PROBLEM SET 2

Issued: 9/7/17  
Due: 9/13/17

Reading: During the past week we began our discussion about sampling continuous-time signals and multi-rate DSP, although because of the holiday we are not very far along yet. Hence this problem set will primarily cover basic material concerning sampling and change of sampling rate. Our discussion of multi-rate DSP will be based primarily on the material in Secs. 3.0-3.3 of Lim and Oppenheim (LO), focussing on Sec. 3.3. You should also look over Secs. 3.4-3.7 in LO even though we may not directly discuss all of the material in this section in class. The applications of multirate DSP described in Sec. 3.7 are interesting, and will be relevant to work you will be doing for the next two problem set! The Lim and Oppenheim chapter is available on the Web.

Note: This problem set is a bit short because we have not covered enough advanced material in class yet.

Problem 2.1:

Introduction: Commercial CD players routinely employ upsampling, although this fact is not as heavily advertised as it was during the early days of compact disks when this feature provided some competitive advantage. (Even today, you may see a spec such as “8X oversampling.”) In this problem we discuss why upsampling is commonly used in commercial digital audio systems. Although digital sound reproduction systems (like compact discs) nominally store and process sound “without distortion,” the signal still has to be converted back to a continuous-time waveform (which was called an “analog” waveform when CD players were first introduced) before it reaches the listener. While rapidly declining hardware costs make it practical to implement discrete-time filters with very sharp transitions between passband and stopband, it is still difficult and expensive to design high-performance continuous-time filters. We will compare the effect that non-ideal continuous-time filtering has on an idealized discrete-time sound reproduction system with and without upsampling.

Assume that an arbitrary audio signal with a bandwidth of 20 kHz (or \(2\pi \times 20 \times 10^3 \) radians/sec) is sampled at a sampling frequency of 44.1 kHz (the standard sampling rate for commercial digital audio). As you know, this means that the upper discrete-time cutoff frequency of the digitized audio signal is 2.85 or 0.907 \(\pi\) radians, based on the relation developed in class \(\omega = \Omega T\), where \(\omega\) is the discrete-time frequency in radians and \(\Omega\) is the continuous-time frequency in radians per second and \(T\) is the sampling period in seconds.

Assume also that a third-order Butterworth filter is used for the continuous-time lowpass anti-aliasing fil-
The magnitude of the continuous-time frequency response is

\[ |H(j\Omega)| = \frac{1}{\left[ 1 + (\Omega / \Omega_c)^6 \right]^{1/2}} \]

where \( \Omega_c \) is the cutoff frequency of the filter. Note that this filter has a gain of 1 at low frequencies.

Following common practice, we will compare the magnitudes of the frequency response of the continuous-time filter at different frequencies in decibels. As you may recall, the ratio in decibels (dB) of the magnitudes of a filter’s transfer functions at two frequencies is equal to

\[ 20 \log \left( \frac{|H(j\Omega_1)|}{|H(j\Omega_2)|} \right), \text{ with the logarithm taken to the base 10.} \]

(a) Now we’re ready to consider a simple discrete-time playback system without upsampling. Consider the system shown in block diagram form in Fig 2.1a. Assume that \( X(e^{j\omega}) \), the DTFT of the digitized signal \( x[n] \), is as shown in Fig 2.1b.

![Figure 2.1a](image)

![Figure 2.1b](image)

where
\[ y_s(t) = \sum_{n=-\infty}^{\infty} x[n] \delta(t - nT) \]

and with sampling period is \( T = 1/(44.1 \, \text{kHz}) \) in this case.

Again, \( H(j\Omega) \) is a three-pole Butterworth filter, and the magnitude of its transfer function was given above. (Please note: this is not the ideal reconstruction filter discussed in class! The function \( y(t) \) is the output of the filter.

1. Sketch and dimension \( Y_s(j\Omega) \), the (continuous-time) Fourier transform of \( y_s(t) \).

2. What is the lowest frequency (in rad/sec) at which extraneous frequency components of \( y_s(t) \) appear that were introduced by the sampling of the original audio signal? (In other words, determine the lowest frequency of the first periodic repetition of the spectrum that is not centered at 0 rad/sec.)

3. Assume that \( \Omega_c \), the cutoff frequency of the continuous-time lowpass filter \( H(j\Omega) \) is set to the Nyquist frequency, \((2\pi)(22.05)(10^3) \) rad/sec. Relative to its response at low frequencies, by how much does the continuous-time lowpass filter attenuate the magnitude of the original audio signal at its upper cutoff frequency \((2\pi)(20)(10^3) \) rad/sec? By how much does the continuous-time lowpass filter attenuate the magnitude of the lowest extraneous frequency component that was your answer to part (a2) of this question? Express your answers to these questions in decibels.

(b) Now we’ll consider a similar system, but with upsampling by a factor of 4:1. In Figure 2.1c, \( v[n] \) is an upsampled or interpolated version of \( x[n] \).

\[ y_s(t) = \sum_{n=-\infty}^{\infty} x[n] \delta(t - nT) \]

To obtain \( v[n] \) we first construct \( w[n] \), a “zero-padded” version of \( x[n] \):
\[ w[n] = \begin{cases} x[n/4], & \text{where } n \text{ equals some integer multiple of 4} \\ 0, & \text{otherwise} \end{cases} \]

\( v[n] \) is obtained by passing \( w[n] \) through an ideal discrete-time lowpass filter with transfer function \( H_f(e^{j\omega}) \) given (for \( |\omega| < \pi \)) by

\[ H_f(e^{j\omega}) = \begin{cases} 1, & |\omega| < \pi/4 \\ 0, & \text{otherwise} \end{cases} \]

The ideal D/C converter works as described above, except that the sampling period has been decreased by a factor of 4 so that now \( T = 1/((4)(44.1 \text{ kHz})) \). \( y_{su}(t) \) is the output of the D/C converter of this second system with upsampling, and \( y_u(t) \) is the total system output after lowpass filtering. As before, we will assume that \( H(j\Omega) \) is a third-order Butterworth lowpass filter.

1. Sketch and dimension \( W(e^{j\omega}) \) and \( V(e^{j\omega}) \), the DTFTs of \( w[n] \) and \( v[n] \), respectively, and \( Y_{su}(j\Omega) \), the CTFT of \( y_{su}(t) \).

2. What is the lowest frequency (in rad/sec) at which extraneous components of \( y_{su}(t) \) (not in the original audio signal) appear in this second system with upsampling?

3. Again assume that the cutoff frequency of the continuous-time lowpass filter \( H(j\Omega) \) is set to \((2\pi)(22.05)(10^3) \text{ rad/sec}\), as before. Relative to its response at low frequencies, by how much does the analog lowpass filter attenuate the magnitude of the original audio signal at its upper cutoff frequency, \((2\pi)(20)(10^3) \text{ rad/sec}\)? By how much does the analog lowpass filter attenuate the magnitude of the lowest extraneous frequency component? Express your answers to these questions in decibels.

**Problem 2.2:**

\[
\begin{array}{c}
\text{\(x[n]\)}\uparrow3\rightarrow\text{\(v[n]\)}\rightarrow\text{\(h[n]\)}\downarrow4\rightarrow\text{\(y[n]\)}
\end{array}
\]

The figure above depicts a rather prosaic multi-rate system that changes the effective sampling rate of the input by a factor of 3/4.

(a) Assume that \( h[n] \) is an ideal lowpass filter. What is its gain and cutoff frequency?

(b) Assume that the input \( x[n] \) has a spectrum equal to \( X(e^{j\omega}) = 1 - (|\omega|/\pi) \) for \( |\omega| \leq \pi \) and that \( X(e^{j\omega}) \) is periodic in \( \omega \) with period \( 2\pi \).
1. Sketch and dimension $V(e^{j\omega})$, $W(e^{j\omega})$, and $Y(e^{j\omega})$, the DTFTs of the time functions $v[n]$, $w[n]$, and $y[n]$.

2. Does aliasing occur? If so, what would be the maximum possible upper frequency that the input signal $x[n]$ could be restricted to in order to avoid aliasing?

(c) Repeat part (b-1), but with the input signal $x[n] = \cos(0.8\pi n)$. Obtain an analytical expression for the output signal $y[n]$.

**Problem 2.3:**

![Diagram of system](image)

In the system above, the input signal is $x[n] = \sin\left(0.4\pi n - \frac{\pi}{4}\right)$

The system decimates the input by a factor of 4, upsamples the result by a factor of 3 (i.e., places two zeros between each successive sample of $v[n]$), and finally passes the system through the ideal lowpass filter $H(e^{j\omega})$ with frequency response

$$H(e^{j\omega}) = \begin{cases} 
3, & 0 < |\omega| < \pi/3 \\
0, & \pi/3 \leq |\omega| < \pi 
\end{cases}$$

As it turns out, this system is not particularly well designed, but do not let that bother you.

(a) Sketch and dimension the following DTFTs:

3. $V(e^{j\omega})$, the DTFT of $v[n]$

4. $W(e^{j\omega})$, the DTFT of $w[n]$

5. $Y(e^{j\omega})$, the DTFT of $y[n]$

(b) It is claimed that the output $y[n]$ can be expressed in the following form:

$$A \cos(\omega_0 n + \phi)$$
Find numerical values for the coefficients $A$, $\omega_0$, and $\phi$.

**Note:** in obtaining the answer to this problem, keep in mind that $\delta(at) = \frac{1}{|a|}\delta(t)$ in the distributional sense. This should be taken into consideration when working with a delta function of a continuous argument, and the scale of the horizontal axis is changed.

**Problem 2.4:**

For the system shown in the figure above, obtain an expression for the output $y[n]$ in terms of the input $x[n]$. Simplify the expression for $y[n]$ to the extent possible.

**MATLAB Problems**

**Problem C2.1:**

In this problem we will implement a change of utterance of a short musical segment by a factor of $3/4$, as in Problem 2.2. The file `PSO_B1short.wav` is a short segment of the First Brandenburg Concerto from a performance that I played at CMU (on harpischord) some years ago with a chamber ensemble from the Pittsburgh Symphony.

(a) Read the file `PSO_B1short.wav` using the MATLAB routine `audioread()`. You will note that the sampling rate has been changed to 16 kHz and the file is in stereo. Create a mono version of the file by averaging the two stereo channels.

(b) Use the MATLAB routine `firpm` (or the MATLAB design tool `fdatool`) to design an appropriate low-pass filter for the change-of-rate system depicted in Problem 2.2 that upsamples by a factor of 3 and then downsamples by a factor of 4. Design your filter to have a passband ripple of 1 dB and stopband ripple of -80 dB. The stopband frequency should be the nominal ideal cutoff frequency and the transition bandwidth should be 10 percent of the cutoff frequency. How many coefficients are needed for the filter?

(c) As you know, aside from the filtering operation, upsampling by $L = 3$ is accomplished by inserting two samples of value zero between each sample of the input. Downsampling by a factor of 4 is accomplished by simply taking every fourth sample of the signal to be downsampled and discarding the rest of it. Write your own MATLAB functions that perform the interpolation and decimation. (Do not use the built-in MATLAB functions `interp` and `decimate` for this part of the problem.)

(d) Run the mono file derived from the musical performance through your interpolator, filter, and decimator in cascade. Listen to the output using the MATLAB routine `soundsc()`. How does it sound? It should sound lower in pitch (by a minor third), but also slower and with a different timbre. We will discuss a way to change pitch without changing rate or sound quality (as in autotuning) in a future problem set. Save this output as `PSO_resampled.wav` using the MATLAB routine `audiowrite()`.

(e) Now pass the audio through the built-in MATLAB utilities `upfirdn` and `resample`, which also
accomplish a change in sampling rate. (Read the help files for upfirdn and resample if you are not already familiar with them.) Play the output using soundsc and listen to it. To what extent does the change of sampling rate utility that you implemented “by hand” produce results that sound the same as the output obtained using the two built-in MATLAB functions upfirdn and resample? And for that matter to what extent do the audio outputs obtained using upfirdn and resample sound like each other?

**What you must turn in on paper:**

- Your answers to the questions asked in parts (a) through (f).

- Block diagrams and descriptions of the three systems as you implement them. Be sure to state the critical choices you made in designing the lowpass filter and other aspects of the system, along with how you arrived at your decisions.

- Hard copy of your MATLAB scripts.

**What you will submit electronically:**

- Electronic copies of the MATLAB scripts that you used to perform the upsampling and downsampling, along with an electronic file of your lowpass filter coefficients.

- Your saved output wave PSOresampled.wav.

Label all files with your andrew id and the suffix “2_1”. Email your sound file to Raymond Xia, CCing me, preferably as an attachment.