EE123 Digital Signal Processing

Lecture 8

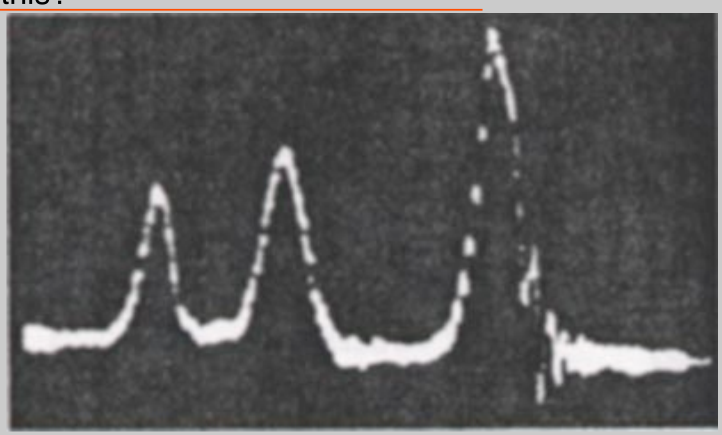
based on slides by J.M. Kahn

Announcements

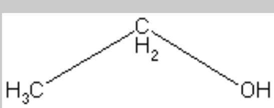
- Last time:
  - FFT
- Today Frequency Analysis with DFT
- Read Ch. 10.1-10.2

- Who started playing with the SDR?

What is this?



The first NMR spectrum of ethanol 1951.



Spectral Analysis with the DFT

The DFT can be used to analyze the spectrum of a signal.

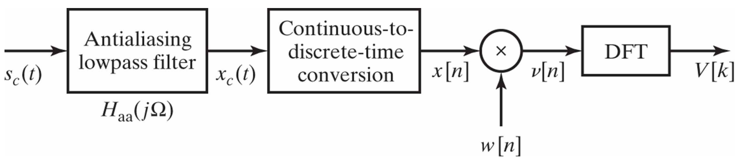
It would seem that this should be simple, take a block of the signal and compute the spectrum with the DFT.

However, there are many important issues and tradeoffs:

- Signal duration vs spectral resolution
- Signal sampling rate vs spectral range
- Spectral sampling rate
- Spectral artifacts

Spectral Analysis with the DFT

Consider these steps of processing continuous-time signals:



Spectral Analysis with the DFT

Two important tools:

- Applying a window to the input signal – reduces spectral artifacts
- Padding input signal with zeros – increases the spectral sampling

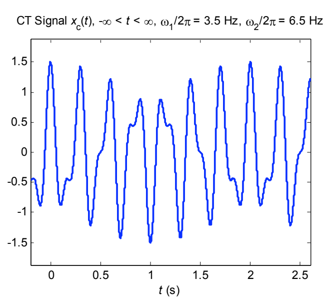
Key Parameters:

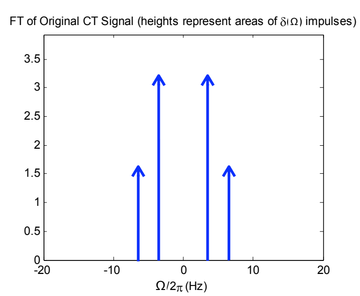
Parameter	**Symbol**	**Units**
Sampling interval	T	s
Sampling frequency	$\Omega_s = \frac{2\pi}{T}$	rad/s
Window length	L	unitless
Window duration	$L \cdot T$	s
DFT length	$N \geq L$	unitless
DFT duration	$N \cdot T$	s
Spectral resolution	$\frac{\Omega_s}{L} = \frac{2\pi}{L \cdot T}$	rad/s
Spectral sampling interval	$\frac{\Omega_s}{N} = \frac{2\pi}{N \cdot T}$	rad/s

Filtered Continuous-Time Signal

We consider an example:

$$x_c(t) = A_1 \cos \omega_1 t + A_2 \cos \omega_2 t$$
$$X_c(j\Omega) = A_1 \pi[\delta(\Omega - \omega_1) + \delta(\Omega + \omega_1)] + A_2 \pi[\delta(\Omega - \omega_2) + \delta(\Omega + \omega_2)]$$





Sampled Filtered Continuous-Time Signal

Sampled Signal

If we sampled the signal over an infinite time duration, we would have:

$$x[n] = x_c(t)|_{t=nT}, \quad -\infty < n < \infty$$

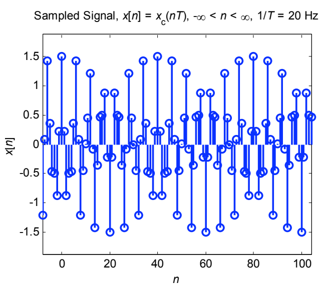
described by the discrete-time Fourier transform:

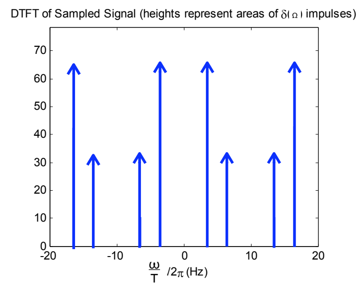
$$X(e^{j\Omega T}) = \frac{1}{T} \sum_{r=-\infty}^{\infty} X_c\left(j\left(\Omega - r\frac{2\pi}{T}\right)\right), \quad -\infty < \Omega < \infty$$

Recall $X(e^{j\omega}) = X(e^{j\Omega T})$, where $\omega = \Omega T$ ... more in ch 4.

Sampled Filtered Continuous-Time Signal

In the examples shown here, the sampling rate is $\Omega_s/2\pi = 1/T = 20$ Hz, sufficiently high that aliasing does not occur.





Windowed Sampled Signal

Block of L Signal Samples

In any real system, we sample only over a finite block of L samples:

$$x[n] = x_c(t)|_{t=nT}, \quad 0 \le n \le L-1$$

This simply corresponds to a rectangular window of duration L.

Recall: in Homework 1 we explored the effect of rectangular and triangular windowing

Windowed Sampled Signal

Windowed Block of L Signal Samples

We take the block of signal samples and multiply by a window of duration L, obtaining:

$$v[n] = x[n] \cdot w[n], \quad 0 \le n \le L-1$$

Suppose the window $w[n]$ has DTFT $W(e^{j\omega})$.

Then the windowed block of signal samples has a DTFT given by the periodic convolution between $X(e^{j\omega})$ and $W(e^{j\omega})$:

$$V(e^{j\omega}) = \frac{1}{2\pi}\int_{-\pi}^{\pi} X(e^{j\theta})W(e^{j(\omega-\theta)})d\theta$$

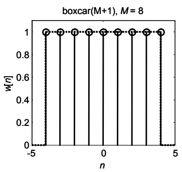
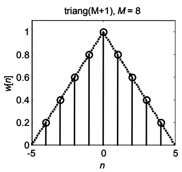
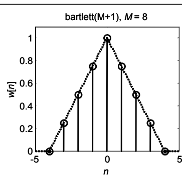
Windowed Sampled Signal

Convolution with $W(e^{j\omega})$ has two effects in the spectrum:

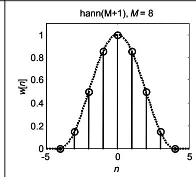
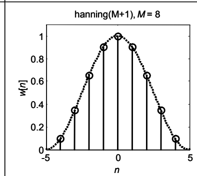
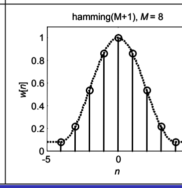
1. It limits the spectral resolution. – Main lobes of the DTFT of the window
2. The window can produce *spectral leakage*. – Side lobes of the DTFT of the window

* These two are always a tradeoff - time-frequency uncertainty principle

Windows (as defined in MATLAB)

Name(s)	Definition	MATLAB Command	Graph ($M = 8$)
Rectangular Boxcar Fourier	$w[n] = \begin{cases} 1 & \lvert n \rvert \le M/2 \\ 0 & \lvert n \rvert > M/2 \end{cases}$	`boxcar(M+1)`	
Triangular	$w[n] = \begin{cases} 1 - \frac{\lvert n \rvert}{M/2+1} & \lvert n \rvert \le M/2 \\ 0 & \lvert n \rvert > M/2 \end{cases}$	`triang(M+1)`	
Bartlett	$w[n] = \begin{cases} 1 - \frac{\lvert n \rvert}{M/2} & \lvert n \rvert \le M/2 \\ 0 & \lvert n \rvert > M/2 \end{cases}$	`bartlett(M+1)`	

Windows (as defined in MATLAB)

Name(s)	Definition	MATLAB Command	Graph ($M = 8$)
Hann	$w[n] = \begin{cases} \frac{1}{2}\left[1 + \cos\left(\frac{\pi n}{M/2}\right)\right] & \lvert n \rvert \le M/2 \\ 0 & \lvert n \rvert > M/2 \end{cases}$	`hann(M+1)`	
Hanning	$w[n] = \begin{cases} \frac{1}{2}\left[1 + \cos\left(\frac{\pi n}{M/2+1}\right)\right] & \lvert n \rvert \le M/2 \\ 0 & \lvert n \rvert > M/2 \end{cases}$	`hanning(M+1)`	
Hamming	$w[n] = \begin{cases} 0.54 + 0.46\cos\left(\frac{\pi n}{M/2}\right) & \lvert n \rvert \le M/2 \\ 0 & \lvert n \rvert > M/2 \end{cases}$	`hamming(M+1)`	

Windows

- All of the window functions $w[n]$ are real and even.
- All of the discrete-time Fourier transforms

$$W(e^{j\omega}) = \sum_{n=-\frac{M}{2}}^{\frac{M}{2}} w[n]e^{-jn\omega}$$

are real, even, and periodic in ω with period 2π.

- In the following plots, we have normalized the windows to unit d.c. gain:

$$W(e^{j0}) = \sum_{n=-\frac{M}{2}}^{\frac{M}{2}} w[n] = 1$$

This makes it easier to compare windows.

Window Example