university of groningen

faculty of science and engineering

Lecture 6: Fourier Transform

Juan Diego Cardenas-Cartagena, M.Sc.

(j.d.cardenas.cartagena@rug.nl)

Signals and Systems

1B - 2024/2025

Preliminaries

- The deadline for lab 2 is this Friday, December 20 at 17:30.
- Remember to submit your **code in Themis and documentation in Brightspace.**
- Lab assignment 3 is on Brightspace.
- The deadline for reports and code is on Friday, January 17, at 17:30.
- The tutorial tomorrow and the Office Hours on Friday are online. See Brightspace for the link.

General feedback from the Lab 1

- There are a lot of unmentioned libraries.
- Some reports rely on pseudocode as a substitute for explaining their code. Pseudocode should support your thought process, not replace clear explanations. Reports often include pseudocode with overly long variable names, unexplained formulas, and misalignments. You must provide clear in-text explanations of their process, derivations, and formulas, using well-formatted pseudocode only as supplementary support.
- Math formatting is often inconsistent, e.g., mixing n and n, which are not equivalent.

General feedback from the Lab 1

- **Reports often lack formal problem definitions, presenting derived equations without context or background on their derivation.** They feel disjointed, resembling bullet points rather than cohesive, reproducible documents.
- (Optional) While the code is generally good, many submissions lack docstrings, type hints, and have unorganized imports. Additionally, despite high Pylint scores, some code has issues such as unclear variable names, functions performing multiple unrelated tasks, inconsistent naming conventions, and non-pythonic constructs like loops that could use list comprehensions.

Please contact the Teaching Team if you have any questions or need help with the documentation or the Lab assignment.

Overview

1. Recap
2. Discrete Fourier Transform
3. Inverse Discrete Fourier Transform
4. Fast Fourier Transform
5. Shape Descriptors
6. Closing Remarks

Table of Contents

Recap

So far, we have discussed that analyzing signals and systems in the **frequency domain** is easier. However, the process to **move signals** from the time domain **to the frequency domain** and vice versa is not trivial. Let us discuss a tool that can help us with this task.

Recap

LTI filters respond to complex exponentials in a particular way. Consider the input signal

$$x[n] = Ae^{j\varphi} e^{j\hat{\omega} n}$$

Recap

LTI filters respond to complex exponentials in a particular way. Consider the input signal

$$x[n] \quad = \quad Ae^{j\varphi}e^{j\hat{\omega}n}$$

The output signal of the filter given $x[n]$ is

$$y[n] \quad = \quad \left(\sum_{k=0}^{M} b_k e^{-j\hat{\omega}k}\right) Ae^{j\varphi}e^{j\hat{\omega}n}$$

Recap

LTI filters respond to complex exponentials in a particular way. Consider the input signal

$$x[n] = Ae^{j\varphi}e^{j\hat{\omega}n}$$

The output signal of the filter given $x[n]$ is

$$\begin{aligned} y[n] &= \left(\sum_{k=0}^{M} b_k e^{-j\hat{\omega}k}\right) Ae^{j\varphi}e^{j\hat{\omega}n} \\ &= H(e^{j\hat{\omega}})Ae^{j\varphi}e^{j\hat{\omega}n} \end{aligned}$$

Recap

Fourier Series

Any periodic function with fundamental period $T_0 = \frac{1}{F_0}$ can be expressed as a (possibly infinite) sum of harmonically related sinusoids

$$x(t) = \sum_{k=-\infty}^{\infty} a_k e^{j(2\pi/T_0)kt} \tag{1}$$

Recap

Fourier Series

Any periodic function with fundamental period $T_0 = \frac{1}{F_0}$ can be expressed as a (possibly infinite) sum of harmonically related sinusoids

$$x(t) = \sum_{k=-\infty}^{\infty} a_k e^{j(2\pi/T_0)kt} \tag{1}$$

Fourier Coefficients

where

$$a_k = \frac{1}{T_0} \int_0^{T_0} x(t) e^{-j(2\pi/T_0)kt} dt \tag{2}$$

Recap

Fourier Series
Any periodic function with fundamental period $T_0 = \frac{1}{F_0}$ can be expressed as a (possibly infinite) sum of harmonically related sinusoids

$$x(t) = \sum_{k=-\infty}^{\infty} a_k e^{j(2\pi/T_0)kt} \tag{1}$$

Fourier Coefficients
where

$$a_k = \frac{1}{T_0} \int_0^{T_0} x(t) e^{-j(2\pi/T_0)kt} dt \tag{2}$$

Can we combine these two ideas?

Table of Contents

Discrete Fourier transform (DFT)

Definition

The Discrete Fourier transform (DFT) of sequence $x[n]$ is

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-j(2\pi/N)kn} \qquad \text{for } k = 0, \ldots, N-1, \tag{3}$$

where k is the frequency index and N is the number of samples in $x[n]$.

Discrete Fourier transform (DFT)

Definition

The Discrete Fourier transform (DFT) of sequence $x[n]$ is

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-j(2\pi/N)kn} \qquad \text{for } k = 0, \dots, N-1, \tag{3}$$

where k is the frequency index and N is the number of samples in $x[n]$.

- DFT transforms N **time domain** samples into N **frequency domain** samples.

Discrete Fourier transform (DFT)

Definition

The Discrete Fourier transform (DFT) of sequence $x[n]$ is

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-j(2\pi/N)kn} \qquad \text{for } k = 0, \dots, N-1, \tag{3}$$

where k is the frequency index and N is the number of samples in $x[n]$.

- DFT transforms N **time domain** samples into N **frequency domain** samples.
- Each extra sample $x[n]$ informs **one** additional frequency component $X[k]$.

Discrete Fourier transform (DFT)

Definition

The Discrete Fourier transform (DFT) of sequence $x[n]$ is

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-j(2\pi/N)kn} \qquad \text{for } k = 0, \ldots, N-1, \tag{3}$$

where k is the frequency index and N is the number of samples in $x[n]$.

- DFT transforms N **time domain** samples into N **frequency domain** samples.
- Each extra sample $x[n]$ informs **one** additional frequency component $X[k]$.
- In general, $X[k]$ are complex terms consisting of an amplitude and a phase.

Discrete Fourier transform (DFT)

Definition

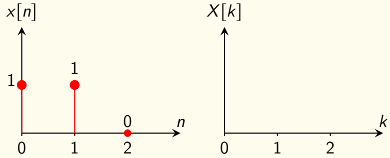
The Discrete Fourier transform (DFT) of sequence $x[n]$ is

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-j(2\pi/N)kn} \qquad \text{for } k = 0, \ldots, N-1, \tag{3}$$

where k is the frequency index and N is the number of samples in $x[n]$.

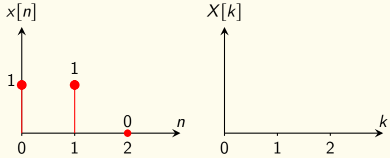
- DFT transforms N **time domain** samples into N **frequency domain** samples.
- Each extra sample $x[n]$ informs **one** additional frequency component $X[k]$.
- In general, $X[k]$ are complex terms consisting of an amplitude and a phase.
- $x[n]$ is a signal in the **time domain** whose representation in the **frequency domain** is $X[k]$.
- We can map k to its corresponding (discrete) frequency using $\hat{\omega} = 2\pi k/N$, with $\hat{\omega} \in [0, 2\pi)$.

Example - Short-length DFT



Suppose we have a periodic signal $x[n] = \{1, 1, 0\}$.

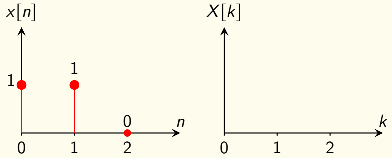
Example - Short-length DFT



Suppose we have a periodic signal $x[n] = \{1, 1, 0\}$.Let us compute the DFT for $x[n]$:

$$X[0] = \sum_{n=0}^{3-1} x[n] e^{-j(2\pi/3)0n}$$

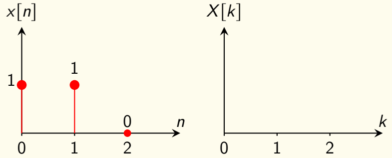
Example - Short-length DFT



Suppose we have a periodic signal $x[n] = \{1, 1, 0\}$.Let us compute the DFT for $x[n]$:

$$
\begin{aligned}
X[0] &= \sum_{n=0}^{3-1} x[n] e^{-j(2\pi/3)0n} \\
&= x[0]e^{-j(2\pi/3)0\cdot 0} + x[1]e^{-j(2\pi/3)0\cdot 1} + x[2]e^{-j(2\pi/3)0\cdot 2}
\end{aligned}
$$

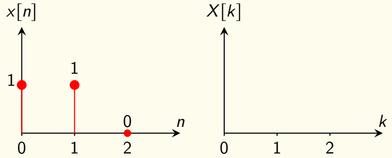
Example - Short-length DFT



Suppose we have a periodic signal $x[n] = \{1, 1, 0\}$.Let us compute the DFT for $x[n]$:

$$
\begin{aligned}
X[0] &= \sum_{n=0}^{3-1} x[n] e^{-j(2\pi/3)0n} \\
&= x[0]e^{-j(2\pi/3)0\cdot 0} + x[1]e^{-j(2\pi/3)0\cdot 1} + x[2]e^{-j(2\pi/3)0\cdot 2} \\
&= 1 \cdot e^{0} + 1 \cdot e^{0} + 0 \cdot e^{0}
\end{aligned}
$$

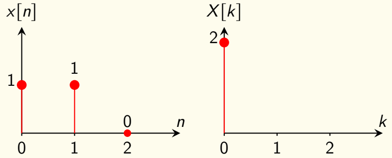
Example - Short-length DFT



Suppose we have a periodic signal $x[n] = \{1, 1, 0\}$.Let us compute the DFT for $x[n]$:

$$
\begin{aligned}
X[0] &= \sum_{n=0}^{3-1} x[n] e^{-j(2\pi/3)0n} \\
&= x[0] e^{-j(2\pi/3)0\cdot 0} + x[1] e^{-j(2\pi/3)0\cdot 1} + x[2] e^{-j(2\pi/3)0\cdot 2} \\
&= 1 \cdot e^0 + 1 \cdot e^0 + 0 \cdot e^0 = 2
\end{aligned}
$$

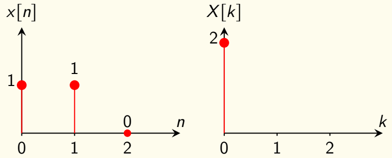
Example - Short-length DFT



Suppose we have a periodic signal $x[n] = \{1, 1, 0\}$.Let us compute the DFT for $x[n]$:

$$
\begin{aligned}
X[0] &= \sum_{n=0}^{3-1} x[n] e^{-j(2\pi/3)0n} \\
&= x[0]e^{-j(2\pi/3)0\cdot 0} + x[1]e^{-j(2\pi/3)0\cdot 1} + x[2]e^{-j(2\pi/3)0\cdot 2} \\
&= 1\cdot e^0 + 1\cdot e^0 + 0\cdot e^0 = 2
\end{aligned}
$$

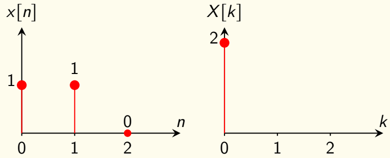
Example - Short-length DFT



Suppose we have a periodic signal $x[n] = \{1, 1, 0\}$. Let us compute the DFT for $x[n]$:

$$X[1] = \sum_{n=0}^{3-1} x[n] e^{-j(2\pi/3)1n}$$

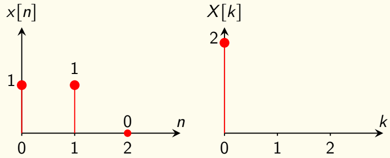
Example - Short-length DFT



Suppose we have a periodic signal $x[n] = \{1, 1, 0\}$. Let us compute the DFT for $x[n]$:

$$
\begin{aligned}
X[1] &= \sum_{n=0}^{3-1} x[n] e^{-j(2\pi/3)1n} \\
&= e^{-j(2\pi/3)1\cdot 0} + e^{-j(2\pi/3)1\cdot 1}
\end{aligned}
$$

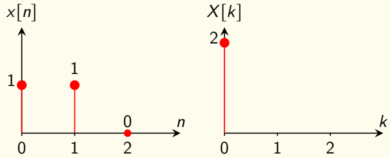
Example - Short-length DFT



Suppose we have a periodic signal $x[n] = \{1, 1, 0\}$. Let us compute the DFT for $x[n]$:

$$\begin{aligned} X[1] &= \sum_{n=0}^{3-1} x[n] e^{-j(2\pi/3)1n} \\ &= e^{-j(2\pi/3)1\cdot 0} + e^{-j(2\pi/3)1\cdot 1} \\ &= e^{0} + e^{-j2\pi/3} \end{aligned}$$

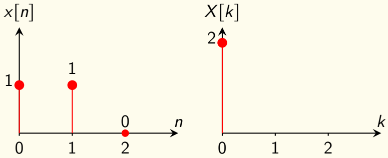
Example - Short-length DFT



Suppose we have a periodic signal $x[n] = \{1, 1, 0\}$. Let us compute the DFT for $x[n]$:

$$
\begin{aligned}
X[1] &= \sum_{n=0}^{3-1} x[n] e^{-j(2\pi/3)1n} \\
&= e^{-j(2\pi/3)1\cdot 0} + e^{-j(2\pi/3)1\cdot 1} \\
&= e^{0} + e^{-j2\pi/3} \\
&= 1 + (\cos(-2\pi/3) + j\sin(-2\pi/3))
\end{aligned}
$$

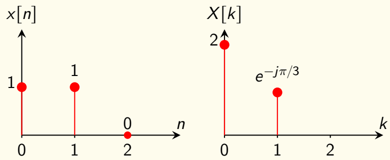
Example - Short-length DFT



Suppose we have a periodic signal $x[n] = \{1, 1, 0\}$. Let us compute the DFT for $x[n]$:

$$
\begin{aligned}
X[1] &= \sum_{n=0}^{3-1} x[n] e^{-j(2\pi/3)1n} \\
&= e^{-j(2\pi/3)1\cdot 0} + e^{-j(2\pi/3)1\cdot 1} \\
&= e^{0} + e^{-j2\pi/3} \\
&= 1 + (\cos(-2\pi/3) + j\sin(-2\pi/3)) \\
&= 1 + (-0.5 - j\sqrt{3/4})j = e^{-j\pi/3}
\end{aligned}
$$

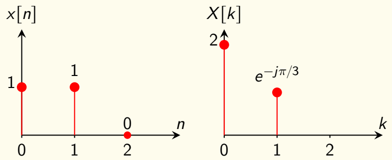
Example - Short-length DFT



Suppose we have a periodic signal $x[n] = \{1, 1, 0\}$. Let us compute the DFT for $x[n]$:

$$
\begin{aligned}
X[1] &= \sum_{n=0}^{3-1} x[n] e^{-j(2\pi/3)1n} \\
&= e^{-j(2\pi/3)1\cdot 0} + e^{-j(2\pi/3)1\cdot 1} \\
&= e^{0} + e^{-j2\pi/3} \\
&= 1 + (\cos(-2\pi/3) + j\sin(-2\pi/3)) \\
&= 1 + (-0.5 - j\sqrt{3/4})j = e^{-j\pi/3}
\end{aligned}
$$

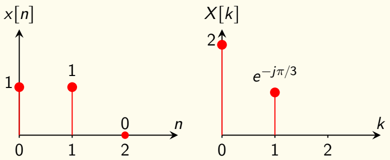
Example - Short-length DFT



Suppose we have a periodic signal $x[n] = \{1, 1, 0\}$. Let us compute the DFT for $x[n]$:

$$X[2] = \sum_{n=0}^{3-1} x[n] e^{-j(2\pi/3)2n}$$

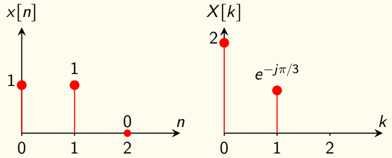
Example - Short-length DFT



Suppose we have a periodic signal $x[n] = \{1, 1, 0\}$. Let us compute the DFT for $x[n]$:

$$
\begin{aligned}
X[2] &= \sum_{n=0}^{3-1} x[n] e^{-j(2\pi/3)2n} \\
&= e^{-j(2\pi/3)2\cdot 0} + e^{-j(2\pi/3)2\cdot 1}
\end{aligned}
$$

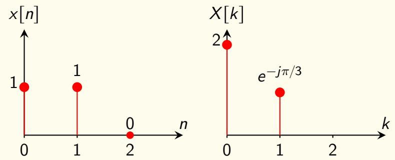
Example - Short-length DFT



Suppose we have a periodic signal $x[n] = \{1, 1, 0\}$. Let us compute the DFT for $x[n]$:

$$
\begin{aligned}
X[2] &= \sum_{n=0}^{3-1} x[n] e^{-j(2\pi/3)2n} \\
&= e^{-j(2\pi/3)2\cdot 0} + e^{-j(2\pi/3)2\cdot 1} \\
&= e^{0} + e^{-j4\pi/3}
\end{aligned}
$$

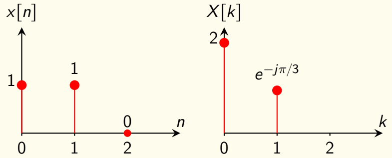
Example - Short-length DFT



Suppose we have a periodic signal $x[n] = \{1, 1, 0\}$. Let us compute the DFT for $x[n]$:

$$
\begin{aligned}
X[2] &= \sum_{n=0}^{3-1} x[n] e^{-j(2\pi/3)2n} \\
&= e^{-j(2\pi/3)2\cdot 0} + e^{-j(2\pi/3)2\cdot 1} \\
&= e^{0} + e^{-j4\pi/3} \\
&= 1 + (\cos(-4\pi/3) + j\sin(-4\pi/3))
\end{aligned}
$$

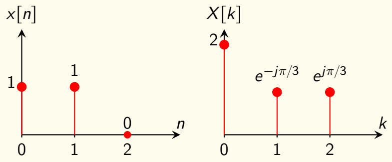
Example - Short-length DFT



Suppose we have a periodic signal $x[n] = \{1, 1, 0\}$. Let us compute the DFT for $x[n]$:

$$
\begin{aligned}
X[2] &= \sum_{n=0}^{3-1} x[n] e^{-j(2\pi/3)2n} \\
&= e^{-j(2\pi/3)2\cdot 0} + e^{-j(2\pi/3)2\cdot 1} \\
&= e^{0} + e^{-j4\pi/3} \\
&= 1 + (\cos(-4\pi/3) + j\sin(-4\pi/3)) \\
&= 1 + (-0.5 + j\sqrt{3/4}) = e^{j\pi/3}
\end{aligned}
$$

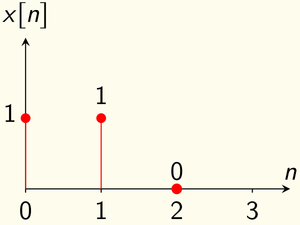
Example - Short-length DFT



Suppose we have a periodic signal $x[n] = \{1, 1, 0\}$. Let us compute the DFT for $x[n]$:

$$
\begin{aligned}
X[2] &= \sum_{n=0}^{3-1} x[n] e^{-j(2\pi/3)2n} \\
&= e^{-j(2\pi/3)2\cdot 0} + e^{-j(2\pi/3)2\cdot 1} \\
&= e^{0} + e^{-j4\pi/3} \\
&= 1 + (\cos(-4\pi/3) + j\sin(-4\pi/3)) \\
&= 1 + (-0.5 + j\sqrt{3/4}) = e^{j\pi/3}
\end{aligned}
$$

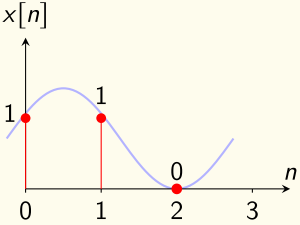
Time and frequency



- DFT $X[k]$ represents input $x[n]$ in the frequency domain[1].

[1] Up to scaling by factor N

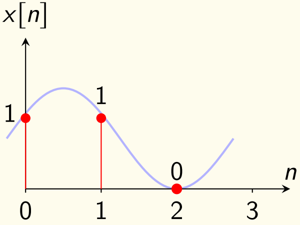
Time and frequency



- DFT $X[k]$ represents input $x[n]$ in the frequency domain[1].

[1] Up to scaling by factor N

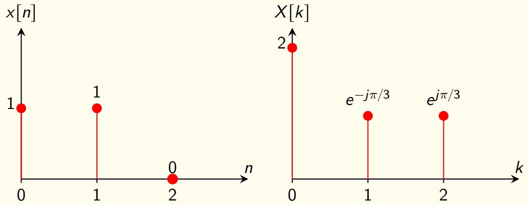
Time and frequency



- DFT $X[k]$ represents input $x[n]$ in the frequency domain[1].
- The original input $x[n]$ can be recovered by sampling these frequencies.

[1] Up to scaling by factor N

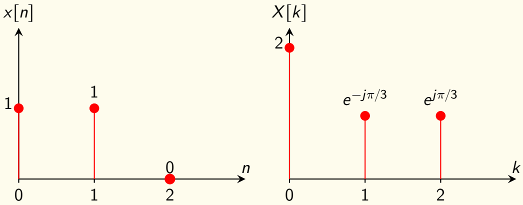
Real-valued signals



Spolier Alert!

Note that the positive and negative frequencies pair are not mirrored in the vertical axis. Does it mean we have a complex signal?

Real-valued signals



Spolier Alert!

Note that the positive and negative frequencies pair are not mirrored in the vertical axis. Does it mean we have a complex signal?
No! The DTF coefficients $X[k]$ have a period of N.

Table of Contents

Inverse discrete Fourier transform (IDFT)

Definition

The inverse DFT of sequence $X[k]$ is

$$x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X[k] e^{j(2\pi/N)kn} \qquad \text{for } k = 0, \ldots, N-1 \tag{4}$$

Inverse discrete Fourier transform (IDFT)

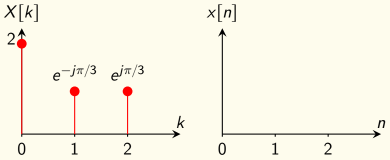
Definition

The inverse DFT of sequence $X[k]$ is

$$x[n] = \frac{1}{N}\sum_{k=0}^{N-1} X[k]e^{j(2\pi/N)kn} \qquad \text{for } k = 0, \ldots, N-1 \tag{4}$$

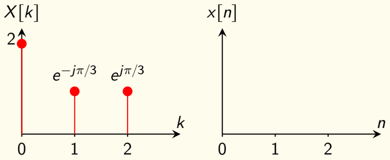
Since $X[k]$ is **periodic**, the reconstructed $x[n]$ is **also periodic**.

Example - Short-length DFT



Consider the sequence $X[k] = \{2, e^{-j\pi/3}, e^{j\pi/3}\}$.

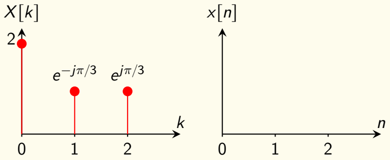
Example - Short-length DFT



Consider the sequence $X[k] = \{2, e^{-j\pi/3}, e^{j\pi/3}\}$. Let us compute the IDTF for X[k].

$$x[0] = \frac{1}{3}\sum_{k=0}^{3-1} X[k]e^{j(2\pi/3)0k}$$

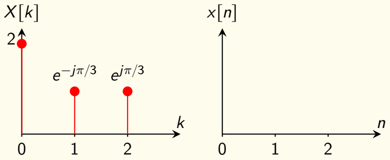
Example - Short-length DFT



Consider the sequence $X[k] = \{2, e^{-j\pi/3}, e^{j\pi/3}\}$. Let us compute the IDTF for X[k].

$$
\begin{aligned}
x[0] &= \frac{1}{3}\sum_{k=0}^{3-1} X[k]e^{j(2\pi/3)0k} \\
&= \frac{1}{3}\left(X[0]e^{j(2\pi/3)0\cdot 0} + X[1]e^{j(2\pi/3)0\cdot 1} + X[2]e^{j(2\pi/3)0\cdot 2}\right)
\end{aligned}
$$

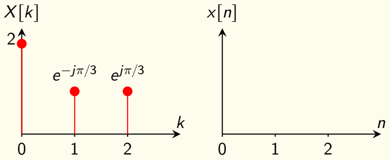
Example - Short-length DFT



Consider the sequence $X[k] = \{2, e^{-j\pi/3}, e^{j\pi/3}\}$. Let us compute the IDTF for X[k].

$$\begin{aligned} x[0] &= \frac{1}{3}\sum_{k=0}^{3-1} X[k]e^{j(2\pi/3)0k} \\ &= \frac{1}{3}\left(X[0]e^{j(2\pi/3)0\cdot 0} + X[1]e^{j(2\pi/3)0\cdot 1} + X[2]e^{j(2\pi/3)0\cdot 2}\right) \\ &= \frac{1}{3}\left(2 + e^{-j\pi/3} + e^{j\pi/3}\right) \end{aligned}$$

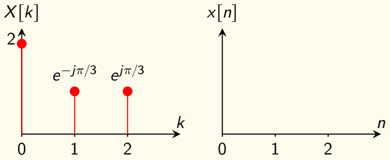
Example - Short-length DFT



Consider the sequence $X[k] = \{2, e^{-j\pi/3}, e^{j\pi/3}\}$. Let us compute the IDTF for X[k].

$$
\begin{aligned}
x[0] &= \frac{1}{3}\sum_{k=0}^{3-1} X[k]e^{j(2\pi/3)0k} \\
&= \frac{1}{3}\left(X[0]e^{j(2\pi/3)0\cdot 0} + X[1]e^{j(2\pi/3)0\cdot 1} + X[2]e^{j(2\pi/3)0\cdot 2}\right) \\
&= \frac{1}{3}\left(2 + e^{-j\pi/3} + e^{j\pi/3}\right) \\
&= \frac{1}{3}\left(2 + 2\cos(\pi/3)\right)
\end{aligned}
$$

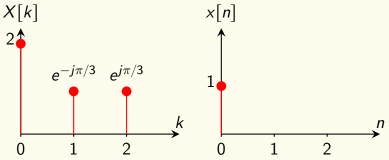
Example - Short-length DFT



Consider the sequence $X[k] = \{2, e^{-j\pi/3}, e^{j\pi/3}\}$. Let us compute the IDTF for X[k].

$$
\begin{aligned}
x[0] &= \frac{1}{3}\sum_{k=0}^{3-1} X[k]e^{j(2\pi/3)0k} \\
&= \frac{1}{3}\left(X[0]e^{j(2\pi/3)0\cdot 0} + X[1]e^{j(2\pi/3)0\cdot 1} + X[2]e^{j(2\pi/3)0\cdot 2}\right) \\
&= \frac{1}{3}\left(2 + e^{-j\pi/3} + e^{j\pi/3}\right) \\
&= \frac{1}{3}\left(2 + 2\cos(\pi/3)\right) \ = 1
\end{aligned}
$$

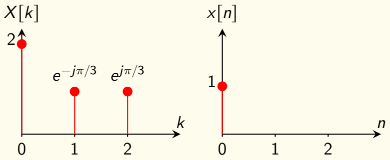
Example - Short-length DFT



Consider the sequence $X[k] = \{2, e^{-j\pi/3}, e^{j\pi/3}\}$. Let us compute the IDTF for X[k].

$$
\begin{aligned}
x[0] &= \frac{1}{3}\sum_{k=0}^{3-1} X[k]e^{j(2\pi/3)0k} \\
&= \frac{1}{3}\left(X[0]e^{j(2\pi/3)0\cdot 0} + X[1]e^{j(2\pi/3)0\cdot 1} + X[2]e^{j(2\pi/3)0\cdot 2}\right) \\
&= \frac{1}{3}\left(2 + e^{-j\pi/3} + e^{j\pi/3}\right) \\
&= \frac{1}{3}\left(2 + 2\cos(\pi/3)\right) = 1
\end{aligned}
$$

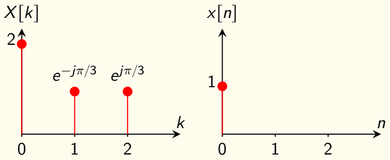
Example - Short-length DFT



Consider the sequence $X[k] = \{2, e^{-j\pi/3}, e^{j\pi/3}\}$. Let us compute the IDTF for X[k].

$$x[1] = \frac{1}{3}\left(X[0]e^{j(2\pi/3)1\cdot 0} + X[1]e^{j(2\pi/3)1\cdot 1} + X[2]e^{j(2\pi/3)1\cdot 2}\right)$$

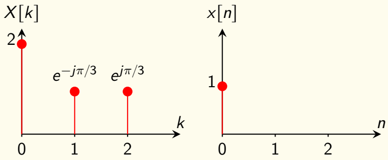
Example - Short-length DFT



Consider the sequence $X[k] = \{2, e^{-j\pi/3}, e^{j\pi/3}\}$. Let us compute the IDTF for X[k].

$$
\begin{aligned}
x[1] &= \frac{1}{3}\left(X[0]e^{j(2\pi/3)1\cdot 0} + X[1]e^{j(2\pi/3)1\cdot 1} + X[2]e^{j(2\pi/3)1\cdot 2}\right) \\
&= \frac{1}{3}\left(2 + e^{-j\pi/3}e^{j2\pi/3} + e^{j\pi/3}e^{j4\pi/3}\right)
\end{aligned}
$$

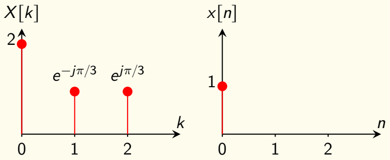
Example - Short-length DFT



Consider the sequence $X[k] = \{2, e^{-j\pi/3}, e^{j\pi/3}\}$. Let us compute the IDTF for X[k].

$$
\begin{aligned}
x[1] &= \frac{1}{3}\left(X[0]e^{j(2\pi/3)1\cdot 0} + X[1]e^{j(2\pi/3)1\cdot 1} + X[2]e^{j(2\pi/3)1\cdot 2}\right) \\
&= \frac{1}{3}\left(2 + e^{-j\pi/3}e^{j2\pi/3} + e^{j\pi/3}e^{j4\pi/3}\right) \\
&= \frac{1}{3}\left(2 + (\cos(\pi/3) + j\sin(\pi/3)) + (\cos(5\pi/3) + j\sin(5\pi/3))\right)
\end{aligned}
$$

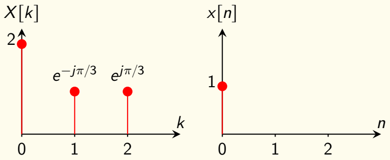
Example - Short-length DFT



Consider the sequence $X[k] = \{2, e^{-j\pi/3}, e^{j\pi/3}\}$. Let us compute the IDTF for X[k].

$$\begin{aligned}
x[1] &= \frac{1}{3}\left(X[0]e^{j(2\pi/3)1\cdot 0} + X[1]e^{j(2\pi/3)1\cdot 1} + X[2]e^{j(2\pi/3)1\cdot 2}\right) \\
&= \frac{1}{3}\left(2 + e^{-j\pi/3}e^{j2\pi/3} + e^{j\pi/3}e^{j4\pi/3}\right) \\
&= \frac{1}{3}\left(2 + (\cos(\pi/3) + j\sin(\pi/3)) + (\cos(5\pi/3) + j\sin(5\pi/3))\right) \\
&= \frac{1}{3}\left(2 + 0.5 + j\sqrt{3/4} + 0.5 - j\sqrt{3/4}\right)
\end{aligned}$$

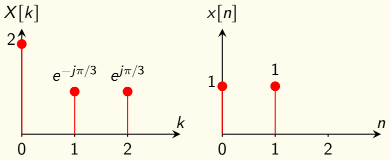
Example - Short-length DFT



Consider the sequence $X[k] = \{2, e^{-j\pi/3}, e^{j\pi/3}\}$. Let us compute the IDTF for X[k].

$$
\begin{aligned}
x[1] &= \frac{1}{3}\left(X[0]e^{j(2\pi/3)1\cdot 0} + X[1]e^{j(2\pi/3)1\cdot 1} + X[2]e^{j(2\pi/3)1\cdot 2}\right) \\
&= \frac{1}{3}\left(2 + e^{-j\pi/3}e^{j2\pi/3} + e^{j\pi/3}e^{j4\pi/3}\right) \\
&= \frac{1}{3}\left(2 + (\cos(\pi/3) + j\sin(\pi/3)) + (\cos(5\pi/3) + j\sin(5\pi/3))\right) \\
&= \frac{1}{3}\left(2 + 0.5 + j\sqrt{3/4} + 0.5 - j\sqrt{3/4}\right) = 1
\end{aligned}
$$

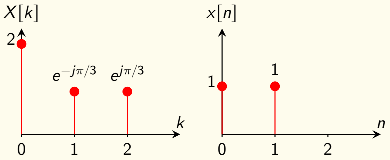
Example - Short-length DFT



Consider the sequence $X[k] = \{2, e^{-j\pi/3}, e^{j\pi/3}\}$. Let us compute the IDTF for X[k].

$$
\begin{aligned}
x[1] &= \frac{1}{3}\left(X[0]e^{j(2\pi/3)1\cdot 0} + X[1]e^{j(2\pi/3)1\cdot 1} + X[2]e^{j(2\pi/3)1\cdot 2}\right) \\
&= \frac{1}{3}\left(2 + e^{-j\pi/3}e^{j2\pi/3} + e^{j\pi/3}e^{j4\pi/3}\right) \\
&= \frac{1}{3}\left(2 + (\cos(\pi/3) + j\sin(\pi/3)) + (\cos(5\pi/3) + j\sin(5\pi/3))\right) \\
&= \frac{1}{3}\left(2 + 0.5 + j\sqrt{3/4} + 0.5 - j\sqrt{3/4}\right) = 1
\end{aligned}
$$

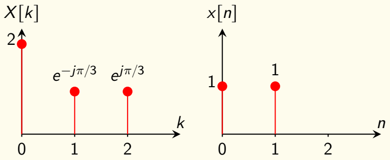
Example - Short-length DFT



Consider the sequence $X[k] = \{2, e^{-j\pi/3}, e^{j\pi/3}\}$. Let us compute the IDTF for X[k].

$$x[2] \quad = \quad \frac{1}{3}\left(X[0]e^{j(2\pi/3)2\cdot 0} + X[1]e^{j(2\pi/3)2\cdot 1} + X[2]e^{j(2\pi/3)2\cdot 2}\right)$$

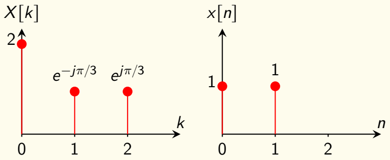
Example - Short-length DFT



Consider the sequence $X[k] = \{2, e^{-j\pi/3}, e^{j\pi/3}\}$. Let us compute the IDTF for X[k].

$$
\begin{aligned}
x[2] &= \frac{1}{3}\left(X[0]e^{j(2\pi/3)2\cdot 0} + X[1]e^{j(2\pi/3)2\cdot 1} + X[2]e^{j(2\pi/3)2\cdot 2}\right) \\
&= \frac{1}{3}\left(2 + e^{-j\pi/3}e^{j4\pi/3} + e^{j\pi/3}e^{j8\pi/3}\right)
\end{aligned}
$$

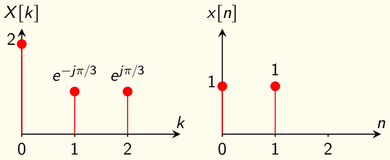
Example - Short-length DFT



Consider the sequence $X[k] = \{2, e^{-j\pi/3}, e^{j\pi/3}\}$. Let us compute the IDTF for X[k].

$$
\begin{aligned}
x[2] &= \frac{1}{3}\left(X[0]e^{j(2\pi/3)2\cdot 0} + X[1]e^{j(2\pi/3)2\cdot 1} + X[2]e^{j(2\pi/3)2\cdot 2}\right) \\
&= \frac{1}{3}\left(2 + e^{-j\pi/3}e^{j4\pi/3} + e^{j\pi/3}e^{j8\pi/3}\right) \\
&= \frac{1}{3}\left(2 + (\cos(\pi) + j\sin(\pi)) + (\cos(3\pi) + j\sin(3\pi))\right)
\end{aligned}
$$

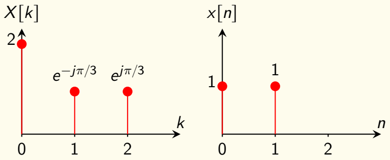
Example - Short-length DFT



Consider the sequence $X[k] = \{2, e^{-j\pi/3}, e^{j\pi/3}\}$. Let us compute the IDTF for X[k].

$$
\begin{aligned}
x[2] &= \frac{1}{3}\left(X[0]e^{j(2\pi/3)2\cdot 0} + X[1]e^{j(2\pi/3)2\cdot 1} + X[2]e^{j(2\pi/3)2\cdot 2}\right) \\
&= \frac{1}{3}\left(2 + e^{-j\pi/3}e^{j4\pi/3} + e^{j\pi/3}e^{j8\pi/3}\right) \\
&= \frac{1}{3}\left(2 + (\cos(\pi) + j\sin(\pi)) + (\cos(3\pi) + j\sin(3\pi))\right) \\
&= \frac{1}{3}\left(2 - 1 + 0 - 1 + 0\right)
\end{aligned}
$$

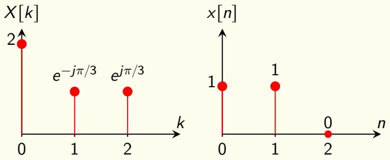
Example - Short-length DFT



Consider the sequence $X[k] = \{2, e^{-j\pi/3}, e^{j\pi/3}\}$. Let us compute the IDTF for X[k].

$$
\begin{aligned}
x[2] &= \frac{1}{3}\left(X[0]e^{j(2\pi/3)2\cdot 0} + X[1]e^{j(2\pi/3)2\cdot 1} + X[2]e^{j(2\pi/3)2\cdot 2}\right) \\
&= \frac{1}{3}\left(2 + e^{-j\pi/3}e^{j4\pi/3} + e^{j\pi/3}e^{j8\pi/3}\right) \\
&= \frac{1}{3}\left(2 + (\cos(\pi) + j\sin(\pi)) + (\cos(3\pi) + j\sin(3\pi))\right) \\
&= \frac{1}{3}\left(2 - 1 + 0 - 1 + 0\right) = 0
\end{aligned}
$$

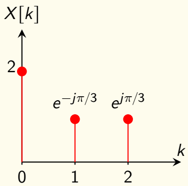
Example - Short-length DFT



Consider the sequence $X[k] = \{2, e^{-j\pi/3}, e^{j\pi/3}\}$. Let us compute the IDTF for X[k].

$$
\begin{aligned}
x[2] &= \frac{1}{3}\left(X[0]e^{j(2\pi/3)2\cdot 0} + X[1]e^{j(2\pi/3)2\cdot 1} + X[2]e^{j(2\pi/3)2\cdot 2}\right) \\
&= \frac{1}{3}\left(2 + e^{-j\pi/3}e^{j4\pi/3} + e^{j\pi/3}e^{j8\pi/3}\right) \\
&= \frac{1}{3}\left(2 + (\cos(\pi) + j\sin(\pi)) + (\cos(3\pi) + j\sin(3\pi))\right) \\
&= \frac{1}{3}(2 - 1 + 0 - 1 + 0) = 0
\end{aligned}
$$

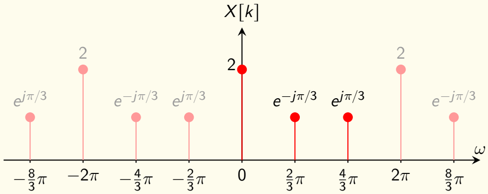
DFT frequencies



The sequence $X[k]$ represents the (discrete) spectrum of $x[n]$.

- The values represent one period in samples
  - The DFT only makes use of **non-negative indices**.
  - So the values represent frequencies $0 \leqslant \hat{\omega} < 2\pi$
- Remaining frequency components are aliases

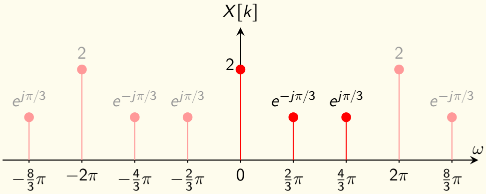
DFT frequencies



The sequence $X[k]$ represents the (discrete) spectrum of $x[n]$.

- The values represent one period in samples
  - The DFT only makes use of **non-negative indices**.
  - So the values represent frequencies $0 \leqslant \hat{\omega} < 2\pi$
- Remaining frequency components are aliases

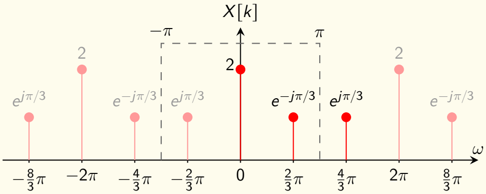
DFT frequencies



The sequence $X[k]$ represents the (discrete) spectrum of $x[n]$.

- The values represent one period in samples
  - The DFT only makes use of **non-negative indices**.
  - So the values represent frequencies $0 \leqslant \hat{\omega} < 2\pi$
- Remaining frequency components are aliases
- The principal frequency interval is $(-\pi < \hat{\omega} \leqslant \pi)$

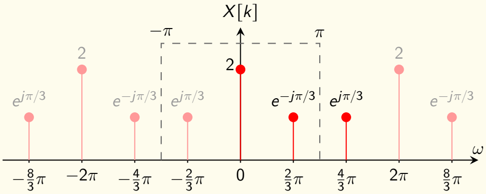
DFT frequencies



The sequence $X[k]$ represents the (discrete) spectrum of $x[n]$.

- The values represent one period in samples
  - The DFT only makes use of **non-negative indices**.
  - So the values represent frequencies $0 \leqslant \hat{\omega} < 2\pi$
- Remaining frequency components are aliases
- The principal frequency interval is $(-\pi < \hat{\omega} \leqslant \pi)$

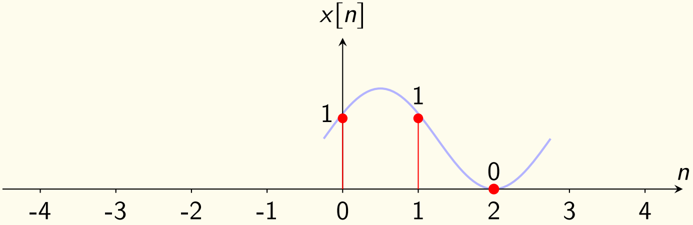
DFT frequencies



The sequence $X[k]$ represents the (discrete) spectrum of $x[n]$.

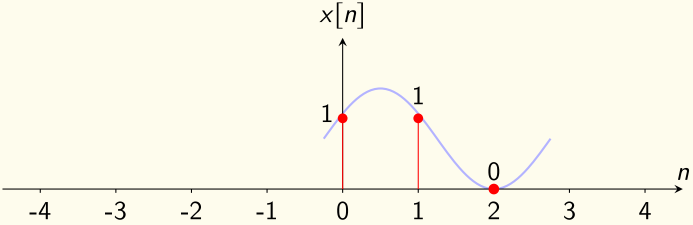
- The values represent one period in samples
  - The DFT only makes use of **non-negative indices**.
  - So the values represent frequencies $0 \leqslant \hat{\omega} < 2\pi$
- Remaining frequency components are aliases
- The principal frequency interval is $(-\pi < \hat{\omega} \leqslant \pi)$
  - This represents $x[n] = 2 + 2\cos(2\pi/3n - \pi/3)$

Time and frequency



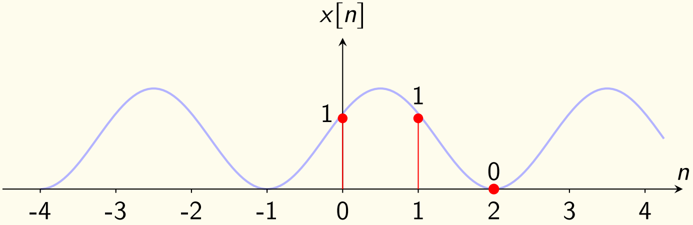
- DFT **X[k]** (in blue) represents the input **x[n]** (in red) as frequencies.

Time and frequency



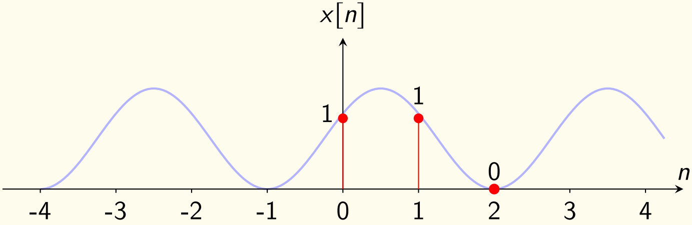
- DFT **X[k]** (in blue) represents the input **x[n]** (in red) as frequencies.
- DFT is periodic.

Time and frequency



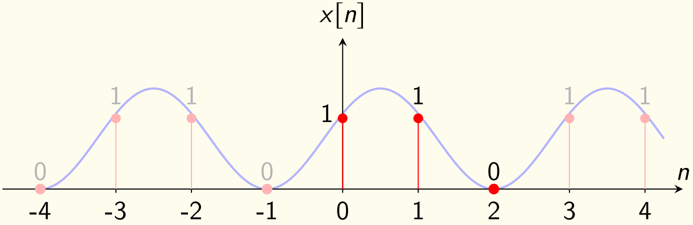
- DFT **X[k]** (in blue) represents the input **x[n]** (in red) as frequencies.
- DFT is periodic.

Time and frequency



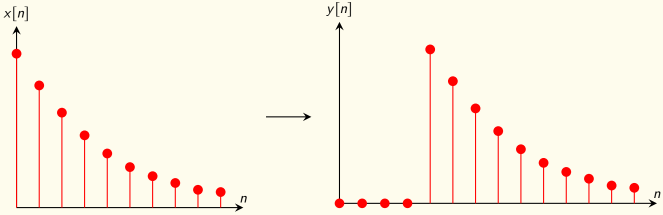
- DFT **X[k]** (in blue) represents the input **x[n]** (in red) as frequencies.
- DFT is periodic.
- DFT technically models a periodic signal $x[n]$.
  - We know where to end the signal through the number of components in $X[k]$

Time and frequency



- DFT **X[k]** (in blue) represents the input **x[n]** (in red) as frequencies.
- DFT is periodic.
- DFT technically models a periodic signal $x[n]$.
  - We know where to end the signal through the number of components in $X[k]$

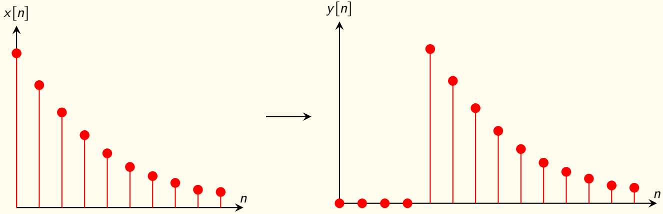
Time delay



A **time delay** $\delta[n - n_0]$ postpones a signal

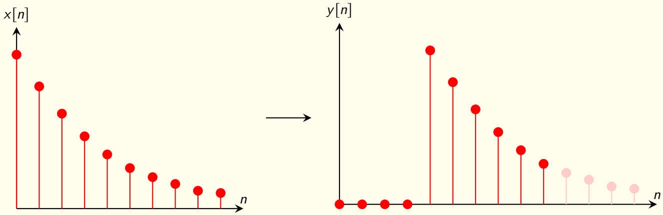
- For periodic signals, this time delay is equivalent to a phase shift $e^{-j\hat{\omega} n_0}$
- For the DFT, the result of applying this phase shift may be unexpected

Applying phase shift $e^{-j\hat{\omega}n_0}$ to the DFT



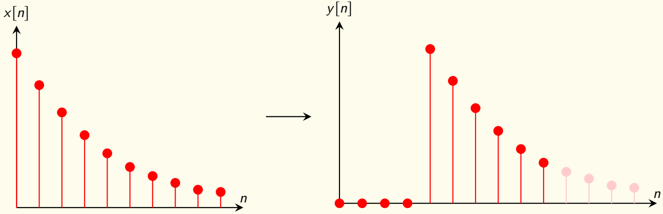
- Applying a phase shift does not change the length of the DFT
  - DFT turns N input samples into N frequency components

Applying phase shift $e^{-j\hat{\omega}n_0}$ to the DFT



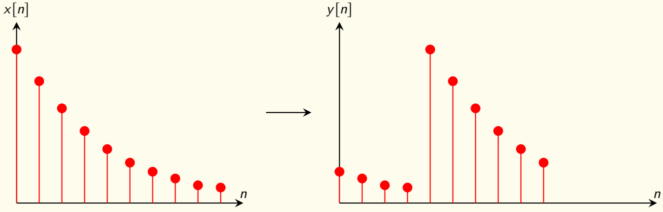
- Applying a phase shift does not change the length of the DFT
  - DFT turns N input samples into N frequency components

Applying phase shift $e^{-j\hat{\omega}n_0}$ to the DFT



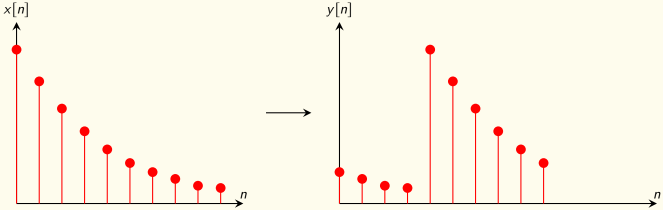
- Applying a phase shift does not change the length of the DFT
  - DFT turns N input samples into N frequency components
- Instead, **the signal is interpreted as being periodic**

Applying phase shift $e^{-j\hat{\omega}n_0}$ to the DFT



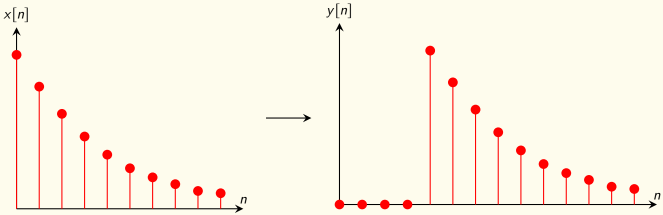
- Applying a phase shift does not change the length of the DFT
    - DFT turns N input samples into N frequency components
- Instead, **the signal is interpreted as being periodic**

Applying phase shift $e^{-j\hat{\omega}n_0}$ to the DFT



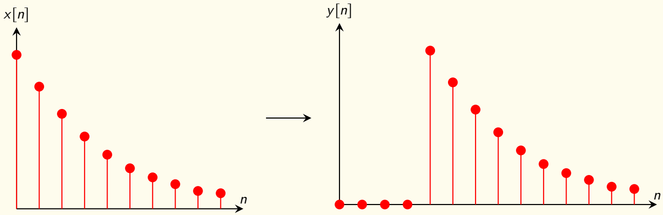
- Applying a phase shift does not change the length of the DFT
  - DFT turns N input samples into N frequency components
- Instead, **the signal is interpreted as being periodic**
- The signal is 'wrapped around' due to the circular shift
- This is a problem: the goal was to delay the signal

Zero padding



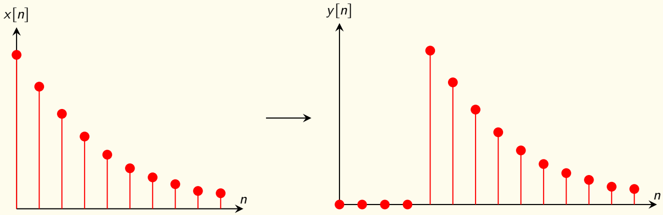
- The intended time delay adds four zeroes to the front

Zero padding



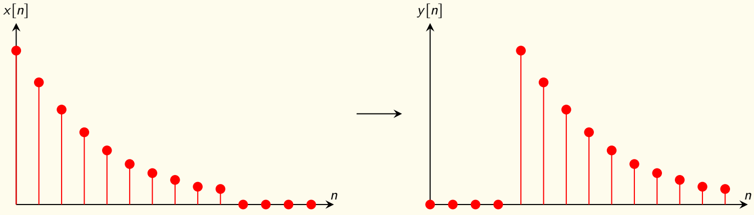
- The intended time delay adds four zeroes to the front
- These zeroes should already be in signal $x[n]$

Zero padding



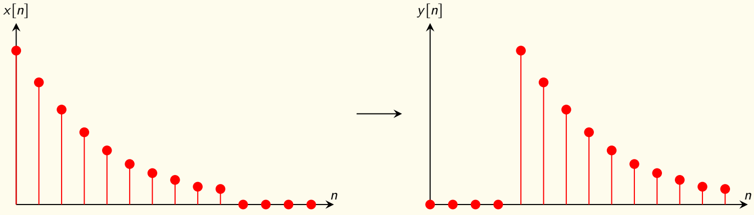
- The intended time delay adds four zeroes to the front
- These zeroes should already be in signal $x[n]$
- We have to pad the original signal with zeroes

Zero padding



- The intended time delay adds four zeroes to the front
- These zeroes should already be in signal $x[n]$
- We have to pad the original signal with zeroes

Zero padding



- The intended time delay adds four zeroes to the front
- These zeroes should already be in signal $x[n]$
- We have to pad the original signal with zeroes
- The delay now coincides with a phase shift
    - This zero padding accounts for periodicity

Break!

See you at ____________________

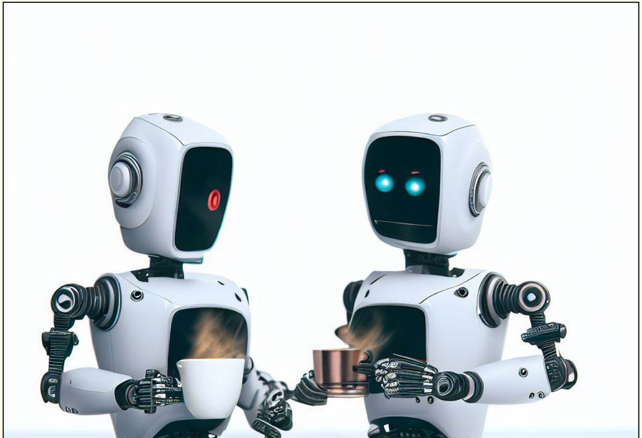


Table of Contents

Computing Fourier transform

The DFT is calculated by

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-j(2\pi/N)kn} \qquad \text{for } k = 0, \ldots, N-1$$

Computing Fourier transform

The DFT is calculated by

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-j(2\pi/N)kn} \qquad \text{for } k = 0, \ldots, N-1$$

Alternatively, in matrix form:

$$\begin{bmatrix} X[0] \\ X[1] \\ X[2] \\ \vdots \\ X[N-1] \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & e^{-j2\pi/N} & e^{-j4\pi/N} & \cdots & e^{-j2(N-1)\pi/N} \\ 1 & e^{-j4\pi/N} & e^{-j8\pi/N} & \cdots & e^{-j4(N-1)\pi/N} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & e^{-j2(N-1)\pi/N} & e^{-j4(N-1)\pi/N} & \cdots & e^{-j(N-1)(N-1)\pi/N} \end{bmatrix} \cdot \begin{bmatrix} x[0] \\ x[1] \\ x[2] \\ \vdots \\ x[N-1] \end{bmatrix}$$

Computing Fourier transform

The DFT is calculated by

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-j(2\pi/N)kn} \qquad \text{for } k = 0, \ldots, N-1$$

Alternatively, in matrix form:

$$\begin{bmatrix} X[0] \\ X[1] \\ X[2] \\ \vdots \\ X[N-1] \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & e^{-j2\pi/N} & e^{-j4\pi/N} & \cdots & e^{-j2(N-1)\pi/N} \\ 1 & e^{-j4\pi/N} & e^{-j8\pi/N} & \cdots & e^{-j4(N-1)\pi/N} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & e^{-j2(N-1)\pi/N} & e^{-j4(N-1)\pi/N} & \cdots & e^{-j(N-1)(N-1)\pi/N} \end{bmatrix} \cdot \begin{bmatrix} x[0] \\ x[1] \\ x[2] \\ \vdots \\ x[N-1] \end{bmatrix}$$

Time complexity of naive implementation is $O(N^2)$. Can we improve on this?[a]

[a] Cooley & Tukey (1965) An algorithm for the machine computation of complex Fourier series, *Mathematics of Computation 19:297-301*

Deriving Fast Fourier transform

We assume that N is a power of 2

$$X[k] \quad = \quad DFT_N\{x[n]\} = \sum_{n=0}^{N-1} x[n] e^{-j(2\pi/N)kn}$$

Deriving Fast Fourier transform

We assume that N is a power of 2

$$\begin{aligned} X[k] &= DFT_N\{x[n]\} = \sum_{n=0}^{N-1} x[n]e^{-j(2\pi/N)kn} \\ &= x[0]e^{-j0} + x[1]e^{-j(2\pi/N)k} + x[2]e^{-j(2\pi/N)2k} + x[3]e^{-j(2\pi/N)3k} + \cdots + \\ &\quad x[N-2]e^{-j(2\pi/N)(N-2)k} + x[N-1]e^{-j(2\pi/N)(N-1)k} \end{aligned}$$

Deriving Fast Fourier transform

We assume that N is a power of 2

$$
\begin{aligned}
X[k] &= DFT_N\{x[n]\} = \sum_{n=0}^{N-1} x[n]e^{-j(2\pi/N)kn} \\
&= x[0]e^{-j0} + x[1]e^{-j(2\pi/N)k} + x[2]e^{-j(2\pi/N)2k} + x[3]e^{-j(2\pi/N)3k} + \cdots + \\
&\quad x[N-2]e^{-j(2\pi/N)(N-2)k} + x[N-1]e^{-j(2\pi/N)(N-1)k} \\
&= x[0]e^{-j0} + x[2]e^{-j(2\pi/N)2k} + \cdots + x[N-2]e^{-j(2\pi/N)(N-2)k} + \\
&\quad x[1]e^{-j(2\pi/N)k} + x[3]e^{-j(2\pi/N)3k} + \cdots + x[N-1]e^{-j(2\pi/N)(N-1)k}
\end{aligned}
$$

Deriving Fast Fourier transform

We assume that N is a power of 2

$$
\begin{aligned}
X[k] &= DFT_N\{x[n]\} = \sum_{n=0}^{N-1} x[n]e^{-j(2\pi/N)kn} \\
&= x[0]e^{-j0} + x[1]e^{-j(2\pi/N)k} + x[2]e^{-j(2\pi/N)2k} + x[3]e^{-j(2\pi/N)3k} + \cdots + \\
&\quad x[N-2]e^{-j(2\pi/N)(N-2)k} + x[N-1]e^{-j(2\pi/N)(N-1)k} \\
&= x[0]e^{-j0} + x[2]e^{-j(2\pi/N)2k} + \cdots + x[N-2]e^{-j(2\pi/N)(N-2)k} + \\
&\quad x[1]e^{-j(2\pi/N)k} + x[3]e^{-j(2\pi/N)3k} + \cdots + x[N-1]e^{-j(2\pi/N)(N-1)k} \\
&= \sum_{m=0}^{N/2-1} x[2m]e^{-j(2\pi/N)(2m)k} + \\
&\quad \sum_{m=0}^{N/2-1} x[2m+1]e^{-j(2\pi/N)(2m+1)k}
\end{aligned}
$$

Deriving Fast Fourier transform

We assume that N is a power of 2

$$
\begin{aligned}
X[k] &= DFT_N\{x[n]\} = \sum_{n=0}^{N-1} x[n] e^{-j(2\pi/N)kn} \\
&= x[0]e^{-j0} + x[1]e^{-j(2\pi/N)k} + x[2]e^{-j(2\pi/N)2k} + x[3]e^{-j(2\pi/N)3k} + \cdots + \\
&\quad x[N-2]e^{-j(2\pi/N)(N-2)k} + x[N-1]e^{-j(2\pi/N)(N-1)k} \\
&= x[0]e^{-j0} + x[2]e^{-j(2\pi/N)2k} + \cdots + x[N-2]e^{-j(2\pi/N)(N-2)k} + \\
&\quad x[1]e^{-j(2\pi/N)k} + x[3]e^{-j(2\pi/N)3k} + \cdots + x[N-1]e^{-j(2\pi/N)(N-1)k} \\
&= \sum_{m=0}^{N/2-1} x[2m] e^{-j(2\pi/N)(2m)k} + \\
&\quad \sum_{m=0}^{N/2-1} x[2m+1] e^{-j(2\pi/N)(2m)k} e^{-j(2\pi/N)k}
\end{aligned}
$$

Deriving Fast Fourier transform

We assume that N is a power of 2

$$
\begin{aligned}
X[k] &= DFT_N\{x[n]\} = \sum_{n=0}^{N-1} x[n] e^{-j(2\pi/N)kn} \\
&= x[0]e^{-j0} + x[1]e^{-j(2\pi/N)k} + x[2]e^{-j(2\pi/N)2k} + x[3]e^{-j(2\pi/N)3k} + \cdots + \\
&\quad x[N-2]e^{-j(2\pi/N)(N-2)k} + x[N-1]e^{-j(2\pi/N)(N-1)k} \\
&= x[0]e^{-j0} + x[2]e^{-j(2\pi/N)2k} + \cdots + x[N-2]e^{-j(2\pi/N)(N-2)k} + \\
&\quad x[1]e^{-j(2\pi/N)k} + x[3]e^{-j(2\pi/N)3k} + \cdots + x[N-1]e^{-j(2\pi/N)(N-1)k} \\
&= \sum_{m=0}^{N/2-1} x[2m]e^{-j(2\pi/N)(2m)k} + \\
&\quad e^{-j(2\pi/N)k} \sum_{m=0}^{N/2-1} x[2m+1]e^{-j(2\pi/N)(2m)k}
\end{aligned}
$$

Deriving fast Fourier transform

Since N is a power of 2, $M = N/2$ is also a power of 2

$$X[k] \quad = \quad DFT_N\{x[n]\} = \sum_{n=0}^{N-1} x[n] e^{-j(2\pi/N)kn}$$

Deriving fast Fourier transform

Since N is a power of 2, $M = N/2$ is also a power of 2

$$\begin{aligned} X[k] &= DFT_N\{x[n]\} = \sum_{n=0}^{N-1} x[n] e^{-j(2\pi/N)kn} \\ &= \left(\sum_{m=0}^{N/2-1} x[2m] e^{-j(2\pi/N)(2m)k} \right) + e^{-j(2\pi/N)k} \left(\sum_{m=0}^{N/2-1} x[2m+1] e^{-j(2\pi/N)(2m)k} \right) \end{aligned}$$

Deriving fast Fourier transform

Since N is a power of 2, $M = N/2$ is also a power of 2

$$
\begin{aligned}
X[k] &= DFT_N\{x[n]\} = \sum_{n=0}^{N-1} x[n] e^{-j(2\pi/N)kn} \\
&= \left(\sum_{m=0}^{N/2-1} x[2m] e^{-j(2\pi/N)(2m)k} \right) + e^{-j(2\pi/N)k} \left(\sum_{m=0}^{N/2-1} x[2m+1] e^{-j(2\pi/N)(2m)k} \right) \\
&= \left(\sum_{m=0}^{M-1} x[2m] e^{-j(2\pi/(2M))(2m)k} \right) + e^{-j(2\pi/N)k} \left(\sum_{m=0}^{M-1} x[2m+1] e^{-j(2\pi/(2M))(2m)k} \right)
\end{aligned}
$$

Deriving fast Fourier transform

Since N is a power of 2, $M = N/2$ is also a power of 2

$$
\begin{aligned}
X[k] &= DFT_N\{x[n]\} = \sum_{n=0}^{N-1} x[n] e^{-j(2\pi/N)kn} \\
&= \left(\sum_{m=0}^{N/2-1} x[2m] e^{-j(2\pi/N)(2m)k} \right) + e^{-j(2\pi/N)k} \left(\sum_{m=0}^{N/2-1} x[2m+1] e^{-j(2\pi/N)(2m)k} \right) \\
&= \left(\sum_{m=0}^{M-1} x[2m] e^{-j(2\pi/(2M))(2m)k} \right) + e^{-j(2\pi/N)k} \left(\sum_{m=0}^{M-1} x[2m+1] e^{-j(2\pi/(2M))(2m)k} \right) \\
&= \left(\sum_{m=0}^{M-1} x[2m] e^{-j(2\pi/M)km} \right) + e^{-j(2\pi/N)k} \left(\sum_{m=0}^{M-1} x[2m+1] e^{-j(2\pi/M)km} \right)
\end{aligned}
$$

Deriving fast Fourier transform

Since N is a power of 2, $M = N/2$ is also a power of 2

$$
\begin{aligned}
X[k] &= DFT_N\{x[n]\} = \sum_{n=0}^{N-1} x[n] e^{-j(2\pi/N)kn} \\
&= \left(\sum_{m=0}^{N/2-1} x[2m] e^{-j(2\pi/N)(2m)k} \right) + e^{-j(2\pi/N)k} \left(\sum_{m=0}^{N/2-1} x[2m+1] e^{-j(2\pi/N)(2m)k} \right) \\
&= \left(\sum_{m=0}^{M-1} x[2m] e^{-j(2\pi/(2M))(2m)k} \right) + e^{-j(2\pi/N)k} \left(\sum_{m=0}^{M-1} x[2m+1] e^{-j(2\pi/(2M))(2m)k} \right) \\
&= \left(\sum_{m=0}^{M-1} x[2m] e^{-j(2\pi/M)km} \right) + e^{-j(2\pi/N)k} \left(\sum_{m=0}^{M-1} x[2m+1] e^{-j(2\pi/M)km} \right) \\
&= DFT_{N/2}\{x[0], x[2], \ldots, x[N-2]\} + e^{-j(2\pi/N)k} \cdot DFT_{N/2}\{x[1], x[3], \ldots, x[N-1]\}
\end{aligned}
$$

Deriving fast Fourier transform

Since N is a power of 2, $M = N/2$ is also a power of 2

$$
\begin{aligned}
X[k] &= DFT_N\{x[n]\} = \sum_{n=0}^{N-1} x[n]e^{-j(2\pi/N)kn} \\
&= \left(\sum_{m=0}^{N/2-1} x[2m]e^{-j(2\pi/N)(2m)k}\right) + e^{-j(2\pi/N)k}\left(\sum_{m=0}^{N/2-1} x[2m+1]e^{-j(2\pi/N)(2m)k}\right) \\
&= \left(\sum_{m=0}^{M-1} x[2m]e^{-j(2\pi/(2M))(2m)k}\right) + e^{-j(2\pi/N)k}\left(\sum_{m=0}^{M-1} x[2m+1]e^{-j(2\pi/(2M))(2m)k}\right) \\
&= \left(\sum_{m=0}^{M-1} x[2m]e^{-j(2\pi/M)km}\right) + e^{-j(2\pi/N)k}\left(\sum_{m=0}^{M-1} x[2m+1]e^{-j(2\pi/M)km}\right) \\
&= DFT_{N/2}\{x[0], x[2], \ldots, x[N-2]\} + e^{-j(2\pi/N)k} \cdot DFT_{N/2}\{x[1], x[3], \ldots, x[N-1]\}
\end{aligned}
$$

Generating a DFT can be split into generating two smaller DFTs.

Deriving fast Fourier transform

Generating a DFT can be split into generating two smaller DFTs

$$
\begin{aligned}
X[k] &= DFT_N\{x[n]\} \\
&= DFT_{N/2}\{x_{even}[n]\} + e^{-j(2\pi/N)k} \cdot DFT_{N/2}\{x_{odd}[n]\} \\
&= X_{even}[k] + e^{-j(2\pi/N)k} \cdot X_{odd}[k].
\end{aligned}
$$

Deriving fast Fourier transform

Generating a DFT can be split into generating two smaller DFTs

$$\begin{aligned} X[k] &= DFT_N\{x[n]\} \\ &= DFT_{N/2}\{x_{even}[n]\} + e^{-j(2\pi/N)k} \cdot DFT_{N/2}\{x_{odd}[n]\} \\ &= X_{even}[k] + e^{-j(2\pi/N)k} \cdot X_{odd}[k]. \end{aligned}$$

But wait...

Deriving fast Fourier transform

Generating a DFT can be split into generating two smaller DFTs

$$\begin{aligned} X[k] &= DFT_N\{x[n]\} \\ &= DFT_{N/2}\{x_{even}[n]\} + e^{-j(2\pi/N)k} \cdot DFT_{N/2}\{x_{odd}[n]\} \\ &= X_{even}[k] + e^{-j(2\pi/N)k} \cdot X_{odd}[k]. \end{aligned}$$

But wait...

- $X[k]$ consists of N frequency components
- $X_{even}[k]$ only has $N/2$ frequency components
- What about $X[k]$ for $k > N/2$?

Deriving fast Fourier transform

Generating a DFT can be split into generating two smaller DFTs

$$\begin{aligned} X[k] &= DFT_N\{x[n]\} \\ &= DFT_{N/2}\{x_{even}[n]\} + e^{-j(2\pi/N)k} \cdot DFT_{N/2}\{x_{odd}[n]\} \\ &= X_{even}[k] + e^{-j(2\pi/N)k} \cdot X_{odd}[k]. \end{aligned}$$

But wait...

- $X[k]$ consists of N frequency components
- $X_{even}[k]$ only has $N/2$ frequency components
- What about $X[k]$ for $k > N/2$?
- $X[k]$ consists of aliases of $X_{even}[k]$ and $X_{odd}[k]$

$$X[k] = X_{even}[k - N/2] + e^{-j(2\pi/N)k} \cdot X_{odd}[k - N/2], \text{ for } k > N/2. \tag{5}$$

Radix-2

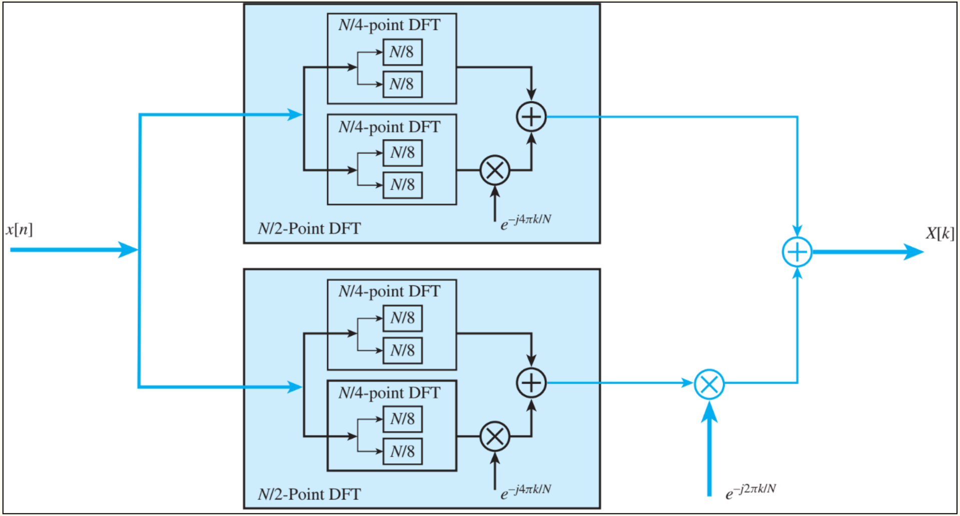
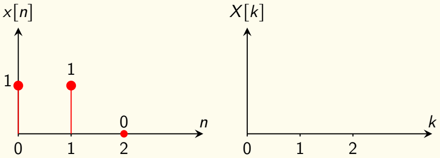


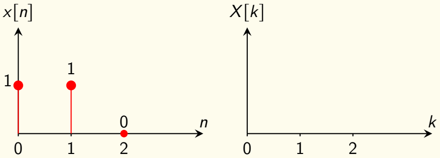
Figure: Inverse DFT is completely analogous to forward DFT. Taken from McClellan, Shafer, & Yoder (2015) *DSP First*

Example - Short-length DFT



Suppose we have a signal $x[n] = \{1, 1, 0\}$

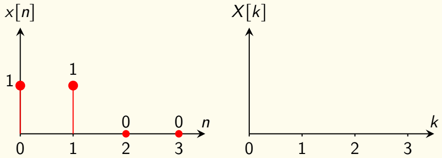
Example - Short-length DFT



Suppose we have a signal $x[n] = \{1, 1, 0\}$

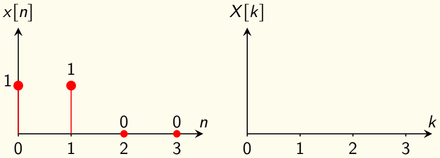
- Since the length is not a power of two, we pad $x[n]$ with zeros

Example - Short-length DFT



Suppose we have a signal $x[n] = \{1, 1, 0, 0\}$

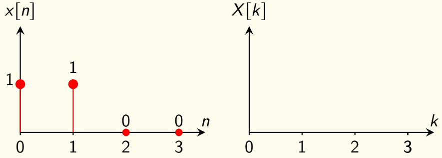
Example - Short-length DFT



Suppose we have a signal $x[n] = \{1, 1, 0, 0\}$

$$X[0] = DFT_4\{1, 1, 0, 0\}[0]$$
$$X[1] = DFT_4\{1, 1, 0, 0\}[1]$$
$$X[2] = DFT_4\{1, 1, 0, 0\}[2]$$
$$X[3] = DFT_4\{1, 1, 0, 0\}[3]$$

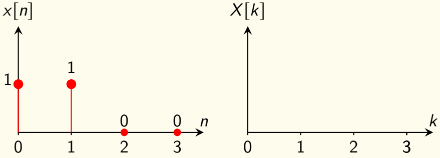
Example - Short-length DFT



Suppose we have a signal $x[n] = \{1, 1, 0, 0\}$

$$
\begin{aligned}
X[0] &= DFT_2\{1,0\}[0] + e^{-j\pi 0/2} DFT_2\{1,0\}[0] \\
X[1] &= DFT_2\{1,0\}[1] + e^{-j\pi 1/2} DFT_2\{1,0\}[1] \\
X[2] &= DFT_2\{1,0\}[2] + e^{-j\pi 2/2} DFT_2\{1,0\}[2] \\
X[3] &= DFT_2\{1,0\}[3] + e^{-j\pi 3/2} DFT_2\{1,0\}[3]
\end{aligned}
$$

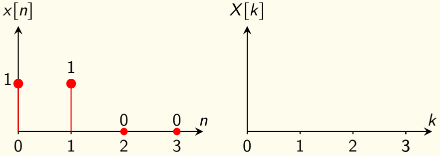
Example - Short-length DFT



Suppose we have a signal $x[n] = \{1, 1, 0, 0\}$

$$
\begin{aligned}
X[0] &= DFT_2\{1,0\}[0] + e^{-j\pi 0/2} DFT_2\{1,0\}[0] \\
X[1] &= DFT_2\{1,0\}[1] + e^{-j\pi 1/2} DFT_2\{1,0\}[1] \\
X[2] &= DFT_2\{1,0\}[0] - e^{-j\pi 0/2} DFT_2\{1,0\}[0] \\
X[3] &= DFT_2\{1,0\}[1] - e^{-j\pi 1/2} DFT_2\{1,0\}[1]
\end{aligned}
$$

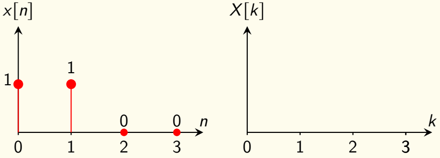
Example - Short-length DFT



Suppose we have a signal $x[n] = \{1, 1, 0, 0\}$

$$
\begin{aligned}
X[0] &= DFT_2\{1,0\}[0] + e^{-j\pi 0/2} DFT_2\{1,0\}[0] \\
X[1] &= DFT_2\{1,0\}[1] + e^{-j\pi 1/2} DFT_2\{1,0\}[1] \\
X[2] &= DFT_2\{1,0\}[0] - e^{-j\pi 0/2} DFT_2\{1,0\}[0] \\
X[3] &= DFT_2\{1,0\}[1] - e^{-j\pi 1/2} DFT_2\{1,0\}[1]
\end{aligned}
$$

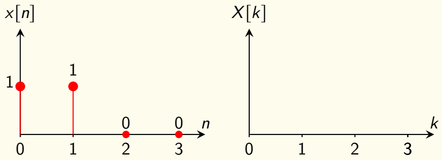
Example - Short-length DFT



Suppose we have a signal $x[n] = \{1, 1, 0, 0\}$

$$
\begin{aligned}
X[0] &= (DFT_1\{1\}[0] + e^{-j\pi 0} DFT_1\{0\}[0]) + e^{-j\pi 0/2}(DFT_1\{1\}[0] + e^{-j\pi 0} DFT_1\{0\}[0]) \\
X[1] &= DFT_2\{1,0\}[1] + e^{-j\pi 1/2} DFT_2\{1,0\}[1] \\
X[2] &= (DFT_1\{1\}[0] + e^{-j\pi 0} DFT_1\{0\}[0]) - e^{-j\pi 0/2}(DFT_1\{1\}[0] + e^{-j\pi 0} DFT_1\{0\}[0]) \\
X[3] &= DFT_2\{1,0\}[1] - e^{-j\pi 1/2} DFT_2\{1,0\}[1]
\end{aligned}
$$

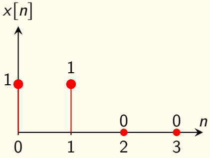
Example - Short-length DFT

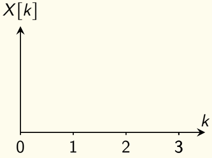


Suppose we have a signal $x[n] = \{1, 1, 0, 0\}$

$$
\begin{aligned}
X[0] &= (DFT_1\{1\}[0] + e^{-j\pi 0} DFT_1\{0\}[0]) + e^{-j\pi 0/2}(DFT_1\{1\}[0] + e^{-j\pi 0} DFT_1\{0\}[0]) \\
X[1] &= (DFT_1\{1\}[1] + e^{-j\pi 1} DFT_1\{0\}[1]) + e^{-j\pi 1/2}(DFT_1\{1\}[1] + e^{-j\pi 1} DFT_1\{0\}[1]) \\
X[2] &= (DFT_1\{1\}[0] + e^{-j\pi 0} DFT_1\{0\}[0]) - e^{-j\pi 0/2}(DFT_1\{1\}[0] + e^{-j\pi 0} DFT_1\{0\}[0]) \\
X[3] &= (DFT_1\{1\}[1] + e^{-j\pi 1} DFT_1\{0\}[1]) - e^{-j\pi 1/2}(DFT_1\{1\}[1] + e^{-j\pi 1} DFT_1\{0\}[1])
\end{aligned}
$$

Example - Short-length DFT

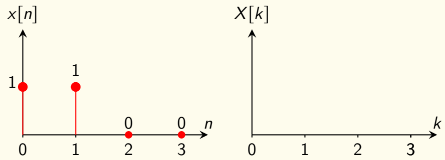




Suppose we have a signal $x[n] = \{1, 1, 0, 0\}$

$$
\begin{aligned}
X[0] &= (DFT_1\{1\}[0] + e^{-j\pi 0} DFT_1\{0\}[0]) + e^{-j\pi 0/2}(DFT_1\{1\}[0] + e^{-j\pi 0} DFT_1\{0\}[0]) \\
X[1] &= (DFT_1\{1\}[0] - e^{-j\pi 0} DFT_1\{0\}[0]) + e^{-j\pi 1/2}(DFT_1\{1\}[0] - e^{-j\pi 0} DFT_1\{0\}[0]) \\
X[2] &= (DFT_1\{1\}[0] + e^{-j\pi 0} DFT_1\{0\}[0]) - e^{-j\pi 0/2}(DFT_1\{1\}[0] + e^{-j\pi 0} DFT_1\{0\}[0]) \\
X[3] &= (DFT_1\{1\}[0] - e^{-j\pi 0} DFT_1\{0\}[0]) - e^{-j\pi 1/2}(DFT_1\{1\}[0] - e^{-j\pi 0} DFT_1\{0\}[0])
\end{aligned}
$$

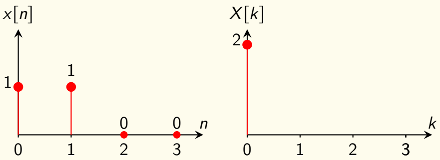
Example - Short-length DFT



Suppose we have a signal $x[n] = \{1, 1, 0, 0\}$

$$
\begin{aligned}
X[0] &= (1 + 1 \cdot 0) + 1(1 + 1 \cdot 0) \\
X[1] &= (1 - 1 \cdot 0) + e^{-j\pi/2}(1 - 1 \cdot 0) \\
X[2] &= (1 + 1 \cdot 0) - 1(1 + 1 \cdot 0) \\
X[3] &= (1 - 1 \cdot 0) - e^{-j\pi/2}(1 - 1 \cdot 0)
\end{aligned}
$$

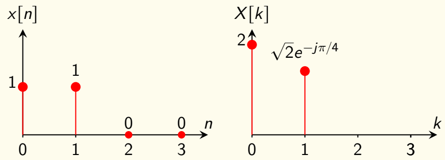
Example - Short-length DFT



Suppose we have a signal $x[n] = \{1, 1, 0, 0\}$

$$\begin{aligned}
X[0] &= (1 + 1 \cdot 0) + 1(1 + 1 \cdot 0) = 2 \\
X[1] &= (1 - 1 \cdot 0) + e^{-j\pi/2}(1 - 1 \cdot 0) \\
X[2] &= (1 + 1 \cdot 0) - 1(1 + 1 \cdot 0) \\
X[3] &= (1 - 1 \cdot 0) - e^{-j\pi/2}(1 - 1 \cdot 0)
\end{aligned}$$

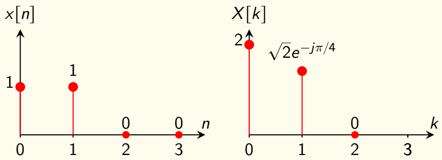
Example - Short-length DFT



Suppose we have a signal $x[n] = \{1, 1, 0, 0\}$

$$
\begin{aligned}
X[0] &= (1 + 1 \cdot 0) + 1(1 + 1 \cdot 0) = 2 \\
X[1] &= (1 - 1 \cdot 0) + e^{-j\pi/2}(1 - 1 \cdot 0) = 1 - j \\
X[2] &= (1 + 1 \cdot 0) - 1(1 + 1 \cdot 0) \\
X[3] &= (1 - 1 \cdot 0) - e^{-j\pi/2}(1 - 1 \cdot 0)
\end{aligned}
$$

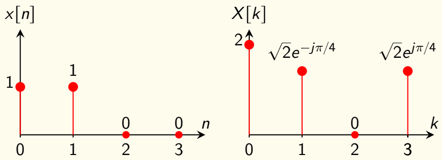
Example - Short-length DFT



Suppose we have a signal $x[n] = \{1, 1, 0, 0\}$

$$\begin{aligned}
X[0] &= (1 + 1 \cdot 0) + 1(1 + 1 \cdot 0) = 2 \\
X[1] &= (1 - 1 \cdot 0) + e^{-j\pi/2}(1 - 1 \cdot 0) = 1 - j \\
X[2] &= (1 + 1 \cdot 0) - 1(1 + 1 \cdot 0) = 0 \\
X[3] &= (1 - 1 \cdot 0) - e^{-j\pi/2}(1 - 1 \cdot 0)
\end{aligned}$$

Example - Short-length DFT



Suppose we have a signal $x[n] = \{1, 1, 0, 0\}$

$$
\begin{aligned}
X[0] &= (1 + 1 \cdot 0) + 1(1 + 1 \cdot 0) = 2 \\
X[1] &= (1 - 1 \cdot 0) + e^{-j\pi/2}(1 - 1 \cdot 0) = 1 - j \\
X[2] &= (1 + 1 \cdot 0) - 1(1 + 1 \cdot 0) = 0 \\
X[3] &= (1 - 1 \cdot 0) - e^{-j\pi/2}(1 - 1 \cdot 0) = 1 + j
\end{aligned}
$$

Time complexity

$$DFT_N\{x[n]\} = \sum_{n=0}^{N-1} x[n]e^{-j(2\pi/N)kn} \tag{6}$$

Naive implementation has time complexity $O(N^2)$

- We sum over N constant time operations per component.

Time complexity

$$DFT_N\{x[n]\} = \sum_{n=0}^{N-1} x[n]e^{-j(2\pi/N)kn} \tag{6}$$

Naive implementation has time complexity $O(N^2)$

- We sum over N constant time operations per component.

$$DFT_N\{x[n]\} = DFT_{N/2}\{x_{even}[n]\} + e^{-j(2\pi/N)k} \cdot DFT_{N/2}\{x_{odd}[n]\} \tag{7}$$

Fast Fourier transform (FFT) has time complexity $O(N\log(N))$

- Each depth level executes in constant time $O(1)$
- There are at most $\log_2(N)$ levels
- There are N components

Convolution

Suppose we have a filter $h[n]$ of length N and want to calculate

$$y[n] = h[n] * x[n] = \sum_{k=0}^{N} h[n]x[n-k]. \tag{8}$$

Convolution

Suppose we have a filter $h[n]$ of length N and want to calculate

$$y[n] = h[n] * x[n] = \sum_{k=0}^{N} h[n]x[n-k]. \tag{8}$$

- Naive implementation has time complexity $O(N^2)$.

Convolution

Suppose we have a filter $h[n]$ of length N and want to calculate

$$y[n] = h[n] * x[n] = \sum_{k=0}^{N} h[n]x[n-k]. \tag{8}$$

- Naive implementation has time complexity $O(N^2)$.
- Time domain convolution is frequency domain multiplication.

$$y[n] = IDFT\{DFT\{h[n]\} \cdot DFT\{x[n]\}\} \tag{9}$$

Convolution

Suppose we have a filter $h[n]$ of length N and want to calculate

$$y[n] = h[n] * x[n] = \sum_{k=0}^{N} h[n]x[n-k]. \tag{8}$$

- Naive implementation has time complexity $O(N^2)$.
- Time domain convolution is frequency domain multiplication.

$$y[n] = IDFT\{DFT\{h[n]\} \cdot DFT\{x[n]\}\} \tag{9}$$

- FFT has time complexity $O(N \log(N))$.

Convolution

Suppose we have a filter $h[n]$ of length N and want to calculate

$$y[n] = h[n] * x[n] = \sum_{k=0}^{N} h[n]x[n-k]. \tag{8}$$

- Naive implementation has time complexity $O(N^2)$.
- Time domain convolution is frequency domain multiplication.

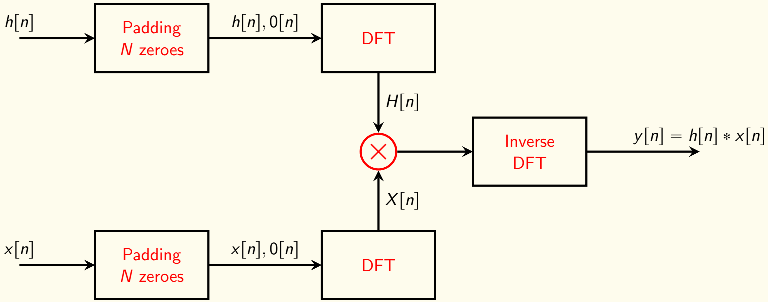
$$y[n] = IDFT\{DFT\{h[n]\} \cdot DFT\{x[n]\}\} \tag{9}$$

- FFT has time complexity $O(N\log(N))$.

Remark

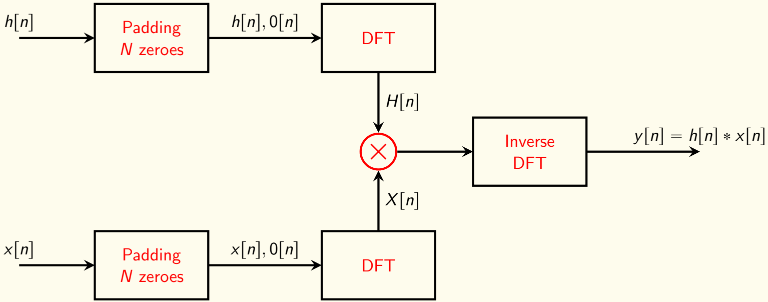
For large filters $h[n]$, **it is faster to do convolution through the frequency domain**.

Convolution



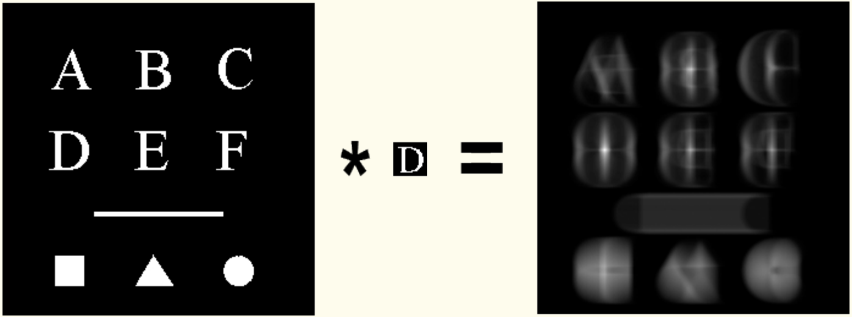
- Convolution can be performed as multiplication in the frequency domain.

Convolution



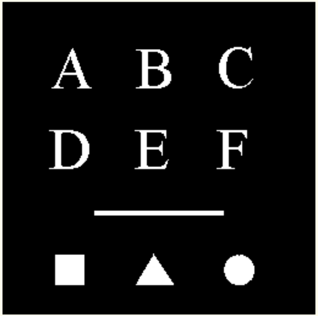
- Convolution can be performed as multiplication in the frequency domain.
- Zero padding is used to avoid periodicity of the FFT. Additional padding makes sure that the length is a power of 2.

2D correlator



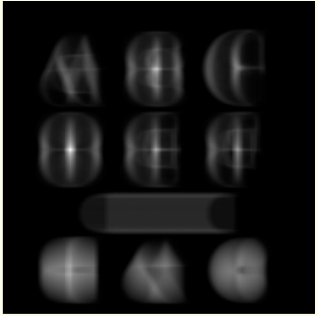
Convolution (or correlation) can be used as pattern matching.

2D correlator



$*$

$=$



Convolution (or correlation) can be used as pattern matching.

- Brighter white indicates a higher correlation between input x and template filter h.
- Highest brightness is achieved when input x is maximal.
- A white square gives at least as good a response as a match.

Pearson correlator (1D representation)

A correlator is sensitive to the magnitude of the input.

- **Solution:** scale the input

Pearson correlator (1D representation)

A correlator is sensitive to the magnitude of the input.

- **Solution:** scale the input
- For example through Pearson correlation.

$$
\begin{aligned}
y[n] &= \frac{\sum_k (x[n+k]-\bar{x})(h[k]-\bar{h})}{\sqrt{\sum_k (x[n+k]-\bar{x})^2} \cdot \sqrt{\sum_k (h[k]-\bar{h})^2}} \\
&= \frac{L(\sum_k x[n+k]h[k]) - (\sum_k x[n+k])(\sum_k h[k])}{\sqrt{L\sum_k (x[n+k])^2 - (\sum_k x[n+k])^2} \cdot \sqrt{L\sum_k (h[k])^2 - (\sum_k h[k])^2}}
\end{aligned}
$$

where L is the length of the filter.

Pearson correlator (2D representation)

A correlator is sensitive to the magnitude of the input.

- **Solution:** scale the input

Pearson correlator (2D representation)

A correlator is sensitive to the magnitude of the input.

- **Solution:** scale the input
- For example through Pearson correlation.

$$
\begin{aligned}
y[n,m] &= \frac{\sum_i \sum_j (x[n+i,m+j]-\bar{x})(h[i,j]-\bar{h})}{\sqrt{\sum_i \sum_j (x[n+i,m+j]-\bar{x})^2}\cdot\sqrt{\sum_i \sum_j (h[i,j]-\bar{h})^2}} \\
&= \frac{L^2 \sum_i \sum_j x[n+i,m+j]h[i,j] - (\sum_i \sum_j x[n+i,m+j])(\sum_i \sum_j h[i,j])}{\sqrt{L^2 \sum_i \sum_j (x[n+i,m+j])^2 - (\sum_i \sum_j x[n+i,m+j])^2}\cdot\sqrt{L^2 \sum_i \sum_j (h[i,j])^2 - (\sum_i \sum_j h[i,j])^2}}
\end{aligned}
$$

where L^2 is the number of elements in the (square) filter. Note this **Pearson correlator** is not a linear filter.

Table of Contents

Rotated patterns

Figure: Image by Pavel Torgashov, Code Project

- When searching for patterns, patterns may not appear upright
- How do we account for rotated patterns?

(Freeman) chain coding

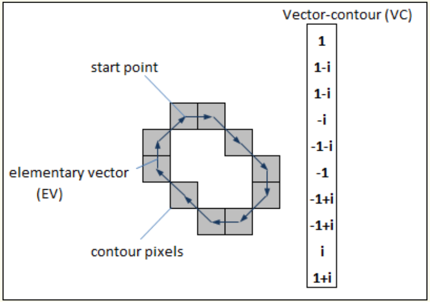
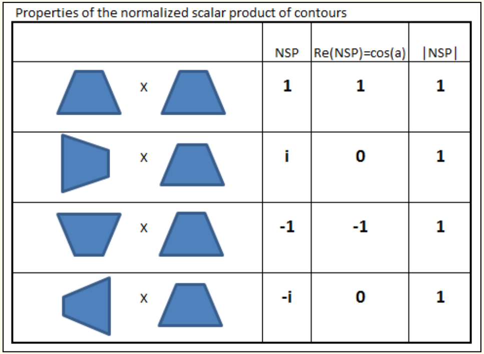


Figure: Image by Pavel Torgashov, Code Project

Define objects by their contours

- A contour is a periodic description
- Each point describes the angle of the next point
- Objects to recognize can be described by their contour

(Freeman) chain coding



Properties of the normalized scalar product of contours

	NSP	Re(NSP)=cos(a)	|NSP|
x	**1**	**1**	**1**
x	**i**	**0**	**1**
x	**-1**	**-1**	**1**
x	**-i**	**0**	**1**

Figure: Image by Pavel Torgashov, Code Project

Detecting a shape is implemented as convolution of the input with a filter contour

Fourier descriptors

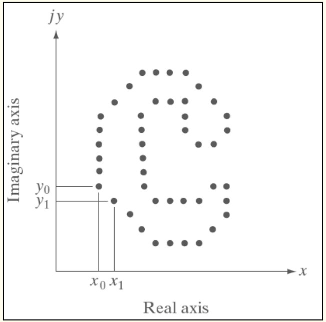


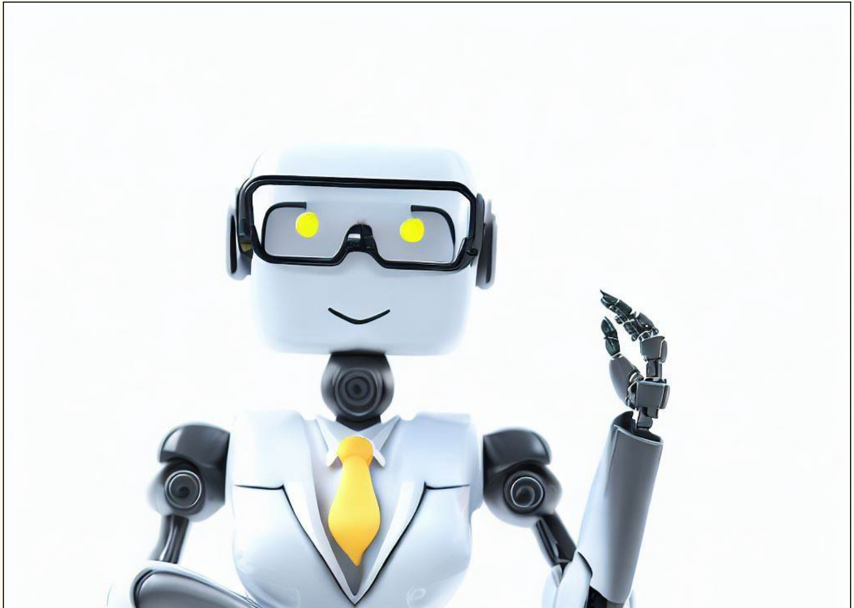
Figure: Image by Shahram Ebadollahi, Columbia University

Alternatively, take the DFT of contour coordinates

- Interpret the vertical axis as the complex axis

Table of Contents

Let us wrap up the lecture!



Take-home Messages

- From previous lectures, we have learned that analyzing signals and systems in the frequency domain is easier.
- The DTF is a tool to move signals in the frequency domain.
- The IDFT is the inverse operation, i.e., moving signals back to the time domain.

Practice Questions

The following questions might appear in the final exam:

- What is the use of the DFT?
- Compute the frequency-domain representation of the signal $x[n] = \{-1, 0, 1, 0, 1\}$.
- What is the principal frequency interval in the frequency domain?
- How can we recover a real-valued signal from its DFT?
- What is the key difference between the DFT and the FFT?

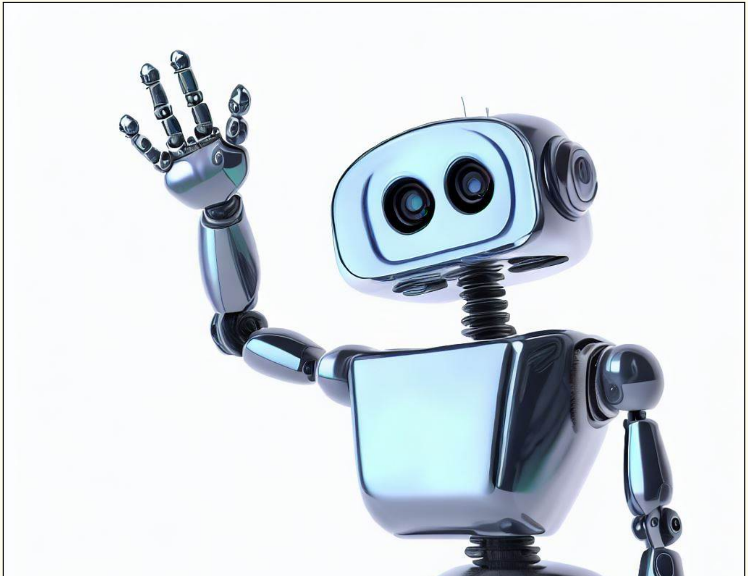
Tutorial exercises

There are no assigned exercises.
We will discuss several applications of the DFT in the next tutorial.

Let us discuss a simpler way to analyze discrete signals and systems...

The Z-Transform

Have a nice day!



Acknowledgements

The material for this lecture series was developed by dr. Arnold Meijster and dr. Harmen de Weerd and modified by Juan Diego Cardenas-Cartagena.

Disclaimer

- Grammar was checked with Grammarly and Grammar checker GPT.
- Images without source were created with the assistance of DALL·E.