Lecture 8: Orthogonal Frequency Division Multiplexing (OFDM)

Scribe: 謝秉昂、陳心如 Lecture Date: 4/26, 2017 Lecturer: I-Hsiang Wang

1 Outline

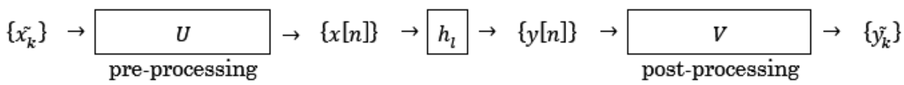
1. Eigenfunction of LTI system
2. Circular Convolution and Cyclic Prefix
3. Matrix View of OFDM

2 Recap

1. Last time we introduce wireline channel and its equivalent discrete-time baseband model.
   input: $x[n]$
   output: $y[n] = \sum_{j=0}^{L-1} h_l x[n-l] + z[n] = h_0 x[n] + \sum_{j=1}^{L-1} h_l x[n-l] + z[n]$ (desired signal + past interference + noise)
   $L \approx \frac{T_d}{T}$: number of taps in the LTI channel
   "Inter-Symbol Interference (ISI)" is a new challenge.
2. We can use Viterbi Algorithm to optimally solve the MLSD pattern. However, the complexity is proportional to 2^L, L is typically 100 to 400 for wideband system. Thus, even though Viterbi Algorithm is an optimal approach, L is still too large in most cases and hence we use OFDM.

3 Eigenfunction of LTI: concept of OFDM

1. Basic Idea of OFDM



$\tilde{y}_k = \tilde{h}_k \tilde{x}_k$ is ISI-free, i.e. still ISI-free if noise-free.

<u>Observation</u>

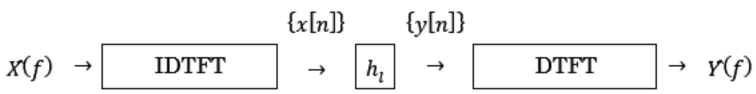
If you are willing to go back to the "analog" world, then there is a simple solution!

<u>Fact</u>

$$u[n] \overset{DTFT}{\longleftrightarrow} \hat{u}(f) = \sum_{n=-\infty}^{\infty} x[n] e^{-j2\pi fn}$$

$$v[n] \overset{DTFT}{\longleftrightarrow} \hat{v}(f)$$

$$(u * v)[n] \overset{DTFT}{\longleftrightarrow} \hat{u}(f) \cdot \hat{v}(f)$$



$$\hat{Y}(f) = \hat{H}(f) \hat{X}(f)$$

But we don't want to go back to analog. Is there a digital solution?

2. Discrete Fourier Transform (DFT) ($\neq$ Discrete-Time Fourier Transform, DTFT)

$$x[n] \overset{DFT}{\longleftrightarrow} \hat{x}_k \triangleq \sum_{n=0}^{N-1} x[n]\, e^{-j2\pi \frac{k}{N} n} = \hat{x}_{DTFT}\left(\frac{k}{N}\right)$$

$$\hat{x}_k \overset{IDFT}{\longleftrightarrow} x[n] \triangleq \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} \hat{x}_k e^{j2\pi \frac{k}{N} n}$$

Can we achieve the same with DFT? No, it cannot be ISI-free.

Remark

For linear convolution, DFT and IDFT cannot translate into multiplication in the other domain, i.e. $DFT(h * x) \neq DFT(h) \cdot DFT(x)$

3. Linear Convolution v.s. Circular Convolution

Recap

Proof of the convolution-multiplication property:

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

$$\hat{h}(f) \triangleq \sum_{n=-\infty}^{\infty} h[n]\, e^{-j2\pi f n}$$

$$y[n] = (h * x)[n] = \sum_{k=-\infty}^{\infty} h[k]\, x[n-k]$$

$$\begin{aligned}
\hat{y}(f) &= \sum_{n=-\infty}^{\infty} \sum_{k=-\infty}^{\infty} h[k]\, x[n-k]\, e^{-j2\pi f(n-k)} e^{-j2\pi f k} \\
&= \sum_{k=-\infty}^{\infty} h[k] \left(\sum_{n-k=-\infty}^{\infty} x[n-k]\, e^{-j2\pi f(n-k)} \right) e^{-j2\pi f k} \\
&= \hat{h}(f) \cdot \hat{x}(f)
\end{aligned}$$

Since the range is not from infinity to infinity for DFT, the function cannot be shifted as in DTFT.

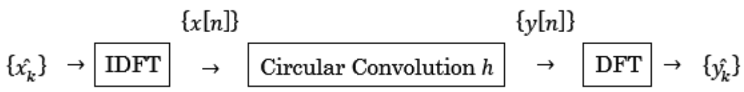
Def. (Circular Convolution)

$$(h \circledast_N x)[n] = \sum_{k=0}^{N-1} x\left[(n-k)_{mod\ N}\right] h[k]$$

Fact

$$DFT_N\{h \circledast_N x\} = \sqrt{N} \cdot DFT\{h\} \cdot DFT\{x\}$$

If the channel is doing circular convolution instead of linear convolution, then the following architecture is ISI-Free!



We need to pay a price for converting linear convolution into circular convolution. But as long as N is large enough, the price can be overlooked.

Redo convolution-multiplication property:

$$\hat{x}_k \triangleq \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} x[n]\, e^{-j2\pi \frac{k}{N} n}$$

$$\hat{h}_k \triangleq \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} h[n]\, e^{-j2\pi \frac{k}{N} n}$$

$$y[n] = (h \circledast_N x)[n] = \sum_{k=0}^{N-1} h[k]\, x\left[(n-k)_{mod\ N}\right]$$

$$\begin{aligned}
\hat{y}_l &= \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} \sum_{k=0}^{N-1} h[k]\, x\left[(n-k)_{mod\ N}\right] e^{-j2\pi \frac{l}{N}\left[(n-k)_{mod\ N}\right]} e^{-j2\pi \frac{l}{N} k} \\
&= \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} h[k] \left(\sum_{t=(n-k)_{mod\ N}=0}^{N-1} x\left[(n-k)_{mod\ N}\right] e^{-j2\pi \frac{l}{N}\left[(n-k)_{mod\ N}\right]} \right) e^{-j2\pi \frac{l}{N} k} \\
&= \frac{1}{\sqrt{N}} \left(\sqrt{N}\hat{h}_k\right)\left(\sqrt{N}\hat{x}_k\right)
\end{aligned}$$

4 Implement Circular Convolution on the LTI Channel

Original Channel (ignore noise)
Linear Convolution

$$y[n] = \sum_{l=0}^{L-1} h_l x[n-l],\ n = 0, 1, \cdots, N-1$$

Desired Channel
Circular Convolution

$$y[n] = \sum_{l=0}^{L-1} h_l x\left[(n-l)_{mod\ N}\right]$$

1. Add "cyclic prefix (CP)" to implement circular convolution on a linear convolution system!

   Original Sequence

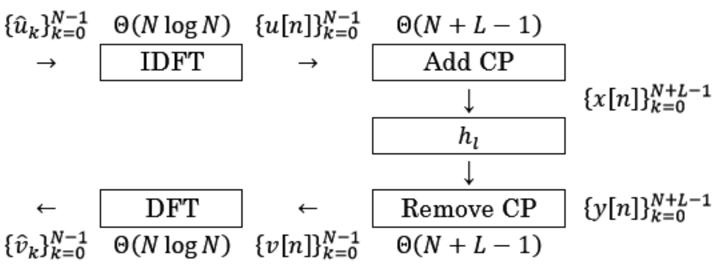
$x[0]$	$x[1]$	$\cdots$	$x[n-1]$

   New Sequence: $numbers\ of\ symbols = N + L - 1$

$x[N-L+1]$	$\cdots$	$x[N-2]$	$x[N-1]$	$x[0]$	$x[1]$	$\cdots$	$x[n-1]$

   Cyclic Prefix, $L-1$ $\qquad N$

2. OFDM Architecture



$$Overhead = \frac{L-1}{N+L-1} \ll 1\ if\ L \ll N$$

5 A Linear Algebra Perspective

$$\underline{y} = \begin{bmatrix} y[0] \\ y[1] \\ \vdots \\ y[N-1] \end{bmatrix}, \underline{x} = \begin{bmatrix} x[0] \\ x[1] \\ \vdots \\ x[N-1] \end{bmatrix}$$

Linear Convolution

$$\underline{y} = H\underline{x}, H = \begin{bmatrix} h_0 & 0 & \cdots & \cdots & \cdots & 0 \\ h_1 & h_0 & 0 & \cdots & \cdots & 0 \\ h_2 & h_1 & h_0 & \cdots & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \cdots & \cdots & \cdots & h_2 & h_1 & h_0 \end{bmatrix}$$

Circular Convolution

$$\underline{y} = H_C\underline{x}, H = \begin{bmatrix} h_0 & 0 & \cdots & 0 & h_2 & h_1 \\ h_1 & h_0 & 0 & \cdots & 0 & h_2 \\ h_2 & h_1 & h_0 & 0 & \cdots & 0 \\ 0 & h_2 & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & 0 & h_2 & h_1 & h_0 \end{bmatrix}$$

H_C is a "circulant matrix", i.e. row $(i+1)$ is the right-shift of row i

$$\underline{y} = H\underline{x} + \underline{z}$$

For post-processing,

$$\underline{\widehat{y}} = V\underline{y} = VH\underline{x} + V\underline{z}$$

For pre-processing,

$$\underline{x} = U\underline{\widehat{x}}$$

$$\underline{\widehat{y}} = (VHU)\,\underline{\widehat{x}} + \underline{\widehat{z}}$$

which is ISI-free iff. (VHU) is a diagonal matrix and that $\{V, U\} \in orthonormal$.

We wish:

$$D = diag\,(\lambda_1, \lambda_2, \cdots, \lambda_N)$$

$$D = V \cdot H \cdot U$$

$$H = V^{-1} D U^{-1}$$

Det: Circulant Matrix

C is an $n \times n$ matrix, and it is circulant if consecutive rows are circular shifts of the same vector.

Prop: Eigenvalue & Eigenvectors of Circulant Matrix

Let $\phi_k\,[n] \triangleq e^{j2\pi\frac{k}{N}n}$

$$\underline{\phi}_k = \begin{bmatrix} \phi_k\,[0] \\ \phi_k\,[1] \\ \vdots \\ \phi_k\,[N-1] \end{bmatrix}$$

For any $N \times N$ circulant matrix, $\underline{\phi}_k$ is the eigenvector.

$\mathbb{C}\underline{\phi}_k = \lambda_k \underline{\phi}_k \; for \; k = 0, 1, \cdots, N-1$

Hence

$$C = \Phi \mathbb{D} \Phi^{-1}$$

where $\Phi = [\underline{\phi}_0, \underline{\phi}_1, \underline{\phi}_{N-1}]$

$$D = \begin{bmatrix} \lambda_0 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \lambda_{N-1} \end{bmatrix}$$

$$C = \begin{bmatrix} c_0 & c_1 & \cdots & c_{N-1} \\ c_{N-1} & c_0 & \cdots & c_{N-2} \\ \vdots & \vdots & \vdots & \vdots \\ c_1 & c_2 & \cdots & c_0 \end{bmatrix}$$

determined by $\underline{c} = [c_0, c_1, \cdots, c_{N-1}]$

$$C\underline{\phi}_k = \begin{bmatrix} \underline{c}_0^T \underline{\phi}_k \\ \underline{c}_1^T \underline{\phi}_k \\ \vdots \\ \underline{c}_{N-1}^T \underline{\phi}_k \end{bmatrix}$$

l^{th} component: $(\phi_k[l])^{-1} \underline{c}_l^T \underline{\phi}_k = \sum_{n=0}^{N-1} \frac{\phi_k[n] \cdot C_{(n-l)_{mod\ N}}}{\phi_k[l]} = \sum_{n=0}^{N-1} C_{(n-l)_{mod\ N}} e^{j2\pi \frac{k}{N}(n-l)_{mod\ N}} = \sqrt{N} \cdot DFT\{c_n\}$
This is true for all l.
$C\underline{\phi}_k = \lambda_k \cdot \underline{\phi}_k$ where $\lambda_k = \sqrt{N} DFT\{c_n\}$

$$\lambda_k = DFT\{\underline{c}\}$$

Back to OFDM,

$$\underline{y} = H_C \cdot \underline{x} + \underline{z}$$

$$H_C = \Phi D \Phi^H$$

$$\underline{\widehat{y}} = \Phi^H \underline{y} = \Phi^H H_C \underline{x} + \Phi^H \underline{z}$$

Let $\underline{x} = \Phi \cdot \underline{\widehat{x}}$

$$\underline{\widehat{y}} = \Phi^{-H} \left(\Phi D \Phi^H \right) \Phi \underline{\widehat{x}} + \underline{\widehat{z}} = D \underline{\widehat{x}} + \underline{\widehat{z}}$$

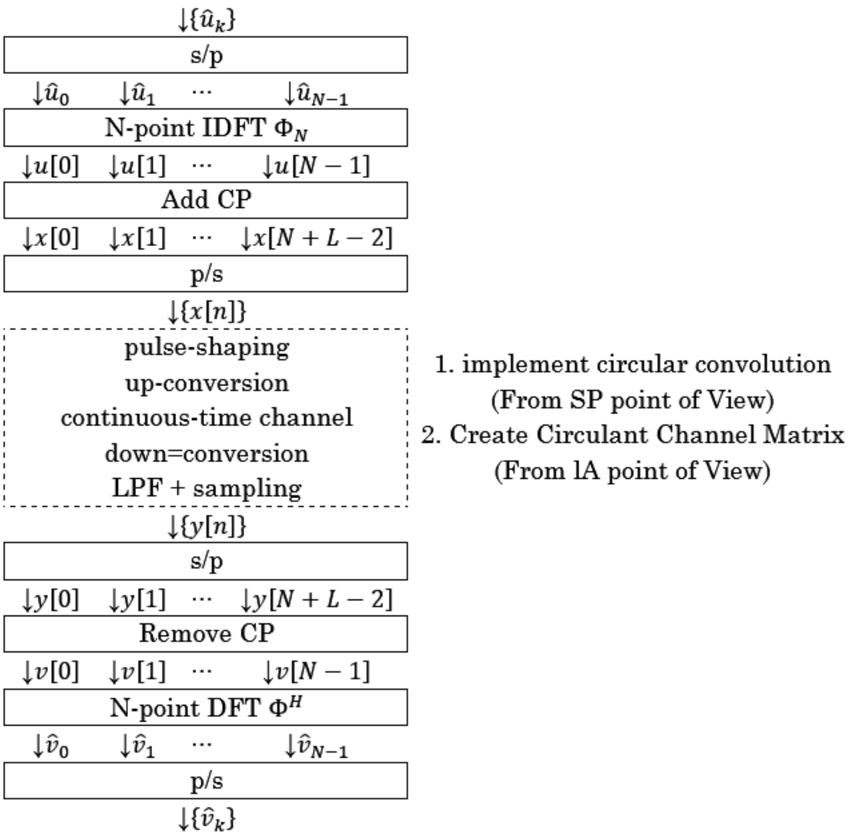
where $D = \begin{bmatrix} \sqrt{N}\hat{h}_0 & & & \\ & \sqrt{N}\hat{h}_1 & & \\ & & \ddots & \\ & & & \sqrt{N}\hat{h}_{N-1} \end{bmatrix}$

The "IDFT Matrix" is

$$\Phi = \frac{1}{\sqrt{N}} (\Phi_{kn})_{k,n \in \{0,1,\cdots,N-1\}}$$

where $\Phi_{kn} = \phi_k[n] = e^{j2\pi \frac{kn}{N}}$

6 Summary



Equivalent System: N Parallel AWGN Channels

$$\hat{v}_k = \left(\sqrt{N}\hat{h}_k\right) \cdot \hat{u}_k + \hat{z}_k, k = 0, 1, \cdots, N-1$$

Remark
How to choose N?

1. If we want to use FFT to implement DFT/IDFT, we will pick $N = 2^p$ for some p
2. N also depends on total bandwidth and the allowed frequency spacing Δf

Example:
DSL modem

$$W = 6 \ MHz$$

$$N = 256 \sim 1024$$

$$\Delta f \approx 12 \ kHz$$

$$\frac{L-1}{N} \approx 10\%$$