EE430 Term Project Part 1

In this part of the project, you are going implement a computer program that computes and displays the short-time Fourier transform (STFT) of a time-domain signal. The STFT is a sequence of Fourier transforms of a windowed signal and can be used to determine the frequency content of local sections of a signal. Our textbook discusses the STFT and related topics in Chapter 10 entitled "Fourier Analysis Using Discrete Fourier Transform". In the textbook, STFT is also referred to as "Time Dependent Fourier Transform". Please read the first 4 sections from this chapter. The program you will implement has to be able to work with various input signals such as computer-generated data, audio files that are already existing in the computer storage and sounds that are recorded by the user from a microphone.

Prepare a descriptive and clearly written (in language and format) report. You are going to upload this report and the MATLAB code to ODTÜClass. The sections below describe the tasks in more detail.

Data acquisition

Sound data from a microphone

System should capture the voice of the user or an audio playback from another device (e.g. mobile phone) by the help of a microphone connected to the computer. Analog-to-digital conversion sampling rate of this process should be adjustable on a user interface. The user interface must be designed to allow playing the captured audio input on the speaker of the computer.

Sound data from a file

The system should also be able to process .wav or .mp3 sound files. In each case, your code has to retrieve the time-domain signal and the sampling frequency. For the files that contain multi-channel data (such as stereo music) you may choose to work with a single channel.

Data generation

The system must be able to generate the samples of time-domain signals with some simple mathematical expressions. The sampling frequency is always a user-specified parameter. The total length of the generated data is also a user-specified parameter. This can be input in seconds or in number of samples.

Some examples of signal types are given below. The user must be able to observe the time-domain or frequency-domain plots.

Sinusoidal signal

<u>Parameters</u>: Amplitude, frequency and phase.

<u>Signal expression</u>:

$$x(t) = A\cos(2\pi f t + \theta)\,.$$

Windowed sinusoidal

<u>Parameters</u>: Amplitude, frequency and phase. Name of the window function or the samples of it. The starting time and the length.

<u>Signal expression</u>:

$$x(t) = w(t)s(t - t_0)$$

where $w(t) = 0$ for $t \notin [0, \Delta)$ and

$$s(t) = A\cos(2\pi f t + \theta)\,.$$

Note that if *rectangle window* is selected, then the generated signal must be of the form

$$x(t) = \begin{cases} A\cos(2\pi f(t - t_0) + \theta)\,, & t_0 \le t < t_0 + \Delta \\ 0, & \text{otherwise}\,. \end{cases}$$

Rectangle windowed linear chirp

Parameters: Amplitude, initial instantaneous frequency, bandwidth, duration and phase.

Signal expression:

$$x(t) = s(t - t_0)$$

where

$$s(t) = \begin{cases} A\cos\left(2\pi\left(f_0 t + \frac{m}{2\Delta}t^2\right) + \theta\right), & 0 \le t < \Delta\,, \\ 0, & \text{otherwise}\,. \end{cases}$$

Square wave

Parameters: Amplitude, frequency, phase, and duty cycle

Sawtooth wave

Parameters: Amplitude, frequency, phase, and width

Signal involving multiple components

Parameters: Number of components, amplitude for each component, phase for each component, …

Signal expression:

$$x(t) = \sum_{m=0}^{M-1} x_m(t)$$

where each $x_m(t)$ is a sinusoidal signal in the form of $x_m(t) = A_m\cos(2\pi f_m t + \theta_m)$.

Spectrogram

You are going write a MATLAB code to display the spectrogram of a discrete-time signal. A spectrogram is a color or grey scale plot of the magnitude of the short-time Fourier transform (STFT) on the time-frequency plane. One axis of time-frequency plane is time and the other is frequency. The third dimension is for the magnitude of the short-time Fourier transform.

The Short-time Fourier transform (STFT)

The STFT can be defined as follows:

$$X[n,\omega) = \sum_{m=0}^{2M} x[n - M + m]w[m]e^{-j\omega m}$$

Here, $x[n]$ denotes the sequence we would like to analyze for its frequency content around the time instant n and $w[n]$ denotes a window sequence of length $2M + 1$. $w[n]$ (window function, tapering function) has nonzero samples for $n = 0,1,\dots,2M$. For example, if all samples of $w[n]$ are the same then it is called a *rectangular window*[1]. By multiplying $w[n]$ and the signal $x[n+n_0-M]$ you extract a portion ($2M+1$ samples) of $x[n]$ around n_0. Doing so, one focuses on the spectral properties of $x[n]$ over a particular time interval centered at n_0. Hence $X[n_0,\omega)$, is the spectrum of a segment of $x[n]$ around n_0, and ω is the continuous-valued discrete-time frequency variable. To visualize the STFT, compute the discrete Fourier transform[2] (DFT) of the product, i.e.,

$$\hat{X}[n_0,k) \triangleq X[n_0,\omega)\big|_{\omega=k\frac{2\pi}{N}} = \sum_{m=0}^{2M} x[n_0-M+m]w[m]e^{-jk\frac{2\pi}{N}m} \quad , \quad k=0,1,..,N-1$$

where N is at least $2M+1$. Then compute its magnitude, $\left|\hat{X}[n_0,k)\right|$.

Note that the DFT of a signal is obtained by sampling the DTFT of that signal. You may use the "FFT" command of MATLAB for this purpose. The **magnitude** values of this DFT can be coded in color or grey scale and plotted on the time-frequency plane (take horizontal axis as time and vertical axis as frequency). This color plot is called the **spectrogram** of $x[n]$. Commonly, n_0 is allowed to take a sequence of integer values that differ by an integer S called the window shift. The window length, $2M+1$, and the window shift, S, are the fundamental parameters. The window shift should <u>at most</u> be equal to window length (Why?). The type of window function is also important. It has a significant role in some tasks. For more details on STFT, you can read the first 4 sections of Chapter 10, "Fourier Analysis Using Discrete Fourier Transform", from the textbook.

Note that MATLAB has its own "spectrogram" command. <u>You are going to write your own</u>. You may use all other MATLAB commands. You can compare your results to those of MATLAB and other tools you can find from the web.

A signal and its spectrogram are shown in Figure 1 and Figure 2, respectively. Note that the horizontal axis (time) in both figures have the same scale.

[1] There are other well-known window functions like Hamming, Hanning, Tukey, Cosine, Triangular, Gaussian, Blackman, Kaiser…

[2] DFT values are the samples of DTFT.

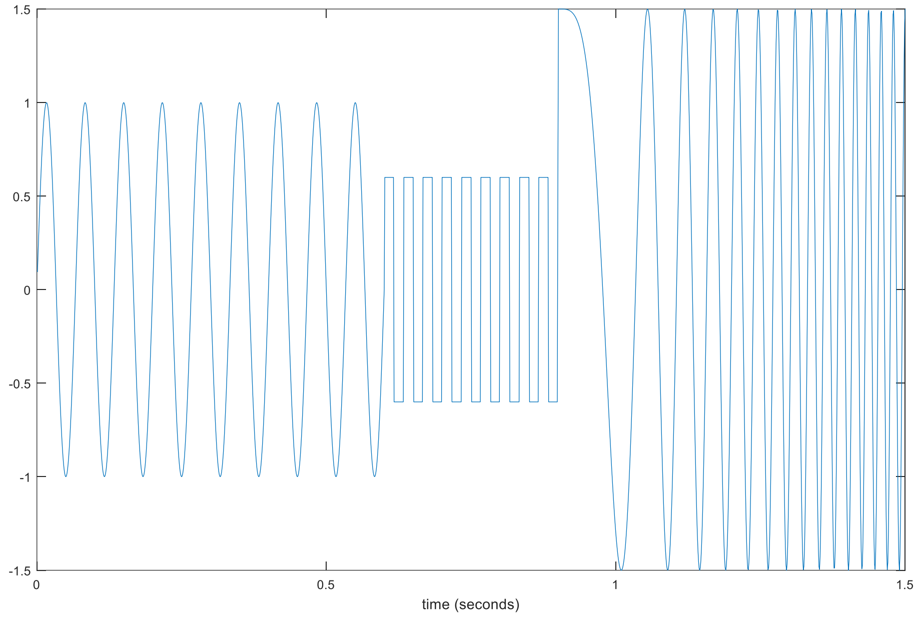

Figure 1: A signal to be analyzed.

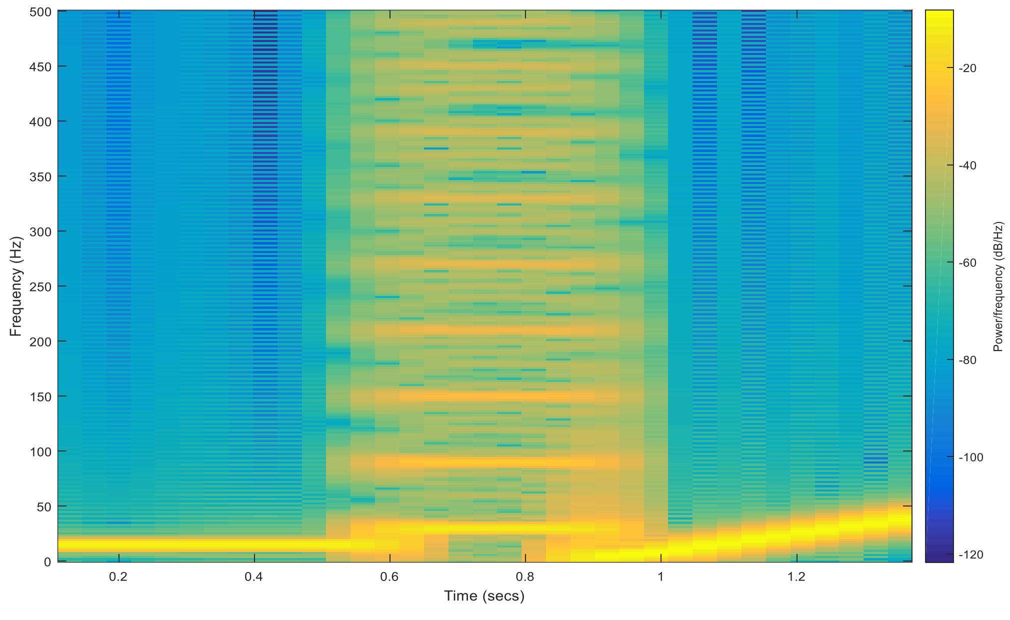

Figure 2: Spectrogram of the signal in Figure 1.

Another signal which is triangular wave followed by periodic sinc function, and its spectrogram are shown in Figure 3 and 4, respectively.

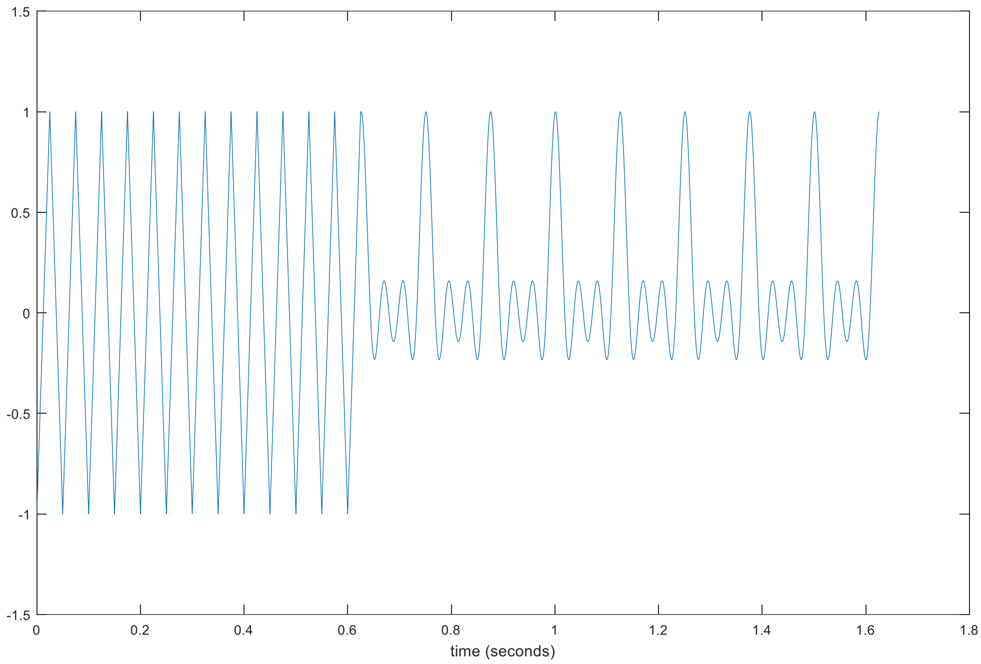

Figure 3: Triangular wave followed by periodic sinc function.

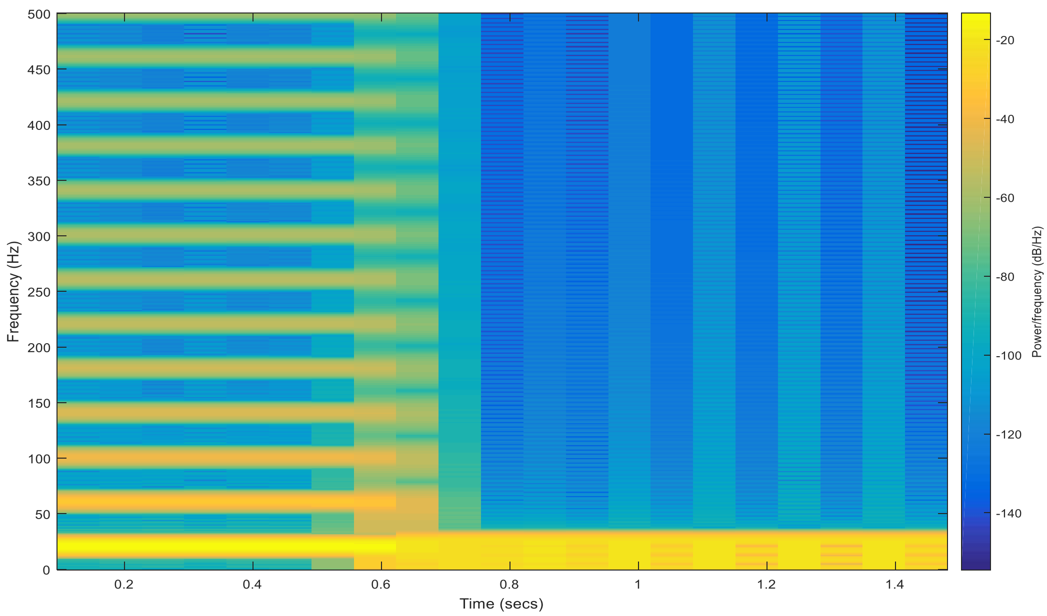

Figure 4: Spectrogram of the signal in Figure 3.

Implementation:

Things to do:

- In your spectrogram code, window length and window type must be variable. For window type, use the window types (Hamming, Hann, Tukey, Cosine, Triangular, Gaussian, Blackman, Kaiser…) provided by MATLAB.
- Try different window lengths and state the differences of the spectrograms obtained by using different window lengths.
- Try different amounts of window overlap and state the differences of the spectrograms obtained by using different amounts of window overlap.
- Try different window types and state the differences of the spectrograms obtained by using different type of windows.

For the above, use all of the following data:

1. Computer generated sinusoidal, linear chirp, square wave, signal involving multiple components. (Note that all the generated signals are rectangular windowed of its infinite versions.)

2. Try signals with more than one sinusoidal components. What happens when the frequencies of the different components are close to each other? What happens when the window lengths change? (Note that the window (nonzero part) of the generated signal and the window of the spectrogram analysis are unrelated in general and they need not be the same.) Play with various data and analysis parameters. Show the results and comment on the issues that attract your attention.