

2018 Fall
CTP431: Music and Audio Computing
Sound Representations

Graduate School of Culture Technology, KAIST
Juhan Nam

Outlines

- Introduction
- Time-domain representation
 - Waveform
- Frequency domain representation
 - Discrete Fourier Transform (DFT)
- Time-Frequency domain representation
 - Short-time Fourier Transform (STFT)
 - Spectrogram



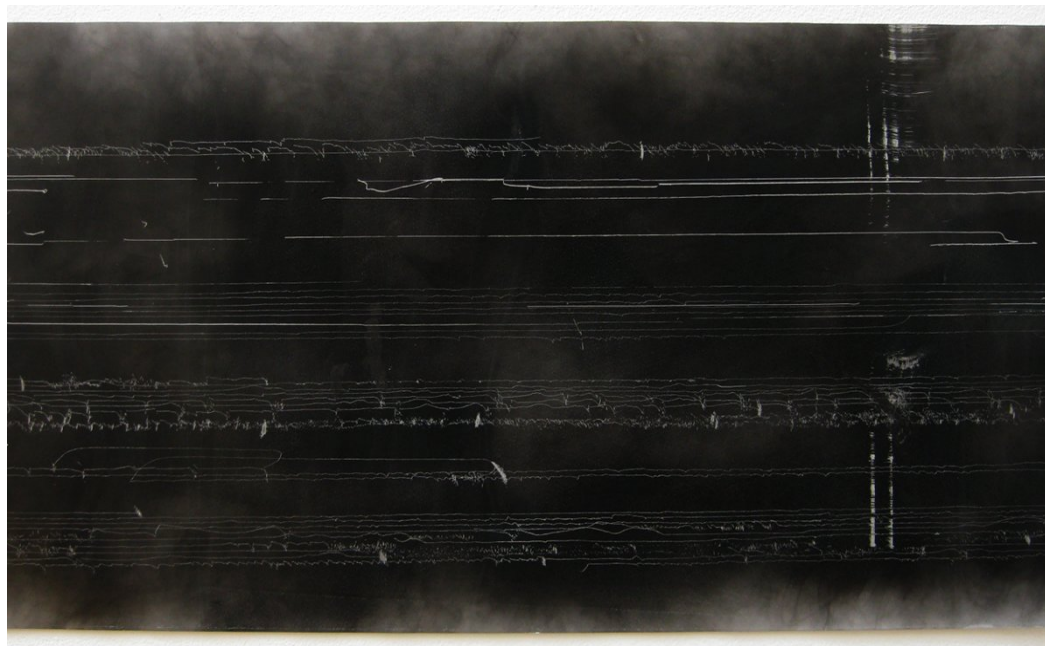
Introduction

- Visualizing sound as image or animation is very important
 - For research purpose
 - Analyzing the properties of sound: loudness, pitch and timbre
 - More complicated patterns in different contexts
 - For artistic purpose
 - Mapping the sound properties to visual elements
 - Visual elements become more important in music
- In this topic, we will focus on visualizing sound “as it is”



Time-domain Representation

- The raw waveform: the amplitude of sound over time
- Phonautograph (Leon Scott, 1857)
 - The first invention of sound recording
 - Recent research on image to sound restoration: <http://firstsounds.org/>



Source: <http://edcarter.net/home/phonautogram/>

Time-domain Representation

- Zoom-In view
 - Loudness: yes
 - Pitch: yes if the waveform is periodic (monophonic)
 - Timbre: to some extent from the wave shape (e.g. round or squared)
- Zoom-out view
 - Loudness: yes
 - Pitch: no
 - Timbre: to some extent from the amplitude envelop

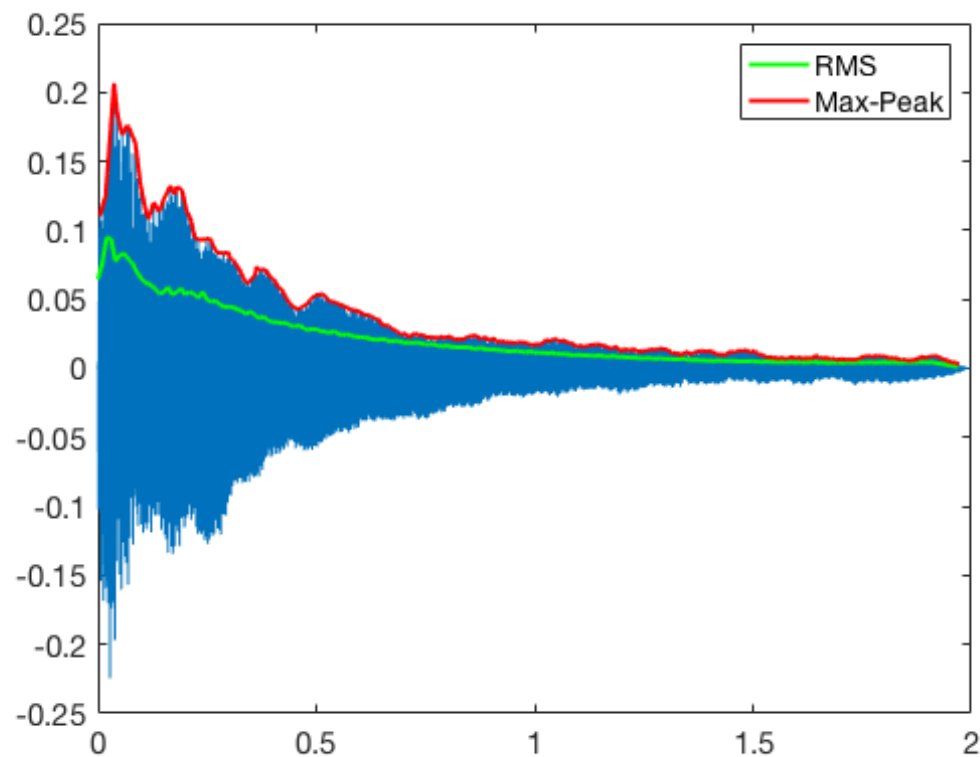


Amplitude Envelope

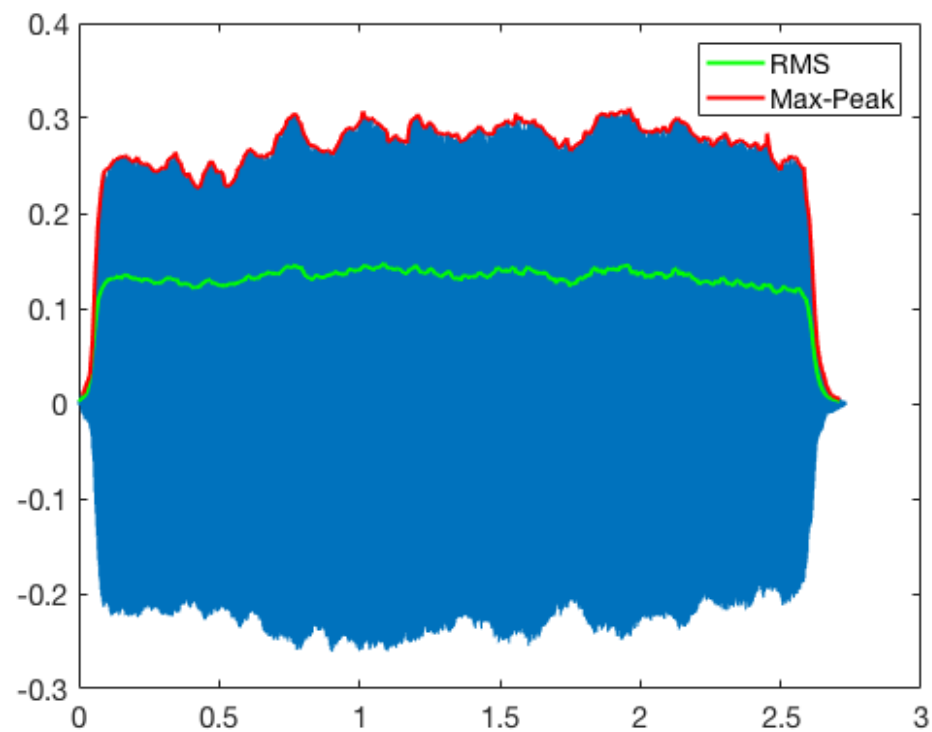
- Summarized visualization of the waveform
 - Computed by max-peak picking or root-mean-square (RMS)
- Parameterized with “ADSR” for musical tones
 - Attack time, Decay time, Sustain level and Release time
- Used to determine gain in dynamic range compression:
 - e.g. compressor, expander



Example: Amplitude Envelope



Piano C4 Note

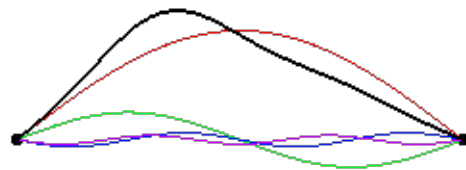


Flute A4 Note

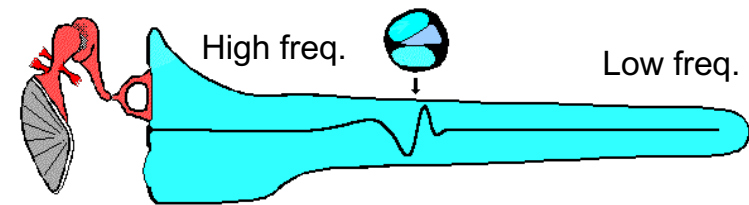


Tone Generation and Perception Perspective

- Musical tones are generated as a combination of (sinusoidal) oscillation modes
- Cochlear has frequency selective responses



Modes



Cochlear

Source: <https://www.acs.psu.edu/drussell/Demos/string/Fixed.html>

Source: <http://acousticslab.org/psychoacoustics/PMFiles/Module03a.htm>

Frequency-Domain Representation

- Can we represent $x(n)$ with a finite set of sinusoids?

- $x(n) = \frac{1}{N} \sum_{k=0}^{N-1} A(k) r_k(n)$

- $r_k(n) = \cos(\frac{2\pi kn}{N} + \phi(k))$: discrete-time sinusoid with length N

- Find $A(k), \phi(k)$



Euler's identity

- Euler's identity

$$e^{j\theta} = \cos\theta + j\sin\theta$$

- Can be proved by Taylor's series
- If $\theta = \pi$, $e^{j\pi} + 1 = 0$ (“the most beautiful equation in math”)

- Properties

$$\cos\theta = \frac{e^{j\theta} + e^{-j\theta}}{2}$$

$$\sin\theta = \frac{e^{j\theta} - e^{-j\theta}}{2j}$$

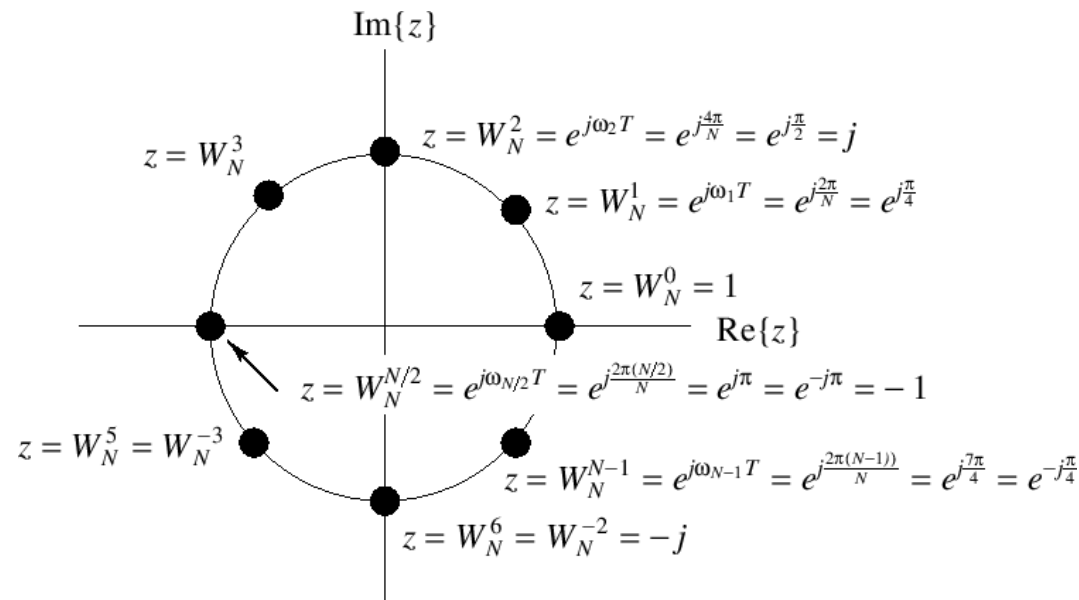


Complex Sinusoids

- Cosine and sine can be represented in a single term

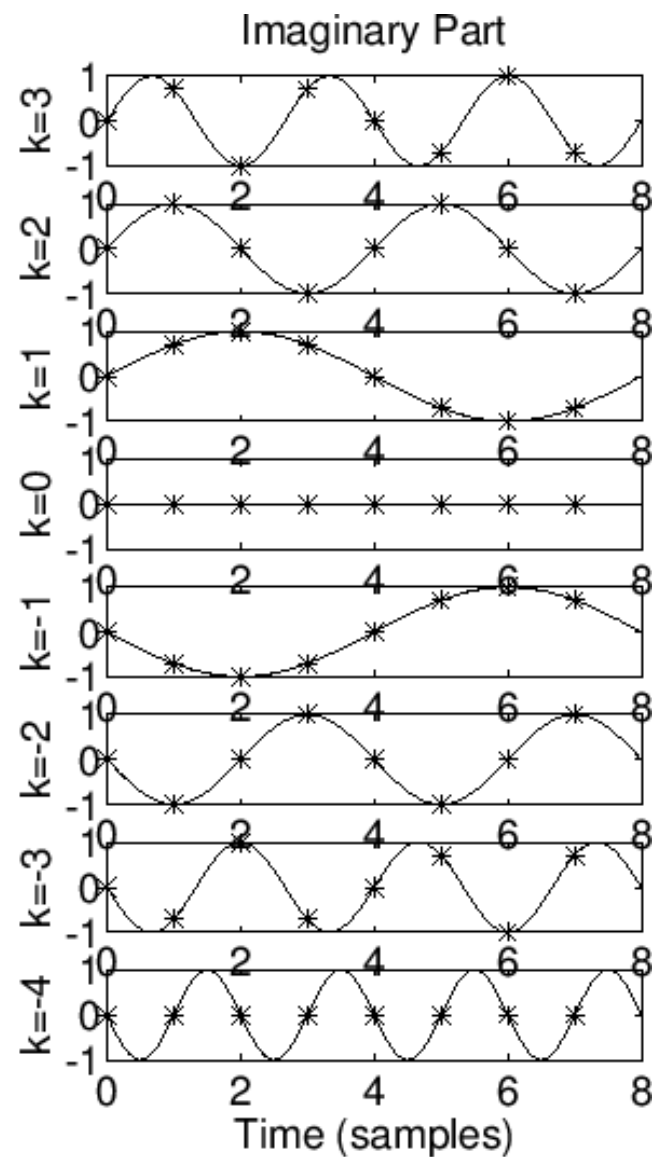
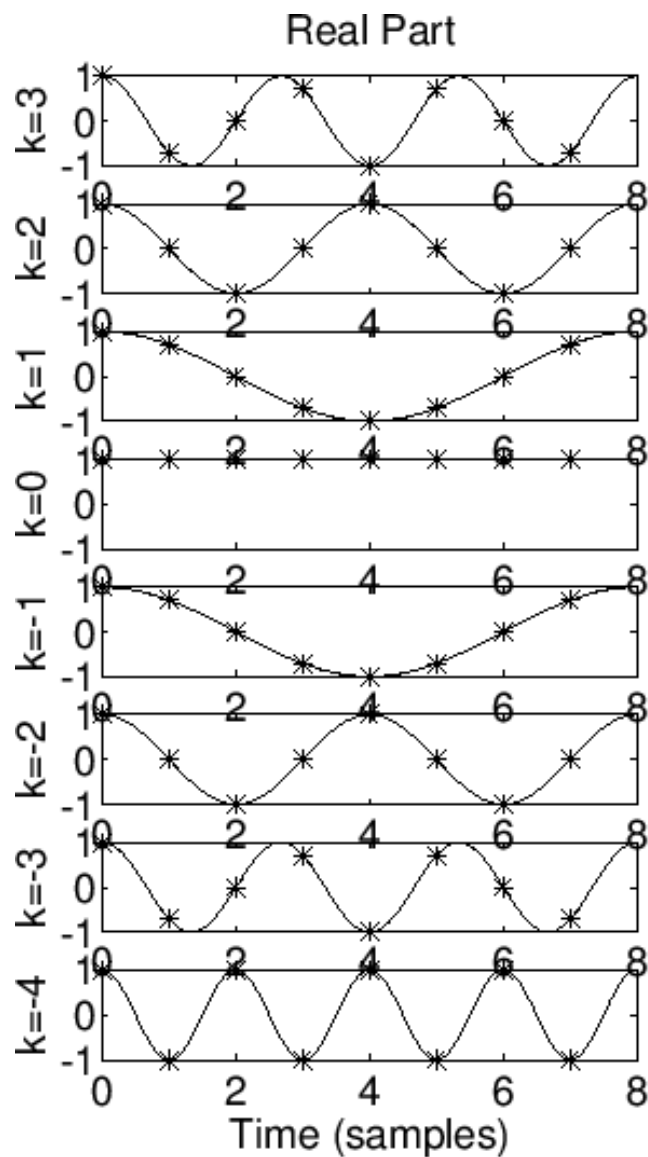
$$s_k(n) = e^{j\frac{2\pi kn}{N}} = \cos \frac{2\pi kn}{N} + j \sin \frac{2\pi kn}{N}$$

- Frequencies: $\frac{2\pi k}{N}$ radian or $\frac{k}{N} F_s$ Hz (F_s : the sampling rate) ($K = 0, 1, 2, \dots, N - 1$)
- Example: $N = 8$



Complex Sinusoids

$N = 8$



Frequency-Domain Representation Using Complex Sinusoids

- $x(n)$ is expressed in a simpler form:

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} A(k) \cos\left(\frac{2\pi kn}{N} + \phi(k)\right)$$

$$= \frac{1}{N} \sum_{k=0}^{N-1} A(k) (e^{j(\frac{2\pi kn}{N} + \phi(k))} + e^{-j(\frac{2\pi kn}{N} + \phi(k))}) / 2 = \frac{1}{N} \sum_{k=0}^{N-1} (A(k) e^{j\phi(k)} e^{j\frac{2\pi kn}{N}} + A(k) e^{-j\phi(k)} e^{-j\frac{2\pi kn}{N}}) / 2$$

$$= \frac{1}{N} \sum_{k=0}^{N-1} (X(k) e^{j\frac{2\pi kn}{N}} + \overline{X(k)} e^{-j\frac{2\pi kn}{N}}) / 2 = \text{Real}\left\{ \frac{1}{N} \sum_{k=0}^{N-1} X(k) e^{j\frac{2\pi kn}{N}} \right\}$$

$$= \frac{1}{N} \sum_{k=0}^{N-1} X(k) e^{j\frac{2\pi kn}{N}}$$

$X(k) = A(k) e^{j\phi(k)} = A(k) (\cos \phi(k) + j \sin \phi(k))$

- Now, how can we find $X(k)$?

Orthogonality of Sinusoids

- Inner product between two complex sinusoids

$$s_p(n) \cdot s_q^*(n) = \sum_{n=0}^{N-1} e^{j\frac{2\pi pn}{N}} \cdot e^{-j\frac{2\pi qn}{N}} = \begin{cases} N & \text{if } p = q \\ 0 & \text{otherwise} \end{cases}$$

$$\sum_{n=0}^{N-1} \sin(2\pi pn / N) \sin(2\pi qn / N) = \begin{cases} 0 & \text{otherwise} \\ N / 2 & \text{if } p = q \\ -N / 2 & \text{if } p = N - q \end{cases} \quad \sum_{n=0}^{N-1} \cos(2\pi pn / N) \sin(2\pi qn / N) = 0$$

$$\sum_{n=0}^{N-1} \cos(2\pi pn / N) \cos(2\pi qn / N) = \begin{cases} N / 2 & \text{if } p = q \text{ or } p = N - q \\ 0 & \text{otherwise} \end{cases}$$

Orthogonal Projection on Complex Sinusoids

- Do the inner product with the signal and sinusoids

$$\begin{aligned}x(n) \cdot s_q(n) &= \sum_{n=0}^{N-1} x(n) e^{-j\frac{2\pi qn}{N}} = \sum_{n=0}^{N-1} \left(\frac{1}{N} \sum_{p=0}^{N-1} X(k) e^{j\frac{2\pi pn}{N}} \right) e^{-j\frac{2\pi qn}{N}} \\&= \frac{1}{N} \sum_{p=0}^{N-1} X(k) \left(\sum_{n=0}^{N-1} e^{j\frac{2\pi pn}{N}} e^{-j\frac{2\pi qn}{N}} \right) = \frac{1}{N} X(k) N = X(k) = A(k) e^{j\phi(k)}\end{aligned}$$

To Wrap Up

- Discrete Fourier Transform

$$X(k) = \sum_{n=0}^{N-1} x(n)e^{-j\frac{2\pi kn}{N}} = X_R(k) + jX_I(k) = A(k)e^{j\phi(k)}$$

- Magnitude spectrum: $|X(k)| = A(k) = \sqrt{X_R^2(k) + X_I^2(k)}$
- Phase spectrum: $\angle X(k) = \phi(k) = \tan^{-1}\left(\frac{X_I(k)}{X_R(k)}\right)$

- Inverse Discrete Fourier Transform

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k)e^{j\frac{2\pi kn}{N}}$$

Properties of DFT

- Periodicity

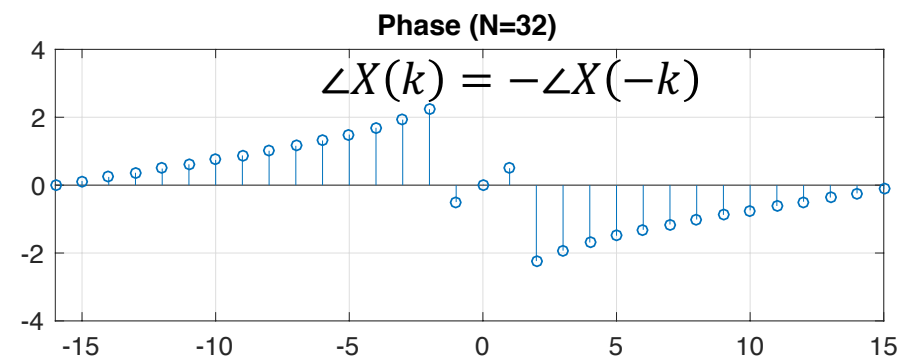
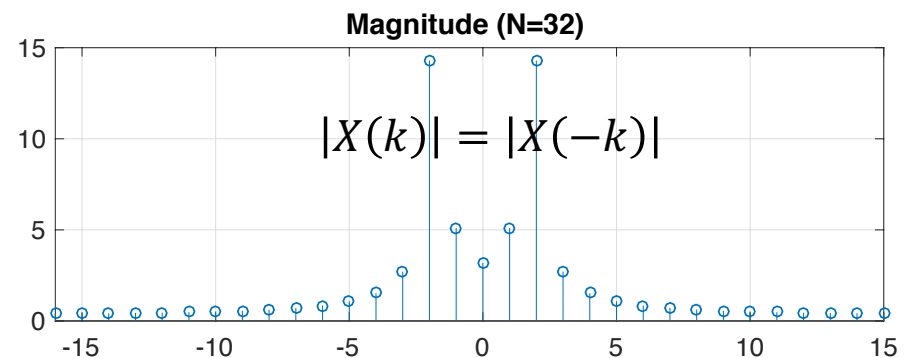
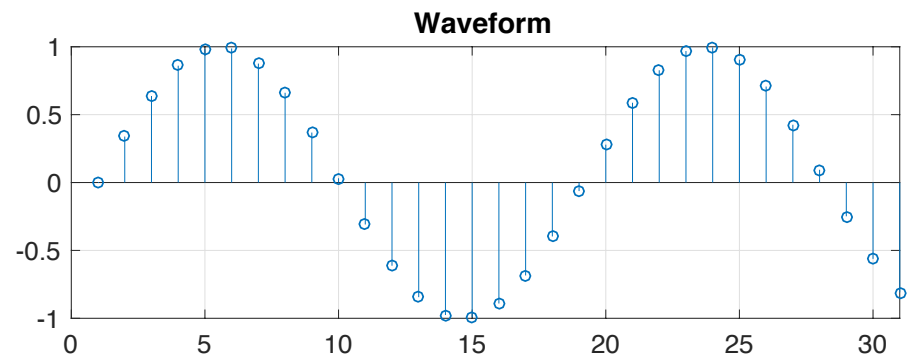
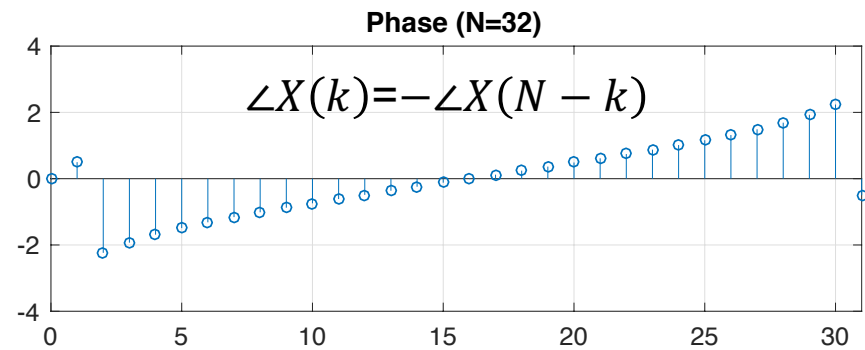
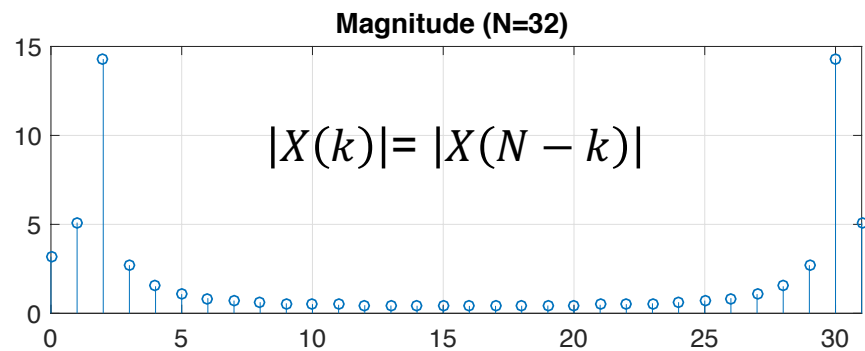
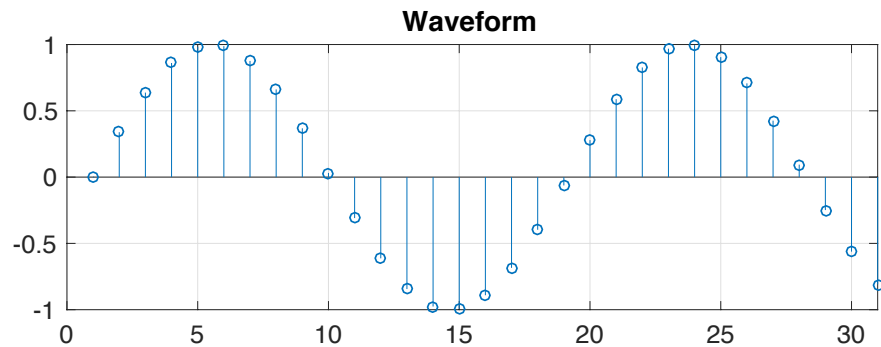
- $X(k) = X(k + N) = X(k + 2N) = \dots$
- $X(k) = X(k - N) = X(k - 2N) = \dots$

- Symmetry

- Magnitude response: $|X(k)| = |X(-k)| = |X(N - k)|$
- Phase response : $\angle X(k) = -\angle X(-k) = -\angle X(N - k)$
- We often display only half the amplitude and phase responses

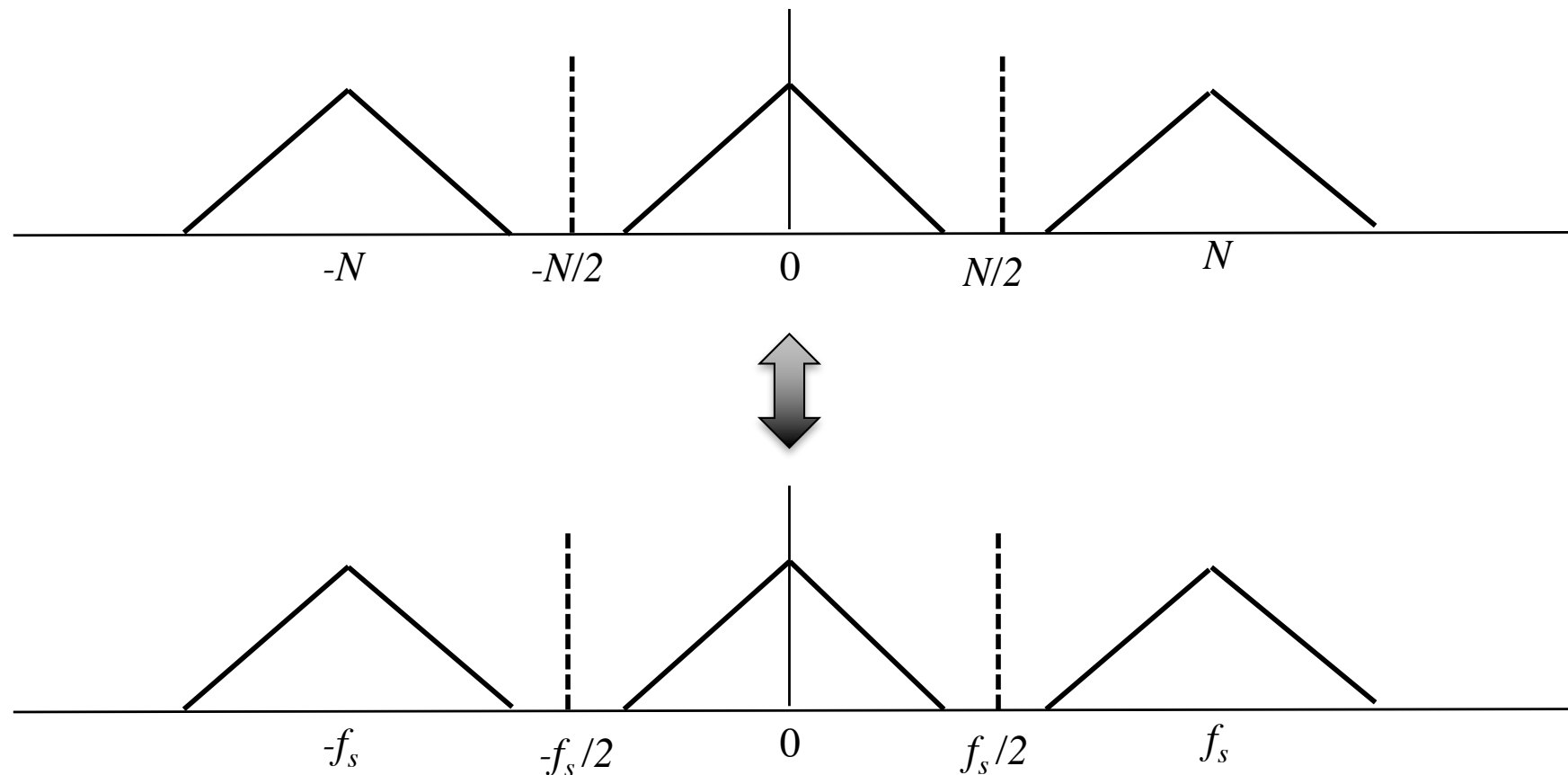


Properties of DFT

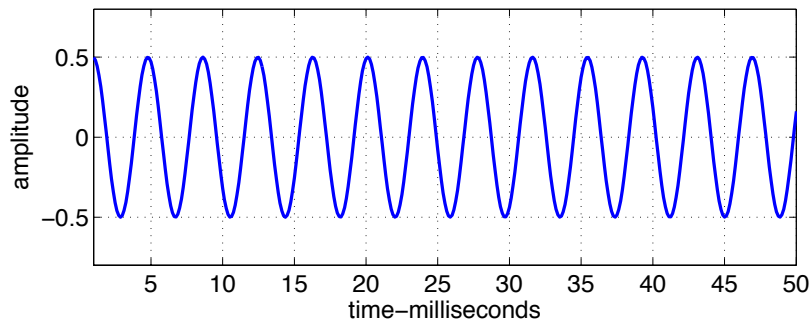


Frequency Scaling

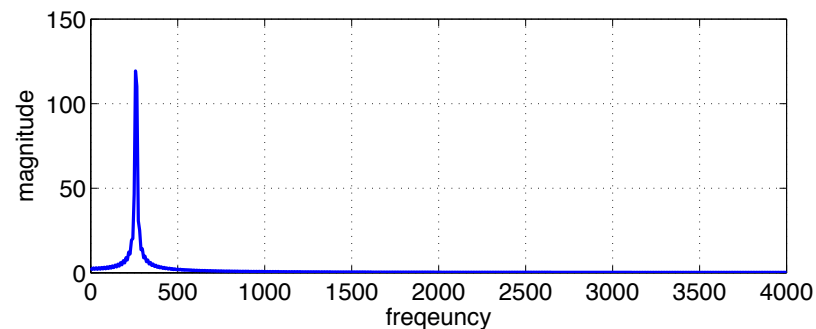
- $X(k)$ ($k = 0, 1, \dots, N$) corresponds to frequency values that are evenly distributed between 0 and f_s in Hz



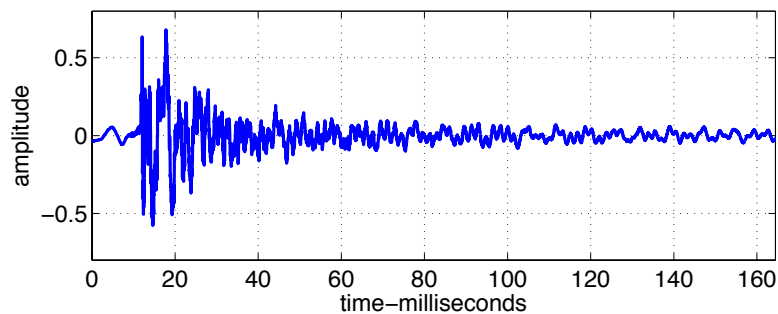
Examples of DFT



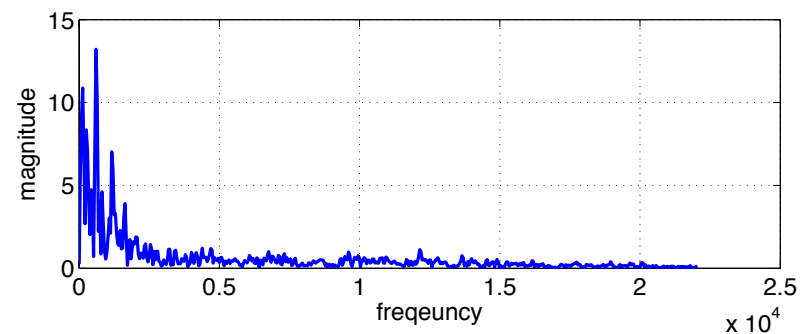
Sine: waveform



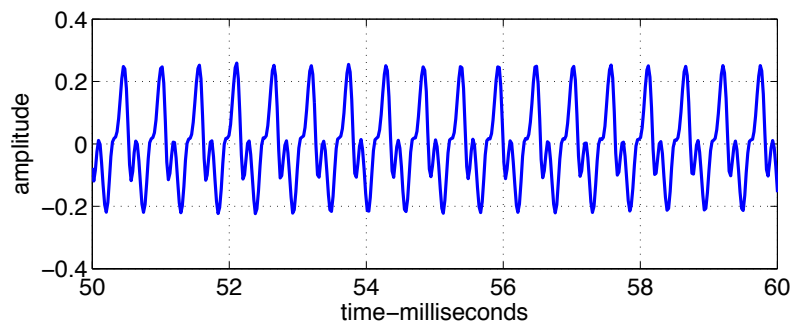
Sine: spectrum



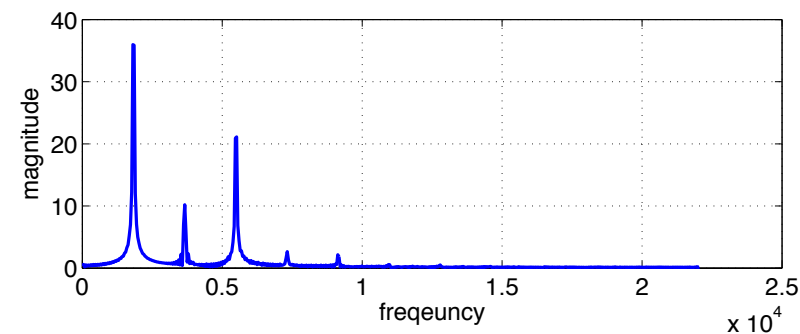
Drum: waveform



Drum: spectrum



Flute: waveform



Flute: spectrum



Fast Fourier Transform (FFT)

- Matrix multiplication view of DFT

$$\begin{bmatrix} X(0) \\ X(1) \\ X(2) \\ X(3) \\ \vdots \\ X(N-2) \\ X(N-1) \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & W_N & W_N^2 & \dots & W_N^{N-1} \\ 1 & W_N^2 & W_N^4 & \dots & W_N^{2(N-1)} \\ 1 & W_N^3 & W_N^6 & \dots & W_N^{3(N-1)} \\ \vdots & \vdots & \vdots & \dots & \vdots \\ 1 & W_N^{N-1} & W_N^{2(N-1)} & \dots & W_N^{(N-1)(N-1)} \end{bmatrix} \begin{bmatrix} x(0) \\ x(1) \\ x(2) \\ x(3) \\ \vdots \\ x(N-2) \\ x(N-1) \end{bmatrix}$$

- In fact, we don't compute this directly. There is a more efficient way, which is called "Fast Fourier Transform (FFT)"
 - Complexity reduction by FFT: $O(N^2) \rightarrow O(N \log_2 N)$
 - Divide and conquer

Time-Frequency Domain Representation

- DFT assumes that the signal is stationary
 - It is not a good idea to apply DFT to a long and dynamically changing signal like music
 - Instead, we segment the signal and apply DFT separately

- Short-Time Fourier Transform

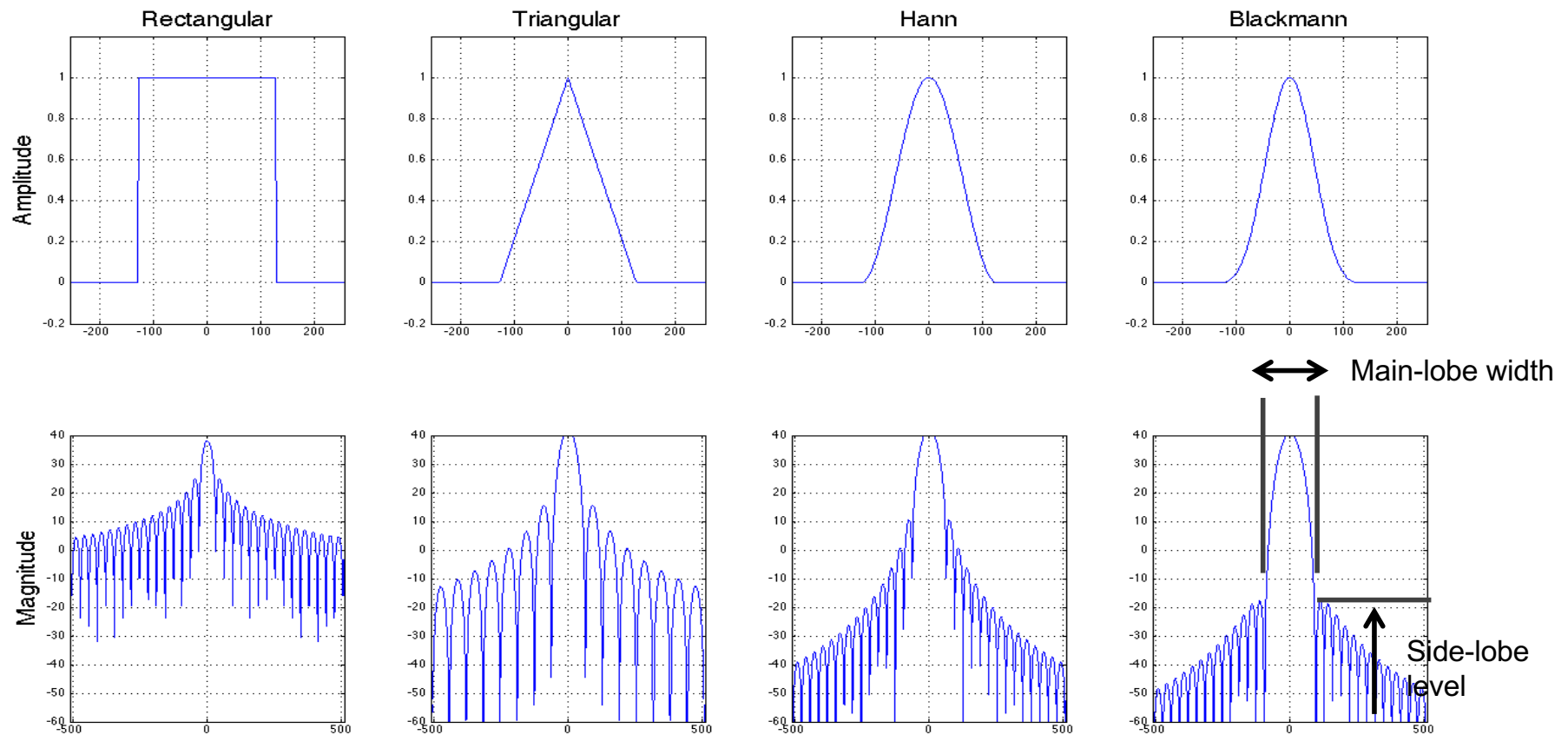
$$X(k, l) = \sum_{n=0}^{N-1} w(n)x(n + l \cdot h)e^{-j\left(\frac{2\pi kn}{N}\right)}$$

h : hop size
 $w(n)$: window
 N : FFT size

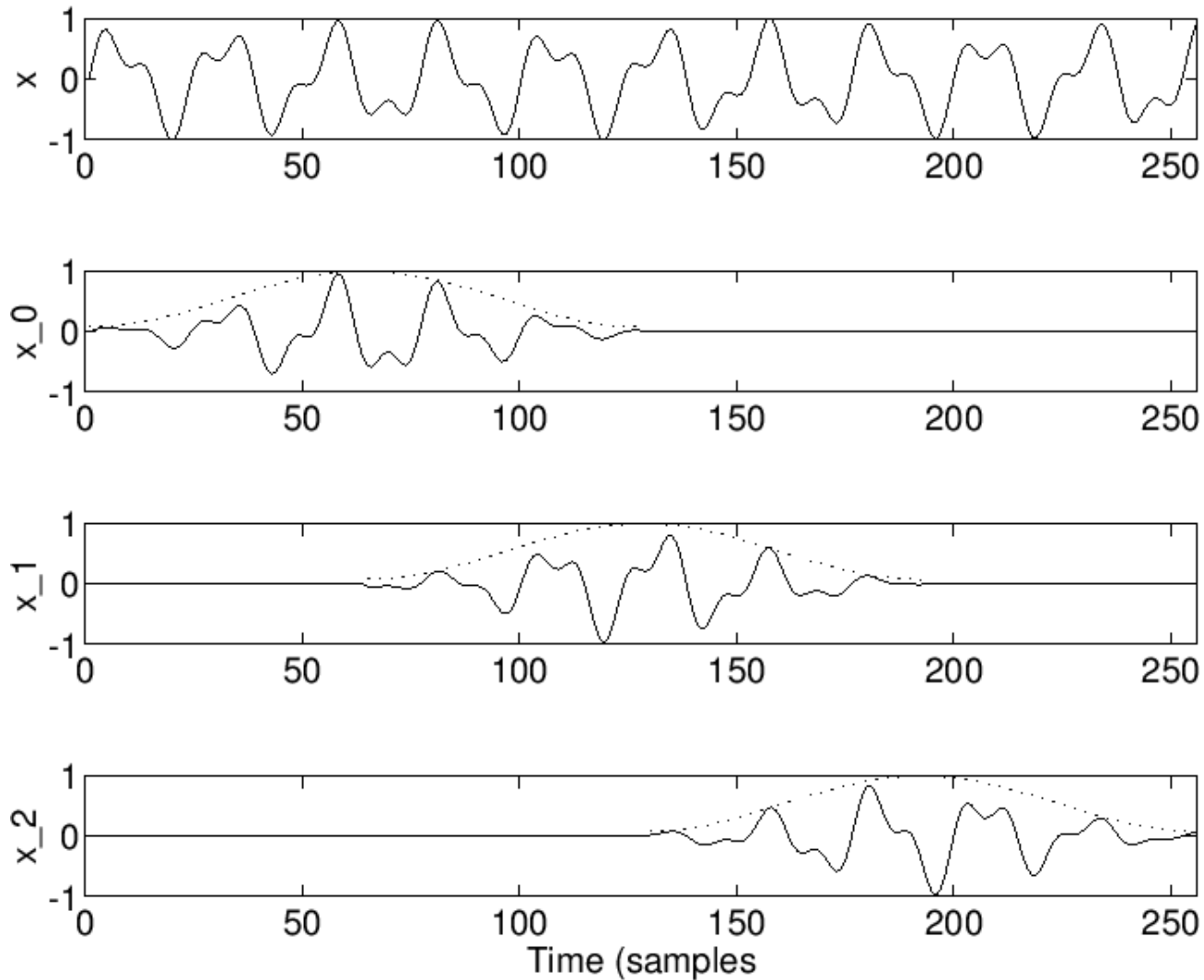
- This produces 2-D time-frequency representations
 - Parameters: window size, window type, FFT size, hop size
 - “Spectrogram” from the magnitude

Windowing

- Types of window functions
 - Trade-off between the width of main-lobe and the level of side-lobe

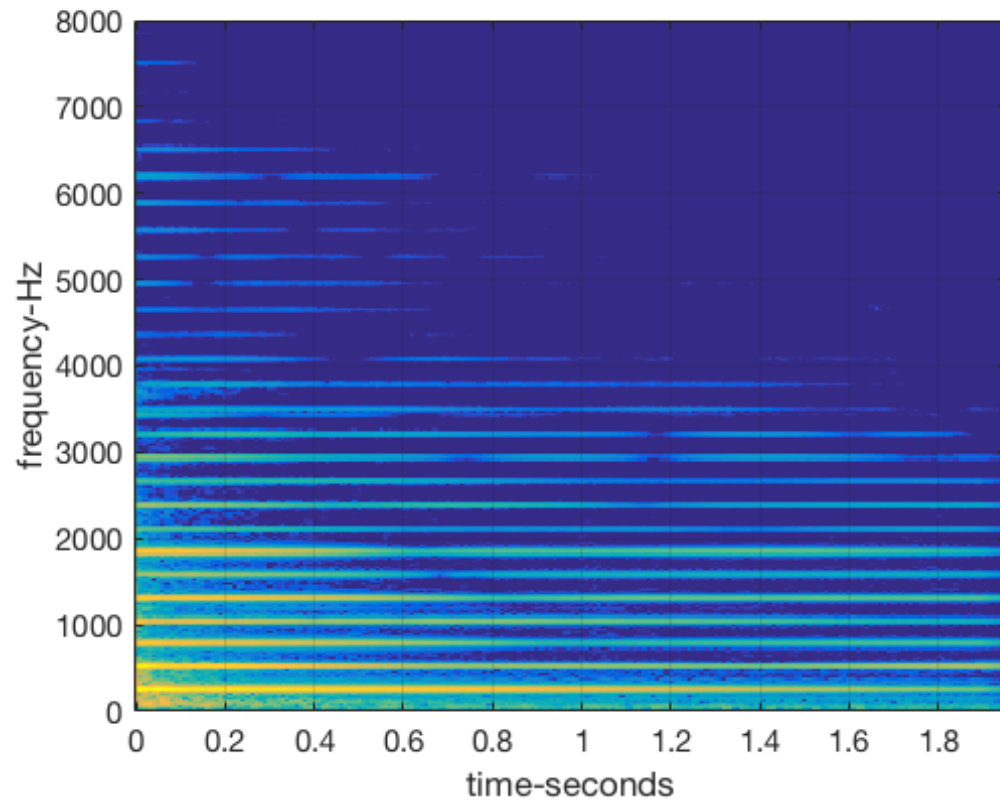


Short-Time Fourier Transform (STFT)

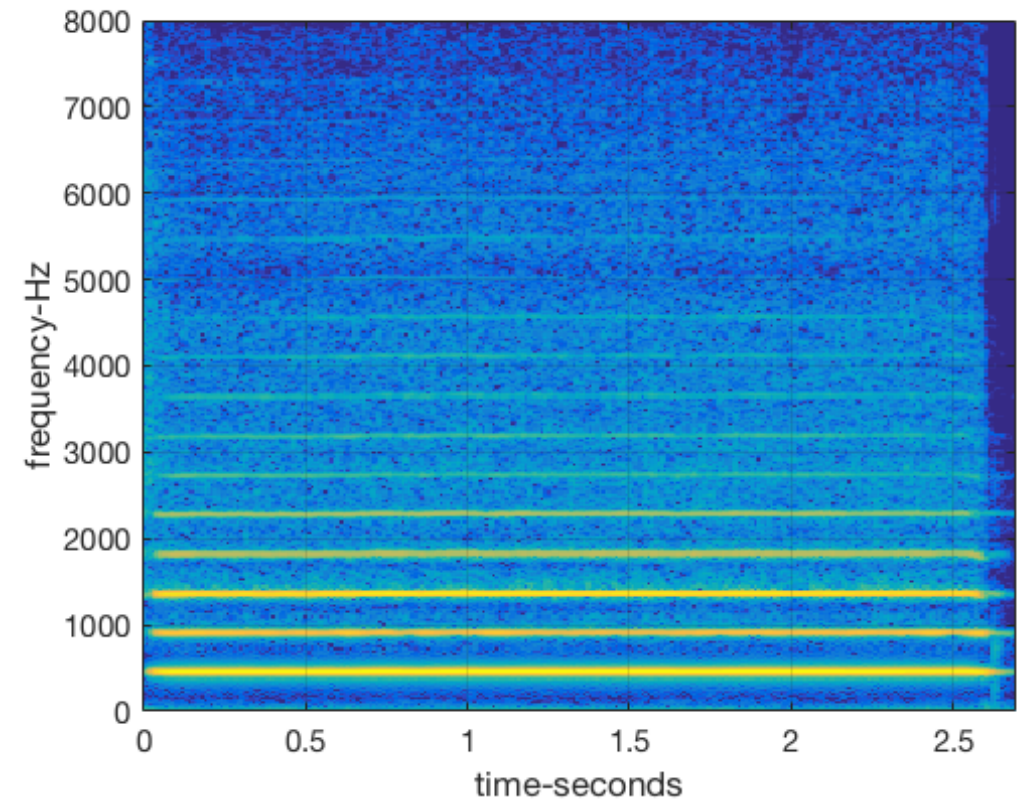


50% overlap

Example: Spectrogram

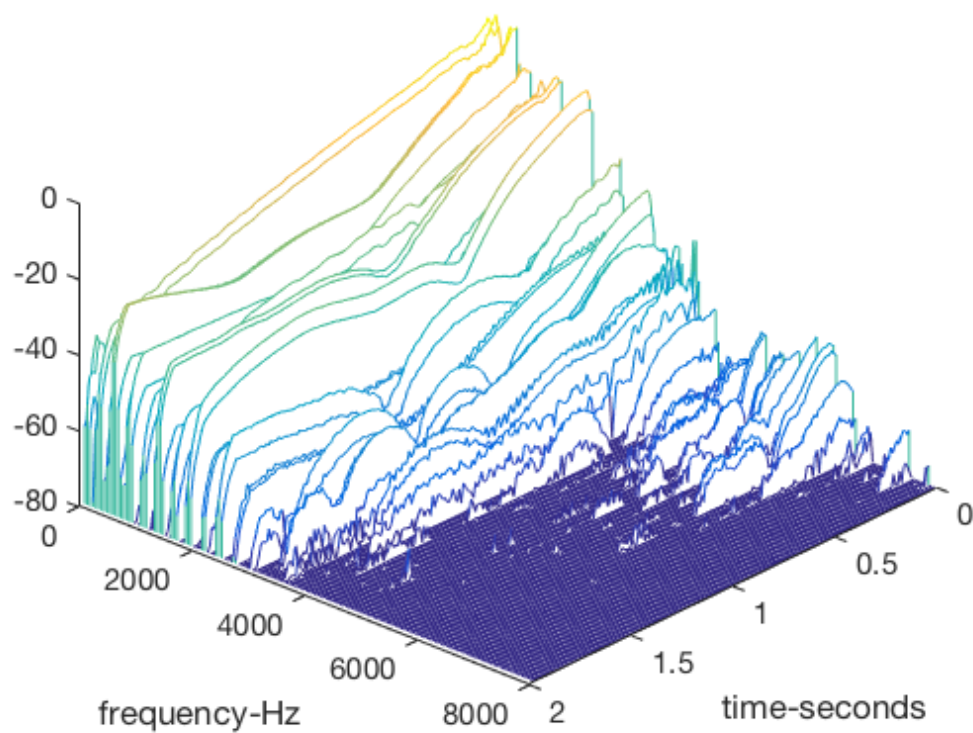


Piano C4 Note

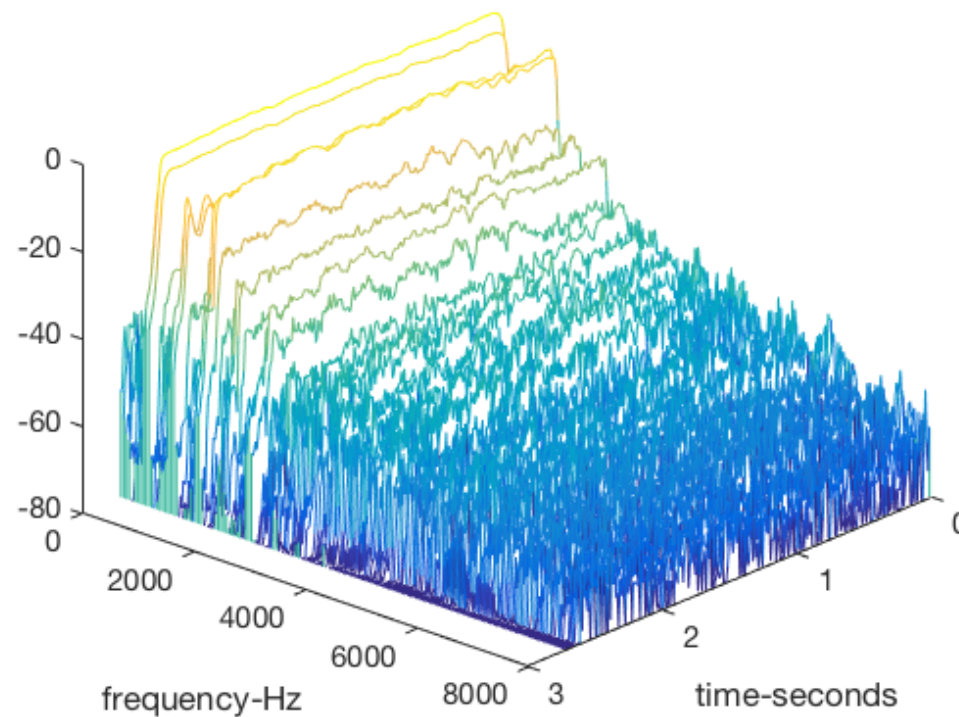


Flute A4 Note

Example: Spectrogram - 3D waterfall

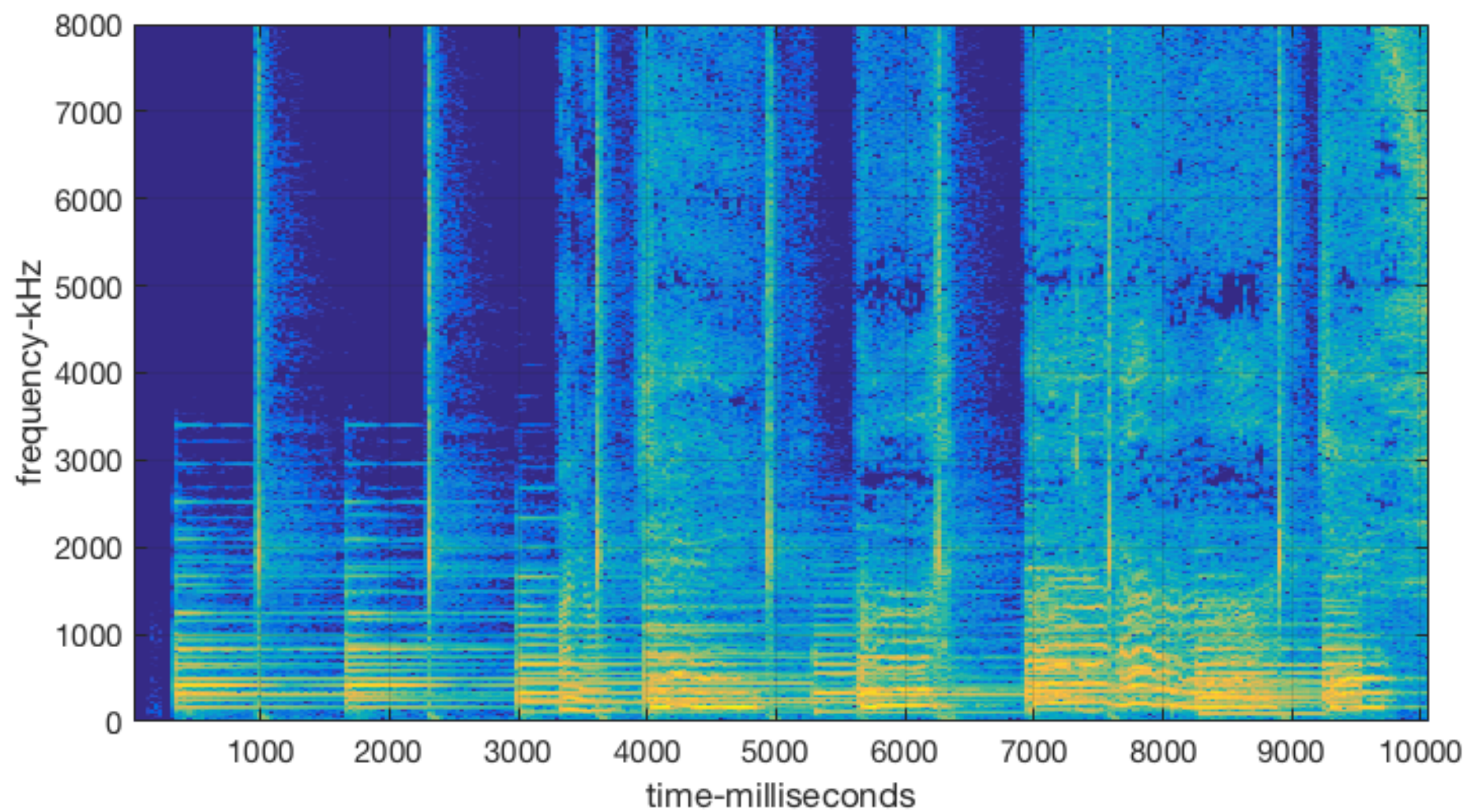


Piano C4 Note

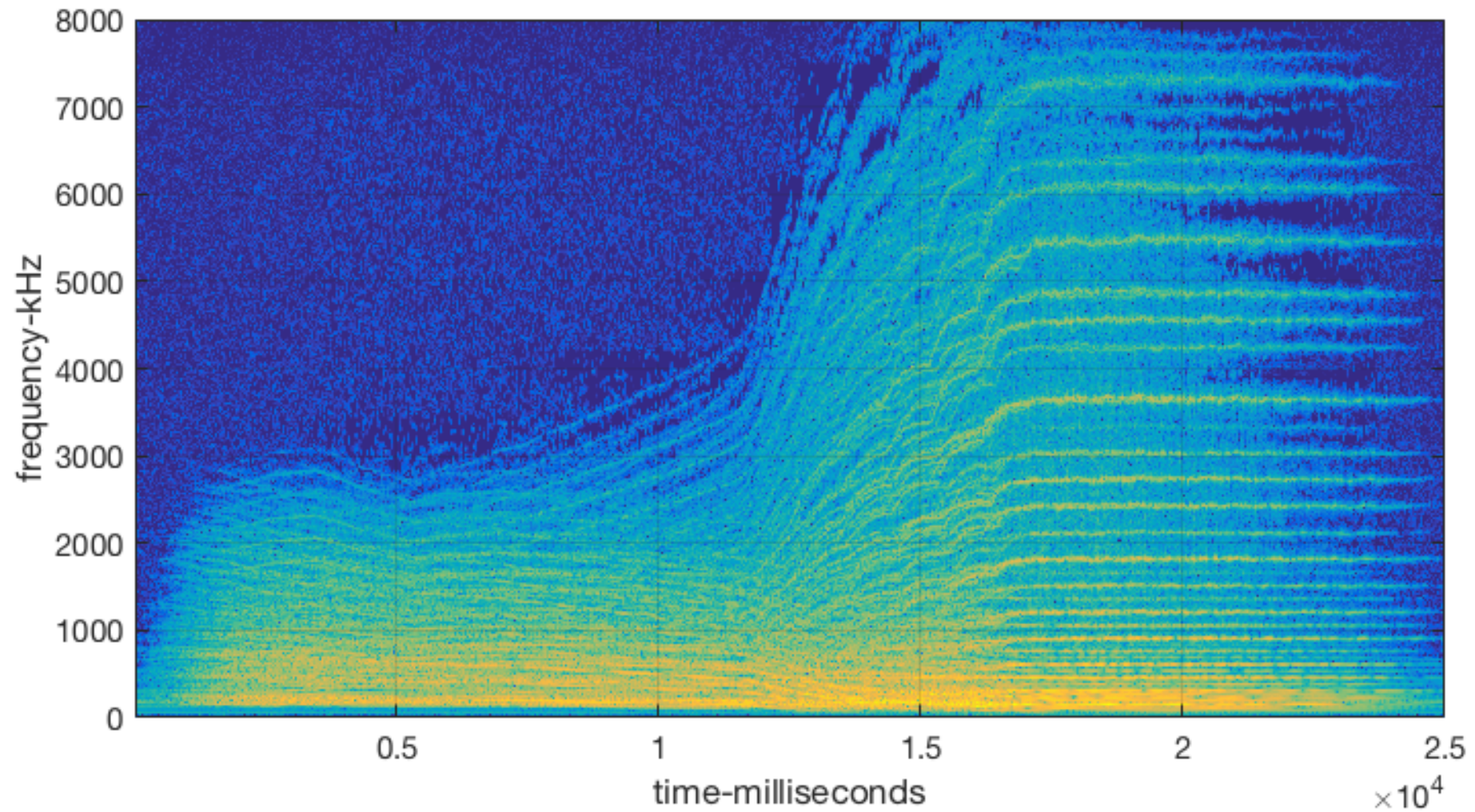


Flute A4 Note

Example: Pop Music

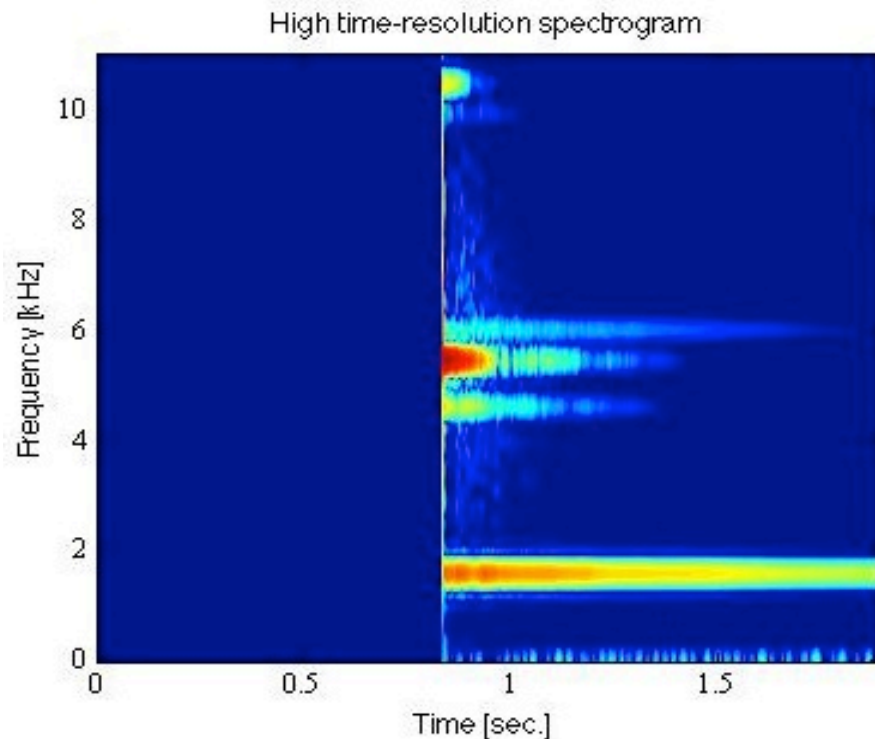


Example: Deep Note

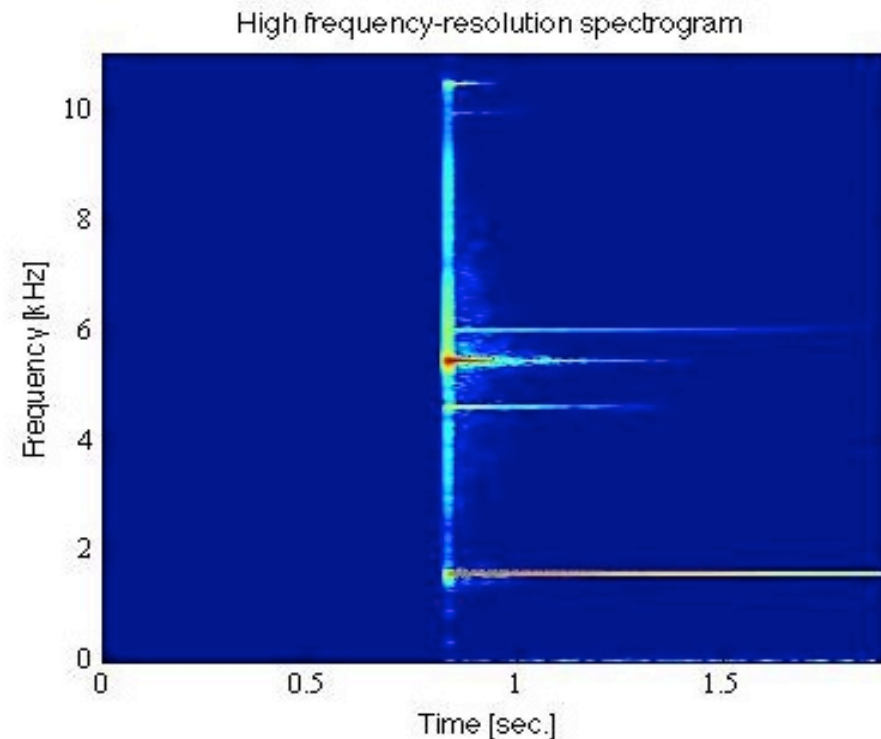


Time-Frequency Resolutions in STFT

- Trade-off between time and frequency resolution by window size



Short window
High time resolution
Low freq. resolution



Long window
High freq. resolution
Low time resolution

