Short-time Fourier Transform (STFT)

**Goal:** To explore the properties of the STFT, including the frequency-time resolution tradeoff, overlap, and extraction of phase information. The Hilbert transform also will be used to estimate instantaneous amplitude and phase of audio signals and some properties of the Wigner distribution will be illustrated.

**Step 1:** We first need some subject audio material. It will be best if we use a recording of a solo instrument and I have placed some examples of solo clarinet music on the website. These are ‘wav’ files, which are easily imported into Matlab as arrays by using the ‘wavread’ command. Pay attention to the sample rate, which is read from the wav file by this command. Import this, or other of your favorite material and familiarize yourself with the ‘wavwrite’, and ‘soundsc’ commands for writing and playing wav files.

**Step 2:** Now that you have imported a bit of audio and are more familiar with its manipulation you will explore the STFT. The basic syntax is the following: `spectrogram(y,WINDOW,NOVERLAP,NFFT,R, 'yaxis')`. Use Matlab Help to understand this in more detail, however y is the input array of sound samples, WINDOW is the FT transform window function, e.g., `hanning(256)`, NOVERLAP is the amount of overlap of the time segments, NFFT is the number of samples for the FFT, R is the sample rate, and by writing ‘yaxis’ at the end of the string the spectrogram is displayed with the frequency along the y axis, the conventional way of displaying a spectrogram. Try various values of NFFT and (with NOVERLAP = 0) and observe what happens to the spectrographic representation in terms of the frequency and time resolution.

**Step 3:** Decimation to improve frequency resolution: Now try sub-sampling the original wav file using the ‘decimate’ command. By reducing the sampling frequency the frequency span of the spectrum is reduced and for a given value of NFFT the frequency resolution improves. However, your time resolution decreases. Try this and find a decimation factor (sample rate reduction) that enables you to clearly identify the fundamental frequencies in your sound files.

**Step 4:** What happens when you employ nonzero values of NOVERLAP? What value might this have for analysis of musical sound, if any?

**Step 5:** Explore the Hilbert transform. Musical sounds, even solo instruments, are much more complex and interesting than simple periodic sine waves or sine waves with overtones. There are rich modulations of the amplitude and the frequency, which give the musical sound its character and emotion. In this part of the exploration you will try the Hilbert transform. The Matlab command for the Hilbert transform the following `xh = hilbert(x)`; where xh is the complex Hilbert transform. To display the instantaneous amplitude and phase use the following `xamp = abs(xh); xpha = unwrap(angle(xh));`
Start by creating synthetic tones and computing the Hilbert transform and plotting it. For example $x = A \cos(2\pi ft + \phi)$ for fixed values of $A$ and $\phi$. Try changing these values to see how it is reflected in the Hilbert transform magnitude and phase. Now let $A \rightarrow A(t)$ and $\phi \rightarrow \phi(t)$ both become simple functions of time, whatever you want them to be and see how this is reflected in the Hilbert transform.

Now apply this to some musical signals. Examples of clarinet, violin and bass are provided on the web-site but find your own samples and explore what you can learn from the Hilbert transform. Provide some examples and commentary.

**Step 6:** Wigner and Wigner-Ville distributions: The Wigner and Wigner-Ville distributions are related to the spectrogram but give better time-frequency resolution at the expense of artifacts and some other non-physical properties. See the following web-site for a brief and useful introduction: [http://case.caltech.edu/tfr/](http://case.caltech.edu/tfr/).

You can write your own version of the Wigner distribution function or find one on Matlab Central. [http://www.mathworks.com/matlabcentral/fileexchange/15637](http://www.mathworks.com/matlabcentral/fileexchange/15637) However you choose to proceed the goal is to obtain the Wigner distribution for a few sample sounds, both synthetic ones with time dependent amplitude and phase and short samples of real musical sounds. Compare these to the STFT spectrographic representations and comment on the time–frequency resolution and any artifacts that are present.

Finally, what is the difference between the Wigner distribution and the Wigner-Ville distribution. Comment on their relative merits and why or when one would be used over the other.