ECE2111 laboratory 5: Sampling, Aliasing and Reconstruction

Prepared by: Dr. Faezeh Marzbanrad

Aim

The aims of this lab are

1. to become familiar with sampling principles for signal processing.
2. to design anti-aliasing filters for appropriate down-sampling and reconstruction.

Introduction

Sampling is a crucial step in most signal and image processing applications. Sampling must be performed appropriately to avoid aliasing and to ensure that signals and images can be reconstructed faithfully. The sampled signal is useful only if it contains the same information as the original signal, which requires sampling at a high enough rate (or with a smaller enough sampling period). According to the Nyquist Sampling Theorem, a real signal with a bandlimited spectrum of W rad/sec (where $X(\omega) = 0$ for $|\omega| > W$) can be reconstructed from its samples taken uniformly with period $T_s < \frac{\pi}{W}$. It is sometimes easier to restate this result in Hz. If $D = W/2\pi$ is the bandwidth in Hz then a signal with bandlimited spectrum of D Hz can be reconstructed from its samples taken uniformly with sampling rate f_s samples/sec as long as $f_s > 2D$.

Since we always work with discrete-time signals in the lab, we can study the effect of different sampling rates by *downsampling* a signal. For instance, if we wanted to downsample a discrete-time signal x_s by a factor of L, this means defining a new signal

$$x_{ds}[n] = x_s[Ln] \quad \text{for all } n.$$

The process of downsampling by factor L can be thought of as keeping every Lth sample. Suppose x_s was obtained by sampling a continuous-time signal with sampling period T_s, i.e.,

$$x_s[n] = x(nT_s) \quad \text{for all } n.$$

If we downsample x_s by a factor of L to obtain x_{ds}, this is equivalent to sampling the original continuous-time signal L times slower, i.e.,

$$x_{ds}[n] = x_s[Ln] = x(nLT_s)$$

which is the same as sampling the continuous-time signal x with sampling period LT_s.

In this lab you will first down-sample an audio signal to study the effect of re-sampling and aliasing. You will then design an appropriate anti-aliasing filter to apply before down-sampling. This should allow reconstruction of the audio recording without distortion.

Scheduling

This lab runs over two weeks (weeks ten and eleven) and is due at the end of the lab session in week 11. It is deliberately a fairly short lab to give you some additional time to work on the assignment for ECE2111.

Prelab

There are three prelab questions in these lab notes. By the start of week 10 (for the precise due date see Moodle), read through the lab document, find the prelab questions, and answer them.

Submit your answers to the prelab questions via the Moodle quiz called 'prelab 5' on the Moodle page. You are expected to do this individually.

Results document

You are required to organize your code and outputs in what we will call a "results document". This must be created following the guidelines in the accompanying file "Formatting requirements for lab results document". Submit this by the end of week 11 (for the precise due date see Moodle) via the 'results document' assignment link on Moodle. **Your submission must reflect your own work!**

End-of-lab quiz

There is a timed, end-of-lab quiz that must be completed by the end of week 11 (for the precise due date see Moodle). **Please do not start this quiz until you are ready.** This quiz tests your understanding of the lab material. **It must be completed individually.**

Average power

Recall that for a periodic continuous-time signal $x(t)$ with fundamental period T_0, the *average power per period* is given by

$$\frac{1}{T_0}\int_0^{T_0} |x(t)|^2\, dt.$$

By Parseval's relation, the average power per period can also be computed as the sum of the squared magnitudes of the Fourier coefficients X_k of x. In other words

$$\frac{1}{T_0}\int_0^{T_0} |x(t)|^2\, dt = \sum_{k=-\infty}^{\infty} |X_k|^2.$$

Computing average power in MATLAB

Recall that a length N vector `x` in MATLAB can be iterpreted as a discrete-time signal x that is periodic with fundamental period N by taking x to be the periodic extension of `x`. The average power (per period) of x is then

$$\frac{1}{N}\sum_{n=0}^{N-1} |x[n]|^2.$$

This can be computed in MATLAB using the command

```
P = (x'*x)/N;
```

Prelab question 1

Find a formula (in terms of A) for the average power per period of the signal

$$x(t) = A\cos(\omega_0 t).$$

In the first question in the prelab Moodle quiz, enter the average power per period for the signal $x(t) = 4\cos(\omega_0 t)$.

Downsampling a sinusoid

Let $x(t) = \cos(5000 \times 2\pi t)$ be a sinusoid with frequency 5000 Hz. If we sample x with sampling frequency 44100 samples/second (i.e., sampling period $T_s = 1/44100$ seconds/sample) to obtain the discrete-time signal

$$x_s[n] = \cos(5000 \times 2\pi n T_s)$$

then aliasing does not occur because $5000 \times 2\pi \leq \pi/T_s = 44100\pi$.

Prelab question 2

Suppose we downsample x_s by a factor of L (equivalent to sampling x with sampling period $T_{ds} = L/44100$) to obtain

$$x_{ds}[n] = \cos(5000 \times 2\pi \times nT_{ds}).$$

Find the largest positive whole number L for which aliasing does not occur. (In other words, find the largest positive whole number L for which no aliasing occurs when we sample x with sampling period $L/44100$.) Enter your answer in the Moodle quiz.

Downsampling and reconstruction in MATLAB

Suppose we have a vector `xs` of length `N` in MATLAB. We can downsample it by a factor of `L` by using the command

```
xds = xs(1:L:end);
```

The length of `xds` is now $\lfloor N/L \rfloor$, the greatest integer less than or equal to N/L.

We can reconstruct an approximation to `xs` by using linear interpolation as follows

```
xrecon = interp1(1:L:N,xds,1:N,'linear',0);
```

The first argument specifies the time indices corresponding to the values in the second argument. The third argument specifies the desired time indices for the reconstructed signal. The fourth argument specifies the interpolation method (other methods are also possible). The fifth argument specifies what to do at the end of the signal (where extrapolation is required). In this case we choose just to append zeros. For more information type `help interp1` in MATLAB.

1 Audio signal sampling

Write a MATLAB script to carry out the following tasks:

1. Load the signal `speech` into your workspace by loading the file `lab5data.mat`. Note that the sampling rate of these signals is 44100 samples/second.
2. Find the average power of the signal `speech`.
3. Create a new signal `speechNoisy` by adding a signal of the form

   $$A\cos(2\pi \times 5000n/44100)$$

   to the signal `speech`. This corresponds to adding a sinusoid of frequency 5000Hz in continuous-time and then sampling at sampling rate 44100 samples/second. Choose the amplitude A so that the power of the sinusoidal disturbance is one tenth of the power of the signal `speech`.

4. Play the signal `speechNoisy` using `soundsc()`. You should be able to hear the sinusoidal noise in the background. Plot the noisy signal in frequency domain (magnitude only) with frequency axis in rad/sample. Use the second type of plot discussed in lab 4.

5. Downsample the noisy signal by a factor of 5, i.e., take every fifth sample. Play the down-sampled signal using `soundsc` (make sure you adjust the sampling frequency appropriately when you call `soundsc`). Does the sinusoidal noise sound different after downsampling?

6. Now reconstruct the original signal by linear interpolation. You can do this using the MATLAB function `interp1`.

7. Play the reconstructed signal using `soundsc`. How does it compare with the original signal `speechNoisy`? Plot the reconstructed signal in the frequency domain (magnitude only) with frequency axis in rad/sample. Use the second type of plot discussed in lab 4. Can the reconstruction process undo the effect of aliasing from the down-sampled signal?

Once you have finished this section:

- copy your script into your results document
- copy your plots into your results document.
- Report the average power of the `speech` signal (item 2) and your choice of A in item 3. Answer the questions in items 5 and 7 in your results document.

Downsampling and anti-aliasing filters

In lectures, we disucssed using an anti-aliasing filter before sampling a continuous-time signal. This is a low-pass filter with cut-off frequency (at most) the Nyquist frequency π/T_s rad/second (where T_s is the sampling period). A variation on this idea is used when we down-sample a discrete-time signal.

Let x_s be a discrete-time signal obtained by sampling a continuous-time signal with sampling period T_s. If we downsample x_s by a factor of L to get $x_{ds}[n] = x_s[nL]$, then the new signal could have equivalently been obtained by sampling the continuous-time signal with sampling period LT_s.

If we were to design an anti-aliasing filter to use before sampling with sampling period LT_s, we would ensure its cut-off frequency were at most $\pi/(LT_s)$ rad/second (the corresponding Nyquist frequency). Instead, in this lab we will design a discrete-time anti-aliasing filter that will be used before we downsample x_s by a factor of L to obtain x_{ds}. We want this discrete-time anti-aliasing filter to have cut-off frequency that corresponds to the continuous-time frequency $\pi/(LT_s)$. In other words, the maximum cutoff frequency should be π/L rad/sample.

Prelab question 3

Suppose $T_s = 1/44100$ and $L = 5$. What should be the maximum cut-off frequency of the anti-aliasing filter we use before downsampling? Express your answer as a fraction of π (correct to one decimal place). In other words, if the desired cutoff frequency is $W\pi$ rad/sample, enter W as your answer.

2 Designing an anti-aliasing filter

Write a MATLAB script to carry out the following tasks.

1. Let `speechNoisy` be the signal you constructed in section 1. Given that we are about to downsample `speechNoisy` by a factor of 5, determine an appropriate choice of cut-off frequency (in *rad/sample*) for an anti-aliasing FIR filter.

2. Use `h = firpm` to design the impulse reponse of an anti-aliasing filter with the appropriate cut-off frequency. You will need to choose a filter order and the location of the passband edge. Use `freqz(h,1)` to plot the magnitude and phase response of the filter you have designed and make sure it behaves as you want it to.

3. Filter `speechNoisy` using your filter coefficients (and the `conv` function) to obtain a filtered signal `speechNoisyAa`. Note that the resulting filtered signal will be longer than `speechNoisy`, and so you will need to use this new length when you work with this signal.

4. Plot `speechNoisyAa` in the frequency domain (magnitude only) with frequency axis in rad/sample. Use the second type of plot discussed in lab 4.

5. Downsample `speechNoisyAa` by a factor of 5.

6. Play the down-sampled signal and plot it in the frequency domain (magnitude only) with frequency axis in rad/sample. Use the second type of plot discussed in lab 4.

7. Now reconstruct the original signal by linear interpolation using `interp1` in MATLAB. Play the reconstructed signal. How does it compare with the original signal `speech` and with the reconstructed signal without anti-aliasing from section 1? Plot the reconstructed signal in the frequency domain (magnitude only) with frequency axis in rad/sample. Use the second type of plot discussed in lab 4.

Once you have finished this section:

- copy your script into your results document
- copy your plots into your results document.
- Report your choice of cut-off frequency in rad/sample from item 1. Answer the question in item 7 in your results document.

Once you have completed the lab tasks and understand them, you are ready for the end-of-lab Moodle quiz. This quiz is closely based on the lab tasks. It must be completed **individually**.You may use MATLAB. Unlike the prelab, you only have **one attempt at each question**.

Assessment

This lab is marked out of 12. Your mark is based on the following:

- **Prelab:** Correct responses to the three prelab questions submitted via the prelab Moodle quiz (3 marks, 1 per question)

- **Results document:** (4 marks, 3 marks for content and 1 mark for presentation)
  Marks per section: Each of the 2 sections is marked out of 1.5 (3 marks):
  - 0 marks if not attempted
  - 0.5 marks if an attempt is made, but many errors or much of the section is incomplete.

  - 1 mark if a reasonable attempt is made, but has clear flaws
  - 1.5 mark if no errors or possibly very minor errors

  Presentation: (1 mark)

  - Results document adheres to the formatting requirements (1 mark)
  - Results document mostly adheres to formatting requirements, but not completely (0.5 mark)
  - Results document rarely adheres to formatting requirements (0 marks)

- **End-of-lab quiz:** Correct responses to the end-of-lab Moodle questions (5 marks, 1 per question).