Section 7.1

- Fourier Coefficients:

$$c_k = \frac{1}{N}\sum_{n=0}^{N-1} x_p(n) e^{-j2\pi k \frac{n}{N}}$$

- $x(n)$ in frequency domain (fourier transform):

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

where $w = 2\pi \frac{k}{N}$

- to obtain discrete samples of X in the frequency domain:

$$X\left(\frac{2\pi k}{N}\right) = \sum_{n=-\infty}^{\infty} x(n) e^{-j2\pi k \frac{n}{N}} \quad , k = 0, 1, \ldots, N-1$$

- meaning $c_k = \frac{1}{N} X\left(\frac{2\pi}{N}k\right)$

and

$$x_p(n) = \frac{1}{N}\sum_{k=0}^{N-1} X\left(\frac{2\pi}{N}k\right) e^{j2\pi k \frac{n}{N}}$$

reconstruction of signal x based on N samples of $X(w)$

- discrete fourier transform:

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{-j2\pi k \frac{n}{N}}$$

where $\frac{2\pi k}{N}$ is the current angular frequency of sampling (radians/sample)

- $\sum_{n=0}^{N-1}$ means we are stepping through N different points across one sampling period

- inverse discrete fourier transform (to recover the sampled and transformed signal $x(n)$)

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) e^{j 2\pi k \frac{n}{N}}$$

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

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

meaning $W_N^{kn} = e^{-2\pi k \frac{n}{N}}$

- DFT matrix operations:

- $x_N = \begin{bmatrix} x(0) \\ x(1) \\ x(2) \\ x(N-1) \end{bmatrix}$ → represents N samples of x in one period

- $X_N = \begin{bmatrix} X(0) \\ X(1) \\ \vdots \\ X(N-1) \end{bmatrix}$ → represents N samples of X (frequency) analysis where $0 < W < 2\pi$ and the frequency samples are equally spaced by $\frac{2\pi}{N}$ $\frac{\text{radians}}{\text{sample}}$

- thus we can construct the $N \times N$ DFT Matrix:

$$\begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & W_N^1 & W_N^{2(1)} & \cdots & W_N^{(N-1)1} \\ 1 & W_N^2 & W_N^{2(2)} & \cdots & W_N^{2(N-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & W_N^{N-1} & W_N^{2(N-1)} & \cdots & W_N^{(N-1)(N-1)} \end{bmatrix}$$

↓ N samples of a given signal

→ N sampling rates of a

N sampling rates of a given signal $0 \to N-1$

- meaning the DFT can be expressed as: $X_N = W_N x_N$
- IDFT = $x_N = W_N^{-1} X_N$

Example 7.1.3

- compute DFT of $x(n) = (0, 1, 2, 3)$

$N = 4$

$$W_4 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & W_4^1 & W_4^2 & W_4^3 \\ 1 & W_4^2 & W_4^4 & W_4^6 \\ 1 & W_4^3 & W_4^6 & W_4^9 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix}$$

$$X_4 = W_4 \cdot x_4 = \begin{bmatrix} 6 \\ 2j+2 \\ -2 \\ -2-2j \end{bmatrix}$$

Section 7.2

- properties of the DFT
- if $x(n)$ and $X(n)$ are N-point DFT pairs:
  - $x(n+N) = x(n)$
  - $X(n+N) = X(n)$ → periodicity
- linearity:

if $x_1(n) \overset{DFT}{\underset{N}{\longleftrightarrow}} X_1(k)$

and $x_2(n) \overset{DFT}{\underset{N}{\longleftrightarrow}} X_2(k)$

then for any real/complex valued constants a_1 and a_2

$$a_1 x_1(n) + a_2 x_2(n) \overset{DFT}{\underset{N}{\longleftrightarrow}} a_1 X_1(k) + a_2 X_2(k)$$

- circular symmetry:
  - $x_p(n) = \sum_{I=-\infty}^{\infty} x(n - IN)$ → x can be extend periodically by its fundamental period (N) and it will still be equivalent
  - meaning if the sequence $x_p(n)$ is shifted by k:
    $x'_p(n) = x_p(n-k) = \sum_{I=-\infty}^{\infty} x(n-k-IN)$
- real valued sequences:
  $X(N-k) = X(-k)$ and $x_I(n) = 0$
- real and even sequences:
  $x(n) = x(N-n)$
  so $X(k) = \sum_{n=0}^{N-1} x(n) \cos \frac{2\pi kn}{N}$
- real and odd sequences:
  $x(n) = -x(N-n)$
  meaning $X_R(k) = 0$
  so $X(k) = -j \sum_{n=0}^{N-1} x(n) \sin \frac{2\pi kn}{N}$
- Multiplication of 2 DFTs and Circular Convolution
  - 2 finite duration sequences of length N: $x_1(n)$ and $x_2(n)$
  - DFTs:
    $X_1(k) = \sum_{n=0}^{N-1} x_1(n) W_N^{kn}$
    $X_2(k) = \sum_{n=0}^{N-1} x_2(n) W_N^{kn}$
  - multiply the two, resulting in $X_3(k)$

$\hookrightarrow X_3(k) = X_1(k) + X_2(k)$

What is the relationship between $x_3(n)$ and $x_1(n)$ and $x_2(n)$

IDFT:

$$x_3(m) = \frac{1}{N}\sum_{k=0}^{N-1} X_3(k)\, e^{j2\pi k \frac{m}{N}}$$

$$= \frac{1}{N}\sum_{k=0}^{N-1} X_1(k) X_2(k)\, e^{j2\pi k \frac{m}{N}}$$

Substituting in expressions for X_1 and X_2:

$$x_3(m) = \frac{1}{N}\sum_{k=0}^{N-1}\left[\sum_{n=0}^{N-1} x_1(n) e^{-j2\pi k \frac{n}{N}}\right]\left[\sum_{l=0}^{N-1} x_2(n) e^{-j2\pi k l/N}\right] e^{j2\pi k m/N}$$

$$= \frac{1}{N}\sum_{n=0}^{N-1} x_1(n) \sum_{l=0}^{N-1} x_2(l)\left[\sum_{k=0}^{N-1} e^{j2\pi k(m-n-l)/N}\right]$$

$$\Rightarrow x_3(m) = \sum_{n=0}^{N-1} x_1(n)\, x_2((m-n))_N \qquad m = 0, 1, \ldots, N-1$$

Example 7.2.2

$x_1(n) = (2, 1, 2, 1)$ $\qquad N = 4$

$x_2(n) = (1, 2, 3, 4)$ $\qquad k = 0, 1, 2, 3$

$X_1(k) = \sum_{n=0}^{N-1} x_1(n) e^{-j2\pi k \frac{n}{N}}$ $\qquad e^{-2\pi k \frac{n}{N}} = \cos(2\pi k \frac{n}{N}) - j\sin(2\pi k \frac{n}{N})$

$X_1(0) = 6 \quad X_1(1) = 0 \quad X_1(2) = 2 \quad X_1(3) = 0$

$X_2(k) = \sum_{n=0}^{N-1} x_2(n) e^{-j2\pi k \frac{n}{N}}$

$X_2(0) = 10 \quad X_2(1) = 2j - 2 \quad X_2(2) = -2 \quad X_2(3) = -2-2j$

$X_3(k) = X_1(k) \cdot X_2(k)$

$X_3(0) = 60 \quad X_3(1) = 0 \quad X_3(2) = -4 \quad X_3(3) = 0$

$x_3(n) = \frac{1}{N}\sum_{k=0}^{3} X_3(k) e^{2\pi j k \frac{n}{4}}$ $\qquad k=2: -4(e^{\pi j n})$

$$= \frac{1}{4}\left(60 - 4\cos(\pi n)\right)$$

$x_3(0) = 14 \quad x_3(1) = 16 \quad x_3(2) = 14 \quad x_3(3) = 16$

$X_3(0) = 14 \quad X_3(1) = 16 \quad X_3(2) = 14 \quad X_3(3) = 16$

- thus circular convolution:

if $x_1(n) \overset{DFT}{\underset{N}{\longleftrightarrow}} X_1(k)$

and $x_2(n) \overset{DFT}{\underset{N}{\longleftrightarrow}} X_2(k)$

then $x_1(n) \, Ⓝ \, x_2(n) \overset{DFT}{\underset{N}{\longleftrightarrow}} X_1(k) X_2(k)$

notation for circular convolution

→ Circular convolution is a way to compute the product of signal frequencies in the time domain

↳ can be thought of as (in a continuous context), taking a window and panning over the region, generating a new output (similar to what happens in a CNN (discrete))

↳ convolution example: $X_1(w)$ is the DFT of the original signal, $X_2(w)$ is a filter, making $x_3(n)$ the result of applying filter $X_2(w)$ onto the signal $x_1(n)$

- Time Reversal of a sequence
  - given a N-point sequence $x(n)$, its reversal:
    $x(-n) = x(N-n) \overset{DFT}{\underset{N}{\longleftrightarrow}} X(-k) = X(N-k)$
- Circular shift (translation of time-domain signal by l):
  $x(n-l) = X(k) e^{-2\pi k l / N}$
- Circular Correlation
  - given 2 n point sequences and their DFTs (both complex)
    $r_{xy}(l) = \sum_{n=0}^{N-1} x(n) y^*((n-l))_N$

    or: $r_{xy}(l) = x(n) \, Ⓝ \, y^*(n)$

    given that $x(n) \, Ⓝ \, x_2(n) \overset{DFT}{\underset{N}{\longleftrightarrow}} X_1(k) X_2(k)$

    $\rightarrow r_{xy}(l) \overset{DFT}{\underset{N}{\longleftrightarrow}} X(k) Y^*(k) = R(k)$

- Question for John:
- Conceptually, what is the circular correlation?
  - → I understand that mathematically, its just a circular convolution, but using the conjugate of a signal instead of the actual signal, but what for?
  - What does the conjugate of a signal represent?
  - What is this used for?
- John's answer:
  - not fully sure, but could be as easy as the correlation between two signals similar to how the dot product is the "correlation" between 2 vectors

as the correlation between two signals similar to how the dot product is the "correlation" between 2 vectors

- conjugate is just a 90 degree phase shift of your original signal

Section 7.3

- linear filtering methods based on the DFT
- Frequency Domain Filtering:
  - $Y(w) = X(w)H(w)$
    - where X is the unprocessed signal in the frequency domain
    - H is the system's (ex: filter's) response when excited by a signal, X
    - and Y is the processed signal in the frequency domain
  - → tie-back to circular convolutions:
    - $x(n) \; (N) \; h(n) \xleftrightarrow[N]{DFT} X(w)Y(w)$
    - meaning $y(n) = \sum_{l=0}^{N-1} h(l)x(n-l)$

Question for John:

→ FIR filter of length M:
  → What does it mean to express a filter in the time domain?
→ what is the importance of the length of the filter in the time domain?
→ why $M+L-1$ length?
→ why doesn't 0 padding impact the DFT?
  → ohhh because coefficient is 0? → $X(k) = \sum_{n=0}^{7} x(n) e^{-2\pi kn/8}$
  ↑ 0 for padded values

John's answer:

→ thinking back in Q&A, the time domain expression of a signal represents the impulse response to a given stimulus → can be found by solving the differential equation equal to a given stimulus

- using the DFT in linear filtering:
  → since $h(n)$ and $x(n)$ are finite duration, their circular convolution will have length $len(x)+len(h)-1$
  - therefore DFT of size $N \geq L+M-1$ is required
    → this is possible by 0 padding the original signal
    → for efficient computing (probably explained why in chapter 8), 2^n point DFTs are optimal
    → when not enough points ($N \leq M+N-1$), values are aliased

chapter 8), 2 point DFTs are optimal

→ when not enough points ($N \le M+N-1$), values are aliased together

- methods for long data analysis
  - → each filter is length M
  - input blocks segmented into L points
  - $L >> M$
  - overlap-save method
    - each input block is $N = L+M-1$
    - each block contains the last $M-1$ and L new data points
    - FIR filter padded by $L-1$ zeros and stored
    - since data is of length N, first $M-1$ points are corrupted by aliasing → shown in example 7.3.2
    - → to avoid loss due to aliasing, last $M-1$ points are stored and used as the beginning of the next segment
  - Overlapp-add method
    - → input block size L, $N = L+M-1$ point DFT
    - add $M-1$ zeros to each block
    - since each block has $M-1$ zeros at the end, the last $M-1$ points must be overlapped to the first $M-1$ points of the next signal