Delta y}{y} = \left[S_x^y\right]\frac{\Delta x}{x}$$

⇒ Normalized variations at the output are determined by the sensitivity function and the normalized variations of the parameter

<u>Example:</u>

If the senstivity function is 10, then variations of $\Delta x/x=0.01(1\%)$ produce $\Delta y/y=0.1(10\%)$

For a good design, the sensitivity functions should be < 5.
Effects of the partial positive feedback (negative resistors)?

Sensitivity

- For a typical amplifier $$A_v = \frac{g_m}{g_0}$$

$$S_{g_m}^{A_v} = 1, \quad S_{g_0}^{A_v} = -1$$

•Sometimes the dc gain is enhanced by using a negative resistor

$$A_v = \frac{g_m}{g_0 - g_{02}} = \frac{g_m}{g_0} \frac{1}{1 - \frac{g_{02}}{g_0}}$$

For large dc gain $g_{02} = g_0$

$$S_{g_m}^{A_v} = 1, \quad S_{g_0}^{A_v} = -\frac{g_0 S_{g_0}^{g_0}}{g_0 - g_{02}} = -\frac{1}{1 - \frac{g_{02}}{g_0}}$$

The larger the gain improvment the larger the sensitivity!!!!

Properties of Stable Network Functions

$$N(s)=\frac{A(s)}{B(s)}=K\frac{1+b_1 s+b_2 s^2+..}{1+a_1 s+a_2 s^2+..}$$

- Typically the transfer function presents the form of a ratio of two polynomials
- For **non-negative** elements the **coefficients are real and positive**
- The poles are located in the left side of the s-plane

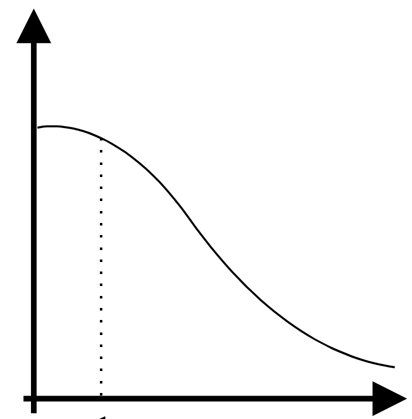
$$H(s)=\frac{1}{s+a}$$

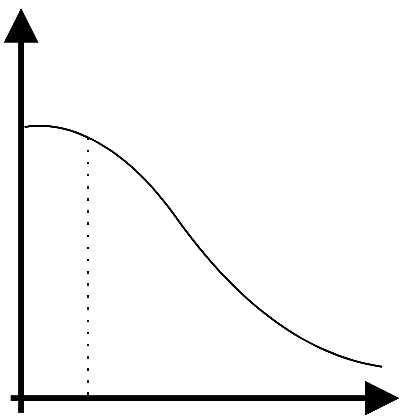
$$h(t)=e^{-at} \qquad t>0$$

- **System is stable**
- **BOUNDED OUTPUT FOR BOUNDED INPUT**

Properties of Network Functions:

Frequency Transformation





For inductors and capacitors:

$$jp(L) \Rightarrow j\omega\left(\frac{L}{\Omega_n}\right)$$

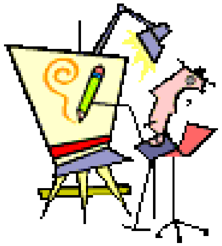
$$jp(C) \Rightarrow j\omega\left(\frac{C}{\Omega_n}\right)$$

- Most of the filter approximations are normalized to 1 rad/sec. Hence, it is necessary to denormalize the transfer function.
- **Using the frequency transformation**

$$\omega = \frac{p}{\Omega_n}$$

-
- 1 rad/sec is translated to
- $1/\Omega_n$ rad/sec

Impedance denormalization



- Typically the network elements are normalized to 1 Ω. Hence an impedance denormalization scheme must be used

$$Z \Rightarrow \Omega_n Z$$

- or

$$R \Rightarrow \Omega_n R$$

$$L \Rightarrow \Omega_n L$$

$$C \Rightarrow C/\Omega_n$$

- Note that the transfer function is invariant with the impedance denormalization (RC and LC products remain constant!!!!)

- In general both frequency and impedance denormalizations are used

Conclusions

• There is a number of conventional filter magnitude approximations

• The choice of a particular approximation is application dependent

• Besides the magnitude specifications, there exists also a phase (group delay) specification. For this the Thompson (Bessel) approximation is used

• There are a host of Filter approximation software programs, including Matlab, Filsyn, and Fiesta2 developed at TAMU

Acknowledgment: Thanks to my colleague Dr. Silva-Martínez for providing some of the material for this presentation