EE482/682: DSP APPLICATIONS

CH4 IIR FILTER DESIGN

http://www.ee.unlv.edu/~b1morris/ee482
INTRODUCTION

CHAPTER 4.1
IIR DESIGN

- Reuse well studied analog filter design techniques (books and tables for design)
- Need to map between analog design and a digital design
  - Mapping between s-plane and z-plane
Laplace transform

\[ X(s) = \int_{-\infty}^{\infty} x(t)e^{-st} dt \]

Complex s-plane

\[ s = \sigma + j\Omega \]
- Complex number with \( \sigma \) and \( \Omega \) real
- \( j\Omega \) – imaginary axis

Fourier transform for \( \sigma = 0 \)
- When region of convergence contains the \( j\Omega \) axis

Convolution relationship

\[ y(t) = x(t) \ast h(t) \rightarrow Y(s) = X(s)H(s) \]

\[ H(s) = \frac{Y(s)}{X(s)} = \int_{-\infty}^{\infty} h(t)e^{-st} dt \]

Stability constraint requires poles to be in the left half s-plane
Z-transform from Laplace by change of variable

\[ z = e^{sT} = e^{\sigma T} e^{j\Omega T} = |z|e^{j\omega} \]

\[ |z| = e^{\sigma T}, \ \omega = \Omega T \]

This mapping is not unique

\[ -\pi / T < \Omega < \pi / T \rightarrow \text{unit circle} \]

\[ 2\pi \ \text{multiples as well} \]

Left half s-plane mapped inside unit circle

Right half s-plane mapped outside unit circle

**Figure 4.1** Mapping properties between the s-plane and the z-plane
FILTER CHARACTERISTICS

- Designed to meet a given/desired magnitude response

- Trade-off between:
  - Phase response
  - Roll-off rate – how steep is the transition between pass and stopband (transition width)
BUTTERWORTH FILTER

- All-pole approximation to ideal filter

- $|H(\Omega)|^2 = \frac{1}{1 + (\Omega/\Omega_p)^{2L}}$
  - $|H(0)| = 1$
  - $|H(\Omega_p)| = 1/\sqrt{2}$
    - -3 dB @ $\Omega_p$
  - Has flat magnitude response in pass and stopband (no ripple)

- Slow monotonic transition band
  - Generally needs larger $L$

Figure 4.2  Magnitude response of Butterworth lowpass filter
CHEBYSHEV FILTER

- Steeper roll-off at cutoff frequency than Butterworth
  - Allows certain number of ripples in either passband or stopband
- Type I – equiripple in passband, monotonic in stopband
  - All-pole filter
- Type II – equiripple in stopband, monotonic in passband
  - Poles and zeros
- Generally better magnitude response than Butterworth but at cost of poorer phase response

Figure 4.3 Magnitude responses of Chebyshev type I (top) and type II lowpass filters
ELLIPTIC FILTER

- Sharpest passband to stopband transition
- Equiripple in both pass and stopbands
- Phase response is highly non-linear in passband
  - Should only be used in situations where phase is not important to design

*Figure 4.4* Magnitude response of elliptic lowpass filter
Design lowpass filter and transform from LP to another type (HP, BP, BS)

Define mapping

\[ H(z) = H_{lp}(z)|_{z^{-1} = G(z^{-1})} \]

Replace \( Z^{-1} \) in LP filter with \( G(z^{-1}) \)

\( \theta \) – frequency in LP

\( \omega \) – frequency in new transformed filter

**TABLE 7.1 TRANSFORMATIONS FROM A LOWPASS DIGITAL FILTER Prototype OF CUTOFF FREQUENCY \( \omega_p \) TO HIGHPASS, BANDPASS, AND BANDSTOP FILTERS**

<table>
<thead>
<tr>
<th>Filter Type</th>
<th>Transformations</th>
<th>Associated Design Formulas</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lowpass</td>
<td>( Z^{-1} = \frac{z^{-1} - \alpha}{1 - \alpha z^{-1}} )</td>
<td>( \alpha = \frac{\sin \left( \frac{\theta_p - \theta}{2} \right)}{\sin \left( \frac{\omega_p - \omega}{2} \right)} )</td>
</tr>
<tr>
<td></td>
<td></td>
<td>( \omega_p = \text{desired cutoff frequency} )</td>
</tr>
<tr>
<td>Highpass</td>
<td>( Z^{-1} = \frac{z^{-1} + \alpha}{1 + \alpha z^{-1}} )</td>
<td>( \alpha = -\frac{\cos \left( \frac{\theta_p + \theta}{2} \right)}{\cos \left( \frac{\omega_p + \omega}{2} \right)} )</td>
</tr>
<tr>
<td></td>
<td></td>
<td>( \omega_p = \text{desired cutoff frequency} )</td>
</tr>
<tr>
<td>Bandpass</td>
<td>( Z^{-1} = \frac{z^{-2} - \frac{2k}{k+1} z^{-1} + \frac{k+1}{k+1} z^{-1}}{z^{-2} - \frac{2k}{k+1} z^{-1} - \frac{k+1}{k+1} z^{-1} + 1} )</td>
<td>( \alpha = \frac{\cos \left( \frac{\theta_p + \theta}{2} + \theta_p \right)}{\cos \left( \frac{\omega_p + \omega}{2} + \omega_p \right)} )</td>
</tr>
<tr>
<td></td>
<td>( k = \cos \left( \frac{\theta_p - \theta_p^1}{2} \right) \tan \left( \frac{\theta_p}{2} \right) )</td>
<td></td>
</tr>
<tr>
<td></td>
<td>( \omega_p^1 = \text{desired lower cutoff frequency} )</td>
<td></td>
</tr>
<tr>
<td></td>
<td>( \omega_p^2 = \text{desired upper cutoff frequency} )</td>
<td></td>
</tr>
<tr>
<td>Bandstop</td>
<td>( Z^{-1} = \frac{z^{-2} - \frac{2k}{k+1} z^{-1} + \frac{k+1}{k+1} z^{-1} + 1}{\frac{1}{k+1} z^{-2} - \frac{2k}{k+1} z^{-1} - \frac{k+1}{k+1} z^{-1} + 1} )</td>
<td>( \alpha = \frac{\cos \left( \frac{\theta_p + \theta}{2} + \theta_p \right)}{\cos \left( \frac{\omega_p + \omega}{2} + \omega_p \right)} )</td>
</tr>
<tr>
<td></td>
<td>( k = \tan \left( \frac{\theta_p - \theta_p^1}{2} \right) \tan \left( \frac{\theta_p}{2} \right) )</td>
<td></td>
</tr>
<tr>
<td></td>
<td>( \omega_p^1 = \text{desired lower cutoff frequency} )</td>
<td></td>
</tr>
<tr>
<td></td>
<td>( \omega_p^2 = \text{desired upper cutoff frequency} )</td>
<td></td>
</tr>
</tbody>
</table>
DESIGN OF IIR FILTERS

CHAPTER 4.2
IIR FILTER DESIGN

- IIR transfer function

\[ H(z) = \frac{\sum_{l=0}^{L-1} b_l z^{-l}}{1 + \sum_{l=0}^{M} a_l z^{-l}} \]

- Need to find coefficients \( a_l, b_l \)
  - Impulse invariance – sample impulse response
    - Have to deal with aliasing
  - Bilinear transform
    - Match magnitude response
    - “Warp” frequencies to prevent aliasing
BILINEAR TRANSFORM DESIGN

- Convert digital filter into an “equivalent” analog filter
  - Use bilinear “warping”
- Design analog filter using IIR design techniques
- Map analog filter into digital
  - Use bilinear transform

**Figure 4.5** Digital IIR filter design using the bilinear transform
BILINEAR TRANSFORMATION

- Mapping from s-plane to z-plane
  
  \[ s = \frac{2}{T} \left( \frac{z-1}{z+1} \right) = \frac{2}{T} \left( \frac{1-z^{-1}}{1+z^{-1}} \right) \]

- Frequency mapping
  
  \[ \Omega = \frac{2}{T} \tan \left( \frac{\omega}{2} \right) \]
  
  \[ \omega = 2 \arctan \left( \frac{\Omega T}{2} \right) \]

- Entire \( j\omega \)-axis is squished into \([-\pi/T, \pi/T]\) to prevent aliasing
  
  - Unique mapping
  - Highly non-linear which requires “pre-warp” in design

**Figure 4.6** Frequency warping of bilinear transform defined by (4.27)
BILINEAR DESIGN STEPS

- Convert digital filter into an “equivalent” analog filter
  - Pre-warp using: \( \Omega = \frac{2}{T} \tan \left( \frac{\omega}{2} \right) \)
- Design analog filter using IIR design techniques
  - Butterworth, Chebyshev, Elliptical
- Map analog filter into digital
  - \( H(z) = H(s) \bigg|_{s = \frac{2}{T} \frac{1 - z^{-1}}{1 + z^{-1}}} \)
Example 4.2

Design filter using bilinear transform

- $H(s) = 1/(s + 1)$
- Bandwidth 1000 Hz
- $f_s = 8000$ Hz
- DT parameters
  - $\omega_c = 2\pi(1000/8000) = 0.25\pi$

1. Pre-warp: $\Omega = \frac{2}{T} \tan \left( \frac{\omega}{2} \right)$
   - $\Omega_c = \frac{2}{T} \tan (0.125\pi) = \frac{0.8284}{T}$

2. Scale frequency (normalize scale)
   - $\hat{H}(s) = H \left( \frac{s}{\Omega_c} \right) = \frac{0.8284}{sT + 0.8284}$

3. Bilinear transform:
   - $H(z) = H(s)|_{s=\frac{2}{T}(\frac{1-z^{-1}}{1+z^{-1}})}$
   - $H(z) = \frac{0.2929(1+z^{-1})}{1-0.4141z^{-1}}$
REALIZATION OF IIR FILTERS

CHAPTER 4.3
Different forms or structures can implement an IIR filter

- All are equivalent mathematically (infinite precision)
- Different practical behavior when considering numerical effects

Want structures to minimize error
DIRECT FORM I (DFI)

- Straight-forward implementation of diff. eq.
  - $b_l$ - feed forward coefficients
    - From $x(n)$ terms
  - $a_l$ - feedback coefficients
    - From $y(n)$ terms
- Requires $(L + M)$ coefficients and delays
Notice that we can decompose the transfer function

\[ H(z) = H_1(z)H_2(z) \]

Section to implement zeros and section to implement poles

Can switch order of operations

\[ H(z) = H_2(z)H_1(p) \]

This allows sharing of delays and saving in memory
CASCADE (FACTORED) FORM

- Factor transfer function and decompose into smaller sub-systems
  - \( H(z) = H_1(z)H_2(z) \ldots H_K(z) \)
- Make each subsystem second order
  - Complex conjugate roots have real coefficients
  - Limit the order of subsystem (numerical effects)
    - Effects limited to single subsystem stage
    - Change in a single coefficient affects all poles in DF
- Preferred over DF because of numerical stability

**Figure 4.10** Cascade realization of digital filter
Decompose transfer function using a partial fraction expansion

- \( H(z) = H_1(z) + H_2(z) + ... + H_K(z) \)
- \( H_k(z) = \frac{b_{0k}+b_{1k}z^{-1}}{1+a_{1k}z^{-1}+a_{2k}z^{-2}} \)

Be sure to remember that PFE requires numerator order less than denominator

- Use polynomial long division
DESIGN OF IIR FILTERS USING MATLAB

CHAPTER 4.4
MATLAB FILTER DESIGN

- **Realization tools:**
  - Finding polynomial roots
    - roots.m
    - tf2zp.m
  - Cascade form
    - $H(z) = G \prod_{k=1}^{K} \frac{b_{0k} + b_{1k}z^{-1} + b_{2k}z^{-2}}{1 + a_{1k}z^{-1} + a_{2k}z^{-2}}$
      - zp2sos.m
  - Parallel form
    - Residuez.m

- **Filter design tools:**
  - Order estimation tool
    - butterord.m
  - Coefficient tool
    - butter.m
  - Frequency transforms
    - lp2hp.m, lp2bp.m, lp2bs.m
  - Useful exploration tool
    - fvtool.m
  - Useful design tool
    - fdatool.m
  - Useful processing tool
    - sptool.m
(Causal) IIR filters are stable if all poles are within the unit circle
- $|p_m| < 1$
- We will not consider marginally stable (single pole on unit circle)
Consider poles of 2nd order filter (used in cascade and parallel forms)
- $A(z) = 1 + a_1 z^{-1} + a_2 z^{-2}$
Factor
- $A(z) = (1 - p_1 z^{-1})(1 - p_2 z^{-1})$
- $A(z) = 1 - (p_1 + p_2) z^{-1} + p_1 p_2 z^{-2}$
- Because poles must be inside the unit circle
  - $|a_2| = |p_1 p_2| < 1$
  - $|a_1| < 1 + a_2$

Figure 4.15 Region of coefficient values for the stable second-order IIR filters
Using fixed word lengths results in a quantized approximation of a filter

\[ H'(z) = \frac{\sum_{k=0}^{L-1} b'_k z^{-k}}{1+\sum_{k=1}^{M} a'_k z^{-k}} \]

This can cause a mismatch from desired system \( H(z) \)

Poles that are close to the unit circle may move outside and cause instability

This is exacerbated with higher order systems
ROUNDING EFFECTS

- Using $B$ bit architecture, products require $2B$ bits
  - Must be rounded into smaller $B$ bit container
- This results in noise error terms
  - Can be simply modeled as additive term
- The order of cascade sections influences power of noise at output
  - How should sections be paired and ordered?
- Need to optimize SQNR
  - Trade-off with probability of arithmetic overflow
  - Need to use scaling factors to prevent overflow
  - Optimality when signal level is maximized without overflow
Good results are obtained using simple rules

Cascade ordering and pairing algorithm:

Pair pole closest to unit circle with zero that is closest in z-plane

- Minimize the chance of overflow

Apply 1 repeatedly until all poles and zeros are paired

Resulting 2nd-order sections can be ordered in two alternative ways

- Increasing closeness to unit circle
- Decreasing closeness to unit circle
PRACTICAL APPLICATIONS
CHAPTER 4.6
RECURSIVE RESONATOR

- Filter with frequency response dominated at a single peak
  - Use complex-conjugate pole pair inside unit circle

- $H(z) = \frac{A}{(1-r_pe^{j\omega_0}z^{-1})(1-r_pe^{-j\omega_0}z^{-1})}$

- $H(z) = \frac{A}{1-2r_p \cos(\omega_0)z^{-1}+r_p^2z^{-2}}$
  - $A$ – normalization constant for unity gain at $\omega_0$
  - $0 < r_p < 1$

- Close to unit circle
  - Bandwidth $\cong 2(1-r_p)$
  - Closer to $r_p = 1$, more peaked

Figure 4.17  Signal-flow diagram of the second-order resonator filter
PARAMETRIC EQUALIZER

- Add nearby zeros to the resonator
  - At same angle as poles $\omega_0$ with similar radius
- Pole and zero counter balance one another
- $r_z < r_p$
  - Pole dominates because it is closer to unit circle
  - Generates peak at $\omega = \omega_0 \rightarrow$ Provides boost to freq
- $r_z > r_p$
  - Zero dominates pole
  - Generates dip at $\omega = \omega_0 \rightarrow$ Cuts freq
- Bandwidth still determined by $r_p$

- **Ex 4.18**
  - Create equalizer by changing gain at given frequency