L29: Fourier analysis

Introduction

The discrete Fourier Transform (DFT)

The DFT matrix

The Fast Fourier Transform (FFT)

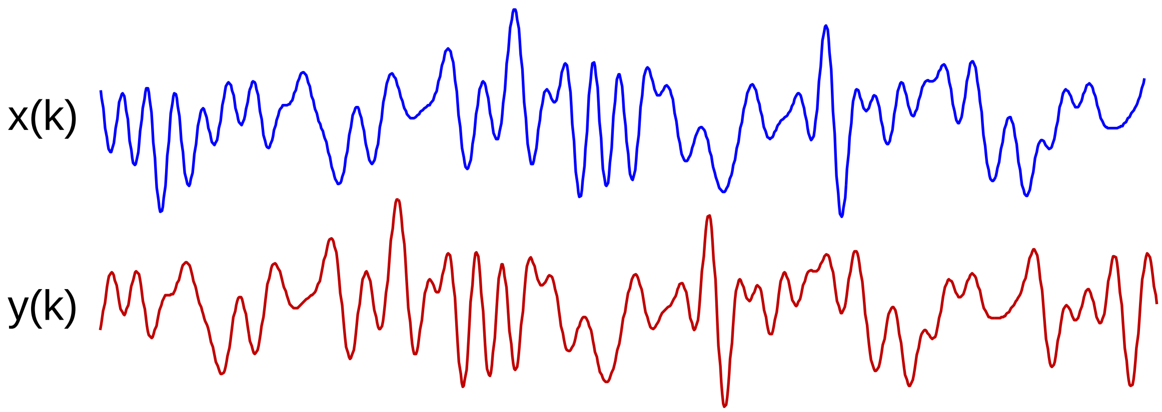
The Short-time Fourier Transform (STFT)

Fourier Descriptors

Introduction

Similarity between time series

– Suppose that you are to determine whether two time series $x(k)$ and $y(k)$ are similar



– One measure of alignment is the inner product of the two signals

$$\langle x, y \rangle = \sum_k x(k)y(k)$$

- If the inner product is large, then the two signals are very much in in alignment
- If the inner product is zero, the two signals are orthogonal

- The Euclidean distance is another measure of (dis)similarity

$$\|x - y\|^2 = \|x\|^2 - 2\langle x, y\rangle + \|y\|^2$$

  - Note that, if we assume that the two signals have unit norm

$$\|x\|^2 = \|y\|^2 = 1$$

  - then the Euclidean distance and the inner product are equivalent
    - Small distance ⇔ large inner product
    - Large distance ⇔ small inner product
  - For this reason, we will use the inner product for the rest of this lecture

Example

– Assume the following time series

$$x = \{\dots 1,1,-1,-1,1,1-1,-1,1,1-1,-1,\dots\}$$
$$y = \{\dots 1,-1,1,-1,1,-1,1,-1,1,-1,1,-1,\dots\}$$

– Compute their inner product
  - What can you say about their degree of similarity?

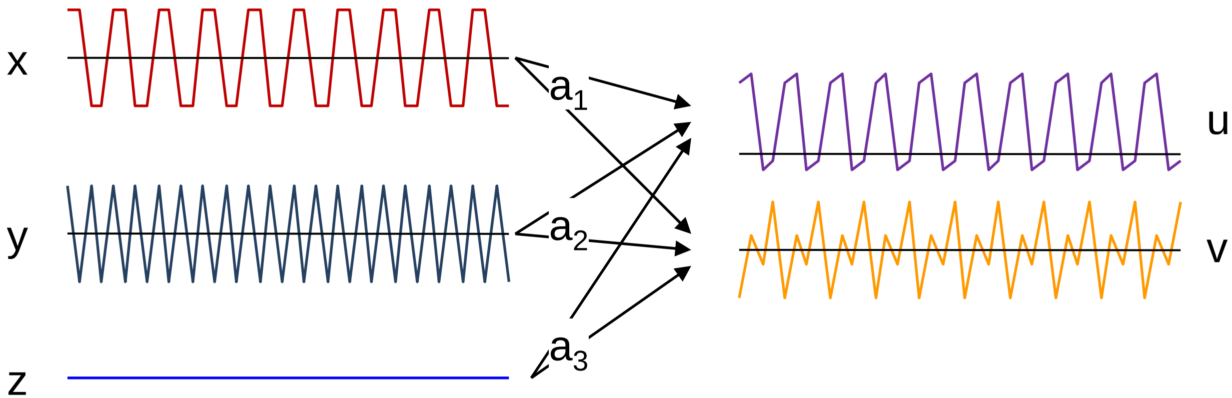
– How about the degree of similarity with the signal z below?

$$z = \{\dots,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,\dots\}$$

– Since all inner products are zero, the three signals (x,y,z) are orthogonal, and therefore independent

– Thus, linear combinations of these signals defines a subspace with three dimensions

$$u = a_1 x + a_2 y + a_3 z$$



- Likewise, two sine waves (shown below) are orthogonal whenever their frequencies are different ($f1 \neq f2$)

$$x(t) = \sin(2\pi f_1 t)$$
$$y(t) = \sin(2\pi f_2 t)$$

- As we will see, a family of sine functions (for all possible frequencies f_i) is at the core of Fourier analysis
- Since sine waves are orthogonal, the analysis is dramatically simplified (e.g., a unique representation exists for every conceivable signal)

Cross-correlation and autocorrelation

Definition

– The inner-product operator allows us to define the cross-correlation between two continuous signals $x(t)$ and $y(t)$ as:

$$R_{xy}(\tau) = \sum_{k=-\infty}^{\infty} x(k)y(k+\tau)$$

- where τ is a shift applied to $y(t)$

– Or, for continuous-time signals

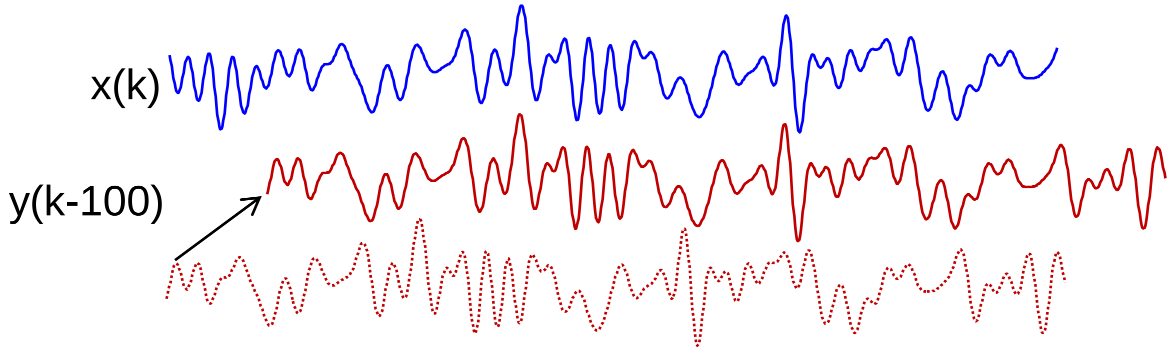
$$R_{xy}(\tau) = \int_{-\infty}^{\infty} x(t)y(t+\tau)\,dt = \langle x(t), y(t+\tau)\rangle$$

– When the cross-correlation is applied to a signal and a copy of itself, it is called the autocorrelation

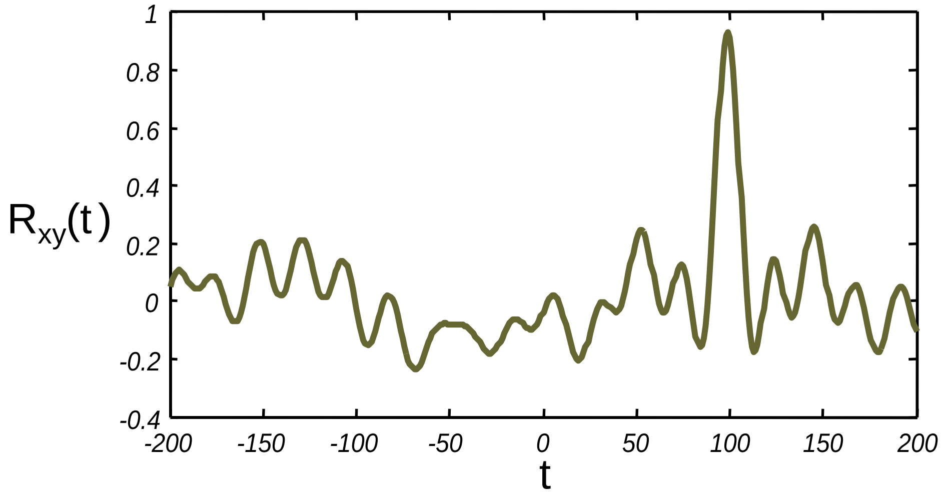
$$R_{xx}(\tau) = \sum_{k=-\infty}^{\infty} x(k)x(k+\tau)$$

Example

– Recall the two signals on slide 2

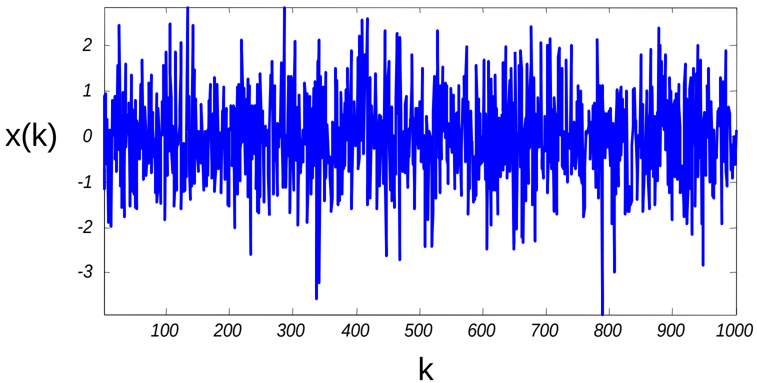


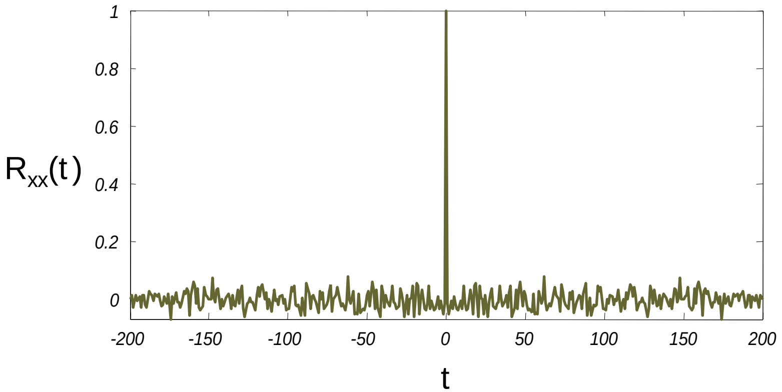
– The cross correlation function reveals that one signal is very close (in our case identical) to a delayed version of the other



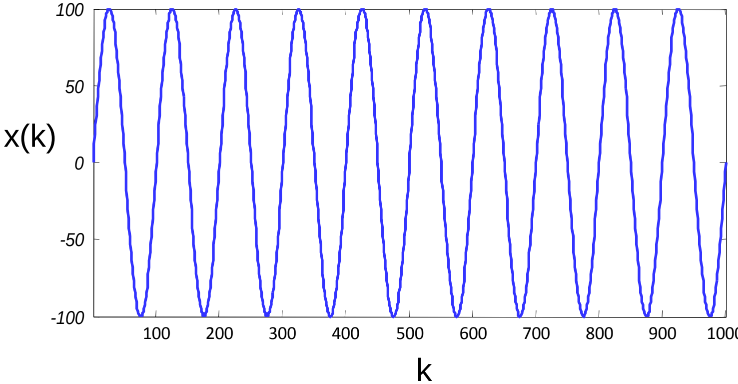
Example II

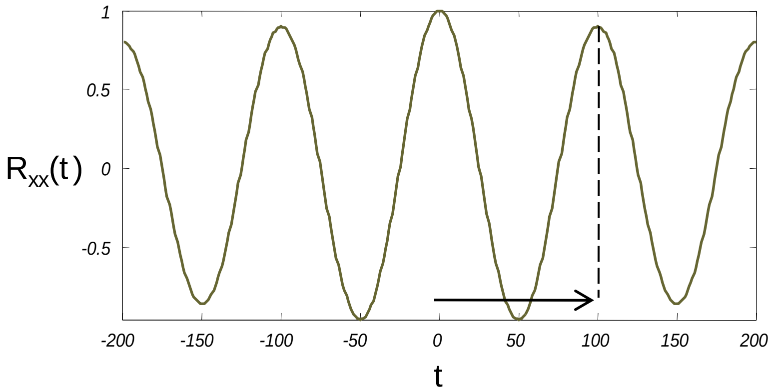
x(k) = randn(1000,1)





x(k) = 100sin(2p0.01k)





The Fourier Transform

- **In Fourier analysis, one represents a signal with a family of sinusoidal functions**
  - Recall from a few slides back that sine waves of different frequencies are orthogonal, so this representation is unique to each signal
  - Fourier analysis transforms the signal from a “time-domain” representation $x(t)$ into a “frequency-domain” representation $X(f)$
  - The collection of values of $X(f)$ at each and every frequency f is called the <u>spectrum</u> of $x(t)$
- **Mathematically, the Fourier Transform is defined as**

$$X(f) = \int_{-\infty}^{\infty} x(t) e^{-j2\pi \mathrm{ft}} dt = \langle x(t), e^{j2\pi \mathrm{ft}} \rangle$$

  - which you can recognize as the inner product between our signal $x(t)$ and the complex sine wave $e^{j2\pi \mathrm{ft}}$
    - Recall Euler’s formula $e^{\pm j\theta} = \cos(\theta) \pm j \sin(\theta)$
    - And the inner product of functions f and g being defined as

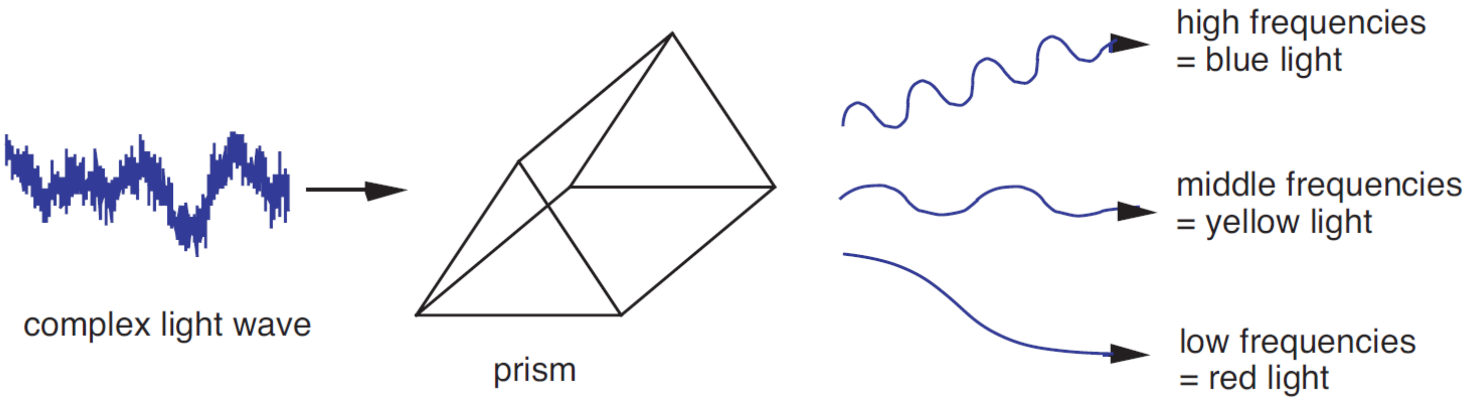
$$\langle f, g \rangle = \int_{-\infty}^{\infty} f g^* dt$$

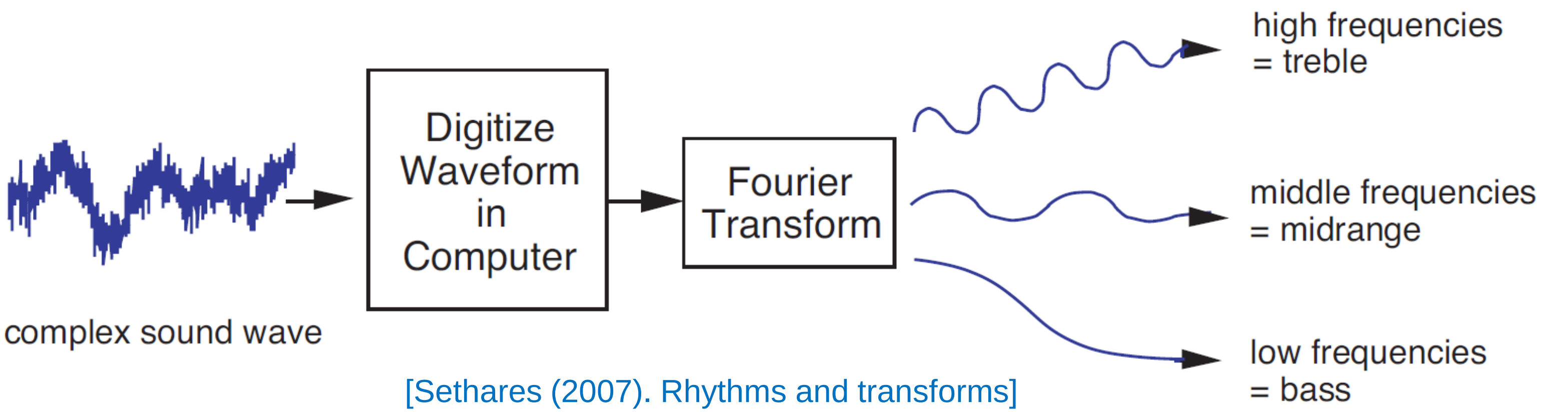
Interpretation of the Fourier Transform

- The Fourier Transform $X(f)$ is defined for each and every frequency f
  - Each term in $X(f)$ represents the inner product of our signal $x(t)$ with a sine wave of frequency f
  - $X(f)$ is a complex number with magnitude m and phase θ , which represent the sine wave that is “closest” to $x(t)$
  - Because the sine waves are orthogonal, their magnitudes m represent the amount of frequency f that is present in $x(t)$
- The collection of values of $X(f)$ for every frequency (each defined by a magnitude m and phase θ) is called the <u>spectrum</u> of $x(t)$
- The Fourier Transform is lossless and invertible, which means that the original signal $x(t)$ can be perfectly reconstructed from $X(f)$
  - This reconstruction is achieved by means of the INVERSE Fourier transform

$$x(t) = \int_{-\infty}^{\infty} X(f) e^{j2\pi \mathrm{ft}} df = \langle X(f), e^{-j2\pi \mathrm{ft}} \rangle$$

The Fourier Transform as a sound "prism"





[Sethares (2007). Rhythms and transforms]

The Discrete Fourier Transform

The DFT differs from the Fourier Transform in three respects

- It applies to discrete-time sequences $x[k] = x(nT)$, where T is the sampling period of a continuous-time signal $x(t)$
- Because we operate in discrete time, the frequency representation is also discrete, and the transform is a summation rather than an integral
- Finally, we work with a finite data record (i.e., we do not have access to the value of the signal for $k \to \infty$)

Mathematically, the DFT is defined as

$$X(n) = \sum_{k=0}^{N-1} x[k] e^{-\frac{j2\pi}{N}nk} = \left\langle x[k], e^{-\frac{j2\pi}{N}nk} \right\rangle, n = 0,1,2 \dots N-1$$

- So the DFT is (again) the inner product of our signal $x[k]$ with a sine wave

Frequency vs. time resolution

The DFT is only defined at frequency multiples of $2\pi/N$, which can be thought of as a "fundamental frequency"

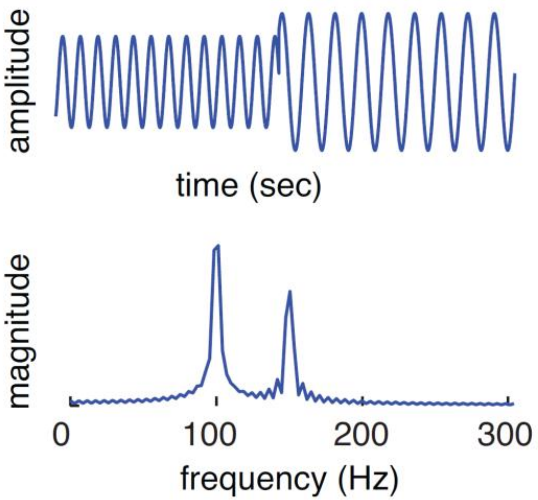
- NOTE: 2π radians correspond to the sampling frequency in Hz
- Therefore, for a given window size, the frequency resolution of the DFT is

$$\Delta f = f_n - f_{n-1} = n\frac{2\pi}{N} - (n-1)\frac{2\pi}{N} = \frac{2\pi}{N} = \frac{sampling\ rate\ (Hz)}{window\ size\ (\#sa.)}$$

- So, the longer the recording, the better the frequency resolution

Why not then use long analysis windows?

- Because longer windows reduce the temporal resolution of frequency events
- Therefore, there is a trade-off between spectral resolution (long windows) and temporal resolution (shorter windows)
- NOTE: Zero-padding can be used to increase the smoothness (or apparent resolution) of the DFT spectrum, but not its true resolution, which remains limited by the length of the original (unpadded) signal



[Sethares, 2007]

The DFT matrix

- Let us denote the "fundamental frequency" signal as

$$W_N = e^{-j\frac{2\pi}{N}} = \cos(2\pi/N) - j\sin(2\pi/N)$$

- Then, the DFT can be expressed as

$$X(n) = \sum_{k=0}^{N-1} x[k](W_N)^{kn}$$

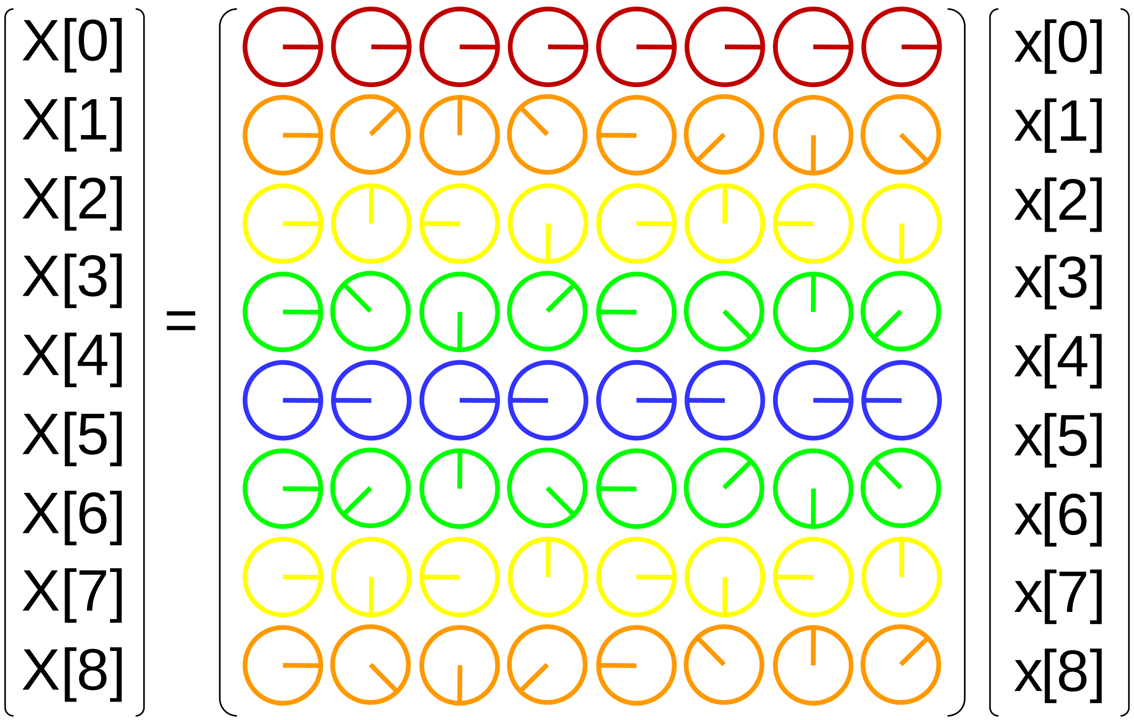
- Or, using matrix notation, as

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

- So the DFT can also be thought of as a projection of the time series data by means of a complex-valued matrix

Symmetry of the DFT matrix

– Note that the k-th row of the DFT matrix consist of a unitary vector rotating clockwise with a constant increment of $2\pi k/N$

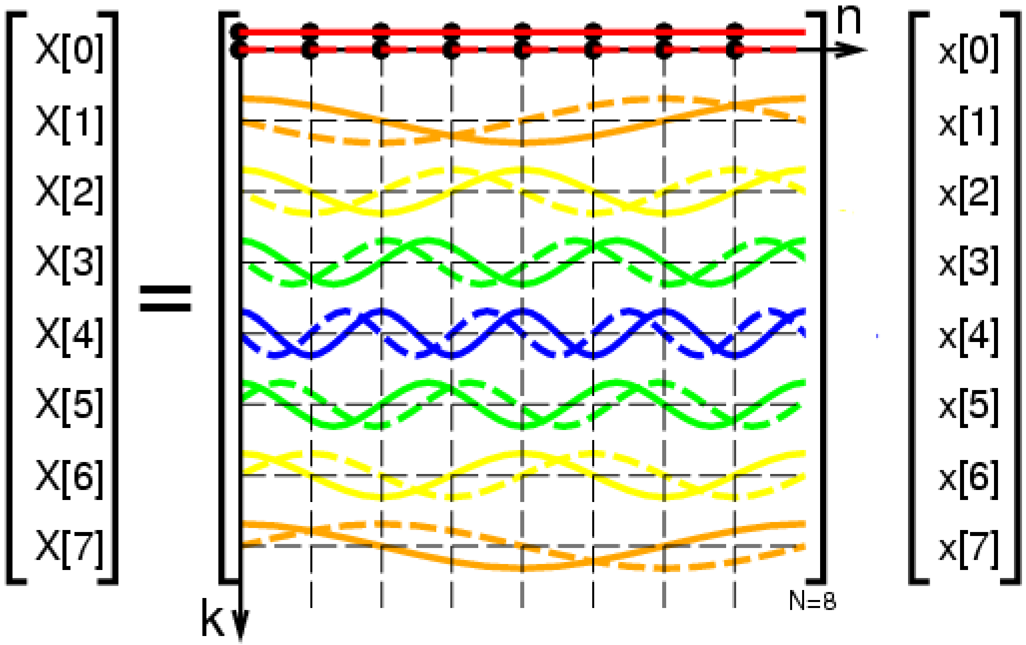


– The second and last row are complex conjugates

– The third and second-to-last are complex conjugates…

Interpretation of the DFT

- So, expressing these rotating unitary vectors in terms of the underlying sine waves, we obtain

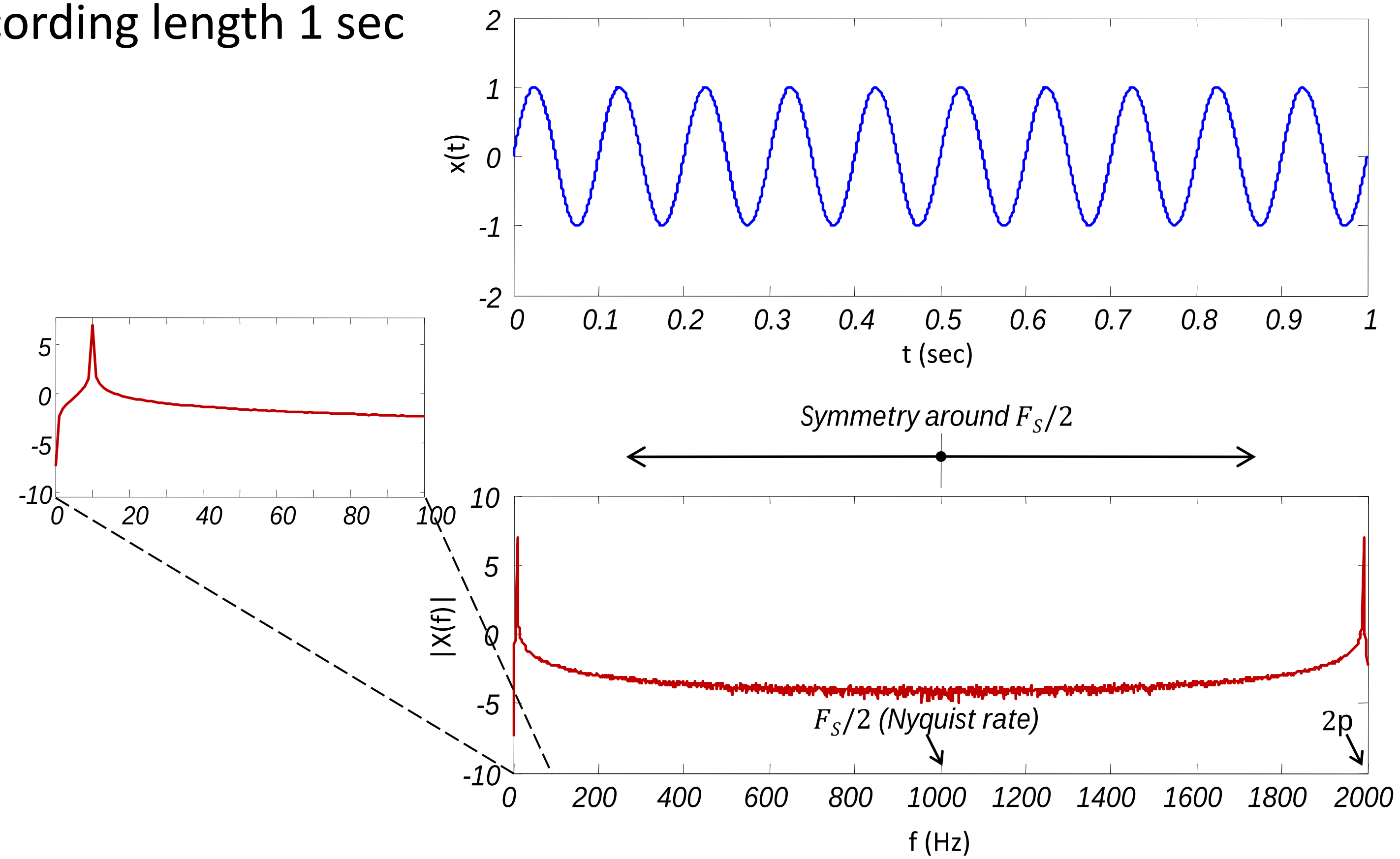


- where the solid line represents the real part and the dashed line represent the imaginary part of the corresponding sine wave
- Note how this illustration brings us back to the definition of the DFT as an inner product between our signal $x[k]$ and a complex sine wave

Illustration borrowed from http://en.wikipedia.org/wiki/DFT_matrix

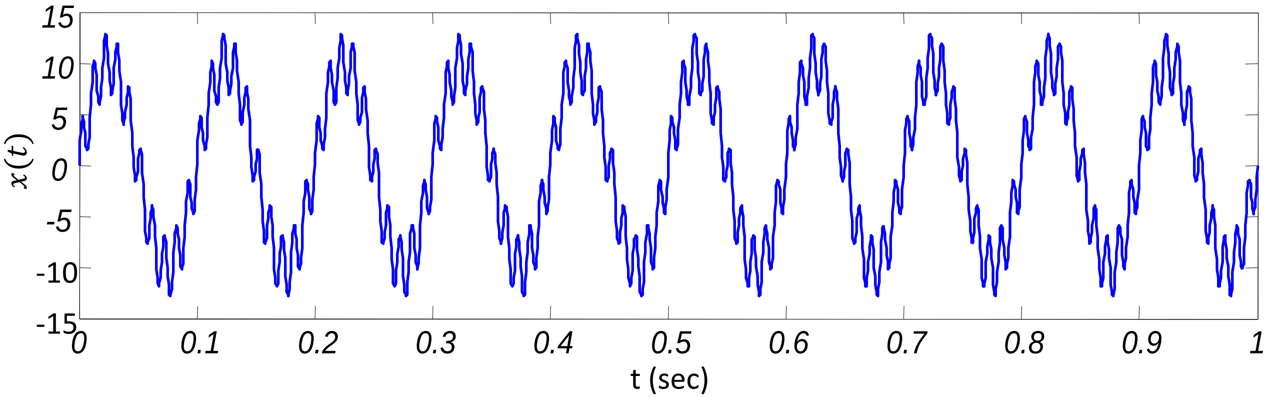
Example I

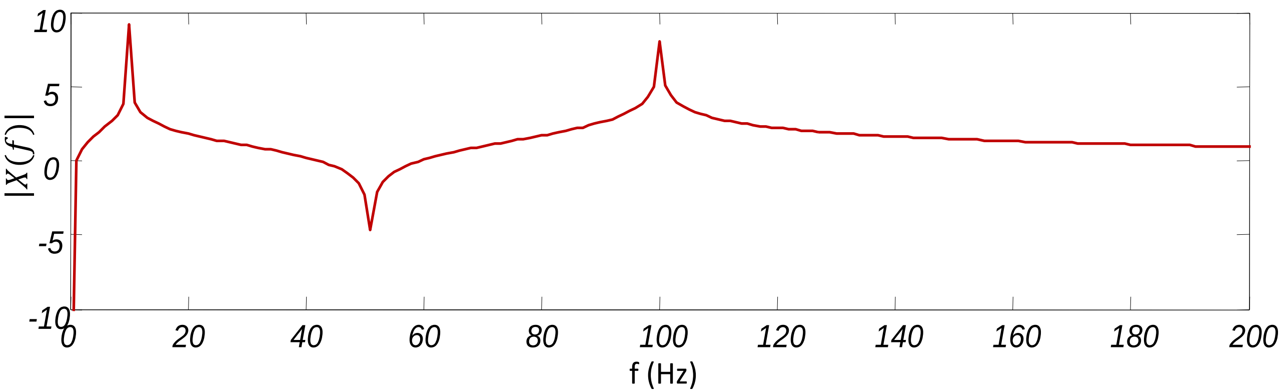
– Sampling rate $F_S = 2kHz$
– Signal $x(t) = \sin(2\pi 10t)$
– Recording length 1 sec



Example II

- Sampling rate $F_S = 2kHz$
- Signal $x(t) = 10\sin(2\pi 10t) + 3\sin(2\pi 100t)$
- Recording length 1 sec

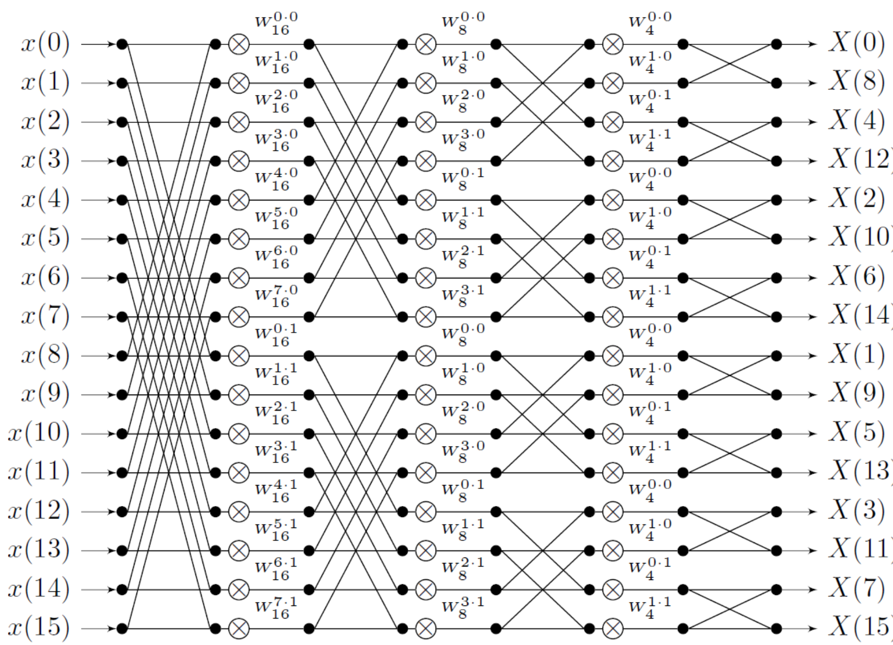




The Fast Fourier Transform

Definition

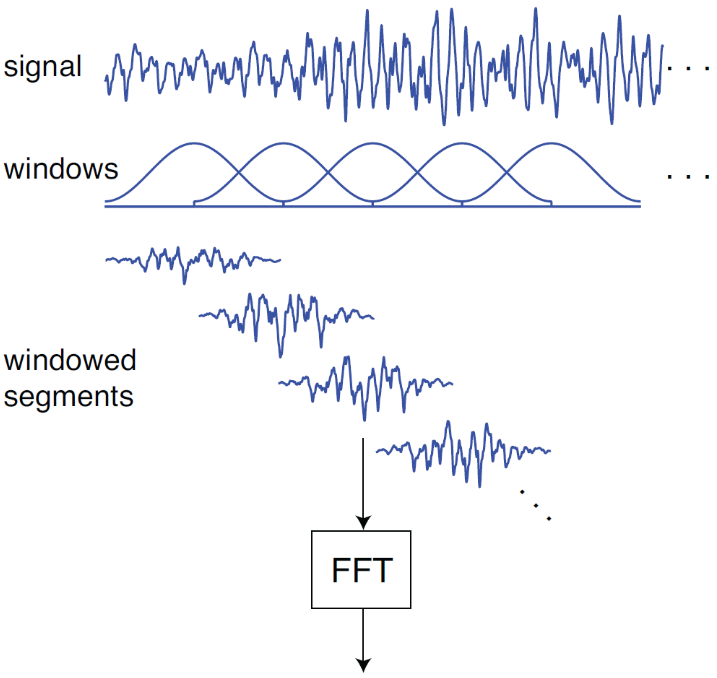
– The FFT refers to any fast algorithm for computing the DFT
– The DFT runs in $O(N^2)$, whereas FFT algorithms run in $O(Nlog_2N)$
– Several FFT algorithms exists, but the most widely used are radix-2 algorithms, which require $N = 2^k$
  - When the number of data points is not a power of 2, it is then just a matter of padding the sequence $x[k]$ with zeros



- **What happens when the signal is not stationary?**
  - As we saw a few slides back, if the DFT/FFT is applied to the entire signal, we will be unable to resolve the spectral changes over time
  - Instead, we can divide the signal into “chunks”, and apply the DFT/FFT to each one of them
  - This strategy is known as the Short-Time Fourier Transform (STFT), and the resulting time-frequency representation is known as a <u>spectrogram</u>
- **The SFTF preserves both temporal and spectral information**
  - By adjusting the size of the “chunks”, the STFT provides a tradeoff between
  - Perfect temporal resolution, as given by the original signal $x(t)$
  - Perfect spectral resolution, as obtained by the Fourier Transform $X(f)$

The SFTF is performed as follows

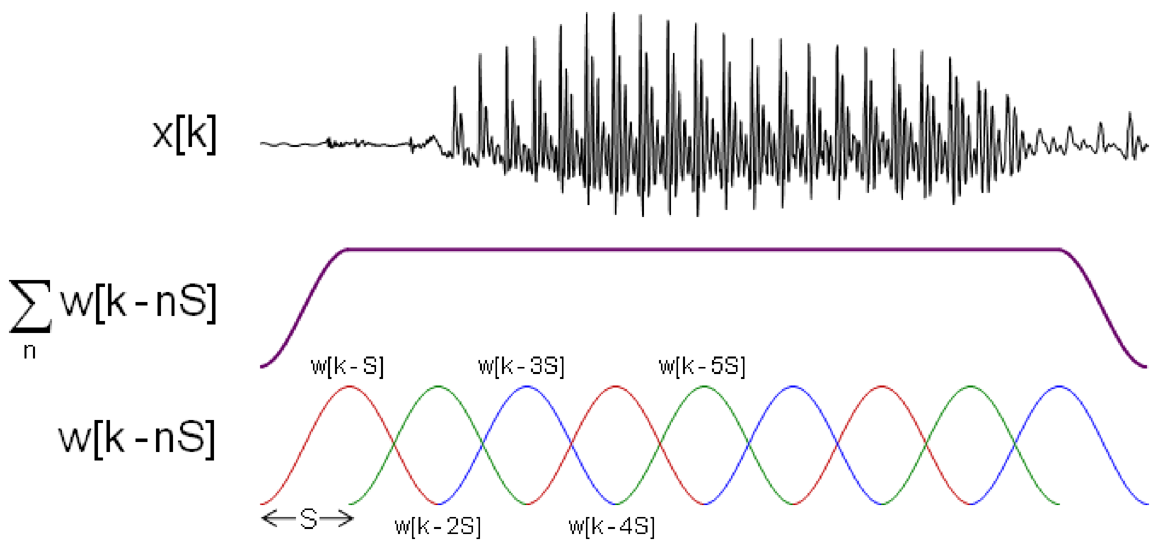
- Define an analysis window size (e.g., 30 ms for narrowband, 5 ms for wideband)
- Define the amount of overlap between windows (e.g., 30%)
- Define a windowing function (e.g., Hann, Gaussian)
- Generate windowed segments (by multiplying signal with the windowing function)
- Apply the FFT to each windowed segment



[Sethares, 2007]

Windowing

- The window function serves several purposes
  - It localizes the Fourier Transform in time, by considering only a short time interval in the signal
  - By having a smooth shape, it minimizes the effects (e.g., high side lobes) of chopping the signal into pieces
  - By overlapping windows, it provides spectral continuity across time
- The windowing functions $w[k - nS]$ must be such that, when overlapped, their sum is unity (or constant)

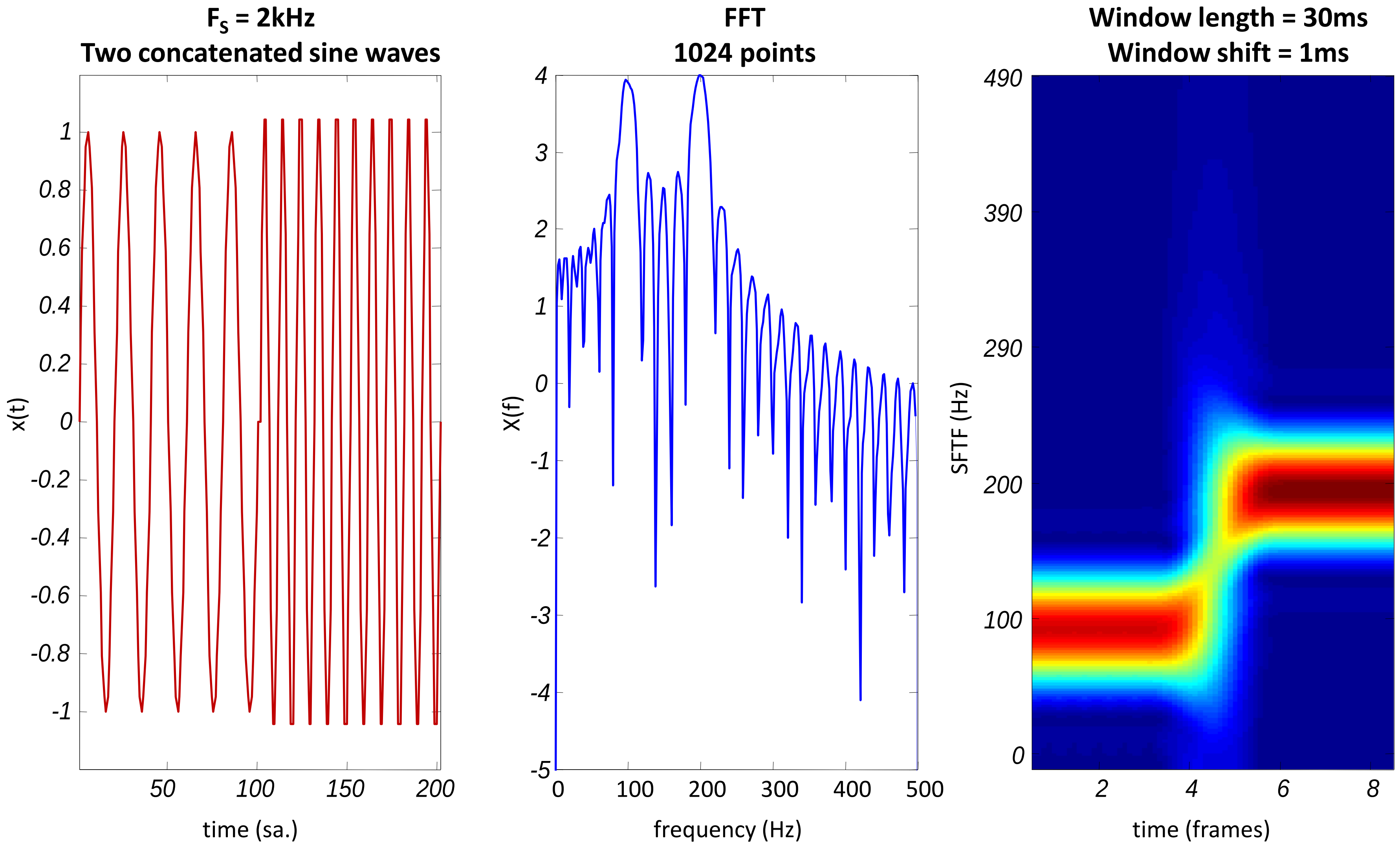


– The STFT is then computed as

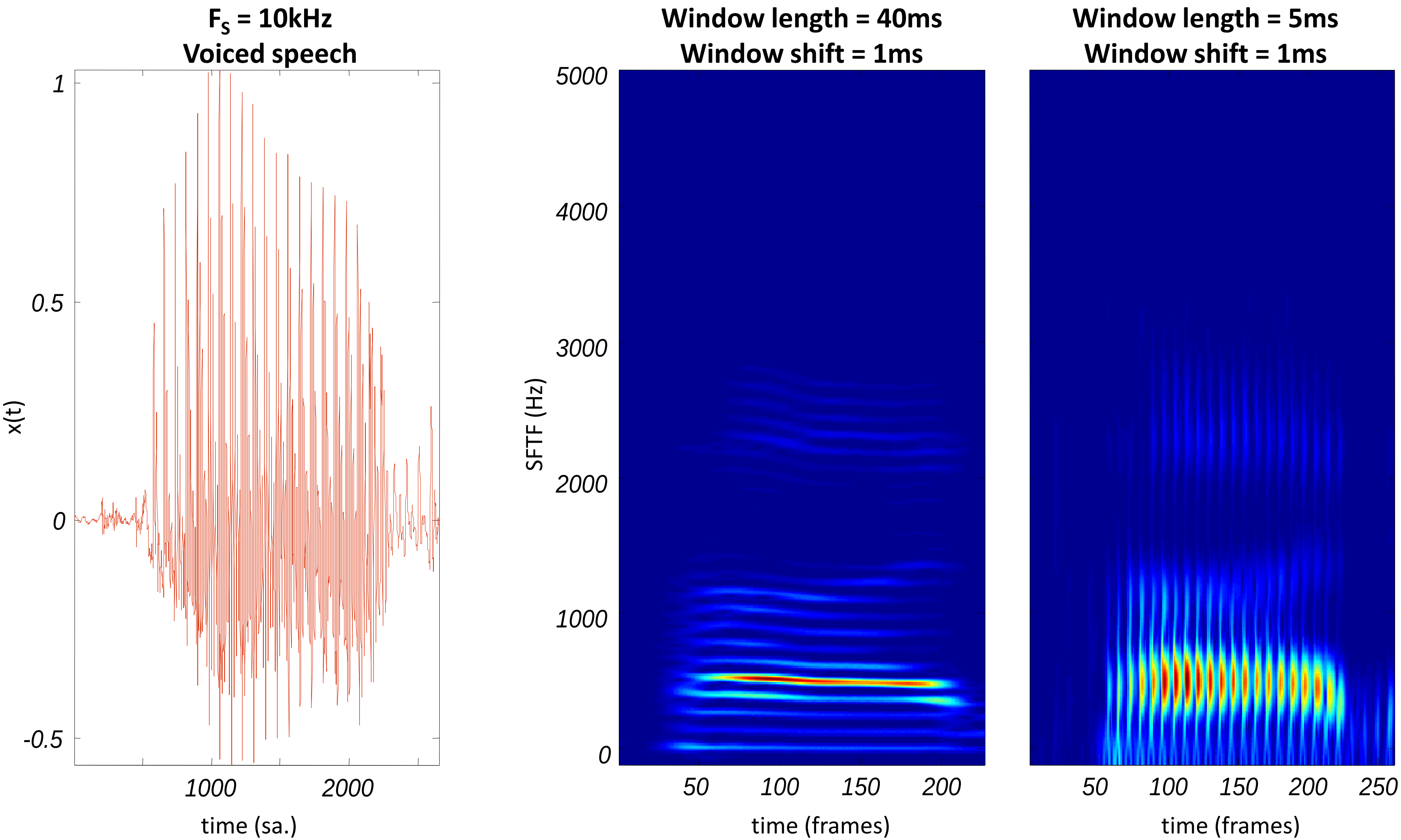
$$X(f_n, t_i) = \sum_{k=0}^{N-1} x[k]w[k-i]e^{-j\frac{2\pi}{N}nk} = \left\langle x[k], w[k-i]e^{j\frac{2\pi}{N}nk} \right\rangle$$

– where f_n is the n-th discrete frequency, and t_i is the starting time of the i-th analysis window

Example I


F_S = 2kHz
Two concatenated sine waves
x(t)
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
50
100
150
200
time (sa.)
FFT
1024 points
X(f)
4
3
2
1
0
-1
-2
-3
-4
-5
0
100
200
300
400
500
frequency (Hz)
Window length = 30ms
Window shift = 1ms
SFTF (Hz)
490
390
290
200
100
0
2
4
6
8
time (frames)

Example II

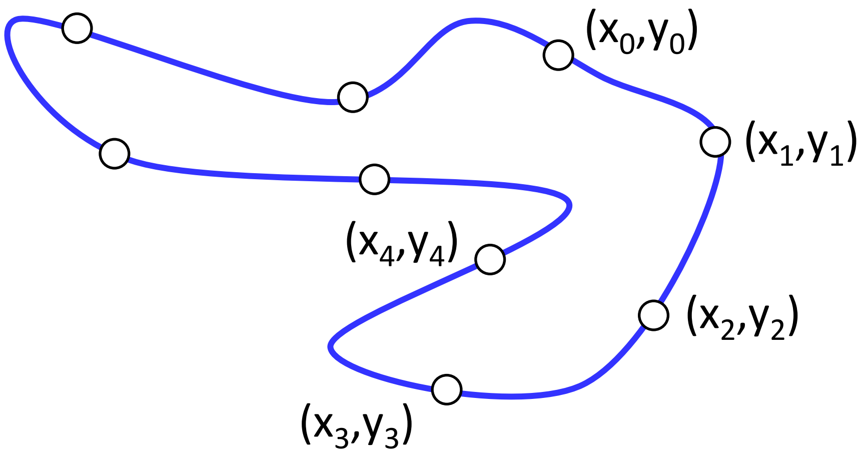


F_S = 10kHz
Voiced speech
Window length = 40ms
Window shift = 1ms
Window length = 5ms
Window shift = 1ms
x(t)
1
0.5
0
-0.5
1000
2000
time (sa.)
SFTF (Hz)
5000
4000
3000
2000
1000
0
50
100
150
200
time (frames)
50
100
150
200
250
time (frames)

Fourier descriptors

Problem definition

- Consider the object below, with contour defined in terms of the coordinates of N points along its periphery
  - We assume that these points are ordered (e.g., CW or CCW)



- which can be represented by a complex vector u as

$$u = \begin{bmatrix} x_0 + jy_0 \\ x_1 + jy_1 \\ \\ x_N + jy_N \end{bmatrix}$$

– Taking the (one-dimensional) DFT of the complex vector u, we obtain

$$U(n) = FFT[u] = \sum_{k=1}^{N-1} u[k] e^{-j\frac{2\pi}{N}nk}$$

– Properties of U(n)

- Translation ($u \to u + d$) only affects the first FD ($U(0) \to U(0) + Nd$)
- Scaling by a factor α ($u \to \alpha u$) scales all FDs accordingly ($U \to \alpha U$)
- Rotation by an angle θ, results in a phase shift ($U \to e^{j\theta} U$)
- Changing the starting point by m positions ($u[k] \to u[k+m]$), also results in a phase shift $U(n) \to e^{j2\pi nm/N} U(n)$

– Hence, by ignoring $U(0)$ and $U(1)$, taking norms, and dividing by $|U(1)|$

$$\tilde{U}(n) = \frac{|U(n)|}{|U(1)|} \quad n = 2,3 \dots, N-1$$

– The coefficients become translation-, scale-, rotation-, and start-point-invariant

– These are known as the Fourier Descriptors of the shape defined by u

  - However, by ignoring the phase of $U(n)$, an essential part of the contour is lost (e.g., two different shapes may have the same FDs)

– Additionally, smooth versions of the original contour can be obtained by performing the IDFT on a subset of the coefficients $U(n)$

Original shape **n=1..5** **n=1..13** **n=1..25** **n=1..65**

[Krzyzak et al. 1988]