

Digital Signal Processing

Chapter IV: The Fourier Transform and its applications

IV.1 Vector spaces of signals (crash course)

- ▶ **Vector space** = a set $V\{v_i\}$ with the following two properties:
 - ▶ one element + another element = still an element of the same space
 - ▶ a scalar constant \times an element = still an element of the same space
- ▶ You **can't escape** a vector space by summing or scaling
- ▶ The elements of a vector space are called **vectors**

Examples of vector spaces

- ▶ Geometric spaces are great intuitive examples:
 - ▶ a line, or the set \mathbb{R} (one-dimensional)
 - ▶ a plane, or the set \mathbb{C} (two-dimensional)
 - ▶ 3D space (three-dimensional)
 - ▶ 4D space (four-dimensional, like the spatio-temporal universe)
 - ▶ arrays with N numbers (N -dimensional)
 - ▶ space of continuous signals (∞ -dimensional)
- ▶ The **dimension** of the space = “how many numbers you need in order to specify one element” (informal)
- ▶ A “vector” like in maths = a sequence of N numbers = a “vector” like in programming
 - ▶ e.g. a point in a plane has two coordinates = a vector of size $N = 2$
 - ▶ e.g. a point in a 3D-space has three coordinates = a vector of size $N = 3$

- ▶ Many vector spaces have a fundamental operation: **the (Euclidean) inner product**

- ▶ for **discrete** signals:

$$\langle \mathbf{x}, \mathbf{y} \rangle = \sum_i x_i y_i^*$$

- ▶ for **continuous** signals:

$$\langle \mathbf{x}, \mathbf{y} \rangle = \int x(t) y^*(t)$$

- ▶ * represents **complex conjugate** (has no effect for real signals)
- ▶ The result is one number (real or complex)
- ▶ Also known as **dot product** or **scalar product** ("produs scalar")

Inner product

- ▶ Each entry in x times the complex conjugate of the one in y , all summed
- ▶ For discrete signals, it can be understood as a row \times column multiplication
- ▶ Discrete vs continuous: just change sum/integral depending on signal type

Inner product properties

- ▶ Inner product is **linear** in both terms:

$$\langle \mathbf{x}_1 + \mathbf{x}_2, \mathbf{y} \rangle = \langle \mathbf{x}_1, \mathbf{y} \rangle + \langle \mathbf{x}_2, \mathbf{y} \rangle$$

$$\langle c \cdot \mathbf{x}, \mathbf{y} \rangle = c \cdot \langle \mathbf{x}_1, \mathbf{y} \rangle$$

$$\langle \mathbf{x}, \mathbf{y}_1 + \mathbf{y}_2 \rangle = \langle \mathbf{x}, \mathbf{y}_1 \rangle + \langle \mathbf{x}, \mathbf{y}_2 \rangle$$

$$\langle \mathbf{x}, c \cdot \mathbf{y} \rangle = c^* \cdot \langle \mathbf{x}_1, \mathbf{y} \rangle$$

The distance between two vectors

- ▶ An inner product induces a **norm** and a **distance** function
- ▶ **The (Euclidean) distance** between two vectors =

$$d(\mathbf{x}, \mathbf{y}) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + \dots + (x_N - y_N)^2}$$

- ▶ This distance is the **usual geometric distance** you know from geometry
- ▶ It has the exact same intuition like in **normal geometry**:
 - ▶ if two vectors have small distance, they are close, they are similar
 - ▶ two vectors with large distance are far away, not similar
 - ▶ two identical vectors have zero distance

The norm of a vector

- ▶ An inner product induces a **norm** and a **distance** function
- ▶ The **norm** (length) of a vector = $\text{sqrt}(\text{inner product with itself})$

$$\|\mathbf{x}\| = \sqrt{\langle \mathbf{x}, \mathbf{x} \rangle} = \sqrt{x_1^2 + x_2^2 + \dots + x_N^2}$$

- ▶ The **norm** of a vector is the distance from \mathbf{x} to point $\mathbf{0}$.
- ▶ It has the exact same intuition like in **normal geometry**:
 - ▶ vector has large norm = has big values, is far from $\mathbf{0}$
 - ▶ vector has small norm = has small values, is close to $\mathbf{0}$
 - ▶ vector has zero norm = it is the vector $\mathbf{0}$
- ▶ Norm of a vector = $\text{sqrt}(\text{the signal energy})$

Norm and distance

- ▶ The norm and distance are related
- ▶ The distance between **a** and **b** = norm (length) of their difference

$$d(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\| = \sqrt{x_1^2 + x_2^2 + \dots + x_N^2}$$

- ▶ Just like in geometry: distance = length of the difference vector

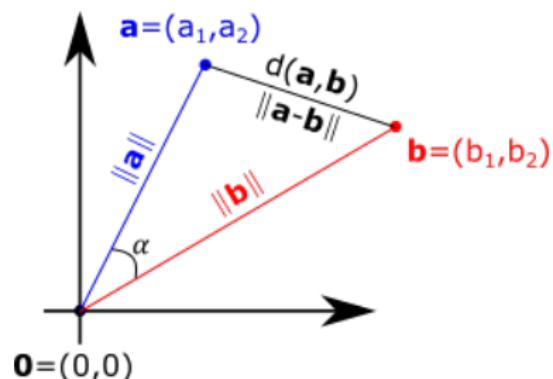


Figure 1: Norm and distance in vector spaces

Angle between vectors

- ▶ The **angle** between two vectors is:

$$\cos(\alpha) = \frac{\langle x, y \rangle}{\|x\| \cdot \|y\|}$$

- ▶ is a value between -1 and 1
- ▶ **Otrhogonal vectors** = two vectors with $\langle x, y \rangle = 0$
 - ▶ their angle = 90 deg
 - ▶ in geometric language, the two vectors are **perpendicular**

Why vector space

- ▶ Why are all these useful?
- ▶ They are a very general **framework** for different kinds of signals
- ▶ We can have **generic** algorithms expressed in terms of distances, norms, angles, and they will work the same in all vector spaces
 - ▶ Example in DEDP class: ML decision with 1, 2, N samples

We deal mainly with the following vector spaces:

- ▶ The vector space of all infinitely-long real signals $x[n]$
- ▶ The vector space of all infinitely-long periodic signals $x[n]$ with period N
 - ▶ for each N we have a different vector space
- ▶ The vector space of all finite-length signals $x[n]$ with only N samples
 - ▶ for each N we have a different vector space

- ▶ A **basis** = a set of N linear independent elements from a vector space

$$B = \{\mathbf{b}^1, \mathbf{b}^2, \dots, \mathbf{b}^N\}$$

- ▶ Any vector in a vector space is expressed as a **linear combination** of the basis elements:

$$\mathbf{x} = \alpha_1 \mathbf{b}^1 + \alpha_2 \mathbf{b}^2 + \dots + \alpha_N \mathbf{b}^N$$

- ▶ The vector is defined by these coefficients:

$$\mathbf{x} = (\alpha_1, \alpha_2, \dots, \alpha_N)$$

Bases and coordinate systems

- ▶ Bases are just like **coordinate systems** in a geometric space

- ▶ any point is expressed w.r.t. a coordinate system

$$\mathbf{x} = x_1 \mathbf{i} + x_2 \mathbf{j}$$

- ▶ any vector is expressed w.r.t. a basis

$$\mathbf{x} = \alpha_1 \mathbf{b}^1 + \alpha_2 \mathbf{b}^2 + \cdots + \alpha_N \mathbf{b}^N$$

- ▶ N = The number of basis elements = The dimension of the space
- ▶ Example: any color = RGB values (monitor) or Cyan-Yellow-Magenta values (printer)

Bases and coordinate systems

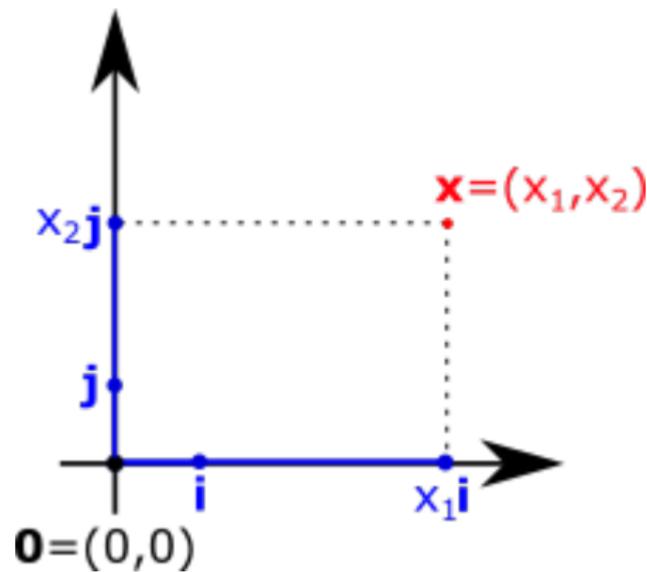


Figure 2: Basis expansion of a vector \mathbf{x}

Choice of bases

- ▶ There is typically an infinite choice of bases
- ▶ The **canonical basis** = all basis vectors are full of zeros, just with one 1
- ▶ You used it already in an exercise:
 - ▶ any signal $x[n]$ can be expressed of a sum of $\delta[n - k]$

$$\{\dots, 3, 6, 2, \dots\} = \dots + 3\delta[n] + 6\delta[n - 1] + 2\delta[n - 2] + \dots$$

- ▶ the canonical basis is $B = \{\dots, \delta[n], \delta[n - 1], \delta[n - 2], \dots\}$

Orthonormal bases

- ▶ An **orthonormal basis** a basis where all elements \mathbf{b}^i are:

- ▶ orthogonal to each other:

$$\langle \mathbf{b}^i, \mathbf{b}^j \rangle = 0, \forall i \neq j$$

- ▶ **normalized** (their norm = 1):

$$||\mathbf{b}^i|| = \sqrt{\langle \mathbf{b}^i, \mathbf{b}^i \rangle} = 1, \forall i$$

- ▶ Example: the canonical basis $\{\delta[n - k]\}$ is orthonormal:

- ▶ $\langle \delta[n - k], \delta[n - l] \rangle = 0, \forall k \neq l$
 - ▶ $\langle \delta[n - k], \delta[n - k] \rangle = 1, \forall k$

Orthonormal bases

- Orthonormal basis = like a coordinate system with orthogonal vectors, of length 1

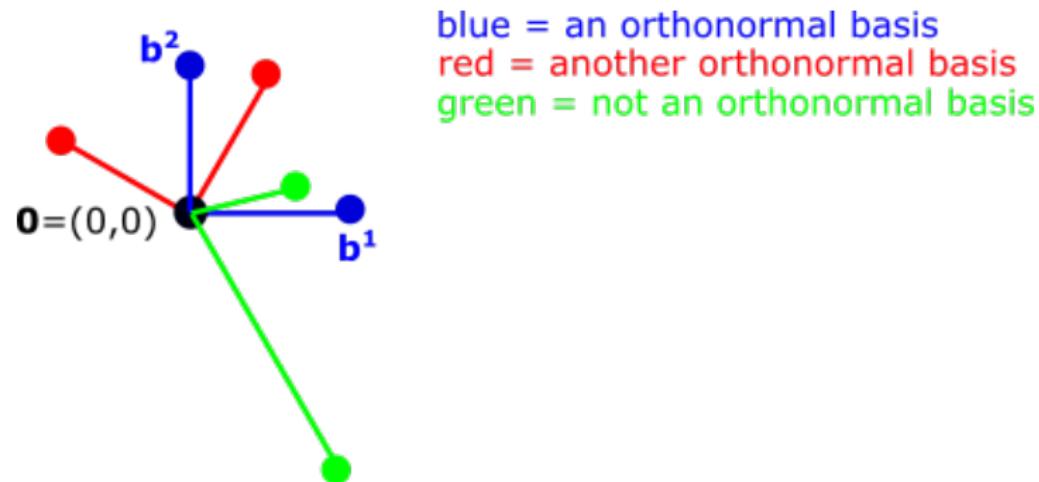


Figure 3: Sample bases in a 2D space

Basis expansion of a vector

- ▶ Suppose we have an **orthonormal basis** $B = \{\mathbf{b}^i\}$
- ▶ Suppose we have a vector \mathbf{x}
- ▶ We can write (expand) \mathbf{x} as:

$$\mathbf{x} = \alpha_1 \mathbf{b}^1 + \alpha_2 \mathbf{b}^2 + \cdots + \alpha_N \mathbf{b}^N$$

- ▶ Question: how to **find** the coefficients α_i ?

Basis expansion of a vector

- If the basis is **orthonormal**, we have:

$$\begin{aligned}\langle \mathbf{x}, \mathbf{b}^i \rangle &= \langle \alpha_1 \mathbf{b}^1 + \alpha_2 \mathbf{b}^2 + \cdots + \alpha_N \mathbf{b}^N, \mathbf{b}^i \rangle \\&= \langle \alpha_1 \mathbf{b}^1, \mathbf{b}^i \rangle + \langle \alpha_2 \mathbf{b}^2, \mathbf{b}^i \rangle + \cdots + \langle \alpha_N \mathbf{b}^N, \mathbf{b}^i \rangle \\&= \alpha_1 \langle \mathbf{b}^1, \mathbf{b}^i \rangle + \alpha_2 \langle \mathbf{b}^2, \mathbf{b}^i \rangle + \cdots + \alpha_N \langle \mathbf{b}^N, \mathbf{b}^i \rangle \\&= \alpha_i\end{aligned}$$

Basis expansion of a vector

- ▶ Any vector \mathbf{x} can be written as:

$$\mathbf{x} = \alpha_1 \mathbf{b}^1 + \alpha_2 \mathbf{b}^2 + \cdots + \alpha_N \mathbf{b}^N$$

- ▶ For orthonormal basis: the coefficients α_i are found by inner product with the corresponding basis vector:

$$\alpha_i = \langle \mathbf{x}, \mathbf{b}^i \rangle$$

Why bases

- ▶ How does all this talk about bases help us?
- ▶ To better understand the Fourier transform
- ▶ The signals $\{e^{j\omega n}\}$ form an **orthonormal basis**
- ▶ The Fourier Transform of a signal x = finding the coefficients of x in this basis
- ▶ The Inverse Fourier Transform = expanding x with the elements of this basis
- ▶ Same **generic** thing every time, only the type of signals differ

IV.2 Introducing the Fourier Transforms

Reminder

► Reminder:

$$e^{jx} = \cos(x) + j \sin(x)$$

$$\cos(x) = \frac{e^{jx} + e^{-jx}}{2}$$

$$\sin(x) = \frac{e^{jx} - e^{-jx}}{2j}$$

$$\sin(x) = \cos\left(x - \frac{\pi}{2}\right)$$

$$\cos(x) = \sin\left(x + \frac{\pi}{2}\right)$$

Why sinusoidal signals

- ▶ Why are sinusoidal signals $\sin()$ and $\cos()$ **so prevalent** in signal processing?
- ▶ Answer: because they are combinations of an e^{jx} and an e^{-jx}
- ▶ Why are these e^{jx} so special?
- ▶ Answer: because they are **eigen-functions** of linear and time-invariant (LTI) systems

Response of LTI systems to harmonic signals

- ▶ Consider an LTI system with $h[n]$
- ▶ Input signal = complex harmonic (exponential) signal $x[n] = Ae^{j\omega_0 n}$
- ▶ Output signal = convolution

$$\begin{aligned}y[n] &= \sum_{k=-\infty}^{\infty} h[k]x[n-k] \\&= \sum_{k=-\infty}^{\infty} h[k]e^{-j\omega_0 k} Ae^{j\omega_0 n} \\&= H(\omega_0) \cdot x[n]\end{aligned}$$

- ▶ Output signal = input signal \times a (complex) constant ($H(\omega_0)$)

- ▶ **Eigen-function** of a system ("funcție proprie") = a function f which, if input in a system, produces an output proportional to it

$$H\{f\} = \lambda \cdot f, \lambda \in \mathbb{C}$$

- ▶ just like **eigen-vectors** of a matrix (remember algebra): $A\tilde{\tilde{x}} = \lambda\tilde{\tilde{x}}$
- ▶ we call the "functions" to allow for continuous signals as well
- ▶ Complex exponential signals $e^{j\omega t}$ (or $e^{j\omega n}$) are **eigen-functions** of Linear and Time Invariant (LTI) systems:
 - ▶ output signal = input signal \times a (complex) constant

Representation with respect to eigen-functions (-vectors)

- ▶ We can understand the effect of a LTI system very easily if we **decompose all signals $x[n]$ as a combination of $\{e^{j\omega n}\}$**
- ▶ Example: RGB color filter
 - ▶ suppose we have some photographic filters (lenses):
 - ▶ one reduces red to 50%
 - ▶ one reduces green to 25%
 - ▶ one reduces blue to 80%
 - ▶ RGB are eigen-functions of the system: input = 200 Blue, output = $0.8 * 200$ Blue
 - ▶ what is the output color if input is “pink”?
 - ▶ Answer is easy if we represent all colors in RGB

Representation with respect to eigen-functions (-vectors)

- ▶ We can understand the effect of a LTI system **very easily** if we decompose all signals as a combination of $\{e^{j\omega n}\}$
- ▶ All vector space theory becomes useful now:
 - ▶ $\{e^{j\omega n}\}$ is an **orthonormal basis**
 - ▶ decomposing signals = finding coefficients α_i ,
 - ▶ we know how to do this, just like for any orthonormal basis

$$x[n] = \sum \alpha_\omega \cdot e^{j\omega n}$$

$$\alpha_\omega = \langle x, e^{j\omega n} \rangle$$

Discrete-Time Fourier Transform (DTFT)

- ▶ Consider the vector space of **non-periodic infinitely-long signals**
- ▶ This vector space is **infinite-dimensional**
- ▶ The signals $\{e^{j2\pi fn}\}, \forall f \in [-\frac{1}{2}, \frac{1}{2}]$ form an **orthonormal basis**
- ▶ We can expand (almost) any x in this basis:

$$x[n] = \int_{f=-1/2}^{1/2} \underbrace{X(f)}_{\alpha_\omega} e^{j2\pi fn} df$$

- ▶ The coefficient of every $e^{j2\pi fn}$ is found by inner product:

$$\alpha_\omega = X(f) = \langle x[n], e^{j2\pi fn} \rangle = \sum_n x[n] e^{-j2\pi fn}$$

Discrete-Time Fourier Transform (DTFT)

Inverse Discrete-Time Fourier Transform (DTFT)

$$x[n] = \int_{f=-1/2}^{1/2} X(f) e^{j2\pi fn} df$$

- ▶ A signal $x[n]$ can be written as a linear combination of $\{e^{j2\pi fn}\}$, $\forall f \in [-\frac{1}{2}, \frac{1}{2}]$, with some coefficients $X(f)$

Discrete-Time Fourier Transform (DTFT)

$$X(f) = \langle x[n], e^{j2\pi fn} \rangle = \sum_{n=-\infty}^{\infty} x[n] e^{-j2\pi fn}$$

- ▶ The coefficient $X(f)$ of every $\{e^{j2\pi fn}\}$ is found using the inner product $\langle x, e^{j2\pi fn} \rangle$

Discrete-Time Fourier Transform (DTFT)

- ▶ Alternative form with ω
- ▶ We can replace $2\pi f = \omega$, and $df = \frac{1}{2\pi}d\omega$

$$x[n] = \frac{1}{2\pi} \int_{\omega=-\pi}^{\pi} X(\omega) e^{j\omega n} d\omega$$

$$X(\omega) = \langle x[n], e^{j\omega n} \rangle = \sum_n x[n] e^{-j\omega n}$$

Discrete-Time Fourier Transform (DTFT)

- ▶ A non-periodic signal $x[n]$ has a **continuous spectrum** $X(\omega)$, with
 $f \in [-\frac{1}{2}, \frac{1}{2}]$
 - ▶ e.g. $\omega \in [-\pi, \pi]$

Discrete Fourier Transform (DFT)

- ▶ Consider the vector space of **periodic** signals with **period N**
 - ▶ for some fixed $N = 2, 3$ or ... etc
- ▶ This is a vector space of **dimension N**
 - ▶ we need N numbers to identify a signal (specify its period)
- ▶ We can consider $x[n]$ only for **one period**, i.e. $n = 0, \dots, N - 1$
- ▶ The signals $\{e^{j2\pi f n}\}, \forall f \in \{0, \frac{1}{N}, \dots, \frac{N-1}{N}\}$ form an **orthonormal basis** with N elements
- ▶ It is a **discrete** set of frequencies: $f = \frac{k}{N}, \forall k \in \{0, 1, \dots, N - 1\}$

Discrete Fourier Transform (DFT)

Inverse Discrete Fourier Transform

$$x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X_k e^{j2\pi kn/N}$$

- ▶ A periodic signal $x[n]$ can be written as a linear combination of k signals $\{e^{j2\pi kn/N}\}$, with some coefficients X_k

Discrete Fourier Transform

$$X_k = \langle x[n], e^{j2\pi fn} \rangle = \sum_{n=0}^{N-1} x[n] e^{-j2\pi kn/N}$$

- ▶ The coefficient $X(f)$ of every $\{e^{j2\pi fn}\}$ is found using the inner product $\langle x, e^{j2\pi fn} \rangle$

Discrete Fourier Transform (DFT)

- ▶ A periodic signal $x[n]$ with period N has a **discrete spectrum** $X(\omega)$ composed of only N frequencies $\{0, \frac{1}{N}, \dots, \frac{N-1}{N}\}$
- ▶ Each frequency $\frac{k}{N}$ has a **coefficient** X_k
 - ▶ also written as c_k
 - ▶ The N coefficients X_k are the equivalent of $X(\omega)$
- ▶ It is also known as the “Fourier Series for Discrete Signals”

IV.3 The Discrete-Time Fourier Transform (DTFT)

Definition

Definitions (again):

Inverse Discrete-Time Fourier Transform (DTFT)

$$x[n] = \int_{f=-1/2}^{1/2} X(f) e^{j2\pi f n} df = \frac{1}{2\pi} \int_{\omega=-\pi}^{\pi} X(\omega) e^{j\omega n} d\omega$$

Discrete-Time Fourier Transform (DTFT)

$$X(f) = \langle x[n], e^{j2\pi f n} \rangle = \sum_{n=-\infty}^{\infty} x[n] e^{-j2\pi f n}$$

Basic properties of DTFT

- ▶ $X(\omega)$ is **defined** only for $\omega \in [-\pi, \pi]$
 - ▶ or $f \in [-\frac{1}{2}, \frac{1}{2}]$
- ▶ $X(\omega)$ is **complex** (has $|X(\omega)|$, $\angle X(\omega)$)
- ▶ If the signal $x[n]$ is real, $X(\omega)$ is **even**

$$x[n] \in \mathbb{R} \rightarrow X(-\omega) = X^*(\omega)$$

- ▶ This means:
 - ▶ modulus is even: $|X(\omega)| = |X(-\omega)|$
 - ▶ phase is odd: $X(\omega) = -X(-\omega)$

Expressing as sum of sinusoids

- Grouping terms with $e^{j\omega n}$ and $e^{j(-\omega)n}$ we get:

$$\begin{aligned}x[n] &= \frac{1}{2\pi} \int_{-\pi}^0 X(\omega) e^{j\omega n} d\omega + \frac{1}{2\pi} \int_0^\pi X(\omega) e^{j\omega n} d\omega \\&= \frac{1}{2\pi} \int_0^\pi (X(\omega) e^{j\omega n} + X(-\omega) e^{j(-\omega)n}) d\omega \\&= \frac{1}{2\pi} \int_0^\pi 2|X(\omega)| (e^{j\omega n + \angle X(\omega)} + e^{-j\omega n - \angle X(\omega)}) d\omega \\&= \frac{1}{2\pi} \int_0^\pi 2|X(\omega)| \cos(\omega n + \angle X(\omega)) d\omega\end{aligned}$$

- Any signal $x[n]$ is a **sum of sinusoids with all frequencies** $f \in [0, \frac{1}{2}]$,
or $\omega \in [0, \pi]$

Expressing as sum of sinusoids

- ▶ The DTFT shows that any signal $x[n]$ is a “**sum**” of sinusoids with all frequencies $f \in [0, \frac{1}{2}]$, or $\omega \in [0, \pi]$
 - ▶ this is the fundamental practical interpretation of the Fourier transform
 - ▶ not really a sum, because we have an integral
- ▶ The **modulus** $|X(\omega)|$ gives the **amplitude** of the sinusoids ($\times 2$)
 - ▶ for $\omega = 0$, $|X(\omega = 0)|$ = the DC component
- ▶ The **phase** $\angle X(\omega)$ gives the initial phase

Properties of DTFT

1. Linearity

$$a \cdot x_1[n] + b \cdot x_2[n] \leftrightarrow a \cdot X_1(\omega) + b \cdot X_2(\omega)$$

Proof: via definition

2. Shifting in time

$$x[n - n_0] \leftrightarrow e^{-j\omega n_0} X(\omega)$$

Proof: via definition

- ▶ The amplitudes $|X(\omega)|$ is not affected, shifting in time affects only the phase

Properties of DTFT

3. Modulation in time

$$e^{j\omega_0 n} x[n] \leftrightarrow X(\omega - \omega_0)$$

4. Complex conjugation

$$x^*[n] \leftrightarrow X^*(-\omega)$$

5. Convolution

$$x_1[n] * x_2[n] \leftrightarrow X_1(\omega) \cdot X_2(\omega)$$

- ▶ Not circular convolution, this is the normal convolution

6. Product in time

Product in time \leftrightarrow convolution of Fourier transforms

$$x_1[n] \cdot x_2[n] \leftrightarrow \frac{1}{2\pi} \int_{-\pi}^{\pi} X_1(\lambda)X_2(\omega - \lambda)d\lambda$$

Properties of DTFT

Correlation theorem

$$r_{x_1 x_2}[l] \leftrightarrow X_1(\omega)X_2(-\omega)$$

Wiener Khinchin theorem

Autocorrelation of a signal \leftrightarrow Power spectral density

$$r_{xx}[l] \leftrightarrow S_{xx}(\omega) = |X(\omega)|^2$$

Parseval theorem

- ▶ **Parseval theorem:** energy of the signal is the same in time and frequency domains

$$E = \sum_{-\infty}^{\infty} |x[n]|^2 = \frac{1}{2\pi} \int_{-\pi}^{\pi} |X(\omega)|^2$$

- ▶ Is true for all orthonormal bases

IV.4 The Discrete Fourier Transform (DFT)

Definitions (again):

Inverse Discrete Fourier Transform (DFT)

$$x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X_k e^{j2\pi kn/N}$$

Discrete Fourier Transform (DFT)

$$X_k = \langle x[n], e^{j2\pi fn} \rangle = \sum_{n=0}^{N-1} x[n] e^{-j2\pi kn/N}$$

- ▶ DFT is defined for periodical signals with period N
- ▶ there are exactly N terms in each sum

Definitions

To remember:

- ▶ DFT: takes a vector with N elements ($x[n]$), produces a vector with N elements (X_k)
 - ▶ for this reason, we can compute it e.g. with Matlab
- ▶ DTFT: takes a vector with ∞ elements ($x[n]$), produces a continuous function ($X(\omega)$) between $[-\pi, \pi]$

- ▶ DTFT has only N coefficients X_k , each X_k corresponding to a frequency $f = \frac{k}{N}$
- ▶ In discrete domain, $f = \frac{N-k}{N} = \frac{-k}{N}$ (aliasing, we can always add/subtract 1 from f)
- ▶ So we can consider X_{N-k} as X_{-k} , due to periodicity
- ▶ Example: a signal with period $N = 6$ has 6 DFT coefficients
 - ▶ we can call them $X_0, X_1, X_2, X_3, X_4, X_5$
 - ▶ we have $X_5 = X_{-1}, X_4 = X_{-2}$
 - ▶ we can also call them $X_{-2}, X_{-1}, X_0, X_1, X_2, X_3$

Basic Properties of the DFT

- ▶ Has only N coefficients X_k
- ▶ X_k are **complex** (has $|X_k|$, $\angle X_k$)
- ▶ If the signal $x[n]$ is real, the coefficients are **even**

$$x[n] \in \mathbb{R} \rightarrow X_{-k} = X_k^*$$

- ▶ This means:
 - ▶ modulus is even: $|X_k| = |X_{-k}|$
 - ▶ phase is odd: $\angle X_{-k} = -\angle X_k$

Expressing as sum of sinusoids, N = odd

- ▶ Grouping terms with k and $-k$:
- ▶ If N is odd, we have X_0 and pairs (X_k, X_{-k}) :

$$\begin{aligned}x[n] &= \frac{1}{N}X_0e^{j0n} + \frac{1}{N}\sum_{k=-(N-1)/2}^{-1}X_ke^{j2\pi kn/N} + \frac{1}{N}\sum_{k=1}^{(N-1)/2}X_ke^{j2\pi kn/N} \\&= \frac{1}{N}X_0 + \frac{1}{N}\sum_{k=1}^{(N-1)/2}(X_ke^{j2\pi kn/N} + X_{-k}e^{-j2\pi kn/N}) \\&= \frac{1}{N}X_0 + \frac{1}{N}\sum_{k=1}^{(N-1)/2}|X_k|(e^{j2\pi kn/N+\angle X(k)} + e^{-j2\pi kn/N-\angle X(k)}) \\&= \frac{1}{N}X_0 + \frac{1}{N}\sum_{k=1}^{(N-1)/2}2|X_k|\cos(2\pi k/Nn + \angle X_k)\end{aligned}$$

- ▶ A sum of sinusoids with frequencies up to $1/2$

Expressing as sum of sinusoids, $N = \underline{\text{even}}$

- ▶ If N is even, we have X_0 and pairs (X_k, X_{-k}) , with an extra term $X_{N/2}$ which has no pair
 - ▶ e.g. $N = 6$: $X_{-2}, X_{-1}, X_0, X_1, X_2, X_3$
- ▶ The term with no pair, $X_{N/2}$, must be a real number, because $X_{N/2} = X_{-N/2}^* = XN/2^*$
- ▶ The extra term will be $\frac{1}{N}X_{N/2}e^{j2\pi N/2n/N} = X_{N/2} \cos(n\pi)$
- ▶ Overall:

$$x[n] = \frac{1}{N}X_0 + \frac{1}{N} \sum_{k=0}^{(N-2)/2} 2|X_k| \cos(2\pi k/Nn + \angle X_k) + \underline{\frac{1}{N}X_{N/2} \cos(n\pi)}$$

- ▶ A sum of sinusoids with frequencies up to 1/2

Expressing as sum of sinusoids

- ▶ DFT says that any periodic signal $x[n]$, with period N , is **a sum of N sinusoids with frequencies:**

$$f = 0, \frac{1}{N}, \frac{2}{N}, \dots \text{ up to } \frac{N-1}{2} \text{ or } \frac{N}{2}$$

(not exceeding 1/2)

- ▶ The **modulus** $|X_k|$ gives the **amplitude** of the sinusoids (sometimes $\times 2$)
 - ▶ for $\omega = 0$, $|X_0|$ = the DC component
 - ▶ when modulus = 0, that frequency has amplitude 0
- ▶ The **phase** $\angle X_k$ gives the initial phase

Example

- ▶ Consider a periodic signal $x[n]$ with period $N = 6$ and the DFT coefficients:

$$X_k = [15.0000 + 0.0000i, -2.5000 + 3.4410i, -2.5000 + 0.8123i, \\ -2.5000 - 0.8123i, -2.5000 - 3.4410i]$$

Write $x[n]$ as a sum of sinusoids.

- ▶ Do the same for a periodic signal $x[n]$ with period $N = 5$ and the DFT coefficients:

$$X_k = [21.0000 + 0.0000i, -3.0000 + 5.1962i, -3.0000 + 1.7321i, \\ -3.0000 + 0.0000i, -3.0000 - 1.7321i, -3.0000 - 5.1962i]$$

Write $x[n]$ as a sum of sinusoids.

- ▶ The DFT (and the inverse IDFT) is equivalent with a matrix multiplication:
 - ▶ on whiteboard
- ▶ In the world of discrete signals, there are many signal transforms possible, and many of them can be expressed as matrix multiplications, just like the DFT.

1. Linearity

If the signal $x_1[n]$ has the DFT coefficients $\{X_k^{(1)}\}$, and $x_2[n]$ has $\{X_k^{(2)}\}$,
then their sum has

$$a \cdot x_1[n] + b \cdot x_2[n] \leftrightarrow \{a \cdot X_k^{(1)} + b \cdot X_k^{(2)}\}$$

Proof: via definition

2. Shifting in time

If $x[n] \leftrightarrow \{X_k\}$, then

$$x[n - n_0] \leftrightarrow \{e^{(-j2\pi kn_0/N)} X_k\}$$

Proof: via definition

- ▶ The amplitudes $|X_k|$ are not affected, shifting in time **affects only the phase**

Properties of the DFT

3. Modulation in time

$$e^{j2\pi k_0 n/N} \leftrightarrow \{X_{k-k_0}\}$$

4. Complex conjugation

$$x^*[n] \leftrightarrow \{X_{-k}^*\}$$

Properties of the DFT

5. Circular convolution

Circular convolution of two signals \leftrightarrow product of coefficients

$$x_1[n] \otimes x_2[n] \leftrightarrow \{N \cdot X_k^{(1)} \cdot X_k^{(2)}\}$$

Circular convolution definition:

$$x_1[n] \otimes x_2[n] = \sum_{k=0}^{N-1} x_1[k] x_2[(n-k)_N]$$

- ▶ takes two periodic signals of period N, result is also periodic with period N
- ▶ Example at the whiteboard: how it is computed

X_k for signal $x_1[n]$
 X_k coeffs for $x_2[n]$

Example

Example (write on slides)

Circular convolution

- ▶ We are in the vector space of **periodic signals** with period N
- ▶ Linear (e.g. normal) convolution produces a result which is longer periodic with period N
- ▶ Circular convolution takes two sequences of length N and produces another sequence of length N
 - ▶ each sequence is a period of a periodic signal
 - ▶ circular convolution = like a convolution of periodic signals

Properties of the DFT

6. Product in time

Product in time \leftrightarrow circular convolution of DFT coefficients

$$x_1[n] \cdot x_2[n] \leftrightarrow \sum_{m=0}^{N-1} X_m^{(1)} X_{(k-m)_N}^{(2)} = X_k^{(1)} \otimes X_k^{(2)}$$

Duality:

multip.	\longleftrightarrow	conv.
conv.	\longleftrightarrow	multip.

Properties of the DFT

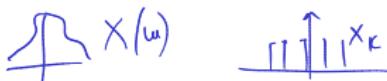
- ▶ **Parseval theorem:** energy of the signal is the same in time and frequency domains

$$X_{1c} \in \mathbb{C}$$

$$E = \sum_0^{N-1} |x[n]|^2 = \frac{1}{2\pi} \sum_k |X_k|^2$$

- ▶ Is true for all orthonormal bases

Plot / sketch DTFT and DFT of various signals



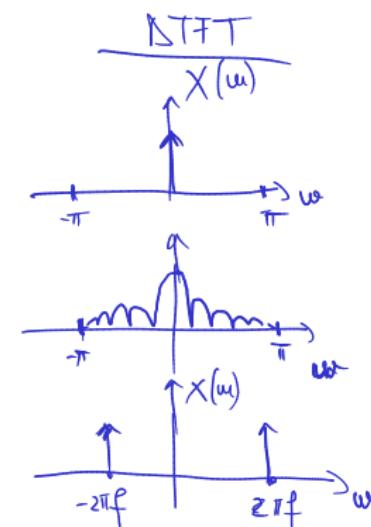
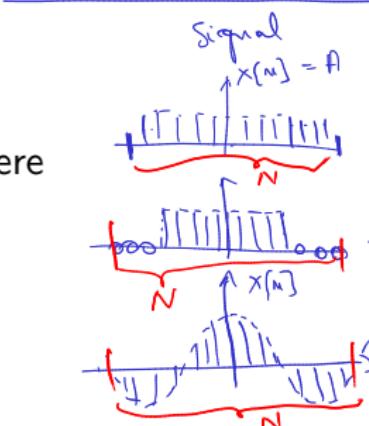
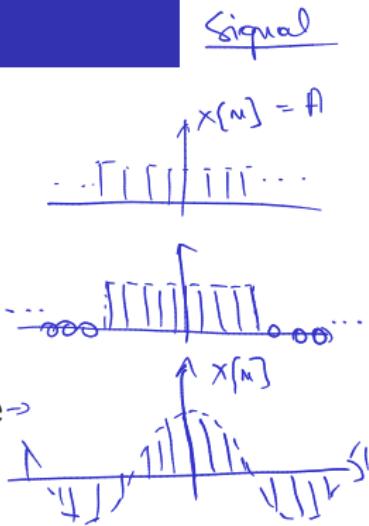
Let's plot / sketch DTFT and DFT of various signals

DTFT of:

- ▶ a constant signal $x[n] = A \rightarrow$
- ▶ a rectangular signal $x[n] = A$ between $-\tau$ and τ , 0 elsewhere \rightarrow
- ▶ a cosine of frequency precisely $f = k/N \rightarrow$
- ▶ a cosine of frequency not $f = k/N$

DFT, with $N=20$, of:

- ▶ a constant signal $x[n] = A$
- ▶ a rectangular signal $x[n] = A$ between $-\tau$ and τ , 0 elsewhere
- ▶ a cosine of frequency precisely $f = k/N$
- ▶ a cosine of frequency not $f = k/N$



DFT:

Two plots for a cosine signal. The top plot shows the DFT $X(n)$ as a rectangular window function of width N centered at $n=0$. The bottom plot is a note: "let's discuss later".

DFT matrix

- The DFT (and the inverse IDFT) is equivalent with a matrix multiplication:

DFT:

$$\mathbf{X}_k = \boxed{\mathbf{W}_N \mathbf{x}}$$

$$\left[\begin{array}{c} X_k \\ \vdots \\ X_{N-1} \end{array} \right] = \left[\begin{array}{c} \mathbf{W}_N \\ \vdots \\ \mathbf{W}_N \end{array} \right] \mathbf{x}$$

$$\mathbf{W}_N$$

$$\left[\begin{array}{c} \mathbf{W}_N \\ \vdots \\ \mathbf{W}_N \end{array} \right], \quad \left[\begin{array}{c} \mathbf{x} \\ \vdots \\ \mathbf{x} \end{array} \right]$$

IDFT:

$$\mathbf{x} = \boxed{\mathbf{W}_N^T \mathbf{X}_k}$$

where

$$\mathbf{X}_k = \begin{bmatrix} X_0 \\ X_1 \\ \vdots \\ X_{N-1} \end{bmatrix}, \quad \mathbf{x} = \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_{N-1} \end{bmatrix},$$

Spectrum Signal

DFT matrix (continued)

$$W_N = \frac{1}{\sqrt{N}} \begin{bmatrix} W_N^{0,0} & W_N^{0,1} & \dots & W_N^{0,(N-1)} \\ W_N^{1,0} & W_N^{1,1} & \dots & W_N^{1,(N-1)} \\ \vdots & \vdots & \ddots & \vdots \\ W_N^{(N-1),0} & W_N^{(N-1),1} & \dots & W_N^{(N-1),(N-1)} \end{bmatrix}$$

$k \rightarrow$ *$n \downarrow$* *\leftarrow complex numbers*

with an element $W_N^{kn} = e^{-j2\pi \frac{k}{N}n}$ (k = row index, n = column index)

- ▶ there might be small variations depending on whether we have $\frac{1}{\sqrt{N}}$ at both DFT and IDFT, or just put $\frac{1}{N}$ just for the IDFT
- ▶ note that for IDFT we have $W^{-1} = W^T$ (orthonormal basis)

$$\begin{array}{c} X_k \\ \vdots \\ X_k \end{array} = \boxed{\dots \quad \dots} \cdot \begin{array}{c} W_N \\ \vdots \\ W_N \end{array} \cdot \begin{array}{c} x \\ \vdots \\ x \end{array}$$

$$\begin{array}{c} x \\ \vdots \\ x \end{array} = \boxed{\dots \quad \dots} \cdot \begin{array}{c} W_N^{-1} = W_N^T \\ \vdots \\ W_N^{-1} = W_N^T \end{array} \cdot \begin{array}{c} X_k \\ \vdots \\ X_k \end{array}$$

DFT matrix multiplication

- ▶ Naive implementation of DFT, IDFT: use matrix multiplication with W, W^{-1}
- ▶ Number of multiplications necessary for a vector of length N is N^2
- ▶ In the world of algorithms, the computational complexity of an algorithm = number of multiplications necessary, depending on some variable N
 - ▶ only the dominant term matters, no coefficient, e.g $O(N^2)$ not $7.3N^2 + 4N$
- ▶ Naive DFT has computation complexity $O(N^2)$
 - ▶ prohibitively large

N multip. on first row
 N - - - - -
znd row

$\begin{matrix} & \end{matrix} = \begin{matrix} & \end{matrix} \cdot \begin{matrix} & \end{matrix}$

$N \times N$ $N \times 1$

N^2 multiplication

$N = 100 \quad 100^2 = 10,000$

$N = 1000 \quad 1000^2 = 1,000,000$

$N = 1,000,000 \quad 10^{12} = \text{huge}$

FFT

~~fft~~

fft()

- ▶ The Fast Fourier Transform (FFT) algorithm = a fast algorithm for computing the DFT, exploiting the particular nature (symmetries) in the DFT matrix
- ▶ FFT computational complexity: $\mathcal{O}(N \log_2(N))$
- ▶ Exercise: for $N = 1024$, how much faster is FFT compared to naive DFT multiplication?
- ▶ Invention and adoption of FFT (~'60s, Cooley & Tukey) = “the birth of Digital Signal Processing”

$$N = 1024 = 2^{10}$$

$$\text{FFT : } \mathcal{O}(1024 \cdot 10) \\ \approx 10^8$$

$$\text{Naive DFT : } \mathcal{O}(1024^2) \\ \approx 10^8$$

For $N = 1024$,
FFT is $100 \times$ faster than
naive DFT

Other transforms

- In the world of discrete signals, there are many signal transforms possible, and many of them can be expressed as matrix multiplications, just like the DFT.

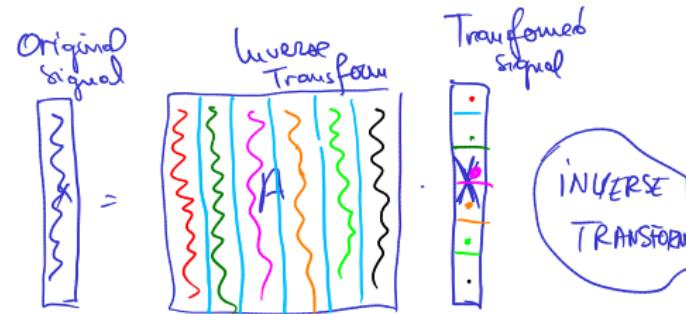
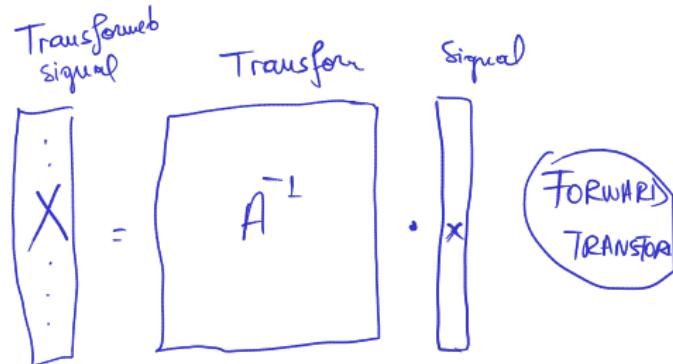
- Transform = expressing a N -dimensional vector x as a linear combination of a set of N basis vectors

- How:

- Put the N vectors of the basis as columns in a matrix A
- Solve the system $x = AX$ (inverse transform)
- Which means $X = A^{-1}x$ (forward transform)

- Why:

- compression: the discrete cosine transform is the basis for lossy JPEG image compression
- ...



$$\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix} = \begin{bmatrix} b_1 & c_1 \\ b_2 & c_2 \\ \vdots & \vdots \\ b_N & c_N \end{bmatrix} + \begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_N \end{bmatrix} + \dots + \begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_N \end{bmatrix}$$

Example

$$x[n] = \frac{1}{3} \cdot u[n+2] + \frac{1}{3} u[n+1] + \frac{1}{3} u[n] - \frac{1}{3} u[n-3]$$

- Consider the exercise from Week 2:

$$x[n] = \{ \dots, 0, \frac{1}{3}, \frac{2}{3}, 1, 1, 1, 1, 1, 0, \dots \}$$

Write the expression of $x[n]$ based on the signal $u[n]$.

Solve this in Matlab using a matrix approach

$$x[n] = \frac{1}{3} \cdot u[n+2] + \frac{1}{3} u[n+1] + \frac{1}{3} u[n] - \frac{1}{3} u[n-4]$$

Transf.: $\boxed{U} = A^{-1} \cdot \boxed{X}$

Inverse: $\boxed{X} = A \cdot \boxed{B}^{coeffs}$

$$\begin{matrix} X \\ \vdots \\ 0 \\ \frac{1}{3} \\ \frac{2}{3} \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 0 \end{matrix} = \begin{matrix} u[n+3] \\ u[n+2] \\ u[n+1] \\ u[n] \\ u[n-1] \\ u[n-2] \\ u[n-3] \end{matrix}$$

Unknowns:

$$\begin{matrix} 0 \\ \frac{1}{3} \\ \frac{1}{3} \\ 0 \\ \frac{1}{3} \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ -1 \end{matrix}$$

$\boxed{A} = \boxed{A} \cdot \boxed{B}$, solve and find \boxed{X}

Unknowns

Another example: JPEG compression

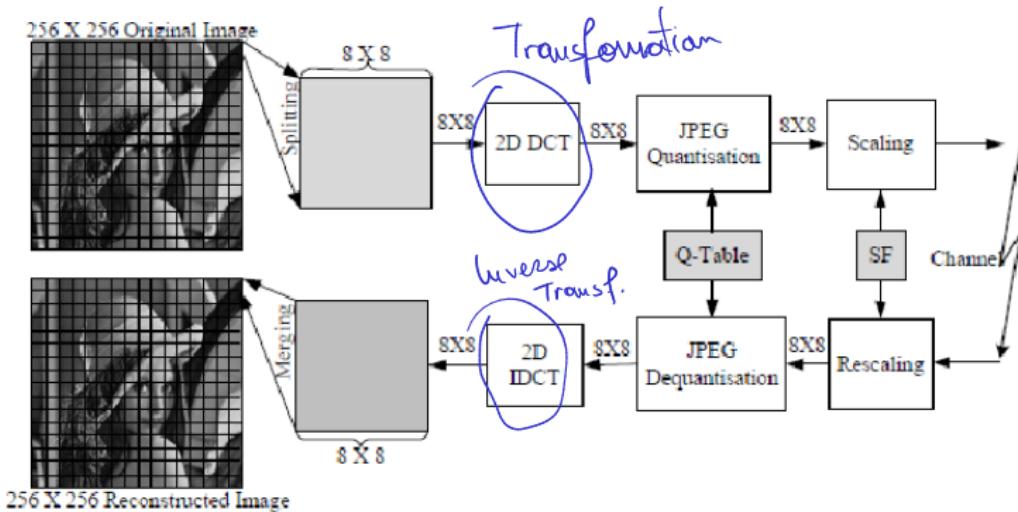
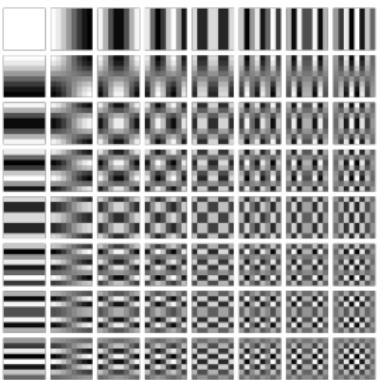


Figure 4: JPEG compression with DCT transform

- ▶ image from: JPEG Picture Compression Using Discrete Cosine Transform, N. K. More, S. Dubey, 2012

Another example: JPEG (cont'd)

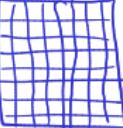
- ▶ Each 8x8 image block is a vector in a 64 dimensional space
- ▶ Each 8x8 image block is decomposed into 64 basis vectors



← Decompose the original
sum of these 64

Figure 5: 8x8 DCT basis vectors

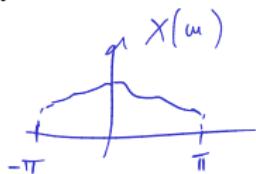
- ▶ Result: 64 coefficients, but many are small, negligible, quantizable => compression
- ▶ image from: Wikipedia

8x8 pixels:

= a vector in
an N -dim space
 $N = 64$
64 values
8x8 block into a
blocks, each with a
coefficient
64 coeffs

Relationship between DTFT and DFT

- ▶ How are DTFT and DFT related?
- ▶ Discrete Time Fourier Transform:

- ▶ for non-periodical signals
- ▶ spectrum is continuous



- ▶ Discrete Fourier Transform

- ▶ for periodical signals $N =$
- ▶ spectrum is discrete $N =$



- ▶ Duality: periodic in time \leftrightarrow discrete in frequency

Relationship between DTFT and DFT

- ▶ Consider a non-periodic signal $x[n]$
- ▶ It has a continuous spectrum $X(\omega)$
- ▶ If we **periodize** it by repeating with period N:

$$x_N[n] = \sum_{k=-\infty}^{\infty} x[n - kN]$$

- ▶ then the Fourier transform is **discrete** (made of Diracs):

$$X_N(\omega) = 2\pi X_k \delta(\omega - k \frac{2\pi}{N})$$

- ▶ The coefficients of the Diracs = the DFT coefficients

$$X_k = X(2\pi k / Nn)$$

- ▶ They are **samples** from the continuous $X(\omega)$ of the non-periodized signal

Relationship between DTFT and DFT

- ▶ Example: consider a sequence of 7 values

$$x = [6, 3, -4, 2, 0, 1, 2]$$

- ▶ If we consider a x surrounded by infinitely long zeros ($x[n]$ non-periodical), we have a continuous spectrum $X(\omega)$ (DTFT)

$$x = [\dots, 0, 6, 3, -4, 2, 0, 1, 2, 0, \dots] \leftrightarrow X(\omega)$$

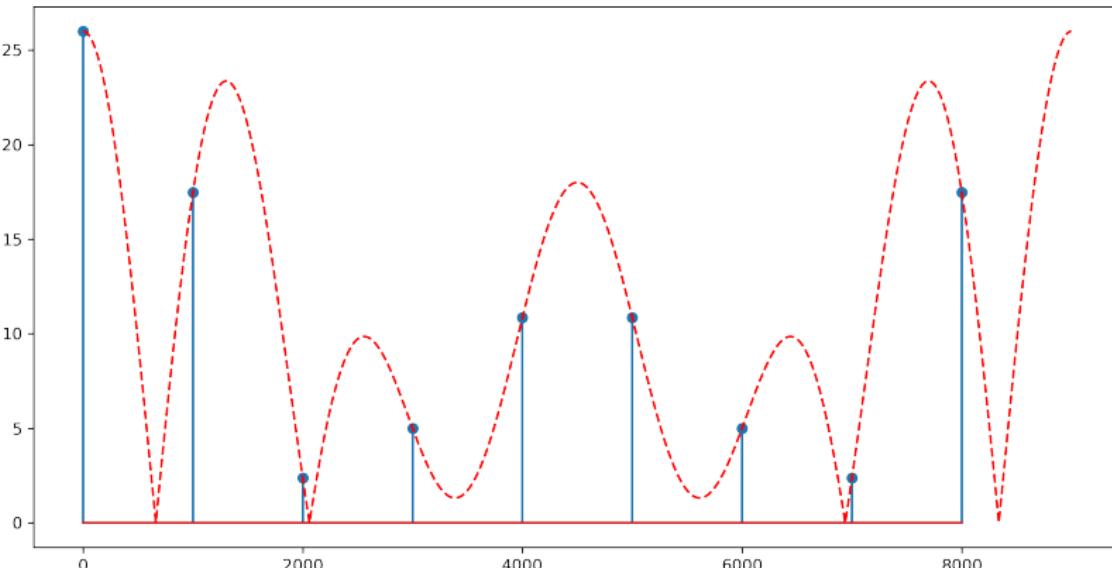
- ▶ If we consider that x is surrounded by repeating the sequences ($x[n]$ periodical), we have a discrete spectrum X_k (DFT)

$$x = [\dots, -4, 2, 0, 1, 2, 0, 6, 3, -4, 2, 0, 1, 2, 0, 6, 3, -4, \dots] \leftrightarrow X_k$$

- ▶ The discrete X_k are just **samples from** $X(\omega)$:

$$X_k = X(2\pi k / Nn)$$

Relationship between DTFT and DFT



$$x = [6, 5, 4, -3, 2, -3, 4, 5, 6]$$

- ▶ red line = DTFT of $x[n]$ (assuming surrounded by zeros)
 - ▶ (actually run as `fft(x, 10000)`, x is extended with 9991 zeros)
- ▶ blue = DFT of x (assumes periodic) = `fft(x)`

Study case

```
n = 0:99;  
f = 0.015;  
x = cos(2*pi*f*n)  
plot(abs(fft(x)))
```

Discuss:

1. Why is the spectrum not just 2 Diracs, like a normal $\cos()$?
2. FFT assumes periodicity. Are there boundary problems?
3. What is the role of the rectangular window?
4. What happens if we run $\text{fft}(x, 10000)$ instead of $\text{fft}(x)$?

The need for windowing

- ▶ When you have finite-length cosine vector x , you have just a part of your signal:

$$x = \cos(2\pi fn) \cdot w[n]$$

- ▶ The spectrum of $x = x[n] \cdot w[n]$ is Diracs * $W(\omega)$
- ▶ Instead of Diracs, you have $W(\omega)$'s:
 - ▶ wide peak
 - ▶ secondary lobes
- ▶ Working with a piece of a signal **always** means a windowing, i.e. the spectrum is affected
 - ▶ every Dirac is “smudged” into a $W(w)$

The need for windowing

- ▶ What if we change the window $w[n]$
 - ▶ Recangular window
 - ▶ Hamming window
 - ▶ Hann window
 - ▶ ...
- ▶ What they do: trade narrow peak vs small secondary lobes
 - ▶ Rectangular window: widest peak, smallest secondary lobes
 - ▶ Other windows : narrower peak, higher secondary lobes
- ▶ What they do: attenuate endings, to reduce boundary problems

The need for windowing

- ▶ Remember: every time we work with a piece of a signal (e.g. we process an audio file in pieces of 1024 samples), we are applying windowing
- ▶ Even cutting a part of a signal, that's still applying a window
- ▶ Even the rectangular window is still a window
- ▶ Consider replacing the rectangular window with another one, if you use `fft()` or other frequency-based operations

How to compute the DTFT

- ▶ The DFT is computed with `fft(x)`
- ▶ How to compute the DTFT?
- ▶ You can't. You need to surround x with infinitely long zeros
- ▶ You can only surround it with many zeros, but still finite
- ▶ Do this with `fft(x, 100000)` (DFT in $N=100000$ points)
 - ▶ x is surrounded with zeros until total length = 100000
 - ▶ it's still just `fft()`, so DFT, so you have points and not the full continuous function
 - ▶ but it's many many points

How to compute the DTFT

- ▶ Computing the `fft()` in $N=100000$ points is unrelated with windowing
- ▶ Windowing changes the Diracs into $W(\omega)$'s
- ▶ `fft()` in N points is just taking N points from the resulting continuous spectrum

Relation between DTFT and Z transform

- Z transform:

$$X(z) = \sum_n x[n]z^{-n}$$

- DTFT:

$$X(\omega) = \sum_n x[n]e^{-j\omega n}$$

- DTFT can be obtained from Z transform with

$$z = e^{j\omega}$$

- These $z = e^{j\omega}$ are **points on the unit circle**

- $|z| = |e^{j\omega}| = 1$ (*modulus*)
- $\angle z = \angle e^{j\omega} = \omega$ (*phase*)

Relation between DTFT and Z transform

- ▶ Fourier transform = Z transform evaluated **on the unit circle**
 - ▶ if the unit circle is in the convergence region of Z transform
 - ▶ otherwise, equivalence does not hold
- ▶ This is true for most usual signals we work with
 - ▶ some details and discussions are skipped

Geometric interpretation of Fourier transform

$$X(z) = C \cdot \frac{(z - z_1) \cdots (z - z_M)}{(z - p_1) \cdots (z - p_N)}$$

$$X(\omega) = C \cdot \frac{(e^{j\omega} - z_1) \cdots (e^{j\omega} - z_M)}{(e^{j\omega} - p_1) \cdots (e^{j\omega} - p_N)}$$

► Modulus:

$$|X(\omega)| = |C| \cdot \frac{|e^{j\omega} - z_1| \cdots |e^{j\omega} - z_M|}{|e^{j\omega} - p_1| \cdots |e^{j\omega} - p_N|}$$

► Phase:

$$\angle X = \angle C + \angle(e^{j\omega} - z_1) + \cdots + \angle(e^{j\omega} - z_M) - \angle(e^{j\omega} - p_1) - \cdots - \angle(e^{j\omega} - p_N)$$

Geometric interpretation of Fourier transform

- ▶ For complex numbers:
 - ▶ modulus of $|a - b|$ = the length of the segment between a and b
 - ▶ phase of $|a - b|$ = the angle of the segment from b to a (direction is important)
- ▶ So, for a point on the unit circle $z = e^{j\omega}$
 - ▶ modulus $|X(\omega)|$ is **given by the distances to the zeros and to the poles**
 - ▶ phase $\angle X(\omega)$ is **given by the angles from the zeros and poles to z**

- ▶ Consequences:
 - ▶ when a **pole** is very close to unit circle -> Fourier transform is **large** at this point
 - ▶ when a **zero** is very close to unit circle -> Fourier transform is **small** at this point
- ▶ Examples: . . .

- ▶ Simple interpretation for modulus $|X(\omega)|$:
 - ▶ Z transform $X(z)$ is like **a landscape**
 - ▶ **poles = mountains** of infinite height
 - ▶ **zeros = valleys** of zero height
 - ▶ Fourier transform $X(\omega) = \text{"Walking over this landscape along the unit circle"}$
 - ▶ The height profile of the walk gives the amplitude of the Fourier transform
 - ▶ When close to a mountain \rightarrow road is high \rightarrow Fourier transform has large amplitude
 - ▶ When close to a valley \rightarrow road is low \rightarrow Fourier transform has small amplitude

Geometric interpretation of Fourier transform

- ▶ Note: $X(z)$ might also have a constant C in front!
 - ▶ It does not appear in pole-zero plot
 - ▶ The value of $|C|$ and $\angle C$ must be determined separately
- ▶ This “geometric method” can be applied for phase as well

- ▶ **Duality** properties related to all Fourier transforms
- ▶ Discrete \leftrightarrow Periodic
 - ▶ **discrete** in time \rightarrow **periodic** in frequency
 - ▶ **periodic** in time \rightarrow **discrete** in frequency
- ▶ Continuous \leftrightarrow Non-periodic
 - ▶ **continuous** in time \rightarrow **non-periodic** in frequency
 - ▶ **non-periodic** in time \rightarrow **continuous** in frequency

Terminology

- ▶ Based on frequency content:
 - ▶ **low-frequency** signals
 - ▶ **mid-frequency** signals (band-pass)
 - ▶ **high-frequency** signals
- ▶ **Band-limited** signals: spectrum is 0 beyond some frequency f_{max}
- ▶ **Bandwidth** B : frequency interval $[F_1, F_2]$ which contains 95% of energy
 - ▶ $B = F_2 - F_1$
- ▶ Based on bandwidth B :
 - ▶ **Narrow-band** signals: $B \ll$ central frequency $\frac{F_1+F_2}{2}$
 - ▶ **Wide-band** signals: not narrow-band