Music 421
Spring 2010-2011
Homework #4
Finding Peaks
Due in one week

Theory Problems

1. (2 pts) What length Blackman window is required to resolve a sinusoid at 100 Hz and another one at 101 Hz? State your definition of resolution in this context, and draw a sketch showing the two sinusoids and the window transform in the frequency domain, with the window transform being centered on one of the sinusoidal frequencies.

2. Suppose we are going to analyze a sinusoid whose frequency is $f_0 = 440$ Hz using a rectangular window. Assume the window length is $M = 255$ and the sampling rate is $f_s = 8192$ Hz.

   (a) (2 pts) What is the relative error (*i.e.*, $\Delta f / f_0 = |f_0 - \hat{f}|/f_0$) between the actual frequency (f_0) of the sinusoid and the peak frequency ($\hat{f}$) when there is no zero-padding?

   (b) (2 pts) Repeat part (a) with the zero-padding factor of 5.

   (c) (3 pts) Repeat part (a) with the zero-padding factor of 5 and the parabolic interpolation using the peak and its two neighbors. You don't need to build algorithms for peak finding or parabolic interpolation for now. Just find the peak and its neighbors graphically.

3. (6 points) [Least Squares Sinusoidal Parameter Estimation] You are trying to estimate a single complex sinusoidal component in additive white noise: $x(n) = \tilde{A}\exp[j\omega_0 n] + v(n)$ where $\tilde{A} = A\exp[j\phi]$ and $v(n)$ is Gaussian white noise. You compute the frequency ω_0 using peak finding in the magnitude response. You finally need to estimate your complex amplitude $\tilde{A}$ using least squares from the measured signal $x(n)$. Your cost function that you wish to minimize is:

   $J(\hat{\tilde{A}}) = \sum_{n=0}^{N-1} |x(n) - \hat{\tilde{A}}\exp[j\omega_0 n]|^2$

   or writing out the complex amplitude,

   $J(\hat{\tilde{A}}) = \sum_{n=0}^{N-1} |x(n) - \hat{A}\exp[j\hat{\phi}]\exp[j\omega_0 n]|^2$

   where $\hat{\tilde{A}}$ is the optimal value for the complex amplitude. When w_0 is known, this is a quadratic optimization problem and can be solved by differentiating $J(\hat{\tilde{A}})$ with respect to $\hat{\tilde{A}}$, equating it to zero, and solving.

(a) (3 points) Find an expression for the optimal $\hat{\hat{A}}$ either by differentiating the cost $J(\hat{\hat{A}})$ or some other method.

(b) (3 points) Use your method to estimate the frequency, complex amplitude (real amplitude and phase) of the signal in the provided in the matlab data file sin_est.mat, given a sampling rate $fs = 100$ Hz. To load the file, type: load sin_est.mat. Report the estimated frequency, phase, and real amplitude.

Lab Assignments

1. In this problem, you will compare the ability of the Hann (a.k.a. "Raised Cosine") window to that of the Hamming window to resolve two sinusoids of significantly different amplitudes[1] and with "significantly different" frequencies[2]. To that end, write a matlab script to perform the following:

   (a) (2 pts) Create a 64-sample sum of two cosines, the first with unity amplitude and normalized frequency 1/8 (cycles per sample), and the second with amplitude 0.001 and normalized frequency 3/8 (cycles per sample).

   (b) (3 pts) Window this signal with a 64-sample Hann window (created using the matlab function `Hann()`)[3], and compute the resulting spectrum. For this problem, you may compute the spectrum of a signal using the matlab `freqz()` function, by setting $b = x$ (where x is your input signal vector), and $a = 1$. Finally, plot the magnitude spectrum in dB on a gridded figure with normalized frequency (in cycles per sample) along the x-axis.

   (c) (3 pts) Repeat the above procedure for a 64-sample Hamming window, overlaying the spectrum with that of the Hann window.

   (d) (2 pts) Which of the windows does a better job of resolving the two sinusoids? What drawback does this window have versus the other in regard to side-lobe levels?

2. (20 pts) **Spectral Peak Estimation**

   (a) (10 pts) Construct a program to find the frequencies (in Hz) and the magnitudes (in linear amplitude) of the positive-frequency peaks of an input DFT of a given signal, using the code skeleton shown below. Be sure to convert matlab frequency index numbers to frequencies in Hz.

[1]Here we mean different by a few orders of magnitude.

[2]Here by "significantly different" we mean spaced by more than a few side-lobe widths.

[3]Throughout this problem, always normalize the window to read 0 dB at DC

```
function [peaks,freqs]=findpeaks(Xwdb,maxPeaks,fs,win,N)
% peaks = a vector containing the peak magnitude estimates (linear) using
%         parabolic interpolation in order from largest to smallest peak.
% freqs = a vector containing the frequency estimates (Hz) corresponding
%         to the peaks defined above
% Xwdb  = DFT magnitude (in dB scale) vector of a windowed signal.
%         NOTE that it may contain
%         only low-frequency (length < N/2+1), positive-frequency
%         (length = N/2+1), or all (length = N) bins of the FFT.
% maxPeaks = the number of peaks we are looking for
% fs       = sampling frequency in Hz
% win      = window used to obtain Xwdb (assumed zero phase)
% N        = NFFT, the number of points used in the FFT creating Xwdb

%-- Find all peaks (magnitudes and indices) by comparing each point of ---%
%-- magnitude spectrum with its two neighbors ---%
allPeaks = [];
for i=2:length(Xwdb)-1
    ...
    ...
end

%-- Order from largest to smallest magnitude, keep only maxPeaks of them --%
peaks = ...

%-- Do parabolic interpolation in dB magnitude to find more accurate peak --%
%-- and frequency estimates --%
for i=1:maxPeaks
    idx=find(Xwdb==peaks(i));
    %parabolic interpolation
    a=Xwdb(idx-1);
    b=Xwdb(idx);
    c=Xwdb(idx+1);
    ...
    ...
end
%-- Return linear amplitude and frequency in Hz --%
% NOTE that we must use knowledge of the window to normalize amplitude here
% if we have a TD cosine of amplitude 0.6, this output should be 0.6
peaks = ...
freqs = ...
```

(b) (4 pts) Test your `findpeaks.m` function on a length-255, 400-Hz cosine having amplitude 1 and no phase offset, i.e.,

$$x(n) = \cos(2\pi \cdot 400nT), \quad n = -127, \ldots, 127.$$

(Implicitly, a rectangular window of the same length is used here). Use your zero-phase zero-pad window function to extend the signal to length 2048 before peak detection. Let the sampling rate be $f_s = 8000$ Hz. Show your plot of the magnitude spectrum with the found peak clearly marked. Also plot the phase spectrum. The spectrum should be real because we zero-phase windowed a cosine, which is even (*i.e.*, the FFT of a real, even sequence is real and even). Thus, the phase should be $\pm\pi$. (Matlab may produce some "imaginary dust" in the spectrum due to round off error.) Here is some code to get you started:

```
subplot(211);
plot(abs(fft(zpzpwin(cos(2*pi*400/8000*(-127:127)'),boxcar(255),2048))));
subplot(212);
plot(angle(fft(zpzpwin(cos(2*pi*400/8000*(-127:127)'),boxcar(255),2048))));
```

(c) (3 pts) Download `s1.wav`[4] from the homework assignments page. Find the amplitudes and frequencies of each component in the signal. You need not zero phase window the signal, nor should you zero pad it. Just use a rectangle window whose length is the same of the signal. To avoid getting NaN in your spectrum due to the log of zero, use something like: `Xwdb = 20*log10(abs(Xlinear)+eps);` .

(d) (3 pts) Again, with `s1.wav` but now using the Hamming window of length equal to that of $x(n)$ (255 samples) and with the same amount of zero-padding (to length 2048), find the frequency and amplitude estimates. Do you expect the estimates to be better in the case of Hamming window here? Why or why not?

[4] `http://ccrma.stanford.edu/~jos/hw421/hw4/s1.wav`