Lab 3 - Sampling and reconstruction

Objectives: In this lab we will use discrete-time samples of continuous-time signals and perform empirical reconstruction using various methods.

3.1 A signal and its samples (script)

Consider the continuous-time signal $x(t) = \sin(5\pi t) + \cos(10\pi t)$ for analysis. Since continuous-time signals cannot be exactly represented in Matlab, we will use a very fine time-grid to approximate continuous nature of time. Let `t_fine = 0:0.001:2` be the time-grid for representing continuous time-signals (note that in this session `t_fine` is a proxy for continuous-time).

>> Plot this signal as a function of time using the `plot()` command. The time axis should be from 0 to 2s. In this lab we will restrict to the time interval [0, 2] and the time vector `t_fine` will be used in all the tasks below. You should use `plot(t,x)` instead of just `plot(x)` to get appropriate markings on the time axis, else matlab will default to positive integer markings (vector index) which is not informative. Same applies for the `stem()` plots below.

>> Let this signal be sampled with sampling interval Ts = 0.1s and denote the discrete-time signal as $x[n] = x(nT_s)$. In the same figure above, plot the samples $x[n]$ using the `stem()` command in the time interval [0, 2]. Plotting would be easier if you generate the time vector corresponding to the location of the samples: `t_samples = 0:Ts:2`. Use appropriate `t_samples` in the tasks below as well.

3.2 Reconstruction methods (script + function)

We will make use of the `interp1()` matlab function for reconstruction of continuous-time signal from samples. Read up the documentation and examples for how to use this function.

For this script, repeat part 3.1 above and plot it in the top-left panel of a figure with 2x2 subplots. In each of the following, perform reconstruction as indicated and plot the samples and reconstructed signal in the remaining panels.

a) From the samples $x[n]$, perform *zero-order hold* reconstruction of $x(t)$. Use `interp1()` command to get this signal with appropriate selection of the 'method'. Note that the reconstruction signal should be computed over the time-grid `t_fine`.
b) From the samples $x[n]$, perform *linear interpolation* based signal reconstruction of $x(t)$. Use the `interp1()` command.
c) Recall that the ideal reconstruction using sinc function is given by the formula

$$x_r(t) = \sum_{n=-\infty}^{\infty} T_s\, x(nT_s) \frac{\sin(\omega_c(t - nT_s))}{\pi(t - nT_s)} \qquad \dots (1)$$

Though this is an infinite sum and cannot be exactly implemented in a computer, we will approximately implement it by restricting to the time interval [0, 2] and using only the samples $x[n]$ we have from that interval.

>> From the samples $x[n]$, find the approximate *sinc interpolated* signal as given by the above formula. For sinc interpolation use a cut-off frequency of $\omega_c = \frac{\omega_s}{2}$ where $\omega_s = \frac{2\pi}{T_s}$. You can use the inbuilt `sinc()` command [with appropriate time scaling to get required cut-off frequency]. Read up the documentation for this function before using it. For computations, restrict each of the sinc in the summation to the interval [0, 2].

>> Hint: It might be easier to easier to write a matlab function `sinc_interp()` as we have to do this interpolation in the later tasks as well. The inputs and outputs of this function should be as follows:

```
function xr = sinc_interp(n,xn,Ts,t_fine)
% n - the integer locations of the samples x[n]
% xn - the sampled signal x[n] = x(n*Ts)
% Ts - the sampling interval
% t_fine - the time-grid for reconstruction of xr
% xr - the reconstructed signal over the time-grid t_fine
```

>> How does the quality of reconstruction vary within the interval [0, 2]? Give explanation for your observations.

>> For each of the three interpolation methods above, compute the maximum absolute error (MAE) between the original signal and the reconstructed signal in the interval [0.5, 1.5].

>> (Optional): try some of the other interpolation 'method' available in the command `interp1()` and check how the quality of reconstruction and the MAE changes.

3.3 Sampling non-band-limited signal (script)

We know that sampling theorem can be applied only for band-limited signals. All the above tasks had band-limited signals. We now consider a non-band-limited signal and investigate its reconstruction as sampling interval Ts is changed.

>> Consider the continuous-time triangular pulse signal of height 1, base in the interval [-1,1], and zero otherwise. Because this is a time-limited signal, only finite number of terms appear in the reconstruction formula (1) above (though the sinc shape is still infinite in time extent).

>> For a sampling interval of Ts, what is the corresponding `t_samples` vector so that we only sample the base of the triangle (assume there is a sample at -1)? Generate the corresponding samples $x[n]$ and the discrete-time indices $\boldsymbol{n}$. Use these as inputs below.

>> Perform sinc interpolation for this signal using samples generated for i) Ts = 0.5s, ii) Ts = 0.2s, and iii) Ts = 0.1s and iv) Ts = 0.05s. For the purpose of reconstruction, use a time-grid of `t_fine = -10:0.001:10`.

>> Create a figure with 2x2 subplots, one panel for each Ts. In each panel plot the samples and the reconstructed signal corresponding to the four sampling intervals. What are your observations as sampling interval is changed?

3.4 Aliasing (script)

Aliasing occurs when the sampling frequency is less than the Nyquist rate required by the sampling theorem. In this part we will look at the effect of aliasing for the signal $x(t) = \sin(5\pi t)$.

>> What is the Nyquist rate for this $x(t)$?

>> Consider samples of $x(t)$ for the following sampling intervals i) Ts = 0.1s, ii) Ts = 0.2s, iii) Ts = 0.3s, and iv) Ts = 0.4s. For each of these cases perform sinc interpolation from samples over the interval [0, 2].

>> Create a figure with 2x2 subplots, one panel for each Ts. In each panel plot the samples and the reconstructed signal corresponding to the four sampling intervals. What are your observations as sampling interval is changed?