# ECE 463: Digital Communications Lab.

## Lecture 6: Channel Estimation & Correction

Haitham Hassanieh

ILLINOIS
UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN

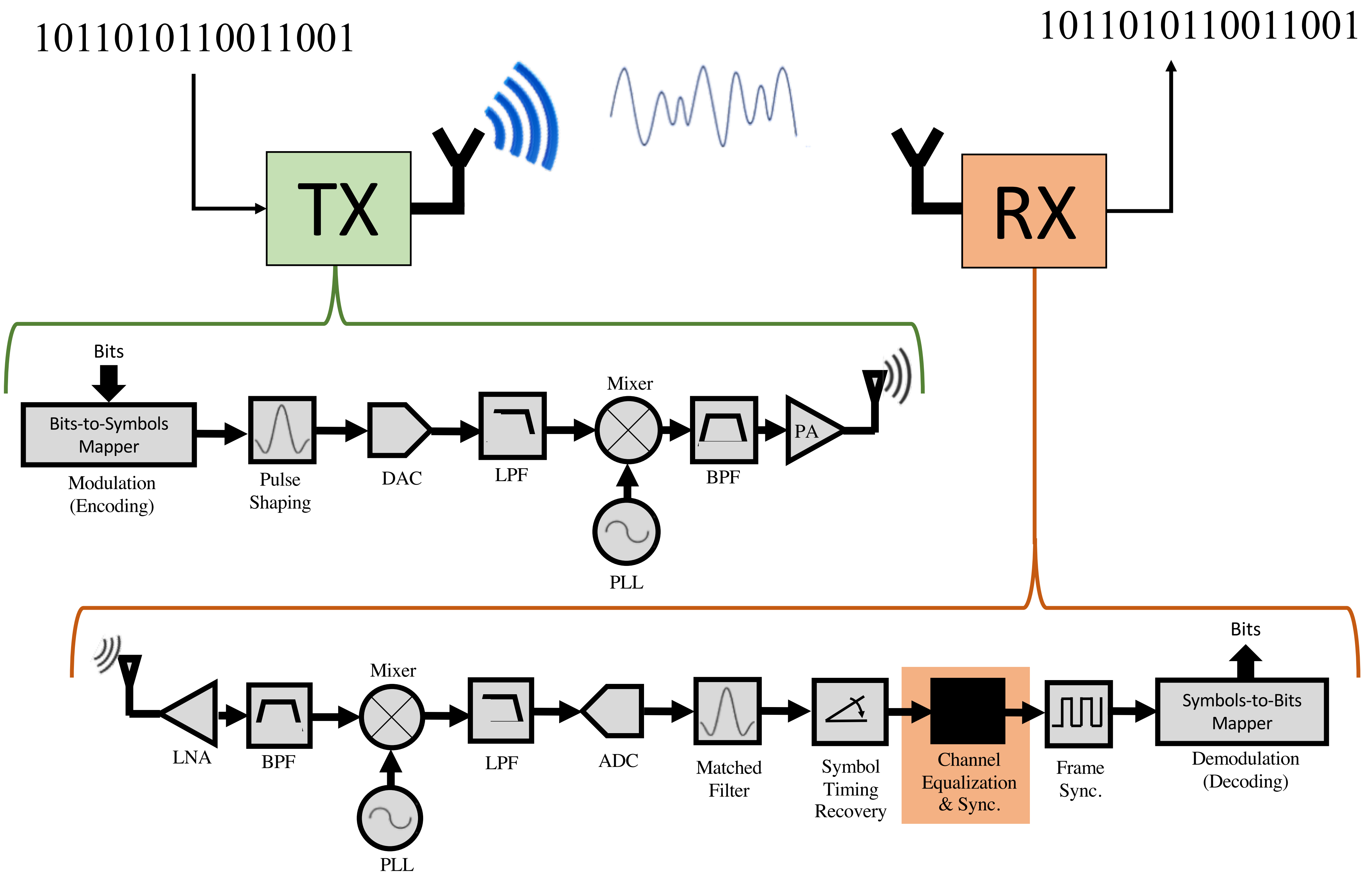
# Previous Lecture:

- ✓ ASK Modulation
- ✓ FSK Modulation (Coherent & Non-Coherent)
- ✓ Frame Synchronization

# This Lecture:

- ❑ Multipath Channel
- ❑ Channel Estimation & Correction
- ❑ Narrowband vs. Wideband Channels
- ❑ Channel Equalization

# Digital Communication System



# The Channel


$x(t)$ → Channel → $y(t) = \mathbf{h}\, x(t-\tau) + v(t)$

Channel:

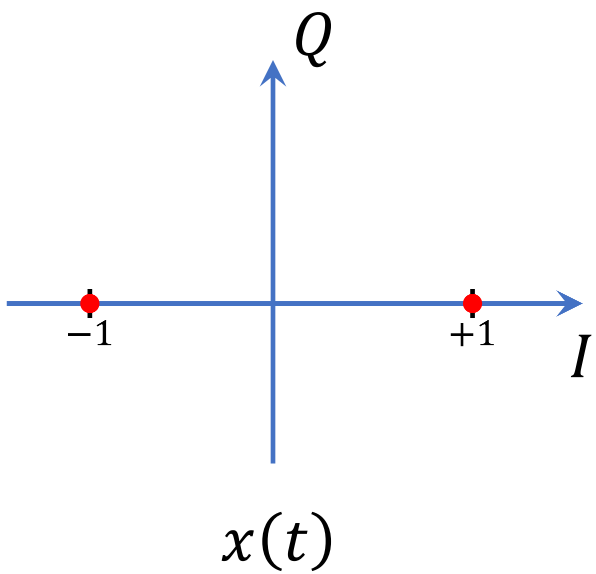
- Adds Noise
- Attenuates the Signal
- Rotates the Phase of the Signal
- Delays the Signal

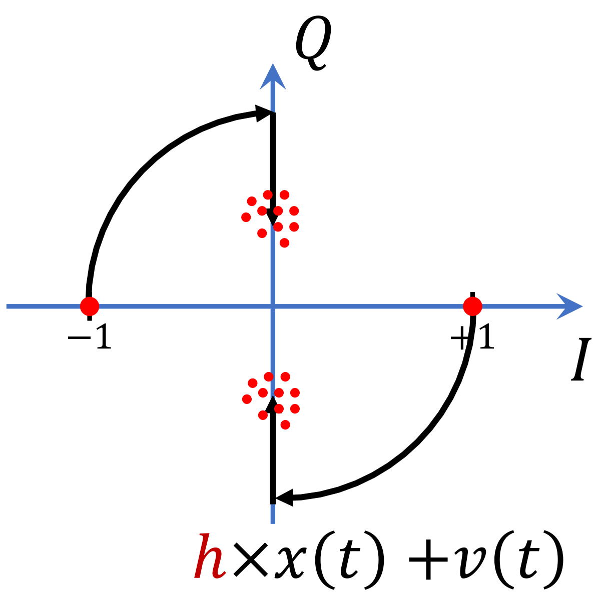
$$h \propto \frac{\lambda}{d'} e^{j2\pi d'/\lambda}$$

# The Channel

Consider BPSK Modulation.

$$0 \rightarrow -1$$
$$1 \rightarrow +1$$

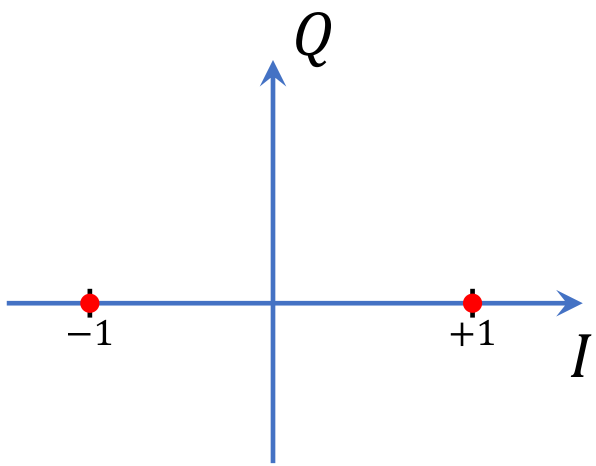


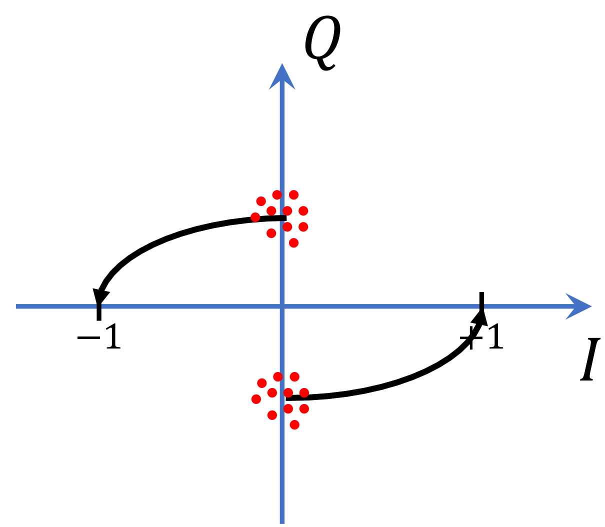
# Channel Estimation & Correction

Consider BPSK Modulation.

$$0 \rightarrow -1$$
$$1 \rightarrow +1$$




$x(t)$




$h \times x(t) + v(t)$

## Send Training Sequence (Preamble Bits): Known Bits

$x(0) = 1 \longrightarrow y(0) = h + v(0)$

$x(1) = 1 \longrightarrow y(1) = h + n(1)$

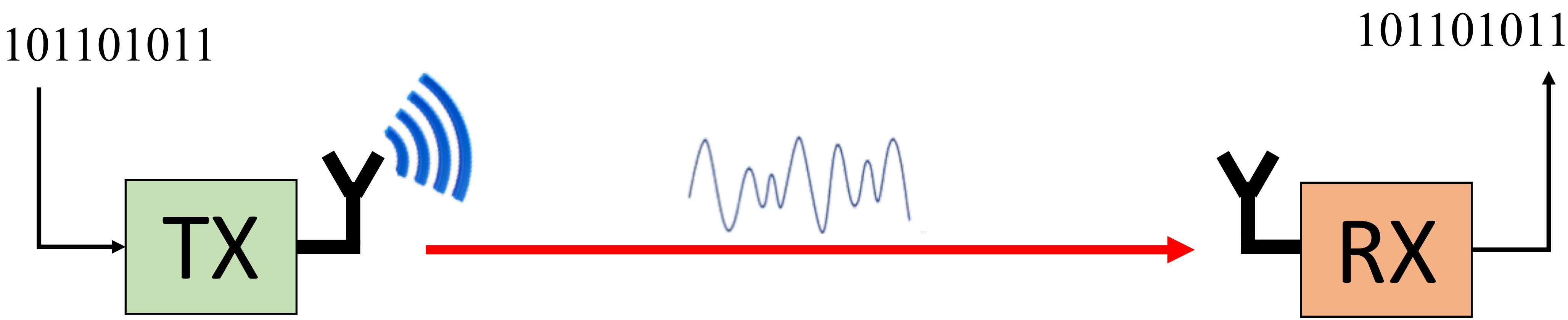
$x(2) = -1 \longrightarrow y(2) = -h + n(2)$

⋮

Estimate channel: $\tilde{h} = \sum_k \frac{y(k)}{x(k)}$

Correct channel: $\tilde{x}(t) = \frac{y(t)}{\tilde{h}}$

# The Channel

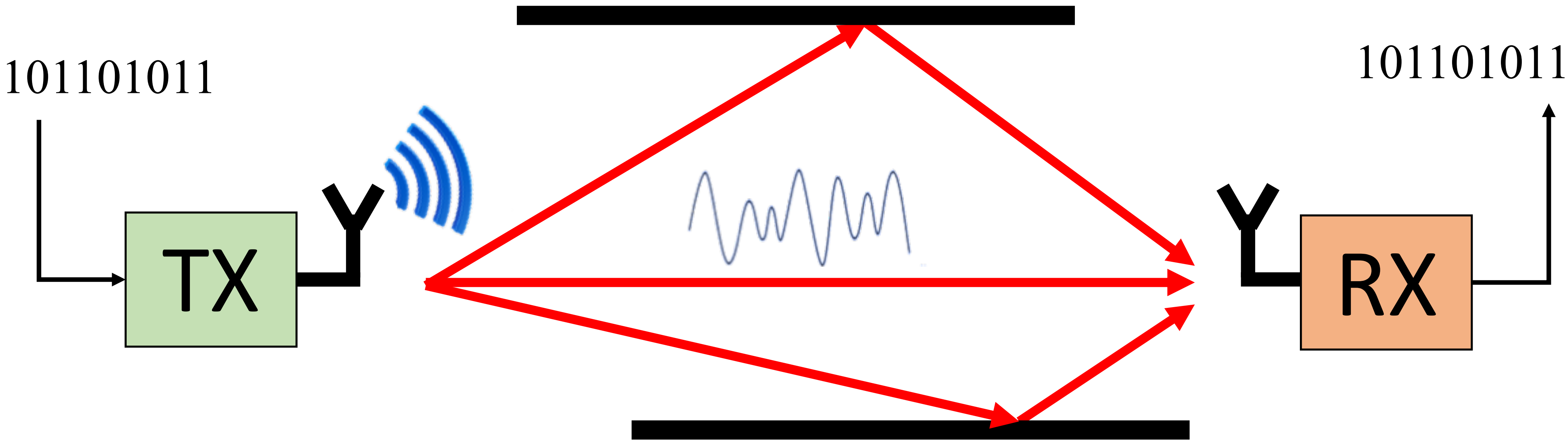



$x(t)$ ➡ **Channel** ➡ $y(t) = \mathbf{h}\, x(t - \tau) + v(t)$

$$h \propto \frac{\lambda}{d} e^{j2\pi d/\lambda}$$

**Assumes single path!**

# Multipath Channel




*Multipath Propagation:* radio signal reflects off objects ground, arriving at destination at slightly different times

$$y(t) = \alpha_1 e^{\phi_1}\, x(t-\tau_1) \; + \alpha_2 e^{\phi_2}\, x(t-\tau_2) \; + \alpha_3 e^{\phi_3}\, x(t-\tau_3) \cdots$$

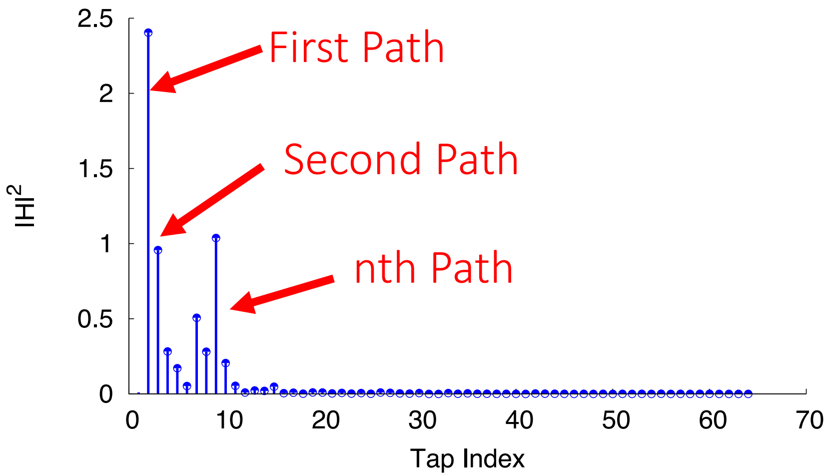
$$y(t) = \sum_k \alpha_k e^{\phi_k}\, x(t-\tau_k) \; = \sum_k h(\tau_k)\, x(t-\tau_k) \; = h(t) * x(t)$$

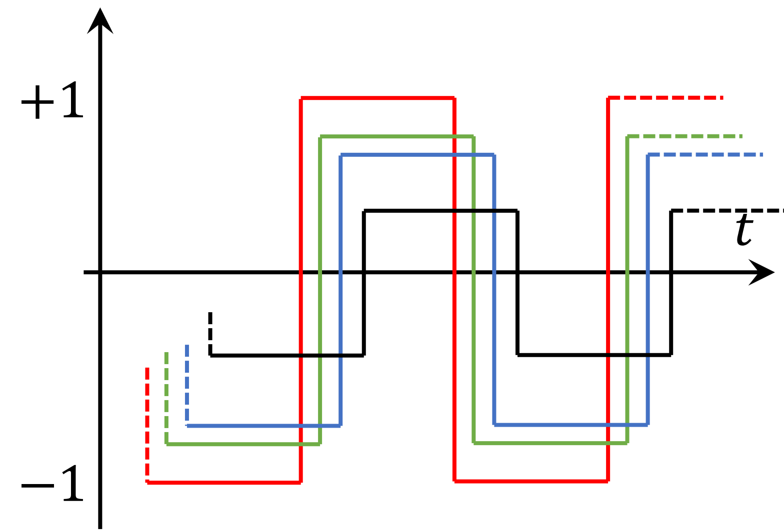
$h(t)$ is channel impulse response.

# Multipath Channel

$h(t)$ is channel impulse response.

$$y(t) = \sum_k h(\tau_k)\, x(t - \tau_k) \;= h(t) * x(t)$$



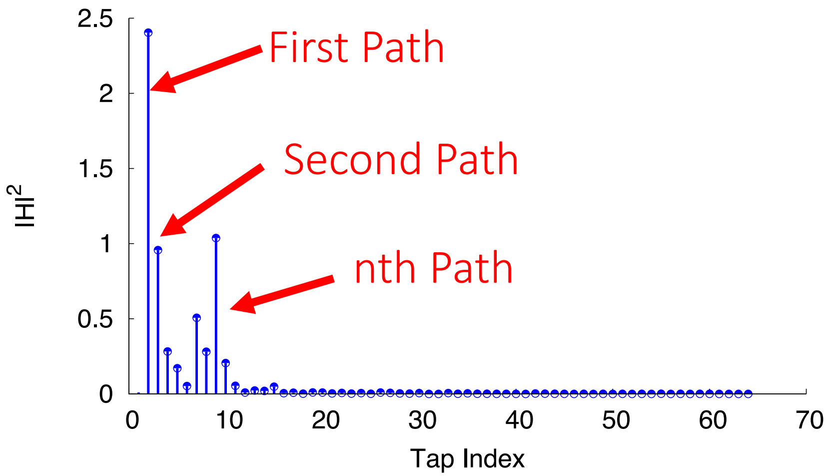

Multi-tap Channel ➡ ISI: Inter-Symbol-Interference

Symbols arriving along late paths interfere with following symbols.

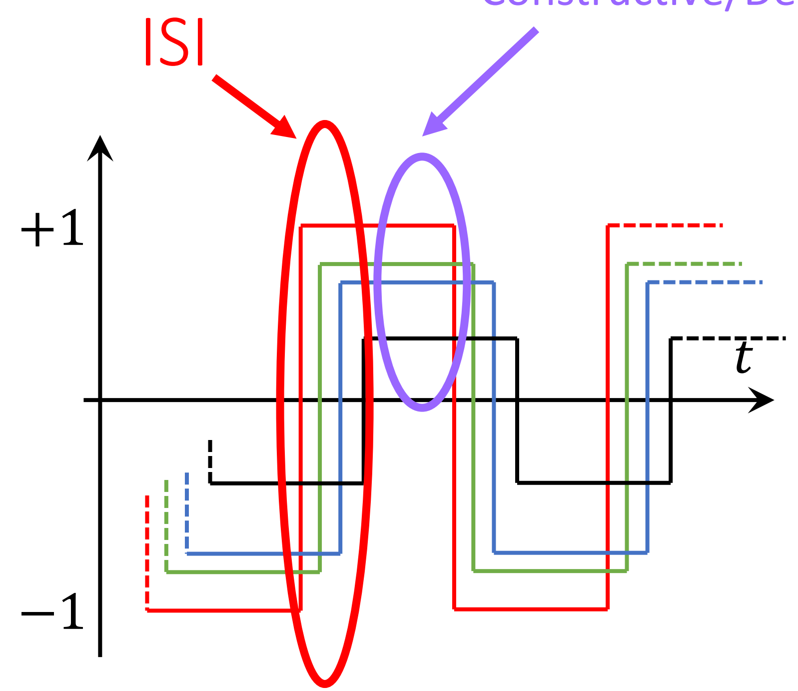
# Multipath Channel

$h(t)$ is channel impulse response.

$$y(t) = \sum_k h(\tau_k)\, x(t - \tau_k) \; = h(t) * x(t)$$




Paths sum with different phases: Constructive/Destructive




Multi-tap Channel ➡

ISI: Inter-Symbol-Interference

Symbols arriving along late paths interfere with following symbols.

➡ Channel Fading

Symbols arriving along different paths sum up destructively

# Multipath Channel

$h(t)$ is channel impulse response.

$$y(t) = \sum_k h(\tau_k)\, x(t - \tau_k) \ = h(t) * x(t)$$

Channel Fading: Symbols arriving along different paths sum up destructively

*Example 2 paths with distance* $d_1 = 1m, d_2 = 1.06m$:

$$h = \ h_1 + h_2 \ = \frac{\lambda}{d_1}\, e^{j2\pi d_1/\lambda} + \frac{\lambda}{d_2}\, e^{j2\pi d_2/\lambda}$$

$$= \frac{\lambda}{d_1}\, e^{j2\pi d_1/\lambda} \left(1 + \frac{d_1}{d_2}\, e^{j2\pi(d_2 - d_1)/\lambda}\right) \qquad \frac{d_1}{d_2} \approx 1$$

$$if\ \frac{d_2 - d_1}{\lambda} \approx \frac{1}{2} \rightarrow \ h = \frac{\lambda}{d_1}\, e^{j2\pi d_1/\lambda}\left(1 + e^{j\pi}\right) \ = 0$$

Destructive Interference!

# Multipath Channel

$h(t)$ is channel impulse response.

$$y(t) = \sum_k h(\tau_k)\, x(t - \tau_k) \;= h(t) * x(t)$$

Channel Fading: Symbols arriving along different paths sum up destructively

*Example 2 paths with distance $d_1 = 1m, d_2 = 1.06m$:*

$$h = \; h_1 + h_2 \; = \frac{\lambda}{d_1}\, e^{j2\pi d_1/\lambda} \; + \frac{\lambda}{d_2}\, e^{j2\pi d_2/\lambda}$$

$@f_1 = 2.5GHz\ (\lambda = 12\ cm)$: $h = 0.12\, e^{j\frac{2\pi}{3}} + 0.113\, e^{j\frac{5\pi}{3}} \approx 0.006$

$@f_2 = 5GHz\ (\lambda = 6\ cm)$: $h = 0.06\, e^{j\frac{5\pi}{3}} + 0.05\, e^{j\frac{5\pi}{3}} \approx 0.116$

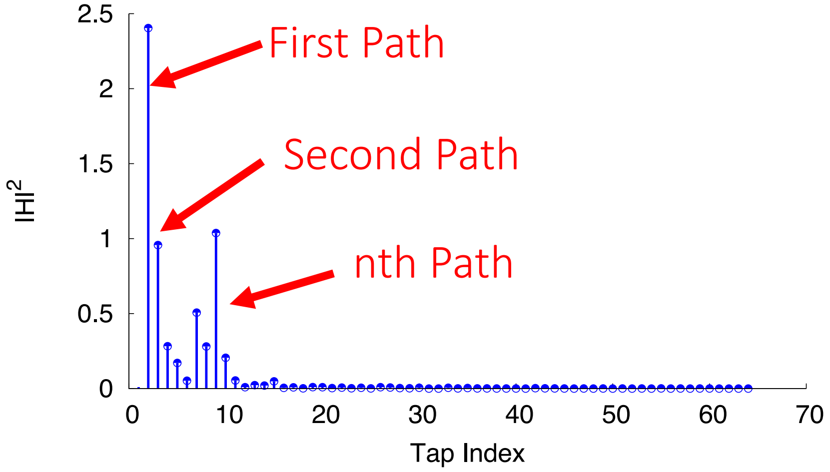
$17\times$
(24dB)

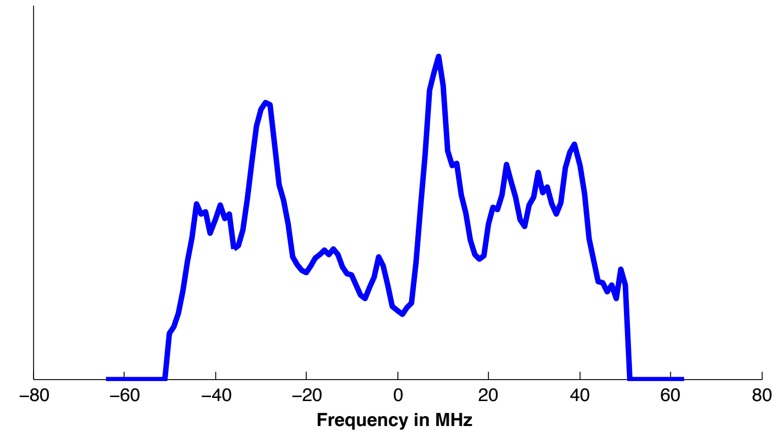
Frequency Selective Fading

# Multipath Channel

$h(t)$ is channel impulse response.

$$y(t) = \sum_k h(\tau_k)\, x(t - \tau_k) \; = h(t) * x(t) \qquad \Leftrightarrow \qquad H(f)X(f)$$



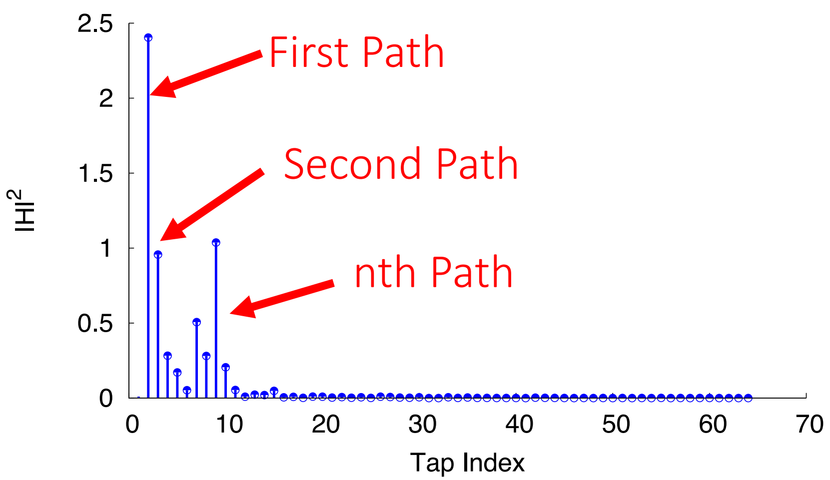



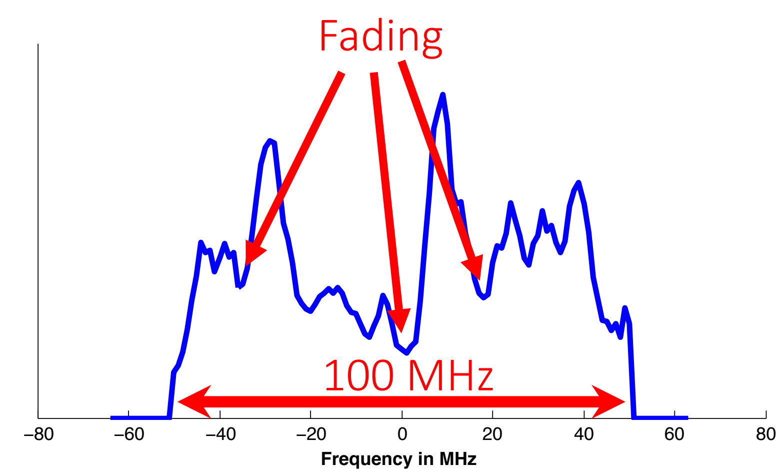

Multi-tap Channel

# Multipath Channel

$h(t)$ is channel impulse response.

$$y(t) = \sum_k h(\tau_k)\, x(t - \tau_k) \;= h(t) * x(t) \quad \Leftrightarrow \quad H(f)X(f)$$




Multi-tap Channel ➡ **Frequency Selective Fading**

Symbols arriving along different paths sum up destructively

⬇

**ISI: Inter-Symbol-Interference**

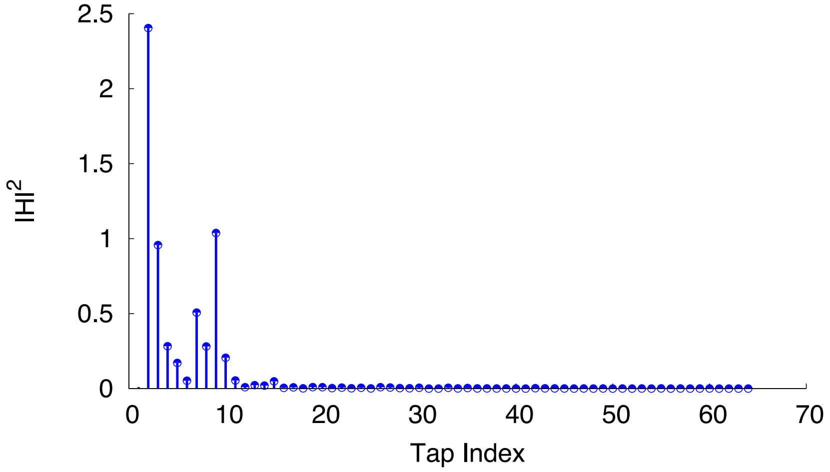
Symbols arriving along late paths interfere with following symbols.

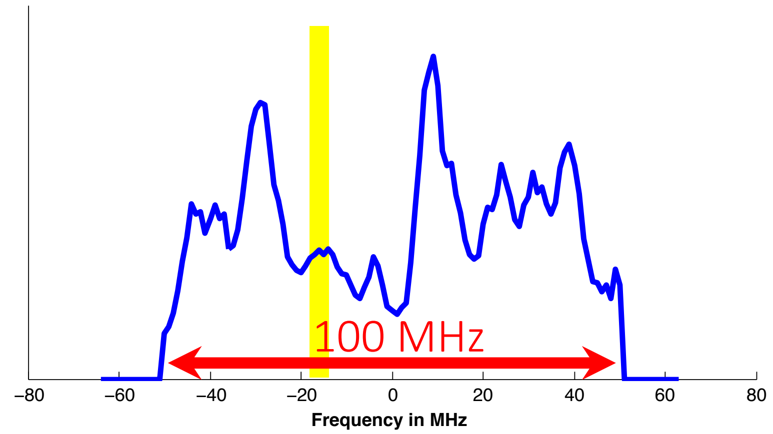
**Problematic in Wideband Channel!**

# Narrowband Channel

$h(t)$ is channel impulse response.

$$y(t) = \sum_k h(\tau_k)\, x(t - \tau_k) \; = h(t) * x(t) \quad \Leftrightarrow \quad H(f)X(f)$$

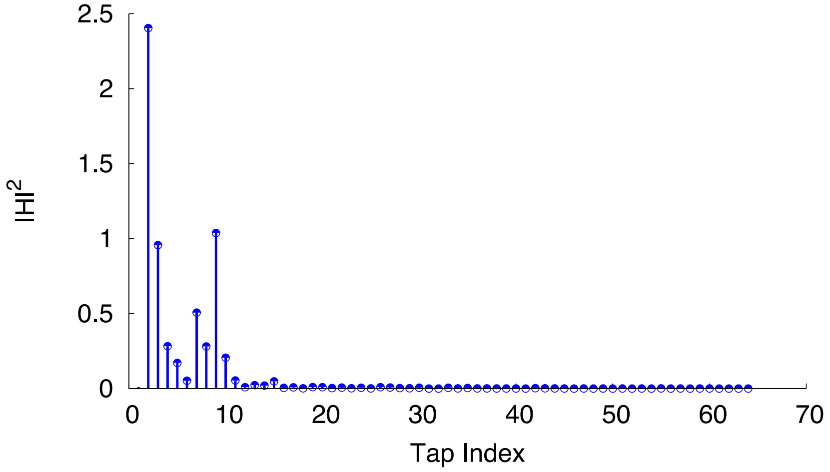


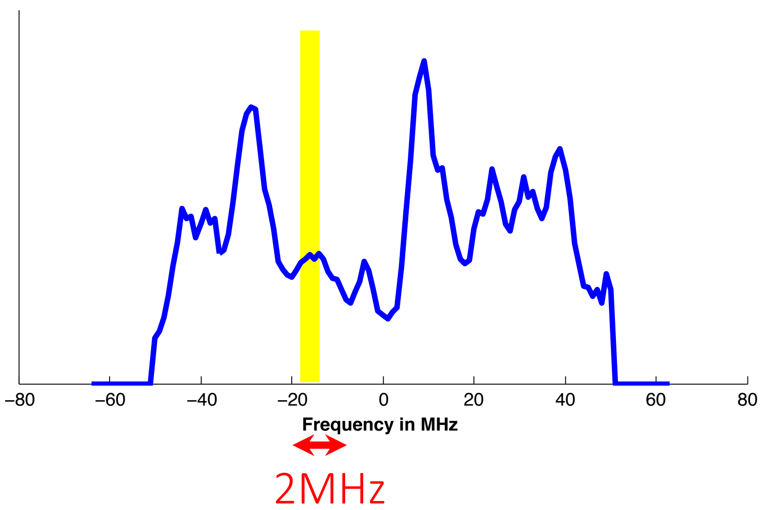



# Narrowband Channel

$h(t)$ is channel impulse response.

$$y(t) = \sum_k h(\tau_k)\, x(t - \tau_k) \ = h(t) * x(t) \quad \Leftrightarrow \quad H(f)X(f)$$

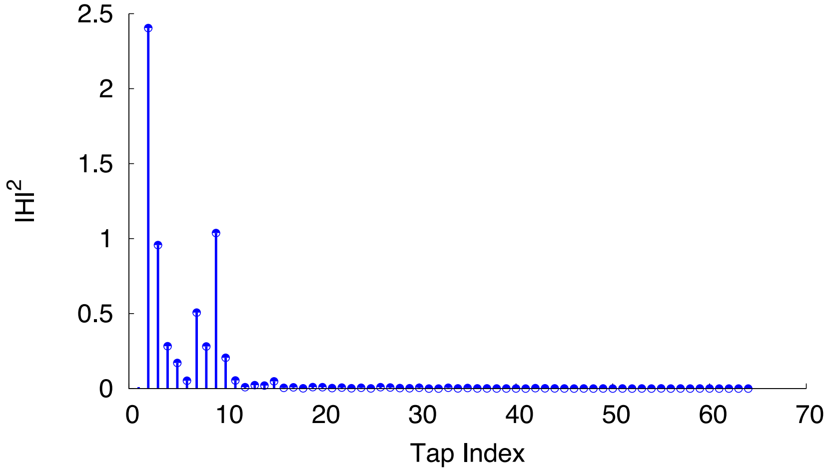


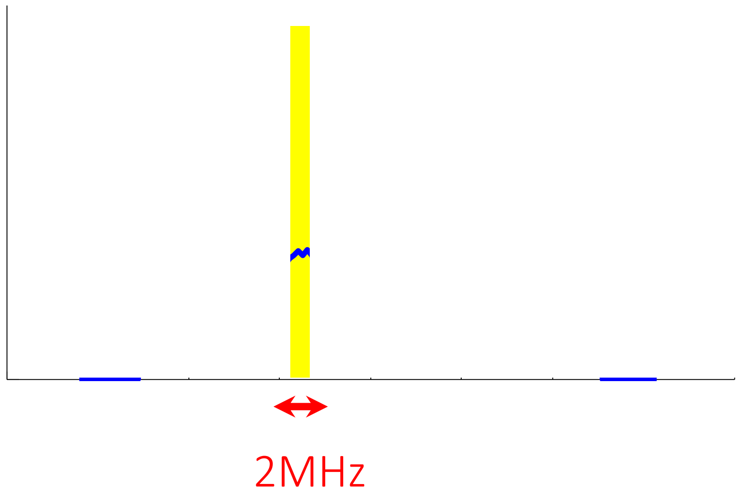



# Narrowband Channel

$h(t)$ is channel impulse response.

$$y(t) = \sum_k h(\tau_k)\, x(t - \tau_k) \; = h(t) * x(t) \qquad \Leftrightarrow \qquad H(f)X(f)$$

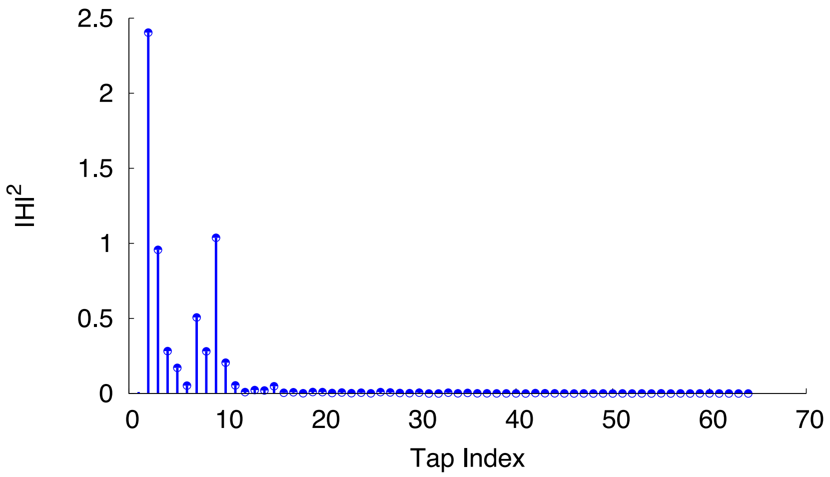


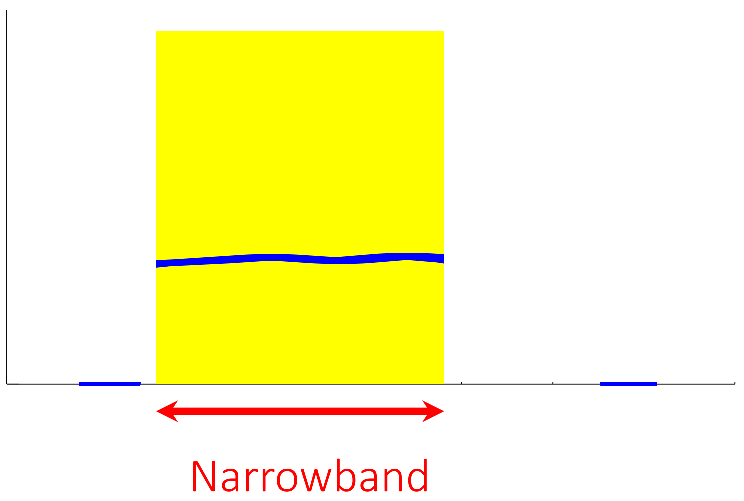



# Narrowband Channel

$h(t)$ is channel impulse response.

$$y(t) = \sum_k h(\tau_k)\, x(t - \tau_k) \; = h(t) * x(t) \quad \Longleftrightarrow \quad H(f)X(f)$$



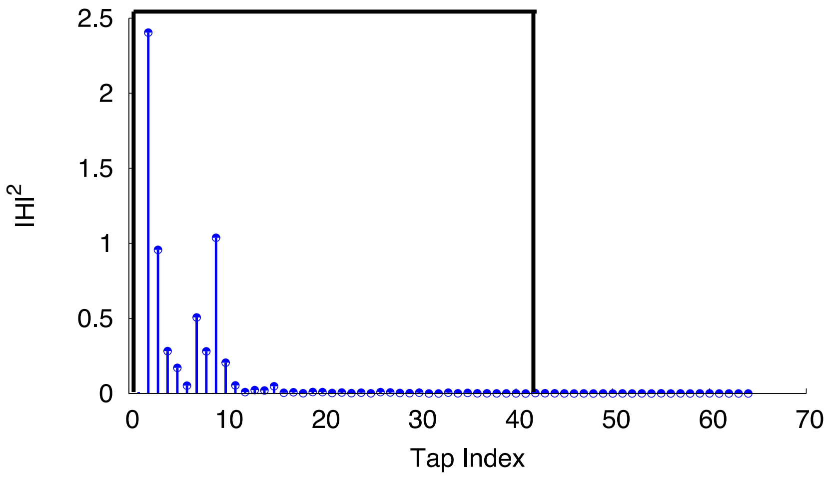

Symbol time: $T \propto \frac{1}{\text{Bandwidth}} \gg \tau_k$

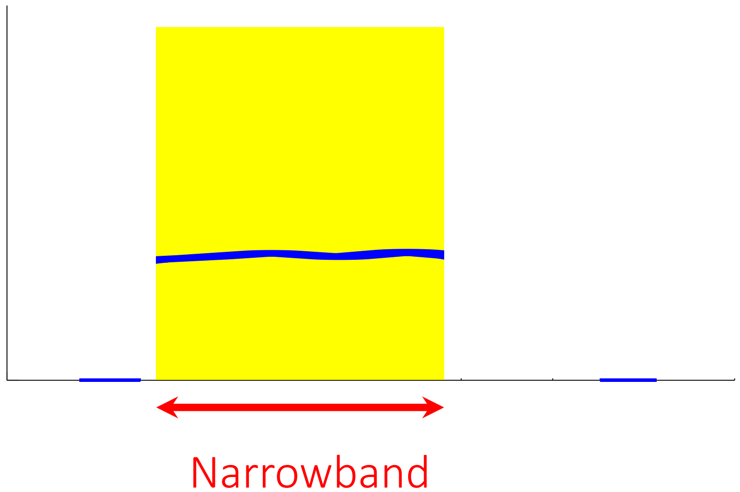
Flat Channel

# Narrowband Channel

$h(t)$ is channel impulse response.

$$y(t) = \sum_k h(\tau_k)\, x(t-\tau_k) \;= h(t) * x(t) \qquad \Leftrightarrow \qquad H(f)X(f)$$



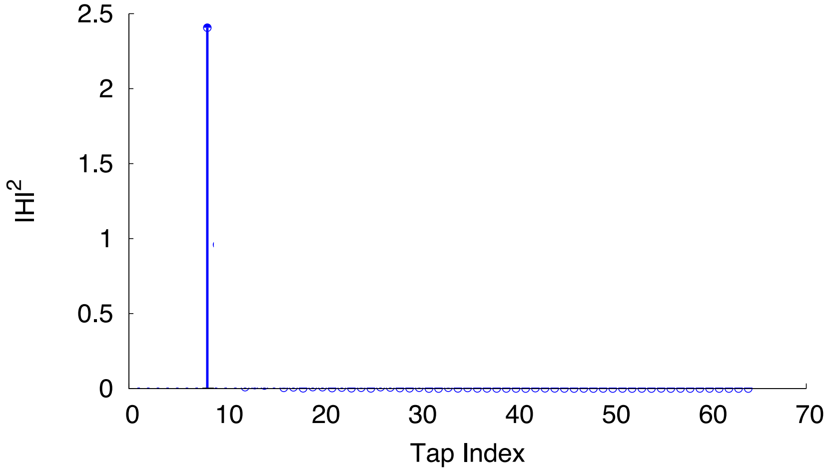

Symbol time: $T \propto \dfrac{1}{\text{Bandwidth}} \gg \tau_k$

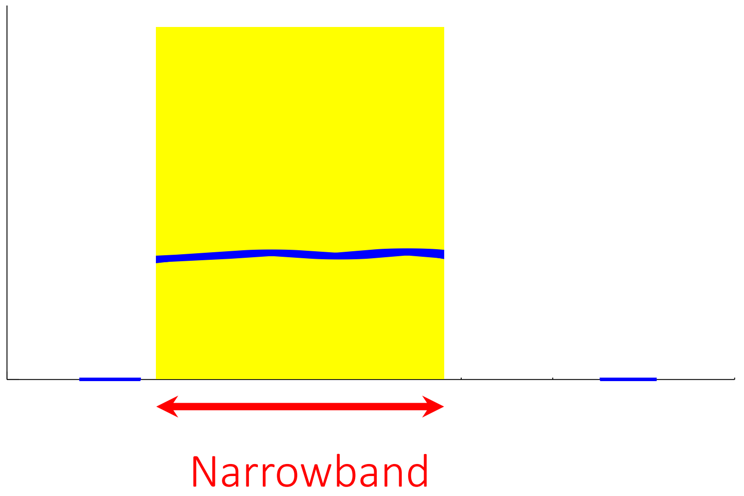
Flat Channel

# Narrowband Channel

$h(t)$ is channel impulse response.

$$y(t) = \sum_k h(\tau_k)\, x(t-\tau_k) \;= h(t) * x(t) \qquad \Leftrightarrow \qquad H(f)X(f)$$




Symbol time: $T \propto \frac{1}{\text{Bandwidth}} \gg \tau_k$

Flat Channel

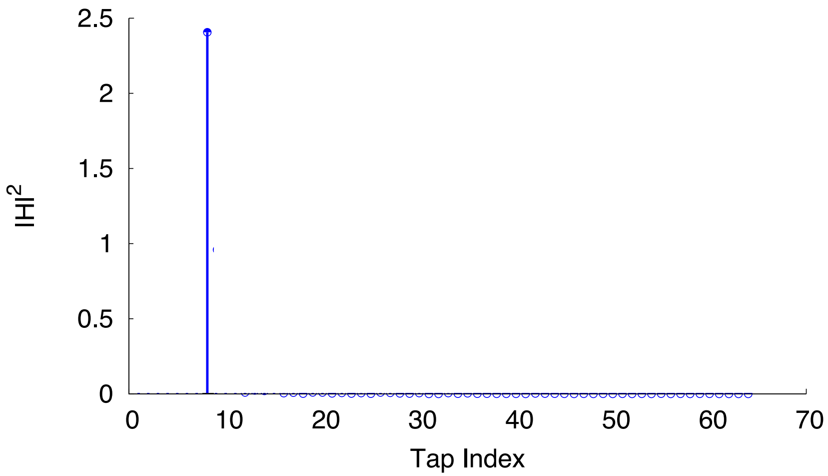
$$y(t) = \sum_k h(\tau_k)\, x(t-\tau_k) \;\approx \sum_k h(\tau_k)\, x(t) \;= \left(\sum_k h(\tau_k)\right) x(t) \;= hx(t)$$

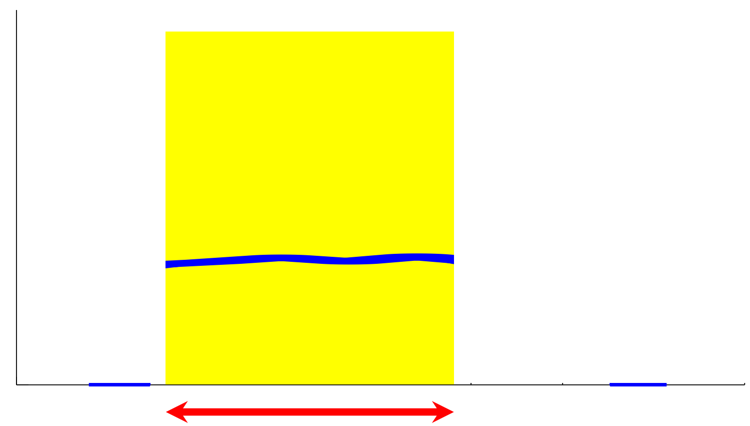
Narrowband Channel is Approximated by a Single Tap Channel

# Narrowband Channel

$h(t)$ is channel impulse response.

$$y(t) = \sum_k h(\tau_k)\, x(t) \ = h\, x(t) \qquad \Leftrightarrow \qquad h\, X(f)$$



Narrowband

Symbol time: $T \propto \frac{1}{\text{Bandwidth}} \gg \tau_k$

Flat Channel

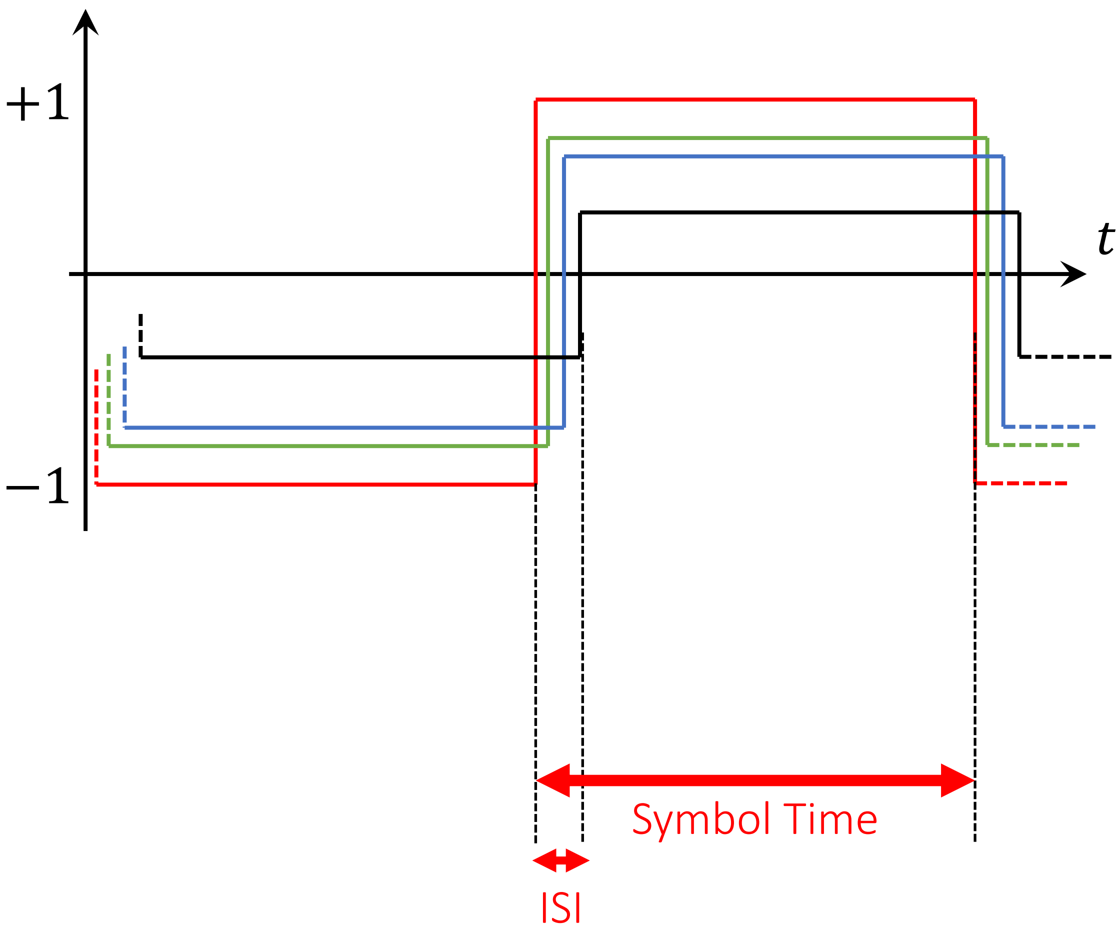
$$y(t) = \sum_k h(\tau_k)\, x(t - \tau_k) \ \approx \sum_k h(\tau_k)\, x(t) \ = \left(\sum_k h(\tau_k)\right) x(t) \ = hx(t)$$

Narrowband Channel is Approximated by a Single Tap Channel

# Narrowband vs. Wideband Channel

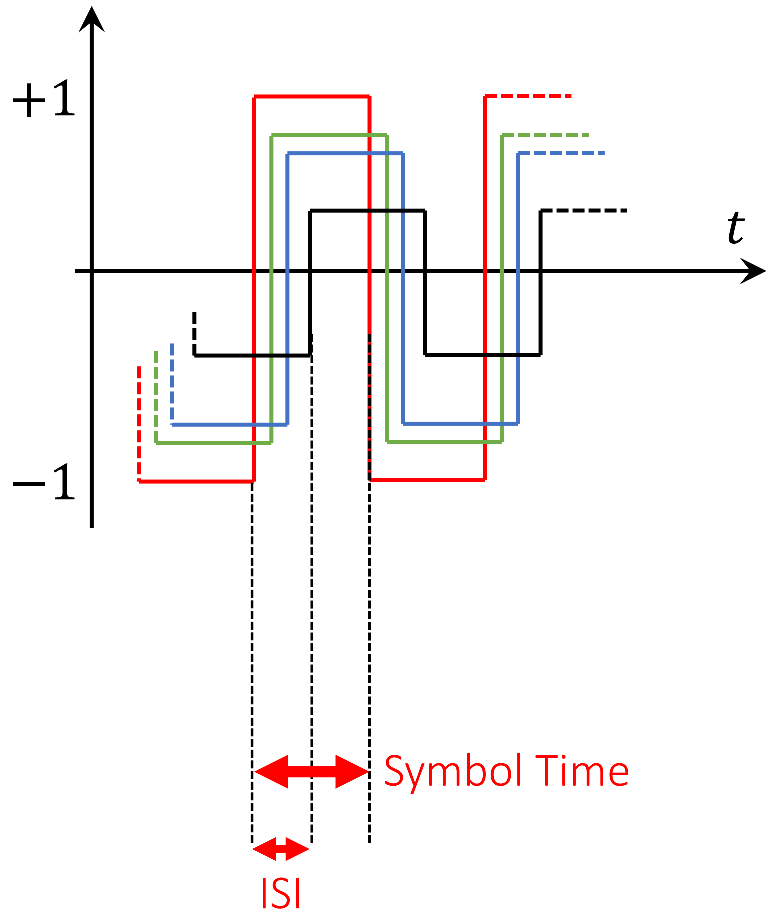
Narrowband Channel

$h\,x(t)$




Wideband Channel

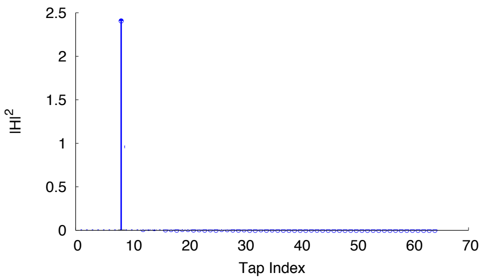
$h(t) * x(t)$



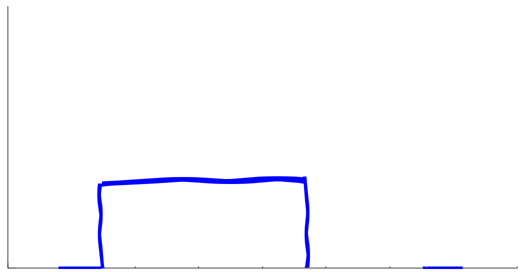
# Narrowband vs. Wideband Channel

**Narrowband Channel**

$h\,x(t) \quad \Longleftrightarrow \quad h\,X(f)$



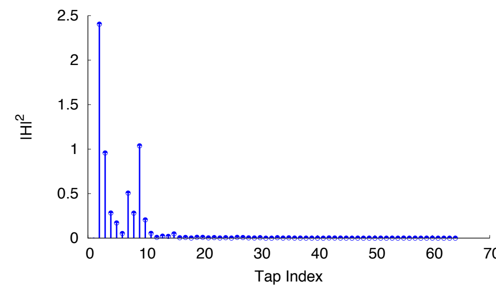

≈ Single Tap



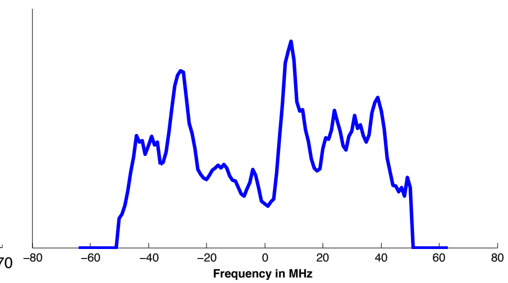
≈ Flat Channel

**Wideband Channel**

$h(t) * x(t) \quad \Longleftrightarrow \quad H(f)\,X(f)$




Multi-tap Channel (ISI)




Frequency Selective Channel

Need to correct for ISI to be able to decode correctly!

# Inter-Symbol-Interference

Sources of ISI:

- Multi-tap Channel

$$y(t) = h(t) * x(t)$$

- Pulse Shaping

$$y(t) = h(t) * p(t) * s(t)$$

- Other hardware filters

## How to deal with ISI?

## Channel Equalization

# Channel Equalization

Estimating $h(t)$

Correcting for $h(t)$

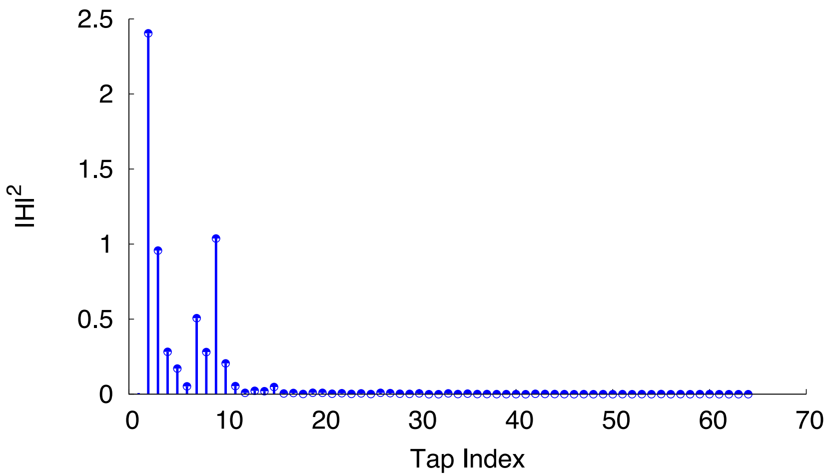
$$y(t) = h(t) * x(t)$$

$$\hat{x}(t) = f(t) * y(t) = f(t) * h(t) * x(t)$$

Ideally, $f(t) = h^{-1}(t)$

$$\hat{x}(t) = h^{-1}(t) * h(t) * x(t) = \delta(t) * x(t) = x(t)$$

# Estimating $h(t)$




$h(t)$ is a multi-tap channel

Can Estimate $h(t)$ as an FIR Filter:

- Limited number of paths
- Longer path power decays quickly
- Estimate as a filter with L taps
- Send a training sequence!

# Estimating $h(t)$

Send a training sequence of length N:

$$y[n] = \sum_{l=0}^{L} h[l]t[n-l] + v[n]$$

We will use Least Squares Estimator:

$$\{\hat{h}[0], \hat{h}[1], \dots, \hat{h}[L]\} = \underset{h[0], h[1], \dots h[L]}{\operatorname{argmin}} \sum_{n=L}^{N-1} \left| y[n] - \sum_{l=0}^{L} h[l]t[n-l] \right|^2$$

# Estimating $h(t)$

Send a training sequence of length N:

$$y[n] = \sum_{l=0}^{L} h[l]t[n-l] + v[n]$$

$$\begin{bmatrix} y[L] \\ y[L+1] \\ \vdots \\ y[N-1] \end{bmatrix} = \begin{bmatrix} t[L] & \cdots & t[0] \\ t[L+1] & \ddots & \vdots \\ \vdots & & \vdots \\ t[N-1] & \ldots & t[N-1-L] \end{bmatrix} \times \begin{bmatrix} h[0] \\ h[1] \\ \vdots \\ h[L] \end{bmatrix} + \begin{bmatrix} v[0] \\ v[1] \\ \vdots \\ v[L] \end{bmatrix}$$

$$\mathbf{y} = \mathbf{A}\mathbf{h} + \mathbf{v}$$

# Estimating $h(t)$

Send a training sequence of length N:

$$y[n] = \sum_{l=0}^{L} h[l]t[n-l] + v[n]$$

$$\mathbf{y} = \mathbf{A}\mathbf{h} + \mathbf{v}$$

To solve for $\mathbf{h}$, we need: $N - L \geq L + 1$ $\rightarrow N \geq 2L + 1$

Least Squares Solution: minimize $\|\mathbf{y} - \mathbf{A}\mathbf{h}\|^2$

$$\hat{\mathbf{h}} = \left(\mathbf{A}^{\dagger}\mathbf{A}\right)^{-1}\mathbf{A}^{\dagger}\mathbf{y}$$

$\mathbf{A}^{\dagger}$: Hermitian, Conjugate Transpose

# Channel Equalization

✔ Estimating $h(t)$

Correcting for $h(t)$

$$y(t) = h(t) * x(t)$$

$$\hat{x}(t) = f(t) * y(t) = f(t) * h(t) * x(t)$$

Ideally, $f(t) = h^{-1}(t)$

$$\hat{x}(t) = h^{-1}(t) * h(t) * x(t) = \delta(t) * x(t) = x(t)$$

# Correcting for $h(t)$

$$y[n] = \sum_{l=0}^{L} h[l]x[n-l] + v[n]$$

Need to find an inverse filter $\mathrm{f}$, such that:

$$\sum_{l=0}^{L'} f[l]\hat{h}[n-l] = \delta[n-d]$$

- $\mathbf{h}$ is FIR, $\mathbf{h}^{-1}$ is IIR → Hard to satisfy exactly.
- $d$ is equalization delay
- Try different $d$ to find the best equalizer

# Correcting for $h(t)$

$$y[n] = \sum_{l=0}^{L} h[l]x[n-l] + v[n]$$

Need to find an inverse filter f, such that:

$$\sum_{l=0}^{L'} f[l]\hat{h}[n-l] = \delta[n-d] \quad L' \geq L$$

$$\begin{bmatrix} \hat{h}[0] & 0 & \cdots & 0 \\ \hat{h}[1] & \hat{h}[0] & & \vdots \\ \vdots & \ddots & & \\ \hat{h}[L] & & & \\ 0 & \hat{h}[L] & \cdots & 0 \\ \vdots & & & \end{bmatrix} \times \begin{bmatrix} f[0] \\ f[1] \\ \vdots \\ \vdots \\ \vdots \\ f[L'] \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{bmatrix}$$

$$\widehat{\mathbf{H}}\mathbf{f} = \boldsymbol{\delta}_{\mathbf{d}}$$

$$\text{minimize} \left\| \widehat{\mathbf{H}}\mathbf{f} - \boldsymbol{\delta}_{\mathbf{d}} \right\|^2$$

$$\hat{\mathbf{f}} = \left(\widehat{\mathbf{H}}^\dagger \widehat{\mathbf{H}}\right)^{-1} \widehat{\mathbf{H}}^\dagger \boldsymbol{\delta}_{\mathbf{d}}$$

$\widehat{\mathbf{H}}$: Toeplitz Matrix

# Channel Equalization

✓ Estimating $h(t)$

✓ Correcting for $h(t)$

$$y(t) = h(t) * x(t)$$

$$\hat{x}(t) = f(t) * y(t) = f(t) * h(t) * x(t)$$

Ideally, $f(t) = h^{-1}(t)$

$$\hat{x}(t) = h^{-1}(t) * h(t) * x(t) = \delta(t) * x(t) = x(t)$$

# Channel Equalization

✔ Estimating $h(t)$

Solve Least Squares

$$\hat{\mathbf{h}} = \left(\mathbf{A}^{\dagger}\mathbf{A}\right)^{-1}\mathbf{A}^{\dagger}\mathbf{y}$$

✔ Correcting for $h(t)$

Solve Least Squares

$$\hat{\mathbf{f}} = \left(\widehat{\mathbf{H}}^{\dagger}\widehat{\mathbf{H}}\right)^{-1}\widehat{\mathbf{H}}^{\dagger}\boldsymbol{\delta}_{\mathbf{d}}$$

Apply $f(t)$ to $y(t)$

Need to Solve Least Squares Twice!

Computationally Intensive!

# Channel Equalization

## Direct Least Squares
## Skip Estimating h altogether

- Send a training sequence of length N:

$$y[n] = \sum_{l=0}^{L} h[l]t[n-l] + v[n]$$

- Find inverse filter f such that:

$$\sum_{l=0}^{L'} f[l]y[n-l] = t[n-d] \;\; \rightarrow t[n] = \sum_{l=0}^{L'} f[l]y[n+d-l]$$

$\mathbf{t} = \mathbf{Y}\mathbf{f}$ ➡ Least Squares: $\hat{\mathbf{f}} = \left(\mathbf{Y}^{\dagger}\mathbf{Y}\right)^{-1}\mathbf{Y}^{\dagger}\mathbf{t}$

# Channel Equalization

| Linear | Non-Linear |
| --- | --- |
| LLSE (Linear Least Squares Error) | DFE (Decision Feedback) |
| MMSE (Minimum Mean Squares Error) | MLSE (Maximum Likelihood Sequence Estimation) |
| ZF (Zero Forcing) | |
| Lower Complexity | Lower Noise |

Many more tradeoffs:

ISI vs Noise vs Computation vs Convergence

# Channel Equalization

## Eliminating ISI vs Amplifying Noise

ZF (Zero Forcing) Equalizer:

$$y(t) = h(t) * x(t) + v(t) \quad \Leftrightarrow \quad Y(f) = H(f)X(f) + V(f)$$

$$\hat{X}(f) = \frac{Y(f)}{H(f)} = \underbrace{X(f)}_{\text{No ISI}} + \frac{V(f)}{H(f)}$$

Noise Amplification

Channel Attenuation: $H(f) < 1$

Channel Fading: $H(f) \approx 0$

**Good Equalizer balances between reducing ISI & amplifying noise**

# Channel Equalization

## Computational Complexity vs Convergence Time

### Algorithms to Solve Least Squares:

$N$: Length of training sequence

| | Computation | Convergence |
|---|---|---|
| LMS: Least Mean Squares | $2N+1$ | $> 10N$ |
| RLS: Recursive Least Squares | $2.5N^2 + 4.5N$ | $\approx N$ |

Larger $N$:

- More computation
- Longer training sequences
- Lower error: ISI & Noise

# Definitions & Variables

- $x(t)$: Transmitted Signal
- $v(t)$: Additive Gaussian Noise
- $y(t)$: Received Signal
- $\tau$: Time delay of the signal
- $h$: Single Tap Channel Coefficient.
- $\tilde{h}$: Estimate of the Channel Coefficient.
- $\tau_k$: Time delay of the $k^{th}$ propagation path
- $\alpha_k$: Attenuation of the $k^{th}$ propagation path
- $\phi_k$: Phase of the $k^{th}$ propagation path
- $h(t)$: Multi-Tap Channel Impulse Response
- $H(f)$: Frequency Response of the Channel
- $X(f)$: Frequency Spectrum of transmitted signal
- $V(f)$: Frequency Spectrum of noise
- $Y(f)$: Frequency Spectrum of received signal
- $h[l]$: Coefficient of the $l^{th}$ channel tap
- $\hat{h}[l]$: Estimate of the channel coefficients

- $T$: Symbol time
- $*$ : Convolution
- $\lambda$: Wavelength of the signal.
- $d'$: Distance between TX and RX
- $f_c$: Carrier Frequency
- $p(t)$: Pulse of pulse shaping filter
- $s(t)$: Modulated Symbols
- $h^{-1}(t)$: Inverse Channel Response
- $f(t)$: Channel Equalization Filter
- $\hat{x}(t)$: Equalized Signal
- $\delta(t)$: Impulse Function
- $y[n]$: Sampled received signal
- $t[n]$: Training Sequence
- $N$: Length of the Training Sequence
- $f[l]$: Equalization Filter coefficients
- $n$: Symbol index

- $\mathbf{y}$: Vector of $y[n]$ samples
- $\mathbf{h}$: Vector of $h[l]$ coefficients
- $\mathbf{A}$: Matrix of $t[n]$ training samples
- $\mathbf{v}$: Vector of $v[n]$ noise samples
- $\mathbf{f}$: Vector oxf $f[l]$ coefficients
- $\hat{\mathbf{f}}$: Estimate of $\mathbf{f}$
- $\hat{\mathbf{h}}$: Estimate of $\mathbf{h}$
- $\hat{\mathbf{f}}$: Estimate of $\mathbf{f}$
- $\widehat{\mathbf{H}}$: Toeplitz matrix of $\mathbf{h}$
- $\mathbf{t}$: Vector of $t[n]$ training samples
- $\mathbf{Y}$: Matrix of $y[n]$ received samples
- $d$: Equalization delay
- $(\quad)^{-1}$: Matrix Inverse
- $(\quad)^{\dagger}$: Conjugate Transpose of Matrix
- $L$: Number of channel taps
- $L'$: Number of Equalization filter taps