EEO 401
Digital Signal Processing
Prof. Mark Fowler

Note Set #30

• Linear Phase FIR Design – Frequency Sampling Method
• Reading: Sect. 10.2.3 of Proakis & Manolakis
**Basic Idea**

Suppose you have a desired frequency response for which you’d like to design a realizable FIR filter: \( H_d^f(\omega) \)

The corresponding impulse response is:

\[
    h_d[n] = \frac{1}{2\pi} \int_0^{2\pi} H_d^f(\omega) e^{jn\omega} d\omega
\]

Note: Changing to 0 to 2\( \pi \) has no effect!

Of course, doing this IDTFT does not get us an FIR filter!

So… recalling that the DFT
- is a sort-of stand-in for the DTFT
- And it acts on a finite-duration signal

… we wonder if we could use the IDFT on frequency-domain samples of \( H_d^f(\omega) \)

Define:

\[
    H[k] = H_d^f \left( k \frac{2\pi}{M} \right), \quad k = 0, 1, 2, \ldots, M - 1
\]

Then an IDFT gives:

\[
    h[n] = \frac{1}{M} \sum_{k=0}^{M-1} H[k] e^{j2\pi km/M}, \quad n = 0, 1, 2, \ldots, M - 1
\]

Then a DTFT of that gives the designed FR:

\[
    H^f(\omega) = \sum_{k=0}^{M-1} h[n] e^{j\omega n}, \quad -\pi \leq \omega \leq \pi
\]
Ensuring Linear Phase Design

The designer specs the magnitude and then a linear phase is imparted:

\[ |H[k]| = \text{specified}, \quad k = 0, 1, 2, \ldots, M - 1 \]

\[ h[n] = \frac{1}{M} \sum_{k=0}^{M-1} \left[ |H[k]| e^{-j \frac{2\pi \cdot k (M-1) n}{M}} \right] e^{j \frac{2\pi \cdot k n}{M}}, \quad n = 0, 1, 2, \ldots, M - 1 \]
Expanded Idea – Using Window

The idea here is to use a finer frequency sampling grid than is set by the desired filter length.

**Basic Form: \( M = \text{FIR Length} \)**

\[
|H[k]| = \left| H_d^f \left( k \frac{2\pi}{M} \right) \right|, \quad k = 0, 1, 2, \ldots, M - 1
\]

\[
h[n] = \frac{1}{M} \sum_{k=0}^{M-1} \left| H[k] \right| e^{-jk \frac{2\pi}{M}(M-1)/2} e^{j\frac{2\pi kn}{N}}, \quad n = 0, 1, 2, \ldots, M - 1
\]

**Extended Form: \( N \gg M = \text{FIR Length} \)**

\[
|H[k]| = \left| H_d^f \left( k \frac{2\pi}{N} \right) \right|, \quad k = 0, 1, 2, \ldots, N - 1
\]

\[
\tilde{h}[n] = \frac{1}{N} \sum_{k=0}^{N-1} \left| H[k] \right| e^{-jk \frac{2\pi}{N}(M-1)/2} e^{j\frac{2\pi kn}{N}}, \quad n = 0, 1, 2, \ldots, N - 1
\]

\[
h[n] = \tilde{h}[n] w_M[n], \quad n = 0, 1, 2, \ldots, M - 1
\]

\( w_M[n] \) is length \( M \) window

Use Finer Frequency Grid
MATLAB fir2

% FIR2   FIR arbitrary shape filter design using the frequency sampling method.
%   B = FIR2(N,F,A) designs an Nth order linear phase FIR digital filter
% with the frequency response specified by vectors F and A and returns
% the filter coefficients in length N+1 vector B.
%
% The vectors F and A specify the frequency and magnitude breakpoints for
% the desired frequency response. The frequencies in F must be given in
% increasing order with 0.0 < F < 1.0 and 1.0 corresponding to half the
% sample rate. The first and last elements of F must equal 0 and 1
% respectively.

f = [0 0.6 0.6 1];  % Frequency breakpoints
m = [1 1 0 0];      % Magnitude breakpoints
b = fir2(30,f,m);   % Frequency sampling-based FIR filter design

Uses 2 x NPT frequency points

Default: NPT = 512

% B = FIR2(N,F,A,NPT) specifies the number of points, NPT, for the grid
% onto which FIR2 linearly interpolates the frequency response. NPT must
% be greater than 1/2 the filter order (NPT > N/2). If desired, you can
% interpolate F and A before passing them to FIR2.
MATLAB fir1 vs. fir2

From the MATLAB website documentation:

- “Use fir1 for window-based standard lowpass, bandpass, highpass, bandstop, and multiband configurations.”
- Use fir2 for windowed filters with arbitrary frequency response.

More details and examples of using fir1 & fir2 are available at: