Signal Processing for Interactive Systems
Lecture 4

Cumhur Erkut
cer@create.aau.dk

AALBORG UNIVERSITY
DENMARK

Agenda

A bit more on the DFT and Human Hearing

The Short-time Fourier Transform

Computing the STFT

Time-Frequency Resolution

Agenda

A bit more on the DFT and Human Hearing

The Short-time Fourier Transform

Computing the STFT

Time-Frequency Resolution

A bit more on the DFT

Motivation

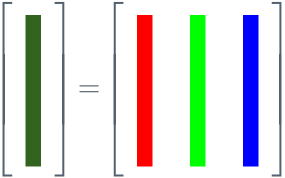
In about 20 minutes, you will know

- another way of interpreting the DFT and the iDFT (recap from last time)
- a basic model for human hearing
- what a filter bank is
- how MP3 works

A bit more on the DFT

A different perspective on the DFT/IDFT

- We wish to draw a dark green color using the RGB color model.
- The RGB color code for the dark green color is $(50, 100, 32)$.



$$\begin{bmatrix} 50 \\ 100 \\ 32 \end{bmatrix}$$

A bit more on the DFT

A different perspective on the DFT/IDFT

The inverse DFT describes how a time-domain signal $\boldsymbol{x}$ can be written as a weighted sum of sinusoids

$$\boldsymbol{x} = K^{-1}\boldsymbol{F}^H\boldsymbol{X} \quad (1)$$

- The columns of $K^{-1}\boldsymbol{F}^H$ form a basis/frame/dictionary for $\boldsymbol{x}$ and the DFT coefficients in $\boldsymbol{X}$ are the weights
- The weights pertaining to a signal $\boldsymbol{x}$ can be found using the DFT

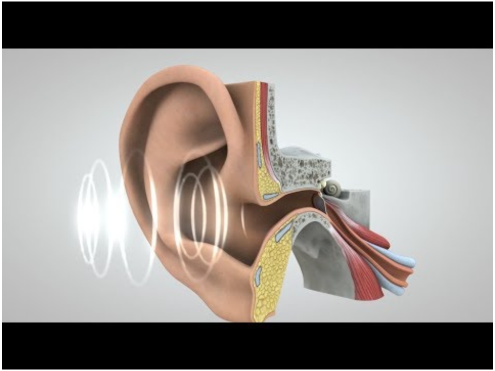
$$\boldsymbol{X} = \boldsymbol{F}\boldsymbol{x} \quad (2)$$

Human Hearing

Have you performed frequency analysis today?

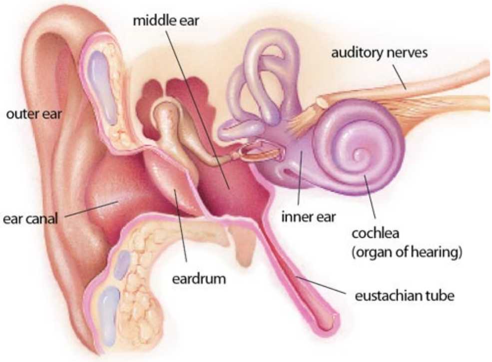
Human Hearing

Have you performed frequency analysis today?



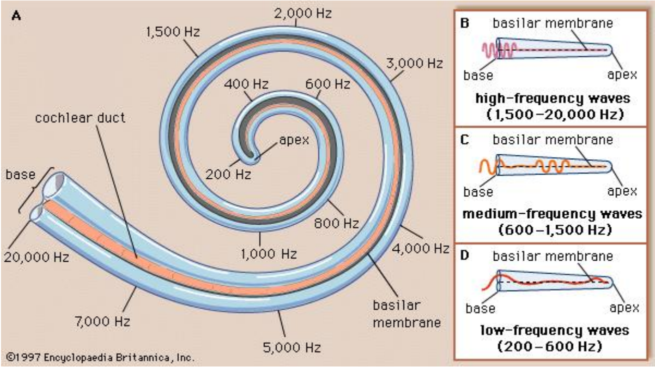
Human Hearing

Frequency analysis in the human ear



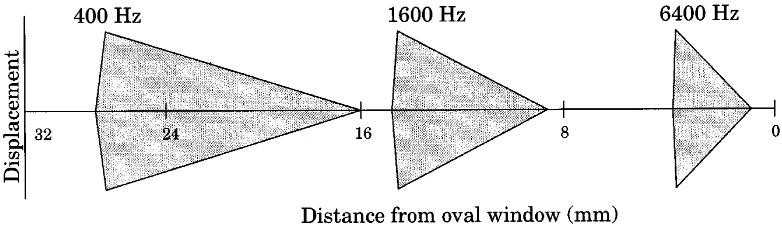
Human Hearing

Frequency analysis in the human ear



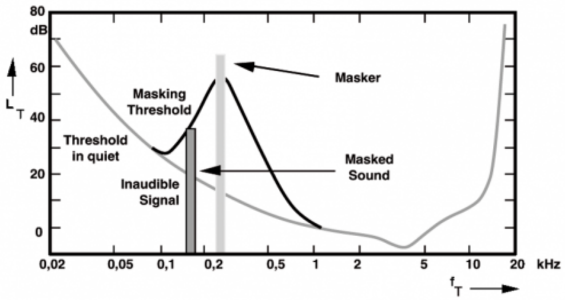
Human Hearing

Simultaneous masking



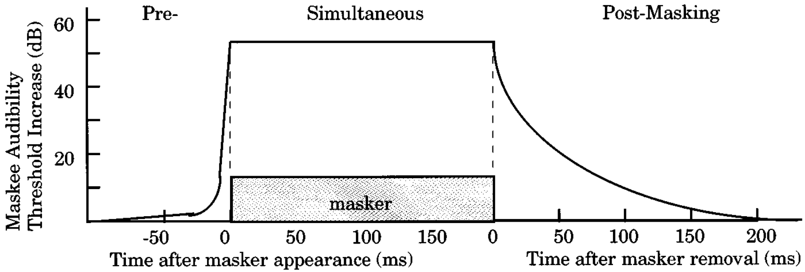
Human Hearing

Simultaneous masking



Human Hearing

Temporal masking



Human Hearing

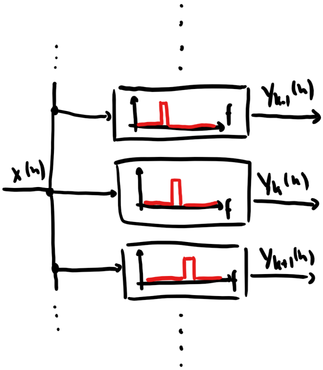
A simple model for human hearing

- Human hearing can be modelled as a 1/3 octave filter bank.
- The center frequency in Hz of the k'th filter is

$$f_c(k) = 2^{k/3} \cdot 1000 \, .$$

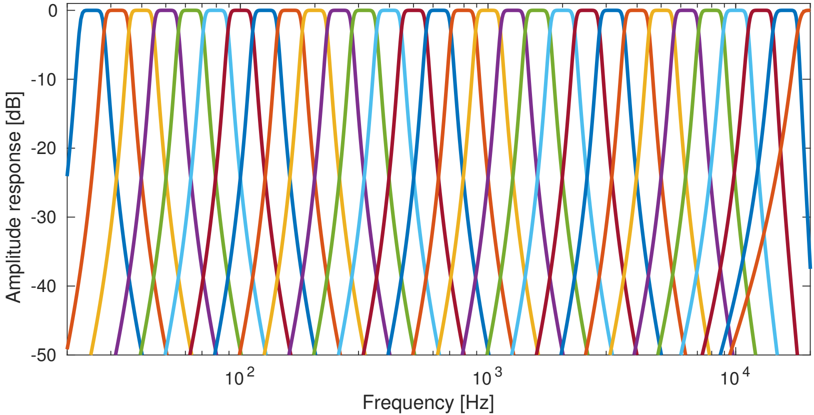
- The bandwidth in Hz of the k'th filter is

$$\mathrm{BW}(k) = f_c(k) \frac{2^{1/3} - 1}{2^{1/6}} \, .$$



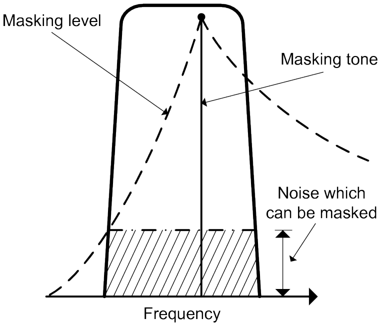
Human Hearing

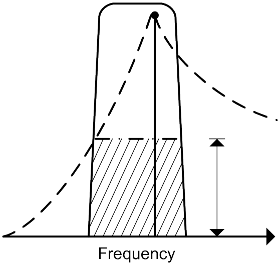
A simple model for human hearing



A bit more on the DFT

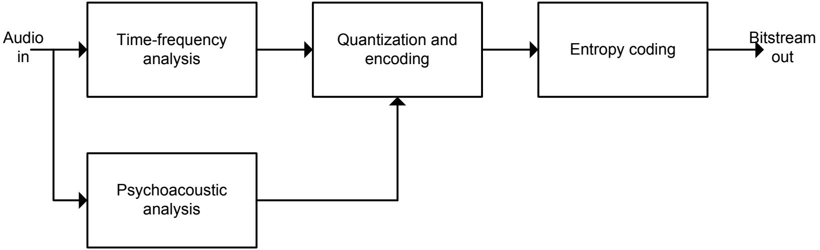
Perceptual Audio Coding (e.g., MP3)





Human Hearing

Perceptual Audio Coding (e.g., MP3)



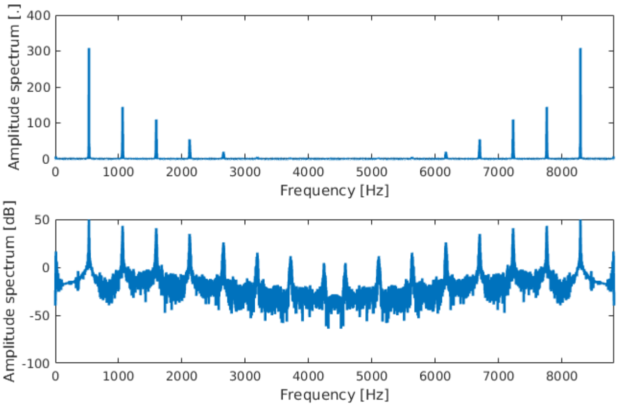
A bit more on the DFT

Motivation for today's lecture

- The human ear performs short-time Fourier analysis
- We want to do something similar on a computer

A bit more on the DFT

Motivation for today's lecture (🔊)



A bit more on the DFT

Motivation for today's lecture

- The trumpet signal is (approximately) stationary
- What if we want to analyse something non-stationary (🔊)?

A bit more on the DFT

Five minutes active break

Assume that a discrete-time signal is given by

$$x(n) = \begin{cases} \cos(\omega_0 n) & 0 \leq n \leq N/2 - 1 \\ \cos(2\omega_0 n) & N/2 \leq n \leq N - 1 \\ 0 & \text{otherwise} \end{cases} . \tag{3}$$

- Create a plot with n on the x-axis and ω on the y-axis. Sketch at which times the sinusoids in $x(n)$ are active.
- In MATLAB, compute the DFT of $x(n)$ for $n = 0, 1, \ldots, N - 1$ using a rectangular window.

Agenda

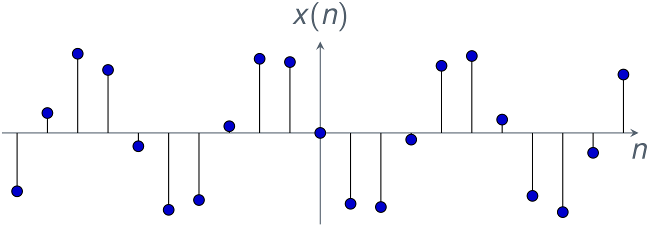
The Short-time Fourier Transform

Motivation

In about 20 minutes, you will know

- how we can analyse the frequency content of a non-stationary signal
- how such an analysis can be implemented using LSI systems
- what the spectrogram is
- what the chromagram is

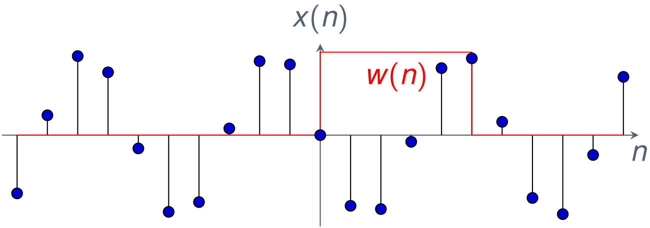
The Short-time Fourier Transform



The DTFT of a sequence $x(n)$ is

$$X(\omega) = \sum_{n=-\infty}^{\infty} x(n) e^{-j\omega n} \tag{4}$$

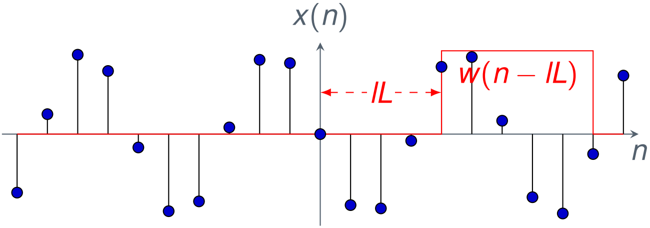
The Short-time Fourier Transform



The DTFT of a windowed sequence $x_N(n) = x(n)w(n)$ is

$$X_N(\omega) = \sum_{n=-\infty}^{\infty} x(n)w(n)e^{-j\omega n} \tag{5}$$

The Short-time Fourier Transform



The STFT of a shifted and windowed sequence is

$$X_N(\omega, l) = \sum_{n=-\infty}^{\infty} x(n) w(n - lL) e^{-j\omega n} \quad (6)$$

where l and L are the frame index and hop size, respectively.

The Short-time Fourier Transform

The STFT (theoretical version)

1. Shift the window by increasing l by one
2. Window the data

$$x_N(n, l) = x(n)w(n - lL) \tag{7}$$

3. Take the DTFT of $x_N(n, l)$

$$X_N(\omega, l) = \sum_{n=-\infty}^{\infty} x_N(n, l)\mathrm{e}^{-\mathrm{j}\omega n} \tag{8}$$

4. Repeat from 1.

In practice, we do something slightly different, but we will return to this later.

The Short-time Fourier Transform

The Spectrogram

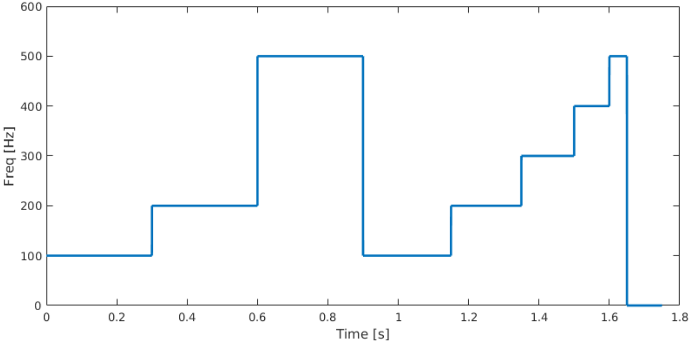
The spectrogram is the magnitude spectrum of the STFT, i.e.,

$$S_x(\omega, l) = |X_N(\omega, l)|^2 \tag{9}$$

Note that MATLAB's `spectrogram` function computes the STFT $X_N(\omega, l)$.

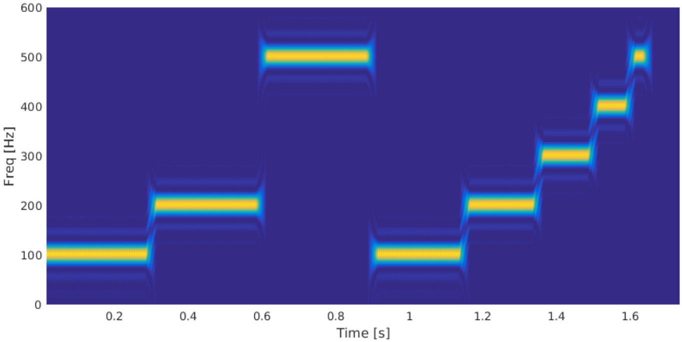
The Short-time Fourier Transform

Example (A time-varying sinusoid (🔊))



The Short-time Fourier Transform

Example (A time-varying sinusoid (🔊))



The Short-time Fourier Transform

The chromagram

- In music, pitch is a really important attribute
- Perceptually, the pitch is better represented as two features instead of just a frequency:

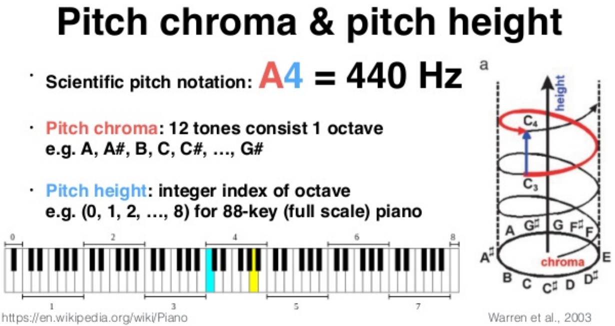
  Chroma The set of pitches (i.e., pitch class) which are a whole numbers of octaves apart. These pitches are perceived as having a similar color.

  Tone height An integer describing the octave number

- On an equal-temperated scale, twelwe different chroma values exist and are denoted by $\{C, C^{\#}, D, D^{\#}, E, F, F^{\#}, G, G^{\#}, A, A^{\#}, B\}$.

The Short-time Fourier Transform

The chromagram



The Short-time Fourier Transform

The chromagram

If we denote the chroma by $c \in [0, 1)$ and the tone height by $h \in \mathbb{Z}$, we can then write a pitch f in Hz as

$$f = 2^{c+h} = 2^c 2^h \,. \tag{10}$$

In the chromagram, we compute the spectrogram as a function of c instead of the traditional frequency f. Since

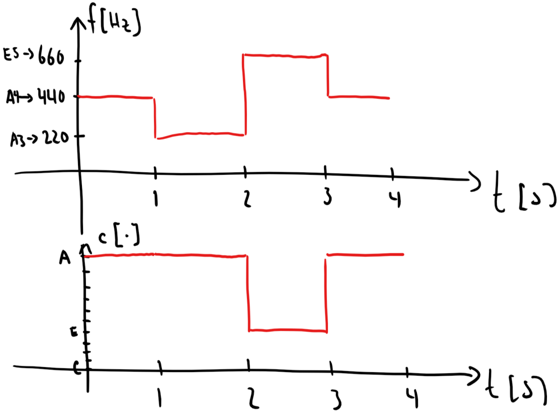
$$\omega = (2\pi/f_\mathrm{s})f = (2\pi/f_\mathrm{s})2^{c+h} \,, \tag{11}$$

we obtain that the chromagram is

$$K_x(c, l) = \sum_{h\in\mathbb{Z}} \left| X_N\left((2\pi/f_\mathrm{s})2^{c+h}, l\right)\right|^2 = \sum_{h\in\mathbb{Z}} S_x\left((2\pi/f_\mathrm{s})2^{c+h}, l\right) \,.$$

The Short-time Fourier Transform

Spectrogram vs. chromagram



The Short-time Fourier Transform

Summary

- We can use the DTFT and a moving window to analyse non-stationary signals
- This combination is called the short-time Fourier transform (STFT)
- The spectrogram is the squared amplitude of the STFT
- The STFT is also sometimes referred to as the
  - short-term Fourier transform
  - windowed Fourier transform
  - local Fourier transform
  - Gabor transform
- The chromagram is computed from the spectrogram

Agenda

Computing the STFT

Motivation

In about 20 minutes, you will know

- how you implement the STFT using the DFT
- how you implement the STFT using a filter bank

Computing the STFT

Recall the theoretical way of computing the STFT.

1. Shift the window by increasing l by one
2. Window the data

$$x_N(n, l) = x(n)w(n - lL) \tag{12}$$

3. Take the DTFT of $x_N(n, l)$

$$X_N(\omega, l) = \sum_{n=-\infty}^{\infty} x_N(n, l)\mathrm{e}^{-\mathrm{j}\omega n} \tag{13}$$

4. Repeat from 1.

Computing the STFT

So for the zeroth frame $l = 0$, we have to compute

$$X_N(\omega, 0) = \sum_{n=-\infty}^{\infty} x_N(n, 0)\mathrm{e}^{-\mathrm{j}\omega n} = \sum_{n=0}^{N-1} x(n)\mathrm{e}^{-\mathrm{j}\omega n} \qquad (14)$$

which we can compute using the DFT.

Computing the STFT

So for the zeroth frame $l = 0$, we have to compute

$$X_N(\omega, 0) = \sum_{n=-\infty}^{\infty} x_N(n, 0)\mathrm{e}^{-\mathrm{j}\omega n} = \sum_{n=0}^{N-1} x(n)\mathrm{e}^{-\mathrm{j}\omega n} \qquad (14)$$

which we can compute using the DFT. For $l = 1$, we obtain instead

$$X_N(\omega, 1) = \sum_{n=-\infty}^{\infty} x_N(n, 1)\mathrm{e}^{-\mathrm{j}\omega n} = \sum_{n=L}^{L+N-1} x(n)\mathrm{e}^{-\mathrm{j}\omega n} \qquad (15)$$

which we cannot compute directly using the DFT due to the start and stop indices.

Computing the STFT

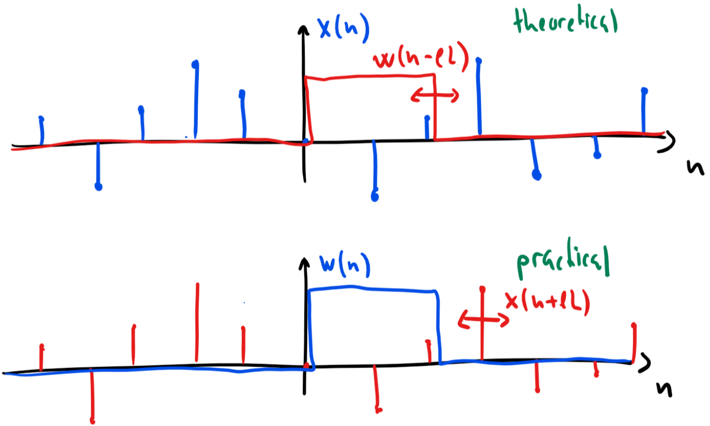
In general, we get for frame l that

$$X_N(\omega, l) = \sum_{n=-\infty}^{\infty} x_N(n, l)\mathrm{e}^{-\mathrm{j}\omega n} = \sum_{n=lL}^{lL+N-1} x(n)\mathrm{e}^{-\mathrm{j}\omega n} \qquad (16)$$

which again cannot be computed directly using the DFT.

Computing the STFT

We can either slide the window (theoretical) or slide the signal (practical).



Computing the STFT

The STFT (practical version)

If we slide the signal instead of the window, we obtain

$$\tilde{X}_N(\omega, l) = \sum_{n=-\infty}^{\infty} x(n + lL)w(n)\mathrm{e}^{-\mathrm{j}\omega n} = \sum_{n=0}^{N-1} x(n + lL)\mathrm{e}^{-\mathrm{j}\omega n} \quad (17)$$

which can be computed directly using the DFT for all segments. We, therefore, refer to it as the practical version.

Computing the STFT

The STFT (theoretical version)

$$X_N(\omega, l) = \sum_{n=-\infty}^{\infty} x(n)w(n - lL)e^{-j\omega n} \tag{18}$$

The STFT (practical version)

$$\tilde{X}_N(\omega, l) = \sum_{n=-\infty}^{\infty} x(n + lL)w(n)e^{-j\omega n} \tag{19}$$

Computing the STFT

The theoretical and practical versions are related by

$$X_N(\omega, l) = \tilde{X}_N(\omega, l)e^{-j\omega lL}\,. \tag{20}$$

Computing the STFT

The theoretical and practical versions are related by

$$X_N(\omega, l) = \tilde{X}_N(\omega, l)\mathrm{e}^{-\mathrm{j}\omega lL}\,. \tag{20}$$

Consequently,

- the spectrogram $S_x(\omega, l)$ is the same for both versions, i.e.,

$$S_x(\omega, l) = |X_N(\omega, l)|^2 = |\tilde{X}_N(\omega, l)\mathrm{e}^{-\mathrm{j}\omega lL}|^2 = |\tilde{X}_N(\omega, l)|^2\,. \tag{21}$$

- the interpretation of the phase response is different, but we often do not care about the phase.

The Short-time Fourier Transform

The STFT as filter bank

Let $m = lL$ with the hop size initially set to $L = 1$. Moreover, let

$$\omega_k = 2\pi \frac{k-1}{K} \tag{22}$$

$$w_k(n) = w(-n)e^{j\omega_k n} \tag{23}$$

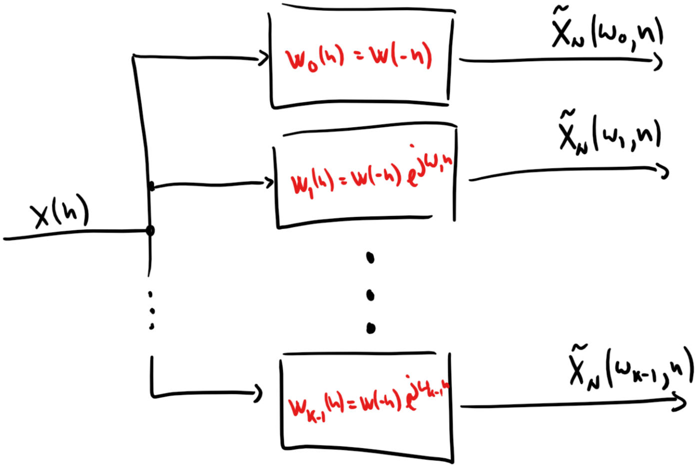
Then, the STFT has the following filter bank interpretation

$$\begin{aligned}\tilde{X}_N(\omega_k, l) &= \sum_{n=-\infty}^{\infty} x(n+m)w(n)e^{-j\omega_k n} = \sum_{n=-\infty}^{\infty} x(m-n)w(-n)e^{j\omega_k n} \\ &= \sum_{n=-\infty}^{\infty} w_k(n)x(m-n) = (x * w_k)(m) \,. \end{aligned} \tag{24}$$

Thus, $\tilde{X}_N(\omega_k, l)$ is $x(n)$ filtered through $w_k(n)$!

Computing the STFT

The STFT Filter Bank with hop size $L = 1$



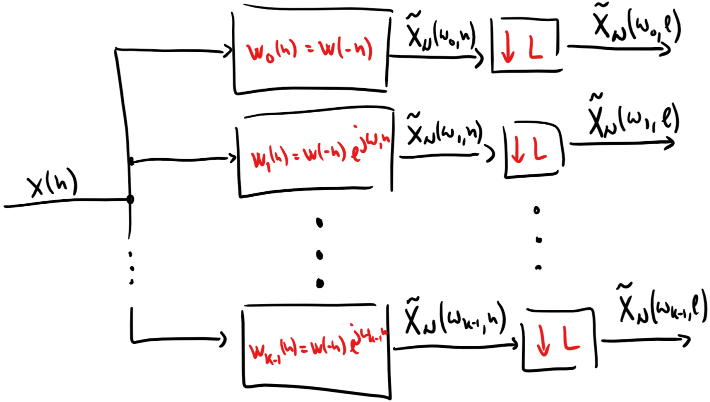
Computing the STFT

The STFT as filter bank

- When the hop size L is 1, then $l = n$. Thus, we get K new STFT coefficients for every new input sample.
- For a general hop size L, we can still implement the STFT as a filter bank, but we have to downsample all filter bank outputs by a factor of L.
- The downsampling can be integrated in the filtering operations to reduce the required number of computations. This is referred to as multirate filtering, but will not be covered in this course.

Computing the STFT

The STFT Filter Bank with hop size L



Computing the STFT

Summary

- In the practical version of the STFT, the signal is slided instead of the window. This can be written as

$$\tilde{X}_N(\omega, l) = \sum_{n=-\infty}^{\infty} x(n + lL)w(n)e^{-j\omega n} \,. \tag{25}$$

- The spectrogram is the same for the theoretical and practical versions of the STFT. The phase spectra, however, are different.
- The practical version of the STFT can be interpreted as a filter bank followed by downsampling.

Computing the STFT

Five minutes active break

Assume that we have the two windows

$$w_1(n) = \delta(n) \tag{26}$$

$$w_2(n) = 1 \,. \tag{27}$$

- Sketch the two windows in the time-domain
- Find the DTFT of the two windows by table look-up and sketch these DTFTs in the frequency-domain
- What can you say about the time- and frequency resolution of these two windows?

Agenda

A bit more on the DFT and Human Hearing

The Short-time Fourier Transform

Computing the STFT

Time-Frequency Resolution

Time-Frequency Resolution

Motivation

In about 20 minutes, you will know

- that you decrease the time-resolution if you increase the frequency resolution and vice versa
- some things to consider when choosing a window for the STFT

Time-Frequency Resolution

We want a high

- frequency resolution to do frequency analysis
- time resolution to handle non-stationary signals

Time-Frequency Resolution

We want a high

- frequency resolution to do frequency analysis
- time resolution to handle non-stationary signals

Heisenberg's Uncertainty Principle/Gabor Limit/Fourier Limit

One cannot have a high frequency and time resolution at the same time.

Time-Frequency Resolution

We want a high

- frequency resolution to do frequency analysis
- time resolution to handle non-stationary signals

Heisenberg's Uncertainty Principle/Gabor Limit/Fourier Limit

One cannot have a high frequency and time resolution at the same time.

Example

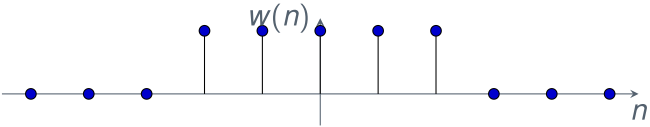
Sequence	DTFT (in $(-\pi, \pi]$)
$w(n) = \delta(n)$	$W(\omega) = 1$
$w(n) = 1$	$W(\omega) = 2\pi\delta(\omega)$

Time-Frequency Resolution

- The time-frequency resolution is determined by the window
- Many windows can be use
  - Rectangular
  - Blackman
  - Hamming
  - Hann
  - Bartlett
  - Truncated Gaussian

  and many more ...
- How do we measure the time-frequency resolution performance of a window?

Time-Frequency Resolution



Compute

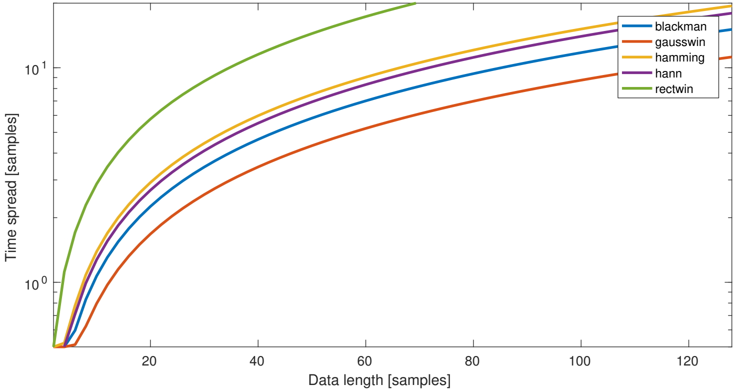
$$p(n) = \frac{|w(n)|^2}{\sum_{k=-\infty}^{\infty} |w(k)|^2} \tag{28}$$

$$\mu_n = \sum_{k=-\infty}^{\infty} k p(k) \tag{29}$$

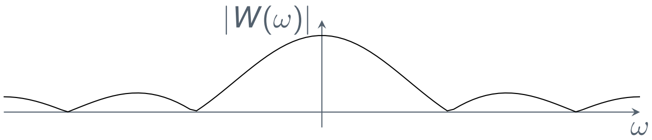
$$\sigma_n = \sqrt{\sum_{k=-\infty}^{\infty} (k - \mu_n)^2 p(k)} \tag{30}$$

Time-Frequency Resolution

Example: Time spread for different windows



Time-Frequency Resolution



Compute

$$Q(\omega) = \frac{|W(\omega)|^2}{\int_{-\pi}^{\pi} |W(\omega)|^2 d\omega} \tag{31}$$

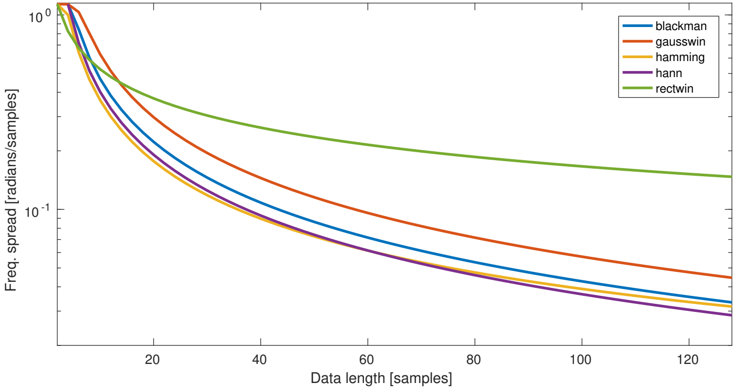
$$\mu_f = \int_{-\pi}^{\pi} \omega Q(\omega) d\omega \tag{32}$$

$$\sigma_f = \sqrt{\int_{-\pi}^{\pi} (\omega - \mu_\omega)^2 Q(\omega)} \tag{33}$$

where $W(\omega)$ is the DTFT of $w(n)$.

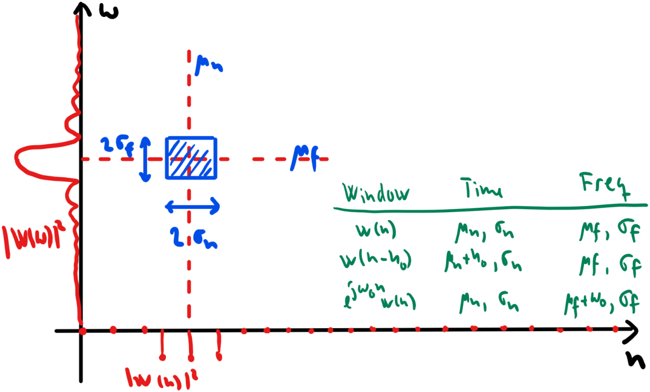
Time-Frequency Resolution

Example: Frequency spread for different windows



Time-Frequency Resolution

The Heisenberg Box

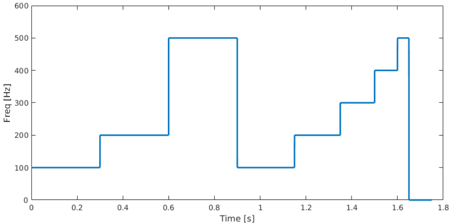


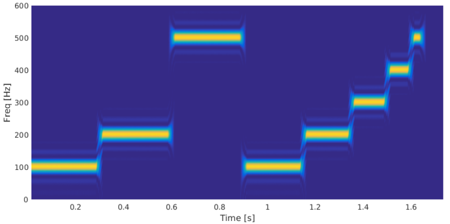
Window	Time	Freq
$w(n)$	μ_n, σ_n	μ_f, σ_f
$w(n-n_0)$	μ_n+n_0, σ_n	μ_f, σ_f
$e^{j\omega_0 n} w(n)$	μ_n, σ_n	$\mu_f+\omega_0, \sigma_f$

$$A = 4\sigma_f\sigma_t \geq 2 \tag{34}$$

Time-Frequency Resolution

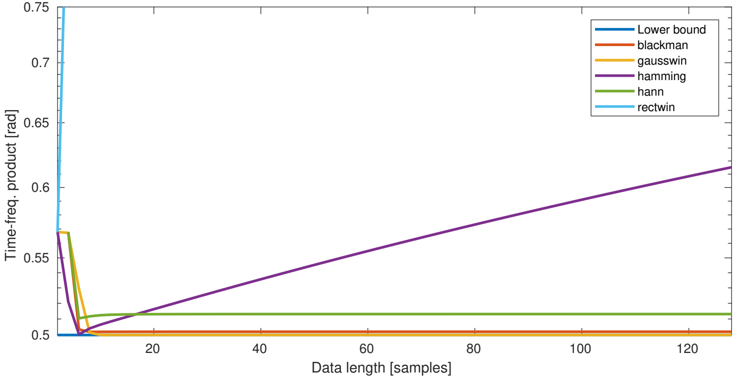
Example (A time-varying sinusoid (🔊))





Time-Frequency Resolution

Time-frequency product ($\sigma_f \sigma_t$) for different windows



Time-Frequency Resolution

Tips and tricks for selecting a window

- The sampled Gaussian window function

$$w(n) = e^{-\frac{n^2}{2\sigma^2}} \tag{35}$$

  produces a Heisenberg box with an area close to the bound.
- Unfortunately, the Gaussian window function is infinite in length and, therefore, not very practical
- Instead, a truncated Gaussian or various other Gaussian like windows like Hann, Hamming, or Kaiser windows can be used

Time-Frequency Resolution

Tips and tricks for selecting a window

- The window should be narrow enough so that the signal is approximately stationary
  - Speech: $\approx 20 - 30$ ms
  - Music: $\approx 25 - 50$ ms
- It is a good idea to use overlapping windows - at least 50 %
- MATLAB has a function called `spectrogram` which computes the STFT or spectrogram (depending on how you call it)
- MATLAB has a function called `gausswin` for computing a truncated Gaussian window

Time-Frequency Resolution

Summary

- You cannot get both a good time resolution and a good frequency resolution at the same time
- You can visualise the time-frequency trade-off and the STFT principle by moving a Heisenberg box around in the time-frequency domain
- The truncated Gaussian window is a good window to use with the STFT

Questions?

AALBORG UNIVERSITY
DENMARK