ECE 348: Digital Signal Processing Lab
Lab 2 (Spring 2020) — 100 points

Waheed U. Bajwa

February 8, 2020

General Instructions

Please submit a written report within two weeks of *your* lab session, describing the purposes of the experiments and the methods used, and include all graphs and Matlab code. Examples of all the graphs that you need to generate in this lab and include in your report are contained in this handout. The reports must be uploaded to Sakai under Assignments within the allotted time frame. The written report will count *only if* you had attended the *full* double-period lab session in which you are registered.

1 Simple filter design

This part is based on textbook Example 4.33 and Problem 4.72.

Problem 1.1 (15 points)**.** It is desired to design an order-4 (i.e., length 5) filter, termed a *finite-impulse response (FIR) filter*, with a symmetric impulse response, $\mathbf{h} = [b_1, b_2, b_3, b_2, b_1]$. The input consists of the sum of the three sinusoids:

$$x(n) = \sin(\omega_1 n) + \sin(\omega_2 n) + \sin(\omega_3 n). \tag{1}$$

The filter is to be designed so that it removes the first and third components ω_1, ω_3, and allows the middle component ω_2 to pass through with unity amplitude, but perhaps with a small delay, that is, the output should have the following form (in the steady state):

$$y(n) = \sin\left(\omega_2(n-d)\right) \tag{2}$$

with the delay d to be determined. Taking advantage of the symmetry of the coefficients, show that the filter's frequency response can be written in the form (10 points):

$$H(\omega) = e^{-2j\omega}\, B(\omega)\,, \qquad B(\omega) = 2b_1\cos(2\omega) + 2b_2\cos(\omega) + b_3. \tag{3}$$

Your filter design conditions are then,

$$\begin{aligned} B(\omega_1) &= 2b_1\cos(2\omega_1) + 2b_2\cos(\omega_1) + b_3 = 0,\\ B(\omega_2) &= 2b_1\cos(2\omega_2) + 2b_2\cos(\omega_2) + b_3 = 1,\\ B(\omega_3) &= 2b_1\cos(2\omega_3) + 2b_2\cos(\omega_3) + b_3 = 0. \end{aligned} \tag{4}$$

Since $H(\omega_2) = e^{-2j\omega_2}B(\omega_2) = e^{-2j\omega_2}$, the delay introduced by the filter on the ω_2 component will be $d = 2$. Using MATLAB, solve the linear system of equations (4) for the three unknown filter coefficients b_1, b_2, b_3, for the following specific values of $\omega_1, \omega_2, \omega_2$ (5 points):

$$\omega_1 = 0.05\pi\,, \quad \omega_2 = 0.1\pi\,, \quad \omega_3 = 0.2\pi \qquad \text{[rads/sample]}.$$

Problem 1.2 (5 points)**.** Make a plot of the magnitude response $|H(\omega)|$ of the designed filter over the interval $0 \leq \omega \leq 0.25\pi$ and add the three frequency points $\omega_1, \omega_2, \omega_3$ on the graph (see example graph below in Fig. 1).

Problem 1.3 (12.5 points)**.** Generate the following signals over the time range $0 \leq n \leq 100$ (5 points):

$$
\begin{aligned}
s(n) &= \sin(\omega_2 n) && \text{(desired signal to pass through)} \\
v(n) &= \sin(\omega_1 n) + \sin(\omega_3 n) && \text{(interference signal to be removed)} \\
x(n) &= s(n) + v(n) && \text{(overall input signal).}
\end{aligned}
\tag{5}
$$

The part $v(n)$ is to be removed by the filter, whereas $s(n)$ is to go through unchanged. Using the function **filter** compute (2.5 points) the output $y(n)$ due to $x(n)$ as well as the separate output $y_v(n)$ due to $v(n)$ alone. The usage of the **filter** function for FIR filters is:

```
y = filter(h,1,x);
```

On the same graph, plot (2.5 points) the signals $x(n), s(n), y(n)$ versus n, and observe how $y(n)$ is essentially the delayed version $s(n-2)$, after the filter transients die out (see Fig. 1 for an example). Note that the delayed signal can be generated simply by the command,

```
s2 = sin(w2*(n-2)).*(n>=2);  % delayed signal s(n-2), assuming n=0:100
```

Using at most three **fprintf** commands (no loops), generate (2.5 points) a table of 10 computed values exactly as shown below, and observe the agreement of $s(n-2)$ with $y(n)$, and the vanishing of $y_v(n)$, after the transients die out (they do after 4 sampling instants because you have an FIR filter of order 4).

```
 n      s(n)      s(n-2)       y(n)       v(n)       y_v(n)
-------------------------------------------------------------
 0     0.0000     0.0000      0.0000     0.0000      0.0000
 1     0.3090     0.0000    -50.6056     0.7442    -35.7580
 2     0.5878     0.0000     93.0613     1.2601     67.9497
 3     0.8090     0.3090    -50.2965     1.4050    -35.7580
 4     0.9511     0.5878      0.5878     1.1756      0.0000
 5     1.0000     0.8090      0.8090     0.7071      0.0000
 6     0.9511     0.9511      0.9511     0.2212      0.0000
 7     0.8090     1.0000      1.0000    -0.0600      0.0000
 8     0.5878     0.9511      0.9511     0.0000      0.0000
 9     0.3090     0.8090      0.8090     0.3999      0.0000
```

Problem 1.4 (17.5 points)**.** Although the filter meets its design specifications exactly, it represents a questionable design. To see how unreasonable it is, make a plot (2.5 points) of the magnitude response $|H(\omega)|$ over the entire right-half of the Nyquist interval, $0 \leq \omega \leq \pi$ (see Fig. 1(a) for an example graph). The extremely large values at high-frequencies will amplify any higher frequency components by a very big amount. More importantly, because all signals are typically observed in the presence of random noise, this particular filter will amplify even a tiny amount of noise by a huge amount, completely masking the desired signal and rendering the entire filtering process useless.

It can be shown theoretically, that the RMS value of a zero-mean white-noise input signal will be amplified upon filtering by an amount given by the so-called noise ratio:

$$
\frac{\text{output RMS}}{\text{input RMS}} = \sqrt{\sum_n h^2(n)} = \text{noise ratio,} \tag{6}
$$

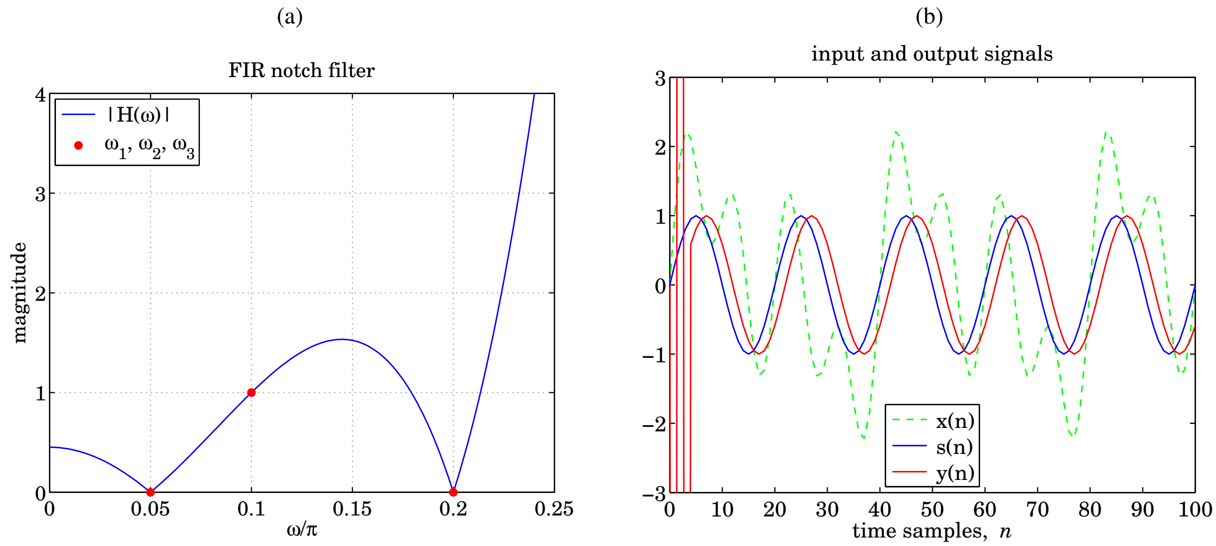

Fig. 1: Plots for Problem 1.2, Problem 1.3, and Problem 1.4.

where $h(n)$ is the impulse response of the (stable LTI) filter. Calculate this quantity for your design (2.5 points).

Other designs that meet the same specifications are possible that do not have the above problem and do not amplify the input noise. As an example, consider the cascade of two second-order infinite-impulse response (IIR) notch filters, designed to have notches at ω_1, ω_3, with a combined response at ω_2 normalized to unity. A particular design (see Appendix (18)) gives the following numerator and denominator coefficients:

$$\begin{aligned}
\mathbf{b} &= [b_0, b_1, b_2, b_3, b_4] = [0.984011, \ -3.535954, \ 5.113142, \ -3.535954, \ 0.984011] \\
\mathbf{a} &= [a_0, a_1, a_2, a_3, a_4] = [1, \ -3.557832, \ 5.093644, \ -3.487380, \ 0.960788]
\end{aligned}$$

where, actually, the vector **b** is a scaled version of impulse response **h** of the above FIR filter. The corresponding transfer function is the ratio of the 4th degree polynomials in z:

$$H(z) = \frac{b_0 + b_1 z + b_2 z^2 + b_3 z^3 + b_4 z^4}{a_0 + a_1 z + a_2 z^2 + a_3 z^3 + a_4 z^4}.$$

In general, IIR filters are defined by specifying the numerator and denominator coefficients, **b**, **a**. Please consult Sections 4.6 and 4.8 of the text for more information on IIR filters and on the MATLAB functions **filter**, **impz**, **freqz**, **phasedelay**, that can be used to compute the output, impulse response, frequency response, and phase-delay of the filter.

Make a plot (5 points) of the magnitude response of this filter in the interval $0 \le \omega \le \pi$, using for example the **freqz** function, and add the three frequency points $\omega_1, \omega_2, \omega_3$ on the graph (see, e.g., Fig. 2),

```
Hmag = abs(freqz(b,a,w));  % magnitude response at a frequency vector w
```

Generate (1 point) the same signals of (5) over the longer interval $0 \le n \le 600$ and plot (1.5 points) $s(n), x(n), y(n)$ over the interval $0 \le n \le 300$, and in another graph, plot $v(n)$ and its filtered output, $y_v(n)$, over the longer time interval $0 \le n \le 600$, so that you can observe the filter transients more clearly (see, e.g., Fig. 2).

The price to pay for this better design is that the transients are much longer. The duration of such transients can be estimated by the so-called 40-dB time constant of the filter, defined as follows in terms of the filter's maximum pole radius, expressed in MATLAB language:

```
n40 = log(0.01) / log(max(abs(roots(a))));
```

Effectively, it represents the time it takes for the transients to drop by a factor of 10^{-2}. Calculate this quantity and verify that it roughly agrees with what you observe in the graph of $y_v(n)$ (2.5 points).

Estimate the noise-ratio of this filter by calculating the sum in (6) over the range $0 \leq n \leq 600$, where the impulse response $h(n)$ can be computed by the function **impz**, and confirm that this filter will have a minor impact on a noise component, neither amplifying nor attenuating it (2.5 points):

```
h = impz(b,a,601);         % calculates h(n) for 0<=n<=600
```

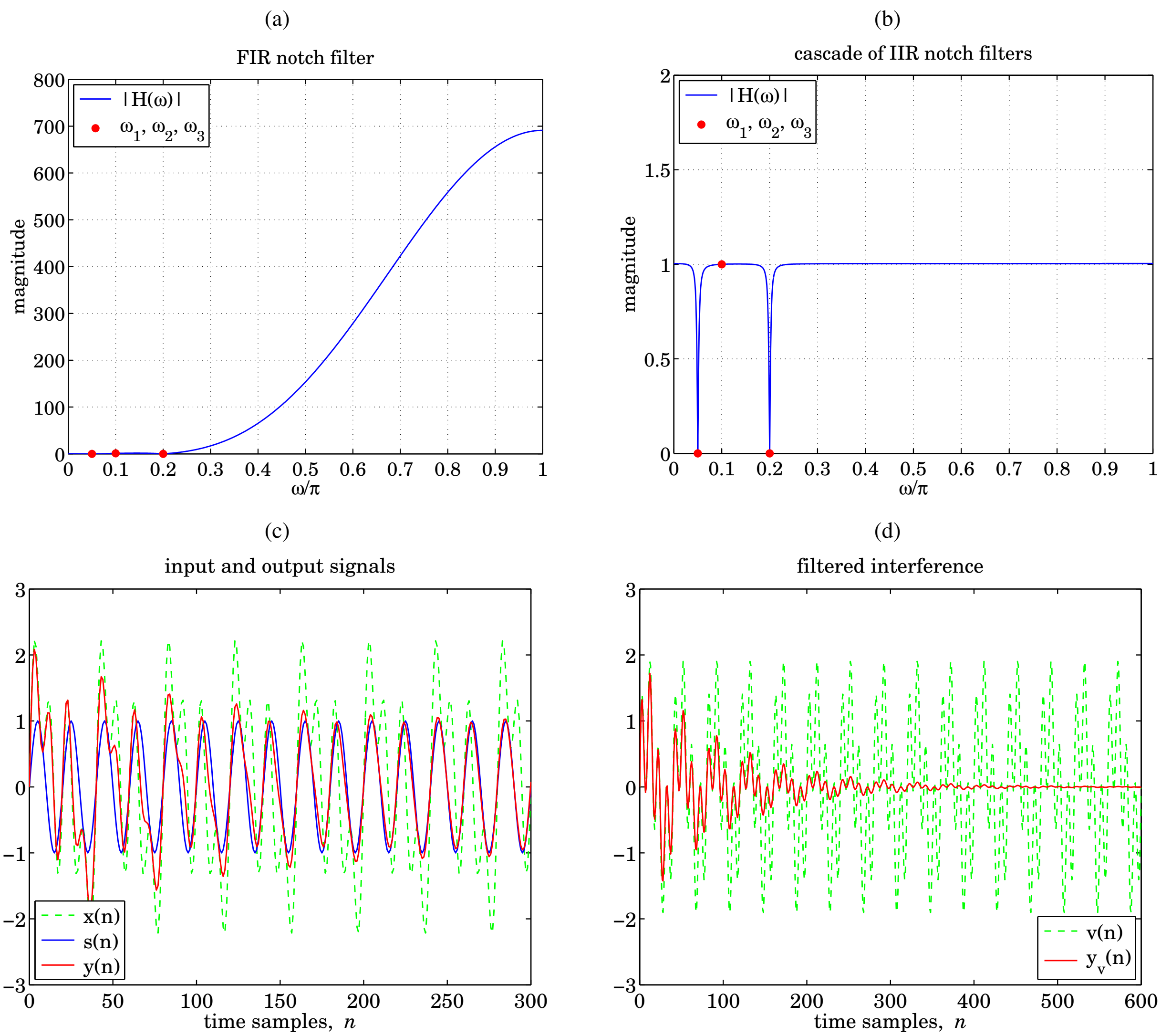

Fig. 2: Plots for Problem 1.4.

2 Phase delay

The FIR filter of the previous problem had linear phase and therefore, its phase-delay was constant independent of frequency. For many filters, the phase delay, $\tau(\omega) = -\angle H(\omega)/\omega$, is a non-trivial function of ω, and therefore, input components of different frequencies suffer different amounts of delays (and attenuations) as they go through the filter.

Consider a peaking filter (see Appendix (13)) with peak frequency ω_0 and bandwidth parameter β. Its derivation will be given later in the course. Here, we simply use the result of (13). The frequency response is obtained by setting $z = e^{j\omega}$ into (13):

$$H(\omega) = \left(\frac{\beta}{1+\beta}\right) \frac{1 - e^{-2j\omega}}{1 - 2\left(\frac{\cos\omega_0}{1+\beta}\right) e^{-j\omega} + \left(\frac{1-\beta}{1+\beta}\right) e^{-2j\omega}}. \tag{7}$$

The corresponding numerator and denominator coefficient vectors are:

$$\mathbf{b} = \frac{\beta}{1+\beta}\,[1\,,\;\; 0\,,\;\; -1]\;, \qquad \mathbf{a} = \left[1\,,\;\; -\frac{2\cos\omega_0}{1+\beta}\,,\;\; \frac{1-\beta}{1+\beta}\right]. \tag{8}$$

Show analytically (you may use MATLAB's symbolic toolbox) that $H(\omega)$ can also be written as (5 points)

$$H(\omega) = \frac{j\beta\sin\omega}{\cos\omega - \cos\omega_0 + j\beta\sin\omega}. \tag{9}$$

It follows that $H(\omega_0) = 1$. Thus, an input sinusoid $x(n) = \sin(\omega_0 n)$ will suffer no phase delay or amplitude modification. From (9), it follows that the phase delay is:

$$\tau(\omega) = -\frac{\angle H(\omega)}{\omega} = -\frac{1}{\omega}\,\arctan\left[\frac{\cos\omega - \cos\omega_0}{\beta\sin\omega}\right]. \tag{10}$$

Next, consider an input sinusoid at some side frequency ω_1 near ω_0:

$$x(n) = \sin(\omega_1 n). \tag{11}$$

The corresponding steady-state output is given by

$$y(n) = H_1\,\sin(\omega_1 n + \theta_1),$$

where, $H_1 = |H(\omega_1)|$, $\theta_1 = \text{Arg}H(\omega_1)$, or, $H(\omega_1) = H_1\,e^{j\theta_1}$.

Now consider the following four problems, example plots for which are given in Fig. 3.

Problem 2.1 (5 points)**.** Consider the following values, $\omega_0 = 0.2\pi$, $\beta = 0.1$, and $\omega_1 = 0.05\pi$. Make a plot (2.5 points) of the magnitude response of the filter, $\big|H(\omega)\big|$, over the range $0 \le \omega \le \pi$ and add the peak and side frequency points ω_0, ω_1 on the graph (see example graph below).

Moreover, calculate the left and right 3-dB frequencies of the filter using (15) of the Appendix, and place them on the graph at the 3-dB level (2.5 points).

Problem 2.2 (5 points)**.** Make a plot of the phase-delay, $\tau(\omega)$, over $0 \le \omega \le \pi$ and place the points ω_0, ω_1 on the graph.

Problem 2.3 (7.5 points)**.** Generate the input signal $x(n)$ of (11) over the time range (1.5 points), $0 \le n \le 100$, and calculate the corresponding output using the function **filter** (1 point):

```
y = filter(b,a,x);
```

Make a stem plot (1.5 points), as well as a regular plot (1 point), of the signals $x(n), y(n)$ versus n (see example graphs below). By counting the bars on the stem plot, estimate the amount of time-advance or time-delay suffered by the ω_1 sinusoidal input, and compare it with the theoretical value $\tau(\omega_1)$ (2.5 points). Note that the exact delay need not be an integer as your counting of bars would imply.

Problem 2.4 (7.5 points)**.** Repeat Problems 2.1–2.3 for the case $\omega_1 = 0.3\pi$.

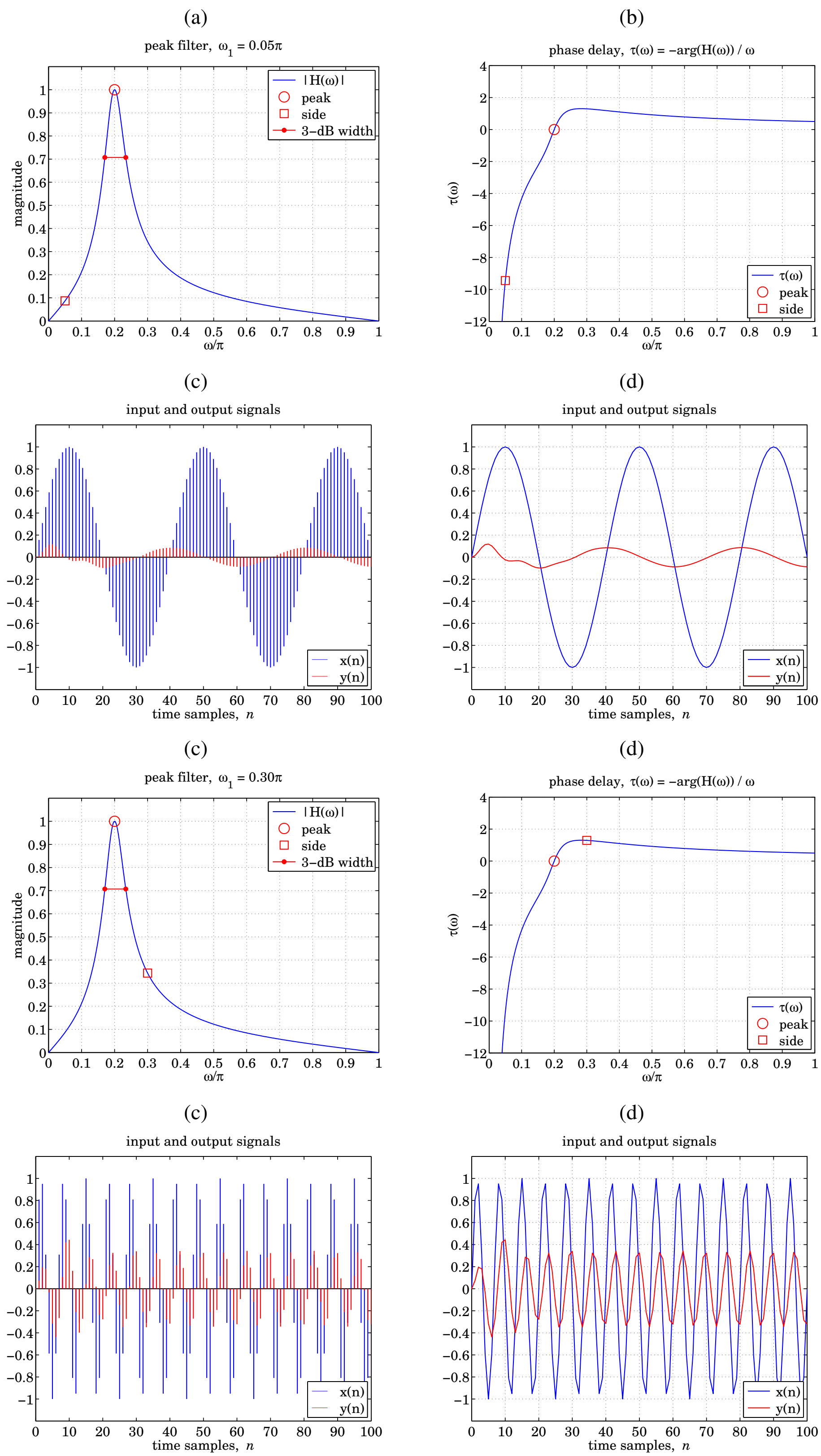

Fig. 3: Plots for Problem 2.1–2.4.

3 Sampling and aliasing

The aim of this part is to demonstrate the effects of aliasing arising from improper sampling. A given analog signal $x(t)$ is sampled at a rate $f_s = 1/T$, the resulting samples $x(nT)$ are then reconstructed by an *ideal* reconstructor into the analog signal $x_a(t)$. Improper choice of f_s will result in different signals $x_a(t) \neq x(t)$, even though they agree at their sample values, that is, $x_a(nT) = x(nT)$. The procedure is illustrated by the following block diagram:

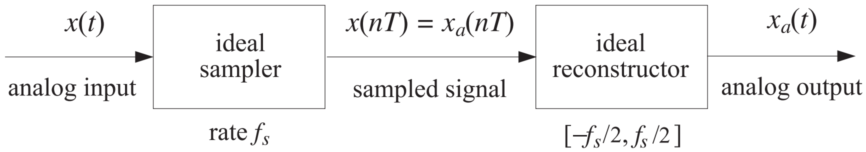

Fig. 4: Block diagram showing the sampling and reconstruction operations.

Problem 3.1 (10 points)**.** Consider an analog signal $x(t)$ consisting of three sinusoids of frequencies of 1, 4, and 6 kHz:

$$x(t) = \cos(2\pi t) + \cos(8\pi t) + \cos(12\pi t)$$

where t is in milliseconds. Show that if this signal is sampled at a rate of $f_s = 5$ kHz, it will be aliased with the following signal, in the sense that their sample values will be the same (5 points):

$$x_a(t) = 3\cos(2\pi t).$$

On the same graph, plot (2.5 points) the two signals $x(t)$ and $x_a(t)$ versus t in the range $0 \leq t \leq 2$ msec. To this plot, add the time samples $x(nT)$ (2.5 points) and verify that $x(t)$ and $x_a(t)$ intersect precisely at these samples (see Fig. 5 for example).

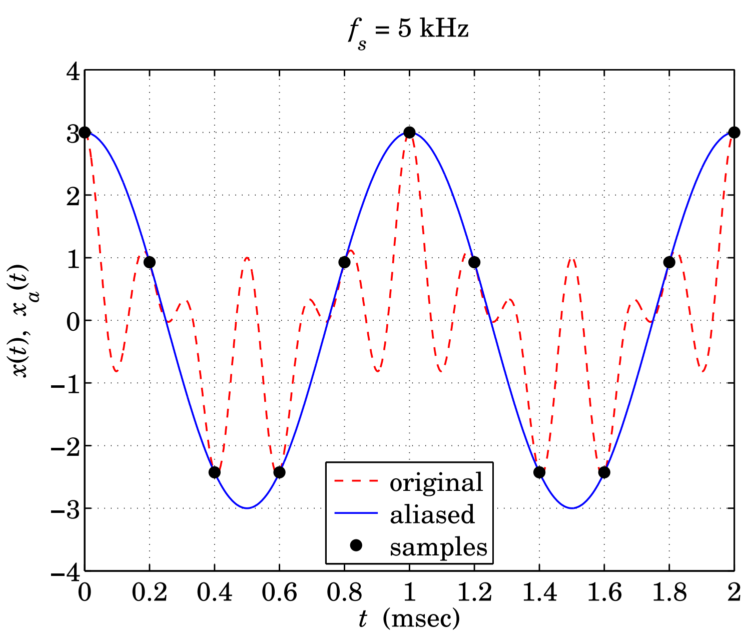

Fig. 5: Plot for Problem 3.1.

Problem 3.2 (10 points)**.** Repeat part (a) with $f_s = 10$ kHz. In this case, determine the signal $x_a(t)$ with which $x(t)$ is aliased (5 points).

Plot both $x(t)$ and $x_a(t)$ on the same graph (2.5 points) over the same range $0 \leq t \leq 2$ msec, and add the samples (2.5 points) $x(nT)$. Verify again that the two signals intersect at the sampling instants (see Fig. 6 for example).

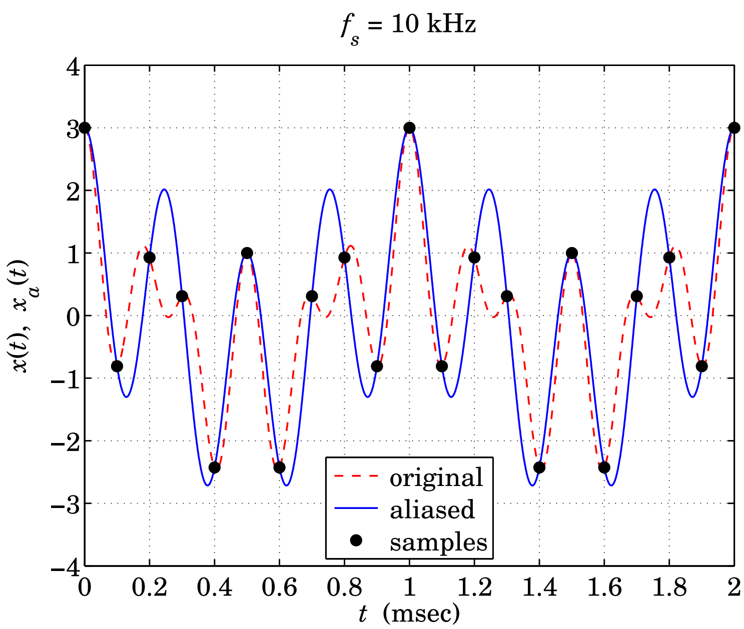

Fig. 6: Plot for Problem 3.2.

Appendix

Notch filter with notch at ω_0 and 3-dB width $\Delta\omega$:

$$H(z) = \left(\frac{1}{1+\beta}\right) \frac{1 - 2\cos\omega_0\, z + z^2}{1 - 2\left(\frac{\cos\omega_0}{1+\beta}\right) z + \left(\frac{1-\beta}{1+\beta}\right) z^2}, \tag{12}$$

Peak filter with peak at ω_0 and 3-dB width $\Delta\omega$:

$$H(z) = \left(\frac{\beta}{1+\beta}\right) \frac{1 - z^2}{1 - 2\left(\frac{\cos\omega_0}{1+\beta}\right) z + \left(\frac{1-\beta}{1+\beta}\right) z^2}, \tag{13}$$

where in both cases, β is expressible in terms of the 3-dB width by

$$\beta = \tan\left(\frac{\Delta\omega}{2}\right). \tag{14}$$

The left and right 3-dB frequencies are computable from the equations:

$$\cos\omega_{\text{left}} = \frac{\cos\omega_0 + \beta\sqrt{\beta^2 + \sin^2\omega_0}}{1+\beta^2} \tag{15}$$

$$\cos\omega_{\text{right}} = \frac{\cos\omega_0 - \beta\sqrt{\beta^2 + \sin^2\omega_0}}{1+\beta^2}. \tag{16}$$

Note that the 3-dB frequencies satisfy $\Delta\omega = \omega_{\text{right}} - \omega_{\text{left}}$, and are the frequencies at which the filter magnitude-squared is down by half from its peak value, that is, they satisfy, $\left|H(\omega)\right|^2 = 1/2$, which corresponds to a 3-dB drop, indeed, $10\log_{10}(1/2) = -3.01$ dB.

Cascade of two notch filters with notches at ω_1, ω_3, with common bandwidth, and unity gain at ω_2:

$$H(z) = G \cdot \frac{1 - 2\cos\omega_1\, z + z^2}{1 - 2\left(\frac{\cos\omega_1}{1+\beta}\right) z + \left(\frac{1-\beta}{1+\beta}\right) z^2} \cdot \frac{1 - 2\cos\omega_3\, z + z^2}{1 - 2\left(\frac{\cos\omega_3}{1+\beta}\right) z + \left(\frac{1-\beta}{1+\beta}\right) z^2} \tag{17}$$

with G found from the condition, where $z_2 = e^{j\omega_2}$,

$$\left| G \cdot \frac{1 - 2\cos\omega_1\, z_2 + z_2^{-2}}{1 - 2\left(\frac{\cos\omega_1}{1+\beta}\right) z_2 + \left(\frac{1-\beta}{1+\beta}\right) z_2^{-2}} \cdot \frac{1 - 2\cos\omega_3\, z_2 + z_2^{-2}}{1 - 2\left(\frac{\cos\omega_3}{1+\beta}\right) z_2 + \left(\frac{1-\beta}{1+\beta}\right) z_2^{-2}} \right| = 1.$$

The 4th order numerator and denominator coefficients of the combined filter are obtained by convolving those of the individual sections, i.e.,

$$\mathbf{b} = G \cdot \text{conv}\big([1,\ -2\cos\omega_1,\ 1],\ \ [1,\ -2\cos\omega_3,\ 1]\big) \tag{18}$$

$$\mathbf{a} = \text{conv}\left(\left[1\,,\ -\frac{2\cos\omega_1}{1+\beta}\,,\ \frac{1-\beta}{1+\beta}\right],\ \left[1\,,\ -\frac{2\cos\omega_3}{1+\beta}\,,\ \frac{1-\beta}{1+\beta}\right]\right). \tag{19}$$