Design of FIR filters

Topics: windowed-sinc filters,stopband, passband, transition width

An FIR filter is, as we've seen, a finite weighted sum of inputs:

$$y[n]=b_0 x[n]+b_1 x[n-1]+\cdots+b_{M-1} x[n-(M-1)]=\sum_{k=0}^{M-1} b_k x[n-k]$$

The coefficients b_k (also called *tap weights*) determine the frequency response (with F in cycles/sample) as follows:

$$H(F)=\sum_{k=0}^{M-1} b_k e^{-j2\pi F k}$$

Because cycles/sample are normalized units that repeat with a period of 1.0, we can , without loss of generality, consider the frequency response for $-0.5\leqslant F\leqslant 0.5$. Note that $F=0.5$ is the Nyquist frequency (half the sampling rate). Furthermore, for any filter with real-valued coefficients (which is all filters that we consider in this class), we have a conjugate-symmetry around the origin[1]:

$$H(-F)=\overline{H(F)}$$

Many methods of designing FIR filters exist, including software optimization programs with graphical user interfaces. The "windowed-sinc" method is popular and well-known, and it is described below. In this method, the coefficients of the mathematically ideal filter, which is a sinc() function, are truncated ("windowed") to a finite number of tap weights.

Design Criteria

The ideal lowpass filter with cutoff frequency F_c has frequency response

$$H(F)=\begin{Bmatrix} 1, & \textit{if } |F|\leq F_c \\ & 0, \textit{else} \end{Bmatrix}$$

As we prove below, this response cannot be achived by any FIR filter. Therefore, we must allow for deviations from the ideal response to obtain a practical filter. Let us divide the response of the filter into two bands: a passband, and a stopband. In the passband, we allow

1 The overline denotes complex-conjugate.

$$1-\epsilon_p \le |H(F)| \le 1+\epsilon_p \quad \text{for} \quad |F| \le F_{pe} \ ,$$

and in the stopband, we allow

$$|H(F)| \le \epsilon_s \quad \text{for} \quad F_{se} \le |F| \le 0.5$$

Here, the parameter ϵ_p determines the allowed variation in the passband, F_{pe} is the passband edge frequency, and ϵ_s , F_{se} play the corresponding roles for the stopband. The gap $\Delta F = F_{se} - F_{pe}$ is called the transition width of the filter. Usually, ϵ_s , and frequently, ϵ_p is specified in terms of decibels or dB. The quantity

$$A_s = -20 \log_{10} \epsilon_s$$

is called the stopband attentuation in dB. So, 20 dB stopband attenuation means the stopband frequencies are at least 90% attenuated (reduced) in amplitude, 40 dB means stopband frequencies are 99% attenuated, etc. Figure 1 illustrates these parameters.

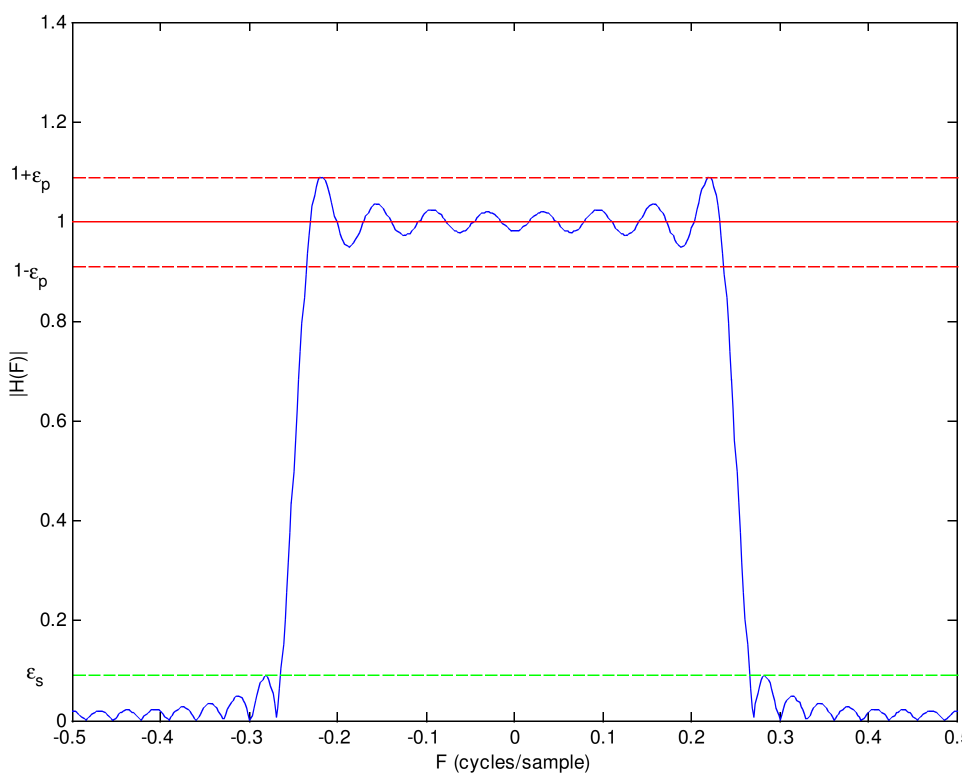

Figure 1: Lowpass filter with passband variation ϵ_p *and stopband attenuation* ϵ_s

Specifying the amplitude response is only half the filter design, of course. We would also need to specify the phase response. Since we want group delay to be constant to avoid phase distortion, we require a linear phase:

$$phase\, H(F) = -2\pi F \frac{(M-1)}{2} \quad \text{, for} \quad |F| \leq F_{pe}$$

With this phase response, the group delay is $\frac{M-1}{2}$ samples, which means all input frequencies emerge delayed an amount equal to the middle tap weight's location.

Design method

Achieving the perfectly flat group delay turns out to be easy . If we require that the tap weights be symmetric about their midpoint

$$b_k = b_{M-1-k} \quad \text{for} \quad k = 0,1,\ldots,M-1$$

then the frequency response is

$$H(F) = b_0\left(1 + e^{-j2\pi F(M-1)}\right) + b_1\left(e^{-j2\pi F} + e^{-j2\pi F(M-2)}\right) + \cdots$$

Note that each of the rightmost terms can be simplified as

$$\begin{aligned} b_k\left(e^{-j2\pi Fk} + e^{-j2\pi F(M-1-k)}\right) &= b_k e^{-j2\pi F(M-1)/2}\left(e^{j2\pi F(M-1-2k)/2} + e^{-j2\pi F(M-1-2k)/2}\right) \\ &= 2 b_k e^{-j2\pi F(M-1)/2} \cos(2\pi F(M-1-2k)/2) \end{aligned}$$

This means we can write for the case (when M is even)

$$H(F) = e^{-j2\pi F(M-1)/2}\left[\sum_{k=0}^{M/2} 2 b_k \cos(2\pi F \frac{M-2k}{2})\right]$$

and, when M is odd, that

$$H(F) = e^{-j2\pi F(M-1)/2}\left[b_{(M+1)/2} + \sum_{k=0}^{(M-1)/2} 2 b_k \cos(2\pi F \frac{M-2k}{2})\right]$$

In each case, we see that in the passband that the term in brackets is non-negative, and therefore

$$phase\ H(F) = -2\pi F \frac{(M-1)}{2} \quad \Rightarrow \quad \tau_g(F) = \frac{M-1}{2}$$

Therefore, ***a symmetric impulse response produces constant group delay.***

We now turn to the problem of designing the coefficients. First, we try to calculate what filter produces the ideal response, and next, determine how to obtain a finite approximation of it. The ideal response is, in both amplitude and phase,

$$H_i(F) = \begin{Bmatrix} e^{-j2\pi F(M-1)/2}, & if\ |F| \le F_c \\ 0, & else \end{Bmatrix}$$

The corresponding impulse response can be found by integration

$$h_i[n] = \int_{-0.5}^{0.5} H_i(F) e^{-j2\pi F n} dF$$

Without going through the details, it can be shown that the solution is

$$h_i[n] = \frac{\sin(2\pi F_c(n-(M-1)/2))}{\pi(n-(M-1)/2)} = 2F_c sinc(2F_c(n-(M-1)/2))$$

where $sinc(x) = \sin(\pi x)/(\pi x)$ by definition. A plot is shown in Figure 2.

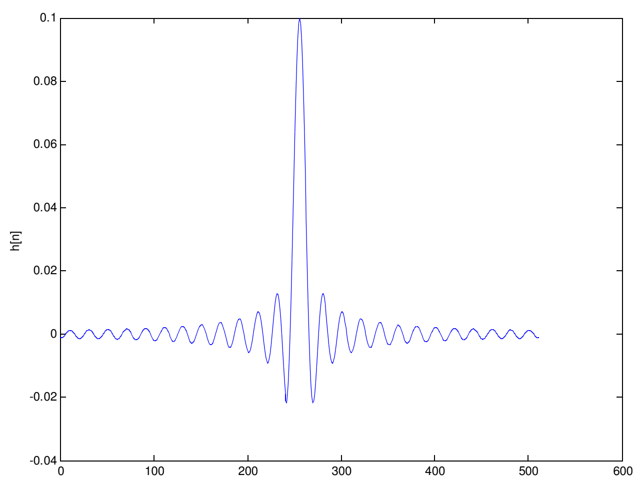

Figure 2: A plot of the ideal filter $h_i[n]$ *for* $M=512$ *and* $Fc=0.05$ *cycles/sample.*

Notice that the ideal filter is infinite in extent, indicating that **the perfect amplitude response cannot be attained by a FIR filter.** Therefore we will have to accept an approximation to the perfect response. However, we still want to control ϵ_p and ϵ_s, the passband and stopband tolerances. In order to do this, we use a window function to weight the ends of the response.

The windowed-sinc filter is defined as

$$h_w[n]=w[n]h_i[n] \quad \text{, for} \quad n=0,\ldots,M-1$$

The window function generally has the shape of a bell-curve. There are many window functions used in DSP, with names like the Hamming window, the von Hann window, the Blackman window, the Kaiser window, and the Chebyshev window. The Kaiser window and the Chebyshev windows are particularly useful, because their characteristics can be adjusted to meet or exceed any desired ϵ_p and ϵ_s.

We will use the Kaiser window in our work, since it is general enough to cover all our design needs. The Kaiser window is defined as

$$w[n]=\frac{I_0\left[\beta\sqrt{1-\frac{(n-\alpha)^2}{\alpha^2}}\right]}{I_0(\beta)} \quad \text{, for} \quad n=0,\ldots,M-1$$

Here, I_0 is a modified Bessel function of zeroth-order (whose values can be computed easily in MATLAB), β is a parameter that controls the shape of the filter, and $\alpha=(M-1)/2$. The value of β is determined experimentally as follows:

$$\beta=\left\{\begin{array}{c} 0.1102(A_s-8.7),\ \mathit{if}\ A_s>50 \\ 0.5842(A_s-21)^{0.4}+0.07886(A_s-21),\ \mathit{for}\ 21\le A_s\le 50 \\ 0,\ A_s\le 21 \end{array}\right\}$$

Note that A_s is the stopband attenuation in decibels, as defined above. With this choice of β, we also obtain $\epsilon_p\approx\epsilon_s$. The filter length M is determined by the formula:

$$M=\frac{A_s-8}{2.285(2\pi\,\Delta F)}$$

With the Kaiser window, the design procedure is as follows:

1. Determine parameters ϵ_p , ϵ_s , F_{pe} , and F_{se} . Then $\Delta F = F_{se} - F_{pe}$ and M is determined as above.
2. Set $F_c = \frac{(F_{se} + F_{pe})}{2}$, and compute the filter response as $h_w[n] = w[n] h_i[n]$
3. If the filter is required to have a null at Nyquist, or $F = 0.5$, then choose M to be an even number. If M is even, the response to a Nyquist frequency input is

$$H(0.5) = \sum_{n=0}^{M-1} h_w[n] e^{-j\pi} = h_w[0] - h_w[1] + h_w[2] - \cdots - h_w[M-3] + h_w[M-2] - h_w[M-1]$$

Since $h[n]$ and $w[n]$ are symmetric, so too is $h_w[n]$, which means that

$$h_w[0] - h_w[M-1] = 0 \quad , \quad h_w[1] - h_w[M-2] = 0 \quad \text{, etc}$$

Therefore, since M is even, $H(0.5) = 0$.

4. Plot the amplitude and phase responses, and verify that the design criteria have been met. See the examples in the Appendix.

5. If necessary, increase M or adjust β until desired response is obtained.

Examples

1. Design a FIR filter having a maximum passband variation of 0.01 for frequencies $F \le 0.2$, and a stopband variation of at least 40 dB for frequencies $F \ge 0.22$.

   Solution: Since, $\Delta F = 0.22 - 0.2 = 0.02$, we obtain that

$$M = \frac{40 - 8}{2.285(2\pi 0.02)} = 111.4 \Rightarrow M = 112$$

   Set $F_c = 0.21$ cycles/sample and compute $h_w[n] = w[n] h_i[n]$, for $n = 0, \ldots, 111$.

2. Design a FIR filter that passses all frequencies less that 100 Hz with no more than 1 percent variation in amplitude, and attenuates frequencies above 120 Hz

by at least 60 dB. The filter should have zero response at Nyquist. The sampling rate is 400 Sa/s.

Solution: From the description, we see that $F_{pe}=0.25$, $F_{se}=0.30$, $\epsilon_p=0.01$, $A_s=60$. We see that

$$M=\frac{60-8}{2.285(2\pi 0.05)}=72.44\Rightarrow M=74$$

where we have chosen the next larger even length to obtain a null at Nyquist. With $F_c=0.275$, we compute the filter coefficients as above.

MATLAB implementations for both examples are shown in the code in the appendix.

Ideal forms for various filters

Filter type	***Description***	***Formula for ideal filter*** $h_i[n]$
Lowpass	Passes freqencies above F_c	$2F_c sinc(2F_c(n-M_i))$
Highpass	Passes frequencies below F_c	$(-1)^n(1-2F_c)sinc\left((1-2F_c)(n-M_i)\right)$
Bandpass	Passband center F_{pc} , half-width of F_{hw}	$\left[2\cos(2\pi F_{pc}n)\right]\left[2F_{hw}sinc(2F_{hw}(n-M_i))\right]$
Bandstop	Stopband from F_1 up to F_2	$2F_1 sinc(2F_1(n-M_i))$ $+(-1)^n\left[(1-2F_2)sinc((1-2F_2)(n-M_i))\right]$

In each case, use $n=0,\ldots,M-1$, and $M_i=\frac{M-1}{2}$. Note that for the bandpass filter, passband is from $F_1=F_{pc}-F_{hw}$ up to $F_2=F_{pc}+F_{hw}$.

```
% FIR filter design examples
% demonstrating the use of the Kaiser
% window.
%
% R Kakarala
% U C Berkeley Extension

% Example 1 parameters
Fpe = 0.2;  Fse = 0.22;    % passband and stopband edges
eps_p = 0.01; As = 40;    % passband variation and stopband
                           % attenuation in dB

eps_s = 10^(As/-20);       % compute equivalent stopband limit
if (eps_s > eps_p)         % do we need to increase attenuation
                           % to get passband criterion?
   eps_s = eps_p;
   As = -20*log10(eps_s);

end;

DeltaF = Fse - Fpe;        % compute number of coefficients
M=(As-8)/(2.285*(2*pi*DeltaF));
M=ceil(M);                 % increase to next largest integer

w = kaiserwindow(M,As);    % compute Kaiser window
Fc = (Fse+Fpe)/2;          % ideal filter cutoff
n = 0:(M-1);
hi = 2*Fc*sinc(2*Fc*(n - (M-1)/2)); % ideal filter

hw = w.*hi;                % windowed sinc;

% frequency response;
[Hw,F] = iirfreqresp(hw);
figure,plot(F,abs(Hw)),title('example 1');

% check if criteria were met

Hstop = Hw(abs(F)>=Fse);  % response in stopband
fprintf(1,'Example 1 results...\n');
fprintf(1,'Number of coefficients used = %4d\n',M);
fprintf(1,'Min Stopband Attenuation (dB)=%8.5f\n',min(-20*log10(abs
(Hstop)+eps)) );
fprintf(1,'Target was=%8.5f (dB)\n',As);
Hpass = Hw(abs(F)<=Fpe);  % response in stopband
fprintf(1,'Max Passband Variation =%8.5f\n',max(abs(abs(Hpass)-1)) );
fprintf(1,'Target was=%8.5f (dB)\n',eps_p);

% Example 2 parameters
Fpe=100/400; Fse=120/400; % passband and stopband edges
eps_p = 0.01; As = 60;    % passband variation and stopband
                           % attenuation in dB

eps_s = 10^(As/-20);       % compute equivalent stopband limit
if (eps_s > eps_p)         % do we need to increase attenuation
                           % to get passband criterion?
   eps_s = eps_p;
```

```
   As = -20*log10(eps_s);

end;

DeltaF = Fse - Fpe;        % compute number of coefficients
M=(As-8)/(2.285*(2*pi*DeltaF));
M=ceil(M);                 % increase to next largest integer

w = kaiserwindow(M,As);    % compute Kaiser window
Fc = (Fse+Fpe)/2;          % ideal filter cutoff
n = 0:(M-1);
hi = 2*Fc*sinc(2*Fc*(n - (M-1)/2)); % ideal filter

hw = w.*hi;                % windowed sinc;

% frequency response;
[Hw,F] = iirfreqresp(hw);
figure,plot(F,abs(Hw)),title('example 2');

% check if criteria were met

Hstop = Hw(abs(F)>=Fse);  % response in stopband
fprintf(1,'Example 2 results...\n');
fprintf(1,'Number of coefficients used = %4d\n',M);
fprintf(1,'Min Stopband Attenuation (dB)=%8.5f\n',min(-20*log10(abs
(Hstop)+eps)) );
fprintf(1,'Target was=%8.5f (dB)\n',As);
Hpass = Hw(abs(F)<=Fpe);  % response in stopband
fprintf(1,'Max Passband Variation =%8.5f\n',max(abs(abs(Hpass)-1)) );
fprintf(1,'Target was=%8.5f (dB)\n',eps_p);
```