Lecture 17: IIR Filters

ECE 401: Signal and Image Analysis

University of Illinois

4/13/2017

Outline

IIR Filters

$$H(z) = \frac{\prod_{m=1}^{M}(1 - r_m/z)}{\prod_{m=1}^{N}(1 - p_m/z)} = \frac{1 + \sum_{m=1}^{M} b_m z^{-m}}{1 - \sum_{m=1}^{N} a_m z^{-m}}$$

- The IIR filter is **designed** by choosing the poles and zeros.
  - The zeros are M different frequencies $z = r_m = e^{-\beta_m + j\phi_m}$ at which $H(z) = 0$.
  - The poles are N different frequencies $z = p_m = e^{-\alpha_m + j\theta_m}$ at which $H(z) = \infty$.
  - The **center frequency** of a pole (zero), in radians/sample, is $\theta_m = \angle p_m$ ($\phi_m = \angle r_m$).
  - The **bandwidth** of a pole (zero), in radians/sample, is $\alpha_m = \ln|p_m|$ ($\beta_m = \ln|r_m|$).
- The IIR filter is **implemented** using an LCCDE with coefficients a_m and b_m.

Frequency Response of an IIR Filter

$$H(e^{j\omega}) = \frac{\prod_{m=1}^{M}(1 - r_m e^{-j\omega})}{\prod_{m=1}^{N}(1 - p_m e^{-j\omega})} = \frac{\sum_{m=0}^{M} b_m e^{-j\omega m}}{1 - \sum_{m=1}^{N} a_m e^{-j\omega m}}$$

- The magnitude spectrum $|H(\omega)|$, phase spectrum $\angle H(\omega)$, and level spectrum $20\log_{10}|H(\omega)|$ can be calculated in python by just finding the magnitude, phase, and log-magnitude of $H(\omega)$ for different values of ω.
- Usually it's not easy to find $|H(\omega)|$ by hand, except at two or three Very Important Frequencies (the VIFs).

Frequency Response of an IIR Filter

$$H(e^{j\omega}) = \frac{\prod_{m=1}^{M}(1 - r_m e^{-j\omega})}{\prod_{m=1}^{N}(1 - p_m e^{-j\omega})} = \frac{\sum_{m=0}^{M} b_m e^{-j\omega m}}{1 - \sum_{m=1}^{N} a_m e^{-j\omega m}}$$

- Usually it's not easy to find $|H(\omega)|$ by hand, except at two or three Very Important Frequencies (the VIFs).
  - $H(e^{j0})$ is easy to calculate, because $e^{j0} = 1$. So you just plug in $z = 1$ to every z in the formula for $H(z)$.
  - $H(e^{j\pi})$ is easy to calculate, because $e^{j\pi} = -1$. So you just plug in $z = -1$ for every z in $H(z)$.
  - If you know $H(e^{j0})$ and $H(e^{j\pi})$, you can tell whether it's approximately a LPF or HPF. If you want to know if it's a BPF, you need a few more frequencies.
  - At the zero-frequency, $\phi_m = \angle r_m$, there will always be a dip. How much of a dip? It depends on the bandwidth, $\beta_m = \ln |r_m|$.
  - At the pole-frequency, $\theta_m = \angle p_m$, there will always be a peak. How much of a peak? It depends on the bandwidth, $\alpha_m = \ln |p_m|$.

Outline

Impulse Response of a One-Pole Filter

Consider a one-pole IIR Filter:

$$H(z) = \frac{C}{1 - p_m z^{-1}}$$

We already know this one. The impulse response is

$$h[n] = C(p_m)^n u[n]$$

Partial Fraction Expansion

Consider a more complicated IIR filter:

$$H(z) = \frac{\prod_{m=1}^{M}(1 - r_m/z)}{\prod_{m=1}^{N}(1 - p_m/z)}$$

The **partial fraction expansion theorem** (PFE) says that, if $M \le N$, then we can write

$$\frac{\prod_{m=1}^{M}(1 - r_m/z)}{\prod_{m=1}^{N}(1 - p_m/z)} = C_0 + \sum_{m=1}^{N} \frac{C_m}{1 - p_m/z}$$

Where $C_0 = 1$ if $M = N$, otherwise $C_0 = 0$.

Proof of the PFE Theorem

Let's prove this:

$$\frac{\prod_{m=1}^{M}(1-r_m/z)}{\prod_{m=1}^{N}(1-p_m/z)} = C_0 + \sum_{m=1}^{N} \frac{C_m}{1-p_m/z}$$

Multiply both sides of the equation by $\prod_{m=1}^{N}(1-p_m/z)$. That gives:

$$\prod_{m=1}^{M}(1-r_m/z) = C_0\prod_{m=1}^{N}(1-p_m/z) + C_1\prod_{m=2}^{N}(1-p_m/z) + \ldots$$

- Since the coefficients $C_1, \ldots, C_N$ are completely up to us, we can choose them so that the LHS equals the RHS.
- The LHS is a polynomial of order M. All of the terms on the RHS are polynomials of order $N-1$ except the C_0 term, which has order N. Therefore, we only need the C_0 term if $M = N$; if $M < N$, we can set $C_0 = 0$.

Finding the Coefficients

To find the k^{th} coefficient, C_k, we start with this equation:

$$\frac{\prod_{m=1}^{M}(1-r_m/z)}{\prod_{m=1}^{N}(1-p_m/z)} = C_0 + \sum_{m=1}^{N}\frac{C_m}{1-p_m/z}$$

Multiply both sides by $(1-p_k/z)$. That gives:

$$\frac{\prod_{m=1}^{M}(1-r_m/z)}{\prod_{m\neq k}(1-p_m/z)} = C_0(1-p_k/z) + C_k + \sum_{m\neq k} C_m \frac{1-p_k/z}{1-p_m/z}$$

Then if we evaluate at $z=p_k$, all terms on the RHS disappear except the C_k term:

$$\left.\frac{\prod_{m=1}^{M}(1-r_m/z)}{\prod_{m\neq k}(1-p_m/z)}\right|_{z=p_k} = C_k$$

Partial Fraction Expansion Example

Suppose

$$H(z) = \frac{(1-(0.9)e^{-j2\pi/3}/z)(1-(0.9)e^{j2\pi/3}/z)}{(1-(0.9)e^{-j\pi/3}/z)(1-(0.9)e^{j\pi/3}/z)}$$

The PFE theorem tells us that

$$H(z) = 1 + \frac{C_1}{1-(0.9)e^{-j\pi/3}/z} + \frac{C_1}{1-(0.9)e^{j\pi/3}/z}$$

where

$$C_1 = \left.\frac{(1-(0.9)e^{-j2\pi/3}/z)(1-(0.9)e^{j2\pi/3}/z)}{(1-0.9e^{j\pi/3})/z}\right|_{z=0.9e^{-j\pi/3}}$$

$$C_1 = \frac{(1+j)(1+1)}{(1-j)} = 2j$$

Partial Fraction Expansion Example

$$H(z) = \frac{(1-(0.9)e^{-j2\pi/3}/z)(1-(0.9)e^{j2\pi/3}/z)}{(1-(0.9)e^{-j\pi/3}/z)(1-(0.9)e^{j\pi/3}/z)}$$

$$C_2 = \left. \frac{(1-0.9e^{-j2\pi/3}/z)(1-0.9e^{j2\pi/3}/z)}{(1-0.9e^{-j\pi/3})/z} \right|_{z=0.9e^{j\pi/3}}$$

$$C_2 = \frac{(1+1)(1-j)}{(1+j)} = -2j$$

So

$$H(z) = 1 + \frac{2j}{1-0.9e^{-j\pi/3}/z} - \frac{2j}{1-0.9e^{j\pi/3}/z}$$

Impulse Response Example

Suppose

$$H(z) = \frac{(1-0.9e^{-j2\pi/3}/z)(1-0.9e^{j2\pi/3}/z)}{(1-0.9e^{-j\pi/3}/z)(1-0.9e^{j\pi/3}/z)}$$

$$H(z) = 1 + \frac{2j}{1-0.9e^{-j\pi/3}/z} - \frac{2j}{1-0.9e^{j\pi/3}/z}$$

So

$$h[n] = \delta[n] + 2j(0.9)^n e^{-j\pi n/3}u[n] - 2j(0.9)^n e^{j\pi n/3}u[n]$$

Combining the $p_1 = 0.9e^{-j\pi/3}$ and $p_2 = 0.9e^{j\pi/3}$ terms, we get

$$h[n] = \delta[n] - (0.9)^n \sin\left(\frac{\pi n}{3}\right) u[n]$$

Impulse Response of an IIR Filter

Consider a more complicated IIR filter:

$$H(z) = \frac{\prod_{m=1}^{M}(1 - r_m/z)}{\prod_{m=1}^{N}(1 - p_m/z)} = C_0 + \sum_{m=1}^{N} \frac{C_m}{1 - p_m/z}$$

The impulse response is

$$h[n] = C_0\delta[n] + \sum_{m=1}^{N} C_m (p_m)^n u[n]$$

Notice that $h[n]$ is real if and only if the poles come in complex-conjugate pairs. Let's assume that this is the case, therefore we have only two possibilities for each pole:

- p_m might be real-valued, in which case C_m is also real-valued, or
- p_m and p_{m+1} might be a complex conjugage pair, $p_{m+1} = p_m^*$, meaning that $|p_{m+1}| = |p_m|$, and $\angle p_{m+1} = -\angle p_m$. In this case, it will always turn out that $C_{m+1} = C_m^*$.

Outline

Implementation: Direct Form

Given r_m and p_m, we need to find the LCCDE coefficients a_m and b_m. We do this by multiplying out the polynomials:

$$\prod_{m=1}^{M}(1 - r_m z^{-1}) = 1 + \sum_{m=1}^{M} b_m z^{-m}$$

$$\prod_{m=1}^{N}(1 - p_m z^{-1}) = 1 - \sum_{m=1}^{N} a_m z^{-m}$$

Then we implement it in direct form as

$$y[n] = x[n] + \sum_{m=1}^{M} b_m x[n-m] + \sum_{m=1}^{N} a_m y[n-m]$$

Direct Form Example

Let's use the same example:

$$(1-0.9e^{-j2\pi/3}z^{-1})(1-0.9e^{j2\pi/3}z^{-1})$$
$$=1-2(0.9)\cos\left(\frac{2\pi}{3}\right)z^{-1}+(0.9)^2z^{-2}$$
$$=1+0.9z^{-1}+0.81z^{-2}$$

$$(1-0.9e^{-j\pi/3}z^{-1})(1-0.9e^{j\pi/3}z^{-1})$$
$$=1-2(0.9)\cos\left(\frac{\pi}{6}\right)z^{-1}+(0.9)^2z^{-2}$$
$$=1-0.9z^{-1}+0.81z^{-2}$$

So we implement it in direct form as

$$y[n]=x[n]+0.9x[n-1]+0.81x[n-2]+0.9y[n-1]-0.81y[n-2]$$

Serial Form

The problem with direct form is that, if N is very large, we have to multiply out a very long polynomial. The resulting floating-point roundoff error sometimes causes the implemented filter to be very different from the filter we want. The solution is to divide the filter into order-2 subfilters:

$$H(z) = H_1(z)H_2(z)\dots H_{\frac{N}{2}}(z)$$

This corresponds to filtering the signal with one filter after another, like this:

$$\begin{aligned} V_1(z) &= H_1(z)X(z) \\ V_2(z) &= H_2(z)V_1(z) \\ &\vdots \\ Y(z) &= H_{\frac{N}{2}}(z)V_{\frac{N}{2}-1}(z) \end{aligned}$$

The reason we choose to group the poles into groups of two is so that we can have real-valued filter coefficients.

$$H_k(z) = \frac{(1 - r_{2k}z^{-1})(1 - r_{2k+1}z^{-1})}{(1 - p_{2k}z^{-1})(1 - p_{2k+1}z^{-1})}$$

If $|r_{2k+1}| = |r_{2k}|$, $\angle r_{2k+1} = -\angle r_{2k}$, $|p_{2k+1}| = |p_{2k}|$, and $\angle p_{2k+1} = -\angle p_{2k}$, then

$$H_k(z) = \frac{1 - 2|r_{2k}|\cos\left(\angle r_{2k}\right) z^{-1} + |r_{2k}|^2 z^{-2}}{1 - 2|p_{2k}|\cos\left(\angle p_{2k}\right) z^{-1} + |p_{2k}|^2 z^{-2}}$$

...so all the filter coefficients are real numbers.

Series Example

Suppose we want to implement a 4th-order filter, with

- Zeros: $r_1 = 0.9e^{j2\pi/3}$, $r_2 = 0.9e^{-j2\pi/3}$, $r_3 = 0.9e^{j3\pi/4}$, $r_4 = 0.9e^{-j3\pi/4}$.
- Poles: $p_1 = 0.9e^{j\pi/3}$, $p_2 = 0.9e^{-j\pi/3}$, $p_3 = 0.9e^{j\pi/4}$, $p_4 = 0.9e^{-j\pi/4}$.

We can group them into two groups:

$$H_1(z) = \frac{(1-0.9e^{j2\pi/3}z^{-1})(1-0.9e^{-j2\pi/3}z^{-1})}{(1-0.9e^{j\pi/3}z^{-1})(1-0.9e^{-j\pi/3}z^{-1})}$$

$$H_2(z) = \frac{(1-0.9e^{j3\pi/4}z^{-1})(1-0.9e^{-j3\pi/4}z^{-1})}{(1-0.9e^{j\pi/4}z^{-1})(1-0.9e^{-j\pi/4}z^{-1})}$$

Series Example

$$H_1(z) = \frac{(1-0.9e^{j2\pi/3}z^{-1})(1-0.9e^{-j2\pi/3}z^{-1})}{(1-0.9e^{j\pi/3}z^{-1})(1-0.9e^{-j\pi/3}z^{-1})}$$

$$H_2(z) = \frac{(1-0.9e^{j3\pi/4}z^{-1})(1-0.9e^{-j3\pi/4}z^{-1})}{(1-0.9e^{j\pi/4}z^{-1})(1-0.9e^{-j\pi/4}z^{-1})}$$

Multiplying them out, we get

$$H_1(z) = \frac{1+0.9z^{-1}+0.81z^{-1}}{1-0.9z^{-1}+0.81z^{-1}}$$

$$H_2(z) = \frac{1+0.9\sqrt{2}z^{-1}+0.81z^{-1}}{1-0.9\sqrt{2}z^{-1}+0.81z^{-1}}$$

Series Example

$$H_1(z) = \frac{1 + 0.9z^{-1} + 0.81z^{-1}}{1 - 0.9z^{-1} + 0.81z^{-1}}$$

$$H_2(z) = \frac{1 + 0.9\sqrt{2}z^{-1} + 0.81z^{-1}}{1 - 0.9\sqrt{2}z^{-1} + 0.81z^{-1}}$$

We can implement the series connection of these two filters as

$$v[n] = x[n] + 0.9x[n-1] + 0.81x[n-2] + 0.9v[n-1] - 0.81v[n-2]$$

$$y[n] = v[n] + 0.9\sqrt{2}v[n-1] + 0.81v[n-2] + 0.9\sqrt{2}y[n-1] - 0.81y[n-2]$$

Outline

Filter Design Methods

1. **Notch Filter:** to cancel narrowband noise at frequency ω_c, you set the zeros to $r_2 = r_1^* = e^{-j\omega_c}$, and set the poles to $p_2 = p_1^* = ae^{-j\omega_c}$, for some real number $a < 1$.
2. **LPC (Linear Predictive Coding):** We can model a resonant production mechanism, like human speech, musical instruments, ventilation ducts, and so on, by finding the resonant frequencies and bandwidths of the system, and by setting p_m to match.
3. **Analog Filter Design Methods:** Low-pass, high-pass, and bandpass filters can be designed using old-fashioned filter design methods, then converted from analog to digital. There are programs that can do this for you. The three most common methods are **Butterworth** (no ripple, wide transition band; this is usually the one you want), **Elliptical** (smallest transition band, but lots of ripple), **Chebyshev** (intermediate between the others).