Workshop HfM Karlsruhe

Music Information Retrieval

Audio Features

Christof Weiß, Frank Zalkow, Meinard Müller
International Audio Laboratories Erlangen

christof.weiss@audiolabs-erlangen.de
frank.zalkow@audiolabs-erlangen.de
meinard.mueller@audiolabs-erlangen.de
Book: Fundamentals of Music Processing

Meinard Müller
Fundamentals of Music Processing
Audio, Analysis, Algorithms, Applications
483 p., 249 illus., hardcover
ISBN: 978-3-319-21944-8
Springer, 2015

Accompanying website:
www.music-processing.de
Book: Fundamentals of Music Processing

<table>
<thead>
<tr>
<th>Chapter</th>
<th>Music Processing Scenario</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Music Representations</td>
</tr>
<tr>
<td>2</td>
<td>Fourier Analysis of Signals</td>
</tr>
<tr>
<td>3</td>
<td>Music Synchronization</td>
</tr>
<tr>
<td>4</td>
<td>Music Structure Analysis</td>
</tr>
<tr>
<td>5</td>
<td>Chord Recognition</td>
</tr>
<tr>
<td>6</td>
<td>Tempo and Beat Tracking</td>
</tr>
<tr>
<td>7</td>
<td>Content-Based Audio Retrieval</td>
</tr>
<tr>
<td>8</td>
<td>Musically Informed Audio Decomposition</td>
</tr>
</tbody>
</table>

Meinard Müller
Fundamentals of Music Processing
Audio, Analysis, Algorithms, Applications
483 p., 249 illus., hardcover
ISBN: 978-3-319-21944-8
Springer, 2015

Accompanying website:
www.music-processing.de
Book: Fundamentals of Music Processing

Meinard Müller
Fundamentals of Music Processing
Audio, Analysis, Algorithms, Applications
483 p., 249 illus., hardcover
ISBN: 978-3-319-21944-8
Springer, 2015

Accompanying website:
www.music-processing.de
Chapter 2: Fourier Analysis of Signals

2.1 The Fourier Transform in a Nutshell
2.2 Signals and Signal Spaces
2.3 Fourier Transform
2.4 Discrete Fourier Transform (DFT)
2.5 Short-Time Fourier Transform (STFT)
2.6 Further Notes

Important technical terminology is covered in Chapter 2. In particular, we approach the Fourier transform—which is perhaps the most fundamental tool in signal processing—from various perspectives. For the reader who is more interested in the musical aspects of the book, Section 2.1 provides a summary of the most important facts on the Fourier transform. In particular, the notion of a spectrogram, which yields a time–frequency representation of an audio signal, is introduced. The remainder of the chapter treats the Fourier transform in greater mathematical depth and also includes the fast Fourier transform (FFT)—an algorithm of great beauty and high practical relevance.
Chapter 3: Music Synchronization

3.1 Audio Features
3.2 Dynamic Time Warping
3.3 Applications
3.4 Further Notes

As a first music processing task, we study in Chapter 3 the problem of music synchronization. The objective is to temporally align compatible representations of the same piece of music. Considering this scenario, we explain the need for musically informed audio features. In particular, we introduce the concept of chroma-based music features, which capture properties that are related to harmony and melody. Furthermore, we study an alignment technique known as dynamic time warping (DTW), a concept that is applicable for the analysis of general time series. For its efficient computation, we discuss an algorithm based on dynamic programming—a widely used method for solving a complex problem by breaking it down into a collection of simpler subproblems.
Fourier Transform

Idea: Decompose a given signal into a superposition of sinusoids (elementary signals).

$$f = s_1 + s_2 + s_3$$
Fourier Transform

Each sinusoid has a physical meaning and can be described by three parameters:

$$s(A, \omega, \varphi)(t) = A \cdot \sin(2\pi(\omega t - \varphi))$$

- $\omega = \text{frequency}$
- $A = \text{amplitude}$
- $\varphi = \text{phase}$

**Interpretation:**
The amplitude $A$ reflects the intensity at which the sinusoidal of frequency $\omega$ appears in $f$.
The phase $\varphi$ reflects how the sinusoidal has to be shifted to best correlate with $f$.

- $A_1 = 1$
- $\omega_1 = 1$
- $\varphi_1 = 0$
- $A_2 = 0.6$
- $\omega_2 = 3$
- $\varphi_2 = -0.2$
- $A_3 = 0.4$
- $\omega_3 = 7$
- $\varphi_3 = 0.4$
Each sinusoid has a physical meaning and can be described by three parameters:

\[ f = s_1 + s_2 + s_3 \]

- For sinusoid 1:
  \[ A_1 = 1, \quad \omega_1 = 1, \quad \varphi_1 = 0 \]

- For sinusoid 2:
  \[ A_2 = 0.6, \quad \omega_2 = 3, \quad \varphi_2 = -0.2 \]

- For sinusoid 3:
  \[ A_3 = 0.4, \quad \omega_3 = 7, \quad \varphi_3 = 0.4 \]
Each sinusoid has a physical meaning and can be described by three parameters:

$$f = s_1 + s_2 + s_3$$

- \( A_1 = 1 \)
- \( \omega_1 = 1 \)
- \( \varphi_1 = 0 \)
- \( A_2 = 0.6 \)
- \( \omega_2 = 3 \)
- \( \varphi_2 = -0.2 \)
- \( A_3 = 0.4 \)
- \( \omega_3 = 7 \)
- \( \varphi_3 = 0.4 \)
Fourier Transform

Example: Superposition of two sinusoids
Fourier Transform

Example: C4 played by piano

\[ f \]

\[ |\hat{f}| \]
Fourier Transform

Example: C4 played by trumpet
Fourier Transform

Example: C4 played by violine
Fourier Transform

Example: C4 played by flute
Fourier Transform

Example: Speech “Bonn”
Fourier Transform

Example: Speech “Zürich”
Fourier Transform

Example: C-major scale (piano)
Fourier Transform

Example: Chirp signal
Fourier Transform

Example: Piano tone (C4, 261.6 Hz)
Fourier Transform

Example: Piano tone (C4, 261.6 Hz)

Analysis using sinusoid with **262 Hz**
→ high correlation
→ large Fourier coefficient
Fourier Transform

Example: Piano tone (C4, 261.6 Hz)

Analysis using sinusoid with 400 Hz
→ low correlation
→ small Fourier coefficient
Fourier Transform

Example: Piano tone (C4, 261.6 Hz)

Analysis using sinusoid with 523 Hz
→ high correlation
→ large Fourier coefficient
Fourier Transform

Role of phase
Fourier Transform

Role of phase

Analysis with sinusoid having frequency 262 Hz and phase $\varphi = 0.05$
Fourier Transform

Role of phase

Analysis with sinusoid having frequency 262 Hz and phase $\phi = 0.24$
Fourier Transform

Role of phase

Analysis with sinusoid having frequency 262 Hz and phase $\phi = 0.45$
Fourier Transform

Role of phase

Analysis with sinusoid having frequency 262 Hz and phase $\phi = 0.6$
Fourier Transform

Each sinusoid has a physical meaning and can be described by three parameters:

\[ s(A, \omega, \varphi)(t) = A \cdot \sin(2\pi(\omega t - \varphi)) \]

- \( \omega \) = frequency
- \( A \) = amplitude
- \( \varphi \) = phase

Complex formulation of sinusoids:

\[ e(c, \omega)(t) = c \cdot \exp(2\pi i \omega t) = c \cdot (\cos(2\pi \omega t) + i \cdot \sin(2\pi \omega t)) \]

- \( \omega \) = frequency
- \( A = |c| \) = amplitude
- \( \varphi = \text{phase} = \arg(c) \)

Polar coordinates:

\[ c = |c| \cdot \exp(2\pi i \varphi) \]
Fourier Transform

Signal \( f : \mathbb{R} \rightarrow \mathbb{R} \)

Fourier representation \( f(t) = \int_{\omega \in \mathbb{R}} c_\omega \exp(2\pi i \omega t) d\omega \)

Fourier transform \( c_\omega = \hat{f}(\omega) = \int_{t \in \mathbb{R}} f(t) \exp(-2\pi i \omega t) dt \)
Fourier Transform

Signal \( f : \mathbb{R} \rightarrow \mathbb{R} \)

Fourier representation \( f(t) = \int_{\omega \in \mathbb{R}} c_{\omega} \exp(2\pi i \omega t) d\omega \)

Fourier transform \( c_{\omega} = \hat{f}(\omega) = \int_{t \in \mathbb{R}} f(t) \exp(-2\pi i \omega t) dt \)

- Tells which frequencies occur, but does not tell when the frequencies occur.
- Frequency information is averaged over the entire time interval.
- Time information is hidden in the phase.
Fourier Transform

\[ f \]

Time (seconds)

\[ |\hat{f}| \]

Frequency (Hz)
Short Time Fourier Transform

Idea (Dennis Gabor, 1946):

- Consider only a **small section** of the signal for the spectral analysis
  
  → recovery of time information

- Short Time Fourier Transform (STFT)

- Section is determined by pointwise multiplication of the signal with a localizing **window function**
Short Time Fourier Transform

![Graph showing the Short Time Fourier Transform with time in seconds on the x-axis and frequency in Hz on the y-axis. The top graph displays a time-domain signal, while the bottom graph shows the corresponding frequency-domain representation.](image-url)
Short Time Fourier Transform

The top graph represents the signal in the time domain, showing a rectangular pulse that lasts for four seconds. The bottom graph shows the frequency domain representation of the same signal, with peaks indicating the frequencies present in the signal.
Short Time Fourier Transform

![Graph showing time-domain signal and its frequency spectrum.](image-url)
Short Time Fourier Transform

The top graph shows a time-domain signal (red) with a periodic component (blue) between 0 and 10 seconds. The bottom graph represents the frequency spectrum (blue) with a peak at approximately 5 Hz, indicating the frequency content of the signal.
Short Time Fourier Transform

![Graph showing time-domain and frequency-domain representations of a signal. The top graph displays a waveform over time (seconds), while the bottom graph shows the frequency spectrum (Hz) with a peak at around 1 Hz. The waves indicate a periodic signal.]
Short Time Fourier Transform

![Graphs of Short Time Fourier Transform]
Short Time Fourier Transform

Time (seconds)

Frequency (Hz)
Short Time Fourier Transform

![Time-domain signal](image1.png)

![Frequency-domain analysis](image2.png)
Short Time Fourier Transform

Window functions

Rectangular window

Triangular window

Hann window
Short Time Fourier Transform

Window functions

→ Trade off between smoothing and “ringing”
Short Time Fourier Transform

Definition

- Signal \( f: \mathbb{R} \to \mathbb{R} \)

- Window function \( g: \mathbb{R} \to \mathbb{R} \) \( (g \in L^2(\mathbb{R}), \|g\|_2 \neq 0) \)

- STFT \( \tilde{f}_g(t, \omega) = \int_{u \in \mathbb{R}} f(u)g(u-t)\exp(-2\pi i \omega u)du = \langle f|g_{t,\omega}\rangle \)

with \( g_{t,\omega}(u) = \exp(2\pi i \omega(u-t))g(u-t) \) for \( u \in \mathbb{R} \)
Short Time Fourier Transform

Intuition:

- $g_{t,\omega}$ is “musical note” of frequency $\omega$ centered at time $t$
- Inner product $\langle f | g_{t,\omega} \rangle$ measures the correlation between the musical note $g_{t,\omega}$ and the signal $f$
Short Time Fourier Transform

Discrete STFT

\[ \mathcal{X}(m, k) := \sum_{n=0}^{N-1} x(n + mH)w(n)\exp(-2\pi ikn/N) \]

- \( x : \mathbb{Z} \to \mathbb{R} \) DT-signal
- \( w : [0 : N - 1] \to \mathbb{R} \) Window function of length \( N \in \mathbb{N} \)
- \( H \in \mathbb{N} \) Hop size
- \( K = N/2 \) Index corresponding to Nyquist frequency
- \( \mathcal{X}(m, k) \) Fourier coefficient for frequency index \( k \in [0 : K] \) and time frame \( m \in \mathbb{Z} \)
Short Time Fourier Transform

Discrete STFT

\[ \mathcal{X}(m, k) := \sum_{n=0}^{N-1} x(n + mH)w(n)\exp(-2\pi i kn/N) \]

Physical time position associated with \( \mathcal{X}(m, k) \):

\[ T_{\text{coef}}(m) := \frac{m \cdot H}{F_s} \quad \text{(seconds)} \]

\( H \) = Hop size
\( F_s \) = Sampling rate

Physical frequency associated with \( \mathcal{X}(m, k) \):

\[ F_{\text{coef}}(k) := \frac{k \cdot F_s}{N} \quad \text{(Hertz)} \]
Short Time Fourier Transform

Discrete STFT

Parameters
\[ N = 64 \]
\[ H = 8 \]
\[ F_s = 32 \text{ Hz} \]
Time-Frequency Representation

Spectrogram

\[ f(t) \]

\[ \omega \]

Frequency (Hz)

Time (seconds)

\[ |\langle f | g_{\omega}, t \rangle| \]
Time-Frequency Representation

Spectrogram
Time-Frequency Representation

Chirp signal and STFT with Hann window of length 50 ms
Time-Frequency Representation

Chirp signal and STFT with box window of length 50 ms
Time-Frequency Representation

Time-Frequency Localization

- Size of window constitutes a trade-off between time resolution and frequency resolution:
  - Large window: poor time resolution, good frequency resolution
  - Small window: good time resolution, poor frequency resolution

- Heisenberg Uncertainty Principle: there is no window function that localizes in time and frequency with arbitrary precision.
Time-Frequency Representation

Signal and STFT with Hann window of length 20 ms
Time-Frequency Representation

Signal and STFT with Hann window of length 100 ms
Audio Features

Example: C-major scale (piano)

Spectrogram
Audio Features

Example: Chromatic scale

![Spectrogram]

Frequency (Hz) vs Time (seconds) with Intensity (dB)
Audio Features

Example: Chromatic scale

Spectrogram
Audio Features

Model assumption: Equal-tempered scale

- MIDI pitches: $p \in [1 : 128]$
- Piano notes: $p = 21$ (A0) to $p = 108$ (C8)
- Concert pitch: $p = 69$ (A4) $\cong 440$ Hz
- Center frequency: $F_{\text{pitch}}(p) = 2^{(p-69)/12} \cdot 440$ Hz

→ Logarithmic frequency distribution
Octave: doubling of frequency
Audio Features

Idea: Binning of Fourier coefficients

Divide up the frequency axis into logarithmically spaced “pitch regions” and combine spectral coefficients of each region to a single pitch coefficient.
Audio Features

Time-frequency representation

Windowing in the time domain

Windowing in the frequency domain
Log-Frequency Spectrogram

Pooling procedure for discrete STFT

Parameters

\[ N = 4096 \]
\[ H = 2048 \]
\[ F_s = 44100 \text{ Hz} \]

Frames

- \( F_{\text{coef}}(43) = 463.0 \)
- \( F_{\text{coef}}(42) = 452.2 \)
- \( F_{\text{coef}}(41) = 441.4 \)
- \( F_{\text{coef}}(40) = 430.7 \)
- \( F_{\text{coef}}(39) = 419.9 \)
- \( F_{\text{coef}}(38) = 409.1 \)
- \( F_{\text{coef}}(37) = 398.4 \)

Frames

- \( F_{\text{pitch}}(69.5) = 452.9 \)
- \( F_{\text{pitch}}(68.5) = 427.5 \)
- \( F_{\text{pitch}}(67.5) = 403.5 \)

Frames
Audio Features

Example: Chromatic scale

Spectrogram
Audio Features

Example: Chromatic scale

Spectrogram
Audio Features

Example: Chromatic scale

Log-frequency spectrogram

C8: 4186 Hz
C7: 2093 Hz
C6: 1046 Hz
C5: 523 Hz
C4: 262 Hz
C3: 131 Hz
### Audio Features

Frequency ranges for pitch-based log-frequency spectrogram

<table>
<thead>
<tr>
<th>Note</th>
<th>MIDI pitch</th>
<th>Center [Hz] frequency $F_{\text{pitch}}(p)$</th>
<th>Left [Hz] boundary $F_{\text{pitch}}(p - 0.5)$</th>
<th>Right [Hz] boundary $F_{\text{pitch}}(p + 0.5)$</th>
<th>Width [Hz]</th>
</tr>
</thead>
<tbody>
<tr>
<td>A3</td>
<td>57</td>
<td>220.0</td>
<td>213.7</td>
<td>226.4</td>
<td>12.7</td>
</tr>
<tr>
<td>A#3</td>
<td>58</td>
<td>233.1</td>
<td>226.4</td>
<td>239.9</td>
<td>13.5</td>
</tr>
<tr>
<td>B3</td>
<td>59</td>
<td>246.9</td>
<td>239.9</td>
<td>254.2</td>
<td>14.3</td>
</tr>
<tr>
<td>C4</td>
<td>60</td>
<td>261.6</td>
<td>254.2</td>
<td>269.3</td>
<td>15.1</td>
</tr>
<tr>
<td>C#4</td>
<td>61</td>
<td>277.2</td>
<td>269.3</td>
<td>285.3</td>
<td>16.0</td>
</tr>
<tr>
<td>D4</td>
<td>62</td>
<td>293.7</td>
<td>285.3</td>
<td>302.3</td>
<td>17.0</td>
</tr>
<tr>
<td>D#4</td>
<td>63</td>
<td>311.1</td>
<td>302.3</td>
<td>320.2</td>
<td>18.0</td>
</tr>
<tr>
<td>E4</td>
<td>64</td>
<td>329.6</td>
<td>320.2</td>
<td>339.3</td>
<td>19.0</td>
</tr>
<tr>
<td>F4</td>
<td>65</td>
<td>349.2</td>
<td>339.3</td>
<td>359.5</td>
<td>20.2</td>
</tr>
<tr>
<td>F#4</td>
<td>66</td>
<td>370.0</td>
<td>359.5</td>
<td>380.8</td>
<td>21.4</td>
</tr>
<tr>
<td>G4</td>
<td>67</td>
<td>392.0</td>
<td>380.8</td>
<td>403.5</td>
<td>22.6</td>
</tr>
<tr>
<td>G#4</td>
<td>68</td>
<td>415.3</td>
<td>403.5</td>
<td>427.5</td>
<td>24.0</td>
</tr>
<tr>
<td>A4</td>
<td>69</td>
<td>440.0</td>
<td>427.5</td>
<td>452.9</td>
<td>25.4</td>
</tr>
</tbody>
</table>
Audio Features

Chroma features

Chromatic circle

Shepard’s helix of pitch

Tone height
Audio Features

Chroma features

- Human perception of pitch is periodic in the sense that two pitches are perceived as similar in color if they differ by an octave (same pitch class).
- Separation of pitch into two components: tone height (octave number) and chroma / pitch class.
- Chroma: 12 pitch classes of the equal-tempered scale. For example:
  Chroma C ≜ \{\ldots, C_0, C_1, C_2, C_3, \ldots\}
- Computation: pitch features → chroma features
  Add up all pitches belonging to the same pitch class
- Result: 12-dimensional chroma vector.
Audio Features

Chroma features
Audio Features

Chroma features

C2  C3  C4

Chroma  C
Audio Features

Chroma features

C#2  C#3  C#4
Audio Features

Chroma features

Chroma D
Audio Features

Example: Chromatic scale

Log-frequency spectrogram
Audio Features

Example: Chromatic scale

Log-frequency spectrogram

Chroma C
Audio Features

Example: Chromatic scale

Log-frequency spectrogram

Chroma C#
Audio Features

Example: Chromatic scale

Chromagram

Time (seconds)
Audio Features

Chroma features

![Musical notation and time-frequency representation of audio features](image-url)
Audio Features

Chroma features
Audio Features

Chroma features
Audio Features

Chroma features

- Sequence of chroma vectors correlates to the harmonic progression

- Normalization $x \rightarrow x/\|x\|$ makes features invariant to changes in dynamics

- Further denoising and smoothing

- Taking logarithm before adding up pitch coefficients accounts for logarithmic sensation of intensity
Audio Features

Logarithmic compression

For a positive constant $\gamma \in \mathbb{R}_{>0}$, the logarithmic compression

$$\Gamma_\gamma : \mathbb{R}_{>0} \rightarrow \mathbb{R}_{>0}$$

is defined by

$$\Gamma_\gamma(v) := \log(1 + \gamma \cdot v)$$

A value $v \in \mathbb{R}_{>0}$ is replaced by a compressed value $\Gamma_\gamma(v)$.
Audio Features

Logarithmic compression

For a positive constant $\gamma \in \mathbb{R}_{>0}$ the logarithmic compression

$$\Gamma_{\gamma} : \mathbb{R}_{>0} \rightarrow \mathbb{R}_{>0}$$

is defined by

$$\Gamma_{\gamma}(v) := \log(1 + \gamma \cdot v)$$

A value $v \in \mathbb{R}_{>0}$ is replaced by a compressed value $\Gamma_{\gamma}(v)$.

The higher $\gamma \in \mathbb{R}_{>0}$ the stronger the compression.
Audio Features

Logarithmic compression

A value \( v \in \mathbb{R}_{>0} \) is replaced by a compressed value \( \Gamma_{\gamma}(v) \)

The higher \( \gamma \in \mathbb{R}_{>0} \) the stronger the compression
Audio Features

Normalization

Replace a vector
by the normalized vector

\[ x / \|x\| \]

using a suitable norm \( \| \cdot \| \)

Example:
Chroma vector \( x \in \mathbb{R}^{12} \)
Euclidean norm

\[
\|x\| := \left( \sum_{i=0}^{11} |x(i)|^2 \right)^{1/2}
\]
Audio Features

Normalization

Replace a vector by the normalized vector
\[ \frac{x}{\|x\|} \]
using a suitable norm \( \| \cdot \| \).

Example:
Chroma vector \( x \in \mathbb{R}^{12} \)
Euclidean norm
\[
\|x\| := \left( \sum_{i=0}^{11} |x(i)|^2 \right)^{1/2}
\]

Example: C4 played by piano

Chromagram

Normalized chromagram
Audio Features

Normalization

Replace a vector by the normalized vector

$$x / \| x \|$$

using a suitable norm $\| \cdot \|$

Example:
Chroma vector $x \in \mathbb{R}^{12}$
Euclidean norm

$$\| x \| := \left( \sum_{i=0}^{11} |x(i)|^2 \right)^{1/2}$$

Example: C4 played by piano

Log-chromagram

Normalized log-chromagram
Audio Features

Chroma features (normalized)
Audio Features

Chroma features

Chromagram

Chromagram after logarithmic compression and normalization

Chromagram based on a piano tuned 40 cents upwards

Chromagram after applying a cyclic shift of four semitones upwards
Audio Features

- There are many ways to implement chroma features
- Properties may differ significantly
- Appropriateness depends on respective application

Chroma Toolbox: Pitch, Chroma, CENS, CRP

- http://www.mpi-inf.mpg.de/resources/MIR/chromatoolbox/
- MATLAB implementations for various chroma variants
Additional Material
Inner Product

\[ \langle x | y \rangle := \sum_{n=0}^{N-1} x(n)\overline{y(n)} \quad \text{for} \quad x, y \in \mathbb{C}^N \]

Length of a vector

\[ \|x\| := \sqrt{\langle x | x \rangle} \]

Angle between two vectors

\[ \cos(\varphi) = \frac{|\langle x | y \rangle|}{\|x\| \cdot \|y\|} \]

Orthogonality of two vectors

\[ \langle x | y \rangle = 0 \]
Inner Product

Measuring the similarity of two functions

\[ \int_{t \in \mathbb{R}} f(t) \cdot g(t) \, dt \]

- Area mostly positive and large
- Integral large
- Similarity high
Inner Product

Measuring the similarity of two functions

\[ \int_{t \in \mathbb{R}} f(t) \cdot g(t) \, dt \]

→ Area positive and negative
→ Integral small
→ Similarity low
Discretization
Discretization

Sampling
Discretization

Sampling
Discretization

Quantization

![Graph showing Discretization and Quantization](image)
Discretization

Quantization

Discretization

Quantization

Quantization step size
Discretization

Sampling

\[ f : \mathbb{R} \rightarrow \mathbb{R} \quad \text{CT-signal} \]
\[ T > 0 \quad \text{Sampling period} \]
\[ x(n) := f(n \cdot T) \quad \text{Equidistant sampling, } n \in \mathbb{Z} \]
\[ x : \mathbb{Z} \rightarrow \mathbb{R} \quad \text{DT-signal} \]
\[ x(n) \quad \text{Sample taken at time } t = n \cdot T \]
\[ F_s := 1/T \quad \text{Sampling rate} \]
Discretization

Aliasing

Original signal
Discretization

Aliasing

Original signal

Sampled signal using a sampling rate of 12 Hz
Discretization

Aliasing

Original signal

Sampled signal using a sampling rate of 12 Hz

Reconstructed signal
Discretization

Aliasing

Original signal

Sampled signal using a sampling rate of \textbf{6 Hz}

Reconstructed signal
Discretization

Aliasing

Original signal

Sampled signal using a sampling rate of 3 Hz

Reconstructed signal
Discretization

Integrals and Riemann sums

CT-signal $f$
Discretization

Integrals and Riemann sums

CT-signal \( f \)
Integral (total area) \( \int_{t \in \mathbb{R}} |f(t)| \, dt \)
Discretization

Integrals and Riemann sums

CT-signal \( f \)

Integral (total area) \( \int_{t \in \mathbb{R}} f(t) \, dt \)

DT-signals (obtained by 1-sampling) \( \chi \)
Discretization

Integrals and Riemann sums

CT-signal \( f \)
Integral (total area) \[ \int_{t \in \mathbb{R}} f(t) \, dt \approx \sum_{n \in \mathbb{Z}} x(n) \]

DT-signals (obtained by 1-sampling) \( x \)
Riemann sum (total area) \( \rightarrow \text{Approximation of integral} \)
Discretization

Integrals and Riemann sums

First CT-signal and DT-signal

Second CT-signal and DT-signal

Product of CT-signals and DT-signals
Discretization

Integrals and Riemann sums

First CT-signal and DT-signal

Second CT-signal and DT-signal

Product of CT-signals and DT-signals

Integral \approx \text{Riemann sum} \quad \int_{t \in \mathbb{R}} f(t)g(t) \, dt \approx \sum_{n \in \mathbb{Z}} x(n)y(n)
Exponential Function

Polar coordinate representation of a complex number

\[ c = a + ib \]
Exponential Function

Real and imaginary part (Euler’s formula)

\[ \exp(i\gamma) = \cos(\gamma) + i\sin(\gamma) \]

\[ |\exp(i\gamma)| = 1 \]

\[ \exp(i\gamma) = \exp(i(\gamma + 2\pi)) \]
Exponential Function

Complex conjugate number

$$\exp(i\gamma) = \exp(-i\gamma)$$
Exponential Function

Additivity property

\[ \exp(i(\gamma_1 + \gamma_2)) = \exp(i\gamma_1) \exp(i\gamma_2) \]
Fourier Transform

Chirp signal with $\lambda = 0.003$

$$f(t) := \begin{cases} 
\sin(\lambda \cdot \pi t^2), & \text{for } t \geq 0 \\
0, & \text{for } t < 0 
\end{cases}$$
Fourier Transform

Chirp signal with $\lambda = 0.004$

$$f(t) := \begin{cases} \sin(\lambda \cdot \pi t^2), & \text{for } t \geq 0 \\ 0, & \text{for } t < 0 \end{cases}$$
Fourier Transform

DFT approximation of Fourier transform

CT-signal

Magnitude Fourier transform

DT-signal (1-sampling)

Magnitude Fourier transform
Fourier Transform

DFT approximation of Fourier transform

CT-signal

Magnitude Fourier transform

DT-signal (1-sampling)

Magnitude Fourier transform
Fast Fourier Transform

**Algorithm:** FFT

**Input:** The length $N = 2^L$ with $N$ being a power of two
The vector $(x(0), \ldots, x(N - 1))^\top \in \mathbb{C}^N$

**Output:** The vector $(X(0), \ldots, X(N - 1))^\top = \text{DFT}_N \cdot (x(0), \ldots, x(N - 1))^\top$

**Procedure:** Let $(X(0), \ldots, X(N - 1)) = \text{FFT}(N, x(0), \ldots, x(N - 1))$ denote the general form of the FFT algorithm.
If $N = 1$ then
$$X(0) = x(0).$$
Otherwise compute recursively:
$$A(0), \ldots, A(N/2 - 1) = \text{FFT}(N/2, x(0), x(2), x(4), \ldots, x(N - 2)),$$
$$B(0), \ldots, B(N/2 - 1) = \text{FFT}(N/2, x(1), x(3), x(5), \ldots, x(N - 1)),$$
$$C(k) = \omega_N^k \cdot B(k) \text{ for } k \in [0 : N/2 - 1],$$
$$X(k) = A(k) + C(k) \text{ for } k \in [0 : N/2 - 1],$$
$$X(N/2 + k) = A(k) - C(k) \text{ for } k \in [0 : N/2 - 1].$$
# Signal Spaces and Fourier Transforms

<table>
<thead>
<tr>
<th>Signal space</th>
<th>$L^2(\mathbb{R})$</th>
<th>$L^2([0,1])$</th>
<th>$\ell^2(\mathbb{Z})$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inner product</td>
<td>$\langle f</td>
<td>g \rangle = \int f(t)\overline{g(t)}dt$</td>
<td>$\langle f</td>
</tr>
<tr>
<td>Norm</td>
<td>$</td>
<td></td>
<td>f</td>
</tr>
<tr>
<td>Definition</td>
<td>$L^2(\mathbb{R}) := { f : \mathbb{R} \rightarrow \mathbb{C} \mid</td>
<td></td>
<td>f</td>
</tr>
<tr>
<td>Elementary frequency function</td>
<td>$\mathbb{R} \rightarrow \mathbb{C}$</td>
<td>$[0,1) \rightarrow \mathbb{C}$</td>
<td>$\mathbb{Z} \rightarrow \mathbb{C}$</td>
</tr>
<tr>
<td>$t \mapsto \exp(2\pi i\omega t)$</td>
<td>$t \mapsto \exp(2\pi ik t)$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Frequency parameter</td>
<td>$\omega \in \mathbb{R}$</td>
<td>$k \in \mathbb{Z}$</td>
<td>$\omega \in [0,1)$</td>
</tr>
<tr>
<td>Fourier representation</td>
<td>$f(t) =$</td>
<td>$f(t) =$</td>
<td>$x(n) =$</td>
</tr>
<tr>
<td>$\int_{\omega \in \mathbb{R}} c_\omega \exp(2\pi i\omega t)d\omega$</td>
<td>$\sum_{k \in \mathbb{Z}} c_k \exp(2\pi ikt)$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Fourier transform</td>
<td>$\hat{f} : \mathbb{R} \rightarrow \mathbb{C}$</td>
<td>$\hat{f} : [0,1) \rightarrow \mathbb{C}$</td>
<td>$\hat{x} : [0,1) \rightarrow \mathbb{C}$</td>
</tr>
<tr>
<td>$\hat{f}(\omega) = c_\omega =$</td>
<td>$\hat{f}(k) = c_k =$</td>
<td>$\hat{x}(\omega) = c_\omega =$</td>
<td></td>
</tr>
<tr>
<td>$\int_{t \in \mathbb{R}} f(t)\exp(-2\pi i\omega t)dt$</td>
<td>$\int_{t \in [0,1)} f(t)\exp(-2\pi ikt)dt$</td>
<td>$\sum_{n \in \mathbb{Z}} x(n)\exp(-2\pi i\omega n)$</td>
<td></td>
</tr>
</tbody>
</table>