ECS455: Chapter 5

OFDM

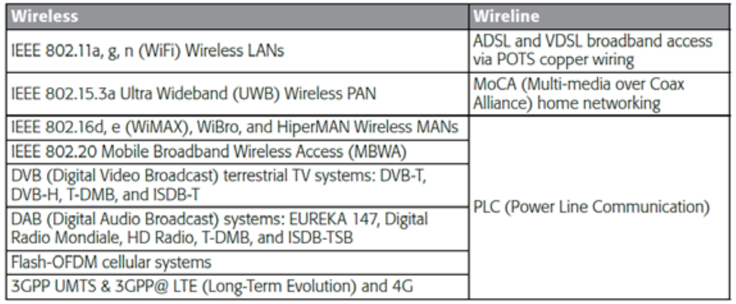
Dr.Prapun Suksompong
prapun.com/ecs455

Office Hours:
BKD, 6th floor of Sirindhralai building

Tuesday	**14:20-15:20**
Wednesday	**14:20-15:20**
Friday	**9:15-10:15**

OFDM Applications

- 802.11 **Wi-Fi**: a/g/n/ac versions
- **DVB-T** (Digital Video Broadcasting — Terrestrial)
  - terrestrial digital TV broadcast system used in most of the world outside North America
- DMT (the standard form of **ADSL** - Asymmetric Digital Subscriber Line)
- **WiMAX, LTE (OFDMA)**



Wireless	Wireline
IEEE 802.11a, g, n (WiFi) Wireless LANs	ADSL and VDSL broadband access via POTS copper wiring
IEEE 802.15.3a Ultra Wideband (UWB) Wireless PAN	MoCA (Multi-media over Coax Alliance) home networking
IEEE 802.16d, e (WiMAX), WiBro, and HiperMAN Wireless MANs	PLC (Power Line Communication)
IEEE 802.20 Mobile Broadband Wireless Access (MBWA)	
DVB (Digital Video Broadcast) terrestrial TV systems: DVB-T, DVB-H, T-DMB, and ISDB-T	
DAB (Digital Audio Broadcast) systems: EUREKA 147, Digital Radio Mondiale, HD Radio, T-DMB, and ISDB-TSB	
Flash-OFDM cellular systems	
3GPP UMTS & 3GPP@ LTE (Long-Term Evolution) and 4G	

OFDM: Overview

- Let $\underline{\mathbf{S}} = (S_1, S_2, \ldots, S_N)$ contains the information symbols.

$\underline{\mathbf{S}}$

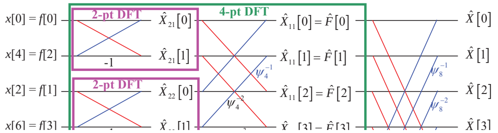
IFFT

Inverse fast Fourier transform

FFT

Fast Fourier transform

$\hat{\underline{\mathbf{S}}}$



ECS455: Chapter 5

OFDM

5.1 Implementation: DFT and FFT

Dr.Prapun Suksompong
prapun.com/ecs455

Office Hours:
BKD, 6th floor of Sirindhralai building

Tuesday	**14:20-15:20**
Wednesday	**14:20-15:20**
Friday	**9:15-10:15**

OFDM and CDMA

- CDMA's key equation $\underline{\mathbf{s}} = \frac{1}{N}(\underline{\mathbf{s}}\mathbf{C})\mathbf{C}^T$
  - All the rows of $\mathbf{C}$ are orthogonal
- Key property of $\mathbf{C}$:

$$\mathbf{C}\mathbf{C}^T = N\mathbf{I}.$$

- For sync. CDMA, we use the **Hadamard matrix $\mathbf{H}_N$**.
- For OFDM, we use **DFT matrix $\boldsymbol{\Psi}_N$**.
  - The matrix is complex-valued.

Discrete Fourier Transform (DFT)

Here, we work with N-point signals (finite-length sequences (vectors) of length N) in both time and frequency domain.

$$\vec{\mathbf{x}} = \begin{pmatrix} x[0] \\ x[1] \\ \vdots \\ x[N-1] \end{pmatrix} \longrightarrow \boxed{\text{DFT}} \longrightarrow \vec{\mathbf{X}} = \begin{pmatrix} X[0] \\ X[1] \\ \vdots \\ X[N-1] \end{pmatrix}$$

$$\vec{\mathbf{X}} = \mathrm{DFT}\{\vec{\mathbf{x}}\} = \boldsymbol{\Psi}_N\vec{\mathbf{x}}$$

DFT matrix $\boldsymbol{\Psi}_N$

$$\boldsymbol{\Psi}_N = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & \psi_N^{-1} & \psi_N^{-2} & \cdots & \psi_N^{-(N-1)} \\ 1 & \psi_N^{-2} & \psi_N^{-4} & \cdots & \psi_N^{-2(N-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \psi_N^{-(N-1)} & \psi_N^{-2(N-1)} & \cdots & \psi_N^{-(N-1)(N-1)} \end{bmatrix}$$

Element on the pth row and qth column is given by

$$\boxed{\psi_N^{-(p-1)(q-1)} \text{ where } \psi_N = e^{j\frac{2\pi}{N}}}$$

Note that the "-1" are there because we start from row 1 and column 1 (not from row 0 and column 0).

Key Property: $\boxed{\boldsymbol{\Psi}_N^{-1} = \frac{1}{N}\boldsymbol{\Psi}_N^*}$ $\Rightarrow$ $\frac{1}{\sqrt{N}}\boldsymbol{\Psi}_N$ is a unitary matrix

$\Downarrow$

$$\frac{1}{N}\boldsymbol{\Psi}_N^*\vec{\mathbf{X}} = (\boldsymbol{\Psi}_N)^{-1}\vec{\mathbf{X}} = \mathrm{IDFT}\{\vec{\mathbf{X}}\} = \vec{\mathbf{x}} \underset{\text{IDFT}}{\overset{\text{DFT}}{\rightleftarrows}} \vec{\mathbf{X}} = \mathrm{DFT}\{\vec{\mathbf{x}}\} = \boldsymbol{\Psi}_N\vec{\mathbf{x}}$$

Example: $N = 2$

- ψ_2
- $\boldsymbol{\Psi}_2$
- Suppose $\vec{\mathbf{x}} = \binom{2}{5} \rightarrow \boxed{\text{DFT}} \rightarrow \vec{\mathbf{X}} =$

```
>> fft([2 5])
ans =
     7    -3
>> ifft([7 -3])
ans =
     2     5
```

Connection to CDMA

- The rows of $\boldsymbol{\Psi}_N$ are orthogonal. So are the columns.
- Proof: Let $\underline{\mathbf{r}}^{(k)}$ be the k^{th} row of $\boldsymbol{\Psi}_N$.

$$\left\langle \underline{\mathbf{r}}^{(k_1)}, \underline{\mathbf{r}}^{(k_2)} \right\rangle = \sum_{q=1}^{N} \psi_N^{-(k_1-1)(q-1)} \left(\psi_N^{-(k_2-1)(q-1)} \right)^* = \sum_{q=1}^{N} \psi_N^{-(k_1-1)(q-1)} \psi_N^{(k_2-1)(q-1)}$$

$$= \sum_{q=1}^{N} \left(\psi_N^{(k_2-k_1)} \right)^{(q-1)} = \sum_{q=0}^{N-1} \left(\psi_N^{(k_2-k_1)} \right)^{q}$$

$$= \begin{cases} \dfrac{1-\psi_N^{(k_2-k_1)N}}{1-\psi_N^{(k_2-k_1)}} = \dfrac{1-\left(e^{j\frac{2\pi}{N}} \right)^{(k_2-k_1)N}}{1-\psi_N^{(k_2-k_1)}} = \dfrac{1-1}{1-\psi_N^{(k_2-k_1)}} = 0, & k_1 \neq k_2, \\ \displaystyle\sum_{q=0}^{N-1} (1)^q = N, & k_1 = k_2. \end{cases}$$

- So, $\boldsymbol{\Psi}_N$ “replaces” the role of $\mathbf{H}_N$ in CDMA.

Discrete Fourier Transform (DFT)

Matrix form:

$$\boxed{\frac{1}{N} \boldsymbol{\Psi}_N^* \vec{\mathbf{X}} = \text{IDFT}\left\{ \vec{\mathbf{X}} \right\} = \vec{\mathbf{x}} \xrightleftharpoons[\text{IDFT}]{\text{DFT}} \vec{\mathbf{X}} = \text{DFT}\left\{ \vec{\mathbf{x}} \right\} = \boldsymbol{\Psi}_N \vec{\mathbf{x}}}$$

Pointwise form:

$$\frac{1}{N} \sum_{k=0}^{N-1} X[k] \psi_N^{nk} = \underset{0 \le n < N}{x[n]} \xrightleftharpoons[\text{IDFT}]{\text{DFT}} \underset{0 \le k < N}{X[k]} = \sum_{n=0}^{N-1} x[n] \psi_N^{-nk}$$

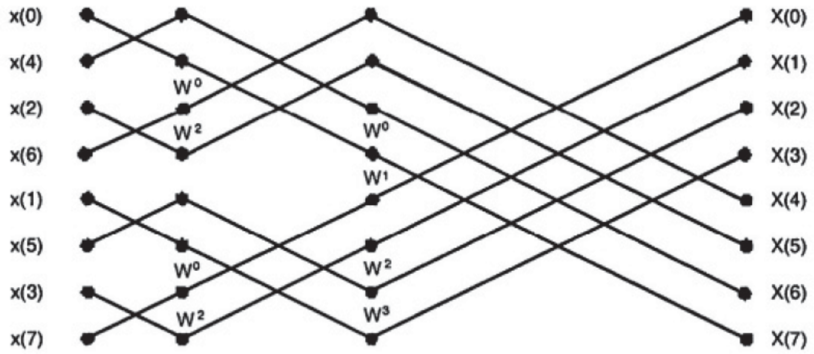
or, equivalently,

$$\boxed{\frac{1}{N} \sum_{n=0}^{N-1} X[k] e^{jnk\frac{2\pi}{N}} = \underset{0 \le n < N}{x[n]} \xrightleftharpoons[\text{IDFT}]{\text{DFT}} \underset{0 \le k < N}{X[k]} = \sum_{n=0}^{N-1} x[n] e^{-jnk\frac{2\pi}{N}}}$$

Comparison with Fourier transform

$$x(t) = \int_{-\infty}^{\infty} X(f) e^{j2\pi ft} df \xrightleftharpoons[\mathcal{F}^{-1}]{\mathcal{F}} x(f) = \int_{-\infty}^{\infty} X(t) e^{-j2\pi ft} dt$$

Efficient Implementation: (I)FFT



[Bahai, 2002, Fig. 2.9]

An N-point FFT requires only on the order of $N\log N$ multiplications, rather than N^2 as in a straightforward computation.

FFT

- The history of the FFT is complicated.
- As with many discoveries and inventions, it arrived before the (computer) world was ready for it.
- Usually done with N a power of two.
  - Very efficient in terms of computing time
  - Ideally suited to the binary arithmetic of digital computers.
  - Ex: From the implementation point of view it is better to have, for example, a FFT size of 1024 even if only 600 outputs are used than try to have another length for FFT between 600 and 1024.

References: E. Oran Brigham, *The Fast Fourier Transform*, Prentice-Hall, 1974.

OFDM with Memoryless Channel

$$h(t)=\beta\delta(t)$$

$\left[\text{should be } h(t)=\beta\delta(t-\tau)\right]$

$$r(t)=h(t)*s(t)+w(t)=\beta s(t)+w(t)$$

Additive white Gaussian noise

Sample every T_s/N

$$r[n]=\beta s[n]+w[n]$$

FFT

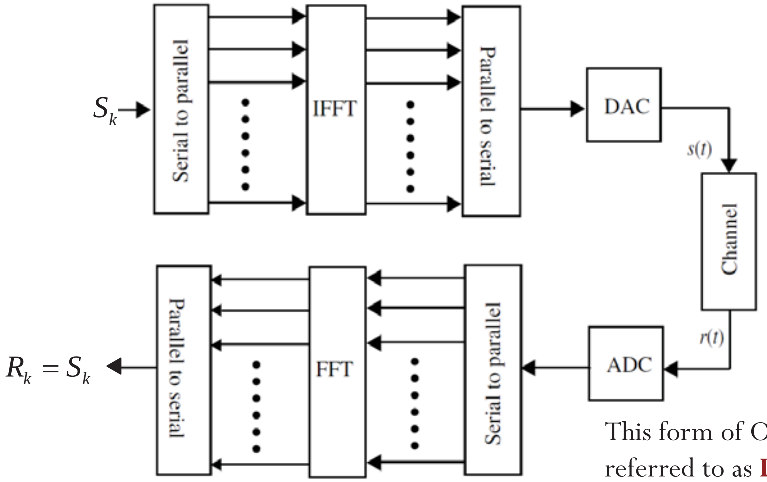
$$s[n]=\sqrt{N}\,\mathrm{IFFT}\{S\}[n]$$

$$R_k=\frac{1}{\sqrt{N}}\mathrm{FFT}\{\mathbf{r}\}[n]=\beta S_k+\frac{1}{\sqrt{N}}W_k$$

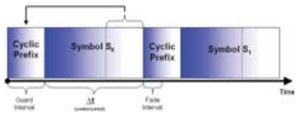
Sub-channel are independent.

(No ICI)

OFDM implementation by IFFT/FFT



This form of OFDM is often referred to as **Discrete Multi-Tone (DMT)**.



ECS455: Chapter 5

OFDM

5.2 Cyclic Prefix (CP)

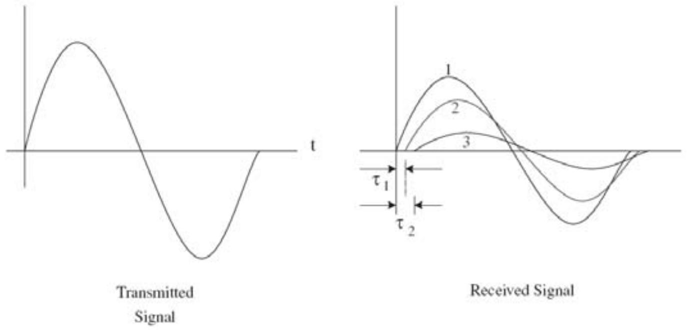
Dr.Prapun Suksompong
prapun.com/ecs455

Multipath Propagation

- In a wireless mobile communication system, a transmitted signal propagating through the wireless channel often encounters multiple reflective paths until it reaches the receiver
- We refer to this phenomenon as **multipath propagation** and it causes fluctuation of the amplitude and phase of the received signal.
- We call this fluctuation **multipath fading**.

Cyclic Prefix: Motivation

- To reduce the ISI, add **guard interval** larger than that of the estimated delay spread.
- If the guard interval is left empty, multipath fading will destroy orthogonality of the sub-carriers, i.e., **ICI** (inter-channel interference) still exists.
- **Solution**: To prevent **both** the **ISI** as well as the **ICI**, OFDM symbol is **cyclically extended** into the guard interval.



Channel with Finite Memory

Discrete time baseband model:

$$y[n] = \{h * s\}[n] + w[n] = \sum_{m=0}^{v} h[m]s[n-m] + w[n]$$

[Tse Viswanath, 2005, Sec. 2.2.3]

where $h[n] = 0$ for $n < 0$ and $n > v$

$$w[n] \overset{i.i.d.}{\sim} \mathcal{CN}(0, N_0)$$

We will assume that $v \ll N$

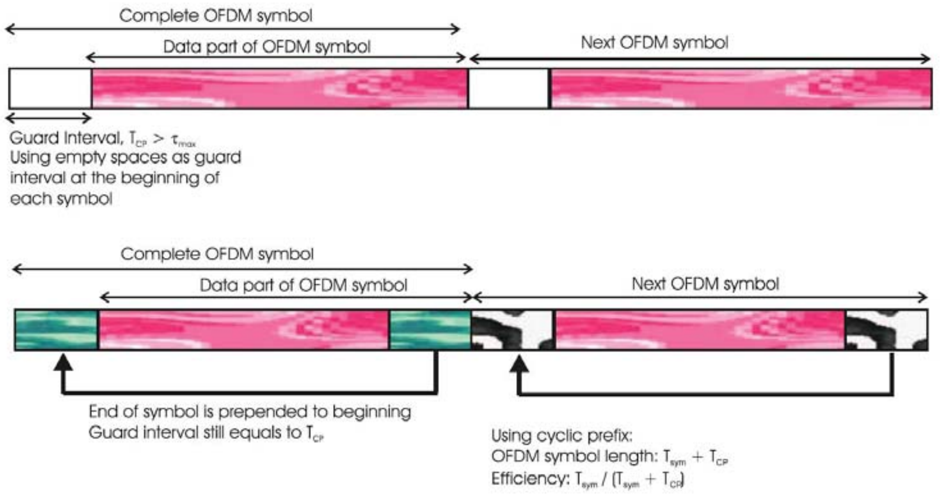
Remarks:

$Z = X + jY$ is a ***complex Gaussian*** if X and Y are jointly Gaussian.

If X, Y is i.i.d. $\mathcal{N}(0,\sigma^2)$, then $Z = X + iY \sim \mathcal{CN}(0,\sigma_Z^2)$ where $\sigma_Z^2 = 2\sigma^2$ with

$$f_Z(z) = f_{X,Y}(\mathrm{Re}\{z\}, \mathrm{Im}\{z\}) = \frac{1}{\pi\sigma_Z^2} e^{-\frac{|z|^2}{\sigma_Z^2}}.$$

Cyclic Prefix



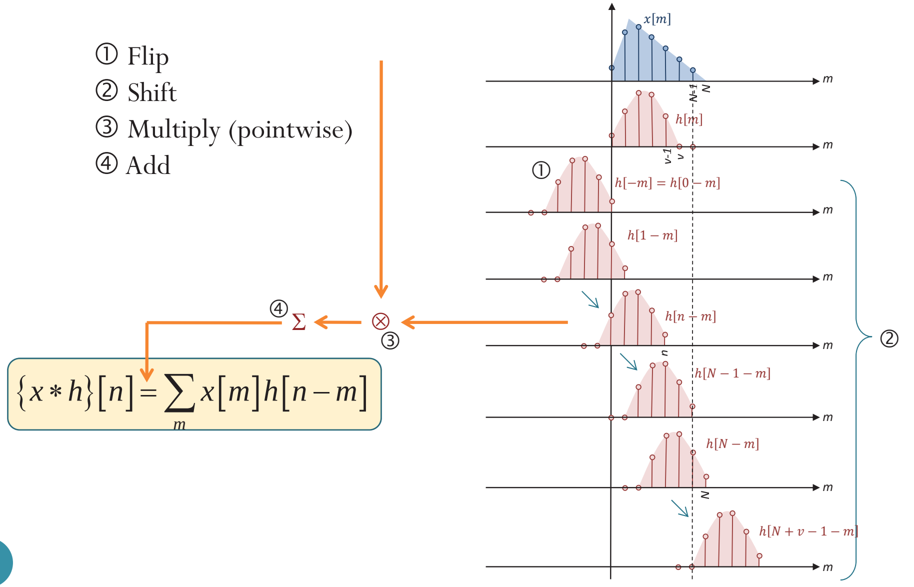
Guard Interval, $T_{CP} > \tau_{max}$
Using empty spaces as guard
interval at the beginning of
each symbol

End of symbol is prepended to beginning
Guard interval still equals to T_{CP}

Using cyclic prefix:
OFDM symbol length: $T_{sym} + T_{CP}$
Efficiency: $T_{sym} / (T_{sym} + T_{CP})$

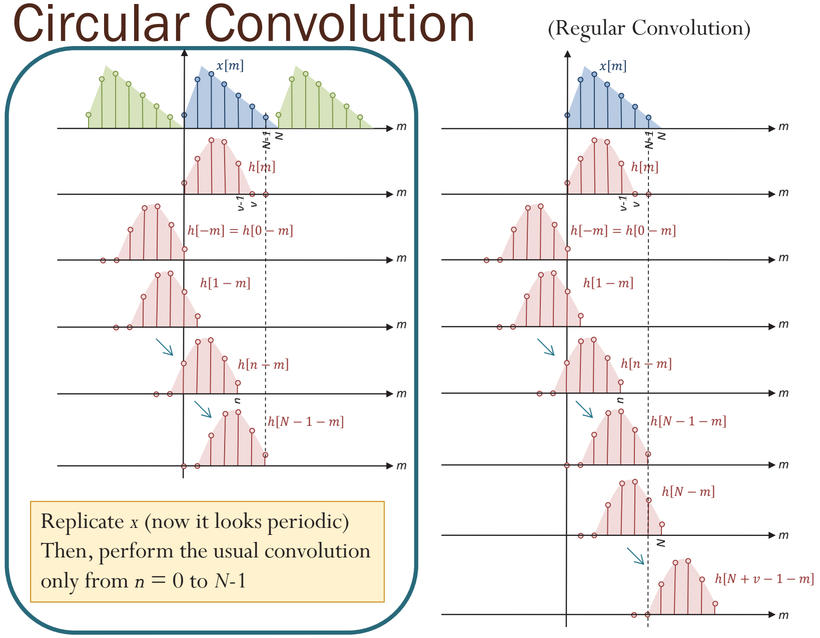
Recall: Convolution

① Flip
② Shift
③ Multiply (pointwise)
④ Add



$$\{x * h\}[n] = \sum_{m} x[m]h[n-m]$$

Circular Convolution



Replicate x (now it looks periodic)
Then, perform the usual convolution
only from $n = 0$ to N-1

Circular Convolution: Examples 1

Find

$$[1\quad 2\quad 3] * [4\quad 5\quad 6]$$

```
>> conv([1,2,3],[4,5,6])
ans =
     4    13    28    27    18
```

$$[1\quad 2\quad 3] \circledast [4\quad 5\quad 6]$$

```
>> cconv([1,2,3],[4,5,6],3)
ans =
    31    31    28
```

$$[1\quad 2\quad 3\quad 0\quad 0] \circledast [4\quad 5\quad 6\quad 0\quad 0]$$

```
>> cconv([1,2,3,0,0],[4,5,6,0,0],5)
ans =
    4.0000   13.0000   28.0000   27.0000   18.0000
```

Discussion

- ***Regular convolution*** of an N_1–point vector and an N_2–point vector gives $(N_1+N_2\text{-}1)$-point vector.
- ***Circular convolution*** is performed between two equal-length vectors. The results also has the same length.
- Circular convolution can be used to find regular convolution by **zero-padding**.
  - Zero-pad the vectors so that their length is $N_1+N_2\text{-}1$.
  - Example:

$$[1\quad 2\quad 3\quad 0\quad 0] \circledast [4\quad 5\quad 6\quad 0\quad 0] = [1\quad 2\quad 3] * [4\quad 5\quad 6]$$

Circular Convolution in Communication

- We want the receiver to obtain the circular convolution of the signal (channel input) and the channel.
- Q: Why?
- A:
  - **CTFT**: **convolution** in time domain corresponds to **multiplication** in frequency domain.
    - This fact does not hold for DFT.
  - **DFT**: circular **convolution** in (discrete) time domain corresponds to **multiplication** in (discrete) frequency domain.
    - We want to have multiplication in frequency domain.
    - So, we want circular convolution and not the regular convolution.
- Problem: Real channel does regular convolution.
- Solution: With **cyclic prefix**, regular convolution can be used to create circular convolution.

Example 2

$$[1\ \ -2\ \ 3\ \ 1\ \ 2]\circledast[3\ \ 2\ \ 1\ \ 0\ \ 0]=?$$

Let's look closer at how we carry out the circular convolution operation. Recall that we replicate the *x* and then perform the regular convolution (for *N* points)

Solution:

1 −2 3 1 2 1 −2 3 1 2 1 −2 3 1 2

0 0 1 2 3 — $1\times1+2\times2+1\times3=1+4+3=8$

0 0 1 2 3 — $2\times1+1\times2+(-2)\times3=2+2-6=-2$

0 0 1 2 3 — $1\times1+(-2)\times2+3\times3=1-4+9=6$

0 0 1 2 3 — $(-2)\times1+3\times2+1\times3=-2+6+3=7$

0 0 1 2 3 — $3\times1+1\times2+2\times3=3+2+6=11$

$$[1\ \ -2\ \ 3\ \ 1\ \ 2]\circledast[3\ \ 2\ \ 1\ \ 0\ \ 0]=\underbrace{[8\ \ -2\ \ 6\ \ 7\ \ 11]}$$

Goal: Get these numbers using regular convolution

Example 2

$$[1\ \ -2\ \ 3\ \ 1\ \ 2]\circledast[3\ \ 2\ \ 1\ \ 0\ \ 0]=?$$

Observation: We don't need to replicate the *x* indefinitely. Furthermore, when *h* is shorter than *x*, we need only a part of one replica.

Not needed in the calculation

1 −2 3 | 1 2 1 −2 3 1 2 | 1 −2 3 1 2

0 0 1 2 3 — $1\times1+2\times2+1\times3=1+4+3=8$

0 0 1 2 3 — $2\times1+1\times2+(-2)\times3=2+2-6=-2$

0 0 1 2 3 — $1\times1+(-2)\times2+3\times3=1-4+9=6$

0 0 1 2 3 — $(-2)\times1+3\times2+1\times3=-2+6+3=7$

0 0 1 2 3 — $3\times1+1\times2+2\times3=3+2+6=11$

$$[1\ \ -2\ \ 3\ \ 1\ \ 2]\circledast[3\ \ 2\ \ 1\ \ 0\ \ 0]=[8\ \ -2\ \ 6\ \ 7\ \ 11]$$

Example 2

Try this: use only the necessary part of the replica and then convolute (regular convolution) with the channel.

$$[\underbrace{1\ \ 2}\ \ 1\ \ -2\ \ 3\ \ 1\ \ 2]*[3\ \ 2\ \ 1]=?$$

Copy the **last ν samples** of the symbols to the **beginning** of the symbol. This partial replica is called the **cyclic prefix**.

1 2 1 −2 3 1 2

1 2 3 — $1\times3=3$

1 2 3 — $1\times2+2\times3=2+6=8$

1 2 3 — $1\times1+2\times2+1\times3=1+4+3=8$

1 2 3 — $2\times1+1\times2+(-2)\times3=2+2-6=-2$

1 2 3 — $1\times1+(-2)\times2+3\times3=1-4+9=6$

1 2 3 — $(-2)\times1+3\times2+1\times3=-2+6+3=7$

1 2 3 — $3\times1+1\times2+2\times3=3+2+6=11$

1 2 3 — $1\times1+2\times2=1+4=5$

1 2 3 — $2\times1=2$

Junk!

Example 2

- We now know that

$$[\underbrace{1\ \ 2}_{\text{Cyclic Prefix}}\ \ 1\ \ -2\ \ 3\ \ 1\ \ 2]*[3\ \ 2\ \ 1]=[3\ \ 8\ \ \underbrace{8\ \ -2\ \ 6\ \ 7\ \ 11}_{[1\ \ -2\ \ 3\ \ 1\ \ 2]\circledast[3\ \ 2\ \ 1\ \ 0\ \ 0]}\ \ 5\ \ 2]$$

- Similarly, you may check that

$$[\underbrace{-2\ \ 1}_{\text{Cyclic Prefix}}\ \ 2\ \ 1\ \ -3\ \ -2\ \ 1]*[3\ \ 2\ \ 1]=[-6\ \ -1\ \ \underbrace{6\ \ 8\ \ -5\ \ -11\ \ -4}_{[2\ \ 1\ \ -3\ \ -2\ \ 1]\circledast[3\ \ 2\ \ 1\ \ 0\ \ 0]}\ \ 0\ \ 1]$$

Example 3

- We know, from Example 2, that

[1 2 1 -2 3 1 2] * [3 2 1] = [3 8 8 -2 6 7 11 5 2]

And that

[-2 1 2 1 -3 -2 1] * [3 2 1] = [-6 -1 6 8 -5 -11 -4 0 1]

- Check that

[1 2 1 -2 3 1 2 0 0 0 0 0 0 0] * [3 2 1]
= [3 8 8 -2 6 7 11 5 2 0 0 0 0 0 0 0]

and

[0 0 0 0 0 0 0 -2 1 2 1 -3 -2 1] * [3 2 1]
= [0 0 0 0 0 0 0 -6 -1 6 8 -5 -11 -4 0 1]

Example 4

- We know that

[1 2 1 -2 3 1 2] * [3 2 1] = [3 8 8 -2 6 7 11 5 2]
[-2 1 2 1 -3 -2 1] * [3 2 1] = [-6 -1 6 8 -5 -11 -4 0 1]

- Using Example 3, we have

[1 2 1 -2 3 1 2 -2 1 2 1 -3 -2 1] * [3 2 1]
= ([1 2 1 -2 3 1 2 0 0 0 0 0 0 0]
+[0 0 0 0 0 0 0 -2 1 2 1 -3 -2 1]) * [3 2 1]
= [3 8 8 -2 6 7 11 5 2 0 0 0 0 0 0 0]
+[0 0 0 0 0 0 0 -6 -1 6 8 -5 -11 -4 0 1]
= [3 8 8 -2 6 7 11 -1 1 6 8 -5 -11 -4 0 1]

Putting results together…

- Suppose $\underline{x}^{(1)}$ = [1 -2 3 1 2] and $\underline{x}^{(2)}$ = [2 1 -3 -2 1]
- Suppose $\underline{h}$ = [3 2 1]
- At the receiver, we want to get
  - [1 -2 3 1 2] $\circledast$ [3 2 1 0 0] = [8 -2 6 7 11]
  - [2 1 -3 -2 1] $\circledast$ [3 2 1 0 0] = [6 8 -5 -11 -4]
- We transmit $[\underbrace{1\ 2}_{\text{Cyclic prefix}}\ 1\ -2\ 3\ 1\ 2\ \underbrace{-2\ 1}_{\text{Cyclic prefix}}\ 2\ 1\ -3\ -2\ 1]$.
- At the receiver, we get

[1 2 1 -2 3 1 2 -2 1 2 1 -3 -2 1] * [3 2 1]
$= [\underbrace{3\ 8}\ 8\ -2\ 6\ 7\ 11\ \underbrace{-1\ 1}\ 6\ 8\ -5\ -11\ -4\ \underbrace{0\ 1}]$

Junk! To be thrown away by the receiver.

Circular Convolution: Key Properties

- Consider an N-point signal $x[n]$
- **Cyclic Prefix (CP) insertion**: If $x[n]$ is extended by copying the last ν samples of the symbols at the beginning of the symbol:

$$\hat{x}[n] = \begin{cases} x[n], & 0 \le n \le N-1 \\ x[n+N], & -v \le n \le -1 \end{cases}$$

- Key Property 1:

$$\{h \circledast x\}[n] = (h * \hat{x})[n] \text{ for } 0 \le n \le N-1$$

- Key Property 2:

$$\{h \circledast x\}[n] \xrightarrow{\text{FFT}} \boxed{H_k X_k}$$

$\mathbf{h} = \left(h[0], h[1], h[2], \ldots h[\nu]\right)$ $\mathbf{H} = \mathrm{FFT}\left(\tilde{\mathbf{h}}\right)$

zero-padded to length N

OFDM with CP for Channel w/ Memory

- To send N samples $\mathbf{S} = (S_0, S_1, \ldots, S_{N-1})$
- First apply IFFT with scaling by $\sqrt{N}$: $\tilde{\mathbf{s}} = \sqrt{N}\,\mathrm{IFFT}(\mathbf{S})$
- Then, add cyclic prefix

$$\mathbf{x} = \left[\tilde{s}[N-\nu], \ldots, \tilde{s}[N-1], \tilde{s}[0], \ldots, \tilde{s}[N-1]\right]$$

- This is inputted to the channel
- The channel output is $\mathbf{y} = \mathbf{x} * \mathbf{h}$ which can be viewed as

$$\mathbf{y} = \left[p[N-\nu], \ldots, p[N-1], r[0], \ldots, r[N-1]\right]$$

- Remove cyclic prefix to get $\mathbf{r}$. (We know that $\mathbf{r} = \tilde{\mathbf{s}} \circledast \mathbf{h}$.)
- Then apply FFT with scaling by $1/\sqrt{N}$: $\tilde{\mathbf{R}} = \frac{1}{\sqrt{N}}\mathrm{FFT}(\mathbf{r})$
- By circular convolution property of DFT,

$$\mathbf{r} = \tilde{\mathbf{s}} \circledast \mathbf{h} \Longrightarrow R_k = H_k \tilde{S}_k \Longrightarrow \tilde{R}_k = H_k S_k \Longrightarrow S_k = \frac{\tilde{R}_k}{H_k}$$

No ICI!

MATLAB Example (1/2)

```
S = [1 -1 2 4 5 -1 2 -3]; % data stream
h = [1 0.3 0.1];

% OFDM transmitter
N = 4;                          % Number of data symbols per OFDM symbol
n = length(S)/N;                % Number of data blocks
St = (reshape(S,N,n)).';        % Reshape stream to matrix for
                                % easier addition of cyclic prefix
st = (sqrt(N))*ifft(St,[],2);   % Calculate the IFFT with scaling
```

```
St =
     1    -1     2     4
     5    -1     2    -3
```

IFFT w/ scaling (row-wise)

```
st =
   3.0 + 0.0i  -0.5 - 2.5i   0.0 + 0.0i  -0.5 + 2.5i
   1.5 + 0.0i   1.5 + 1.0i   5.5 + 0.0i   1.5 - 1.0i
```

```
v = length(h)-1;
xt = [st(:,(N-(v-1)):N) st];        % Add Cyclic Prefix
x = (reshape(xt.',((N+v)*n),1)).'; % Reshape back to stream
```

```
x =
   0.0 + 0.0i  -0.5 + 2.5i   3.0 + 0.0i  -0.5 - 2.5i   0.0 + 0.0i   -0.5 + 2.5i   5.5 + 0.0i   1.5 - 1.0i   1.5 + 0.0i
1.5 + 1.0i   5.5 + 0.0i   1.5 - 1.0i
```

MATLAB Example (2/2)

```
%---------------------------
% Convolve with channel
y = conv(x,h);
H = fft([h zeros(1,N-v-1)]);
%---------------------------
% OFDM receiver
y = y(1:((N+v)*n));
yt = reshape(y,(N+v),n).';          % Reshape matrix for easier
                                    % removal of cyclic prefix
r = yt(:,v+1:v+N);                  % Eliminate junk (cyclic prefix)
Rt = (1/sqrt(N))*fft(r,[],2);       % Calculate the FFT with scaling
```

```
y =
   0.00 + 0.00i  -0.50 + 2.50i   2.85 + 0.75i   0.35 - 2.25i   0.15 - 0.75i
-0.55 + 2.25i   5.35 + 0.75i   3.10 - 0.75i   2.50 - 0.30i   2.10 + 0.90i
6.10 + 0.30i   3.30 - 0.90i
```

```
r =
   2.85 + 0.75i   0.35 - 2.25i   0.15 - 0.75i  -0.55 + 2.25i
   2.50 - 0.30i   2.10 + 0.90i   6.10 + 0.30i   3.30 - 0.90i
```

FFT w/ scaling (row-wise)

```
Rt =
   1.4 + 0.0i  -0.9 + 0.3i   1.6 + 0.0i   3.6 + 1.2i
   7.0 + 0.0i  -0.9 + 0.3i   1.6 + 0.0i  -2.7 - 0.9i
```

```
% "Equalization"
S_hatt = zeros(size(Rt));
for i=1:length(H)
    S_hatt(:,i) = Rt(:,i)/H(i);
end
S_hat = reshape(S_hatt.',1,N*n)
```

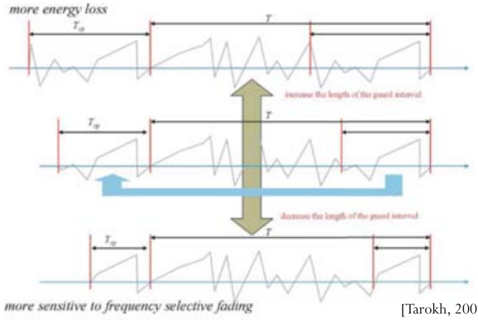
% Divide Rk (the kth column) by Hk

```
Shatt =
     1    -1     2     4
     5    -1     2    -3
```

```
S_hat = [1 -1 2 4 5 -1 2 -3]
```

OFDM System Design: CP

- A good ratio between the CP interval and symbol duration should be found, so that all multipaths are resolved and not significant amount of energy is lost due to CP.
- As a thumb rule, the CP interval must be two to four times larger than the root mean square (RMS) delay spread.

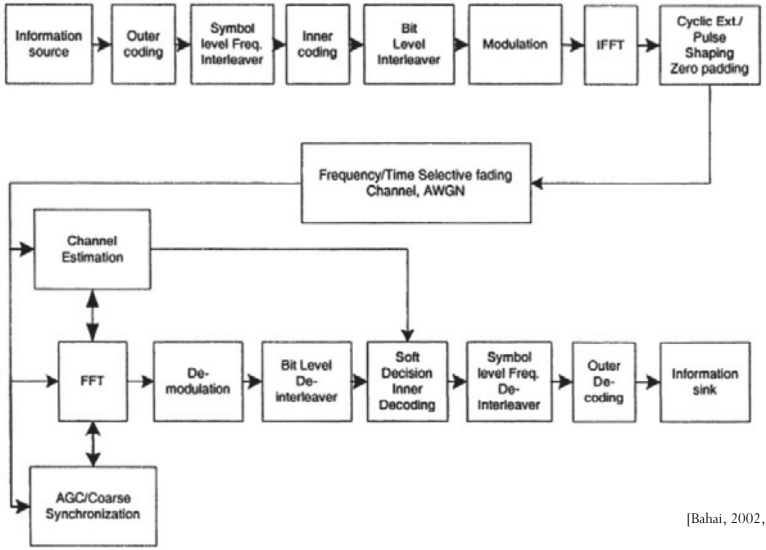


[Tarokh, 2009, Fig 2.9]

Summary

- The CP at the beginning of each block has two main functions.
- As guard interval, it prevents contamination of a block by ISI from the previous block.
- It makes the received block appear to be periodic of period N.
  - Turn regular convolution into circular convolution
  - Point-wise multiplication in the frequency domain

OFDM Architecture



[Bahai, 2002, Fig 1.11]

Reference

- A. Bahai, B. R. Saltzberg, and M. Ergen, ***Multi-Carrier Digital Communications: Theory and Applications of OFDM***, 2nd ed., New York: Springer Verlag, 2004.