Elec4621:
Advanced Digital Signal Processing
Chapter 5: Digital Filter Design

Dr. D. S. Taubman

April 1, 2011

Over the years many different strategies have been developed to design digital filters. In this chapter, we will cover some of these strategies and hint at others, but we note from the outset that many modern filters are not directly designed to match a set of specifications, but instead adapt themselves to the signals at hand. Adaptive filters will be considered separately as the second major segment of this subject.

While filter design strategies vary from the simple to the complex, they also vary in regard to the nature of the problem. It is most important that you appreciate the distinction between design strategies in terms of their different design criteria. In some cases, the aim is to minimize the maximum deviation from an ideal frequency response characteristic. In other cases, the aim is to minimize the mean square deviation from an ideal frequency response characteristic. In yet other cases, the aim might be to match some initial portion of the impulse response as accurately as possible, and so forth.

We begin by considering FIR filter designs and then move on to IIR filters. We will conclude with some summary remarks concerning the relative merits of FIR vs. IIR filters.

1 FIR Filter Design

Recall that an M^{th} order FIR filter has $M+1$ coefficients (filter taps), denoted a_0 through a_M. Its transfer function is given by the M^{th} order polynomial (in Z^{-1})

$$H(z) = a_0 + a_1 z^{-1} + \cdots + a_M z^{-M}$$

which is often written as a rational function of z,

$$H(z) = \frac{a_0 z^M + a_1 z^{M-1} + \cdots + a_M}{z^M}$$

As a rational function in z, the transfer function has M non-trivial zeros and M trivial poles. The impulse response is

$$h[n] = \begin{cases} a_n & 0 \le n \le M \\ 0 & \text{otherwise} \end{cases}$$

Recall also that an FIR filter has linear phase if its impulse response is symmetric

$$h[n] = h[M-n]$$

In the Z-transform domain, this means that

$$H(z) = z^{-M} H\left(z^{-1}\right)$$

and so the zeros occur in reciprocal pairs. The definition of linear phase is often broadened to include all filters whose zeros occur in reciprocal pairs. Such filter satisfy

$$H(z) = \pm z^{-M} H\left(z^{-1}\right)$$

and include both symmetric impulse responses and anti-symmetric impulse responses,

$$h[n] = -h[M-n]$$

1.1 Summary of Strategies

Leaving adaptive filters aside, the more common FIR filter design techniques are:

Windowing: The objective here is to design a filter with a desired frequency response, $\hat{h}_d(\omega)$. Since the corresponding impulse response, $h_d[n]$, is generally infinite in extent, a finite support window is applied to $h_d[n]$ to get $h[n] = w[n] \cdot h_d[n]$. Since $w[n]$ is non-zero only over the interval, $0 \le n \le M$, $h[n]$ is an order M FIR filter. The properties of $w[n]$ are adjusted in a somewhat ad-hoc manner so as to minimize undesirable artefacts of the approximation. While not generally optimal in any sense, window-based filter design is simple and commonly used in practice. We discuss windowing further in Section 1.2.

Frequency Sampling: The idea behind frequency sampling is to design a filter, $h[n]$, whose frequency response is identical to a desired frequency response, $\hat{h}_d(\omega)$, at a collection of $M+1$ matching frequencies, ω_0 through ω_M. Since there are $M+1$ unknown filter coefficients and $M+1$ equations (the equations are linear and hence easily solved), the problem generally has a unique solution. However, the frequency response in between the matching frequencies is uncontrolled. If matching frequencies are of the form $\omega_k = \frac{2\pi k}{M+1} \bmod 2\pi$, then $\hat{h}(\omega_k)$ is nothing other than the k^{th} DFT

coefficient in the $M+1$-point DFT of $h[n]$. In this case, $h[n]$ is essentially the inverse DFT of the desired frequency response. In practice, the matching frequencies are either of this form, or else of the form

$$\omega_k = \frac{2\pi\left(k+\frac{1}{2}\right)}{M+1} \bmod 2\pi$$

in which case an equally simple solution exists for $h[n]$.

As a design technique, frequency sampling has relatively little to offer, and we shall not discuss it further here. We note, however, that the use of uniformly spaced matching frequencies, as described above, leads to an interesting implementation structure which can be particularly efficient if $\hat{h}_d(\omega)$ is 0 at most of the matching frequencies, ω_k. The interested reader is referred to Proakis and Salehi, §7.2.3.

Equi-ripple Designs: In the general case, equi-ripple designs aim to minimize the weighted maximum deviation between a desired frequency response, $\hat{h}_d(\omega)$, and the actual frequency response, $\hat{h}(\omega)$, of the FIR filter. More specifically, an optimization algorithm searches for the filter taps which minimize

$$R_{\max} = \max_{0\leq\omega\leq\pi} |\hat{\rho}(\omega)| \cdot \left| \left|\hat{h}_d(\omega)\right| - \left|\hat{h}(\omega)\right| \right| \tag{1}$$

The weighting (or prioritization) function, $\hat{\rho}(\omega)$, serves to emphasize the accuracy at some frequencies relative to others.

In 1972, Parks and McClellan described an optimal solution to this problem for the specific case of linear phase FIR filters. The design strategy is named after them and its solution has the property that the weighted deviation (or ripple), $|\hat{\rho}(\omega)|\left(\left|\hat{h}_d(\omega)\right| - \left|\hat{h}(\omega)\right|\right)$, alternates between $\pm R_{\max}$. The number of alternations is on the order of (actually, slightly more than) half the filter length, depending on whether M is odd or even and whether the filter impulse response is to be symmetric or anti-symmetric.

The Parks-McClellan algorithm has become a standard design technique for linear phase filters. It is most commonly applied to problems in which a filter is specified in terms of its passband frequency, ω_p, stopband frequency, ω_s, and its passband and stopband ripple, δ_p and δ_s. Figure 1 gives an example of such specifications for a low-pass filter design. Setting

$$\hat{\rho}(\omega) = \begin{cases} 1 & 0 \leq \omega \leq \omega_p \\ 0 & \omega_p < \omega < \omega_s \\ \delta_p/\delta_s & \omega_s \leq \omega \leq \pi \end{cases}$$

an equi-ripple design for which $R_{\max} = \delta_p$ is guaranteed to satisfy the constraints precisely. The design equi-ripple filters of increasing order, M, until $R_{\max}$ falls below δ_p. No other design can produce a lower order linear phase filter which satisfies the design constraints.

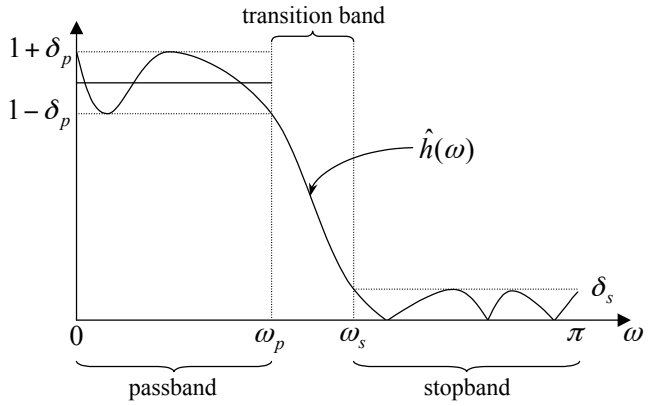

Figure 1: *Typical tolerance specifications used in equi-ripple filter design. Note that the Parks-McClellan algorithm offers a great deal more generality than is reflected through specifications such as these.*

Although equi-ripple designs are of great interest, there is relatively little value in developing the theory here. In most cases, you will simply use existing implementations of the design procedure. The key thing is that you understand what the optimization procedure achieves.

Least-Squares Techniques: While the Parks-McClellan algorithm can be said to yield an optimal design, it is only optimal subject to the linear phase constraint and also subject to the specific design objective, expressed in equation (1). Least-squares designs are optimal with respect to a different objective. Specifically, the filter is sought which minimizes the weighted mean squared deviation,

$$\int_{-\pi}^{\pi} \left|\hat{\rho}\left(\omega\right)\right|^2 \cdot \left|\hat{h}_d\left(\omega\right) - \hat{h}\left(\omega\right)\right|^2 d\omega$$

Again, $\hat{\rho}\left(\omega\right)$ can be set to 0 at frequencies where deviation from the target frequency response is unimportant. We have selected least-squares methods for a more thorough discussion in Section 1.3, in part due to their close connection with adaptive filtering techniques to be explored later in the subject.

1.2 Windowing Method

As discussed briefly above, the window design method involves the selection of a window function, $w\left(\omega\right)$, such that

$$h\left[n\right] = h_d\left[n\right] \cdot w\left[n\right] \tag{2}$$

has a frequency response, $\hat{h}(\omega)$, which sufficiently represents the desired response, $\hat{h}_d(\omega)$. The window itself has finite support, with $w[n] = 0$ for $0 \leq n \leq M$, the order of the FIR filter to be designed.

1.2.1 Effect of Windowing on the Frequency Response

In the Fourier domain, the modulation of equation (2) becomes circular convolution. Specifically, the DTFT's of $h[n]$ and $h_d[n]$ are related through

$$\begin{aligned} \hat{h}(\omega) &= \frac{1}{2\pi}\int_{-\pi}^{\pi} \hat{h}_d(\theta)\,\hat{w}\left((\omega-\theta) \bmod 2\pi\right) d\theta \\ &= \frac{1}{2\pi}\int_{-\pi}^{\pi} \hat{w}(\theta)\,\hat{h}_d\left((\omega-\theta) \bmod 2\pi\right) d\theta \end{aligned} \tag{3}$$

To understand and interpret this circular convolution formula, let $w(t)$ and $h_d(t)$ be Nyquist band-limited continuous time waveforms whose unit sample sequences are

$$w[n] = w(t)|_{t=n}, \quad \text{and} \quad h_d[n] = h(t)|_{t=n}$$

Obviously, $h[n]$ may be understood as the unit sample sequence of the product of these continuous waveforms, i.e.,

$$\begin{aligned} h[n] &= [w(t)\,h_d(t)]_{t=n} \\ &= h_c(t)|_{t=n}, \quad \text{where } h_c(t) = w(t)\,h_d(t) \end{aligned}$$

The true Fourier transforms of $h_d(t)$ and $w(t)$ are identical to the DTFT's of $h_d[n]$ and $w[n]$ by our definition of these continuous functions as Nyquist bandlimited, but $h_c(t)$ is not generally Nyquist bandlimited. In fact,

$$\hat{h}_c(\omega) = \frac{1}{2\pi}\int_{-\infty}^{\infty} \hat{w}(\theta)\,\hat{h}_d(\omega-\theta)\,d\theta$$

The effect of the window is to smooth (or spread out) $\hat{h}_d(\omega)$, as shown in Figure 2. The DTFT of $h[n]$ is related to the true Fourier transform of $\hat{h}_c(\omega)$ through the aliasing relationship

$$\hat{h}(\omega) = \sum_{k\in\mathbb{Z}} \hat{h}_c(\omega + 2\pi k), \quad -\pi \leq \omega \leq \pi \tag{4}$$

which provides an alternate means of understanding the effect of the circular convolution operation in equation (3).

Interestingly, the analysis above does not in any way rely upon the fact that $w(t)$ is a bandlimited function. Often it is more natural to understand $w[n]$ as a sampled version of a continuous-time window function, $w(t)$, which is not bandlimited. Then $h[n]$ is still obtained by sampling $h_c(t) = w(t)\,h_d(t)$, and $\hat{h}_c(\omega)$ and $\hat{h}(\omega)$ are still related through the aliasing equation (4).

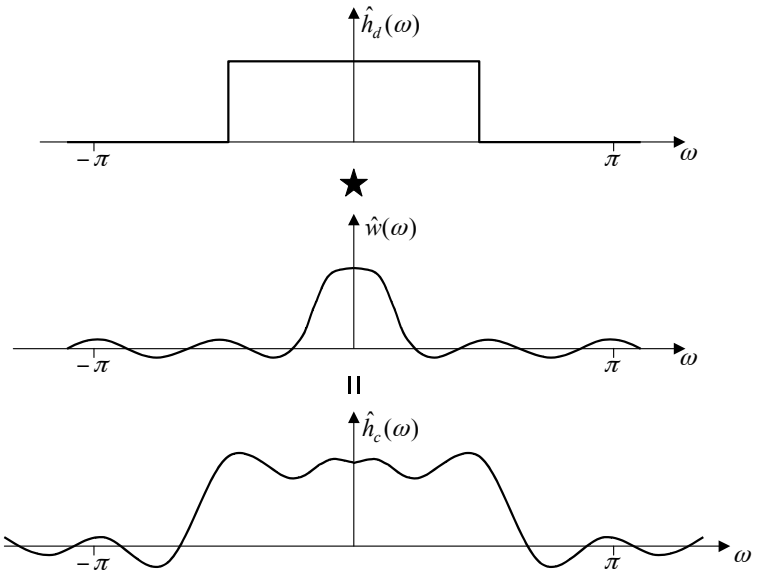

Figure 2: *Effect of windowing on continuous-time signals.*

1.2.2 Windowing Functions

We can now investigate the effect of various windows.

All-or-Nothing Window In the simplest case, we might simply discard the impulse response coefficients which lie outside the range $0 \leq n \leq M$, leaving

$$h[n] = \begin{cases} \frac{1}{2\pi} \int_{-\pi}^{\pi} \hat{h}_d(\omega) e^{j\omega n} & 0 \leq n \leq M \\ 0 & \text{otherwise} \end{cases}$$

This is equivalent to applying the window function,

$$w[n] = \begin{cases} 1 & 0 \leq n \leq M \\ 0 & \text{otherwise} \end{cases}$$

We shall refer to this as the "all-or-nothing" window. Its DTFT is given by

$$\begin{aligned} \hat{w}(\omega) &= \sum_{n=0}^{M} e^{-j\omega n} = \frac{1 - e^{-j\omega(M+1)}}{1 - e^{-j\omega}} \\ &= e^{-j\omega\frac{M}{2}} \cdot \frac{e^{j\omega\frac{M+1}{2}} - e^{-j\omega\frac{M+1}{2}}}{e^{j\omega\frac{1}{2}} - e^{-j\omega\frac{1}{2}}} \\ &= e^{-j\omega\frac{M}{2}} \cdot \frac{\sin\frac{M+1}{2}\omega}{\sin\frac{1}{2}\omega} \end{aligned}$$

For small ω, $\hat{w}(\omega)$ is approximately a sinc function, with a linear phase shift (delay)

$$\hat{w}(\omega) \approx e^{-j\omega\frac{M}{2}} \cdot \frac{\sin\frac{M+1}{2}\omega}{\frac{1}{2}\omega} = e^{-j\omega\frac{M}{2}}(M+1)\,\mathrm{sinc}\left(\frac{M+1}{2\pi}\omega\right)$$

The effect of the all-or-nothing window is to convolve $\hat{h}_d(\omega)$ by $\hat{w}(\omega)$, applying the aliasing equation (4) to wrap out-of-band frequencies back into the range $-\pi$ to π. Evidently, this is similar to convolving $\hat{h}_d(\omega)$ by a sinc function. In fact, the effect is exactly identical to convolving $\hat{h}_d(\omega)$ by a sinc function and wrapping the out-of-band frequencies in the usual manner. To see this, we may follow the suggestion at the end of the previous section of working with any convenient windowing function, $w(t)$, whose samples are identical to $w[n]$. Such a function is

$$w(t) = \begin{cases} 1 & -\frac{1}{2} < t < M+\frac{1}{2} \\ 0 & \text{otherwise} \end{cases}$$

whose Fourier transform is exactly

$$\hat{w}(\omega) = e^{-j\omega\frac{M}{2}}(M+1)\,\mathrm{sinc}\left(\frac{M+1}{2\pi}\omega\right)$$

This sinc function has zeros at the locations $\omega = \frac{2\pi}{M+1}k$, for k an integer. Thus, the main lobe of the all-or-nothing window in the Fourier domain has width $\frac{4\pi}{M+1}$. The width of the main lobe may be interpreted as the transition bandwidth for the windowed filter. However, a key figure of merit for the window is the rate at which the side-lobes decay outside the main lobe. In the case of the all-or-nothing window, $\hat{w}(\omega)$ decays very slowly, as $\frac{2}{(M+1)\omega}$, causing substantial ringing in $\hat{h}(\omega)$.

Hanning Window To improve the decay of $\hat{w}(\omega)$ and hence reduce the ringing in $\hat{h}(\omega)$, a much better example of a window function is the Hanning (or raised-cosine) window. As a continuous-time function, the Hanning window is given by

$$w(t) = \begin{cases} \frac{1}{2}\left(1+\cos\frac{\pi\left(t-\frac{M}{2}\right)}{\frac{M}{2}+1}\right) & -1 < t < M+1 \\ 0 & \text{otherwise} \end{cases}$$

The function is plotted in Figure 3.Letting $\tau = \frac{M}{2}+1$, the Fourier transform of

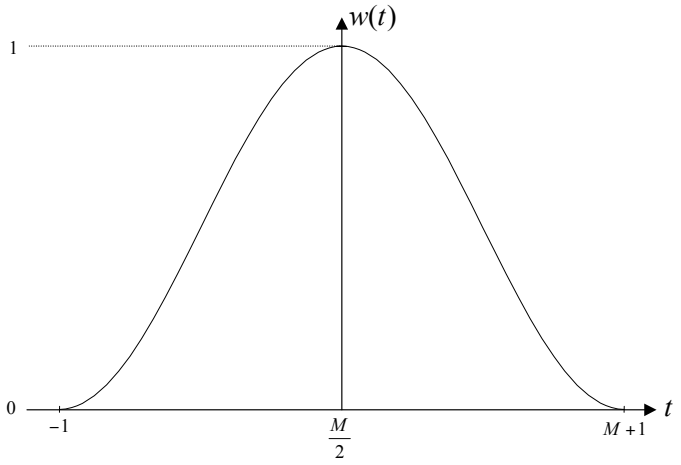

Figure 3: *Hanning window, $w(t)$, for an order M FIR filter.*

$w(t)$ may be found from

$$\begin{aligned}
\hat{w}(\omega) &= e^{-j\omega\frac{M}{2}}\int_{-\tau}^{\tau}\frac{1}{2}\left(1+\cos\frac{\pi t}{\tau}\right)e^{-j\omega t}dt \\
&= e^{-j\omega\frac{M}{2}}\int_{-\tau}^{\tau}\left[\frac{1}{2}e^{-jt\omega}+\frac{1}{4}e^{-jt\left(\omega+\frac{\pi}{\tau}\right)}+\frac{1}{4}e^{-jt\left(\omega-\frac{\pi}{\tau}\right)}\right]dt \\
&= e^{-j\omega\frac{M}{2}}\frac{1}{2}\left[\frac{e^{-j\tau\omega}-e^{j\tau\omega}}{-j\omega}\right] \\
&\quad +e^{-j\omega\frac{M}{2}}\frac{1}{4}\left[\frac{e^{-j\tau\left(\omega+\frac{\pi}{\tau}\right)}-e^{j\tau\left(\omega+\frac{\pi}{\tau}\right)}}{-j\left(\omega+\frac{\pi}{\tau}\right)}\right] \\
&\quad +e^{-j\omega\frac{M}{2}}\frac{1}{4}\left[\frac{e^{-j\tau\left(\omega-\frac{\pi}{\tau}\right)}-e^{j\tau\left(\omega-\frac{\pi}{\tau}\right)}}{-j\left(\omega-\frac{\pi}{\tau}\right)}\right] \\
&= e^{-j\omega\frac{M}{2}}\left[\frac{\sin\tau\omega}{\omega}+\frac{1}{2}\left(e^{j\tau\omega}-e^{-j\tau\omega}\right)\left(\frac{j}{\omega+\frac{\pi}{\tau}}+\frac{j}{\omega-\frac{\pi}{\tau}}\right)\right] \\
&= e^{-j\omega\frac{M}{2}}\left[\frac{\sin\tau\omega}{\omega}-\frac{1}{2}\left(\frac{1}{\omega+\frac{\pi}{\tau}}+\frac{1}{\omega-\frac{\pi}{\tau}}\right)\sin\tau\omega\right] \\
&= e^{-j\omega\frac{M}{2}}\left[\frac{\sin\tau\omega}{\omega}-\frac{\omega\sin\tau\omega}{\omega^2-\left(\frac{\pi}{\tau}\right)^2}\right] \\
&= e^{-j\omega\frac{M}{2}}\frac{-\left(\frac{\pi}{\tau}\right)^2\sin\tau\omega}{\omega\left(\omega^2-\left(\frac{\pi}{\tau}\right)^2\right)} \\
&= e^{-j\omega\frac{M}{2}}\frac{-\left(\frac{2\pi}{M+2}\right)^2\sin\frac{M+2}{2}\omega}{\omega\left(\omega^2-\left(\frac{2\pi}{M+2}\right)^2\right)}
\end{aligned}$$

We see from this expression that $\hat{w}(\omega)$ decays as ω^3 for large ω. The first zeros occur at $\omega = \pm\frac{4\pi}{M+2}$, so the width of the main lobe is $\frac{8\pi}{M+2}$, roughly twice that of the all-or-nothing window. This suggests that the filter will have roughly twice the transition bandwidth, but much less ringing.

Other Window Functions Many different window functions have been described, each with their own particular trade-off between the shape of the main lobe of $\hat{w}(\omega)$ and rate at which its side-lobes decay. Although one could make quite a study of such windows, this would not be a good use of our time here. The window method does not produce an optimal design in any objective sense. Nevertheless, windowing is a very common design strategy precisely because it is simple. In the interest of preserving this simplicity, we will limit our discussion to the windows discussed above. The Hanning window has excellent properties and is extremely simple to remember, while most other interesting window functions have complex expressions for which you would need to refer to books.

1.2.3 Time Shifts and Causality

Up until now we have insisted that our filters all be strictly causal. One unfortunate consequence of causality is that all of our filters must have delay and so an appropriate delay (phase lag) must be introduced into the desired frequency response, $\hat{h}_d(\omega)$. It is often conceptually easier to allow ourselves the luxury of designing non-causal filters and accounting for the delay elsewhere in the implementation. The following example illustrates the benefits of this perspective.

Example 1 *Consider the ideal digital differentiator, whose impulse response was found in Tut 2 to be*

$$h_d[n] = \left[\frac{\partial}{\partial t}\operatorname{sinc}(t)\right]_{t=n}$$

Since $\operatorname{sinc}(t)$ *is symmetric, its derivative must be anti-symmetric, i.e.*

$$h_d[n] = -h_d[-n]$$

from which we may also infer that $h_d[0] = 0$. *If we wish to use a causal window,* $w[n]$, *the impulse response should first be delayed so that its centre of symmetry corresponds with the centre of the window. Otherwise, the linear phase properties of the differentiator will be lost in the windowed filter.*

It is conceptually much easier to allow ourselves the luxury of thinking of $h[n]$ *as a non-causal filter with support* $-L \le n \le L$. *A raised cosine window function with the same region of support is*

$$w[n] = w(t)|_{t=n}, \quad \textit{with } w(t) = \begin{cases} \frac{1}{2}\left(1+\cos\frac{\pi t}{L+1}\right) & |t| < L+1 \\ 0 & |t| \ge L+1 \end{cases}$$

and the designed filter is

$$h[n] = \left[w(t) \cdot \frac{\partial}{\partial t} \operatorname{sinc}(t) \right]_{t=n}$$

To implement this non-causal filter, the input, $x[n]$, must be delayed by L time units to obtain $x'[n] = x[n-L]$. We can then apply the non-causal filter to $x'[n]$ since at time instant n we have access to samples as far into the future as $x'[n+L]$. Of course, applying the filter to a delayed input is no different to applying a delayed version of the impulse response (a causal impulse response) directly to the input. The implementation is no different. The conceptual advantage of designing non-causal FIR filters is that we can match a non-causal impulse response directly during the design phase and then introduce the necessary delay afterwards, depending on the length of the filter which we find we need to design.

1.2.4 Gain Adjustments

It is important to realize that windowing generally has a small effect upon the passband gain of a filter. Suppose, for example, that we are designing a low-pass filter, which is expected to have unit DC gain. This means that the desired impulse response satisfies

$$\sum_{n=-\infty}^{\infty} h_d[n] = \hat{h}_d(0) = 1$$

Standard window functions generally take on values in the range $0 \leq w[n] \leq 1$, so that

$$\sum_{n=0}^{M} h[n] \leq \sum_{n=0}^{M} h_d[n] < \sum_{n=-\infty}^{\infty} h_d[n]$$

To correct for the reduction in DC gain, the windowed filter coefficients should subsequently be re-normalized. It is easy to overlook the need for this final normalization step.

1.3 Least-Squares Designs

In this section, we consider the design of FIR filters whose frequency characteristic matches a desired response, $\hat{h}_d(\omega)$, as closely as possible in the sense of minimizing a squared error criterion. Specifically, our objective is to minimize

$$\mathcal{E} = \frac{1}{2\pi} \int_{-\pi}^{\pi} |\hat{\rho}(\omega)|^2 \left| \hat{h}(\omega) - \hat{h}_d(\omega) \right|^2 d\omega \tag{5}$$

where $\hat{\rho}(\omega)$ is a weighting (or prioritization) function, which can be used to emphasize particular frequencies of interest.

1.3.1 Relationship to Windowing

By Parseval's theorem, our squared-error objective may be restated in the time domain as that of minimizing

$$\sum_{n=-\infty}^{\infty} |\kappa[n]|^2$$

where $\kappa[n]$ is the sequence whose DTFT is $\hat{\kappa}(\omega) = \hat{\rho}(\omega)\left(\hat{h}(\omega) - \hat{h}_d(\omega)\right)$. That is,

$$\kappa[n] = \rho[n] \star (h[n] - h_d[n])$$

This perspective is instructive since it tells us that the unweighted minimum mean squared error design (i.e., when $\hat{\rho}(\omega) = 1$) is equivalent to finding the filter taps, $h[n]$, which minimize

$$\sum_n |h[n] - h_d[n]|^2$$

The solution is obviously to set $h[n] = h_d[n]$ over the entire region of support allowed for the FIR filter. But this is just the windowed filter design strategy, using the all-or-nothing window.

Evidently, then, the all-or-nothing window yields a filter whose frequency response matches the desired response with minimum average squared error. The reality is that this is rarely what we want. Instead, the application is usually very much more sensitive to error in the stopband than in the passband. Consider, for example, a low-pass filter which is required to have unit gain at DC and -60 dB rejection for frequencies above $\pi/2$. This means that the RMS error in the stop band should be less than 0.001, while the RMS error (or ripple) in the passband can probably be 0.1 without causing too many problems. This line of reasoning explains why the all-or-nothing window technique performs so poorly and leads to large ringing artefacts in the stopband, precisely because it views deviations in the frequency response as equally important at all frequencies. To avoid these difficulties, we will generally select highly non-uniform weighting function, $\hat{\rho}(\omega)$.

1.3.2 General Solution Method

Returning now to our weighted squared error criterion in the frequency domain, observe that the objective may be expressed in terms of the filter taps, $h[n] = a_n$, as

$$\mathcal{E} = \frac{1}{2\pi}\int_{-\pi}^{\pi} |\hat{\rho}(\omega)|^2 \left(\sum_{n=L}^{n=U} a_n e^{-j\omega n} - \hat{h}_d(\omega)\right)\left(\sum_{n=L}^{n=U} a_n e^{j\omega n} - \hat{h}_d^*(\omega)\right) d\omega \quad (6)$$

where the * denotes complex conjugation as usual, and the support of the FIR filter, $h[n]$, is given by $L \le n \le U$. For a causal filter, $L = 0$ and $U = M$ is

the order of the filter. As noted previously, however, it is often conceptually simpler to design non-causal filters, introducing the necessary delays only for implementation purposes. For this reason, we keep the lower and upper bounds of the filter's support general here.

Equation (6) represents a quadratic minimization problem, in the coefficients a_L through a_U. These problems always have simple solutions which can be found by solving a set of linear equations. To find the solution, we proceed in the usual way by setting each of the partial derivatives, $\frac{\partial \mathcal{E}}{\partial a_k}$, equal to 0. This gives us

$$\begin{aligned}
0 &= \frac{\partial \mathcal{E}}{\partial a_k} \\
&= \frac{1}{2\pi}\int_{-\pi}^{\pi} |\hat{\rho}(\omega)|^2 \left[\begin{array}{l} e^{-j\omega k}\left(\sum_{n=L}^{n=U} a_n e^{j\omega n} - \hat{h}_d^*(\omega)\right) \\ \quad + e^{j\omega k}\left(\sum_{n=L}^{n=U} a_n e^{-j\omega n} - \hat{h}_d(\omega)\right) \end{array} \right] d\omega \\
&= \left\{ \sum_{n=L}^{n=U} a_n \frac{1}{2\pi}\int_{-\pi}^{\pi} |\hat{\rho}(\omega)|^2 \left(e^{j\omega(n-k)} + e^{-j\omega(n-k)}\right) d\omega \right\} \\
&\quad -\frac{1}{2\pi}\int_{-\pi}^{\pi} |\hat{\rho}(\omega)|^2 \left(\hat{h}_d(\omega) e^{j\omega k} + \hat{h}_d^*(\omega) e^{-j\omega k}\right) d\omega \\
&= 2\left\{ \sum_{n=L}^{n=U} a_n r[n-k] \right\} - 2d[k] \qquad (7)
\end{aligned}$$

where we have defined $r[n]$ to be the sequence whose DTFT is

$$\hat{r}(\omega) = |\hat{\rho}(\omega)|^2$$

and $d[n]$ to be the sequence whose DTFT is

$$\hat{d}(\omega) = |\hat{\rho}(\omega)|^2 \hat{h}_d(\omega)$$

We have used the fact that $\frac{1}{2\pi}\int_{-\pi}^{\pi} \hat{r}(\omega) e^{j\omega(n-k)} d\omega$ is the inverse DTFT of $\hat{r}(\omega)$, evaluated at $n-k$ and, similarly, $\frac{1}{2\pi}\int_{-\pi}^{\pi} \hat{d}(\omega) e^{j\omega k} d\omega$ is the inverse DTFT of $\hat{d}(\omega)$, evaluated at k.

Equation (7) gives us a set of $U-L+1$ linear equations for the $U-L+1$ unknown filter taps, $h[n] = a_n$. The equations may be arranged in matrix form as follows:

$$\underbrace{\begin{pmatrix} r[0] & r[1] & \cdots & r[U-L] \\ r[-1] & r[0] & \cdots & r[U-L-1] \\ \vdots & \vdots & \ddots & \vdots \\ r[L-U] & r[L-U+1] & \cdots & r[0] \end{pmatrix}}_{R} \underbrace{\begin{pmatrix} a_L \\ a_{L+1} \\ \vdots \\ a_U \end{pmatrix}}_{\mathbf{a}} = \underbrace{\begin{pmatrix} d[L] \\ d[L+1] \\ \vdots \\ d[U] \end{pmatrix}}_{\mathbf{d}}$$

From which we deduce that the vector, $\mathbf{a}$, of filter taps is given by

$$\mathbf{a} = R^{-1}\mathbf{d}$$

The matrix, R, has a very special structure known as Toeplitz. It is easy to verify that $r[k] = r[-k]$ so that R is a symmetric matrix. In fact, it can be shown that R is a symmetric, positive-definite, Toeplitz matrix. These properties make inversion of R particularly simple and robust. We shall encounter very similar minimum mean squared error problems when we consider adaptive and Wiener filters in the coming weeks. These problems all have similar solutions, which we shall come to understand very well.

It is worth reminding the reader that if the weighting function, $\hat{\rho}(\omega) = 1$, $r[n] = \delta[n]$ is the unit impulse so that $R = I$ is the identity matrix, and $\hat{d}(\omega) = \hat{h}_d(\omega)$, so that $d[n] = h_d[n]$. Then $a[n] = h_d[n]$ for $L \leq n \leq U$, exactly as we discussed earlier. More generally, when $\hat{\rho}(\omega)$ is not constant, the optimal FIR filter is a function of $d[L]$ through $d[U]$, which depend upon the desired impulse response coefficients $h_d[n]$ outside the range $L \leq n \leq U$. Windowing techniques always completely ignore $h_d[n]$ outside the window and so windowing cannot generally lead to FIR filter designs which are optimal in the sense of minimizing the weighted squared error between $\hat{h}(\omega)$ and $\hat{h}_d(\omega)$, except when all frequencies are weighted equally – usually, a highly undesirable selection.

2 IIR Filter Design

2.1 Summary of Strategies

As for FIR filters, we begin with a brief summary of some of the strategies which have been used to design IIR filters. We ellaborate on two particularly interesting strategies only, in the sub-sections which follow.

Impulse Invariant Method: This is one of a number of methods for designing digital filters which approximate the response of a given analog filter, with transfer function, $H_a(s)$, and impulse response, $h_a(t)$. The design has the property that the digital filter's impulse response, $h[n]$, is a sampling of the analog filter's impulse response, i.e., $h[n] = h_a(t)|_{t=n}$. The design procedure is quite simple; a partial fraction expansion of $H_a(s)$ about its N poles yields

$$H_a(s) = \sum_{i=1}^{N} \frac{\gamma_i}{s - p_i}$$

reducing the problem to that of designing a set of digital filters, each of which has an exponentially decaying impulse response which is a sampled version of the exponentially decaying impulse response of the corresponding single-pole analog filter. We find easily that the digital filter has transfer function

$$H(z) = \sum_{i=1}^{N} \frac{\gamma_i z}{z - e^{p_i}}$$

Since analog filters never have strictly bandlimited frequency responses, $\hat{h}(\omega)$ will always be an aliased version of the analog spectrum, $\hat{h}_a(\omega)$. The significance of these aliasing effects depends upon the rate at which $\hat{h}_a(\omega)$ decays with ω. Not surprisingly, the method is useful primarily for designing low-pass filters.

Derivative Approximation Methods: Another method for designing digital filters which approximate analog filters is to recognize that analog filters arise in the context of ordinary differential equations of the form,

$$y(t) = \sum_{i=0}^{M} a_i x^{(i)}(t) + \sum_{i=1}^{N} b_i y^{(i)}(t)$$

where $x^{(i)}(t)$ denotes the i^{th} derivative of $x(t)$. Now recall that differentiation is a filtering operation, which can be implemented digitally so long as $x(t)$ is Nyquist band-limited. Let $G(z)$ be the transfer function of a digital differentiator. Then the unit sample sequences, $x[n]$ and $y[n]$, of $x(t)$ and $y(t)$ are related through the following Z-domain expression

$$Y(z) = \sum_{i=0}^{M} a_i \left[G(z)\right]^i X(z) + \sum_{i=1}^{N} b_i \left[G(z)\right]^i Y(z)$$

The idea is to construct FIR filters, $G(z)$, which approximate the ideal differentiator to a desired level of accuracy and the implement the digital filter in the manner suggested above. The transfer function is obtained by replacing s with $G(z)$ in $H_a(s)$. By careful selection of $G(z)$, it is possible to ensure that the frequency response of the digital filter approaches that of the analog filter arbitrarily closely. However, high order differentiators, $G(z)$, lead to digital filters with much higher orders than the original analog filters.

Bi-Linear Transformation Method: The bi-linear transform is yet another method for designing digital filters from analog filters. The digital transfer function, $H(z)$ is found by substituting $s = M(z)$ in $H_a(s)$, where $M(z) = \frac{z-1}{z+1}$. This may be understood as a special type of derivative approximation method in which the ideal digital differentiator is approximated by the transfer function function, $G(z) = M(z)$. The transformation can be shown to have a number of important properties. In particular, stable analog filters are always mapped to stable digital filters and the frequency responses of the digital and analog filters are related through a simple warping of the frequency scale: specifically,

$$\hat{h}(\omega) = \hat{h}_a(\omega_a)\Big|_{\omega_a = \tan\frac{\omega}{2}}$$

It is important to realize that this frequency distortion limits the areas of application for the bi-linear transform. The digital frequency response can

be quite different to that of the analog frequency response due to frequency warping. However, the bi-linear transform is of significant interest if we have the luxury of custom designing the original analog filter. In this case, the digital filter's specifications may first be warped into the analog frequency scale and used to drive the analog filter design. The bi-linear transform is discussed further in Section 2.2.

Padé Approximation Method: In this case, the objective is to design a recursive digital filter whose impulse response matches some desired impulse response, $h_d[n]$, as closely as possible. There is no analog filter whose characteristics are being replicated here. Since no filter of finite order can be expected to match an arbitrary $h_d[n]$ perfectly, the Padé method focuses exclusively on the first $M+N+1$ samples of $h_d[n]$, where M and N are the orders of the numerator and denominator polynomials in $H(z)$, respectively. This leads to a set of $M+N+1$ linear equations for the same number of unknown coefficients in

$$H(z) = \frac{a_0 + a_1 z^{-1} + \cdots + a_M z^{-M}}{1 - b_1 z^{-1} - \cdots - b_N z^{-N}}$$

While conceptually simple, the Padé approximation has the very significant drawback that it in no way controls the behaviour of $h[n]$ except in the interval $0 \leq n \leq M+N$. If $H(z)$ has poles close to the unit circle, $h[n]$ decays only slowly and $h[n]$ may deviate significantly from $h_d[n]$ when $n > M+N$.

Least-Square Methods: Least-squares methods attempt to improve on the Padé approximation by attempting to match much more of the impulse response, as closely as possible in the mean-square sense. The least-squares optimization problem is not so straightforward in the case of IIR filters as it was for FIR filters, so the formulation is usually modified in a non-optimal manner, so as to produce linear equations in the unknown filter coefficients. In Section 2.3, we develop a design strategy due to Shanks.

2.2 Bi-Linear Transform Method

As already noted, the bi-linear transform provides a useful method for designing digital filters with similar properties to analog designs. Let $H_a(s)$ denote the transfer function of an analog filter with impulse response $h_a(t)$. So $H_a(s)$ is the Laplace transform of $h(t)$, given by

$$H_a(s) = \int_0^\infty h_a(t)\, e^{-st} dt$$

The frequency response of the analog filter is given by

$$\hat{h}_a(\omega_a) = H_a(s)\big|_{s=j\omega_a}$$

and the analog filter is stable so long as the poles of $H_a(s)$ all lie in the left half plane, i.e., $\Re(p_i) < 0$.

We have already seen that the usefulness of the Z-transform derives from the fact that the only digital filters we can implement with finite computation and memory are those for which $H(z)$ has a finite number of poles and zeros. The Laplace transform is useful in the study of analog filters for exactly the same reason, since the only analog filters which we can construct from a finite number of *lumped* circuit elements are those for which $H_a(s)$ has a finite number of poles and zeros. For this reason, it is important that any mapping between an analog design with transfer function $H_a(s)$ and a digital design with transfer function $H(z)$ preserve the finite-ness of poles and zeros. This property is guaranteed if the mapping is of the following form:

$$H(z) = H_a(s)|_{s=M(z)}$$

where $M(z)$ is a rational function of z.

In addition to $M(z)$ being a rational function of z, useful mappings should have the following properties.

1. $M(z)$ should be invertible, so that the mapping $z = M^{-1}(s)$ can be used to transfer filter design specifications from the digital domain to corresponding design specifications in the analog domain.

2. $M^{-1}(s)$ should map the left half of the s-plane to the interior of the unit circle in the z-plane, so that stable analog filters map to stable digital filters and vice-versa.

3. Ideally, $M^{-1}(s)$ should map the entire imaginary axis in the s-plane onto the unit circle in the z-plane, so that there is a direct mapping between the frequency response, $\hat{h}_a(\omega)$, of the analog filter and the frequency response, $\hat{h}(\omega)$, of the digital filter.

A simple mapping which satisfies all of these properties is the bi-linear transform, given by

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

The inverse relationship is easily found from

$$\begin{aligned}(z+1)\,s &= (z-1)\\ \Longrightarrow z &= \frac{1+s}{1-s}\end{aligned}$$

Now let $z = re^{j\omega}$ be a point in the z-plane. The bi-linear transform maps

this point to

$$\begin{aligned} s &= \frac{re^{j\omega}-1}{re^{j\omega}+1} = \frac{(r\cos\omega-1)+jr\sin\omega}{(r\cos\omega+1)+jr\sin\omega} \\ &= \frac{[(r\cos\omega-1)+jr\sin\omega]\cdot[(r\cos\omega+1)-jr\sin\omega]}{(r\cos\omega+1)^2+r^2\sin^2\omega} \\ &= \frac{r^2-1+2jr\sin\omega}{r^2+1+2r\cos\omega} \end{aligned}$$

If $r<1$, $\Re(s)<0$, while if $r>1$, $\Re(s)>0$. Thus, the bi-linear transform does indeed map the left half of the s-plane to the interior of the unit circle in the z-plane and vice-versa. Finally, if $r=1$, we get

$$\begin{aligned} s &= \frac{j\sin\omega}{1+\cos\omega} = j\frac{2\sin\frac{\omega}{2}\cos\frac{\omega}{2}}{1+\cos^2\frac{\omega}{2}-\sin^2\frac{\omega}{2}} \\ &= j\frac{\sin\frac{\omega}{2}\cos\frac{\omega}{2}}{\cos^2\frac{\omega}{2}} = j\tan\frac{\omega}{2} \end{aligned}$$

It follows that the unit circle in the z-plane is mapped in a continuous fashion onto the entire imaginary axis in the z-plane, with analog frequency, ω_a, related to digital frequency, ω, according to

$$\omega_a = \tan\frac{\omega}{2}, \quad -\pi<\omega<\omega$$

2.2.1 Design Strategy

To design a digital filter using the bi-linear transform, the desired frequency response for the digital filter, $\hat{h}_d(\omega)$, is first mapped to a desired analog filter response through

$$\hat{h}_a(\omega_a) = \hat{h}_d(\omega)\Big|_{\omega=2\tan^{-1}\omega_a}$$

The analog filter is then designed to approximate $\hat{h}_a(\omega_a)$ and then the bi-linear transform is applied to recover the rational transfer function,

$$H(z) = H_a(s)|_{s=\frac{z-1}{z+1}}$$

This design strategy is useful only because it allows us to leverage a significant body of useful analog filter designs. Typical analog filter designs for which tables of specifications can be found include Butterworth, Chebychev, Inverse Chebychev and Elliptical (or Cauer) designs.

2.3 Least-Squares Method

In this section, we consider the problem of finding coefficients, a_i and b_i, such that the impulse response associated with the filter

$$H(z) = \frac{a_0+a_1z^{-1}+\cdots+a_Mz^{-M}}{1-b_1z^{-1}-\cdots-b_Nz^{-N}}$$

matches a desired impulse response, $h_d[n]$, as closely as possible in the least-squares (or weighted least-squares) sense. This problem is much more complicated than the FIR optimization problem considered in Section 1.3, because $\hat{h}(\omega)$ is a non-linear function of the feedback coefficients, b_1 through b_N. The approach proposed by Prony and later improved by Shanks is find these coefficients first, using a non-optimal strategy, and then later to optimize the numerator coefficients, a_0 through a_M, explicitly.

The transfer function given above corresponds to the recursive difference equation:

$$y[n] = \sum_{i=0}^{M} a_i x[n-i] + \sum_{i=1}^{N} b_i y[n-i]$$

By substituting $x[n] = \delta[n]$, we find that the impulse response satisfies

$$h[n] = \begin{cases} 0 & n < 0 \\ a_n + \sum_{i=1}^{N} b_i h[n-i] & 0 \le n \le M \\ \sum_{i=1}^{N} b_i h[n-i] & n > M \end{cases}$$

In Prony's method, the b_i are determined first by considering the desired impulse response, $h_d[n]$, at locations $n > M$. If we were able to find an exact match between $h[n]$ and $h_d[n]$, then the b_i coefficients should satisfy $h_d[n] = \sum_{i=1}^{N} b_i h_d[n-i]$ for all $n > M$. Since we do not expect to be so lucky, we try instead to minimize the following squared error objective,

$$\mathcal{E}_1 = \sum_{n=M+1}^{\infty} \left(h_d[n] - \sum_{i=1}^{N} b_i h_d[n-i] \right)^2$$

Setting each of the partial derivatives, $\frac{\partial \mathcal{E}_1}{\partial b_k}$, to 0, we obtain

$$\begin{aligned} 0 &= \frac{\partial \mathcal{E}_1}{\partial b_k} = 2 \sum_{n=M+1}^{\infty} h_d[n-k] \left(h_d[n] - \sum_{i=1}^{N} b_i h_d[n-i] \right) \\ &= \underbrace{2 \left(\sum_{n=M+1}^{\infty} h_d[n-k]\, h_d[n] \right)}_{r_{k,0}} - 2 \sum_{i=1}^{N} b_i \underbrace{\left(\sum_{n=M+1}^{\infty} h_d[n-k]\, h_d[n-i] \right)}_{r_{k,i}} \end{aligned}$$

The feedback coefficients can then be found by solving the linear equations

$$\begin{pmatrix} r_{1,1} & r_{1,2} & \cdots & r_{1,N} \\ r_{2,1} & r_{2,2} & \cdots & r_{2,N} \\ \vdots & \vdots & \ddots & \vdots \\ r_{N,1} & r_{N,2} & \cdots & r_{N,N} \end{pmatrix} \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_N \end{pmatrix} = \begin{pmatrix} r_{1,0} \\ r_{2,0} \\ \vdots \\ r_{N,0} \end{pmatrix}$$

Once the feedback coefficients have been found, we can formulate a minimum mean-squared error problem for the numerator (feedforward) coefficients,

a_0 through a_M. This can be formulated in the time domain or the frequency domain, but the frequency domain formulation allows us to introduce a frequency-dependent weighting function, $\hat{\rho}(\omega)$, which was found to be a key element in the successful design of FIR filters using least-squares methods. For this reason, we will attempt to minimize

$$\mathcal{E}_2 = \frac{1}{2\pi} \int_{-\pi}^{\pi} |\hat{\rho}(\omega)|^2 \left| \hat{h}(\omega) - \hat{h}_d(\omega) \right|^2 d\omega \tag{8}$$

which is identical to equation (5). Writing

$$H(z) = \frac{A(z)}{B(z)}$$

and noting that we have already found the polynomial, $B(z) = 1 - b_1 z^{-1} - \cdots - b_N z^{-N}$, we may rewrite the optimization objective as

$$\begin{aligned}
\mathcal{E}_2 &= \frac{1}{2\pi} \int_{-\pi}^{\pi} \left| \frac{\hat{\rho}(\omega)}{B(e^{j\omega})} \right|^2 \cdot \left| A(e^{j\omega}) - B(e^{j\omega}) \hat{h}_d(\omega) \right|^2 d\omega \\
&= \frac{1}{2\pi} \int_{-\pi}^{\pi} \left| \frac{\hat{\rho}(\omega)}{\hat{b}(\omega)} \right|^2 \cdot \left| \hat{a}(\omega) - \hat{b}(\omega) \hat{h}_d(\omega) \right|^2 d\omega \\
&= \frac{1}{2\pi} \int_{-\pi}^{\pi} \left| \hat{\rho}'(\omega) \right|^2 \cdot \left| \hat{a}(\omega) - \hat{h}'_d(\omega) \right|^2 d\omega
\end{aligned}$$

So the optimization proceeds in exactly the same manner as it did for FIR filters, replacing the desired impulse response by

$$\hat{h}'_d(\omega) = \hat{b}(\omega) \hat{h}_d(\omega)$$

and replacing the weighting function by

$$\hat{\rho}'(\omega) = \frac{\hat{\rho}(\omega)}{\hat{b}(\omega)}$$

We conclude this section by noting that the final solution does not strictly minimize the squared error objective in equation (8), since the b_i and a_i coefficients are not optimized jointly.

3 FIR vs. IIR Considerations

Having surveyed and, in some cases detailed, the major filter design techniques it is useful at this point to consider the relative merits of FIR and IIR filter types.

Complexity: The complexity of an FIR filter is proportional to the length of the impulse response. For IIR filters, no such simple relationship exists. As a general rule, filters with sensitive magnitude response characteristics (high stopband attenuation, uniform passband gain, etc.) can generally be realized with much lower complexity as IIR filters than FIR filters.

Leverage Analog Designs: IIR filters can be designed with similar behaviour to analogue filters realized with lumped circuit components. This existing analogue filter design techniques to be leveraged for digital design. FIR filters, by contrast, have no analogue counterparts.

Linear Phase: IIR filters cannot have linear phase, meaning that the group delay of an IIR filter cannot be constant. This property is shared with analogue filters constructed from lumped circuit elements. Some applications are particularly sensitive to variations in the group delay. These include modulation/demodulation processing systems for communications, especially when spread spectrum modulation techniques are employed. Image processing applications also generally demand linear phase since the human visual system is extremely sensitive to phase information. FIR filters can have exactly linear phase and are generally preferred for such applications.

Complex Structure: Some applications require the design of filters with complex structure. For example, a filter designed to immitate or enhance the acoustic properties of a music hall, must be capable of replicating complex reflections at numerous boundaries, generally with very long time constants. The needs of such applications cannot feasibly be met using IIR filters. Instead FIR filters are used.

FFT Methods: FFT methods may be employed to dramatically reduce the complexity of an FIR filter implementation, for filters of very high order. Similar techniques do not exist for high order IIR filters.

Finite Word Length Effects: IIR filters are much more sensitive to coefficient quantization and the propagation of numerical round-off errors than FIR filters. A typical transversal FIR filter implementation presents a single source of round-off noise at the output of the filter, since there is only one accumulator and hence one downshifting operation. By contrast, in an IIR filter, at least one source of round-off noise is presented to every section of the implementation whose transfer function involves poles. The poles can dramatically amplify round-off errors at selective frequencies.