2.161 Signal Processing: Continuous and Discrete
Fall 2008

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
DEPARTMENT OF MECHANICAL ENGINEERING
2.161 Signal Processing – Continuous and Discrete

Introduction to Continuous Time Filter Design [1]

1 Introduction

The design techniques described here are based on the creation of a prototype low-pass filter, with subsequent transformation to other filter forms (high-pass, band-pass, band-stop) if necessary.

2 Low-Pass Filter Design

The prototype low-pass filter is based upon the magnitude-squared of the frequency response function $|H(j\Omega)|^2$, or the frequency response power function. The phase response of the filter is not considered. We begin by defining tolerance regions on the power frequency response plot, as shown in Fig. 1.

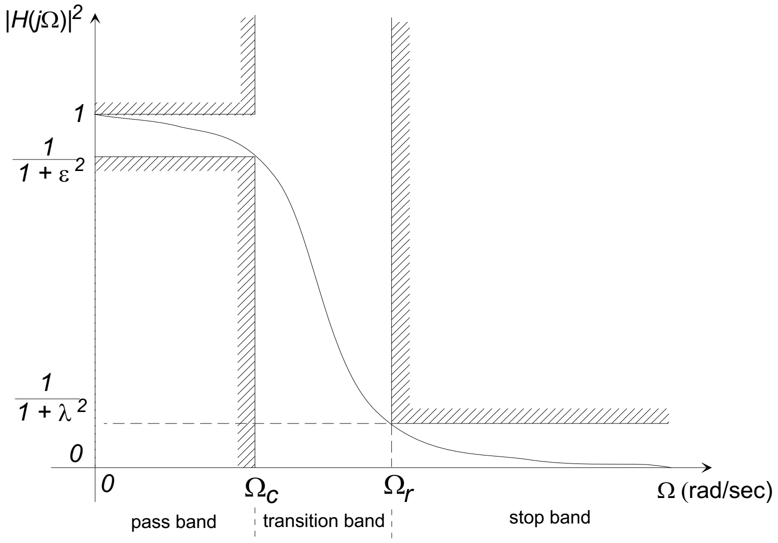


Figure 1: Tolerance regions in the frequency response plot.

The filter specifications are that

$$1 \geq |H(j\Omega)|^2 > \frac{1}{1+\epsilon^2} \qquad \text{for } |\Omega| < \Omega_c \tag{1}$$

[1] D. Rowell, October 2, 2008

$$\text{and} \quad |H(j\Omega)|^2 < \frac{1}{1+\lambda^2} \qquad \text{for } |\Omega| > \Omega_r, \tag{2}$$

where Ω_c is the cut-off frequency, Ω_r is the rejection frequency, and ϵ and λ are design parameters that select the filter attenuation at the two critical frequencies. For example, if $\epsilon = 1$, at Ω_c the power response response $|H(j\Omega_c)|^2 = 0.5$, the -3 dB response frequency. In general we expect the response function to be monotonically decreasing in the transition band.

The filter functions examined in this document will be of the form

$$|H(j\Omega)|^2 = \frac{1}{1+f^2(\Omega)}. \tag{3}$$

where $f(\Omega) \to 0$ as $\Omega \to 0$, and $f(\Omega) \to \infty$ as $\Omega \to \infty$ to generate a low-pass filter action.

2.1 The Butterworth Filter

The Butterworth filter is defined by the power gain

$$|H(j\Omega)|^2 = \frac{1}{1+\epsilon^2 \left(\Omega/\Omega_c\right)^{2N}} \tag{4}$$

where N is a positive integer defining the filter order. Note that λ does not appear in this formulation, but clearly N and λ are interrelated, since at $\Omega = \Omega_r$

$$\frac{1}{1+\lambda^2} \geq \frac{1}{1+\epsilon^2 \left(\Omega_r/\Omega_c\right)^{2N}} \tag{5}$$

which may be solved to show

$$N \geq \frac{\log(\lambda/\epsilon)}{\log(\Omega_r/\Omega_c)} \tag{6}$$

The power response function of Butterworth filters for $N = 1 \ldots 5$ is shown in Fig. 2. Butterworth filters are also known as "maximally flat" filters because the response has the maximum number of vanishing derivatives at $\Omega = 0$ and $\Omega = \infty$ for filters of the form of Eq. 3.

2.1.1 The Transfer Function of the Butterworth Filter

The poles of the power gain transfer function may be found from the characteristic equation

$$1 + \epsilon^2 \left(\frac{s}{j\Omega_c}\right)^{2N} = 0 \tag{7}$$

which yields $2N$ roots (poles) that lie on a circle:

$$p_n = \Omega_c \epsilon^{-1/N} e^{j\pi(2n+N-1)/2N} \qquad n = 1 \ldots 2N \tag{8}$$

with radius $r = \Omega_c \epsilon^{-1/N}$, and angular separation of π/N rad. Notice that if N is odd a pair of poles will lie on the real axis at $s = \pm\Omega_c \epsilon^{-1/N}$, while if N is even the roots will form complex conjugate pairs.

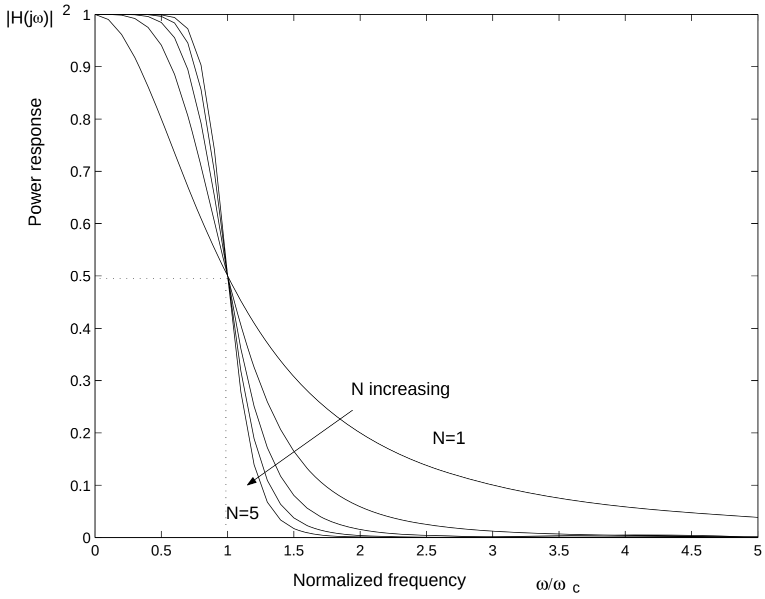


Figure 2: Power response of Butterworth filters for $N = 1 \dots 5$ ($\epsilon = 1$).

Figure 3a shows the six poles of $|H(s)|^2$ for a third-order ($N = 3$) Butterworth filter. For a causal system we must have

$$|H(j\Omega)|^2 = H(j\Omega)\overline{H(j\Omega)} = H(s)H(-s)|_{s=j\Omega}$$

which allows us to take take the N poles of $|H(j\Omega)|^2$ in the left half-plane as the poles of the filter $H(s)$, that is the poles specified by ($n = 1 \dots N$) in Eq. (8) above,

$$p_n = \Omega_c \epsilon^{-1/N} e^{j\pi(2n+N-1)/2N} \qquad n = 1 \dots N \tag{9}$$

as is shown in Fig. 3b.

If the filter is to have unity gain at low frequencies

$$\lim_{\Omega \to 0} |H(j\Omega)| = 1$$

we require the complete Butterworth transfer function to be

$$\begin{aligned} H(s) &= \frac{(-p_1)(-p_2)\dots(-p_N)}{(s-p_1)(s-p_2)\dots(s-p_N)} \\ &= \frac{(-1)^N p_1 p_2 \dots p_N}{(s-p_1)(s-p_2)\dots(s-p_N)}, \end{aligned}$$

where only the N stable left-half-plane poles are included.

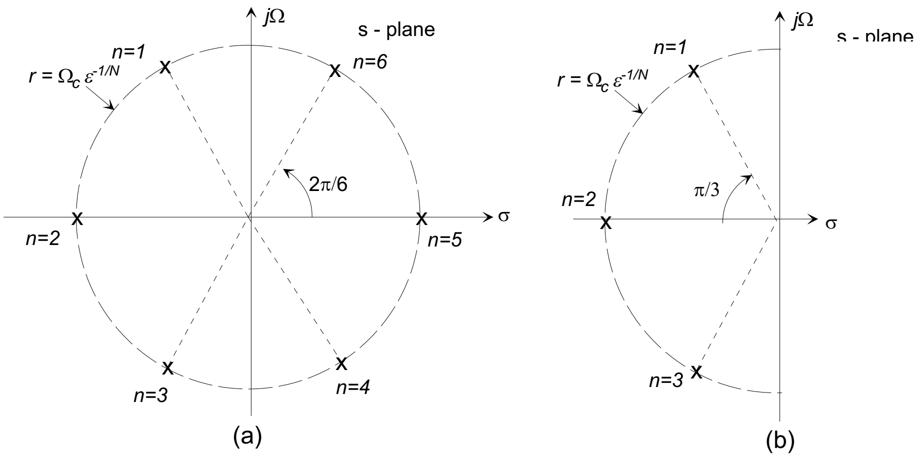


Figure 3: The poles of (a) $|H(s)|^2$, and (b) $H(s)$ for a third-order ($N = 3$) Butterworth filter.

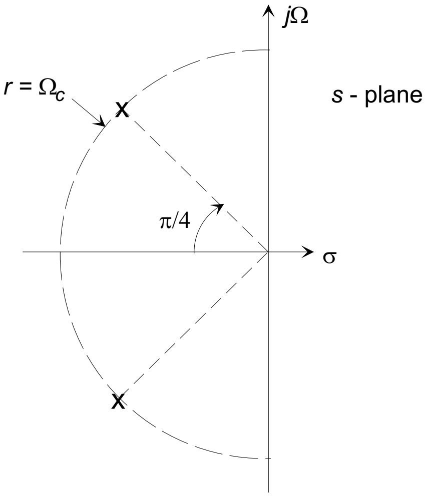


Figure 4: Poles of a second-order Butterworth filter ($N = 2, \epsilon = 1$).

■ Example

Consider a second-order ($N = 2$, $\epsilon = 1$) Butterworth filter with cut-off frequency Ω_c. Equation (9) generates a pole pair as shown shown in the pole-zero of Fig. 4. The transfer function is that of a second-order system with damping ratio $\zeta = 0.707$ and undamped natural frequency $\Omega_n = \Omega_c$. Show that the power frequency response satisfies the Butterworth specification of Eq. (4).

The transfer function for the second-order filter is

$$H(s) = \frac{\Omega_c^2}{s^2 + \sqrt{2}\Omega_c s + \Omega_c^2} \tag{10}$$

and the frequency response is

$$H(j\Omega) = H(s)|_{s=j\Omega} = \frac{\Omega_c^2}{(\Omega_c^2 - \Omega^2) + j\sqrt{2}\Omega_c\Omega} \tag{11}$$

$$|H(j\Omega)|^2 = H(j\Omega)\overline{H(j\Omega)} = \frac{\Omega_c^4}{(\Omega_c^4 + \Omega^4)} = \frac{1}{1 + (\Omega/\Omega_c)^4} \tag{12}$$

which is of the Butterworth form of Eq. (4).

■ Example

Design a Butterworth low-pass filter to meet the power gain specifications shown in Fig. 5. Comparing Figs. 1 and 5

$$\frac{1}{1+\epsilon^2} = 0.9 \quad \longrightarrow \quad \epsilon = 0.3333$$
$$\frac{1}{1+\lambda^2} = 0.05 \quad \longrightarrow \quad \lambda = 4.358$$

Then

$$N \geq \frac{\log(\lambda/\epsilon)}{\log(\Omega_r/\Omega_c)} = 3.70$$

we therefore select N=4. The 4 poles ($p_1 \ldots p_4$) lie on a circle of radius $r = \Omega_c \epsilon^{-1/N} = 13.16$ and are given by

$$\begin{aligned} |p_n| &= 13.16 \\ \angle p_n &= \pi(2n+3)/8 \end{aligned}$$

for $n = 1 \ldots 4$, giving a pair of complex conjugate pole pairs

$$\begin{aligned} p_{1,4} &= -5.04 \pm j12.16 \\ p_{2,3} &= -12.16 \pm j5.04 \end{aligned}$$

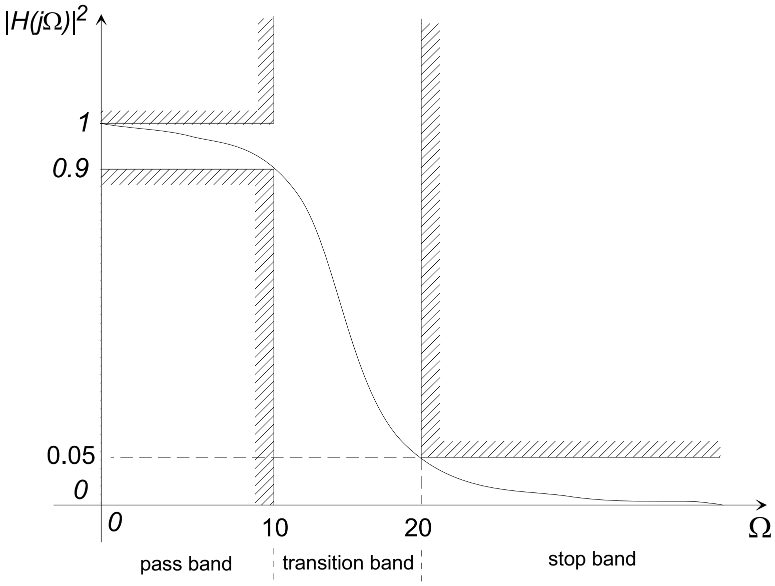


Figure 5: Filter specification for Butterworth design.

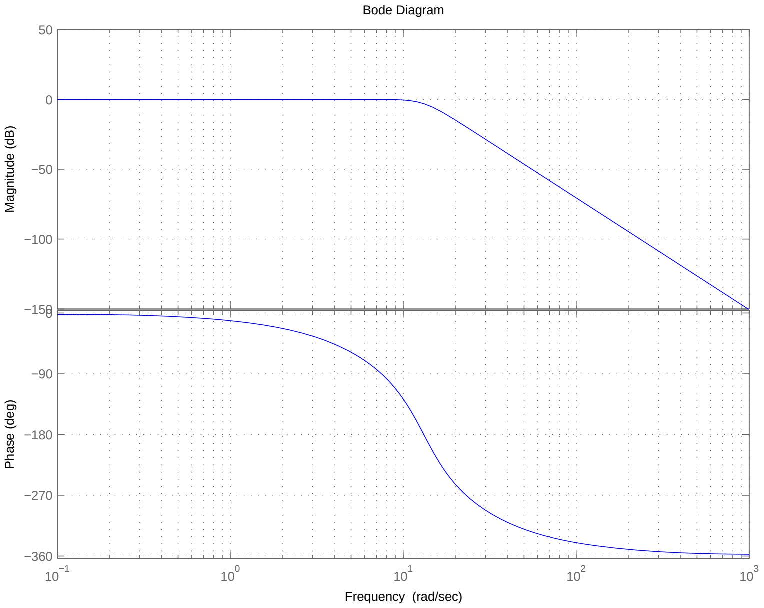


Figure 6: Bode plots for fourth-order Butterworth design example.

The transfer function, normalized to unity gain, is

$$H(s) = \frac{29993}{(s^2 + 10.07s + 173.2)(s^2 + 24.32s + 173.2)}$$

and the filter Bode plots are shown in Fig. 6.

2.2 Chebyshev Filters

The order of a filter required to met a low-pass specification may often be reduced by relaxing the requirement of a monotonically decreasing power gain with frequency, and allowing "ripple" to occur in either the pass-band or the stop-band. The Chebyshev filters allow these conditions:

$$\text{Type 1} \qquad |H(j\Omega)|^2 = \frac{1}{1 + \epsilon^2 T_N^2(\Omega/\Omega_c)} \tag{13}$$

$$\text{Type 2} \qquad |H(j\Omega)|^2 = \frac{1}{1 + \epsilon^2 \left(T_N^2(\Omega_r/\Omega_c)/T_N^2(\Omega_r/\Omega)\right)} \tag{14}$$

Where $T_N(x)$ is the Chebyshev polynomial of degree N. Note the similarity of the form of the Type 1 power gain (Eq. (13)) to that of the Butterworth filter, where the function $T_N(\Omega/\Omega_c)$ has replaced $(\Omega/\Omega_c)^N$. The Type 1 filter produces an all-pole design with slightly different pole placement from the Butterworth filters, allowing resonant peaks in the pass-band to introduce ripple, while the Type 2 filter introduces a set of zeros on the imaginary axis above Ω_r, causing a ripple in the stop-band.

The Chebyshev polynomials are defined recursively as follows

$$\begin{aligned}
T_0(x) &= 1 \\
T_1(x) &= x \\
T_2(x) &= 2x^2 - 1 \\
T_3(x) &= 4x^3 - 3x \\
&\vdots \\
T_N(x) &= 2xT_{N-1}(x) - T_{N-2}(x), \qquad N > 1
\end{aligned} \tag{15}$$

with alternate definitions

$$\begin{aligned}
T_N(x) &= \cos(N \cos^{-1}(x)) & (16) \\
&= \cosh(N \cosh^{-1}(x)) & (17)
\end{aligned}$$

The Chebyshev polynomials have the *min-max* property:

> *Of all polynomials of degree N with leading coefficient equal to one, the polynomial*
>
> $$T_N(x)/2^{N-1}$$
>
> *has the smallest magnitude in the interval $|x| \leq 1$. This "minimum maximum" amplitude is 2^{1-N}.*

In low-pass filters given by Eqs. (13) and (14), this property translates to the following characteristics:

Filter	Pass-Band Characteristic	Stop-Band Characteristic
Butterworth	Maximally flat	Maximally flat
Chebyshev Type 1	Ripple between 1 and $1/(1+\epsilon^2)$	Maximally flat
Chebyshev Type 2	Maximally flat	Ripple between 1 and $1/(1+\lambda^2)$

2.2.1 The Chebyshev Type 1 Filter

With the power response from Eq. (13)

$$|H(j\Omega)|^2 = \frac{1}{1+\epsilon^2 T_N^2(\Omega/\Omega_c)}$$

and the filter specification from Fig. 1, the required filter order may be found as follows. At the edge of the stop-band $\Omega = \Omega_r$

$$|H(j\Omega_r|^2 = \frac{1}{1+\epsilon^2 T_N^2(\Omega_r/\Omega_c)} \leq \frac{1}{1+\lambda^2}$$

so that

$$\lambda \leq \epsilon T_N(\Omega_r/\Omega_c) = \epsilon \cosh\left(N \cosh^{-1}\left(\Omega_r/\Omega_c\right)\right)$$

and solving for N

$$N \geq \frac{\cosh^{-1}\left(\lambda/\epsilon\right)}{\cosh^{-1}\left(\Omega_r/\Omega_c\right)} \tag{18}$$

The characteristic equation of the power transfer function is

$$1+\epsilon^2 T_N^2\left(\frac{s}{j\Omega_c}\right) = 0 \quad \text{or} \quad T_N\left(\frac{s}{j\Omega_c}\right) = \pm\frac{j}{\epsilon}$$

Now $T_N(x) = \cos(N\cos^{-1}(x))$, so that

$$\cos\left(N\cos^{-1}\left(\frac{s}{j\Omega_c}\right)\right) = \pm\frac{j}{\epsilon} \tag{19}$$

If we write $\cos^{-1}\left(s/j\Omega_c\right) = \gamma + j\alpha$, then

$$\begin{aligned} s &= \Omega_c\left(j\cos\left(\gamma+j\alpha\right)\right) \\ &= \Omega_c\left(\sinh\alpha\sin\gamma + j\cosh\alpha\cos\gamma\right) \end{aligned} \tag{20}$$

which defines an ellipse of width $2\Omega_c\sinh(\alpha)$ and height $2\Omega_c\cosh(\alpha)$ in the s-plane. The poles will lie on this ellipse. Substituting into Eq. (16)

$$\begin{aligned} T_N\left(\frac{s}{j\Omega_c}\right) &= \cos\left(N\left(\gamma+j\alpha\right)\right) \\ &= \cos N\gamma\cosh N\alpha - j\sin N\gamma\sinh N\alpha, \end{aligned}$$

the characteristic equation becomes

$$\cos N\gamma \cosh N\alpha - j \sin N\gamma \sinh N\alpha = \pm\frac{j}{\epsilon}. \tag{21}$$

Equating the real and imaginary parts in Eq. (21), (1) since $\cosh x \neq 0$ for real x we require $\cos N\gamma = 0$, or

$$\gamma_n = \frac{(2n-1)\pi}{2N} \qquad n = 1, \ldots, 2N \tag{22}$$

and, (2) since at these values of γ, $\sin N\gamma = \pm 1$ we have

$$\alpha = \pm\frac{1}{N}\sinh^{-1}\frac{1}{\epsilon} \tag{23}$$

As in the Butterworth design procedure, we select the left half-plane poles as the poles of the filter frequency response. The design procedure is:

1. Determine the filter order

$$N \geq \frac{\cosh^{-1}(\lambda/\epsilon)}{\cosh^{-1}(\Omega_r/\Omega_c)}$$

2. Determine α

$$\alpha = \pm\frac{1}{N}\sinh^{-1}\frac{1}{\epsilon}$$

3. Determine γ_n, $n = 1 \ldots N$

$$\gamma_n = \frac{(2n-1)\pi}{2N} \qquad n = 1, \ldots, N$$

4. Determine the N left half-plane poles

$$p_n = \Omega_c\left(\sinh\alpha \sin\gamma_n + j\cosh\alpha\cos\gamma_n\right) \qquad n = 1, \ldots, N$$

5. Form the transfer function

(a) If N is odd

$$H(s) = \frac{-p_1 p_2 \ldots p_N}{(s-p_1)(s-p_2)\ldots(s-p_N)}$$

(b) If N is even

$$H(s) = \frac{1}{1+\epsilon^2}\frac{p_1 p_2 \ldots p_N}{(s-p_1)(s-p_2)\ldots(s-p_N)}$$

The difference in the gain constants in the two cases arises because of the ripple in the pass-band. When N is odd, the response $|H(j0)|^2 = 1$, whereas if N is even the value of $|H(j0)|^2 = 1/(1+\epsilon^2)$.

■ Example

Repeat the previous Butterworth design example using a Chebyshev Type 1 design.
From the previous example we have $\Omega_c = 10$ rad/s., $\Omega_r = 20$ rad/s., $\epsilon = 0.3333$, $\lambda = 4.358$. The required order is

$$N \geq \frac{\cosh^{-1}(\lambda/\epsilon)}{\cosh^{-1}(\Omega_r/\Omega_c)} = \frac{\cosh^{-1} 13.07}{\cosh^{-1} 2} = 2.47$$

Therefore take $N = 3$. Determine α:

$$\alpha = \frac{1}{N}\sinh^{-1}\left(\frac{1}{\epsilon}\right) = \frac{1}{3}\sinh^{-1}(3) = 0.6061$$

and $\sinh\alpha = 0.6438$, and $\cosh\alpha = 1.189$. Also, $\gamma_n = (2n-1)\pi/6$ for $n = 1 \dots 6$ as follows:

n:	1	2	3	4	5	6
γ_n:	$\pi/6$	$\pi/2$	$5\pi/6$	$7\pi/6$	$3\pi/2$	$11\pi/6$
$\sin\gamma_n$:	$1/2$	1	$1/2$	$-1/2$	-1	$-1/2$
$\cos\gamma_n$:	$\sqrt{3}/2$	0	$-\sqrt{3}/2$	$-\sqrt{3}/2$	0	$\sqrt{3}/2$

Then the poles are

$$\begin{aligned}
p_n &= \Omega_c\left(\sinh\alpha\sin\gamma_n + j\cosh\alpha\cos\gamma_n\right)\\
p_1 &= 10\left(0.6438 \times \frac{1}{2} + j1.189 \times \frac{\sqrt{3}}{2}\right) = 3.219 + j10.30\\
p_2 &= 10\left(0.6438 \times 1 + j1.189 \times 0\right) = 6.438\\
p_3 &= 10\left(0.6438 \times \frac{1}{2} - j1.189 \times \frac{\sqrt{3}}{2}\right) = 3.219 - j10.30\\
p_4 &= 10\left(-0.6438 \times \frac{1}{2} - j1.189 \times \frac{\sqrt{3}}{2}\right) = -3.219 - j10.30\\
p_5 &= 10\left(-0.6438 \times 0 - j1.189 \times 0\right) = -6.438\\
p_6 &= 10\left(-0.6438 \times \frac{1}{2} + j1.189 \times \frac{\sqrt{3}}{2}\right) = -3.219 + j10.30
\end{aligned}$$

and the gain adjusted transfer function of the resulting Type 1 filter is

$$H(s) = \frac{750}{(s^2 + 6.438s + 116.5)(s + 6.438)}$$

The pole-zero plot for the Chebyshev Type 1 filter is shown in Fig. 7.

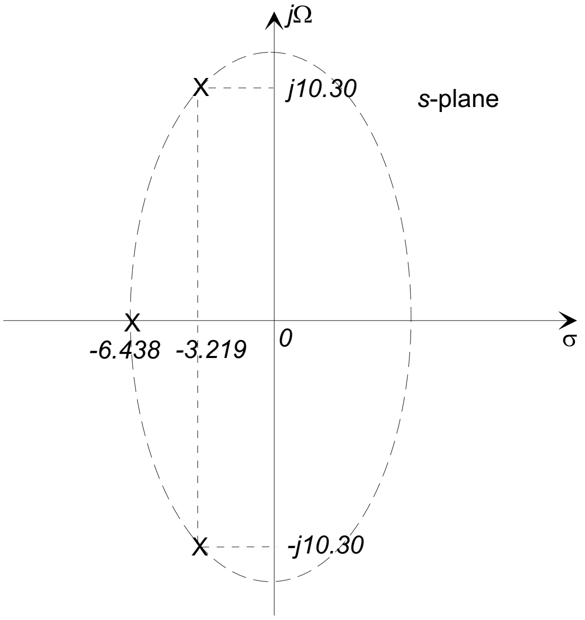


Figure 7: Pole-zero plot for third-order Chebyshev Type 1 design example.

2.2.2 The Chebyshev Type 2 Filter

The Chebyshev Type 2 filter has a monotonically decreasing magnitude function in the pass-band, but introduces equi-amplitude ripple in the stop-band by the inclusion of system zeros on the imaginary axis. The Type 2 filter is defined by the power gain function:

$$|H(j\Omega)|^2 = \frac{1}{1+\epsilon^2 \frac{T_N^2(\Omega_r/\Omega_c)}{T_N^2(\Omega_r/\Omega)}} \tag{24}$$

If we make the substitutions

$$\nu = \frac{\Omega_r \Omega_c}{\Omega} \quad \text{and} \quad \hat{\epsilon} = \frac{1}{\epsilon T_N(\Omega_r/\Omega_c)}$$

Eq. 24 may be written in terms of the modified frequency ν

$$|H(j\nu)|^2 = \frac{\hat{\epsilon}^2 T_N^2(\nu/\Omega_c)}{1+\hat{\epsilon}^2 T_N^2(\nu/\Omega_c)} \tag{25}$$

which has a denominator similar to the Type 1 filter, but has a numerator that contains a Chebyshev polynomial, and is of order $2N$. We can use a method similar to that used in the Type 1 filter design to find the poles as follows:

1. First define a complex variable, say $\tau = \mu + j\nu$ (analogous to the Laplace variable $s = \sigma + j\Omega$ used in the type 1 design) and write the power transfer function:

$$|H(\tau)|^2 = \frac{\hat{\epsilon}^2 T_N^2(\tau/j\Omega_c)}{1+\hat{\epsilon}^2 T_N^2(\tau/j\Omega_c)}$$

The poles are found using the method developed for the Type 1 filter, the zeros are found as the roots of the polynomial $T_N(\tau/j\Omega_c)$ on the imaginary axis $\tau = j\nu$. From the definition $T_N(x) = \cos\left(N\cos^{-1}(x)\right)$ it is easy to see that the roots of the Chebyshev polynomial occur at

$$x = \cos\left(\frac{(n-1/2)\pi}{N}\right) \qquad n = 1\ldots N$$

and from Eq. (25) the system zeros will be at

$$\tau_n = j\Omega_c \cos\left(\frac{(n-1/2)\pi}{N}\right) \qquad n = 1\ldots N.$$

2. The poles and zeros are mapped back to the s-plane using $s = \Omega_r\Omega_c/\tau$ and the N left half-plane poles are selected as the poles of the filter.

3. The transfer function is formed and the system gain is adjusted to unity at $\Omega = 0$.

■ Example

Repeat the previous Chebyshev Type 1 design example using a Chebyshev Type 2 filter.

From the previous example we have $\Omega_c = 10$ rad/s., $\Omega_r = 20$ rad/s., $\epsilon = 1/3$, $\lambda = 4.358$. The procedure to find the required order is the same as before, and we conclude that $N = 3$. Next, define

$$\begin{aligned} \nu &= \frac{\Omega_r\Omega_c}{\Omega} = \frac{200}{\Omega} \\ \hat{\epsilon} &= \frac{1}{\epsilon T_N(\Omega_r/\Omega_c)} = \frac{3}{T_3(2)} = 0.1154 \end{aligned}$$

Determine α:

$$\alpha = \frac{1}{N}\sinh^{-1}\left(\frac{1}{\hat{\epsilon}}\right) = \frac{1}{3}\sinh^{-1}(8.666) = 0.9520$$

and $\sinh\alpha = 1.1024$, and $\cosh\alpha = 1.4884$.

The values of $\gamma_n = (2n-1)\pi/6$ for $n = 1\ldots 6$ are the same as the design for the Type 1 filter, so that the poles of $|H(\tau)|^2$ are

$$\begin{aligned} p_n &= \Omega_c\left(\sinh\alpha\sin\gamma_n + j\cosh\alpha\cos\gamma_n\right) \\ \tau_1 &= 10\left(1.1024\times\frac{1}{2} + j1.4884\times\frac{\sqrt{3}}{2}\right) = 5.512 + j12.890 \\ \tau_2 &= 10\left(1.1024\times 1 + j1.4884\times 0\right) = 11.024 \end{aligned}$$

$$
\begin{aligned}
\tau_3 &= 10\left(1.1024 \times \frac{1}{2} - j1.488 \times \frac{\sqrt{3}}{2}\right) = 5.512 - j12.890 \\
\tau_4 &= 10\left(-1.1024 \times \frac{1}{2} - j1.4884 \times \frac{\sqrt{3}}{2}\right) = -5.512 - j12.890 \\
\tau_5 &= 10\left(-1.1024 \times \frac{1}{2} - j1.488 \times 0\right) = -11.024 \\
\tau_6 &= 10\left(-1.1024 \times \frac{1}{2} + j1.4884 \times \frac{\sqrt{3}}{2}\right) = -5.512 + j12.890
\end{aligned}
$$

The three left half-plane poles (τ_4, τ_5, τ_6) are mapped back to the s-plane using $s = \Omega_r \Omega_c / \tau$ giving three filter poles

$$
\begin{aligned}
p_1, p_2 &= -5.609 \pm j13.117 \\
p_3 &= -18.14
\end{aligned}
$$

The system zeros are the roots of

$$
T_3(\nu/j\Omega_c) = 4(\nu/j\Omega_c)^3 - 3(\nu/j\Omega_c) = 0
$$

from the definition of $T_N(x)$, giving $\nu_1 = 0$ and $\nu_2, \nu_3 = \pm j8.666$. Mapping these back to the s-plane gives two finite zeros $z_1, z_2 = \pm j23.07, z_3 = \infty$ (the zero at ∞ does not affect the system response) and the unity gain transfer function is

$$
\begin{aligned}
H(s) &= \frac{-p_1 p_2 p_3}{z_1 z_2} \frac{(s - z_1)(s - z_2)}{(s - p_1)(s - p_2)(s - p_3)} \\
&= \frac{6.9365(s^2 + 532.2)}{(s + 18.14)(s^2 + 11.22s + 203.5)}
\end{aligned}
$$

The pole-zero plot for this filter is shown in Fig. 8. Note that the poles again lie on ellipse, and the presence of the zeros in the stop-band.

2.3 Comparison of Filter Responses

Bode plot responses for the three example filters are shown in Fig. 9. While all filters meet the design specification, it can be seen that the Butterworth and the Chebyshev Type 1 filters are all-pole designs and have an asymptotic high-frequency magnitude slope of $-20N$ dB/decade, in this case -80 dB/decade for the Butterworth design and -60 dB/decade for the Chebyshev Type 1 design. The Type 2 Chebyshev design has two finite zeros, with the result that its asymptotic high frequency response has a slope of only -20 dB/decade. Note also the singularity in the phase response of the Type 2 Chebyshev filter, caused by the two purely imaginary zeros.

The pass-band and stop-band power responses are shown in Fig. 10. Notice that the design method developed here guarantees that the response will meet the specification at the cut-off frequency (in this case $|H(j\Omega)|^2 = 0.9$ at $\Omega_c = 10$. Other design methods (such as used by Matlab) may not use this criterion.

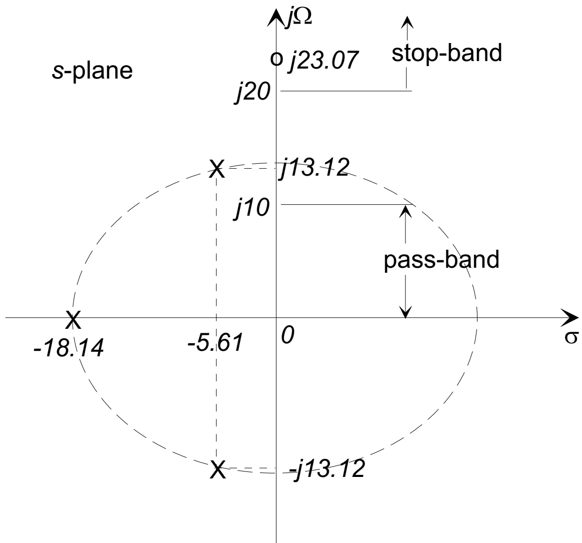


Figure 8: Pole-zero plot for third-order Chebyshev Type 2 design example.

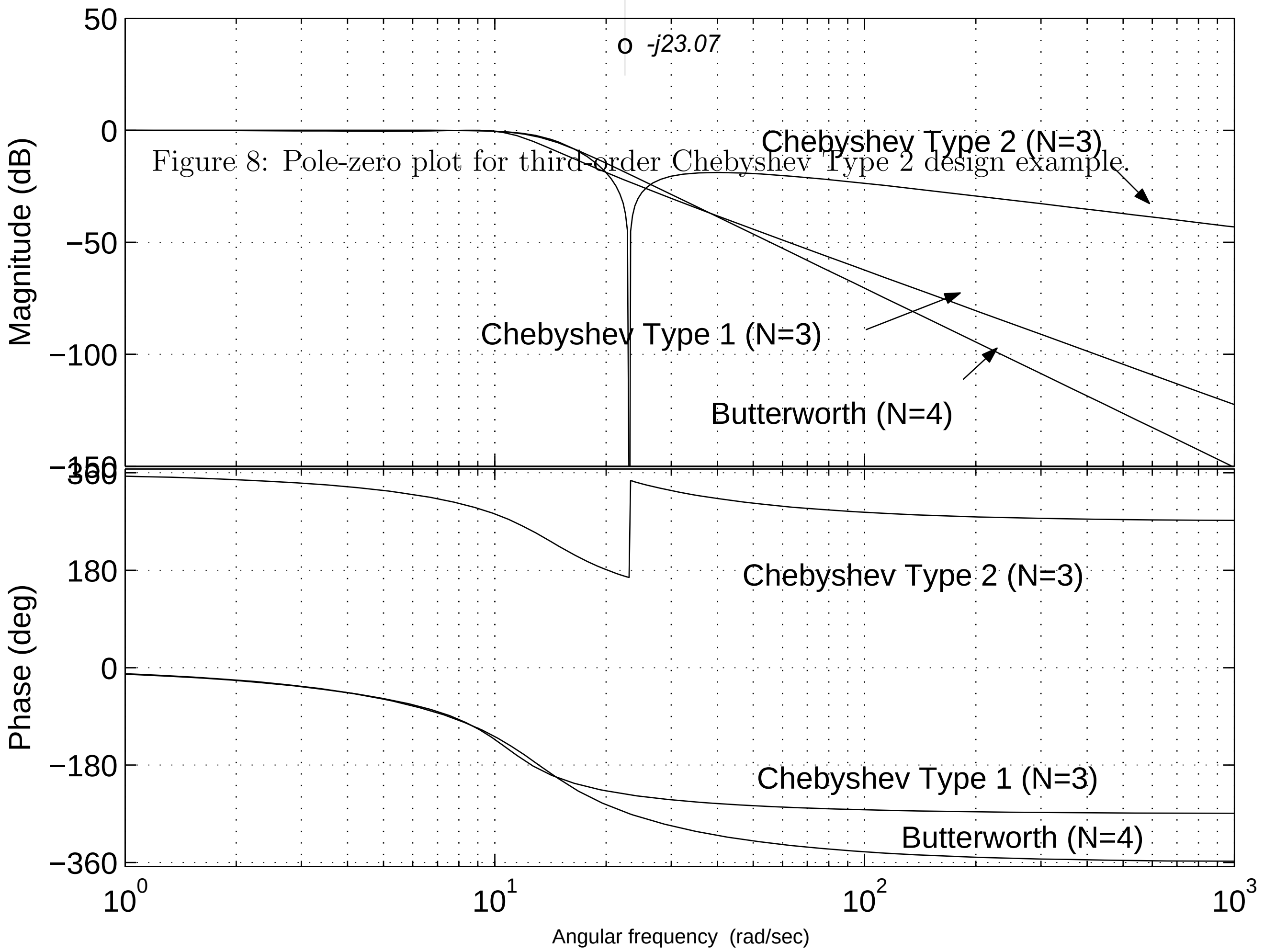


Figure 9: Comparison of Bode plots for the three design examples.

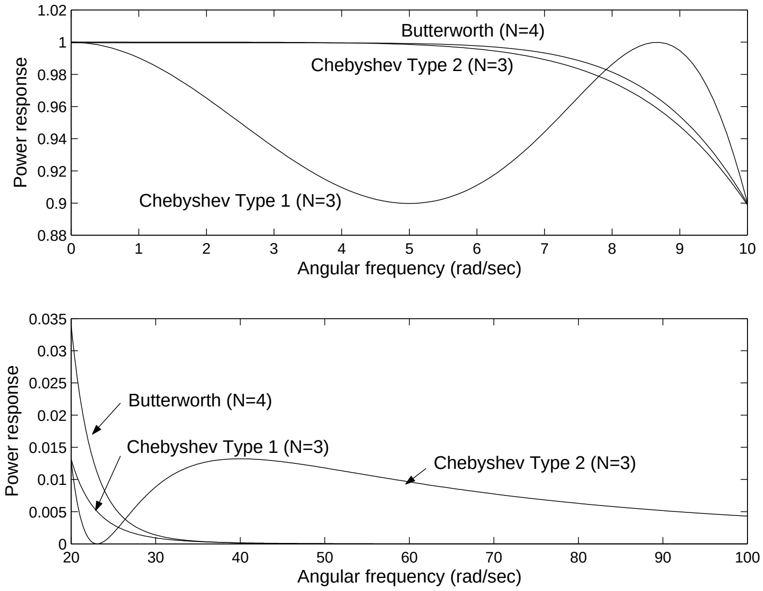


Figure 10: Comparison of the pass-band (top) and stop-band (bottom) power responses for the three design examples.

2.4 Other Filter Designs

While the Butterworth and Chebyshev filters are perhaps the most common designs, there are other filter groups that have additional advantages.

2.4.1 Bessel Filters

The Bessel (or sometimes known as the Thomson) filter has a phase response that most closely approximates a pure time delay, which is often an advantage if phase distortion is to be avoided. In addition the Bessel filters have a very small overshoot in their step response, which make them attractive for the transmission of pulse-like waveforms. The improvement in the phase response comes at the expense of the of the gain response. Although the gain response is monotonic, Bessel filters are not maximally flat, and do not have as steep a transition-band as other filter types. Bessel filters are all-pole filters.

2.4.2 Legendre Filters

The filters, also known as *optimally monotonic*, combine properties of the Butterworth and Chebyshev designs. The Legendre filter has no ripples in its magnitude response, but it is not as flat as the Butterworth design. It has better transition-band cut-off rate than the Butterworth filter, and no ripple in either the passband or the stop-band.

2.4.3 The Elliptic Filter

Also known as the Chebyshev-Cauer filter, allows a very sharp cut-off in the transition-band by allowing ripples in both the pass-band and in the stop-band.

2.5 MATLAB Low-Pass Filter Design Functions

MATLAB filter design functions are contained in the *Signal Processing Toolbox.* See MATLAB's help for complete descriptions.

- Continuous low-pass filter order calculation. The following return the required order, and equivalent -3dB cut-off frequency (W_n) or stop-band edge (W_s) to meet the given specifications:

  `[N, Wn] = buttord(Wc, Wr, Rp, Rs, 's')` - Butterworth filter.
  `[N, Wn] = cheb1ord(Wc, Wr, Rp, Rs, 's')` - Chebyshev Type I filter.
  `[N, Ws] = Cheb2ord(Wc, Wr, Rp, Rs, 's')` - Chebyshev Type II filter.
  `[N, Wn] = ellipord(Wc, Wr, Rp, Rs, 's')` - Elliptic filter.

- Continuous low-pass filter prototypes. Return the zeros, poles, and gain for an nth order filter with a -3dB cut-off($\epsilon = 1$), and cut-off frequency of $\Omega_c = 1$:

  `[Z,P,K] = buttap(N)` - Butterworth filter prototype.
  `[Z,P,K] = cheb1ap(N)` - Chebyshev Type I filter prototype.
  `[Z,P,K] = cheb2ap(N)` - Chebyshev Type II filter prototype.
  `[Z,P,K] = ellipap(N)` - Elliptic filter prototype.

- Continuous filter design. Note that these functions will design both continuous and discrete filters, and that continuous filters require the last argument to be `'s'`, for example
  `[z,p,k] = butter(3,2*pi*150,'s')`
  will design as 3rd-order Butterworth filter with a -3 dB cut-off frequency of 150 Hz and return the zeros, poles and gain. A filter "object" may then be created using
  `filt = zpk(z,p,k)`
  Similarly,
  `[num,den] = butter(3,2*pi*150,'s')`
  will design the same filter and return the transfer function numerator and denominator polynomial coefficients. A filter "object" may then be created using `filt = tf(num, den)`

  `butter(N,Wn,'s')` - Butterworth filter design.
  `cheby1(N,Rp,Wn,'s')` - Chebyshev Type I filter design.
  `cheby2(N,Rs,Ws,'s')` - Chebyshev Type II filter design.
  `ellip(N,Rp,Rs,Wn,'s')` - Elliptic filter design.
  `besself(N,Wo)` - Bessel analog filter design (continuous).

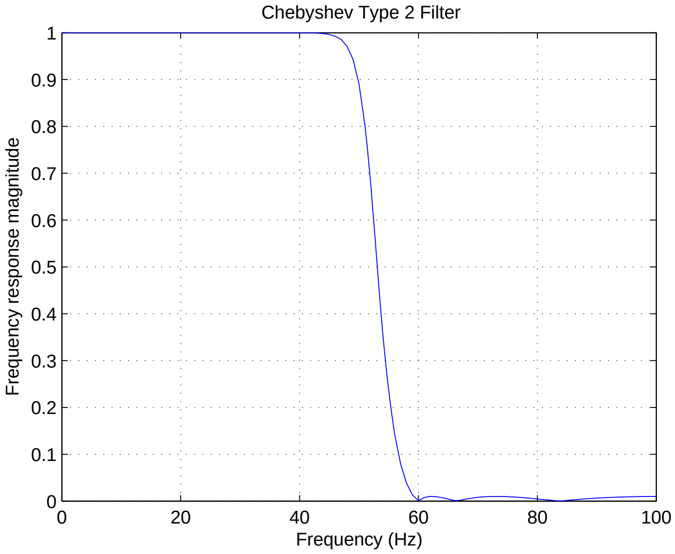
■ Example

Use Matlab to design a Chebyshev Type 2 low-pass filter with $\Omega_c = 50$ Hz, $\Omega_s = 60$ Hz, with maximum attenuation in the pass-band of 1 dB, and minimum attenuation in the stop-band of 40 dB.

The following MATLAB commands design and plot the required filter. (Note that `cheb2ord()` returns `ws`. the stop-band edge frequency, which is then passed to `cheby2()`):

```
[n,ws] = cheb2ord(2*pi*50, 2*pi*60, 1, 40, 's');
[num,den] = cheby2(n,40,ws,'s');
cheby2_lpf = tf(num,den);
f=[0:1:100];
w=2*pi*f;
[mag,phase]=bode(cheby2_lpf);
plot(f, squeeze(mag))
```

generates the following plot



3 Transformation to Other Filter Classes

Filter specification tolerance bands for high-pass, band-pass and band-stop filters are shown in Fig. 11. The most common procedure for the design of these filters is to design a *prototype low-pass filter* using the methods described above, and then to *transform* the low-pass filter to the desired form by a substitution in the transfer function, that is we substitute a function $g(s)$ for s in the low-pass transfer function $H(s)$, so that the new transfer function is $H'(s) = H(g(s))$. The effect is to modify the filter poles and zeros to produce the desired frequency response characteristic. The critical frequencies used in the design are as shown in Fig. 11. For band-pass and band-stop filters it is convenient to define a center frequency Ω_o as the geometric mean of the pass-pand edges, and a band width $\Delta\Omega$:

$$\begin{aligned}\Omega_o &= \sqrt{\Omega_{cu}\Omega_{cl}}\\ \Delta_\Omega &= \Omega_{cu} - \Omega_{cl}.\end{aligned}$$

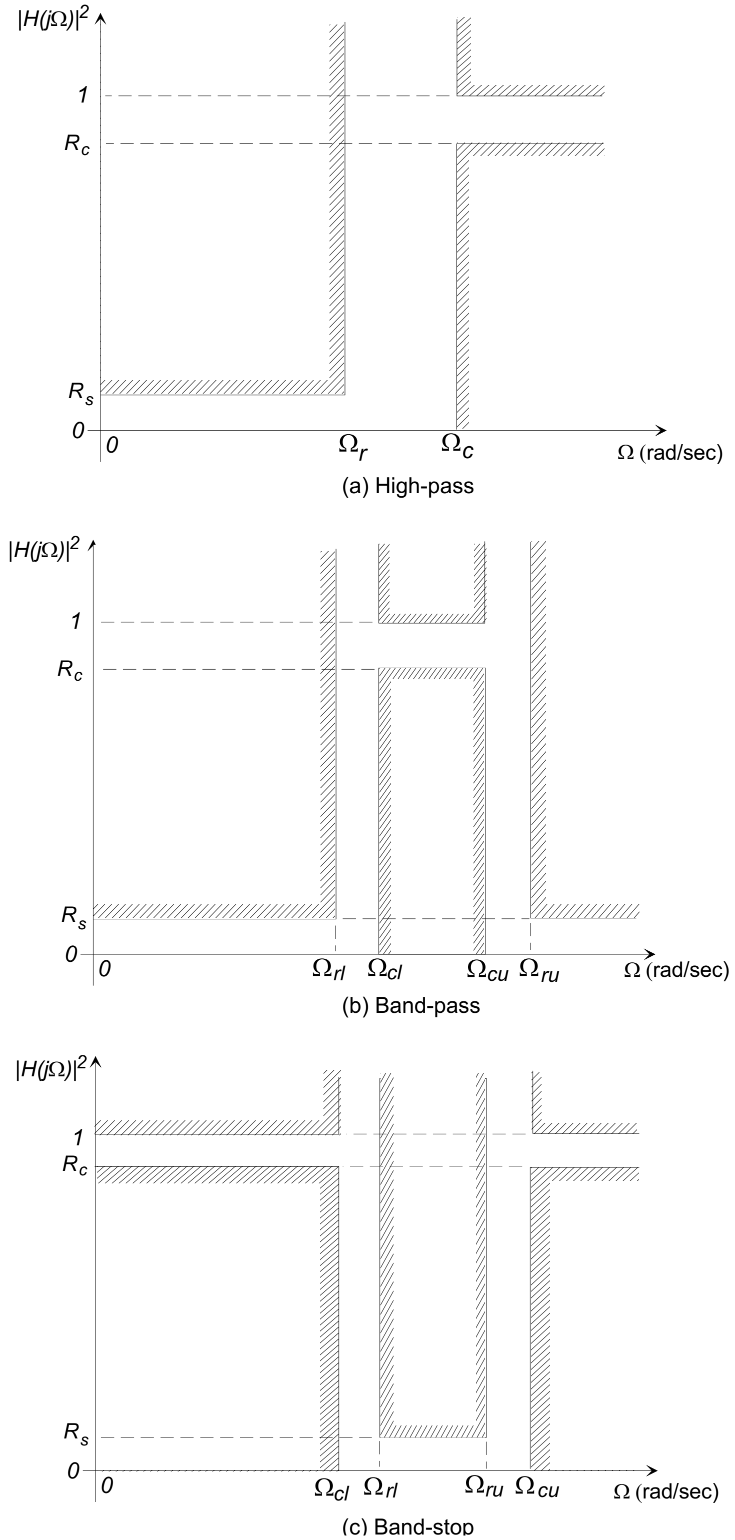


Figure 11: Specification tolerance bands for (a) high-pass, (b) band-pass, and (c) band-stop filters.

The transformation formulas for a low-pass filter with cut-off frequency Ω_c are given in Table 1. The band-pass and band-stop transformations both double the order of the filter, since s^2 is involved it the transformation. In the above table Ω_1 and Ω_2 are the edges of the pass/stop-band, and the low-pass filter is designed to have a cut-off frequency $\Omega_c = \Omega_2 - \Omega_1$.

Low-pass (Ω_{c_1}) → Low-pass (Ω_{c_2})	$g(s) = \frac{\Omega_{c_1}}{\Omega_{c_2}} s$
Low-pass (Ω_c) → High-pass (Ω_c)	$g(s) = \frac{\Omega_c^2}{s}$
Low-pass ($\Omega_c = \Delta_\Omega$) → Band-pass (Ω_{cl}, Ω_{cu})	$g(s) = \frac{s^2 + \Omega_o^2}{s}$
Low-pass ($\Omega_c = \Delta_\Omega$) → Band-stop (Ω_{cl}, Ω_{cu})	$g(s) = \frac{s\Omega_c^2}{s^2 + \Omega_o^2}$

Table 1: Transformation to other filter forms from a prototype low-pass filter with cut-off frequency Ω_c.

The above transformations will create an ideal gain characteristic from an ideal low-pass filter. For practical filters, however, the "skirts" of the pass-bands will be a warped representation of the low-pass prototype filter. This does not usually cause problems.

■ Example

Show the effect of the low-pass to high-pass conversion by examining the poles and zeros of the transformed first-order filter

$$H(s) = \frac{\Omega_c}{s + \Omega_c}.$$

The transformation Ω_c^2/s for s in $H(s)$ gives

$$H'(s) = \frac{s}{s + \Omega_c}.$$

which generates a zero at $s = 0$ (creating the high-pass action) and leaves the pole unchanged. It is easy to show that the low-pass to high-pass transformation on an nth order all-pole filter will create n zeros at the origin.

3.1 MATLAB Filter Transformation Functions

MATLAB's Signal Processing Toolbox contains filter transformation functions `lp2lp()`, `lp2hp()`, `lp2bp()` and `lp2bs()` that transform a prototype low-pass filter with a cut-off frequency of 1 rad/s to another low-pass, a high-pass, a band-pass, or a band-stop filter respectively:

`[NUMT,DENT] = lp2lp(NUM,DEN,Wc)` - Low-pass to low-pass.
`[NUMT,DENT] = lp2hp(NUM,DEN,Wc)` - Low-pass to high-pass.
`[NUMT,DENT] = lp2bp(NUM,DEN,Wo,Bw)` - Low-pass to band-pass.
`[NUMT,DENT] = lp2bs(NUM,DEN,Wo,Bw)` - Low-pass to band-stop.

The requirement that the prototype low-pass filter have a cut-off frequency of $\Omega_c = 1$ rad/s changes the required substitutions in the transfer functions from Table 1 to those given in Table 2. Let the prototype low-pass filter have frequency response $H(j\Omega)$ with cut-off frequency $\Omega_c = 1$. Then Table 2 allows the stop-band limit Ω_r to be computed from

$$\Omega_r = g(s)|_{s=j\Omega_r}$$

as shown in Table 3. Because band-pass and band-stop filters have two stop-band edges Ω_{rl} and Ω_{ru}, we should choose the edge that imposes the the more stringent filter specification, that is the edge that defines the lowest value of Ω_r.

Low-pass ($\Omega_c = 1$) → Low-pass (Ω_c)	$g(s) = \dfrac{s}{\Omega_c}$
Low-pass ($\Omega_c = 1$) → High-pass (Ω_c)	$g(s) = \dfrac{\Omega_c}{s}$
Low-pass ($\Omega_c = 1$) → Band-pass (Ω_{cl}, Ω_{cu})	$g(s) = \dfrac{s^2 + \Omega_o^2}{\Delta_\Omega s}$
Low-pass ($\Omega_c = 1$) → Band-stop (Ω_{cl}, Ω_{cu})	$g(s) = \dfrac{\Delta_\Omega s}{s^2 + \Omega_o^2}$

Table 2: Transformation to other filter forms from a prototype low-pass filter with cut-off frequency $\Omega_c = 1$ as required by MATLAB functions.

	Ω_c	Ω_r
Low-pass	1	$\dfrac{\Omega_r}{\Omega_c}$
High-pass	1	$\dfrac{\Omega_c}{\Omega_r}$
Band-pass	1	$\min\left(\dfrac{\lvert\Omega_o^2 - \Omega_{rl}^2\rvert}{\Delta_\Omega.\Omega_{rl}}, \dfrac{\lvert\Omega_o^2 - \Omega_{ru}^2\rvert}{\Delta_\Omega.\Omega_{ru}}\right)$
Band-stop	1	$\min\left(\dfrac{\Delta_\Omega.\Omega_{rl}}{\lvert\Omega_o^2 - \Omega_{rl}^2\rvert}, \dfrac{\Delta_\Omega.\Omega_{ru}}{\lvert\Omega_o^2 - \Omega_{ru}^2\rvert}\right)$

Table 3: Mapping of critical frequencies as defined in Fig. 11 to a prototype low-pass filter $H(j\Omega)$ with $\Omega_c = 1$ for transformation using MATLAB functions `lp2lp()`, `lp2hp()`,`lp2bp()` and `lp2bs()`

Design Procedure:

1. Determine the filter specifications and choose band edge frequencies and attenuation values as in Fig. 11.
2. Use Table 3 to define Ω_r for the prototype filter.
3. Design the prototype filter using $\Omega_c = 1$, Ω_r, R_c, and R_s.
4. Transform the prototype filter to the desired form.

■ Example

A band-stop filter is to be used to minimize 60 Hz ac interference superimposed on experimental data. A suitable set of specifications was selected as

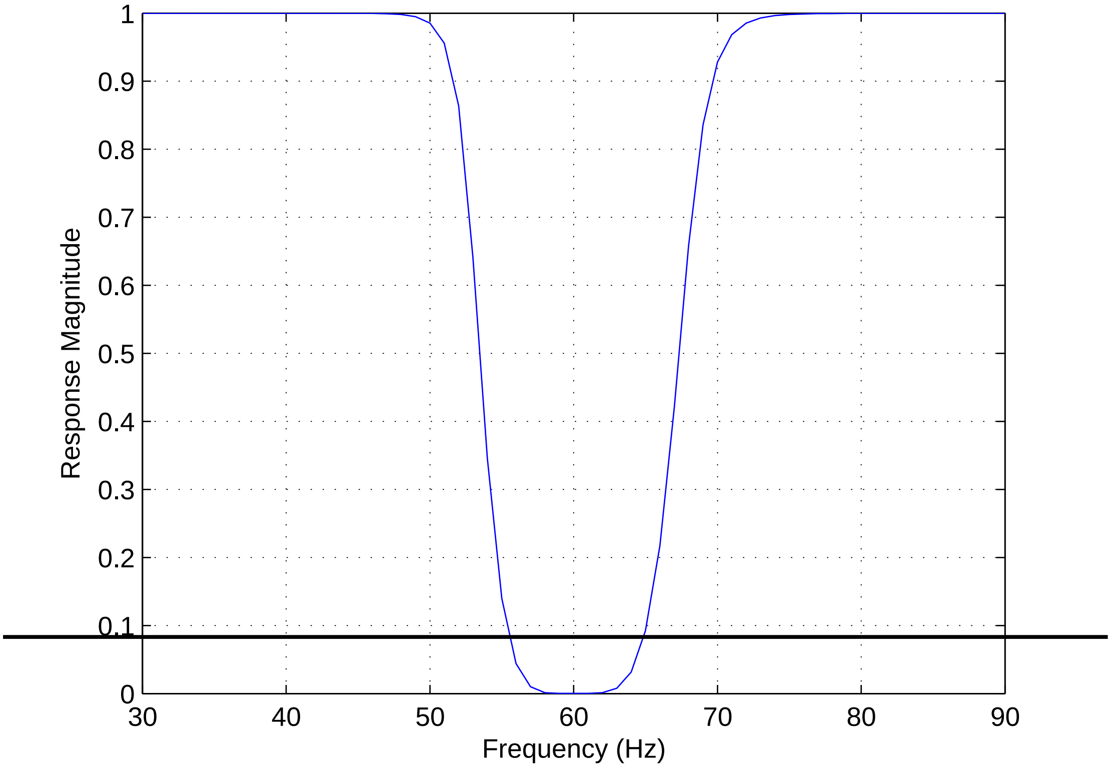
$$\begin{aligned}
\Omega_o &= 2\pi 60 \text{ rad/s} \quad (60 \text{ Hz}) \\
\Omega_{cu} &= 2\pi 70 \text{ rad/s} \quad (70 \text{ Hz}) \\
\Omega_{ru} &= 2\pi 63 \text{ rad/s} \quad (63 \text{ Hz}) \\
\Omega_{r1} &= 2\pi 57 \text{ rad/s} \quad (57 \text{ Hz}) \\
R_c &= 1 \text{ dB} \\
R_s &= 40 \text{ dB}
\end{aligned}$$

The following MATLAB script was used to design the band-stop filter:

```
% Design a band-stop filter
% First define some ctitical frequencies
%  Passband edges
wo=2*pi*60; wcu=2*pi*70;
wcl = wo^2/wcu;
BW=(wcu-wcl);
% Stop band edges
wsu=2*pi*63; wsl=2*pi*57;
%
Rc = 1; Rs = 40;
% Determine the stop-band edge in the lp prototype
%
W1 = BW*wsu/(wo^2-wsu^2);
W2 = BW*wsl/(wo^2-wsl^2);
Wr = min(abs(W1),abs(W2));
%
% design the prototype low-pass filter
%
[N,Wn]      = buttord(1,Wr, Rc, Rs, ’s’);
[num,den]   = butter(N, Wn, ’s’);
% Convert to a band-stop filter
```

```
[num_stop,den_stop] = lp2bs(num,den,wo,BW);
filt = tf(num_stop,den_stop);
%
% Plot the frequency response magnitude
%
f=[30:1:90];
[mag,phase]=bode(filt,2*pi*f);
plot(f,(squeeze(mag)))
grid; xlabel('Frequency (Hz)'); ylabel('Response Magnitude')
```

which produces the following plot



A Sample Low-Pass Filter Design Routines

The following MATLAB ".m" routines are included to demonstrate the Butterworth and Chebyshev pole placement methods described in this handout.

A.1 Butterworth Low-Pass Filter

```
%     *** 2.161 Signal Processing - Continuous and Discrete  ***
%
%   lpbutter -  Butterworth continuous lowpass filter design based on
%       derivation in the 2.161 class notes.
%
%   [z,p,k,n] = lpbutter(wc, Rc, ws, Rs) or
%   [z,p,k,n] = lpbutter(wc, Rc, ws, Rs,’zpk’) returns the zeros, poles, and gain
%                  and order for a Butterworth continuous lowpass filter.
%   [b,a,n]   = lpbutter(wc, Rc, ws, Rs,’tf’) returns the transfer function
%                  numerator and denominator coefficients, and order for a
%                  Butterworth continuous lowpass filter.
%   Arguments:
%      wc  - passband cut-off frequency (rad/s)
%      Rc  - attenuation at passband cut-off (dB) (Rc > 0)
%      ws  - stopband frequency (rad/s)
%      Rs  - attenuation at stopband (dB) (Rs > 0).
%
%   Author:   D. Rowell
%   Revision: 2.0 9-23-2007
%
%------------------------------------------------------------
function [varargout] = lpbutter(wc,Rc,ws,Rs,varargin)
%
% Compute the required order n
%
  epsilon = sqrt(10^(Rc/10)-1);
  lambda  = sqrt(10^(Rs/10)-1);
  n       = ceil(log(lambda/epsilon)/log(ws/wc));
%
% There are no zeros in the lpf - create an empty zeros vector:
%
  z = [];
%
% Create the complex conjugate poles on a circle
%
  p = wc*epsilon^(-1/n)*exp(i*(pi*(1:2:n-1)/(2*n) + pi/2));
  p = [p; conj(p)];
  p = p(:);
% If n is odd, add a real pole
  if rem(n,2)==1   % n is odd
```

```
    p = [p; -wc*epsilon^(-1/n)];
  end
%
% For unity low freq gain, the k is the product of the negative of the poles.
%
  k = real(prod(-p));
%
% Return the appropriate system representation
%
  if (nargin == 5 )
      if (varargin{1}(1:2) == 'tf')
         sys = zpk(z,p,k);
         [b,a] = tfdata(sys);
         varargout(1) = b;
         varargout(2) = a;
         varargout(3) = {n};
      elseif (varargin{1}(1:3) == 'zpk')
         varargout(1)={z};
         varargout(2)={p};
         varargout(3)={k};
         varargout(4)={n};
      end
  else
      varargout(1)={z};
      varargout(2)={p};
      varargout(3)={k};
      varargout(4)={n};
  end
```

A.2 Chebyshev Type 1 Low-pass Filter

```
%     *** 2.161 Signal Processing - Continuous and Discrete  ***
%
%   lpcheby1 -  Chebyshev Type 1 continuous lowpass filter design based on
%       derivation in the 2.161 class notes.
%
%   [z,p,k,n] = lpcheby1(wc, Rc, ws, Rs) or
%   [z,p,k,n] = lpcheby1(wc, Rc, ws, Rs,’zpk’) returns the zeros, poles, and gain
%                  and order for a Chebyshev Type 1 continuous lowpass filter.
%   [b,a,n]   = lpcheby1(wc, Rc, ws, Rs,’tf’) returns the transfer function
%                  numerator and denominator coefficients, and order for a
%                  Chebyshev Type 1 continuous lowpass filter.
%   Arguments:
%      wc  - passband cut-off frequency (rad/s)
%      Rc  - attenuation at passband cut-off (dB) (Rc > 0)
%      ws  - stopband frequency (rad/s)
%      Rs  - attenuation at stopband (dB) (Rs > 0).
%
%   Author:   D. Rowell
%   Revision: 2.0 9-23-2007
%
%---------------------------------------------------------------
function [varargout] = lpcheby1(wc,Rc, ws, Rs,varargin)
%
% Compute the required order n
%
  epsilon = sqrt(10^(Rc/10)-1);
  lambda  = sqrt(10^(Rs/10)-1);
  n       = ceil(acosh(lambda/epsilon)/acosh(ws/wc));
%
% There are no zeros in the lpf - create an empty zeros vector:
%
  z = [];
%
% Determine the poles - choosing first only complex conjugates in the
% left-half plane
%
alpha = asinh(1/epsilon)/n;
p = [];
for j=1:floor(n/2)
   gamma = (2*j - 1)*pi/(2*n);
   newp = -wc*(sinh(alpha)*sin(gamma)+ i*cosh(alpha)*cos(gamma));
   p = [p; newp; conj(newp)];
end
% If n is odd - add a real pole
  if rem(n,2)==1   % n is odd
     gamma = (2*(floor(n/2)+1) - 1)*pi/(2*n);
     p = [p; -wc*(sinh(alpha)*sin(gamma))];
  end
% Compute the gain
k = real(prod(-p));
%
% If n is even adjust the gain to unity at zero frequency
%
if ~rem(n,2)    % n is even so patch k
    k = k/sqrt((1 + epsilon^2));
end
%
% Return the appropriate system representation
%
  if (nargin == 5 )
      if (varargin{1}(1:2) == 'tf')
         sys = zpk(z,p,k);
         [b,a] = tfdata(sys);
         varargout(1) = b;
         varargout(2) = a;
         varargout(3) = {n};
      elseif (varargin{1}(1:3) == 'zpk')
         varargout(1)={z};
         varargout(2)={p};
         varargout(3)={k};
         varargout(4)={n};
      end
  else
      varargout(1)={z};
      varargout(2)={p};
      varargout(3)={k};
      varargout(4)={n};
  end
```

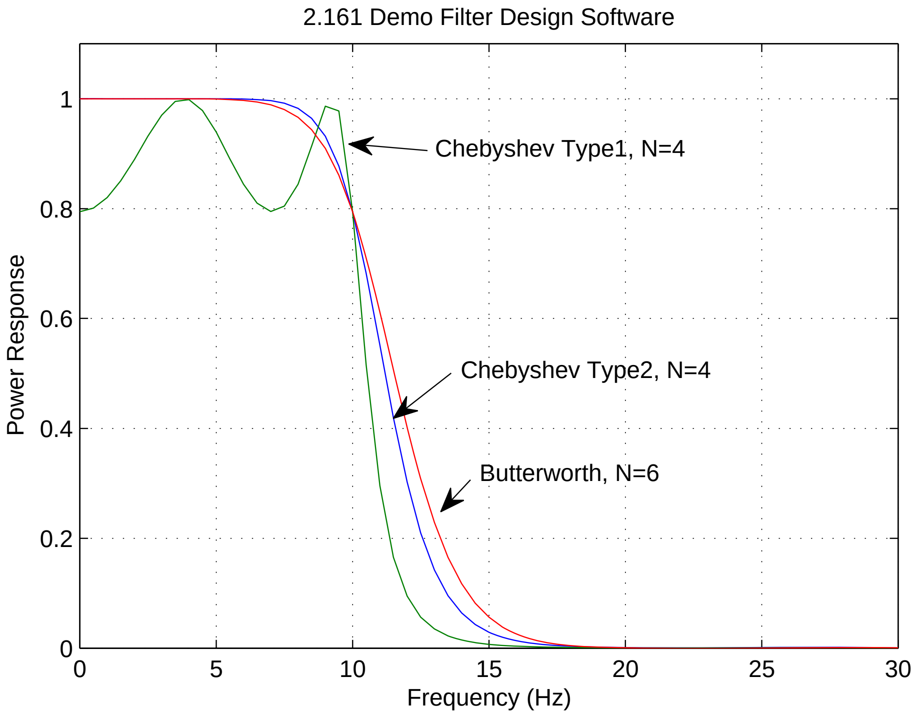
A.3 Chebyshev Type 2 Low-pass Filter

```
%     *** 2.161 Signal Processing - Continuous and Discrete  ***
%
%   lpcheby2 -  Chebyshev Type 2 continuous lowpass filter design based on
%       derivation in the 2.161 class notes.
%
%   [z,p,k,n] = lpcheby2(wc, Rc, ws, Rs) or
%   [z,p,k,n] = lpcheby2(wc, Rc, ws, Rs,'zpk') returns the zeros, poles, and gain
%                  and order for a Chebyshev Type 2 continuous lowpass filter.
%   [b,a,n]   = lpcheby2(wc, Rc, ws, Rs,'tf') returns the transfer function
%                  numerator and denominator coefficients, and order for a
%                  Chebyshev Type 2 continuous lowpass filter.
%   Arguments:
%      wc  - passband cut-off frequency (rad/s)
%      Rc  - attenuation at passband cut-off (dB) (Rc > 0)
%      ws  - stopband frequency (rad/s)
%      Rs  - attenuation at stopband (dB) (Rs > 0).
%
%   Author:   D. Rowell
%   Revision: 2.0 9-23-2007
%
%---------------------------------------------------------------
function [varargout] = lpcheby2(wc,Rc, ws, Rs,varargin)
%
% Compute the required order n
%
  epsilon = sqrt(10^(Rc/10)-1);
  lambda  = sqrt(10^(Rs/10)-1);
  n       = ceil(acosh(lambda/epsilon)/acosh(ws/wc));
  wprod   = wc*ws;
  epsilonhat= 1/(epsilon*cos(n*acos(ws/wc)));
%
% Create the finite (imaginary) zeros
% Handle even and odd orders separately to eliminate the zero at infinity
% for n odd
if (rem(n,2))
    m = n - 1;
    z = cos([1:2:n-2 n+2:2:2*n-1]*pi/(2*n))';
else
    m = n;
    z = cos((1:2:2*n-1)*pi/(2*n))';
end
z = (z - flipud(z))./2;
% Transform back to the s-plane
z = i*ws./z;
% Organize zeros in complex pairs:
j = [1:m/2; m:-1:m/2+1];
```

```
z = z(j(:));
%
% Determine the poles - choosing first only complex conjugates in the
% left-half plane
%
alpha = asinh(1/epsilonhat)/n;
p = [];
for j=1:floor(n/2)
   gamma = (2*j - 1)*pi/(2*n);
   newp = -wc*(sinh(alpha)*sin(gamma)+ i*cosh(alpha)*cos(gamma));
   p = [p; newp; conj(newp)];
end
% If n is odd - add a real pole
  if rem(n,2)==1   % n is odd
     gamma = (2*(floor(n/2)+1) - 1)*pi/(2*n);
     p = [p; -wc*(sinh(alpha)*sin(gamma))];
  end
  p=wprod./p;
% Compute the gain
k = real(prod(-p)/prod(-z));
%
% Return the appropriate system representation
%
  if (nargin == 5 )
      if (varargin{1}(1:2) == 'tf')
         sys = zpk(z,p,k);
         [b,a] = tfdata(sys);
         varargout(1) = b;
         varargout(2) = a;
         varargout(3) = {n};
      elseif (varargin{1}(1:3) == 'zpk')
         varargout(1)={z};
         varargout(2)={p};
         varargout(3)={k};
         varargout(4)={n};
      end
  else
      varargout(1)={z};
      varargout(2)={p};
      varargout(3)={k};
      varargout(4)={n};
  end
```

A.4 Demonstration of Filter Design Routines

```
% Test program for low-pass filter design.
% Designs lp filters using the example in the class handout
% D.Rowell   9/20/2007
%
% Define the filter specifications:
%  cut-off frequency: 10 Hz,    max pass-band attenuation: 1 dB
wc = 2*pi*10;   passbandatten = 1;
%  stop-band: 20Hz:    min stop-band attenuation: 30 dB
ws = 2*pi*20;   stopbandatten = 30;
% Design the filters
[z,p,k,nbutt] = lpbutter(wc,passbandatten,ws,stopbandatten);
buttfilt=zpk(z,p,k);
[z,p,k,ncheby1] = lpcheby1(wc,passbandatten,ws,stopbandatten);
cheby1filt=zpk(z,p,k);
[z,p,k,ncheby2] = lpcheby2(wc,passbandatten,ws,stopbandatten);
cheby2filt=zpk(z,p,k);
% plot the three power responses on a linear scale
f=[0:.5:30];   w=2*pi*f;
[buttmag, buttphase]     = bode(buttfilt,w);
[cheby1mag, cheby1phase] = bode(cheby1filt,w);
[cheby2mag, cheby2phase] = bode(cheby2filt,w);
plot(f,squeeze(buttmag).^2, f,squeeze(cheby1mag).^2, f,squeeze(cheby2mag).^2);
title('2.161 Demo Filter Design Software');
xlabel('Frequency (Hz)');   ylabel('Power Response');
grid;
```


2.161 Demo Filter Design Software
Chebyshev Type1, N=4
Chebyshev Type2, N=4
Butterworth, N=6
Power Response
Frequency (Hz)