Discrete Fourier Transform

Suppose $x \in \mathbb{C}^N$, then the N-point DFT of x is given by

$$F : \mathbb{C}^N \to \mathbb{C}^N$$
$$x \mapsto \hat{x} \text{ with}$$

$$\boxed{\hat{x}(k) = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} x_n e^{-2\pi i k n/N}}$$

so $\hat{x}(k) = \langle x, \varphi^{(k)} \rangle = \varphi^{(k)*} x$

$$\varphi^{(k)} = \begin{pmatrix} 1 \\ e^{2\pi i k/N} \\ e^{2\pi i \cdot 2k/N} \\ \vdots \\ e^{2\pi i (N-1)k/N} \end{pmatrix} \cdot \frac{1}{\sqrt{N}}$$

(= inner product with complex exponential of frequency k/N)

Matrix representation:

Let $w := w_N = e^{-2\pi i/N} = N^{th}$ root of unity

$$\hat{x} = F x$$

$$\hat{x} = \frac{1}{\sqrt{N}} \underbrace{\begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & \omega & \omega^2 & \cdots & \omega^{N-1} \\ 1 & \omega^2 & \omega^4 & & \omega^{2N-2} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & \omega^{N-1} & \omega^{2N-2} & & \omega^{(N-1)^2} \end{bmatrix}}_{\text{DFT Matrix (note the adjoint)}} x$$

Theorem: F is a unitary matrix

Proof: Check that $\langle \varphi^{(k)}, \varphi^{(\ell)} \rangle = \delta_{k,\ell}$

$$= \begin{cases} 1, & k=\ell \\ 0, & \text{oth} \end{cases}$$

Corollary: $\hat{x} = F x$

$$\Longrightarrow x = \underbrace{F^*}_{\text{Inverse DFT.}} \hat{x}$$

Some comments:

Continuous time Fourier transform:

- $f \in L^1$ (or L^2 with appropriate "work")

$$\hat{f}(\omega) = c\int_{-\infty}^{\infty} f(t)\, e^{-i\omega t}\, dt \approx \langle f, \varphi_\omega \rangle$$

$$\varphi_\omega(t) = e^{-i\omega t}$$

gives the "coefficients" of f at frequency ω.

- The DFT gives the frequency coefficients of x at frequency $\frac{k}{N}$.

Convenient Properties of the DFT

- $$x(n) = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} \hat{x}(k)\, e^{2\pi i k n/N}$$

$$= \langle \hat{x}, \bar{\varphi}^{(k)} \rangle$$

(Inversion)

- $$\left.\begin{array}{l} x \longmapsto \hat{x} \\ y \longmapsto \hat{y} \end{array}\right\} \Rightarrow \underbrace{\langle x, y \rangle = \langle \hat{x}, \hat{y} \rangle}_{\text{Parseval}}$$

- $$\left.\begin{array}{l} \|x\|^2 = \|\hat{x}\|^2 \\ \hookrightarrow \sum_{n=0}^{N-1} |x_n|^2 \end{array}\right\} \text{Plancharel}$$

- Phase shift:

Suppose $x \longmapsto \hat{x}$

Let y : $y(n) = x(n)\, e^{2\pi i m n/N}$

then $\hat{y}(k) = \hat{x}(k-m)$

Proof: exercise (change of variable)

- Convolution: (circular)

Define the circular convolution of $a, b \in \mathbb{C}^N$ by

$$(a \circledast b)_n := \sum_{\ell=0}^{N-1} a_\ell \, b_{(n-\ell) \bmod N}$$

Then

(I) if $x \mapsto \hat{x}$, $y \mapsto \hat{y}$

$$\frac{1}{\sqrt{N}} (x \circledast y)_n = \left[F^{-1} (\hat{x} \odot \hat{y}) \right]_n$$

pointwise multiplication

(II)

$$F(x \odot y)_k = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} x_n y_n e^{-2\pi i \frac{kn}{N}}$$

$$= \frac{1}{\sqrt{N}} (\hat{x} \circledast \hat{y})_k$$

Proof: HW

Corollary: The DFT matrix diagonalizes any <u>circulant</u> matrix.

Proof: HW

—x—

Applications: (just a couple of examples)

- Signal analysis: understanding the frequency content of a signal
- extension: STDFT

$x \in \mathbb{C}^N \longmapsto \tilde{x} : \tilde{x}(k, \ell) =$

$$c \sum_{n=0}^{N-1} x(n) \underbrace{W(n-\ell)} e^{-2\pi i k n / N}$$

W: compactly supported or approximately so

Exercise: • Visualize the STFT of a few seconds of music

• use w supported on approximately 60 – 120 msec.

• Compression (e.g. Via the DCT)

JPEG works this way.

• Major advantage: the Fast Fourier Transform algorithm.

Normally to compute a matrix-vector product of the form

$$\hat{x} = \underset{N\times N}{F}\ \underset{N\times 1}{x}$$

we need $O(N^2)$ operations.

In the case of the DFT, the FFT algorithm allows us to do it in $O(N \log N)$!

Idea (Cooley-Tukey 1965, Gauss?)

Let $N = 2^n$ and ignore the $\frac{1}{\sqrt{N}}$ normalization for now.

$$\hat{x}_k = \sum_{n=0}^{N-1} x_n e^{-2\pi i k n / N}$$

$$= \sum_{n \text{ even}} \left(\qquad \right) + \sum_{n \text{ odd}} \left(\qquad \right)$$

$$= \underbrace{\sum_{m=0}^{N/2-1} x_{2m}\, e^{-2\pi i (2m) k/N}}_{e_k} + \underbrace{\sum_{m=0}^{N/2-1} x_{2m+1}\, e^{-2\pi i (2m+1) k/N}}_{e^{-2\pi i k/N} \underbrace{\sum_{m=0}^{N/2-1} x_{2m+1}\, e^{-2\pi i \frac{km}{N/2}}}_{O_k}}$$

So $$\hat{x}_k = e_k + e^{-2\pi i k/N}\, O_k$$

$\frac{N}{2}$ point DFT of $(x_0, x_2, \dots, x_{2n-2})$

$N/2$ pt DFT of $(x_1, x_3, \dots, x_{2n-1})$

for $k \in \{0, \dots, N/2 - 1\}$

Similarly

$$\hat{x}_{k+N/2} = e_k - e^{-2\pi i k/N}\, O_k$$

for $k \in \{N/2, \dots, N-1\}$

Now, we can compute each of the 2 $N/2$ pt DFTs by repeating this even/odd ~~splitting~~

$\Rightarrow N/4$ pt DFT $\rightarrow N/8$ pt DFT

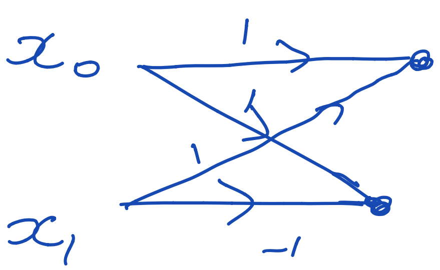
$\rightarrow \cdots \longrightarrow$ 2 pt DFT

(Butterfly)

$$F_2 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

Examples:

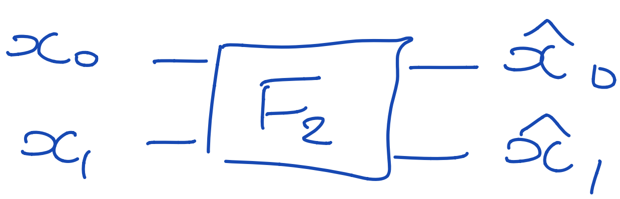
2 pt DFT:



$\hat{x}_0 = x_0 + x_1$

$\hat{x}_1 = x_0 - x_1$

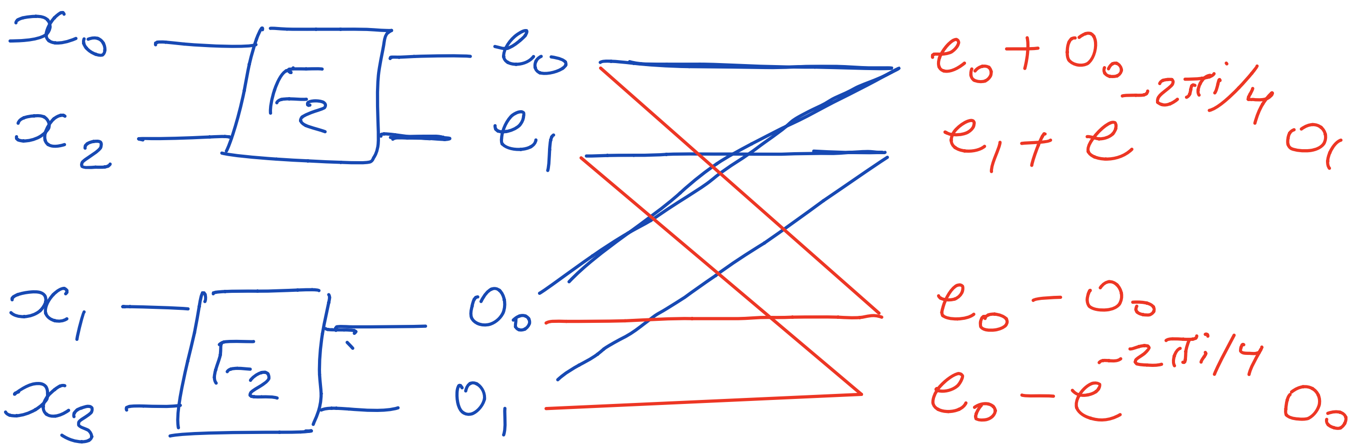
write this as



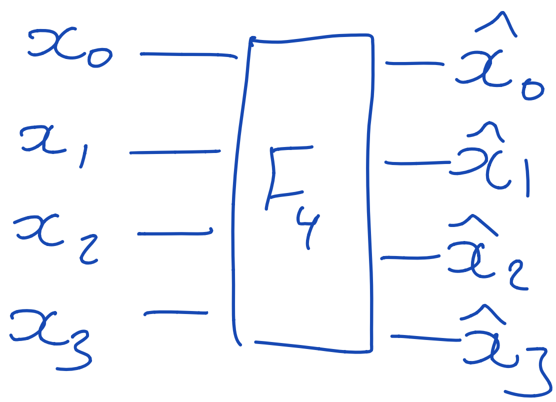
4 pt DFT

Recall $\hat{x}_k = e_k + e^{-2\pi i k/N} o_k$

$\hat{x}_{k+N/2} = e_k - e^{-2\pi i k/N} o_k$



write this as

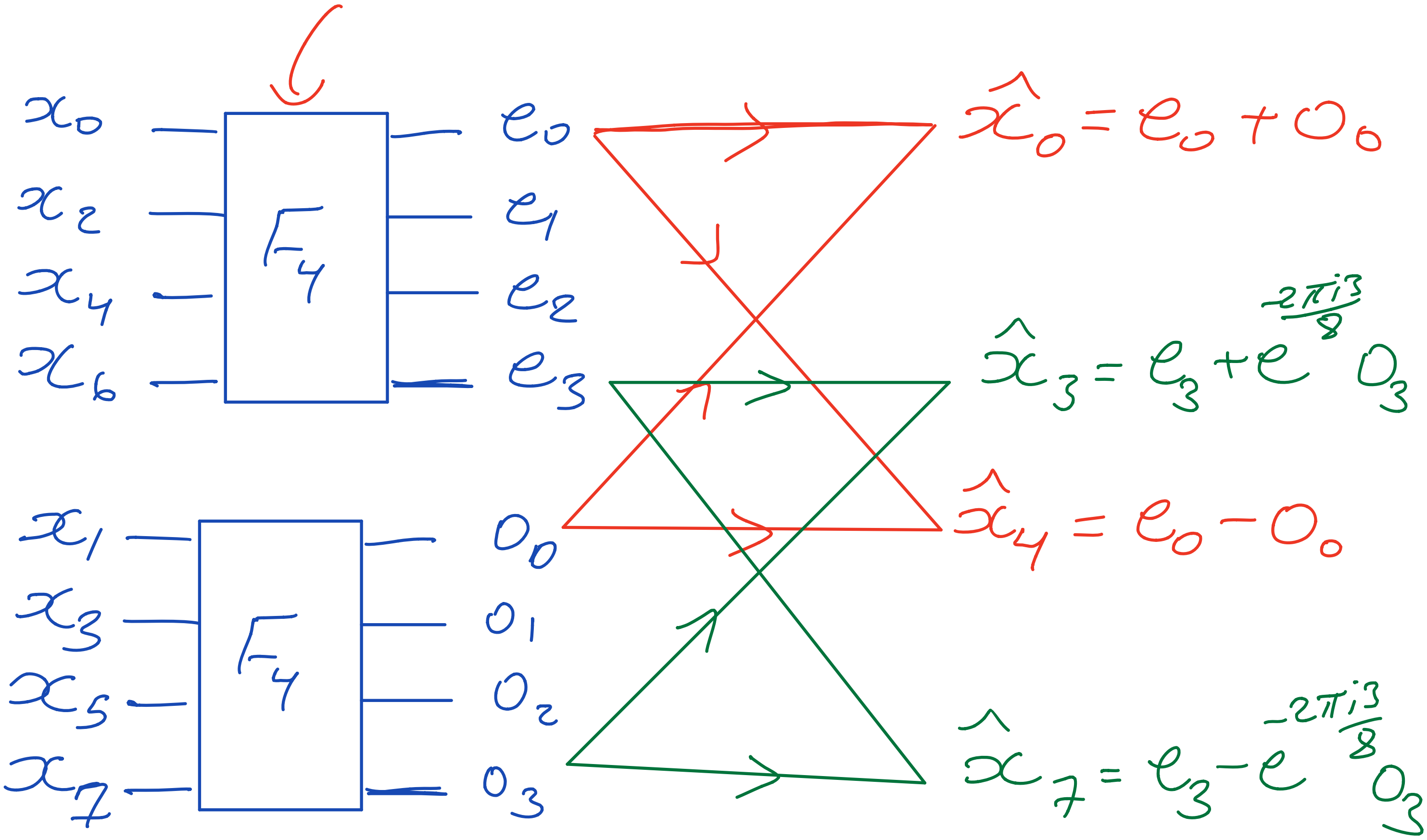


8 pt DFT

Same idea

as above

as above



Computational Complexity

- We have $\log_2(N)$ levels
- Each level has $N/2$ butterfly operations
- Each butterfly is

  2 complex adds
  & 1 complex mult.

$\Rightarrow O(N \log N)$

Remark: Need to reorder the input (due to even/odd splitting at every stage)

	binary rep of index	bit reversal
$x_0 \rightarrow$	000	000
$x_4 \rightarrow$	100	001
$x_2 \rightarrow$	010	010
$x_6 \rightarrow$	110	011
$x_1 \rightarrow$	001	100
$x_5 \rightarrow$	101	101
$x_3 \rightarrow$	011	110
$x_7 \rightarrow$	111	111

What about F^{-1}?

Can show that with unitary DFT

$$x = F(\overline{\hat{x}}) \Rightarrow O(N \log N)$$

also