COM-303 - Signal Processing for Communications

Homework #9

Exercise 1. Zero Order Hold

Consider a discrete-time sequence $x[n]$ with DTFT $X(e^{j\omega})$. Next, consider the continuous-time interpolated signal

$$x_0(t) = \sum_{n=-\infty}^{\infty} x[n] \mathrm{rect}(t-n)$$

i.e. the signal interpolated with a zero-centered zero-order hold and $T_s = 1$ sec.

(a) Express $X_0(j\Omega)$ (the spectrum of $x_0(t)$) in terms of $X(e^{j\omega})$.

(b) Compare $X_0(j\Omega)$ to $X(j\Omega)$, i.e. the Fourier transform of the sinc interpolation of $x[n]$ (always with $T_s = 1$):

$$x(t) = \sum_{n\in\mathbb{Z}} x[n] \mathrm{sinc}(t-n).$$

Comment on the result: you should point out two major problems.

(c) The signal $x(t)$ can be obtained from the zero-order hold interpolation $x_0(t)$ as $x(t) = x_0(t) * g(t)$ for some filter $g(t)$. Sketch the frequency response of $g(t)$.

(d) Propose two solutions (one in the continuous-time domain, and another in the discrete-time domain) to eliminate or attenuate the distortion due to the zero-order hold. Discuss the advantages and disadvantages of each.

Exercise 2. Another View of Sampling

One of the standard ways of describing the sampling operation relies on the concept of "modulation by a pulse train". Choose a sampling interval T_s and define a continuous-time pulse train $p(t)$ as:

$$p(t) = \sum_{k=-\infty}^{\infty} \delta(t - kT_s).$$

The Fourier Transform of the pulse train is

$$P(j\Omega) = (2\pi/T_s) \sum_{k=-\infty}^{\infty} \delta(\Omega - k(2\pi/T_s))$$

This is tricky to show, so just take the result as is. The "sampled" signal is simply the modulation of an arbitrary-continuous time signal $x(t)$ by the pulse train:

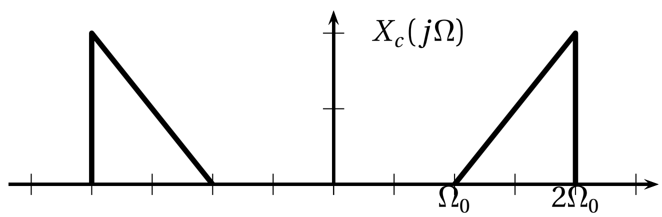
$$x_s(t) = p(t)\,x(t)$$

Note that now, this sampled signal is still continuous time but, by the properties of the delta function, is non-zero only at multiples of T_s; in a sense, $x_s(t)$ is a discrete-time signal brutally embedded in the continuous time world.

Derive the Fourier transform of $x_s(t)$ and show that if $x(t)$ is bandlimited to π/T_s then we can reconstruct $x(t)$ from $x_s(t)$.

Exercise 3. Aliasing Can Be Good!

Consider a real, continuous-time signal $x_c(t)$ with the following spectrum:



(a) What is the bandwidth of the signal? What is the minimum sampling period in order to satisfy the sampling theorem?

(b) Take a sampling period $T_s = \pi/\Omega_0$; clearly, with this sampling period, there will be aliasing. Plot the DTFT of the discrete-time signal $x_a[n] = x_c(nT_s)$.

(c) Suggest a block diagram to reconstruct $x_c(t)$ from $x_a[n]$.

(d) With such a scheme available, we can therefore exploit aliasing to reduce the sampling frequency necessary to sample a bandpass signal. In general, what is the minimum sampling frequency to be able to reconstruct with the above strategy a real signal whose frequency support on the positive axis is $[\Omega_0, \Omega_1]$ (with the usual symmetry around zero, of course)?

Exercise 4. Sampling and Aliasing

Consider the linear chirp defined as

$$x(t) = cos(\varphi(t)) = cos(\pi a t^2 + 2\pi b t + \theta)$$

where a, b and θ are real-valued parameters.

(a) What can you say about the instantaneous frequency of the signal $x(t)$? In particular, if we sample the signal what can you expect at the reconstruction?

(b) Write in Python a function `chirp(a,b,sf)` that computes the sampled version of a chirp (with initial phase $\theta = 0$) at a sampling frequency `sf` for $0 \le t \le 1$ second.

(c) Look at a typical shape of a chirp before aliasing by typing `chirp(40,4,400)`. Now type `chirp(400,4,400)` and explain what you observe.

(d) Consider the case where $a = 50$ and $b = 600$. What is the minimum sampling frequency so that aliasing does not occur before $t = 1$ second?

(e) Play the chirp using the Matlab command `sound` with $a = 40$, $b = 4$ and $sf = 900$ on a duration of 10 seconds (in order to hear something). Repeat the operation with $sf = 700$. What can you conclude?

Exercise 5. Digital processing of continuous-time signals

For your birthday, you receive an unexpected present: a 4 MHz A/D converter, complete with anti-aliasing filter. This means you can safely sample signals up to a frequency of 2 MHz; since this frequency is above the AM radio frequency band, you decide to hook up the A/D to your favorite signal processing system and build an entirely digital radio receiver. In this exercise we will explore how to do so.

Simply, assume that the AM radio spectrum extends from 1 Mhz to 1.2 Mhz and that in this band you have ten channels side by side, each one of which occupies 20 KHz.

(a) Sketch the digital spectrum at the output of the A/D converter, and show the bands occupied by the channels, numbered from 1 to 10, with their beginning and end frequencies.

The first thing that you need to do is to find a way to isolate the channel you want to listen to and to eliminate the rest. For this, you need a bandpass filter centered on the band of interest. Of course, this filter must be tunable in the sense that you must be able to change its spectral location when you want to change station. An easy way to obtain a tunable bandpass filter is by modulating a low-pass filter with a sinusoidal oscillator whose frequency is controllable by the user:

(b) As an example of a tunable filter, assume $h[n]$ is an ideal low-pass filter with cutoff frequency $\frac{\pi}{8}$. Plot the magnitude response of the filter $h_m[n] = cos(\omega_m n)h[n]$, where $\omega_m = \frac{\pi}{2}$; ω_m is called the *tuning frequency*.

(c) Specify the cutoff frequency of a low-pass filter which can be used to select one of the AM channels above.

(d) Specify the tuning frequencies for channel 1, 5 and 10.

Now that you know how to select a channel, all that is left to do is to demodulate the signal and feed it to a D/A converter and to a loudspeaker.

(e) Sketch the complete block diagram of the radio receiver, from the antenna going into the A/D converter to the final loudspeaker. Use only one sinusoidal oscillator. Do not forget the filter before the D/A (specify its bandwidth).

The whole receiver now works at a rate of 4 MHz; since it outputs audio signals, this is clearly a waste.

(f) Which is the minimum D/A frequency you can use? Modify the receiver's block diagram with the necessary elements to use a low frequency D/A

Exercise 6. Aliasing in Time?

Consider an N - periodic discrete-time signal $\tilde{x}[n]$, with N an *even* number, and let $\tilde{X}[k]$ be its DFS:

$$\tilde{X}[k] = \sum_{n=0}^{N-1} \tilde{x}[n] e^{-j\frac{2\pi}{N}nk} \qquad k \in \mathbb{Z}$$

Let $\tilde{Y}[m] = \tilde{X}[2m]$, i.e. a "subsampled" version of the DFS coefficients; clearly this defines a $\frac{N}{2}$-periodic sequence of DFS coefficients. Now consider the $\frac{N}{2}$ -point inverse DFS of $\tilde{Y}[m]$ and call this $\frac{N}{2}$-periodic signal $\tilde{y}[n]$:

$$\tilde{y}[n] = \frac{2}{N} \sum_{k=0}^{N/2-1} \tilde{Y}[k] e^{j\frac{2\pi}{N/2}nk} \qquad n \in \mathbb{Z}$$

Express $\tilde{y}[n]$ in terms of $\tilde{x}[n]$ and describe in a few words what has happened to $\tilde{x}[n]$ and why.

Exercise 7. Other interpolators

Consider the continuous-time signal

$$x(t) = \sum_{n=-\infty}^{\infty} x[n] I\left(\frac{t - nT_s}{T_s}\right)$$

obtained by interpolating a discrete-time signal $x[n]$ with the following triangular interpolator:

$$I(t) = \begin{cases} 1 - 2|t| & \text{for } |t| \le \frac{1}{2} \\ 0 & \text{otherwise} \end{cases}$$

Assume the discrete-time signal's spectrum is

$$X(e^{j\omega}) = \begin{cases} 1 & \text{for } |\omega| \le \frac{2\pi}{3} \\ 0 & \text{otherwise} \end{cases}$$

(with the usual 2π-periodicity). Sketch the Fourier transform $X(j\Omega)$ of the interpolated signal $x(t)$.

Exercise 8. Time and Frequency

Consider a real continuous-time signal $x(t)$. All you know about the signal is that $x(t)=0$ for $|t|>t_0$. Can you determine a sampling frequency F_s so that when you sample $x(t)$, there is no aliasing? Explain.
