Chapter 7

IIR Digital Filter Design

We have focused our discussion so far on FIR or nonrecursive filter design. There are, however, some disadvantages in using nonrecursive filters:

(i) Get ripples. Usually need long filters to suppress them.

(ii) Must have smooth transitions since it is a sum of smooth functions. Difficult to get sharp transitions.

(iii) For real-time operation, it is necessary to introduce a delay approximately equal to half of the filter length (see P.S. #6). If the filter length is long, this may not be acceptable.

In this chapter, we study the use of recursive filters of the form

$$y_n = \sum_{k=1}^{N} a_k y_{n-k} + \sum_{k=0}^{M} b_k u_{n-k}$$

(Note: this is causal since responds to past values. Noncausal recursive filters are possible but almost never used).

Since frequency response is

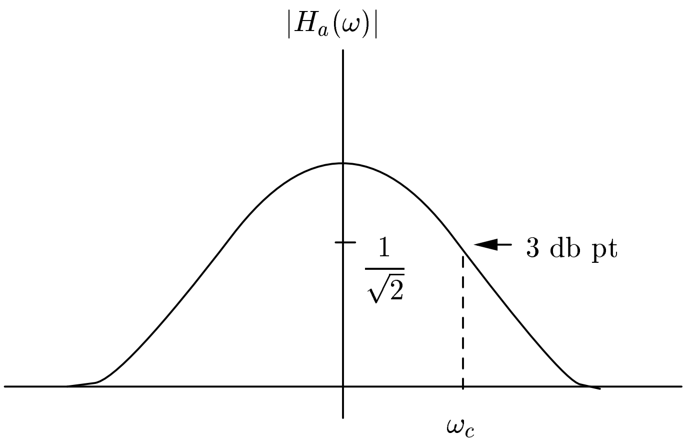
$$\tilde{H}(e^{i2\pi f}) = \frac{\displaystyle\sum_{k=0}^{M} b_k e^{-2\pi i k f}}{1 - \displaystyle\sum_{k=1}^{N} a_k e^{-2\pi i k f}}$$

can get narrow transitions as denominator comes near zero. Also recursive filters generally have infinite memory, and may thus need fewer coefficients.

Fourier approach is clearly not applicable. How do we go about determining a_k, b_k? We proceed indirectly and take advantage of known results from analog filter design. It is known that the following class of filters has properties that allow them to give good l.p. filters.

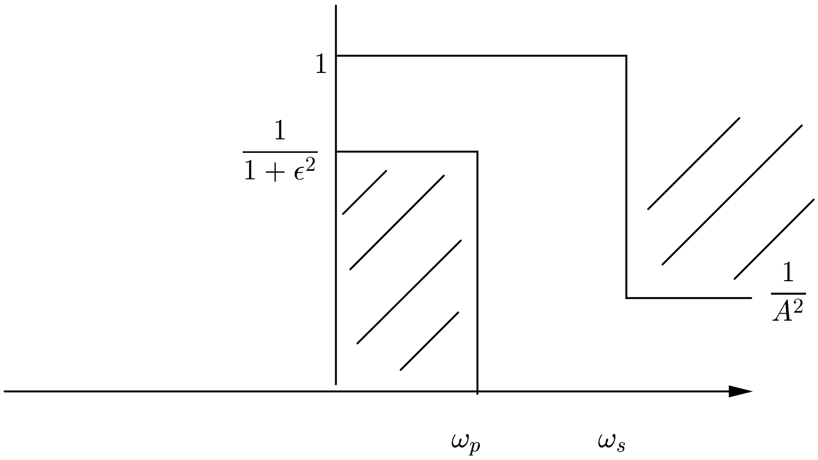
7.1 Butterworth filters

$$H_a(\omega)H_a(-\omega) = \frac{1}{1 + \left(\dfrac{\omega}{\omega_c}\right)^{2N}}$$



Filter parameters are ω_c and N. Butterworth filters are also called maximally-flat filters as their amplitude characteristics have the largest number of derivatives with respect to ω vanishing at 0.

Suppose we want to design an analog l.p. Butterworth filter



Then

$$\frac{1}{1+\varepsilon^2} = \frac{1}{1+\left(\dfrac{\omega_p}{\omega_c}\right)^{2N}} \Rightarrow \left(\frac{\omega_p}{\omega_c}\right)^{2N} = \varepsilon^2$$

and

$$\frac{1}{A^2} = \frac{1}{1+\left(\dfrac{\omega_s}{\omega_c}\right)^{2N}} \Rightarrow \left(\frac{\omega_s}{\omega_c}\right)^{2N} = A^2 - 1$$

$$\Rightarrow \left(\frac{\omega_p}{\omega_s}\right)^{2N} = \frac{\varepsilon^2}{A^2-1}$$

Solving this equation for N will generally not yield an integer. Since N is an integer, we use the ceil function where $ceil(x)$ is the smallest integer $\geq x$.

$$N = \mathrm{ceil}\left(\frac{\log \dfrac{\varepsilon}{\sqrt{A^2-1}}}{\log \dfrac{\omega_p}{\omega_s}}\right)$$

For the critical frequency ω_c, there are two choices:

$$\omega_c = \frac{\omega_p}{\left(\varepsilon^{1/N}\right)}$$

or

$$\omega_c = \frac{\omega_s}{(A^2-1)^{1/2N}}$$

As shown in P.S. #8, the first choice gives a filter which meets the passband specs exactly, and exceeds the stopband specs, which the second choice gives a filter which meets the stopband specs exactly, and exceeds the passband specs.

Once N and ω_c are found, we have determined $H_a(i\omega)H_a(-i\omega)$. Since the analog filter has transfer function $H(s)$, we still need to obtain $H(s)$ from $H_a(i\omega)H_a(-i\omega)$ to complete the analog filter design.

Example:

$$\omega_p = 10^4$$
$$\omega_s = 10^5$$
$$\varepsilon^2 = \frac{1}{999} \Rightarrow \frac{1}{1+\varepsilon^2} = \frac{999}{1000}$$
$$A^2 = 10^3$$

Since N is an integer, we take

$$N = ceil\left(\frac{1}{\log 10^{-1}} \log\left(\frac{\frac{1}{999}}{999}\right)^{1/2}\right) = ceil(\log 999) = 3$$

Two choices for the critical frequency ω_c:

$$\begin{aligned} \omega_{c1} &= \frac{\omega_p}{(\varepsilon^2)^{1/2N}} \\ &= \frac{10^4}{(0.999)^{\frac{1}{6}}} = 1.00017 \times 10^4 \end{aligned}$$

or

$$\begin{aligned} \omega_{c2} &= \frac{\omega_s}{(A^2-1)^{1/2N}} \\ &= \frac{10^5}{(999)^{\frac{1}{6}}} = 3.1628 \times 10^4 \end{aligned}$$

As mentioned before, using ω_{c1} gives a filter whose passband specs are exactly met and stopband specs exceeded, while using ω_{c2} gives a filter whose stopband specs are exactly met and passband specs exceeded.

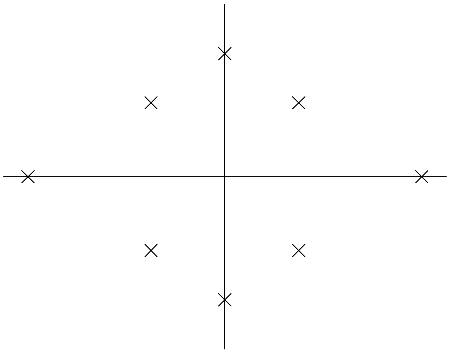
7.2 Factorization

First we substitute s for $i\omega$ to get $H(s)H(-s)$ from $H(i\omega)H(-i\omega)$. For fixed ω_c and N, we then have

$$H(s)H(-s) = \frac{1}{1+\left(\frac{s}{i\omega_c}\right)^{2N}}$$

The factorization problem is to factor the above product to get a stable $H(s)$.

First, let us note the following property of the distribution of poles of $H(s)H(-s)$: If $s = \sigma + i\omega$ is a pole, then $-s = -(\sigma + i\omega)$ is also a pole. Furthermore, since $H(s)$ is a rational function with real coefficients, $s^* = \sigma - i\omega$ is also a pole. But then $-s^* = -\sigma + i\omega$ is also a pole. This shows that complex poles of $H(s)H(-s)$ with nonzero real and imaginary parts must occur in a 4-pole pattern symmetric with respect to the origin. Real poles occur in pairs symmetric with respect to the imaginary axis while purely imaginary poles occur in pairs symmetric with respect to the real axis.



Pole Patterns for $H(s)H(-s)$

Poles of $H(s)H(-s)$ are at

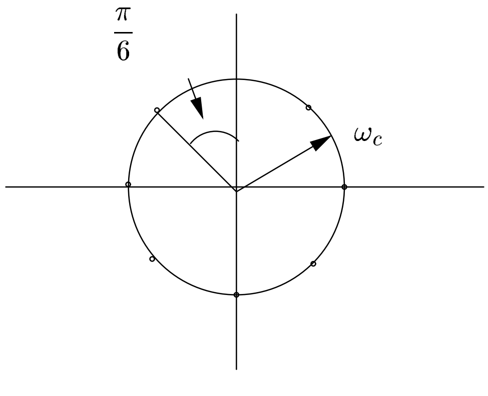
$$s = (-1)^{1/2N}(i\omega_c)$$

$$= \omega_c e^{i\left(\frac{\pi}{2} + \frac{(2k+1)}{2N}\pi\right)} \qquad k = 0, \ldots, 2N-1$$

(i) No pole on imaginary axis, since this would require $\frac{(2k+1)}{N}$ to be even (so that $\frac{2k+1}{2N}$ is an integer), impossible.

(ii) One on real axis for N odd (corresponds to $k = \frac{N-1}{2}$, which is an integer for N odd), but not for N even ($\frac{2k+1}{N}$ cannot be an integer if N even).

For example $N = 3$, 1^{st} pole at $e^{\frac{\pi}{2}i + i\frac{\pi}{6}}$. It is easy to see that the poles corresponding to $k = 0, 1, \ldots, N-1$ are poles located in the left half plane. In fact, once the poles in the second quadrant are determined, the rest of the stable poles can be determined through symmetry.



For this example, the stable poles are $\omega_c(-0.5 + i0.866)$, $-\omega_c$, and $\omega_c(-0.5 - i0.866)$.

Once the poles of $H(s)$ have been found, the complete transfer function can be written as

$$H(s) = \prod_{j=1}^{N} \frac{1}{1 + \frac{s}{s_j}}$$

where $-s_j$, $j = 1, \ldots, N$ are the N stable poles of $H(s)$. Later we shall describe some shortcuts in writing down the complete transfer function $H(s)$ using the symmetry properties of the poles.

Now that the analog filter has been found to satisfy the lowpass filter specs, the next task is to translate the analog design into a digital filter which satisfies the digital lowpass filter specs. As in analog filters, we require digital IIR filters to be stable.

Stability of recursive filters:
Equivalent to stability of difference equation

$$y_n = \sum_{k=1}^{N} a_k y_{n-k} + \sum_{k=0}^{M} b_k u_{n-k}$$

Readily seen that if the impulse response is h_k then system stable (BIBO) $\Leftrightarrow$

$$\sum_{k=0}^{\infty} |h_k| < \infty$$

$\Leftrightarrow$ $H(z)$ has no poles on or outside the unit circle.
Now recursive filter has transfer function

$$H(z) = \frac{z^{N-M}(b_0 z^M + b_1 z^{M-1} + \ldots + b_M)}{z^N - a_1 z^{N-1} - \ldots - a_N}$$

Hence it is stable $\Leftrightarrow$ the polynomial $z^N - a_1 z^{N-1} - \ldots - a_N$ has no roots on or outside the unit circle.
Since the analog Butterworth filter design gives a stable analog filter, it is important to ensure that going from analog to digital gives stable recursive digital filters.

7.3 Getting recursive digital filters from analog designs

Consider the mapping

$$z = \frac{1+s}{1-s} \quad \text{or} \quad s = \frac{z-1}{z+1}$$

This mapping is called the bilinear transformation.
For

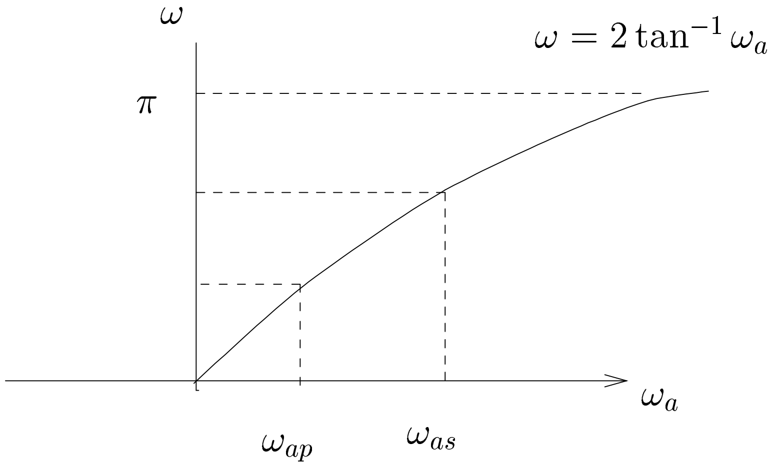
$$\begin{aligned} s &= \sigma_a + i\omega_a \\ z &= \frac{1+\sigma_a+i\omega_a}{1-\sigma_a-i\omega_a} \end{aligned}$$

If

$$\begin{array}{lll} \sigma_a < 0 & |z| < 1 & \qquad \sigma_a > 0 \Rightarrow |z| > 1 \\ \sigma_a = 0 & |z| = 1 & \end{array}$$

This means that the open left half plane for s is mapped onto the open unit disk for z, the imaginary axis for s is mapped onto the unit circle for z, and the open right half plane for s is mapped onto the exterior of the unit circle for z. This means that the bilinear transformation preserves stability.
What happens to frequencies? If we denote the digital frequency by ω

$$\begin{aligned} i\omega_a &= \frac{e^{i\omega}-1}{e^{i\omega}+1} = \frac{i\sin\frac{\omega}{2}}{\cos\frac{\omega}{2}} \\ \Rightarrow \omega_a &= \tan\frac{\omega}{2} \qquad \text{or} \quad 2\tan^{-1}\omega_a = \omega \end{aligned}$$



The compression of the analog frequency ω_a due to the bilinear transformation is called frequency warping. Knowing that we are going to use the bilinear transformation allows us to pre-compensate for the effects of frequency warping. This procedure is called frequency prewarping.

Given digital frequency specs with passband frequency ω_p and stopband frequency ω_s, translate these specs into those for the analog filter using

$$\left.\begin{array}{lll} \omega_s & \text{to} & \omega_{as} = \tan\dfrac{\omega_s}{2} \\ \omega_p & \text{to} & \omega_{ap} = \tan\dfrac{\omega_p}{2} \end{array}\right\} \text{prewarping}$$

By attaining specs for $H_a\left(\tan\dfrac{\omega_s}{2}\right)$ etc. for the analog filter, the bilinear transformation will then yield a digital filter that will attain the specs for $H_d(e^{i\omega})$.

Example: Design a digital IIR filter satisfying the following specs:

$$0.89125 \le |H(e^{i\omega})| \le 1 \quad 0 \le \omega \le 0.2\pi \tag{7.1}$$

$$|H(e^{i\omega})| \le 0.17783 \quad 0.3\pi \le \omega \le \pi \tag{7.2}$$

Step 1. Prewarp the digital filter specs to the following analog filter specs: $\omega_{ap} = \tan\frac{0.2\pi}{2} = 0.3249$, and $\omega_{as} = \tan\frac{0.3\pi}{2} = 0.5095$. The analog filter specs corresponding to 7.1 and 7.2 become

$$0.89125 \le |H(i\omega_a)| \le 1 \quad 0 \le \omega_a \le 0.3249 \tag{7.3}$$

$$|H(i\omega_a)| \le 0.17783 \quad 0.5095 \le \omega_a < \infty \tag{7.4}$$

Step 2. Determine the analog Butterworth filter parameters N and ω_c:

$$\frac{1}{1+\left(\dfrac{\omega_{ap}}{\omega_c}\right)^{2N}} = 0.89125^2 = 0.7943$$

$$\frac{1}{1+\left(\dfrac{\omega_{as}}{\omega_c}\right)^{2N}} = 0.17783^2 = 0.03162$$

Combining these equations gives

$$\left(\frac{\omega_{as}}{\omega_{ap}}\right)^{2N} = \frac{\dfrac{1}{0.03162}-1}{\dfrac{1}{0.7943}-1} = 118.265$$

Therefore

$$N = ceil\left(\frac{\log(118.265)}{2\log(\frac{0.5095}{0.3249})}\right) = ceil(5.30428) = 6$$

Of the 2 choices for ω_c, for this example we take

$$\omega_c = \frac{0.5095}{\frac{1}{0.17783^2} - 1} = \frac{0.5095}{1.329925} = 0.383115$$

Step 3. Factorize $H(s)H(-s)$: The poles in the second quadrant are located at these angles on the circle of radius ω_c

$$(\frac{\pi}{2} + \frac{\pi}{12}), \quad (\frac{\pi}{2} + \frac{3\pi}{12}), \quad (\frac{\pi}{2} + \frac{5\pi}{12})$$

Their real parts are given by -0.09916, -0.2709, and -0.37, respectively. They are paired with their complex conjugates to make up the 6 poles needed for $H(s)$.

Step 4. Now note that for every complex number s_1 lying on the circle of radius ω_c, we can express the product

$$\begin{aligned} \frac{1}{1+\frac{s}{s_1}} \frac{1}{1+\frac{s}{s_1^*}} &= \frac{|s_1|^2}{s^2 + 2\,Re(s_1)s + |s_1|^2} \\ &= \frac{\omega_c^2}{s^2 + 2\,Re(s_1)s + \omega_c^2} \end{aligned}$$

where Re denotes the real part. Thus knowing the real part of the complex conjugate pair of poles will enable us to write down the quadratic factor immediately. In the above product, the poles are identified as $-s_1$ and $-s_1^*$. Thus, if $-s_1$ is in the second quadrant, $Re(s_1) > 0$. We can now write down the complete analog filter transfer function

$$H(s) = \frac{\omega_c^6}{(s^2 + 0.198315s + \omega_c^2)(s^2 + 0.54181s + \omega_c^2)(s^2 + 0.7401s + \omega_c^2)}$$

Step 5. Apply the bilinear transformation to get the digital filter: We take the bilinear transformation in the form

$$s = \frac{1 - z^{-1}}{1 + z^{-1}}$$

It is then easy to verify that an analog quadratic factor transforms in the following manner:

$$\frac{b}{s^2 + as + b} \to \frac{b(1 + z^{-1})^2}{(1 + a + b) + (2b - 2)z^{-1} + (1 - a + b)z^{-2}} \tag{7.5}$$

while a linear factor transforms in the following manner:

$$\frac{\omega_c}{s + \omega_c} \to \frac{\omega_c(1 + z^{-1})}{1 + \omega_c + (\omega_c - 1)z^{-1}} \tag{7.6}$$

For this example, we can then write down the IIR digital filter transfer function

$$H(z) = \frac{0.003162(1 + z^{-1})^6}{(1.3451 - 1.7064z^{-1} + 0.94846z^{-2})(1.6886 - 1.7064z^{-1} + 0.60497z^{-2})(1.8869 - 1.7064z^{-1} + 0.40665z^{-2})}$$

It is customary to normalize the denominator polynomial to have leading coefficient 1, which may be achieved by normalizing each quadratic factor. This gives

$$H(z) = \frac{0.00073783(1 + z^{-1})^6}{(1 - 1.2686z^{-1} + 0.7051z^{-2})(1 - 1.0106z^{-1} + 0.3583z^{-2})(1 - 0.9044z^{-1} + 0.2155z^{-2})}$$

This completes the digital lowpass Butterworth filter design procedure.

For implementation of the digital filter, we do not need to multiply out the denominator to get the sixth degree polynomial. It is usually implemented as a cascade of quadratic sections, with the output from the first section acting as input to the second quadratic section, etc.

So far we have only discussed Butterworth filters in the design of lowpass filters. Other families include Chebyshev filters I and II, and elliptic filters. Their descriptions are beyond the scope of this course.

7.4 Optional Material: Frequency transformations

To be added later!