Spectral Transformations of IIR Digital Filters

- Objective - Transform a given lowpass digital transfer function $G_L(z)$ to another digital transfer function $G_D(\hat{z})$ that could be a lowpass, highpass, bandpass or bandstop filter
- z^{-1} has been used to denote the unit delay in the prototype lowpass filter $G_L(z)$ and $\hat{z}^{-1}$ to denote the unit delay in the transformed filter $G_D(\hat{z})$ to avoid confusion

Spectral Transformations of IIR Digital Filters

- Unit circles in z- and $\hat{z}$-planes defined by

$$z = e^{j\omega}, \quad \hat{z} = e^{j\hat{\omega}}$$

- Transformation from z-domain to $\hat{z}$-domain given by

$$z = F(\hat{z})$$

- Then

$$G_D(\hat{z}) = G_L\{F(\hat{z})\}$$

Spectral Transformations of IIR Digital Filters

- From $z = F(\hat{z})$, thus $|z| = |F(\hat{z})|$, hence

$$|F(\hat{z})| \begin{cases} >1, & \text{if } |z| > 1 \\ =1, & \text{if } |z| = 1 \\ <1, & \text{if } |z| < 1 \end{cases}$$

- Recall that a causal stable allpass function $\mathcal{A}(z)$ satisfies the condition

Spectral Transformations of IIR Digital Filters

$$\left|\mathcal{A}(z)\right| \begin{cases} <1, \text{ if } |z|>1 \\ =1, \text{ if } |z|=1 \\ >1, \text{ if } |z|<1 \end{cases}$$

- Therefore $1/F(\hat{z})$ must be a causal stable allpass function whose general form is

$$\frac{1}{F(\hat{z})} = \pm\prod_{\ell=1}^{L}\left(\frac{1-\alpha_\ell^*\hat{z}}{\hat{z}-\alpha_\ell}\right), \quad \left|\alpha_\ell\right|<1$$

Lowpass-to-Lowpass Spectral Transformation

- To transform a lowpass filter $G_L(z)$ with a cutoff frequency ω_c to another lowpass filter $G_D(\hat{z})$ with a cutoff frequency $\hat{\omega}_c$, the transformation is

$$z^{-1} = \frac{1}{F(\hat{z})} = \frac{1-\lambda\hat{z}}{\hat{z}-\lambda}$$

where λ is a function of the two specified cutoff frequencies

Lowpass-to-Lowpass Spectral Transformation

- On the unit circle we have

$$e^{-j\omega} = \frac{e^{-j\hat{\omega}} - \lambda}{1 - \lambda e^{-j\hat{\omega}}}$$

- From the above we get

$$e^{-j\omega} \mp 1 = \frac{e^{-j\hat{\omega}} - \lambda}{1 - \lambda e^{-j\hat{\omega}}} \mp 1 = (1 \pm \lambda)\frac{e^{-j\hat{\omega}} \mp 1}{1 - \lambda e^{-j\hat{\omega}}}$$

- Taking the ratios of the above two expressions

$$\tan(\omega/2) = \left(\frac{1+\lambda}{1-\lambda}\right)\tan(\hat{\omega}/2)$$

Lowpass-to-Lowpass Spectral Transformation

- Solving we get $$\lambda = \frac{\sin\left((\omega_c - \hat{\omega}_c)/2\right)}{\sin\left((\omega_c + \hat{\omega}_c)/2\right)}$$
- Example - Consider the lowpass digital filter

$$G_L(z) = \frac{0.0662(1+z^{-1})^3}{(1-0.2593z^{-1})(1-0.6763z^{-1}+0.3917z^{-2})}$$

which has a passband from dc to 0.25π with a 0.5 dB ripple

- Redesign the above filter to move the passband edge to 0.35π

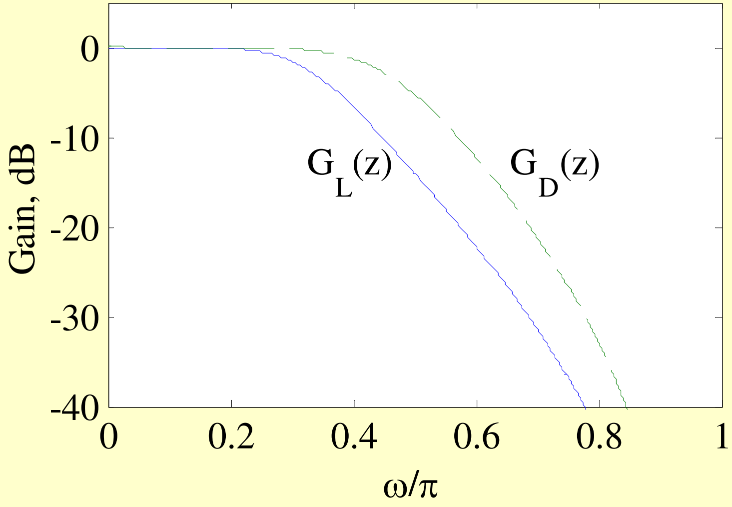
Lowpass-to-Lowpass Spectral Transformation

- Here

$$\lambda = -\frac{\sin(0.05\pi)}{\sin(0.3\pi)} = -0.1934$$

- Hence, the desired lowpass transfer function is

$$G_D(\hat{z}) = G_L(z)\Big|_{z^{-1} = \frac{\hat{z}^{-1} + 0.1934}{1 + 0.1934\,\hat{z}^{-1}}}$$



Lowpass-to-Lowpass Spectral Transformation

- The lowpass-to-lowpass transformation

$$z^{-1} = \frac{1}{F(\hat{z})} = \frac{1-\lambda\hat{z}}{\hat{z}-\lambda}$$

can also be used as highpass-to-highpass, bandpass-to-bandpass and bandstop-to-bandstop transformations

Lowpass-to-Highpass Spectral Transformation

- Desired transformation

$$z^{-1} = -\frac{\hat{z}^{-1} + \lambda}{1 + \lambda \hat{z}^{-1}}$$

- The transformation parameter λ is given by

$$\lambda = -\frac{\cos\left((\omega_c + \hat{\omega}_c)/2\right)}{\cos\left((\omega_c - \hat{\omega}_c)/2\right)}$$

where ω_c is the cutoff frequency of the lowpass filter and $\hat{\omega}_c$ is the cutoff frequency of the desired highpass filter

Lowpass-to-Highpass Spectral Transformation

- Example - Transform the lowpass filter

$$G_L(z) = \frac{0.0662(1+z^{-1})^3}{(1-0.2593\,z^{-1})(1-0.6763\,z^{-1}+0.3917\,z^{-2})}$$

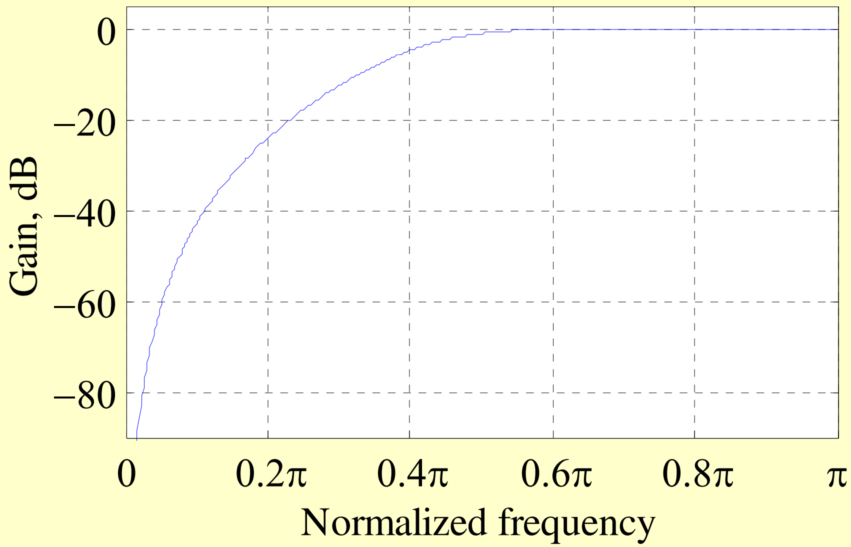
- with a passband edge at 0.25π to a highpass filter with a passband edge at 0.55π
- Here $\lambda = -\cos(0.4\pi)/\cos(0.15\pi) = -0.3468$
- The desired transformation is

$$z^{-1} = -\frac{\hat{z}^{-1} - 0.3468}{1 - 0.3468\hat{z}^{-1}}$$

Lowpass-to-Highpass Spectral Transformation

- The desired highpass filter is

$$G_D(\hat{z}) = G(z)\Big|_{z^{-1} = -\frac{\hat{z}^{-1} - 0.3468}{1 - 0.3468\,\hat{z}^{-1}}}$$



Lowpass-to-Highpass Spectral Transformation

- The lowpass-to-highpass transformation can also be used to transform a highpass filter with a cutoff at ω_c to a lowpass filter with a cutoff at $\hat{\omega}_c$

  and transform a bandpass filter with a center frequency at ω_o to a bandstop filter with a center frequency at $\hat{\omega}_o$

Lowpass-to-Bandpass Spectral Transformation

- Desired transformation

$$z^{-1} = -\frac{\hat{z}^{-2} - \frac{2\lambda\rho}{\rho+1}\hat{z}^{-1} + \frac{\rho-1}{\rho+1}}{\frac{\rho-1}{\rho+1}\hat{z}^{-2} - \frac{2\lambda\rho}{\rho+1}\hat{z}^{-1} + 1}$$

Lowpass-to-Bandpass Spectral Transformation

- The parameters λ and ρ are given by

$$\lambda = \frac{\cos\big((\hat{\omega}_{c2} + \hat{\omega}_{c1})/2\big)}{\cos\big((\hat{\omega}_{c2} - \hat{\omega}_{c1})/2\big)}$$

$$\rho = \cot\big((\hat{\omega}_{c2} - \hat{\omega}_{c1})/2\big)\tan(\omega_c/2)$$

where ω_c is the cutoff frequency of the lowpass filter, and $\hat{\omega}_{c1}$ and $\hat{\omega}_{c2}$ are the desired upper and lower cutoff frequencies of the bandpass filter

Lowpass-to-Bandpass Spectral Transformation

- Special Case - The transformation can be simplified if $\omega_c = \hat{\omega}_{c2} - \hat{\omega}_{c1}$
- Then the transformation reduces to

$$z^{-1} = -\hat{z}^{-1} \frac{\hat{z}^{-1} - \lambda}{1 - \lambda \hat{z}^{-1}}$$

where $\lambda = \cos(\hat{\omega}_o)$ with $\hat{\omega}_o$ denoting the desired center frequency of the bandpass filter

Lowpass-to-Bandstop Spectral Transformation

- Desired transformation

$$z^{-1} = \frac{\hat{z}^{-2} - \frac{2\lambda}{1+\rho}\hat{z}^{-1} + \frac{1-\rho}{1+\rho}}{\frac{1-\rho}{1+\rho}\hat{z}^{-2} - \frac{2\lambda}{1+\rho}\hat{z}^{-1} + 1}$$

Lowpass-to-Bandstop Spectral Transformation

- The parameters λ and ρ are given by

$$\lambda = \frac{\cos\left((\hat{\omega}_{c2} + \hat{\omega}_{c1})/2\right)}{\cos\left((\hat{\omega}_{c2} - \hat{\omega}_{c1})/2\right)}$$

$$\rho = \tan\left((\hat{\omega}_{c2} - \hat{\omega}_{c1})/2\right)\tan(\omega_c/2)$$

where ω_c is the cutoff frequency of the lowpass filter, and $\hat{\omega}_{c1}$ and $\hat{\omega}_{c2}$ are the desired upper and lower cutoff frequencies of the bandstop filter

Generation of Allpass Function Using MATLAB

- The allpass function needed for the spectral transformation from a specified lowpass transfer function to a desired highpass or bandpass or bandstop transfer function can be generated using MATLAB

Generation of Allpass Function Using MATLAB

- **Lowpass-to-Highpass Transformation**
- Basic form:

```
[AllpassNum,AllpassDen] =
allpasslp2hp(wold,wnew)
```

where `wold` is the specified angular bandedge frequency of the original lowpass filter, and `wnew` is the desired angular bandedge frequency of the highpass filter

Generation of Allpass Function Using MATLAB

- **Lowpass-to-Bandpass Transformation**
- Basic form:

```
[AllpassNum,AllpassDen] =
allpasslp2bp(wold,wnew)
```

where `wold` is the specified angular bandedge frequency of the original lowpass filter, and `wnew` is the desired angular bandedge frequency of the bandpass filter

Generation of Allpass Function Using MATLAB

- **Lowpass-to-Bandstop Transformation**
- Basic form:

```
[AllpassNum,AllpassDen] =
allpasslp2bs(wold,wnew)
```

where `wold` is the specified angular bandedge frequency of the original lowpass filter, and `wnew` is the desired angular bandedge frequency of the bandstop filter

Generation of Allpass Function Using MATLAB

- **Lowpass-to-Highpass Example –**

  wold = 0.25π, wnew = 0.55π
- The MATLAB statement

  ```
  [APnum, APden]
  = allpasslp2hp(0.25, 0.55)
  ```

  yields the mapping

$$z^{-1} \rightarrow \frac{-z^{-1} + 0.3468}{-0.3468 z^{-1} + 1}$$

Spectral Transformation Using MATLAB

- The pertinent M-files are `iirlp2lp`, `iirlp2hp`, `iirlp2bp`, and `iirlp2bs`
- **Lowpass-to-Highpass Example –**

$$G_{LP}(z) = \frac{0.066(1+z^{-1})^3}{1-0.9353z^{-1}+0.5669z^{-2}-0.1015z^{-3}}$$

Passband edge wold = 0.25π

Desired passband edge of highpass filter wnew = 0.55π

Spectral Transformation Using MATLAB

- The MATLAB code fragments used are

```
b = 0.066*[1 3 3 1];
a = [1.00  -0.9353  0.5669 -0.1015];
[num,den,APnum,APden]
= iirlp2hp(b,a,0.25,0.55);
```

- The desired highpass filter obtained is

$$G_{HP}(z) = \frac{0.218(1-z^{-1})^3}{1-0.3521z^{-1}+0.3661z^{-2}-0.0329z^{-3}}$$

IIR Digital Filter Design Using MATLAB

- Order Estimation -
- For IIR filter design using bilinear transformation, MATLAB statements to determine the order and bandedge are:

[N, Wn] = buttord(Wp, Ws, Rp, Rs);

[N, Wn] = cheb1ord(Wp, Ws, Rp, Rs);

[N, Wn] = cheb2ord(Wp, Ws, Rp, Rs);

[N, Wn] = ellipord(Wp, Ws, Rp, Rs);

IIR Digital Filter Design Using MATLAB

- Example - Determine the minimum order of a Type 2 Chebyshev digital highpass filter with the following specifications:

  $F_p = 1$ kHz, $F_s = 0.6$ kHz, $F_T = 4$ kHz,

  $\alpha_p = 1$ dB, $\alpha_s = 40$ dB

- Here, $\mathrm{Wp} = 2 \times 1/4 = 0.5$, $\mathrm{Ws} = 2 \times 0.6/4 = 0.3$
- Using the statement

```
[N,Wn] = cheb2ord(0.5,0.3,1,40);
```

we get `N = 5` and `Wn = 0.3224`

IIR Digital Filter Design Using MATLAB

- Filter Design -
- For IIR filter design using bilinear transformation, MATLAB statements to use are:

```
[b,a] = butter(N,Wn)
[b,a] = cheby1(N,Rp,Wn)
[b,a] = cheby2(N,Rs,Wn)
[b,a] = ellip(N,Rp,Rs,Wn)
```

IIR Digital Filter Design Using MATLAB

- The form of transfer function obtained is

$$G(z) = \frac{B(z)}{A(z)} = \frac{b(1) + b(2)z^{-1} + \cdots + b(N+1)z^{-N}}{1 + a(2)z^{-1} + \cdots + a(N+1)z^{-N}}$$

- The frequency response can be computed using the M-file `freqz(b,a,w)` where `w` is a set of specified angular frequencies
- It generates a set of complex frequency response samples from which magnitude and/or phase response samples can be computed

IIR Digital Filter Design Using MATLAB

- Example - Design an elliptic IIR lowpass filter with the specifications: $F_p = 0.8$ kHz, $F_s = 1$ kHz, $F_T = 4$ kHz, $\alpha_p = 0.5$dB, $\alpha_s = 40$ dB
- Here, $\omega_p = 2\pi F_p / F_T = 0.4\pi$, $\omega_s = 2\pi F_s / F_T = 0.5\pi$
- Code fragments used are:

```
[N,Wn]=ellipord(0.4,0.5,0.5,40);
[b,a]=ellip(N,0.5,40,Wn);
```

IIR Digital Filter Design Using MATLAB

- Gain response plot is shown below: