# DSP Lecture 10 The Discrete Fourier transform



Here's what's up

| | Continuous | Discrete | |
|---|---|---|---|
| Periodic | Fourier Series | DFT (Discrete Fourier transform)<br>DTFS (Discrete time Fourier Series) | FFT |
| Aperiodic | Fourier Transform | DTFT | |

FFT (Fast Fourier transform)
optimised and customized for efficient DFT

---

Intuition, Fourier series take a continuous, periodic signal and represents it as a sum of (complex) sines/cosines

$$x(t) = \sum_{k=-\infty}^{\infty} \underbrace{X[k]}_{a_k} e^{jk\overbrace{\frac{2\pi}{T}}^{\omega_0}t} \qquad t \in [0,T]$$

$x(t)$ may wiggle $\infty$-ly fast we may need $\infty$ many of $a_k$'s

→ So what of discrete time?
periodic discrete-time signal

(2) Say a signal of period 6, $N=6$

fixed # of discrete-time sinusoids

→ can only wiggle so fast (rapidly between 1, -1

→ $\pi$ frequency

write

$n = 0, 1, \ldots, N-1$

$$x[n] = \sum_{k=-\infty}^{\infty} X[k] e^{jk\frac{2\pi}{N}n}$$

→ complex exponential

This periodic signal, $x[n]$, has unique values, N values

weird to use $\infty$ # of ans

$$e^{jk\frac{2\pi}{N}(n+N)} = e^{jk\frac{2\pi}{N}n} e^{j2\pi} = e^{jk\frac{2\pi}{N}n}$$

only N unique complex exponentials of period N

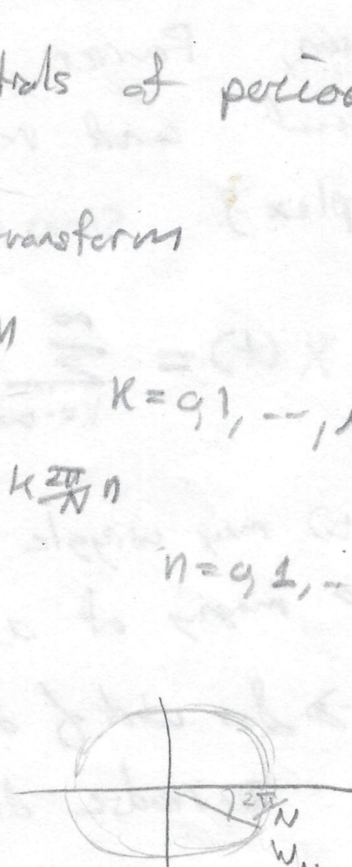
Define Discrete fourier transform

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-jk\frac{2\pi}{N}n} \quad k = 0, 1, \ldots, N-1$$

$$x[n] = \frac{1}{N} \sum_{\cancel{k=0}}^{N-1} X[k] e^{jk\frac{2\pi}{N}n} \quad n = 0, 1, \ldots, N-1$$

9.13 $\quad$ $k=0$ ✓

Define $W_N = e^{-j\frac{2\pi}{N}}$




Note $(W_N)^N = 1$

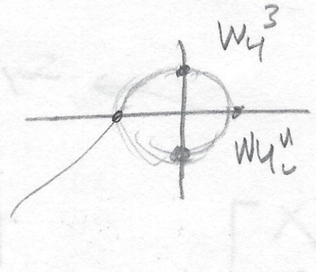
So write

$$X[K] = \sum_{n=0}^{N-1} x[n]\, W_N^{kn}$$

(See other Notes for derivation)

$$x[n] = \frac{1}{N} \sum_{K=0}^{N-1} X[K]\, W_N^{-kn}$$

$$W_4 = e^{-\frac{2\pi}{4}}$$

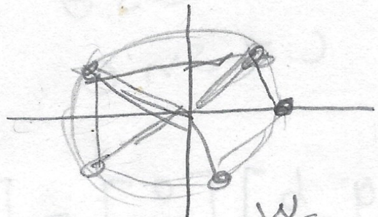



The 4th roots of 1

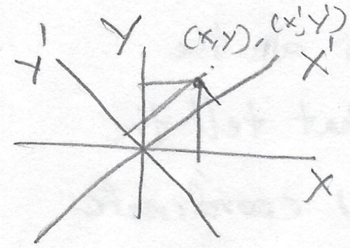
$$z^4 = 1$$

| | |
|---|---|
| $W_4 = e^{-\frac{2\pi}{4}i} = e^{-\pi/2}$ | $e^{-2\pi i}$ |
| $W_4^2 = e^{-\pi i}$ | $e^{-4\pi i}$ |
| $W_4^3 = e^{-\frac{3}{2}\pi i}$ | $e^{-6\pi i}$ |
| $W_4^4 = e^{-2\pi i}$ | $e^{-3\pi i}$ |
| $z$ | $z^4$ |

$W_5$,

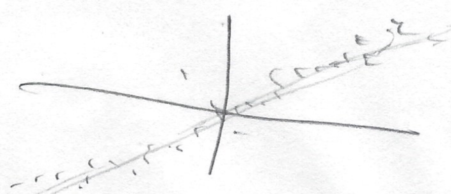



Change of basis




equivalent ways ~~of~~ representing the same point in 2D space

Principle Component Analysis

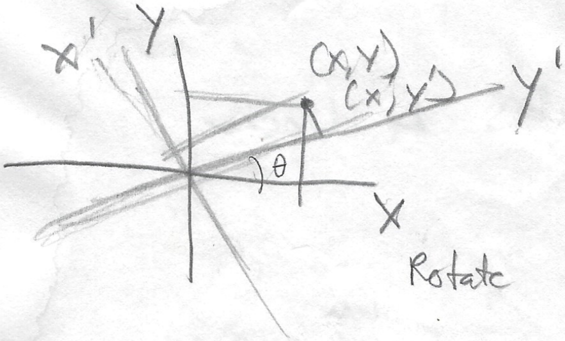


Uli von Videl

Videl von Uli

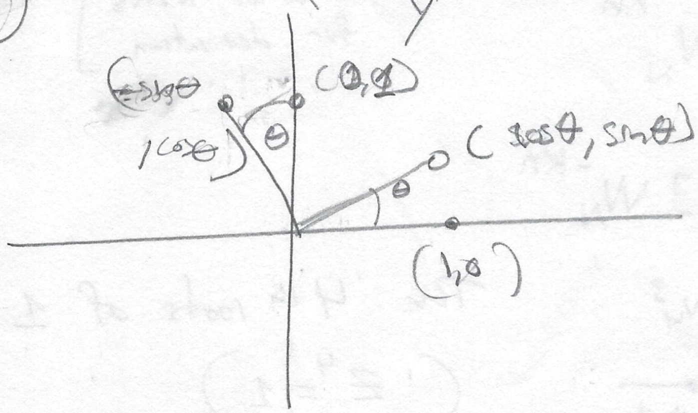
→ change of coordinates is a matrix ~~multiplied by~~ multiples a Vector

So $\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} & \\ & \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$




Rotate θ° clockwise

4




Rotate $\theta°$ counter-clockwise

→ Say 2x2 $\underset{r\ c}{2\times1}$ $\underset{r\ c}{2\times1}$

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} & \\ & \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

$$= \begin{bmatrix} \cos\theta & -\sin\theta \\ \underline{\sin\theta} & \underline{\cos\theta} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} a \\ c \end{bmatrix}$$

$a = \cos\theta$
$c = \sin\theta$

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} b \\ d \end{bmatrix}$$

$b = -\sin\theta$
$d = \cos\theta$

These guys are the new vectors that tell the axis of the new coordinate systems

→ dot product of

$\cos\theta(-\sin\theta) + \sin\theta\cos\theta$

→ orthogonal

DFT is doing this in N-dimension in the complex plane

→ write Fourier formulas in Linear combinations

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

Think

$$\begin{bmatrix} X[0] \\ X[1] \\ \vdots \\ \vdots \\ X[N-1] \end{bmatrix} = \begin{bmatrix} W_N^0 & W_N^0 & \cdots & W_N^0 \\ W_N^0 & W_N^1 & \cdots & W_N^{N-1} \\ W_N^0 & W_N^2 & \cdots & W_N^{2(N-1)} \\ \vdots & & & \\ W_N^0 & W_N^{N-1} & \cdots & W_N^{(N-1)^2} \end{bmatrix} \begin{array}{c} x[n]\text{'s} \\ \downarrow \\ \begin{bmatrix} x[0] \\ x[1] \\ \vdots \\ \vdots \\ x[N-1] \end{bmatrix} \end{array}$$

$N \times 1$ complex vector

$N \times N$ complex matrix

$N \times 1$

$$\underline{X} = \underline{F}\, x$$

output vector

fourier matrix

$\nwarrow$ input

F is orthogonal

columns all have length N, are all perpendicular

$F = \text{dftmtx}(4)$ (4x4) DFT

$$F' * F = \begin{bmatrix} 4 & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 \\ 0 & 0 & 4 & 0 \\ 0 & 0 & 0 & 4 \end{bmatrix}$$

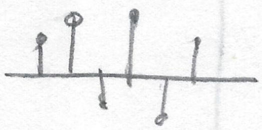
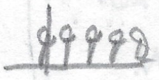
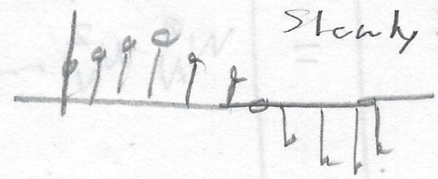
4 is length of vector

So DFT is a coordinate transformation

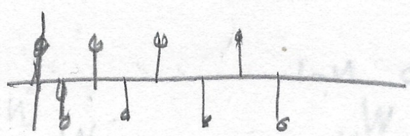
→ we are using the same analogy used in the Discrete-time Fourier Series

1st basis vector

$$\underbrace{\begin{bmatrix} \\ \\ \\ \end{bmatrix}}_{\substack{\text{matrix} \\ N\times 1}} = \underbrace{X[0]}_{\substack{\text{coefficient} \\ N}} \underbrace{\begin{bmatrix} \\ \\ \\ \end{bmatrix}}_{N\times 1} + \underbrace{X[1]}_{N} \underbrace{\begin{bmatrix} \\ \\ \\ \end{bmatrix}}_{N\times 1} + \dots$$

 $= X[0]$  $+ X[1]$ 

Slowly varying sinusoid

$+ X$ 

highest frequency

---

How is this connected to DTFT?

Say we have a finite-length Discrete-time signal

$x[0], x[1], \dots\dots x[N-1]$ (other $x[n] = 0$)

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

$e^{j\omega}$ $\omega \in [-\pi, \pi]$ range

$$= \sum_{n=0}^{N-1} x[n] e^{-j\omega n}$$

Say if $\omega = \frac{2\pi}{N}$

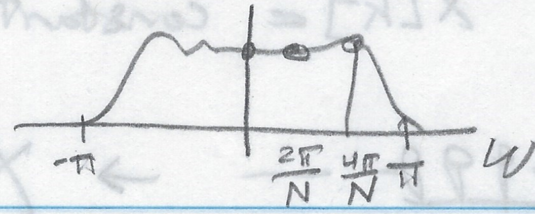
Then the DTFT looks like the DFT

$$\underset{\text{DFT}}{X[k]} = X\left(\frac{2\pi k}{N}\right)$$

or $X\left(e^{j\frac{2\pi}{N}k}\right)$

So now






Going to get N values of DFT equaly spaced like so

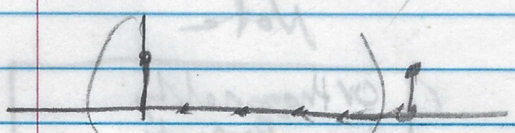
Taking the DTFT, and Sampling it at N equaly spaced points

This is why we want to take the DFT of signals

if we want to sample the continuous DTFT crank up the N

Insight DFT is the DTFT of (discrete-time) finite-length signal evaluated at N equaly spaced points $w = \frac{2\pi k}{N}$ $k = 0, 1, \ldots N-1$

Example DFTs Assumption, input repeats every N units

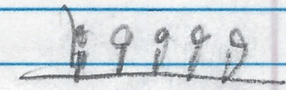



$\delta[n]$

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

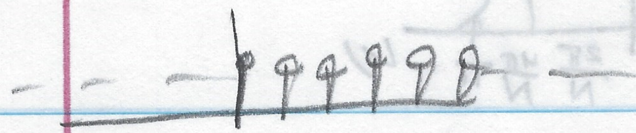
$$= W_N^{(0)k} = 1 \quad \text{all } k$$

So $X[k] =$



8

So if $X[k] =$ constant?



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

$$= \sum_{n=0}^{N-1} W_N^{nk}$$

equivalently

$$= W_N^0 + W_N^k + \cdots W_N^{(N-1)k} = X[k]$$

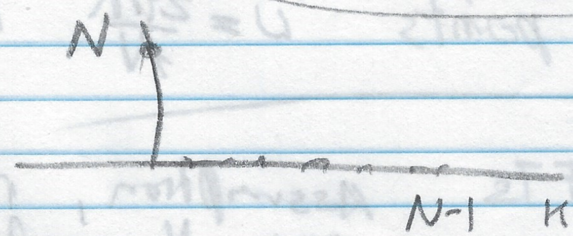
$$= \frac{1-(W_N^k)^N}{1-W_N^k}$$

if $k=0$, $\lim_{x \to 1} \frac{1-x^N}{1-x} = \lim_{x \to 1} \frac{0-N}{0-1} = N$

if $k \neq 0$

See that $W_N = e^{j\frac{2\pi}{N}}$ So $W_N^N = 1$

Then $X[k] = \begin{cases} N & k=0 \\ 0 & k \neq 0 \end{cases}$



$\rightarrow$ duality!

$\delta \rightarrow$ Constant

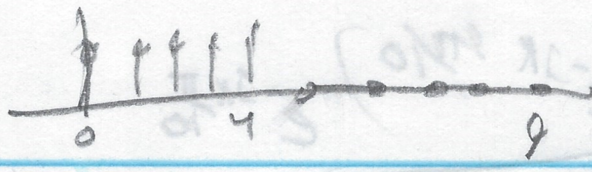
Constant $\rightarrow \delta$

Note

Orthogonality Property

$$\sum_{n=0}^{N-1} W_N^{Mn} = \begin{cases} N & \text{if } M \text{ is a int. mult. of } N \\ 0 & \text{o.w.} \end{cases}$$

Say




finite sum formula

$$\underset{\text{DTFT}}{X(e^{jw})} = \sum_{n=0}^{4} e^{-jwn} = \frac{1-e^{-5jw}}{1-e^{-jw}}$$

recall

$$X(e^{jw}) = \sum_{n=-\infty}^{\infty} e^{-jwn}$$

need

$$\frac{1-e^{-5jw}}{1-e^{-jw}} = \frac{e^{-5jw/2}\left(e^{5jw/2}-e^{-5jw/2}\right)}{e^{-jw/2}\left(e^{jw/2}-e^{-jw/2}\right)} = e^{-2jw}\,\frac{\sin\left(\frac{5w}{2}\right)}{\sin\left(\frac{w}{2}\right)}$$

Length-ten discrete-time signal (10)

Take length-10 DFT

$$X[k] = \sum_{n=0}^{9} x[n]\,\underbrace{e^{-j\frac{2\pi}{10}kn}}_{W_{10}}$$

$$= \sum_{n}^{4} e^{-j\frac{2\pi}{10}kn} = \sum_{n=0}^{4} W_{10}^{kn}$$

$$= \frac{1-W_{10}^{5k}}{1-W_{10}^{k}} = \frac{1-e^{-jk\frac{2\pi}{10}(5)}}{1-e^{-jk\frac{2\pi}{10}}}$$

(10)

which comes to

$$= \frac{\left(e^{jk\frac{\pi}{10}} - e^{-jk\frac{4\pi}{10}}\right)}{e^{-jk\pi/10}\left(e^{jk\pi/10} - e^{-jk\pi/10}\right)} e^{-jk\frac{\pi}{10}}$$

$$= e^{-j4k\frac{\pi}{10}} \frac{\sin\frac{\pi k}{2}}{\sin\frac{\pi k}{10}} \qquad k=0,1 \cdots 9$$

So now see

$$e^{-2jw} \frac{\sin\left(\frac{5w}{2}\right)}{\sin\left(w/2\right)}\Bigg|_{w=\frac{2\pi k}{10}} = e^{-2j\frac{2\pi k}{10}} \frac{\sin\left(\frac{5}{2}\frac{2\pi}{10}k\right)}{\sin\left(\frac{1}{2}\frac{2\pi}{10}k\right)}$$

$$= e^{-j4\frac{\pi}{10}k} \frac{\sin\left(\frac{\pi}{2}k\right)}{\sin\left(\frac{\pi}{10}k\right)} \qquad \underline{5.2.33}$$

when you sample the DTFT you get DFT values (5.2.35)

The DTFT Values are basically sampling the continuous values

DFT samples are hitting the DTFT signal at equally spaced values

If we want a finer resolution DFT

do zero padding

instead of thinking [11111 00000] as

a Length 10 Signal

→ Think of as a Length 100 Signal

⟹ S = [ones(1,5), zeros(1,95)]

By taking a Longer DFT we can Sample the DTFT much More finely

→ we may be interested in a continuous time analog Signal (Sampled to be in digital world)

Do finite-length DFT to get Fourier transform in Matlab (Need Sampling theorem)

---

Properties;

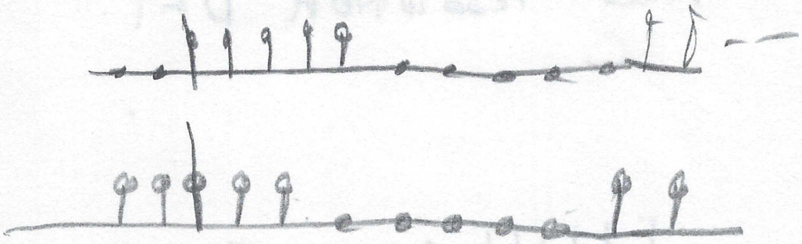
- Linearity, Symmetry
- shift

$W_N = e^{-j\frac{2\pi}{N}}$

$$X[n-m] \xleftrightarrow{DFT} W_N^{km} X[k]$$

59:00

(12) shift by 3 units to the left

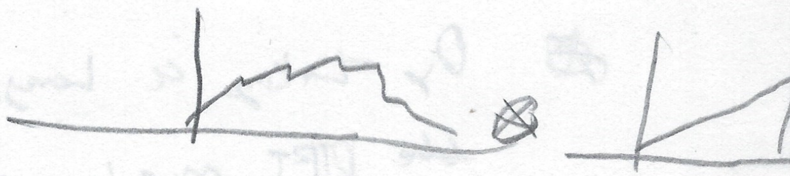


when shifting
→ the period is not increased or decreased
so each pulse shifts as well

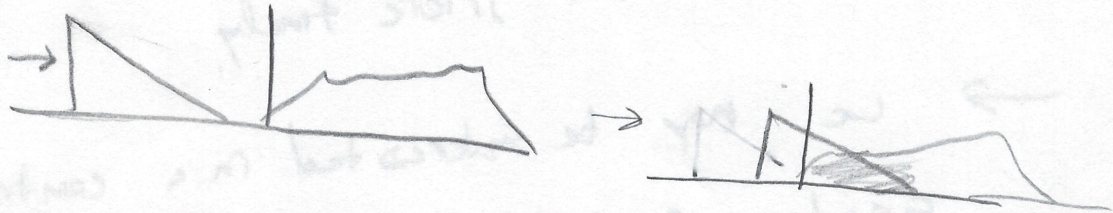
Cyclic shift

when we talk about convolving 2 signals
→ New concept, cyclic convolution ⊛

→ regular convolution:

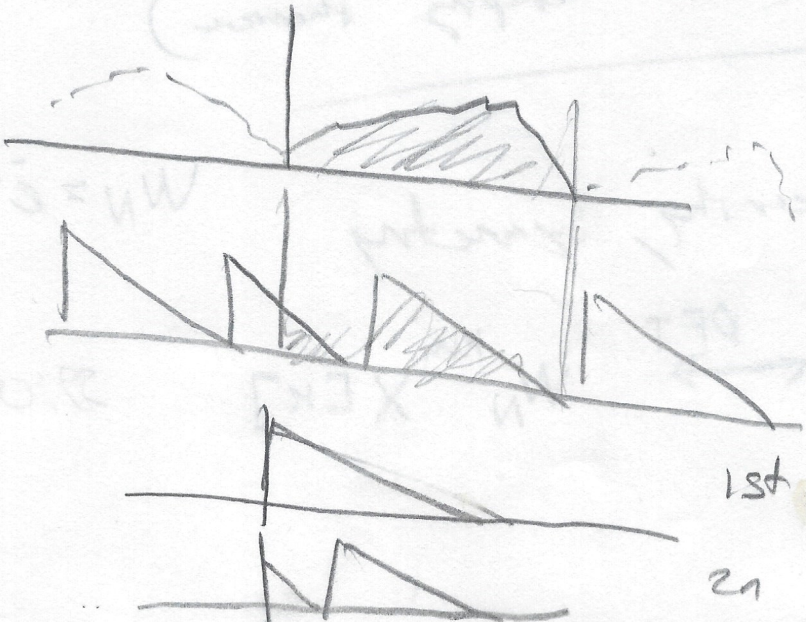


→



In cyclic convolution, both signals are treated as if they are constantly repeating

cyclic convolution



1st

2n

Look at the integral of this area and shift the triangle

3rd

The Idea is that the triangle never falls off the page, it shifts to the left and coARS are comming from the rear

→ So we have a periodic convolution
Signals wrap around in a way they don't for regular convolution

✱ The way it works for the DFT is that the convolution property holds, but in the cyclic convolution sense

---

For the DFT, we have the same kind of convolution multiplication property but they are cyclic

So see that

$$x[n] \circledast h[n] \overset{DFT}{\longleftrightarrow} X[k]\,H[k]$$

Length N signals

→ unsettling because we want regular convolution
to get regular convolution, try zero padding the signal

(11)

How to get the "regular" convolution we need for LTI System?

Suppose that $x[n]$ and $h[n]$ are length $N$ signals, what is $x[n] \circledast h[n]$?

set of $x$'s : $x[0]\ x[1] \cdots x[N-1]$

flip $h$ → run across vectors

$h[2]\ h[1]\ h[0]$

due to the cycling property

$$\begin{array}{cc|ccccc|} & & x[0] & x[1] & \cdots & & x[N-1] \\ h[2] & h[1] & h[0] & h[N-1] & \cdots & h[2] & h[1] \end{array}$$

So $Y[0]$ is the product of all these things

$$\begin{vmatrix} x[0] & x[1] & \cdots & & x[N-1] & \\ h[1] & h[0] & h[N-1] & \cdots & h[3] & h[2] \end{vmatrix} \quad Y[1]$$

$\vdots$

$N$ different produces for $N$ shifts of $h[n]$

$$Y[0] = h[0]x[0] + x[1]h[N-1] + \cdots h[1]x[N-1]$$

$$Y[1] = h[1]x[0] + h[0]x[1] + \cdots h[2]x[N-1]$$

$$Y[N-1] = h[N-1]x[0] + \cdots\cdots h[0]x[N-1]$$

In a way, this is a linear transformation of the input

→ for each of the Y-values, we are getting a combination of the N $x[n]$ values

→ think of as a matrix product

→ write as a LA thing

$$
\begin{bmatrix} y[0] \\ y[1] \\ \vdots \\ y[N-1] \end{bmatrix}
=
\begin{bmatrix}
h[0] & h[N-1] & \cdots & h[1] \\
h[1] & h[0] & & h[2] \\
 & & \ddots & \\
h[N-1] & h[N-2] & h[1] & h[0]
\end{bmatrix}
\begin{bmatrix} x[0] \\ x[1] \\ \vdots \\ x[N-1] \end{bmatrix}
$$

recognize that all of the diagonal values are the same $h[0]$, not just the center values, but those parallel as well

→ rewritten

$$
\begin{bmatrix}
h[0] & h[N-1] & h[N-2] & \cdots & h[2] & h[1] \\
h[1] & h[0] & h[N-1] & h[N-2] & & h[2] \\
h[2] & h[1] & h[0] & h[N-1] & & \\
\vdots & & & & & \\
h[N-1] & h[N-2] & & h[2] & h[1] & h[0]
\end{bmatrix}
$$

⊛ circulant matrix

(16) there are only $N$ uniq values being circulated along the diagonals

→ finally: how to get the desired Linear ~~combinations~~ convolution?

This circulant matrix-vector product corresponds to circular convolution

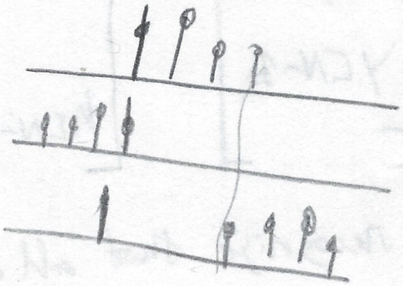
what do we want when we have Linear convolution?

---

consider a four length vector $x[n]$ and a four length impulse response

$x[0] \; ---- \; x[3]$

$h[0] \; ---- \; h[3]$



$y[0] = x[0]h[0]$

$y[1] = x[0]h[1] + x[1]h[0]$

$y[2] = x[0]h[2] + x[1]h[1] + x[2]h[0]$

$$\begin{bmatrix} y[0] \\ \\ \vdots \\ \\ y[6] \end{bmatrix} = \underbrace{\begin{bmatrix} h[0] & 0 & 0 & 0 \\ h[1] & h[0] & 0 & 0 \\ h[2] & h[1] & h[0] & 0 \\ h[3] & h[2] & h[1] & h[0] \\ 0 & h[3] & h[2] & h[1] \\ 0 & 0 & h[3] & h[2] \\ 0 & 0 & 0 & h[3] \end{bmatrix}}_{7 \times 4} \underbrace{\begin{bmatrix} x[0] \\ x[1] \\ x[2] \\ x[3] \end{bmatrix}}_{4 \times 1}$$

← Not square matrix

Similar to circulant matrix, but No wrap arounds modify it as such



$$\begin{bmatrix} y[0] \\ y[1] \\ y[2] \\ y[3] \\ y[4] \\ y[5] \\ y[6] \end{bmatrix} = \left[\begin{array}{cccc|ccc} h[0] & 0 & 0 & 0 & h[3] & h[2] & h[1] \\ h[1] & h[0] & 0 & 0 & 0 & h[3] & h[2] \\ h[2] & h[1] & h[0] & 0 & 0 & 0 & h[3] \\ h[3] & h[2] & h[1] & h[0] & 0 & 0 & 0 \\ 0 & h[3] & h[2] & h[1] & h[0] & 0 & 0 \\ 0 & 0 & h[3] & h[2] & h[1] & h[0] & 0 \\ 0 & 0 & 0 & h[3] & h[2] & h[1] & h[0] \end{array}\right] \begin{bmatrix} x[0] \\ x[1] \\ x[2] \\ x[3] \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

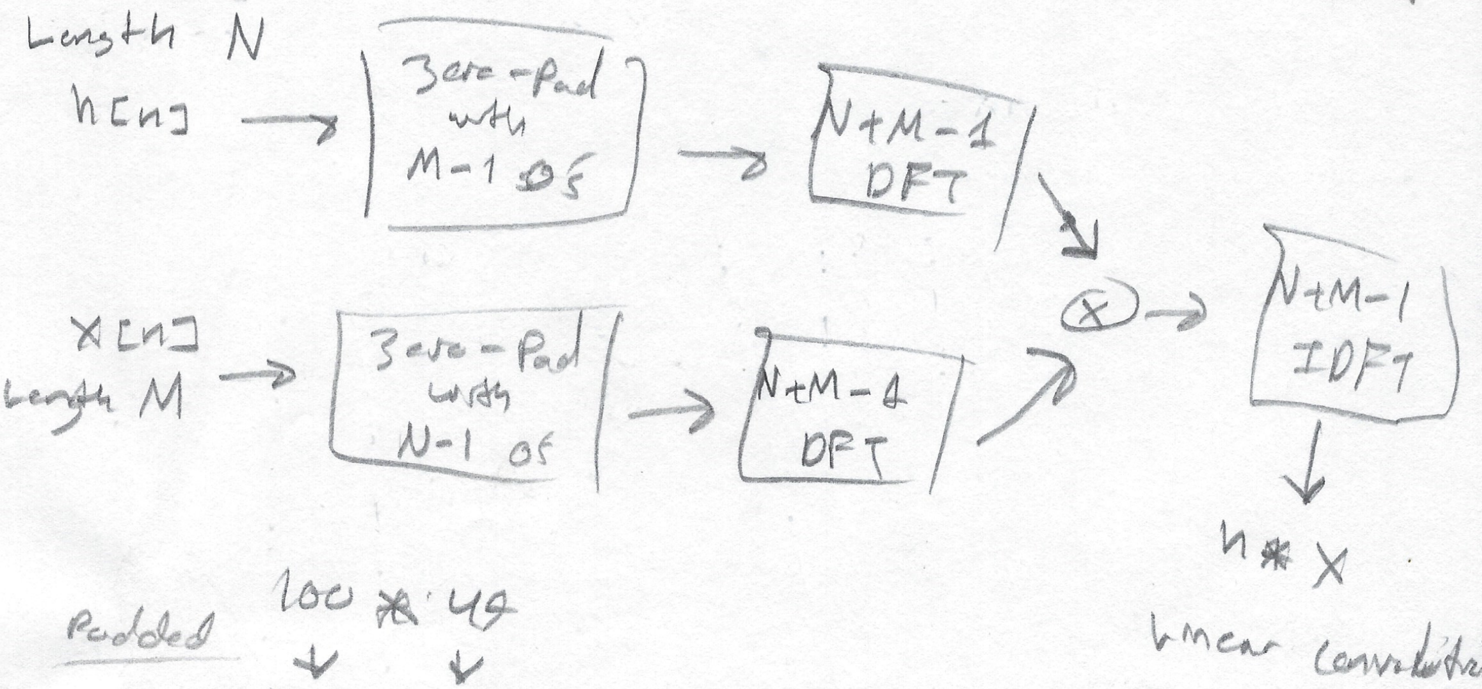
doesn't matter as multiplied by $\emptyset$

Now, this Looks Like a circulant matrix

We can get the Linear convolution we want by zero padding the input and impulse response

---

conclusion, to Do Linear convolution with the DFT




Padded

$$\begin{array}{ccc} 100 & * & 49 \\ \downarrow & & \downarrow \\ 149 & * & 149 \end{array}$$