ENGIN 100: Music Signal Processing

LAB #4: Spectrograms and Data Windows

Professor Andrew E. Yagle

Dept. of EECS, The University of Michigan, Ann Arbor, MI 48109-2122

Revised July 2016

I. ABSTRACT

This lab will teach you how to use another tool central to signal processing: time-varying spectral analysis of signals whose spectra vary with time. Music signals, in which different notes have different spectra, are candidates for this analysis. The signal is segmented in time and a discrete Fourier transform of each segment computed. This results in a 3-D picture, called the spectrogram, of a signal spectrum varying in time. The goals of this lab are to: (1) gain the ability to compute and interpret spectrograms of time-varying signals, such as chirps and simple tonal music signals; (2) apply the spectrogram to more complicated synthetic and actual musical signals; and (3) gain the ability to use data windows to facilitate spectrum interpretation.

II. BACKGROUND

This lab is concerned with the *interpretation* of signal spectra. We will trade off some mathematical rigor to obtain more useful signal information. This lab will feature both spectrograms and data windows.

A. Spectrograms

In Lab #3 you learned the basics of discrete-time Fourier analysis. This represents a periodic discrete-time signal as a weighted sum of sinusoids with frequencies that are integer multiples of the reciprocal of the signal's period. The problem with this representation is that many real-world signals are easier to interpret if they are allowed to have a Fourier series expansion whose coefficients vary with time.

It should be emphasized at the start that this is an imperfect representation: While the Fourier series expansion is exact and reversible, representation using sinusoids whose frequencies vary with time is necessarily inexact, since a sinusoid is only a sinusoid if it has the form $A\cos(2\pi ft+\theta)$ for *all* $-\infty < t < \infty$. So the subject of Lab #5 is not as rigorous as the subject of Lab #3.

Nonetheless, time-varying spectral analysis is often useful for *interpretation*. A musical signal, if properly segmented in time, will have a different Fourier series expansion in each segment, if each segment were periodically extended to $-\infty < t < \infty$. This is clearly useful for the musical transcription problem.

To illustrate these ideas, consider the periodic (period=2) signal

$$x(t) = \begin{cases} 3\cos(2\pi 100t) & \text{for } 0 < |t| < 1 \text{ and } 2 < |t| < 3 \text{ and } 4 < |t| < 5 \ldots \\ 3\cos(2\pi 250t) & \text{for } 1 < |t| < 2 \text{ and } 3 < |t| < 4 \text{ and } 5 < |t| < 6 \ldots \end{cases} \tag{1}$$

The Fourier series expansion of this signal has two main components at 100 and 250 Hertz, as you can easily see. However, a more *useful* representation would show that the signal is a 100 Hertz sinusoid that switches its frequency to 250 Hertz. Clearly the way to obtain such a representation is to compute Fourier series expansions of short time segments of the signal. This is the basic idea behind the spectrogram.

In its simplest form, the spectrogram $C(m,n)$ of a periodic signal $x(t)$ (period=T) is defined as follows:

- $x(t)$ is sampled at $F\frac{\text{SAMPLE}}{\text{SECOND}}$ (every $\frac{1}{F}$ seconds), resulting in $x[n] = x(t = n/F)$ for integers n;
- Frequency is also discretized: $f = \frac{m}{T}$ where T is the period of $x(t)$;
- Then the sampled $x[n]$ is periodic with period=N=FT (F is chosen so N is an integer);
- The period=N is segmented in time into L segments of length M each, so N=LM;
- Then we compute the M-point DFT of each of the L segments having length M each:

$$\begin{aligned} A(m,n) &= \sum_{i=nM}^{(n+1)M-1} x[i]\cos(2\pi im/N); \quad m = 0,1\ldots(N-1); \quad n = 0,1\ldots(L-1) && (2) \\ B(m,n) &= \sum_{i=nM}^{(n+1)M-1} x[i]\sin(2\pi im/N); \quad m = 0,1\ldots(N-1); \quad n = 0,1\ldots(L-1) && (3) \\ C(m,n) &= \sqrt{A(m,n)^2 + B(m,n)^2} && (4) \end{aligned}$$

- If $L = 1$ (only one time segment), this is just the discrete Fourier transform from Lab #3;
- If $L > 1$, this is the discrete Fourier transform of $\{x[n], nM \le i \le nM + M - 1\}$ where $0 \le n \le L - 1$;
- $C(m,n)$ can be viewed as the Fourier series coefficient at frequency $f = \frac{m}{T}$ Hertz of $x(t = nM/F)$,
- which is computed using a *window* of width M/F seconds and starting at $t = nM/F$;
- Note $0 \le f < F$ and $0 \le t < T$ as m and n vary from 0 to $N - 1$ since $N = FT$;
- So the spectrogram computes the Fourier series expansion of each time segment of $x[n]$;

This basic spectrogram can easily be computed in Matlab as follows:

- `C=abs(fft(reshape(X,M,L)));` where `X` is the vector of $x[n]$;
- `fft` computes the discrete Fourier transform of length M of each column of the $M \times L$ array `reshape(X,M,L)`;
- The spectrogram can be visualized using either `imagesc(C)` or `waterfall(C)` (try both during lab);
- I have omitted all constant factors throughout, since we are only interested in *relative* values of `C`.

B. Spectra of Sinusoidal Signals

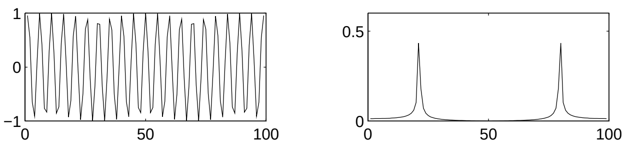
In Lab #3 you saw that the discrete Fourier transform of a pure sinusoid did NOT result in a pure spike in its spectrum at its frequency unless the length of the data was an integer multiple of the period of the sinusoid. In real life this doesn't happen, and the spike in the spectrum has a "base" around it. We would like to remove this base to facilitate interpretation of the spectrum as a single sinusoid with a single peak.

To do this, we mathematically analyze the "base" around the spike in Appendix A (math alert!).

It is important to emphasize that the base *belongs* there–a sinusoid is only a pure sinusoid if it extends over $-\infty < t < \infty$. A truncated sinusoid has a spectrum that includes components near, as well as at, f_o.

As an example, consider one second of a 20.3 Hertz sinusoid sampled at $99\frac{\text{SAMPLE}}{\text{SECOND}}$:

- >> `X=cos(2*pi*20.3*[-49:49]/99);subplot(321),plot(X)` plotted below left;
- >> `FX=abs(fft(X))/99;,subplot(322),plot(FX)` plotted below right:



- Note that the signal does not include a complete period of the sinusoid (although it comes close);
- Note the broad base holding up the spike (remember to delete the second half of the spectrum).

C. Data Windows ("I don't do windows")

What can we do about this, if we are interested in interpreting the truncated sinusoid with a single peak in its spectrum? That is, how can we alter the truncated sinusoid into a finite-duration signal whose spectrum consists of a single peak?

One idea is to gradually *window* the data segment to zero at its two endpoints. Then the effect of not having a complete period of the sinusoid is minimal, since the discontinuity at the two endpoints is reduced to zero since both endpoints are themselves zero. A commonly used (and easy to analyze) data window is the Hanning window, which is

$$w(n) = \begin{cases} \frac{1}{2} + \frac{1}{2}\cos(\frac{2\pi n}{2N+1}) & \text{for } |n| \leq N \\ 0 & \text{otherwise} \end{cases} = \begin{cases} 1 & \text{at n=0} \\ \approx 0 & \text{at } n = \pm N \end{cases} \tag{5}$$

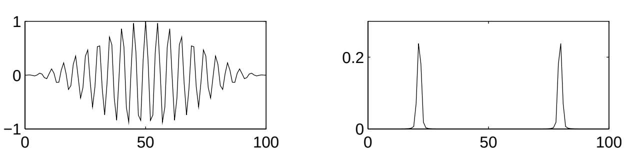
Then the spectrum of $\cos(2\pi f_o n)w(n)$ is (omitting a factor of $\frac{1}{4N+2}$ and the second sinc term)

$$A_k = \text{sinc}[f_o - \frac{k}{2N+1}] + \frac{1}{2}\text{sinc}[f_o - \frac{k+1}{2N+1}] + \frac{1}{2}\text{sinc}[f_o - \frac{k-1}{2N+1}] \tag{6}$$

and $B_k = 0$ again since $w(n) = w(-n)$ (see the argument above that $B_k = 0$).

To see that this helps, redo the previous example with the 20.3 Hertz sinusoid using a Hanning window:

- >> `X=cos(2*pi*20.3*[-49:49]/99);X=X'.*hanning(99);subplot(321),plot(X)`
- >> `FX=abs(fft(X))/99;,subplot(322),plot(FX)` produces the following signal and its spectrum



- Although the peak is broader, its base has disappeared (despite the vertical axis magnification);

- Matlab's `hanning(99)` generates the Hanning window of length 99 as a *column* vector.

  There are many other types of data windows, including the Hamming (sic) window

$$w(n) = 0.54 + 0.46\cos(\frac{2\pi n}{2N+1}), |n| \le N \tag{7}$$

That slight change from (0.50,0.50) to (0.54,0.46) improves performance.

And yes, there were two different people named Hanning and Hamming!

D. Chirps

An important signal with time-varying content is the *chirp* $x(t) = \cos(2\pi f_o t^2)$. At first glance, $x(t)$ seems to have a frequency $f_o t$ Hertz that increases linearly with time. However, the instantaneous frequency is in fact $2f_o t$ Hertz. To see why, consider $x(t)$ over the small interval $t_o < t < t_o + \Delta$. Then we have

$$x(t) = \cos(2\pi f_o t^2) = \cos(2\pi f_o [t_o + (t-t_o)]^2) = \cos(2\pi f_o [t_o^2 + 2t_o(t-t_o) + (t-t_o)^2]) \approx \cos(2\pi[(2f_o t_o)t - f_o t_o^2]) \tag{8}$$

since $(t-t_o)^2 < \Delta^2$ is negligible. This shows $x(t)$ can be interpreted as a sinusoid with instantaneous frequency $2f_o t_o$ and time delay $f_o t_o^2$. You will see this for yourself when you do Lab #4 below.

III. LAB #4: WHAT YOU HAVE TO DO

Use `subplot` to put the 3 plots from (A) and (B) on one page, and the 2 plots from (C) on another page.

- A. Chirp signal

  – Type >> `X=cos([1:1000].^2/1000);plot(X)` Describe this signal.

  – Type >> `imagesc(abs(fft(reshape(X,40,25)))),colormap(gray)`.

  – Determine an expression for the instantaneous frequency of `X` from the plot.

- B. Tonal music

  – Type >> `[Y,FS]=audioread('victorstone.wav');` after downloading *victorstone.wav*.

  – Type >> `imagesc(abs(fft(reshape(Y,300,260)))),colormap(gray)`

  – Can you read off the relative pitches and durations of the tones from the spectrogram?

- C. Removing interference from a signal

  – A Michigan State fan broke into the Engin 100 Canvas web site and corrupted "The Victors" by adding to it the MSU fight song! Can you undo this heresy?

  – Type >> `[Z,FS]=audioread('victorsmsu.wav');` after downloading *victorsmsu.wav*.

  – Type >> `F=fft(Z);plot(abs(F))` Can you distinguish the UM and MSU fight song spectra?

  – Type >> `imagesc(abs(fft(reshape(Z,300,260)))),colormap(gray)`. How about now?

  – Set some values of `F` to zero, then `sound(real(ifft(F)))`. Did you eliminate MSU?

IV. APPENDIX A: ANALYSIS OF SPECTRA OF SINUSOIDAL SIGNALS

Consider a sinusoid of frequency f Hertz sampled at $F\frac{\text{SAMPLE}}{\text{SECOND}}$ over the interval $|t| \leq \frac{N}{F}$

$$x(t) = \cos(2\pi f t), |t| \leq (N/F) \rightarrow x[n] = x(t = n/F) = \cos(2\pi(f/F)n) = \cos(2\pi f_0 n), -N \leq n \leq N$$

where $f_o = \frac{f}{F}$. The discrete Fourier transform of this segment of data having length $2N+1$ (not $2N$) is

$$B_k = 2\sum_{n=-N}^{n=N} \cos(2\pi f_o n)\sin(\frac{2\pi nk}{2N+1}) = 2\sum_{n=1}^{n=N} \cos(2\pi f_o n)\sin(\frac{2\pi nk}{2N+1}) + 2\sum_{n=-1}^{n=-N} \cos(2\pi f_o n)\sin(\frac{2\pi nk}{2N+1}) = 0$$

since $\cos(-x)$=$\cos(x)$ and $\sin(-x)$=$-\sin(x)$. Then $2\cos(x)\cos(y)$=$\cos(x-y)+\cos(x+y)$ gives

$$A_k = 2\sum_{n=-N}^{n=N} \cos(2\pi f_o n)\cos(\frac{2\pi nk}{2N+1}) = \sum_{n=-N}^{n=N}\left[\cos(2\pi[f_o - \frac{k}{2N+1}]n) + \cos(2\pi[f_o + \frac{k}{2N+1}]n)\right]$$

To evaluate these two sums, multiply the first sum by $2\sin(\pi[f_o - \frac{k}{2N+1}])$ (note the missing factor of $2n$)

$$\begin{aligned} 2\sin(\pi[f_o - \frac{k}{2N+1}])\sum_{n=-N}^{n=N}\cos(2\pi[f_o - \frac{k}{2N+1}]n) &= 2\sum_{n=-N}^{n=N}\sin(\pi[f_o - \frac{k}{2N+1}])\cos(2\pi[f_o + \frac{k}{2N+1}]n) \\ &= \sum_{n=-N}^{n=N}\left[\sin(2\pi[f_o - \frac{k}{2N+1}](n+1/2)) - \sin(2\pi[f_o - \frac{k}{2N+1}](n-1/2))\right] \end{aligned}$$

$$= \sin(2\pi[f_o - \frac{k}{2N+1}](N+1/2)) - \sin(2\pi[f_o - \frac{k}{2N+1}](-N-1/2)) = 2\sin(2\pi[f_o - \frac{k}{2N+1}](N+1/2)) \quad (9)$$

since the sum "telescopes" (all of the terms cancel except the first and last) and $\sin(-x) = -\sin(x)$. Now dividing by $2\sin(\pi[f_o - \frac{k}{2N+1}])$ shows that the first sum in the expression for A_k is

$$\frac{\sin(2\pi[f_o - \frac{k}{2N+1}](N+1/2))}{\sin(\pi[f_o - \frac{k}{2N+1}])} = \text{sinc}[f_o - \frac{k}{2N+1}]$$

The second sum in the expression for A_k gives the same thing, except $[f_o - \frac{k}{2N+1}]$ is replaced with $[f_o + \frac{k}{2N+1}]$:

$$A_k = \frac{\sin(2\pi[f_o - \frac{k}{2N+1}](N+1/2))}{\sin(\pi[f_o - \frac{k}{2N+1}])} + \frac{\sin(2\pi[f_o + \frac{k}{2N+1}](N+1/2))}{\sin(\pi[f_o + \frac{k}{2N+1}])} = \text{sinc}[f_o - \frac{k}{2N+1}] + \text{sinc}[f_o + \frac{k}{2N+1}]$$

- The first term in A_k has its peak at the value of $k/(2N+1)$ closest to f_o. But it has a broad base;
- The second term is negligible, since it peaks in the 2^{nd} (discarded) half of the discrete Fourier transform;
- If $f_o = k_o/(2N+1)$ for some integer k_o, $A_k = 0$ *exactly* except at $k = k_o$ and $k = 2N+1-k_o$, since $\sin(2\pi[\frac{k_o}{2N+1} - \frac{k}{2N+1}](N+1/2)) = \sin(\pi[k_o - k]) = 0$. At $k = k_o$ or $k = 2N+1-k_o$ this is $2N+1$;
- $C_k = |A_k| = \frac{\sin(2\pi f_o(N+1/2))}{\sin(\pi[f_o - \frac{k}{2N+1}])}$ since $\sin(2\pi[\frac{k}{2N+1}](N+1/2)) = \sin(\pi k)$ which only changes the sign of A_k;
- We have omitted a factor of $1/(2N+1)$ throughout for readability.