Digital Signal Processing HW9 MATLAB Part

Name: Jingyang Zhang
Net ID: jz2807

Speech filtering exercise, same as in HW 8, but this time design a linear phase FIR filter. Please try to design a Type I FIR filter using the method where you specify the designed amplitude response and weight functions over the continuous domain. Please compare the results with two different weighting options under the same filter length: i) Constant weight over entire frequency range; ii) Weight function turned to emphasize noise removal and allow a transition bank. Also compare the results obtained with two different lengths. One relatively long (e.g. <=10), one relatively short (e.g. >10). You should show the desired amplitude response, your weight functions, the resulting filter impulse response and frequency response, the spectrogram of the original and filtered signals.

Optional: trying other filter types and other design methods discussed in the class.

Solution:

.m file(s): jyz_HW9.m

Code:

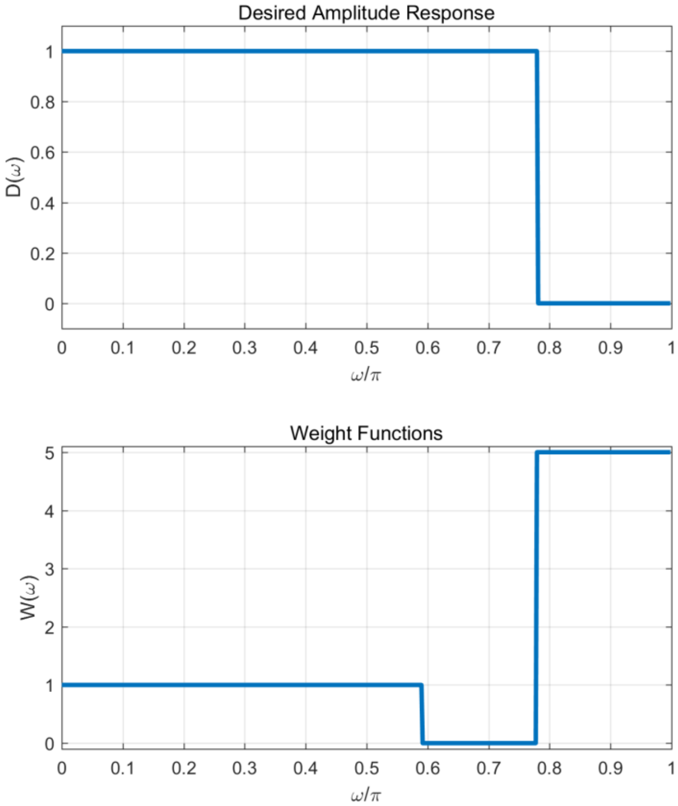
```
close all
clear

%Load signals and basic processing ↓
load NoisySpeech.txt
load mtlb
Noi = NoisySpeech;
L = length(mtlb);
Noif = fft(Noi, 4096);
mtlbf = fft(mtlb, 4096);

NL = 15; ML = (NL-1)/2; nL = 0 : NL-1;    %relatively long filter length
NS = 7;    MS = (NS-1)/2; nS = 0 : NS-1;    %relatively short filter length
wp = (2200/(Fs/2))*pi; fp = wp/pi;           %pass-band ends at
ws = (2900/(Fs/2))*pi; fs = ws/pi;            %stop-band strats from
wv = (2900/(Fs/2))*pi; fv = wv/pi;           %no transition band
K = 5;                                        %weight of stop-band
D = zeros(1, 512);                           %set the Desired Amplitude Response
D(1 : fix(512 * fv)) = 1;
W = zeros(1, 512);                            %set the Weight Functions
W(1 : fix(512 * fp)) = 1;
W(fix(512 * fs) : 512) = K;
aL = zeros(1,NL);aL(1) =1;                    %numerator of Long filter
aS = zeros(1,NS);aS(1) =1;                   %numerator of Short filter

%Long Unweighted filter ↓
hLUnWei = (wv/pi) * sinc((wv/pi) * (nL - ML));
[HLUnWei, wLUnWei] = freqz(hLUnWei,aL);

%Long Unweighted filter ↓
hSUnWei = (wv/pi) * sinc((wv/pi) * (nS - MS));
[HSUnWei, wSUnWei] = freqz(hSUnWei,aS);
```

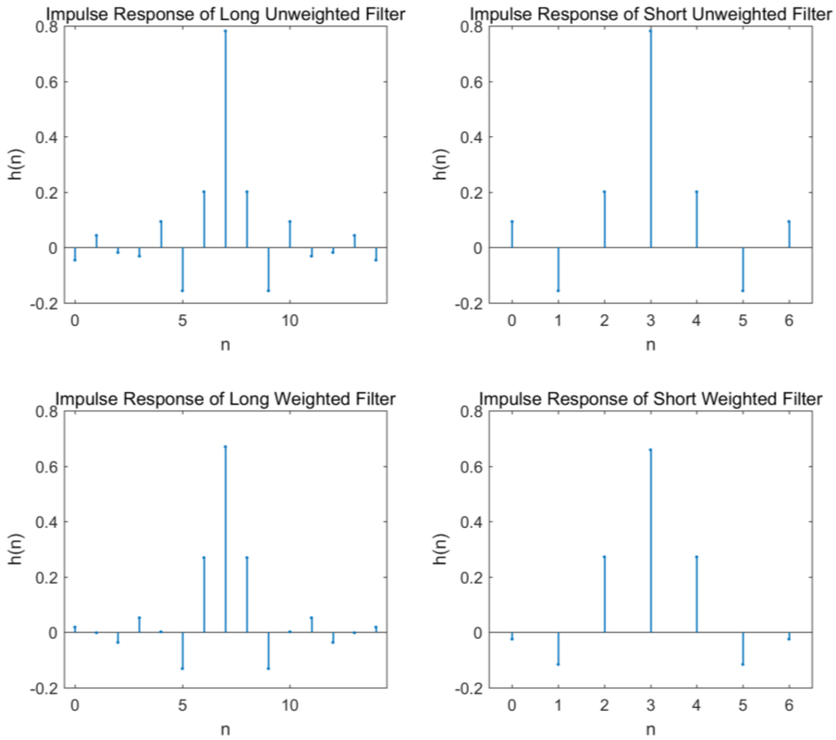
```
%Long Weighted filter ↓
qLWei = [fp+K*(1-fs), fp*sinc(fp*[1:2*ML])-K*fs*sinc(fs*[1:2*ML])];
QLWei1 = toeplitz(qLWei([0:ML]+1));
QLWei2 = hankel(qLWei([0:ML]+1),qLWei([ML:2*ML]+1));
QLWei = (QLWei1 + QLWei2)/2;
bLWei = fp*sinc(fp*[0:ML]');
aLWei = QLWei\bLWei;
hLWei = [aLWei(ML+1:-1:2); 2*aLWei(1); aLWei(2:ML+1)]/2;
[HLWei, wLWei] = freqz(hLWei,aL);

%Short Weighted filter ↓
qSWei = [fp+K*(1-fs), fp*sinc(fp*[1:2*MS])-K*fs*sinc(fs*[1:2*MS])];
QSWei1 = toeplitz(qSWei([0:MS]+1));
QSWei2 = hankel(qSWei([0:MS]+1),qSWei([MS:2*MS]+1));
QSWei = (QSWei1 + QSWei2)/2;
bSWei = fp*sinc(fp*[0:MS]');
aSWei = QSWei\bSWei;
hSWei = [aSWei(MS+1:-1:2); 2*aSWei(1); aSWei(2:MS+1)]/2;
[HSWei, wSWei] = freqz(hSWei,aS);

%Signal filtered↓
yLUnWei = filter(hLUnWei, aL, Noi); %by Long Unweighted filter
yLUnWeif = fft(yLUnWei, 4096);
ySUnWei = filter(hSUnWei, aS, Noi); %by Short Unweighted filter
ySUnWeif = fft(ySUnWei, 4096);
yLWei = filter(hLWei, aL, Noi);         %by Long Weighted filter
yLWeif = fft(yLWei, 4096);
ySWei = filter(hSWei, aS, Noi);         %by Short Weighted filter
ySWeif = fft(ySWei, 4096);

figure(1)
subplot(2,1,1)
plot(0 : 1/512 : 1-1/512, D,'LineWidth',2.5)
ylim([-0.1 1.1])
xlabel('\omega/\pi');ylabel('D(\omega)')
title('Desired Amplitude Response')
grid on
subplot(2,1,2)
plot(0 : 1/512 : 1-1/512, W,'LineWidth',2.5)
ylim([-0.1 5.1])
xlabel('\omega/\pi');ylabel('W(\omega)')
title('Weight Functions')
grid on

figure(2)
subplot(2,2,1)
stem(nL, hLUnWei, 'r.', 'LineWidth',1)
xlim([-0.5, NL - 0.5])
```

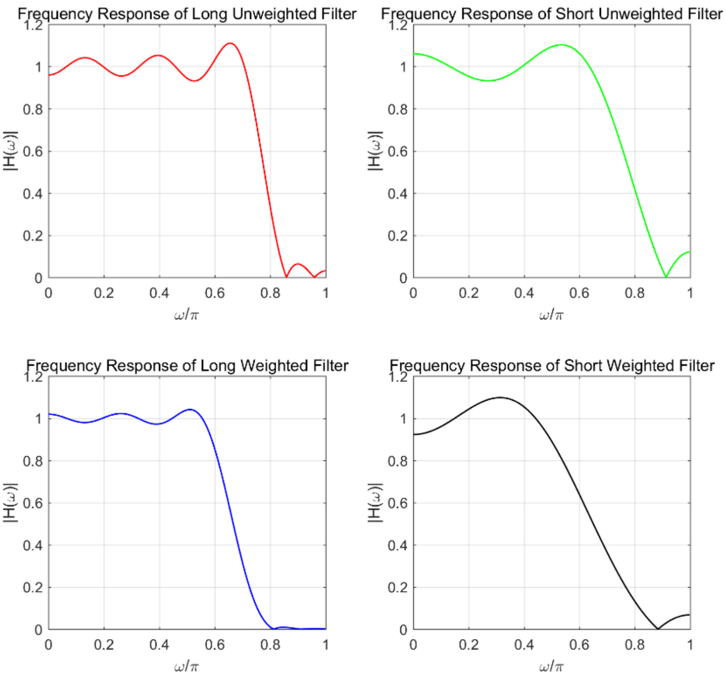
```
xlabel('n');ylabel('h(n)');
title('Impulse Response of Long Unweighted Filter')
subplot(2,2,2)
stem(nS, hSUnWei, 'g.', 'LineWidth',1)
xlim([-0.5, NS - 0.5])
xlabel('n');ylabel('h(n)');
title('Impulse Response of Short Unweighted Filter')
subplot(2,2,3)
stem(nL, hLWei, 'b.', 'LineWidth',1)
xlim([-0.5, NL - 0.5])
xlabel('n');ylabel('h(n)');
title('Impulse Response of Long Weighted Filter')
subplot(2,2,4)
stem(nS, hSWei, 'k.', 'LineWidth',1)
xlim([-0.5, NS - 0.5])
xlabel('n');ylabel('h(n)');
title('Impulse Response of Short Weighted Filter')

figure(3)
subplot(2,2,1)
plot(wLUnWei/pi, abs(HLUnWei), 'r', 'LineWidth',1)
xlim([0,1]);grid on;
xlabel('\omega/\pi');ylabel('|H(\omega)|');
title('Frequency Response of Long Unweighted Filter')
subplot(2,2,2)
plot(wSUnWei/pi, abs(HSUnWei), 'g', 'LineWidth',1)
xlim([0,1]);grid on;
xlabel('\omega/\pi');ylabel('|H(\omega)|');
title('Frequency Response of Short Unweighted Filter')
subplot(2,2,3)
plot(wLWei/pi, abs(HLWei), 'b', 'LineWidth',1)
xlim([0,1]);grid on;
xlabel('\omega/\pi');ylabel('|H(\omega)|');
title('Frequency Response of Long Weighted Filter')
subplot(2,2,4)
plot(wSWei/pi, abs(HSWei), 'k', 'LineWidth',1)
xlim([0,1]);grid on;
xlabel('\omega/\pi');ylabel('|H(\omega)|');
title('Frequency Response of Short Weighted Filter')

figure(4)
subplot(3,2,1)
spectrogram(mtlb)
title('Clean Signal')
subplot(3,2,2)
spectrogram(Noi)
title('Noisy Signal')
subplot(3,2,3)
spectrogram(yLUnWei)
```

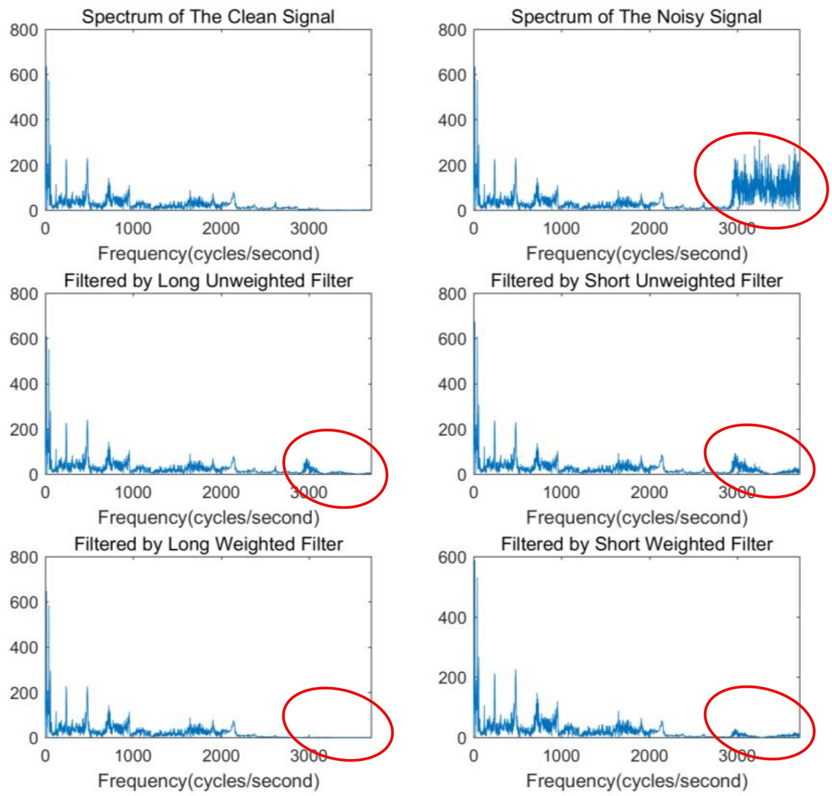
```
title('Filtered by Long Unweighted Filter')
subplot(3,2,4)
spectrogram(ySUnWei)
title('Filtered by Short Unweighted Filter')
subplot(3,2,5)
spectrogram(yLWei)
title('Filtered by Long Weighted Filter')
subplot(3,2,6)
spectrogram(ySWei)
title('Filtered by Short Weighted Filter')

figure(5)
subplot(3,2,1)
plot(0:Fs/4096:Fs-Fs/4096,abs(mtlbf))
title('Spectrum of The Clean Signal')
xlabel('Frequency(cycles/second)');xlim([0, 3708]);
subplot(3,2,2);
plot(0:Fs/4096:Fs-Fs/4096,abs(Noif))
title('Spectrum of The Noisy Signal');
xlabel('Frequency(cycles/second)');xlim([0, 3708]);
subplot(3,2,3);
plot(0:Fs/4096:Fs-Fs/4096,abs(yLUnWeif))
title('Filtered by Long Unweighted Filter')
xlabel('Frequency(cycles/second)');xlim([0, 3708]);
subplot(3,2,4);
plot(0:Fs/4096:Fs-Fs/4096,abs(ySUnWeif))
title('Filtered by Short Unweighted Filter')
xlabel('Frequency(cycles/second)');xlim([0, 3708]);
subplot(3,2,5);
plot(0:Fs/4096:Fs-Fs/4096,abs(yLWeif))
title('Filtered by Long Weighted Filter')
xlabel('Frequency(cycles/second)');xlim([0, 3708]);
subplot(3,2,6);
plot(0:Fs/4096:Fs-Fs/4096,abs(ySWeif))
title('Filtered by Short Weighted Filter')
xlabel('Frequency(cycles/second)');xlim([0, 3708]);
```

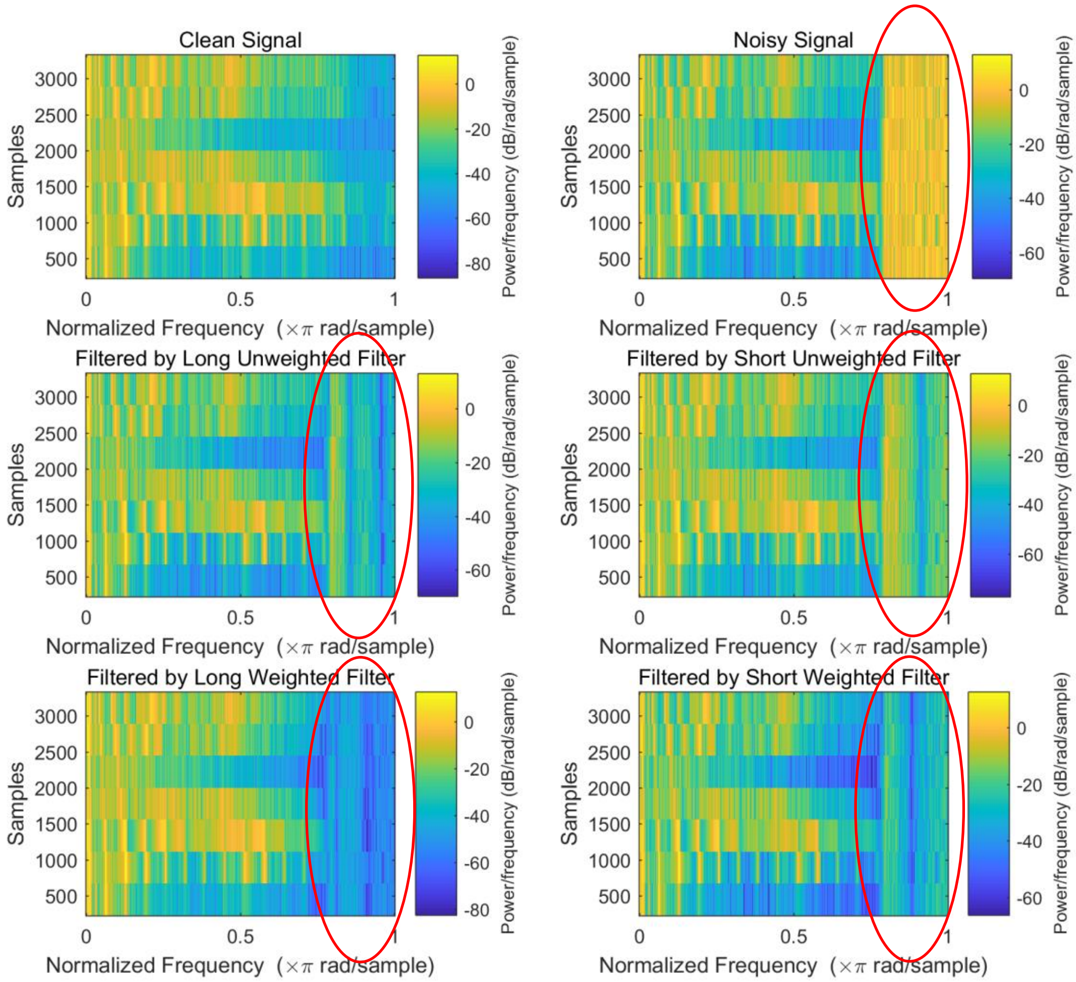
Result(plots):











Comments: The desired amplitude response, weight functions, the resulting filter impulse response and frequency response, the spectrogram of the original and filtered signals are shown in the plots.

By comparing the spectrum and the spectrogram of the original noisy and filtered signals, it could be concluded that the long weighted filter has the best noise-removing effect. And generally speaking, weighted filters have better noise-removing effect than unweighted filters, and the longer the filters are, the better they could annihilate the noise.

Note that the frequency response is not exactly 1 in the pass band, there is some amplitude change for sure.