

ESE 531: Digital Signal Processing

Lec 23: April 17, 2018
Spectral Analysis



Lecture Outline

- ❑ Frequency analysis with DFT
- ❑ Windowing
- ❑ Effect of zero-padding
- ❑ Time-dependent Fourier transform
 - Aka short-time Fourier transform

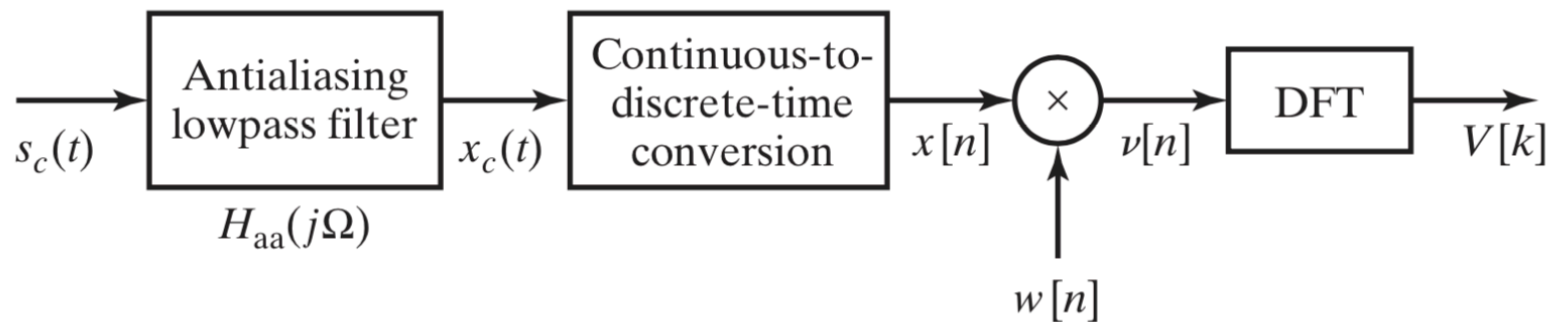


Spectral Analysis Using the DFT

- ❑ DFT is a tool for spectrum analysis
- ❑ Should be simple:
 - Take a block, compute spectrum with DFT
- ❑ But, there are issues and tradeoffs:
 - Signal duration vs spectral resolution
 - Sampling rate vs spectral range
 - Spectral sampling rate
 - Spectral artifacts

Spectral Analysis Using the DFT

- Steps for processing continuous time (CT) signals



Spectral Analysis Using the DFT

- ❑ Two important tools:
 - Applying a window → reduced artifacts
 - Zero-padding → increases spectral sampling

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



CT Signal Example

$$x_c(t) = A_1 \cos(\omega_1 t) + A_2 \cos(\omega_2 t)$$

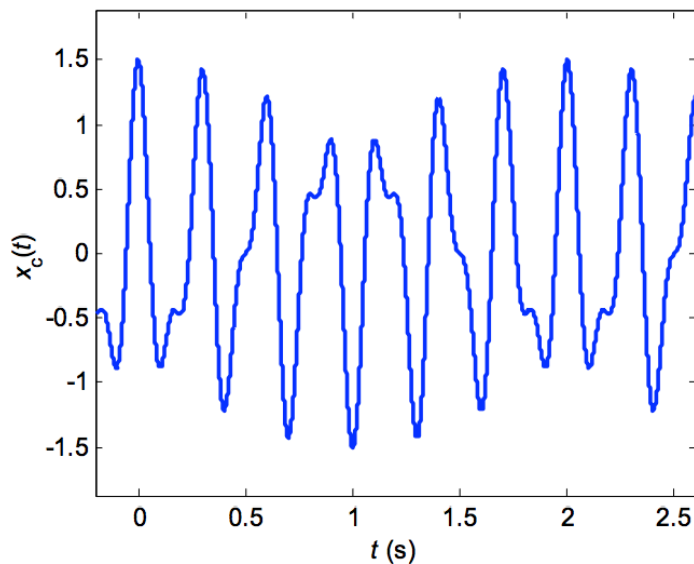
$$X_c(j\Omega) = A_1 \pi [\delta(\Omega - \omega_1) + \delta(\Omega + \omega_1)] + A_2 [\delta(\Omega - \omega_2) + \delta(\Omega + \omega_2)]$$

CT Signal Example

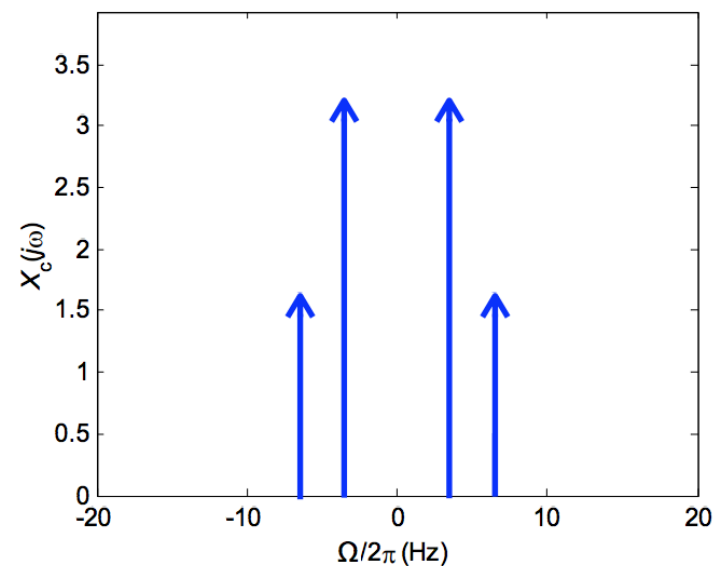
$$x_c(t) = A_1 \cos(\omega_1 t) + A_2 \cos(\omega_2 t)$$

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

CT Signal $x_c(t)$, $-\infty < t < \infty$, $\omega_1/2\pi = 3.5$ Hz, $\omega_2/2\pi = 6.5$ Hz



FT of Original CT Signal (heights represent areas of $\delta(\Omega)$ impulses)





Sampled CT Signal Example

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

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

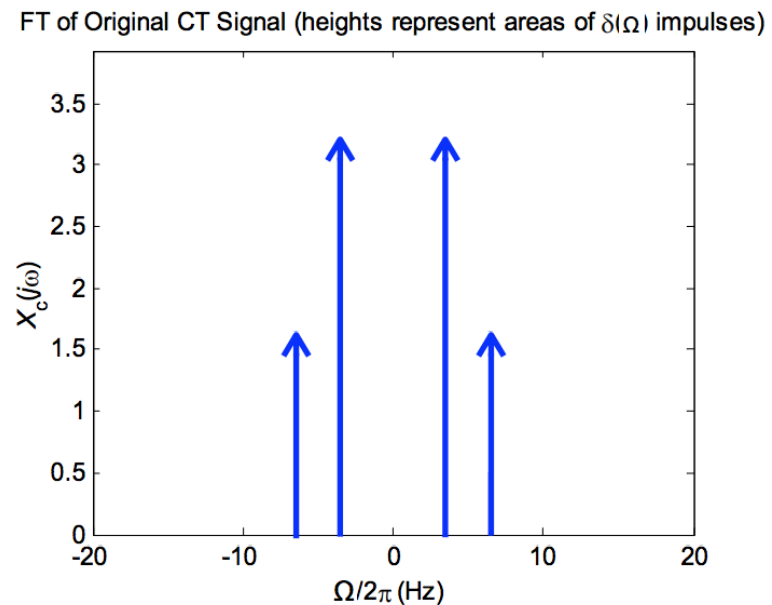
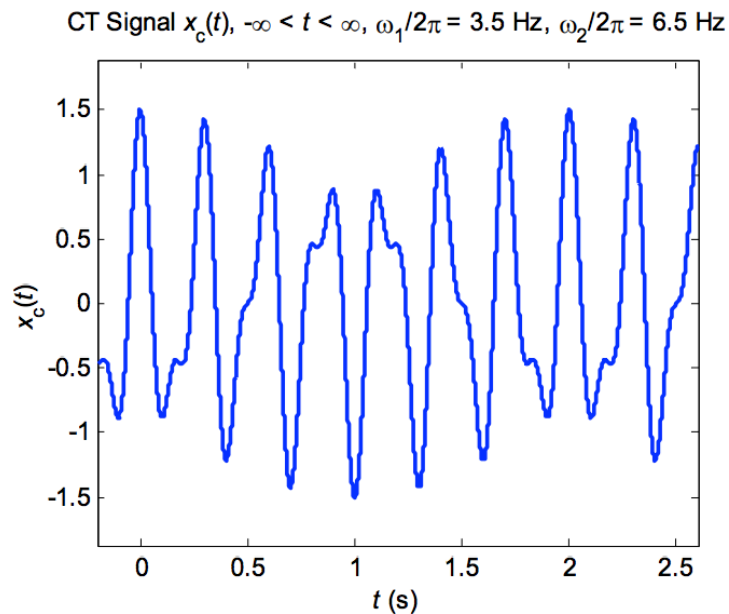
- With the discrete time Fourier transform (DTFT):

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

CT Signal Example

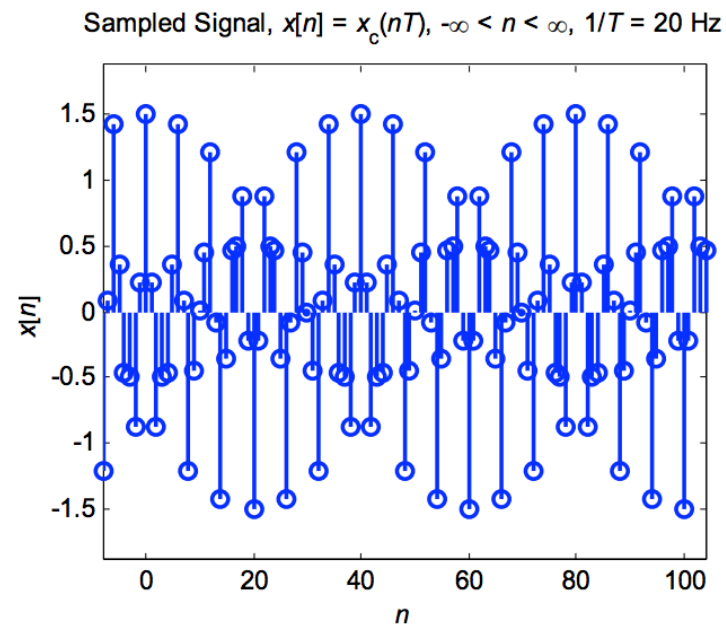
$$x_c(t) = A_1 \cos(\omega_1 t) + A_2 \cos(\omega_2 t)$$

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

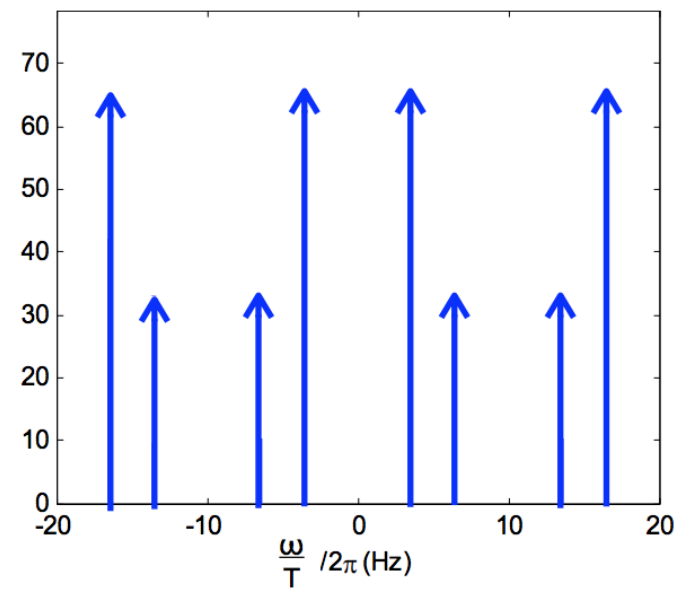


Sampled CT Signal Example

- Sampling with $\Omega_s / 2\pi = 1/T = 20\text{Hz}$



DTFT of Sampled Signal (heights represent areas of $\delta(\Omega)$ impulses)





Windowed Sampled CT Signal

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

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

- ❑ This simply corresponds to a rectangular window of duration L
- ❑ Recall there are many other window types
 - Hann, Hamming, Blackman, Kaiser, etc.



Windowed Sampled CT Signal

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

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

- If 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 $W(e^{j\omega})$:

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

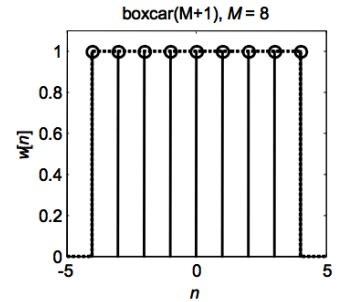
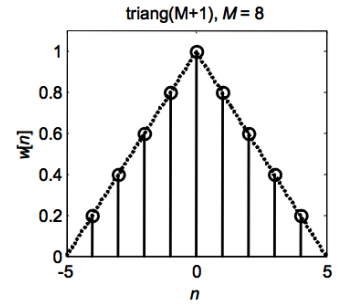
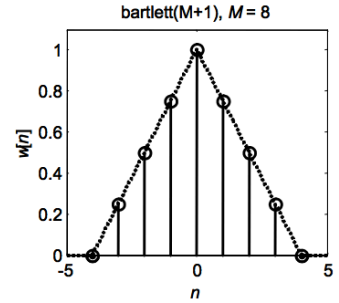


Windowed Sampled CT Signal

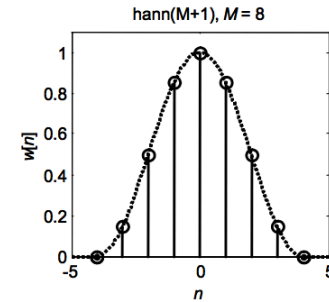
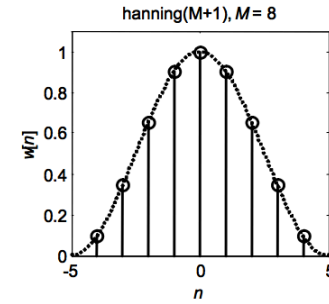
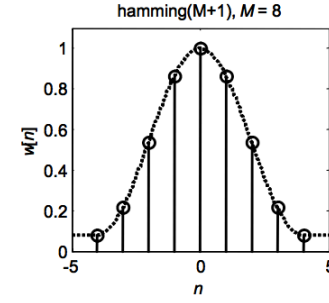
- ❑ Convolution with $W(e^{j\omega})$ has two effects in the spectrum:
 - It limits the spectral resolution
 - Main lobes of the DTFT of the window
 - The window can produce spectral leakage
 - Side lobes of the DTFT of the window

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

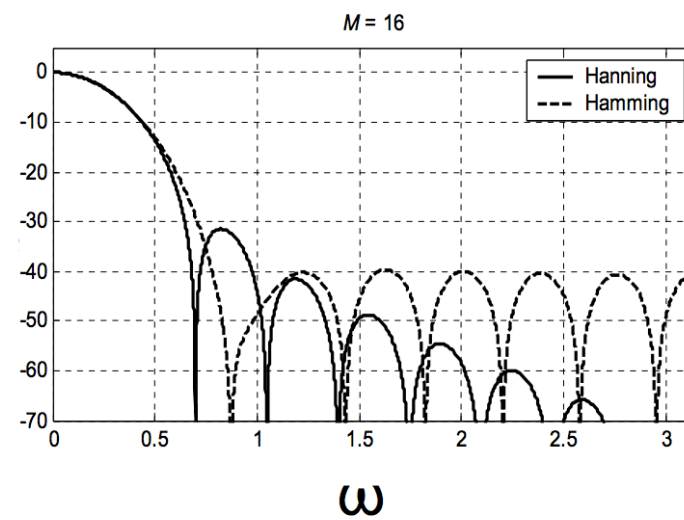
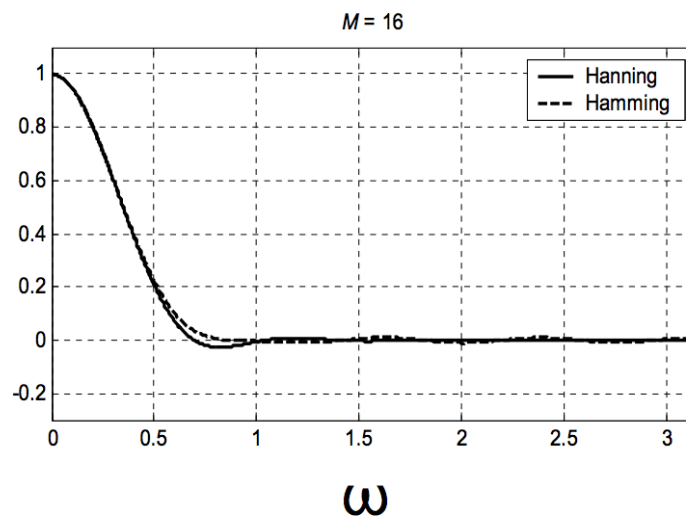
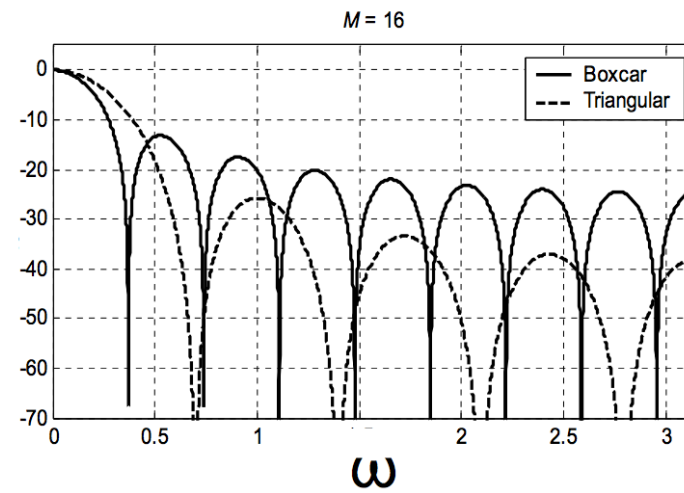
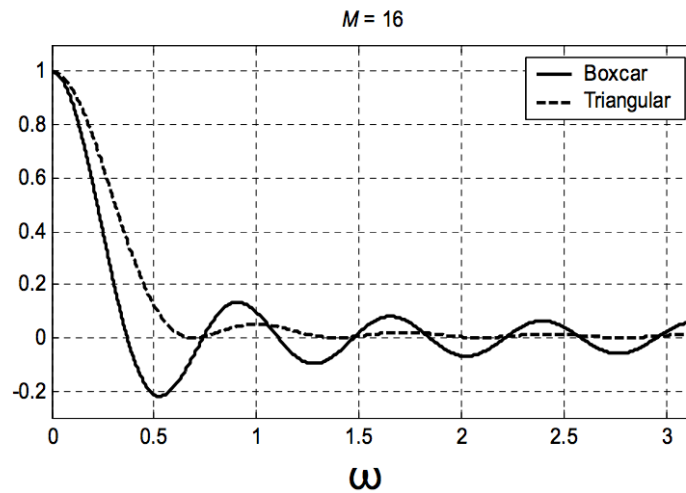
Windows

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

Windows

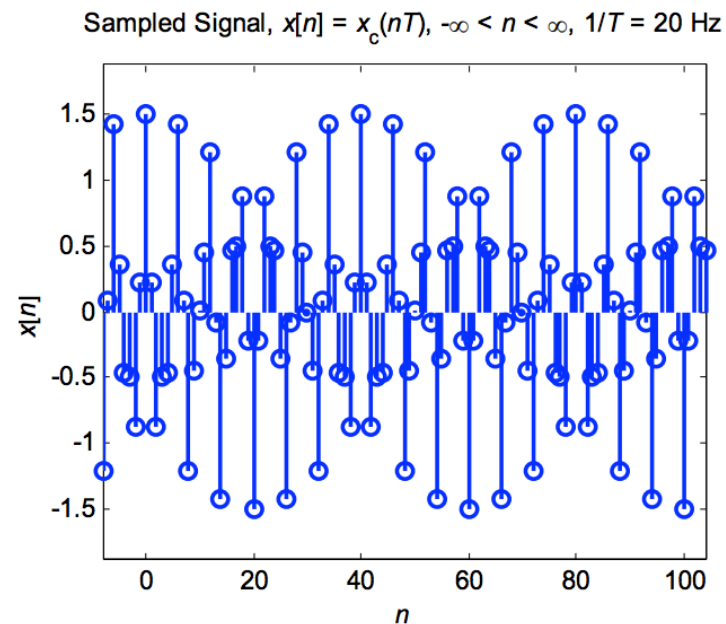
Name(s)	Definition	MATLAB Command	Graph ($M = 8$)
Hann	$w[n] = \begin{cases} \frac{1}{2} \left[1 + \cos\left(\frac{\pi n}{M/2}\right) \right] & n \leq M/2 \\ 0 & n > M/2 \end{cases}$	hann (M+1)	
Hanning	$w[n] = \begin{cases} \frac{1}{2} \left[1 + \cos\left(\frac{\pi n}{M/2 + 1}\right) \right] & n \leq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

Windows

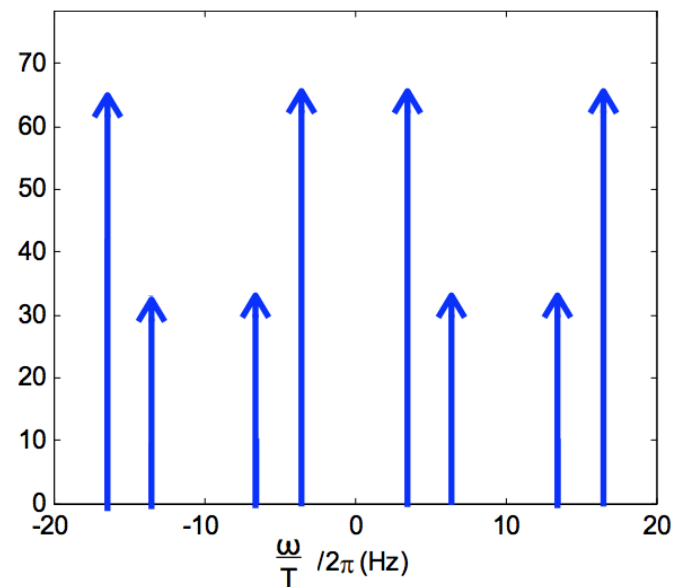


Sampled CT Signal Example

- Sampling with $\Omega_s/2\pi = 1/T = 20\text{Hz}$

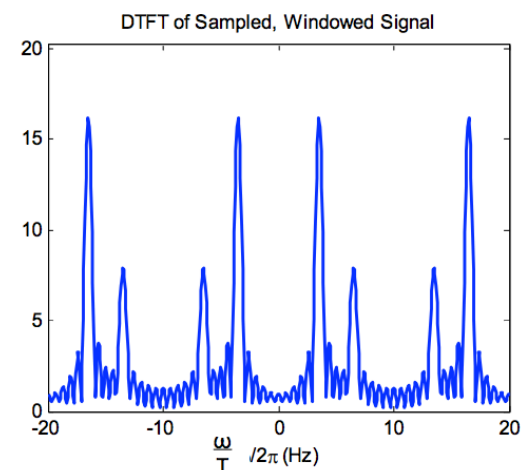
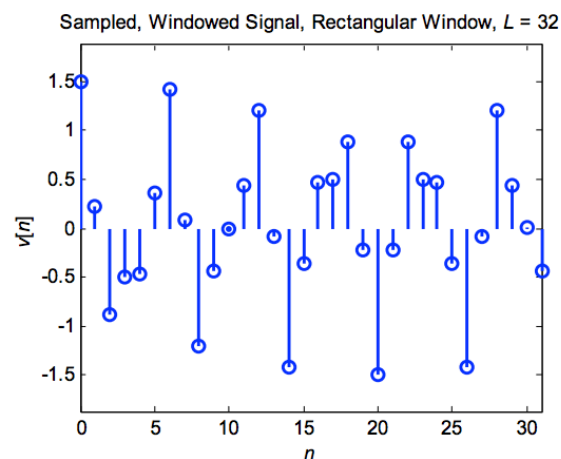
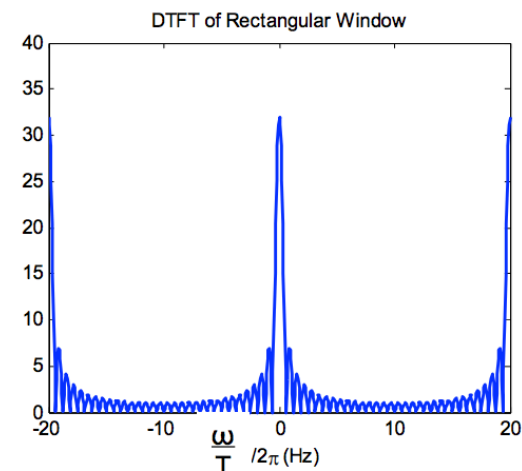
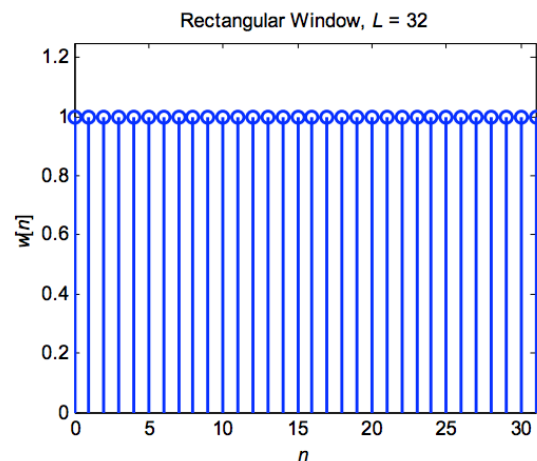


DTFT of Sampled Signal (heights represent areas of $\delta(\omega)$ impulses)



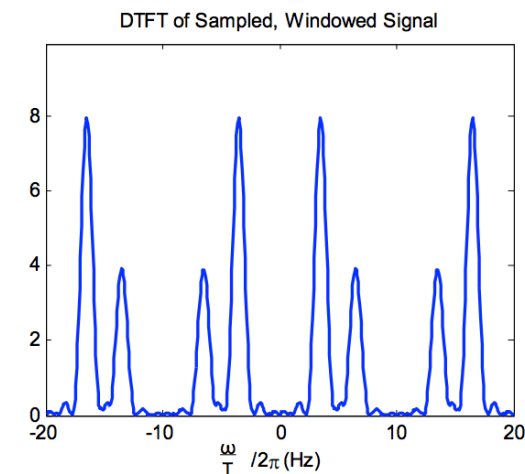
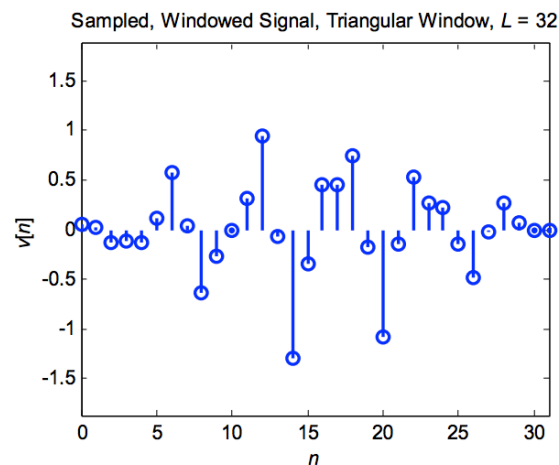
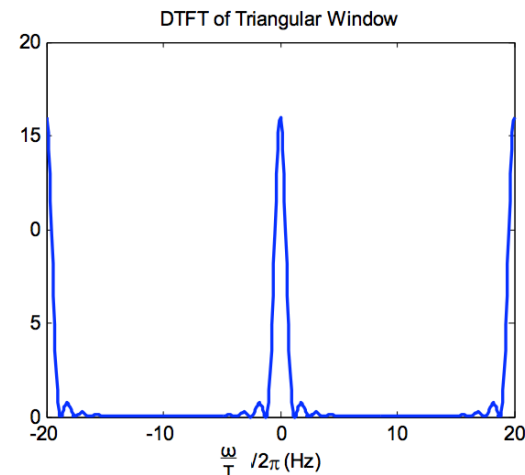
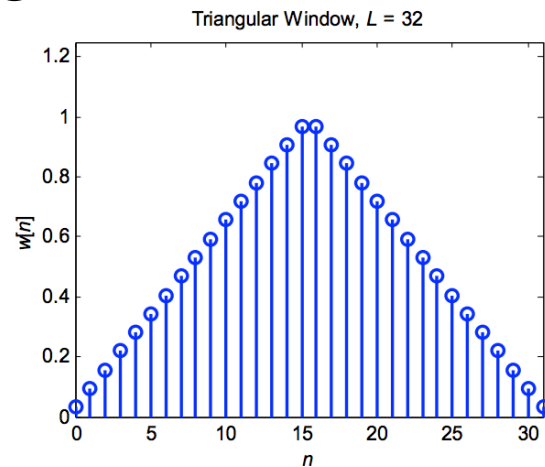
Windowed Sampled CT Signal Example

- ❑ As before, the sampling rate is $\Omega_s/2\pi = 1/T = 20\text{Hz}$
- ❑ Rectangular Window, $L = 32$



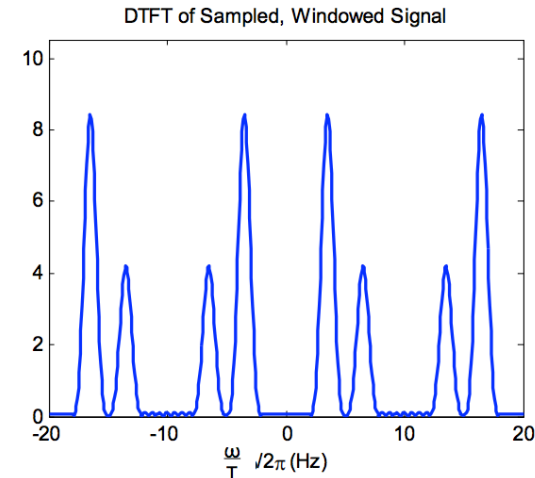
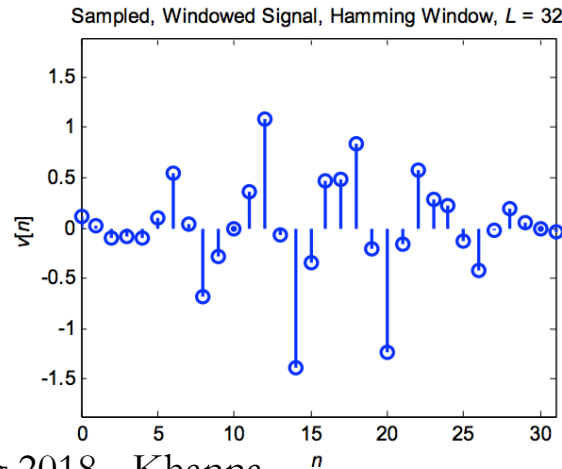
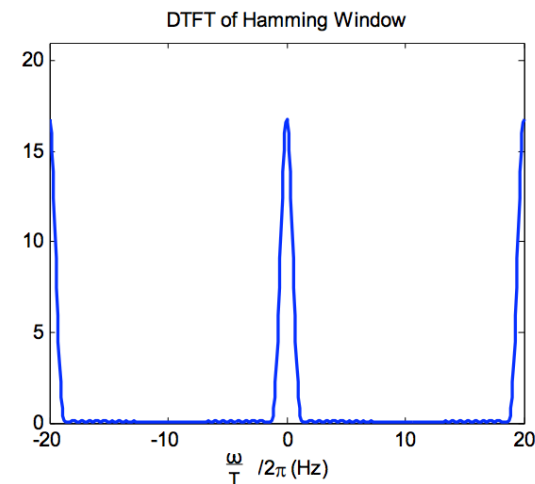
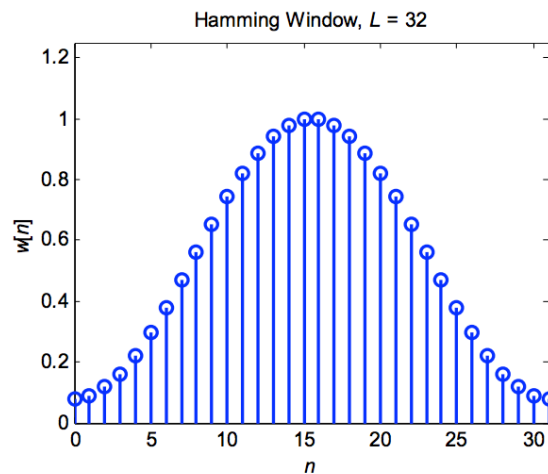
Windowed Sampled CT Signal Example

- As before, the sampling rate is $\Omega_s/2\pi = 1/T = 20\text{Hz}$
- Triangular Window, $L = 32$



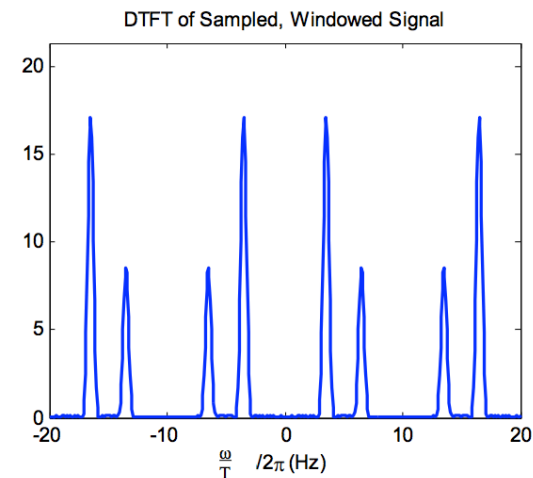
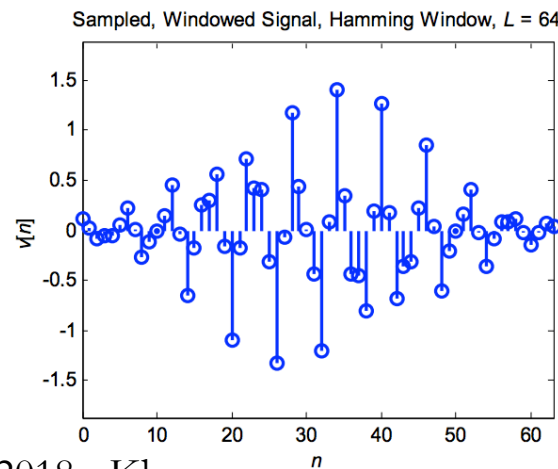
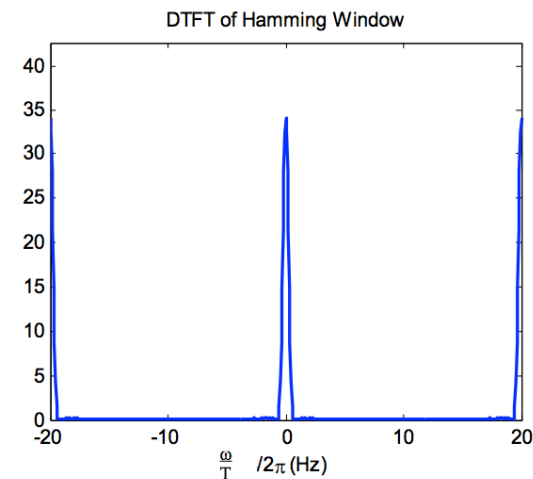
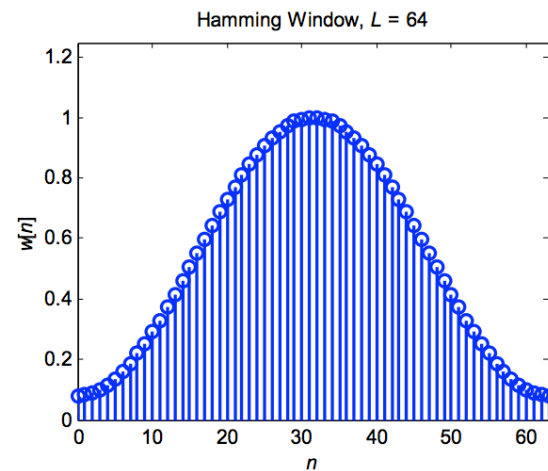
Windowed Sampled CT Signal Example

- ❑ As before, the sampling rate is $\Omega_s/2\pi = 1/T = 20\text{Hz}$
- ❑ Hamming Window, $L = 32$



Windowed Sampled CT Signal Example

- ❑ As before, the sampling rate is $\Omega_s/2\pi = 1/T = 20\text{Hz}$
- ❑ Hamming Window, $L = 64$





Optimal Window: Kaiser

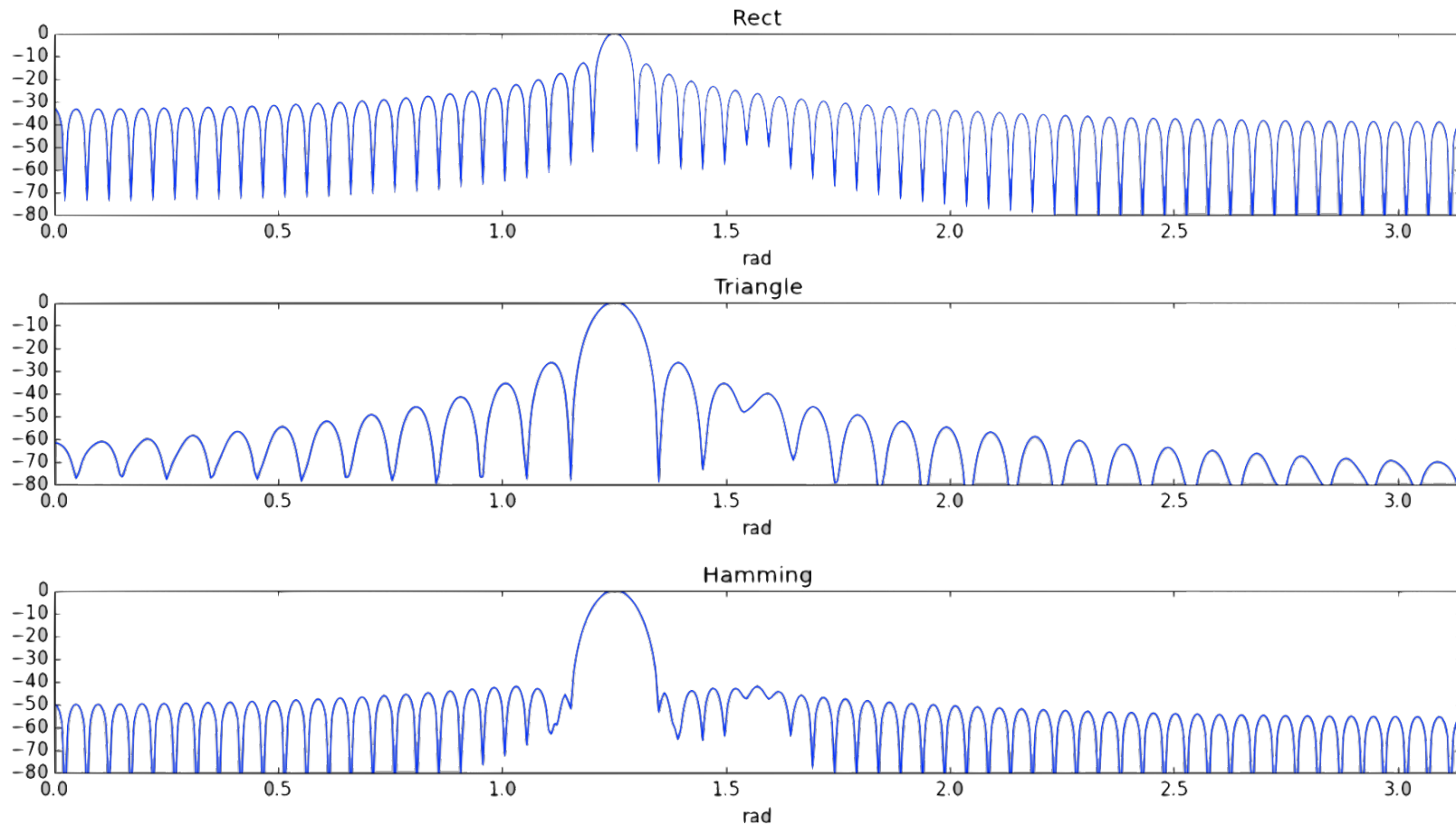
- ❑ Minimum main-lobe width for a given sidelobe energy percentage

$$\frac{\int_{\text{sidelobes}} |H(e^{j\omega})|^2 d\omega}{\int_{-\pi}^{\pi} |H(e^{j\omega})|^2 d\omega}$$

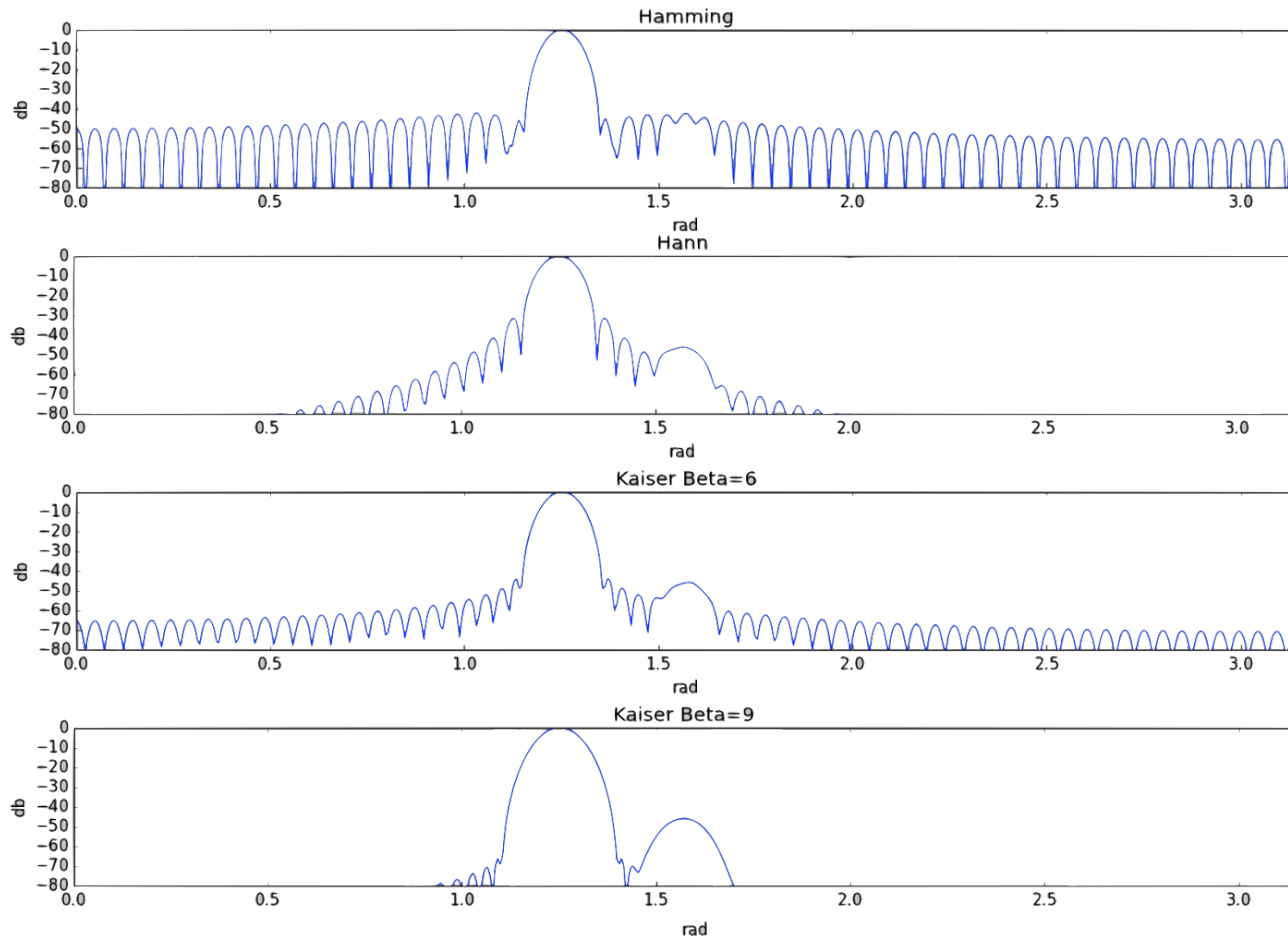
- ❑ Window is parameterized with L and β
 - β determines side-lobe level
 - L determines main-lobe width

Window Comparison Example

$$y[n] = \sin(2\pi 0.1992n) + 0.005\sin(2\pi 0.25n) \mid 0 \leq n \leq 128$$



Window Comparison Example





Zero-Padding

- ❑ In preparation for taking an N-point DFT, we may zero-pad the windowed block of signal samples

$$\begin{cases} v[n] & 0 \leq n \leq L-1 \\ 0 & L \leq n \leq N-1 \end{cases}$$

- ❑ This zero-padding has no effect on the DTFT of $v[n]$, since the DTFT is computed by summing over infinity
- ❑ Effect of Zero Padding
 - We take the N-point DFT of the zero-padded $v[n]$, to obtain the block of N spectral samples:



Zero-Padding

- Consider the DTFT of the zero-padded $v[n]$. Since the zero-padded $v[n]$ is of length N , its DTFT can be written:

$$V(e^{j\omega}) = \sum_{n=0}^{N-1} v[n]e^{-nj\omega}, \quad -\infty < \omega < \infty$$

- The N -point DFT of $v[n]$ is given by:

$$V[k] = \sum_{n=0}^{N-1} v[n]W_N^{kn} = \sum_{n=0}^{N-1} v[n]e^{-j(2\pi/N)nk}, \quad 0 \leq k \leq N-1$$

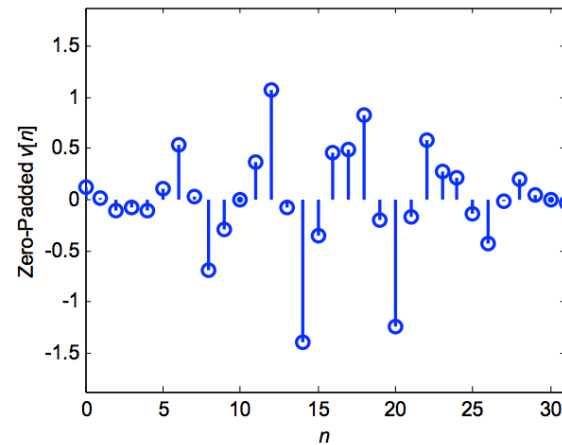
- We know that the DFT is a sample $V(e^{j\omega})$:

$$V[k] = V(e^{j\omega}) \Big|_{\omega=k\frac{2\pi}{N}}, \quad 0 \leq k \leq N-1$$

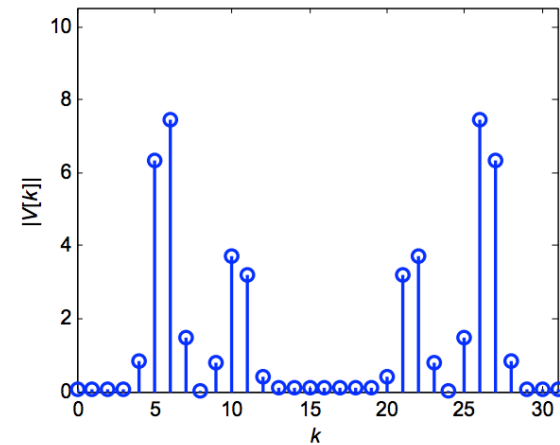
Frequency Analysis with DFT

□ Hamming window, $L = N = 32$

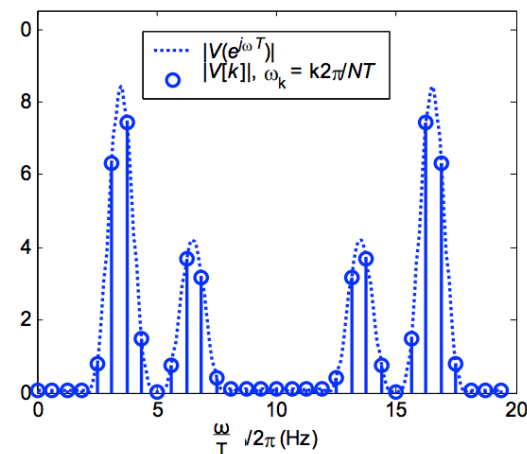
Sampled, Windowed Signal, Hamming Window, $L = 32$, Zero-Padded to $N = 32$



N -Point DFT of Sampled, Windowed, Zero-Padded Signal



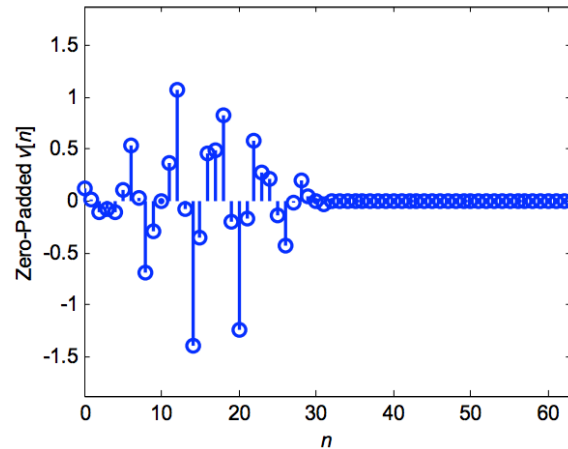
Spectrum of Sampled, Windowed, Zero-Padded Signal



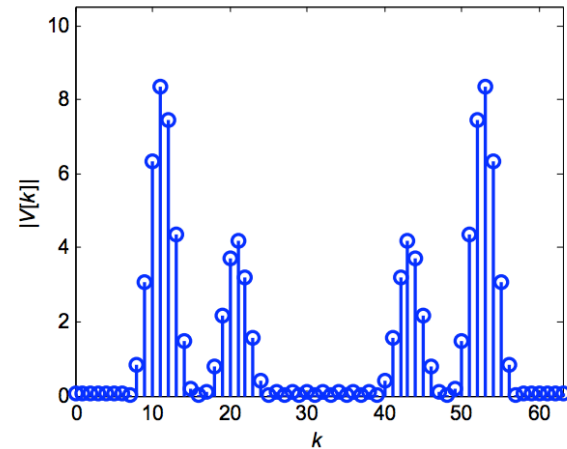
Frequency Analysis with DFT

- ❑ Hamming window, $L = 32$, Zero-padded to $N = 64$

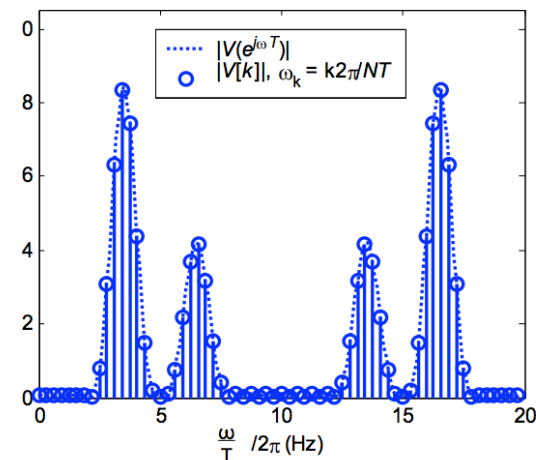
Sampled, Windowed Signal, Hamming Window, $L = 32$, Zero-Padded to $N = 64$



N -Point DFT of Sampled, Windowed, Zero-Padded Signal



Spectrum of Sampled, Windowed, Zero-Padded Signal





Frequency Analysis with DFT

- ❑ Length of window determines spectral resolution
- ❑ Type of window determines side-lobe amplitude
 - Some windows have better tradeoff between resolution-sidelobe)
- ❑ Zero-padding approximates the DTFT better. Does not introduce new information!



Potential Problems and Solutions

- ❑ 1. Spectral error
 - a. Filter signal to reduce frequency content above $\Omega_s/2 = \pi/T$.
 - b. Increase sampling frequency $\Omega_s = 2\pi/T$.
- ❑ 2. Insufficient frequency resolution
 - a. Increase L
 - b. Use window having narrow main lobe.
- ❑ 3. Spectral error from leakage
 - a. Use window having low side lobes.
 - b. Increase L
- ❑ 4. Missing features due to spectral sampling
 - a. Increase L
 - b. Increase N by zero-padding $v[n]$ to length $N > L$

Time Dependent DFT

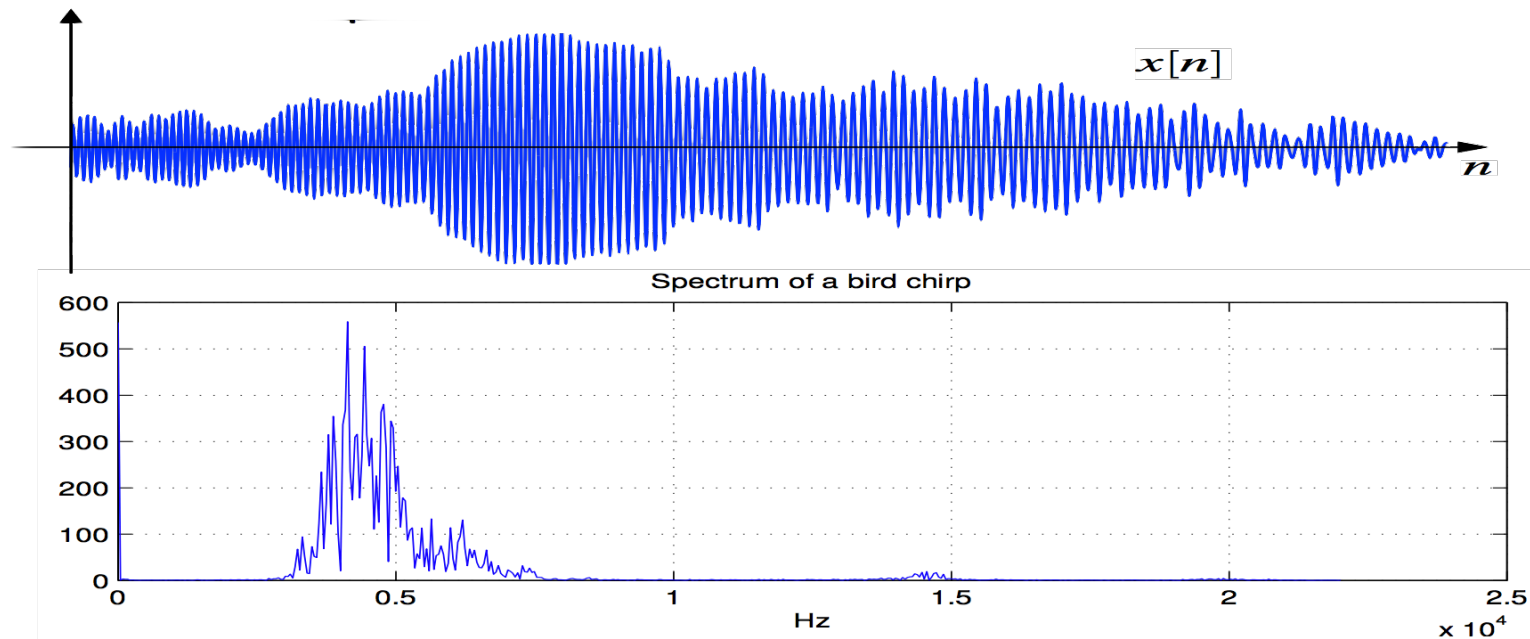


DFT

- ❑ DFT is only one out of a LARGE class of transforms
- ❑ Used for:
 - Analysis
 - Compression
 - Denoising
 - Detection
 - Recognition
 - Approximation (Sparse)

Example of Spectral Analysis

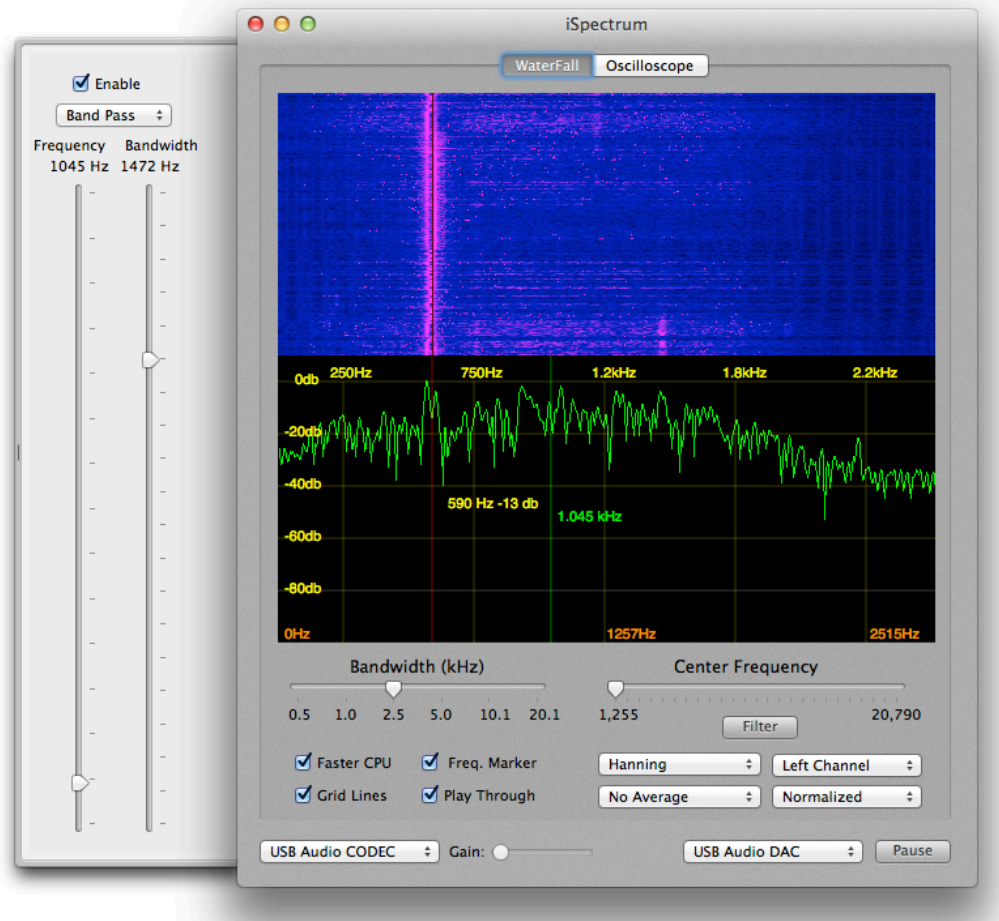
- Spectrum of a bird chirping
 - Interesting,... but...
 - Does not tell the whole story
 - No temporal information!





iSpectrum Demo

❑ <https://dogparksoftware.com/iSpectrum.html>





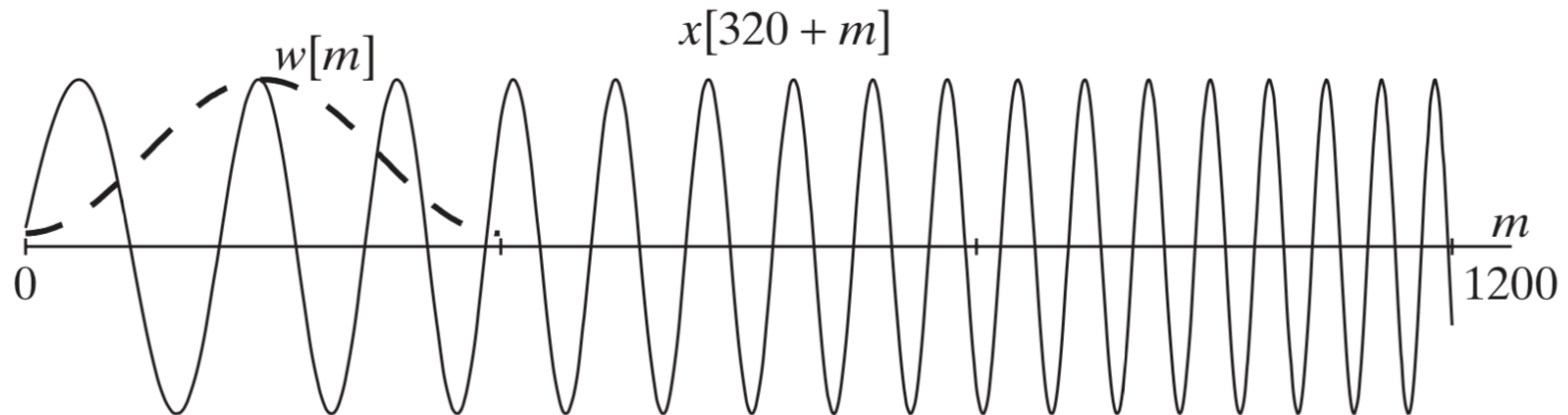
Time Dependent Fourier Transform

- ❑ Also called short-time Fourier transform
- ❑ To get temporal information, use part of the signal around every time point

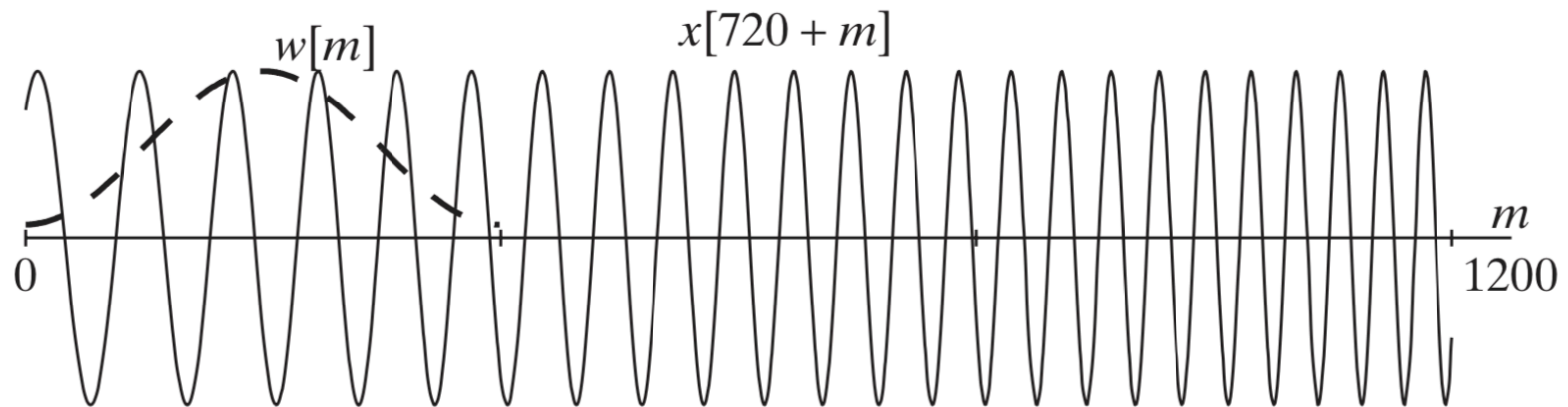
$$X[n, \lambda) = \sum_{m=-\infty}^{\infty} x[n+m]w[m]e^{-j\lambda m}$$

- ❑ Mapping from 1D $\rightarrow$ 2D, n discrete, λ cont.
- ❑ Simply slide a window and compute DTFT

Time Dependent Fourier Transform

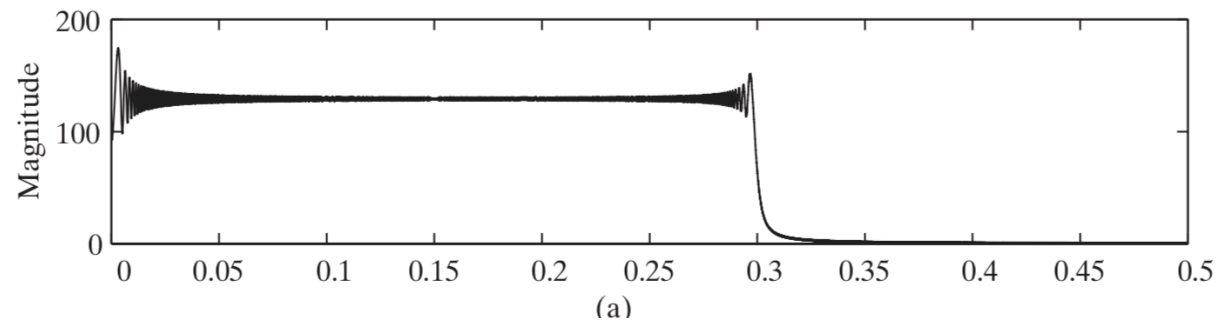


(a)

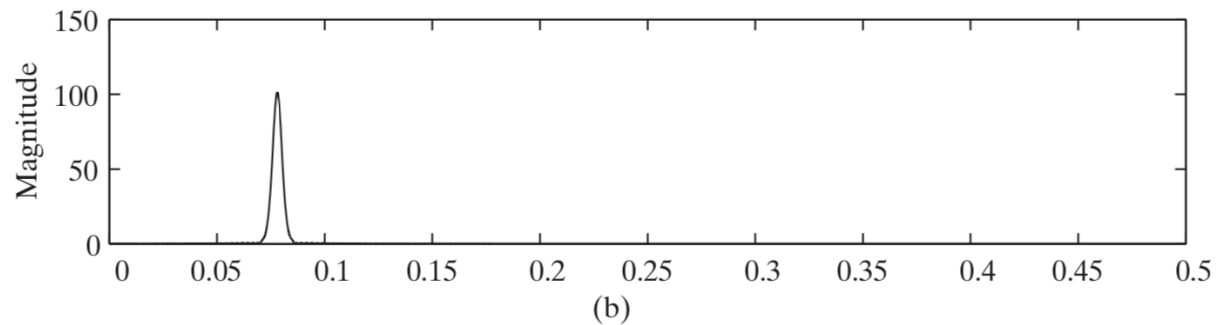
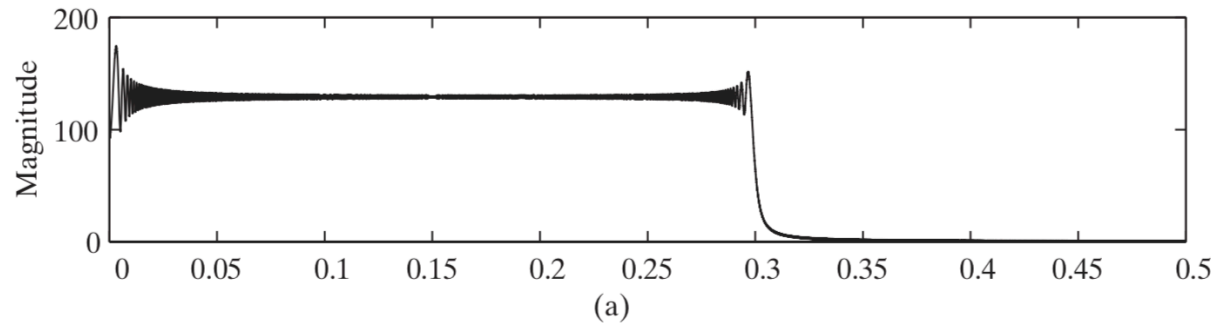


(b)

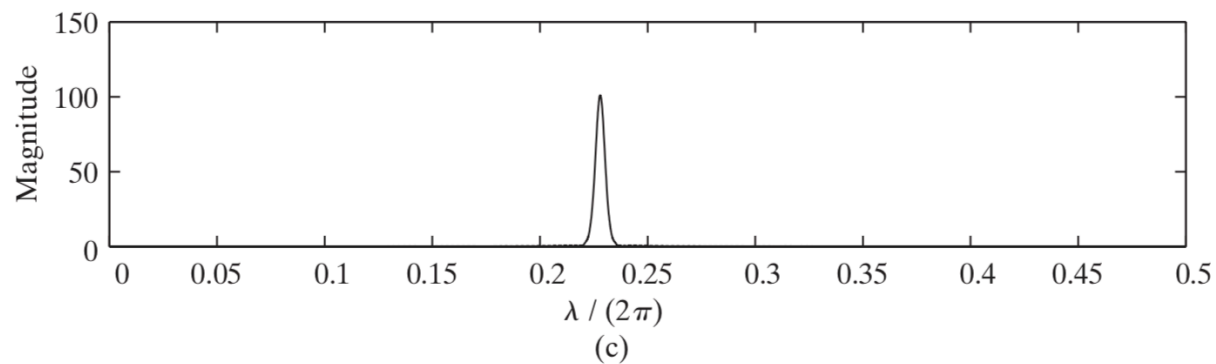
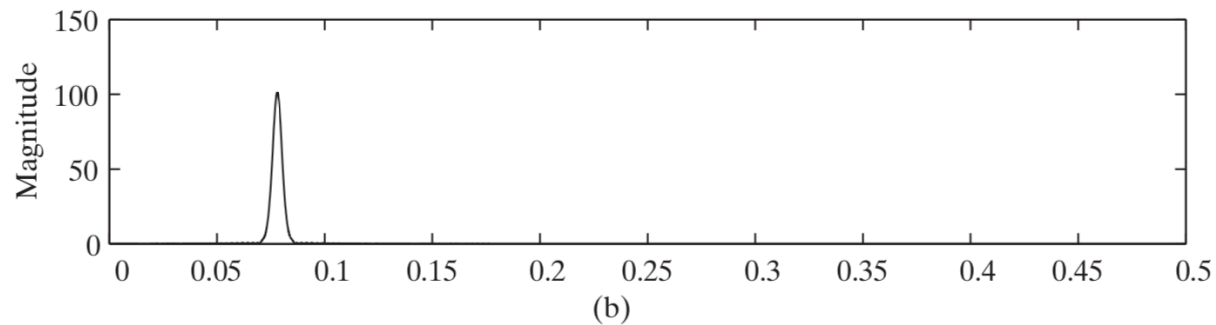
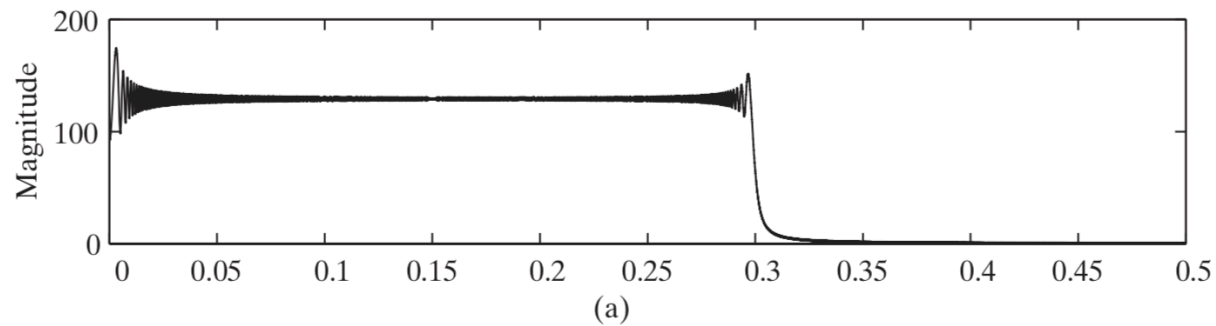
Time Dependent Fourier Transform



Time Dependent Fourier Transform



Time Dependent Fourier Transform





Spectrogram

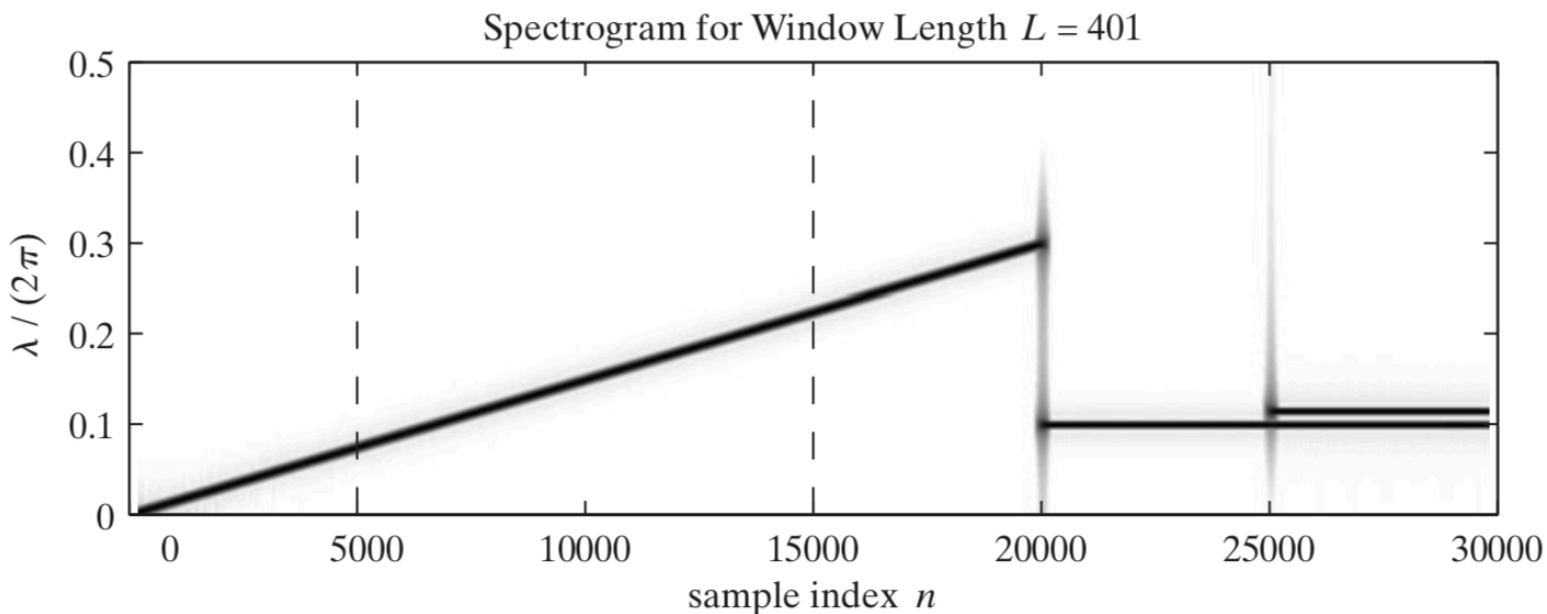
□ Plotting $Y[n, \lambda]$

$$y[n] = \begin{cases} 0 & n < 0 \\ \cos(\alpha_0 n^2) & 0 \leq n \leq 20,000 \\ \cos(0.2\pi n) & 20,000 < n \leq 25,000 \\ \cos(0.2\pi n) + \cos(0.23\pi n) & 25,000 < n. \end{cases}$$

Spectrogram

□ Plotting $Y[n, \lambda]$

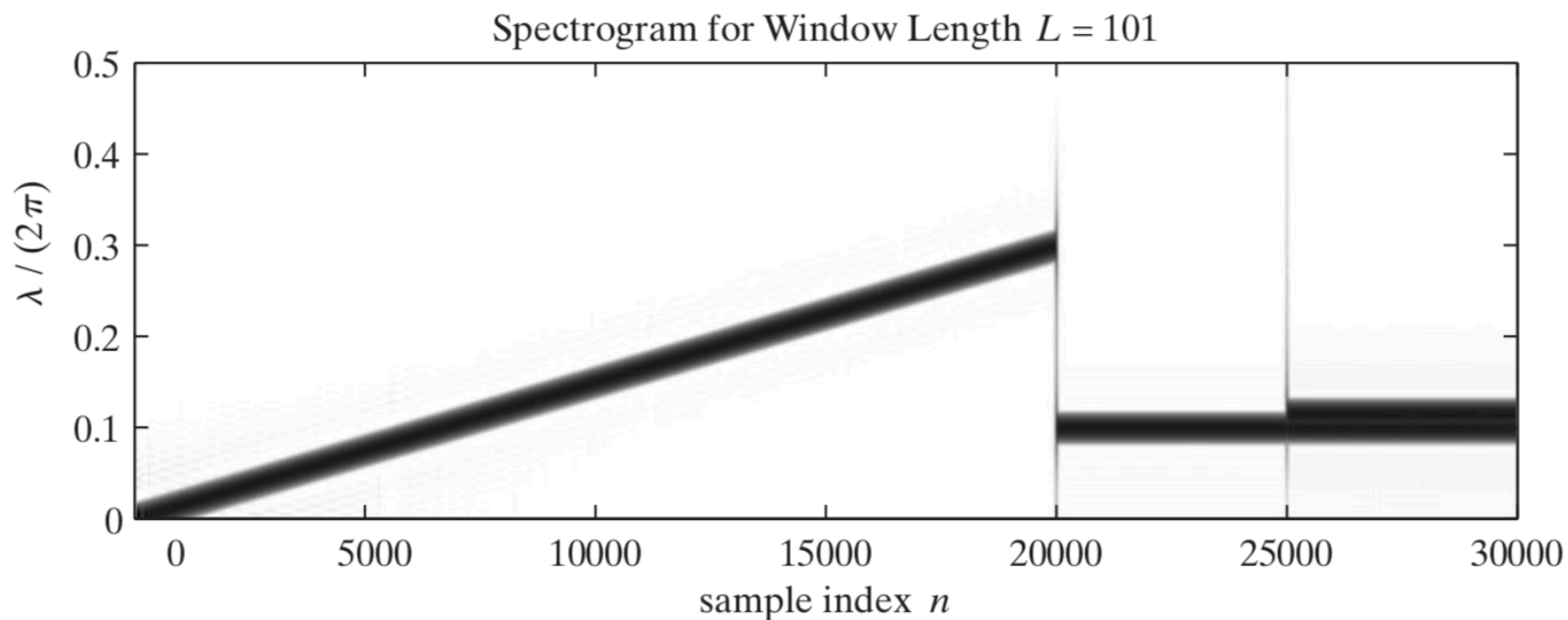
$$y[n] = \begin{cases} 0 & n < 0 \\ \cos(\alpha_0 n^2) & 0 \leq n \leq 20,000 \\ \cos(0.2\pi n) & 20,000 < n \leq 25,000 \\ \cos(0.2\pi n) + \cos(0.23\pi n) & 25,000 < n. \end{cases}$$



Spectrogram

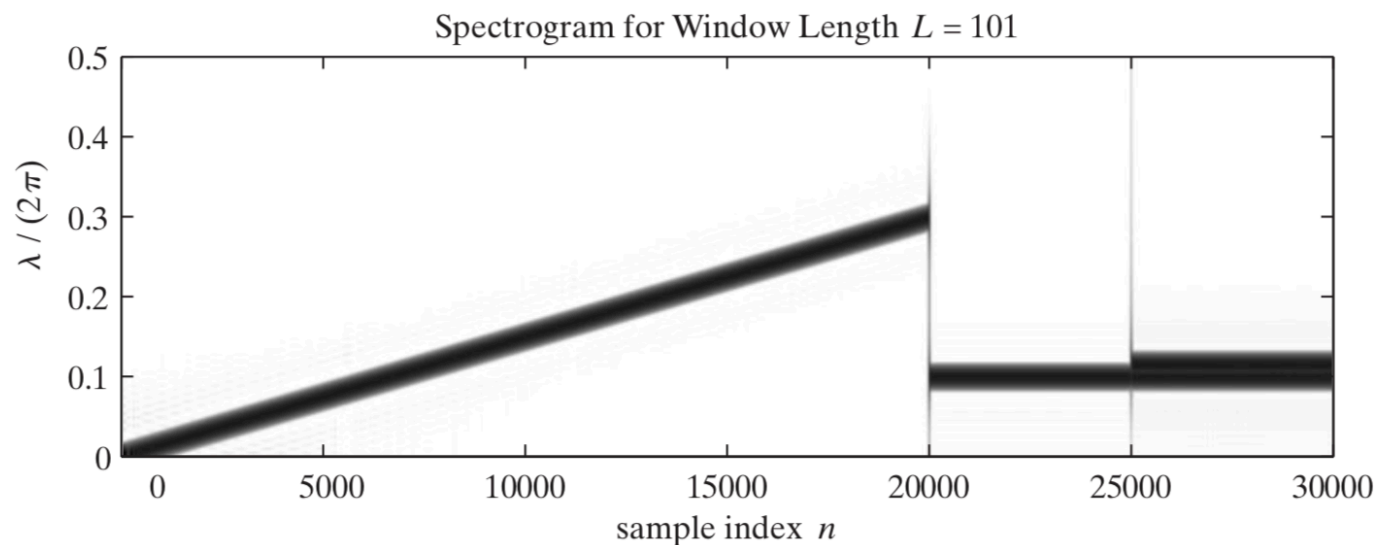
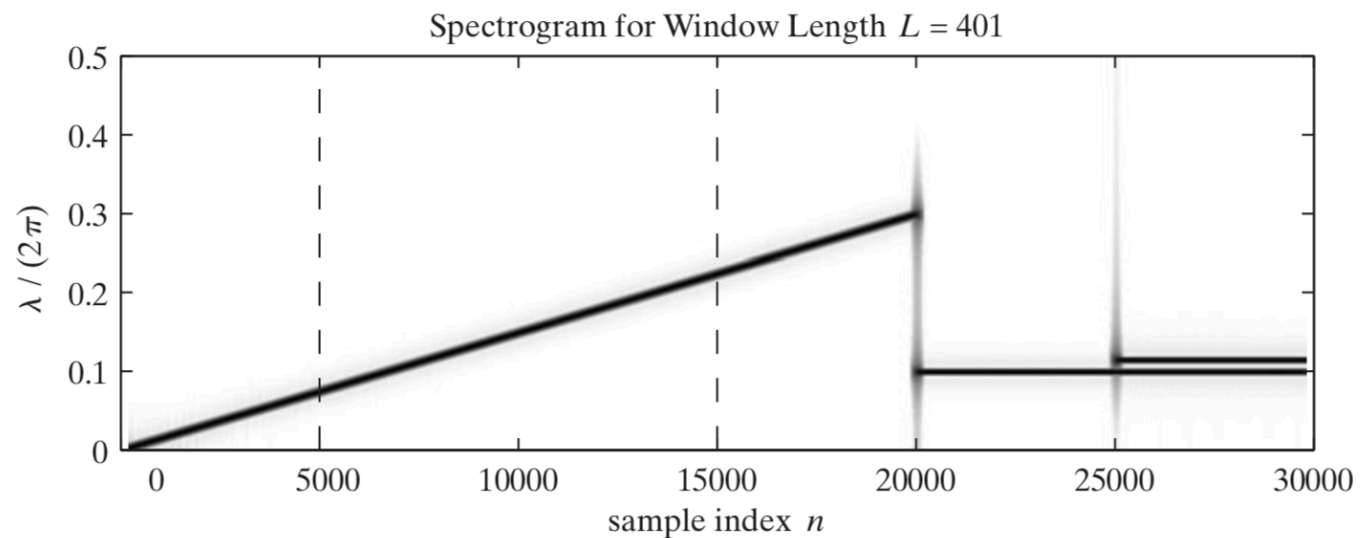
□ Plotting $Y[n, \lambda]$

$$y[n] = \begin{cases} 0 & n < 0 \\ \cos(\alpha_0 n^2) & 0 \leq n \leq 20,000 \\ \cos(0.2\pi n) & 20,000 < n \leq 25,000 \\ \cos(0.2\pi n) + \cos(0.23\pi n) & 25,000 < n. \end{cases}$$

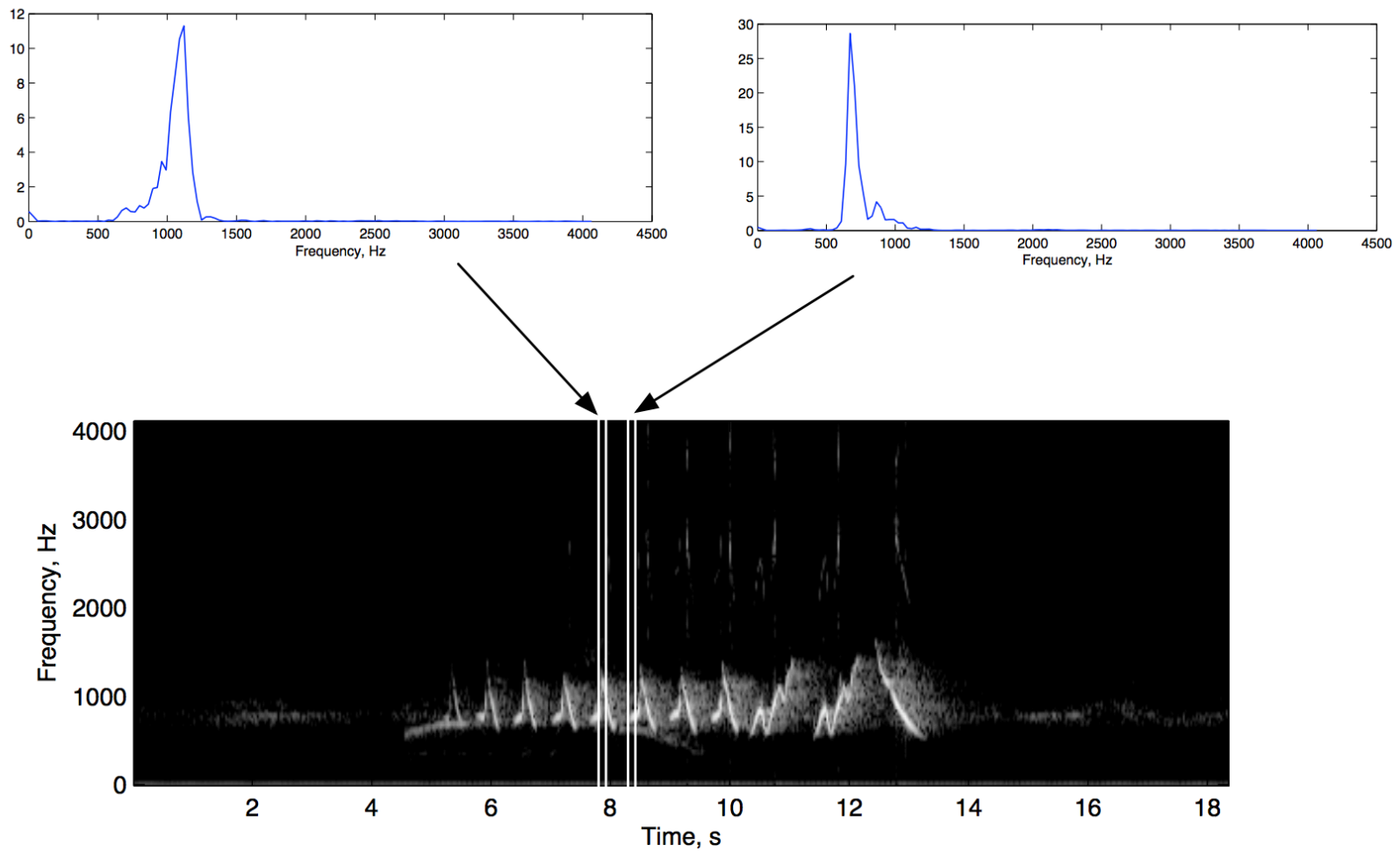




Spectrogram



Spectrogram Example





Discrete Time-Dependent Fourier Transform

$$X[n, \lambda) = \sum_{m=-\infty}^{\infty} x[n+m]w[m]e^{-j\lambda m}$$

$$X[rR, k] = X[rR, 2\pi k / N) = \sum_{m=0}^{L-1} x[rR+m]w[m]e^{-j(2\pi/N)km}$$

- ❑ L - Window length
- ❑ R - Jump of samples
- ❑ N - DFT length



Discrete Time-Dependent Fourier Transform

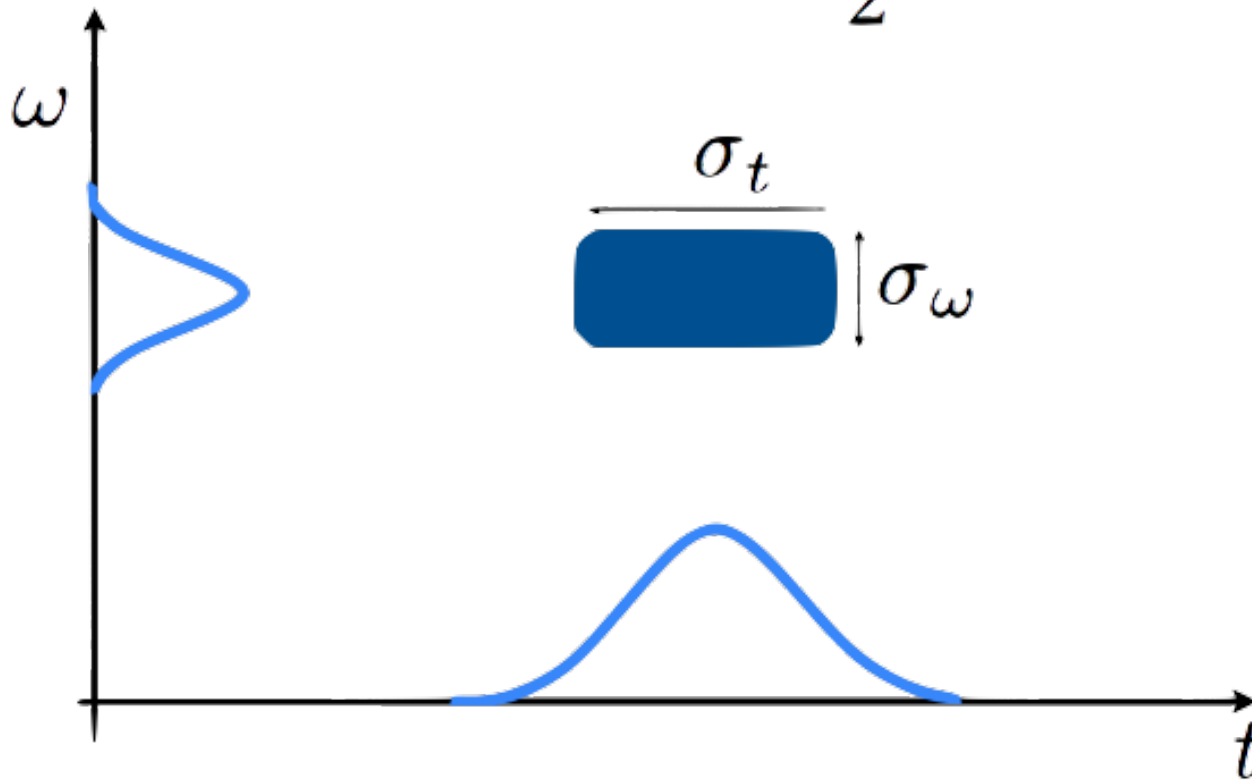
$$X[n, \lambda) = \sum_{m=-\infty}^{\infty} x[n+m]w[m]e^{-j\lambda m}$$

$$X[rR, k] = X[rR, 2\pi k / N) = \sum_{m=0}^{L-1} x[rR+m]w[m]e^{-j(2\pi/N)km}$$

$$X_r[k] = \sum_{m=0}^{L-1} x[rR+m]w[m]e^{-j(2\pi/N)km}$$

Time-Frequency Uncertainty Principle

$$\sigma_t \cdot \sigma_\omega \geq \frac{1}{2}$$



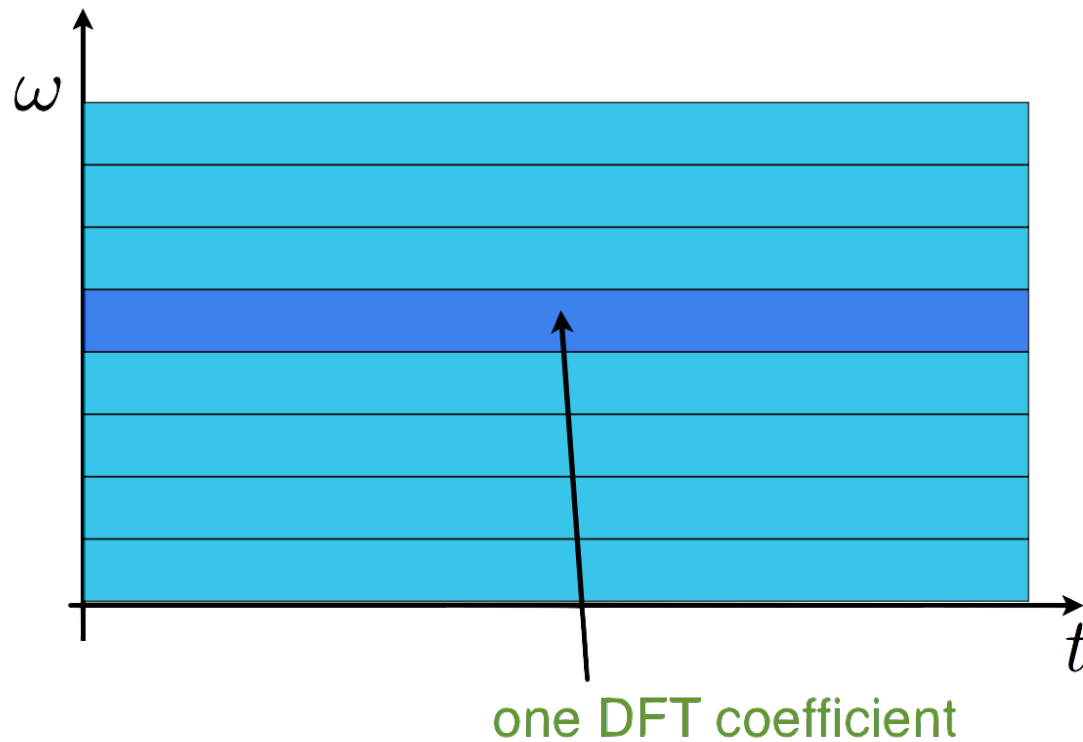
DFT

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-j2\pi kn/N}$$

$$\Delta\omega = \frac{2\pi}{N}$$

$$\Delta t = N$$

$$\Delta\omega \cdot \Delta t = 2\pi$$

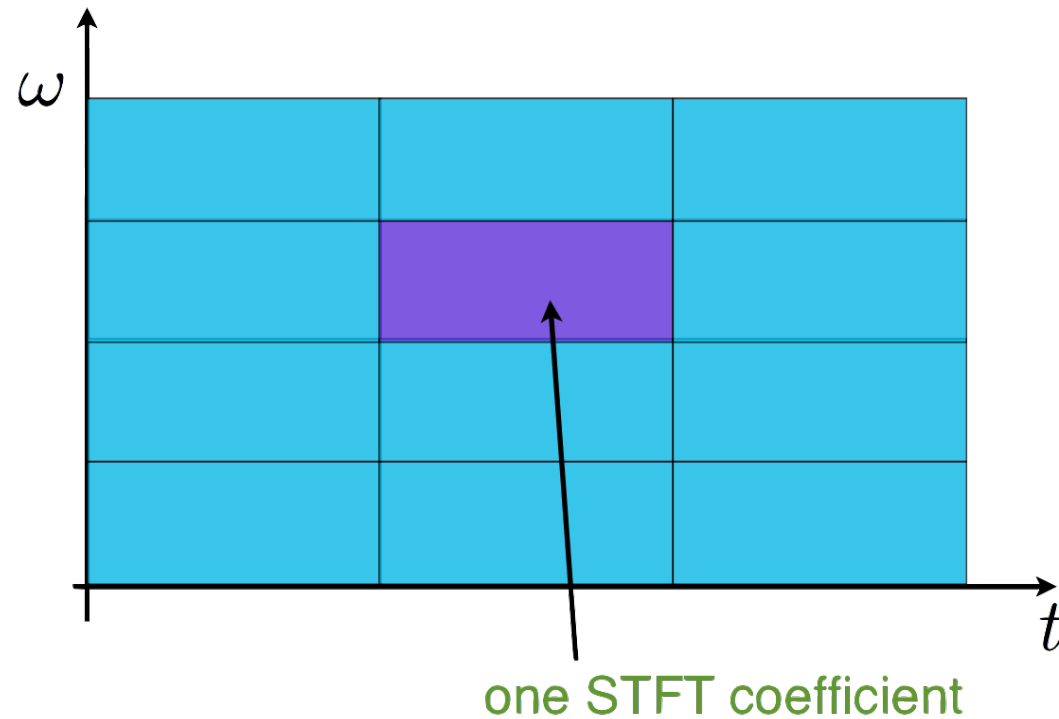


Discrete STFT

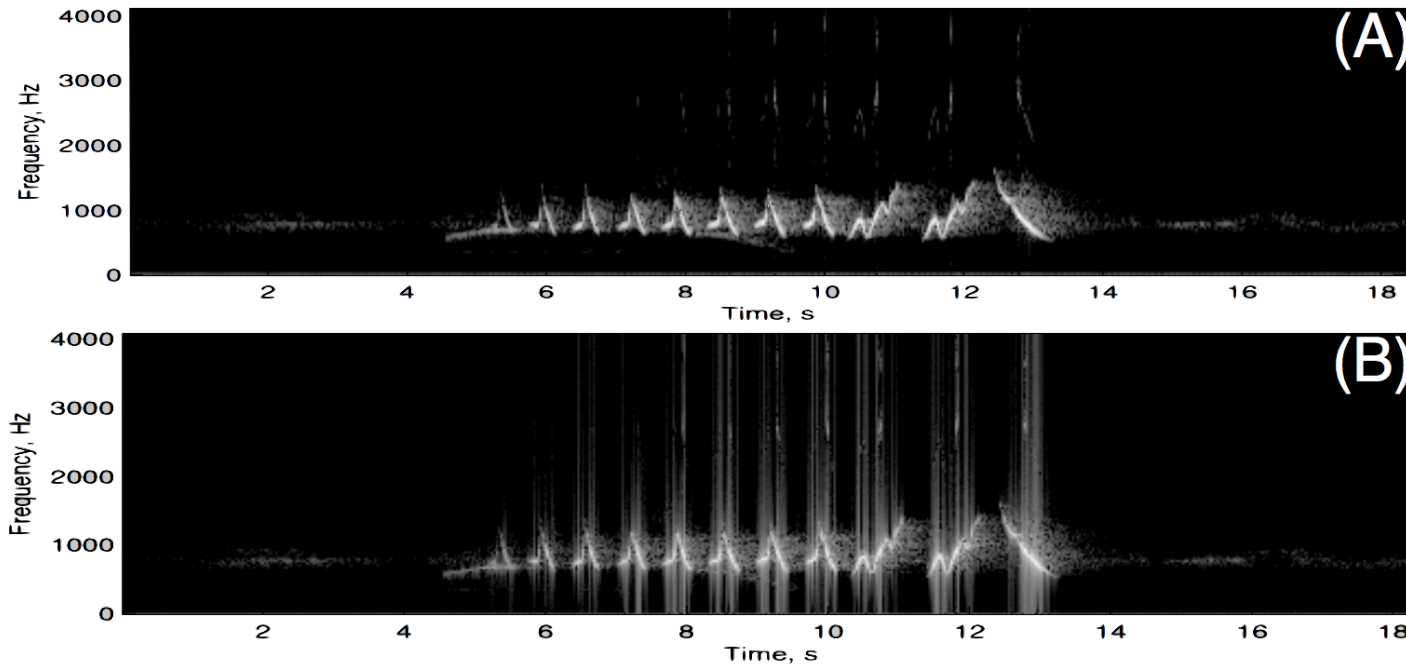
$$X[r, k] = \sum_{m=0}^{L-1} x[r \overset{\text{optional}}{\downarrow} R + m] w[m] e^{-j2\pi km/N}$$

$$\Delta\omega = \frac{2\pi}{L}$$

$$\Delta t = L$$

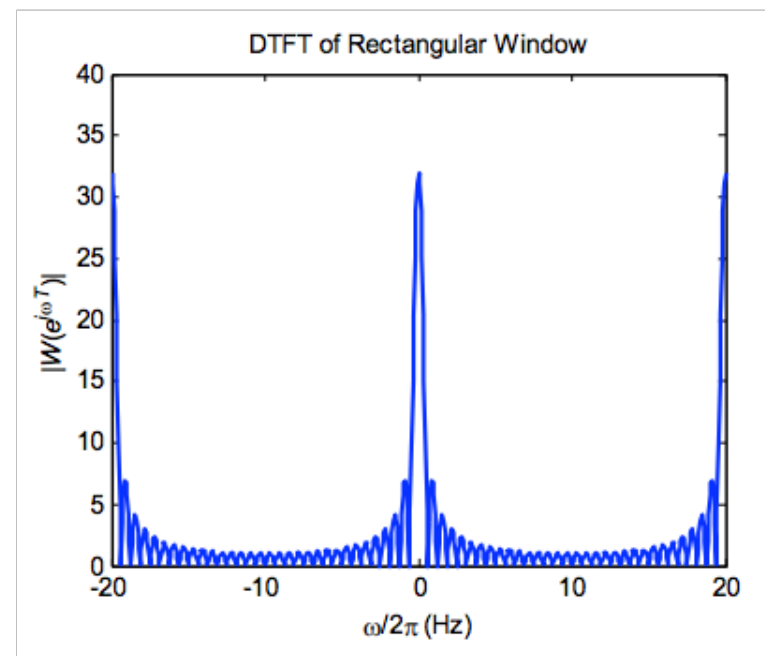
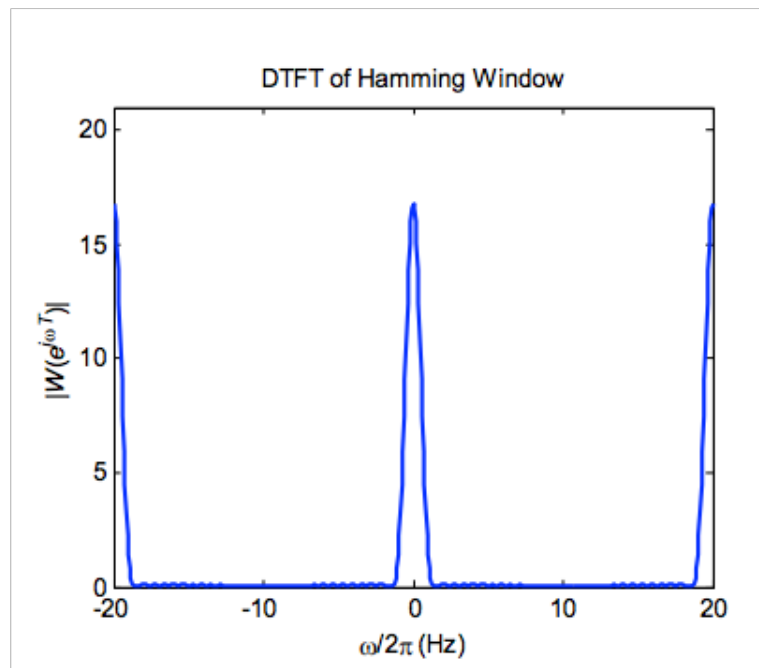


Spectrogram

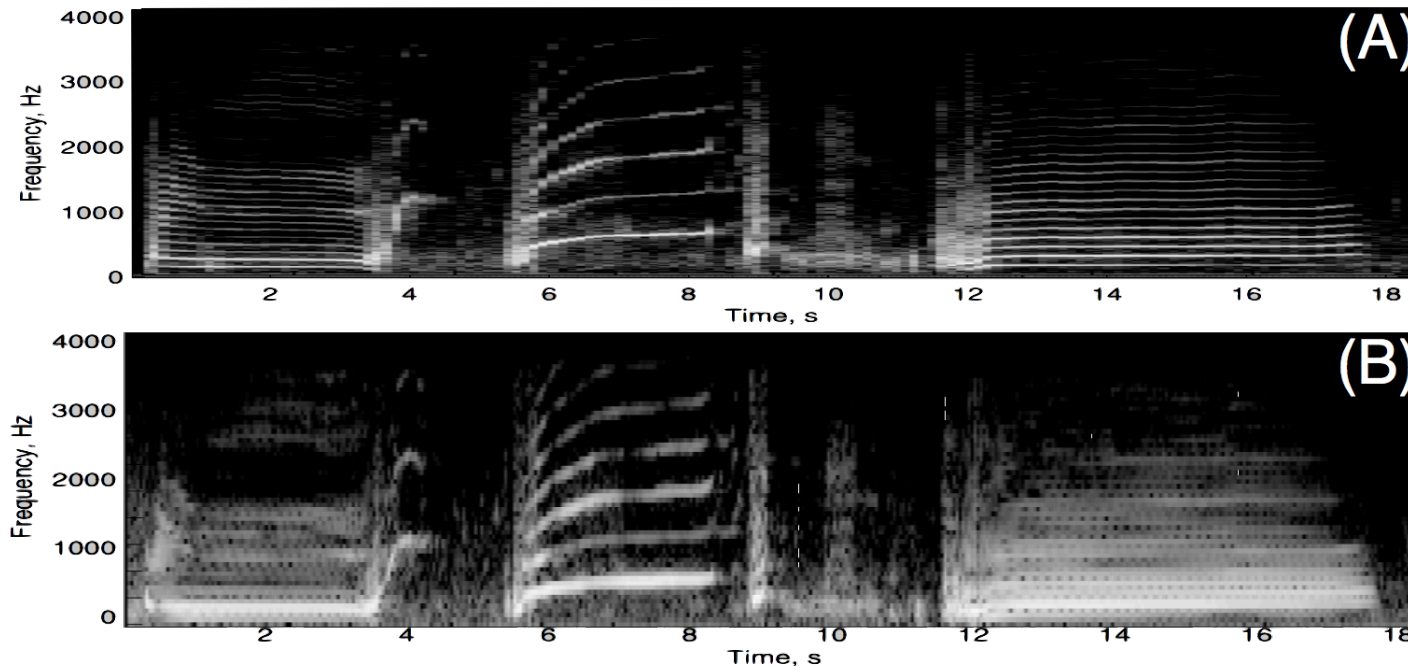


- What is the difference between the spectrograms?
 - a) Window size $B < A$
 - b) Window size $B > A$
 - c) Window type is different
 - d) (A) uses overlapping window

Sidelobes of Windows



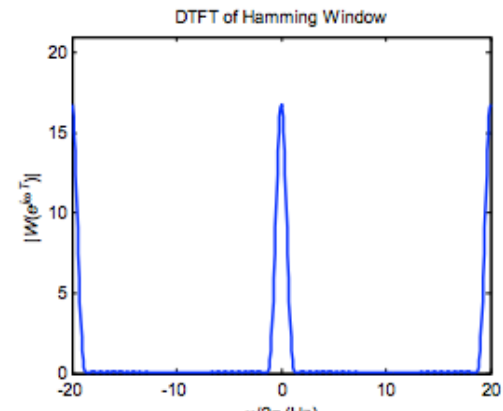
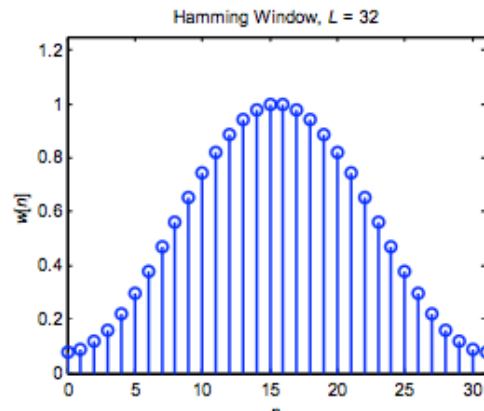
Spectrogram



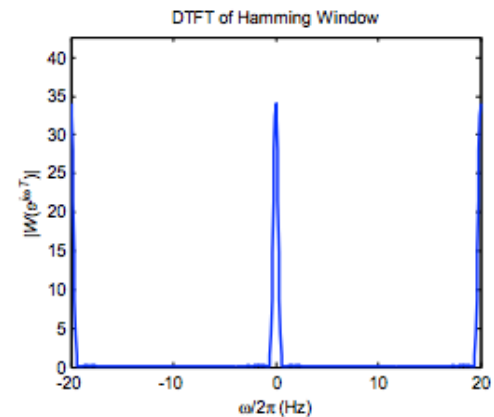
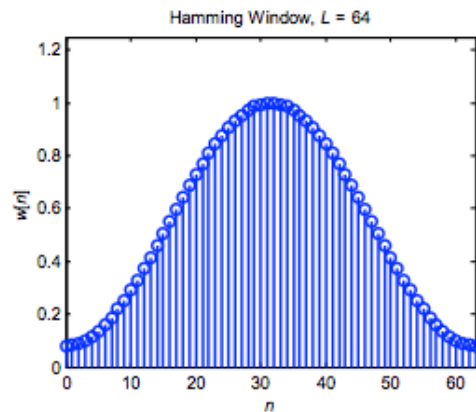
- What is the difference between the spectrograms?
 - a) Window size $B < A$
 - b) Window size $B > A$
 - c) Window type is different
 - d) (A) uses overlapping window

Window Size

Hamming Window, $L = 32$



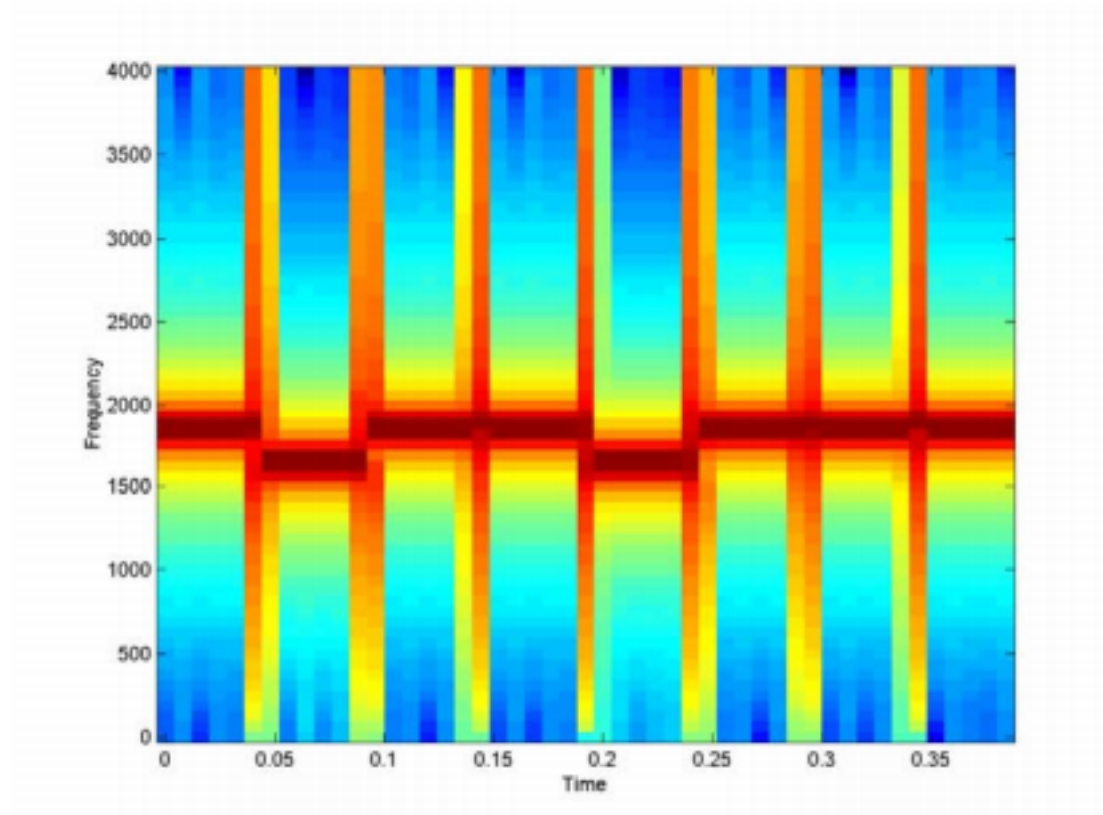
Hamming Window, $L = 64$



Application – Frequency Shift Keying

□ FSK Communications

- Spectrogram transmitting 'H' (ASCII H = 01001000)





STFT Reconstruction

- ❑ If $R \leq L \leq N$, then we can recover $x[n]$ block-by-block from $X_r[k]$
- ❑ For non-overlapping windows, $R=L$

$$x_r[m] = \frac{1}{N} \sum_{k=0}^{N-1} X_r[k] e^{j2\pi km/N}$$



STFT Reconstruction

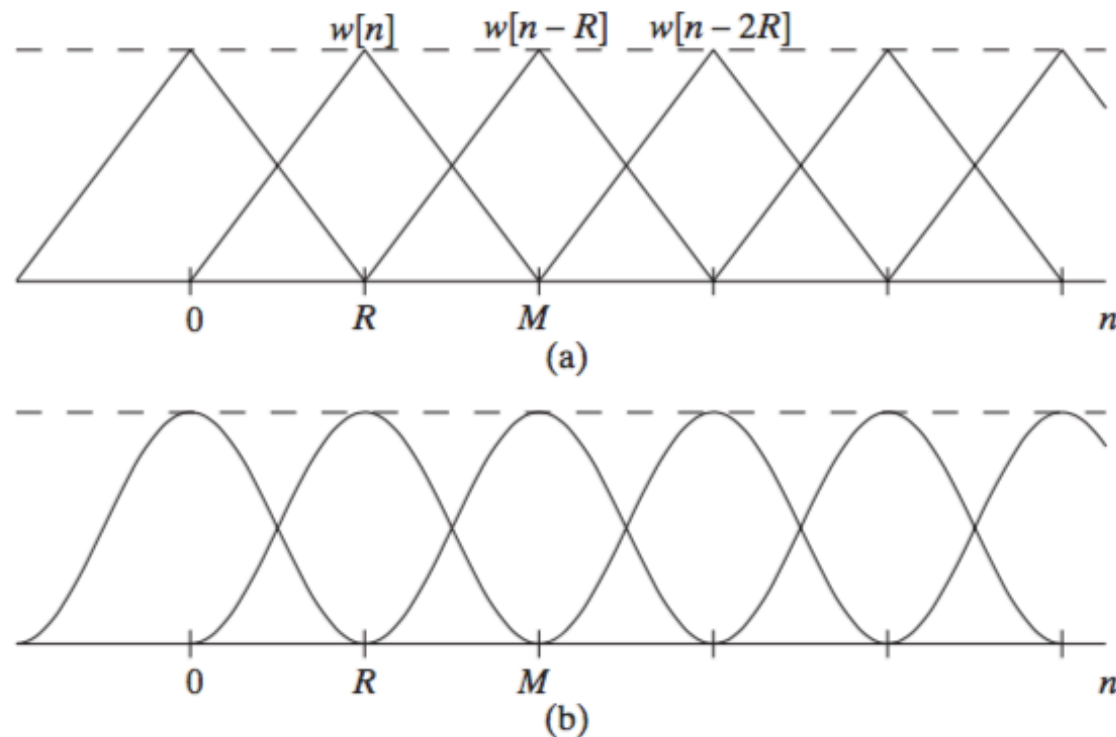
- If $R \leq L \leq N$, then we can recover $x[n]$ block-by-block from $X_r[k]$
- For non-overlapping windows, $R=L$

$$x_r[m] = \frac{1}{N} \sum_{k=0}^{N-1} X_r[k] e^{j2\pi km/N}$$

$$x[n] = \frac{x_r[n - rR]}{w[n - rR]} \quad \forall \quad rR \leq n \leq (r+1)R - 1$$

SFTF Reconstruction with overlap

- ❑ Practically make $R < L < N$
- ❑ If we choose R , L , and N appropriately with window, the overlap-add will negate the window effect





Big Ideas

- ❑ Frequency analysis with DFT
 - Nontrivial to choose sampling frequency, signal length, window type, DFT length (zero-padding)
 - Get accurate representation of DFT
- ❑ Time-dependent Fourier transform
 - Aka short-time Fourier transform
 - Includes temporal information about signal
 - Useful for many applications
 - Analysis, Compression, Denoising, Detection, Recognition, Approximation (Sparse)
 - Overlap for reconstruction



Admin

- Project

- Due 4/24