Analog Filters Design

Filter Transformations

Low-Pass Filter Design

Remember

- Steps for Designing a Low-Pass Butterworth Filter Approximation:

Design Procedure:

1. Determine the filter order

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

2. Determine the Butterworth radius

$$r = \Omega_c \epsilon^{-1/N}$$

3. Determine The Butterworth angles

$$\theta_n = \pi \frac{2n + N - 1}{2N} \qquad n = 1 \ldots 2N$$

4. Determine the N left half-plane poles

$$p_n = r e^{j\theta_n} \qquad n = 1, \ldots, N$$

5. Form the transfer function

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

➢ Steps for Designing a Low-Pass Chebyshev Type 1 Filter Approximation:

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

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

the response $|H(j0)|^2 = 1$

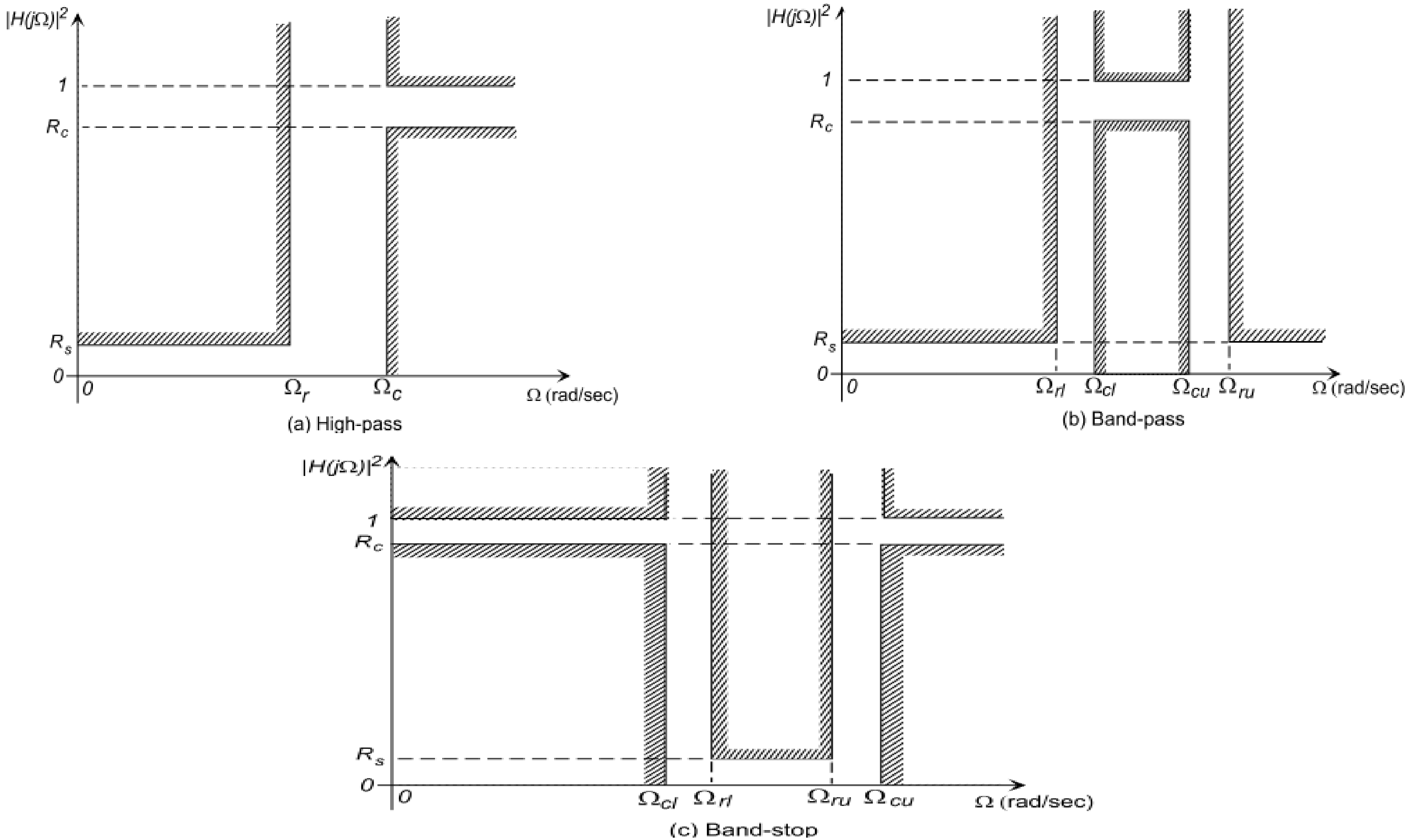
(b) If N is even

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

$$|H(j0)|^2 = 1/(1+\epsilon^2)$$

Transformation to Other Filter Classes

- Filter specification tolerance bands for high-pass, band-pass and band-stop filters are shown in Figure 11.



Transformation to Other Filter Classes

- The most common procedure for the design of these filters is to design a prototype LPF using the methods described above, and then
- Transform the LPF to the desired form by a substitution in the transfer function H(s):

$$S \rightarrow g(s)$$

$$H(s) \rightarrow H'(s) = H(g(s))$$

- The effect is to modify the filter poles and zeros to produce the desired frequency response characteristic.

Where the center frequency of the band-pass and band-stop filters is:

And the bandwidth is

$$\Omega_o = \sqrt{\Omega_{cu}\Omega_{cl}}$$

$$\Delta_\Omega = \Omega_{cu} - \Omega_{cl}.$$

Transformation to Other Filter Classes

The transformation formulas for a low-pass filter with cut-off frequency Ω_c are given below

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

■ 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

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

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

Transformation to Other Filter Classes

MATLAB Filter Transformation Functions

```
[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.

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

$$\Omega_c = 1 \text{ rad/s}$$

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

Table 2.

Transformation to Other Filter Classes

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.

stop-band limit Ω_r to be computed from

Table 3:

	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	$\min\left(\dfrac{\Delta_\Omega.\Omega_{rl}}{\lvert\Omega_o^2-\Omega_{rl}^2\rvert}, \dfrac{\Delta_\Omega.\Omega_{ru}}{\lvert\Omega_o^2-\Omega_{ru}^2\rvert}\right)$

MATLAB Low-Pass Filter Design Functions

MATLAB filter design functions are contained in the *Signal Processing Toolbox.*

1. Continuous low-pass filter order calculation.

- The following return the required order (N), and equivalent -3dB cut-off frequency **(Wn)** to meet the given specifications.
- Specifications: no more than Ap dB of pass-band ripple and at least As dB of attenuation in the stop-band.

```
[n,Wn] = buttord    (Wp,Ws,Ap,As,'s')
[n,Wn] = cheb1ord (Wp,Ws,Rp,Rs,'s')
[n,Wn] = cheb2ord (Wp,Ws,Ap,As,'s')
```

```
%% Example: design  the  butterworth prototype  low-pass  filter
   % Wp =1, Wr =3.059 , Ap =1 dB, As = 40 dB

[N,Wn]  =  buttord(1,3.059,  1,  40,  's')
```

```
N = 5
Wn = 1.0027  %% cutoff frequency at -3 dB
```

MATLAB Low-Pass Filter Design Functions

2. Continuous filter design.

- [b,a] = butter(n,Wn);
  - returns the transfer function coefficients of an nth-order L.P.F Butterworth filter with cutoff frequency Wn.
- [b,a] = butter(n,Wn,ftype)
  - designs a ('low' - 'bandpass' - 'high' -'stop') Butterworth filter, depending on the value of ftype .
- [Z,P,K] = butter(N, Wn, ‘s’);
  - designs a ('low' - 'bandpass' - 'high' -'stop') Butterworth filter and returns its zeros, poles, and gain

$$H(s) = \frac{B(s)}{A(s)} = \frac{b(1)\ s^{n} + b(2)\ s^{n-1} + \dots + b(n+1)}{a(1)\ s^{n} + a(2)\ s^{n-1} + \dots + a(n+1)}.$$

$$H(s) = k\frac{(s - z(1))\ (s - z(2)) \cdots (s - z(n))}{(s - p(1))\ (s - p(2)) \cdots (s - p(n))}.$$

[z,p,k] =

or

[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).

MATLAB Low-Pass Filter Design Functions

```
%% Example: design  the  butterworth prototype  low-pass  filter
   % Wp =1, Wr =3.059 , Ap =1 dB, As = 40 dB

[N,Wn]  =  buttord(1,3.059,  1,  40,  's')
```

```
N = 5
Wn = 1.0027  %% cutoff frequency at -3 dB
```

```
[num,den]  =  butter(N,  Wn,  ’s’);
filt  =  tf(num,den);
%  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’)
```

■ 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.

```
[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))
```