OFDM: Orthogonal Frequency Division Multiplexing

The key issue is in combating "dispersive channels". Broadly speaking these are the channels which have a frequency selective property that leads to distortion in the time domain pulse transmissions. Inter-Symbol Interference (ISI) is the technical term used for such a scenario where bits of a certain symbol leak into the neighbor thereby leading to demodulation errors.

An alternative to channel estimation and equalization is to transmit bits across "sub-carriers" so that mutual interference across unequal subcarriers is null. This is a consequence of the fact that for any LTI system ($h(t) \leftrightarrow H(f)$), the complex exponential or cisoid $e^{j2\pi ft}$ is an eigenfunction.

It is known that for a Hermitian (and hence a real valued) operator the eigenfunctions form an orthogonal basis. That is $\langle e^{j2\pi f_1 t}, e^{j2\pi f_2 t}\rangle = \int_{-\infty}^{\infty} e^{j2\pi(f_1-f_2)t}\,dt$

This fact can also be verified using the definition of the Fourier Transform.

$$= \delta(f_1 - f_2)$$
$$= 0 \text{ when } f_1 \neq f_2$$

Thus, if we transmit $x(t) = \sum_1 B_n e^{j2\pi f_n t}$

then we can recover the bits using the orthogonality relation.

$$r(t) = x(t) * h(t) = \sum_1 H(f_n) B_n e^{j2\pi f_n t}$$

(Prove this)

All we need is a way of estimating $H(f_n)$ for exact recovery of B_n.

$$\int_{-\infty}^{\infty} r(t) e^{-j2\pi f_n t}\,dt = H(f_n) B_n.$$

Notice that this operation is nothing but the FT of $r(t) \longleftrightarrow R(f)$ evaluated at $f = f_n$. This is also not surprising as the FT is closely linked to the convolution property of LTI impulse response functions

Theorem: (Spectral Theorem) Any Hermitian linear operator has purely real eigenvalues and eigenvectors that form an orthogonal basis. A matrix/operator is Hermitian if

$A^{\dagger} = A$ where $\dagger$ = transpose conjugate

proof: If $(\lambda, \underline{v})$ is an eigenpair of A then

$$A\underline{v} = \lambda \underline{v} \Longrightarrow \underline{v}^{\dagger} A^{\dagger} = \lambda^{*} \underline{v}^{\dagger} \Longrightarrow \underline{v}^{\dagger} A = \lambda^{*} \underline{v}^{\dagger}$$

$$\Longrightarrow \underline{v}^{\dagger}(A\underline{v} - \lambda^{*}\underline{v}) = 0 \Longrightarrow \lambda = \lambda^{*}.$$

Now if $\underline{v}_1, \underline{v}_2$ are two eigenvectors such that

$\lambda_1 \neq \lambda_2$ then $A\underline{v}_1 = \lambda_1 \underline{v}_1$, $A\underline{v}_2 = \lambda_2 \underline{v}_2$

$$\underline{v}_1^{\dagger} A = \underline{v}_1^{\dagger} \lambda_1^{*}, \quad \underline{v}_2^{\dagger} A = \underline{v}_2^{\dagger} \lambda_2^{*}$$

$$\Longrightarrow \underline{v}_2^{\dagger} A \underline{v}_1 = \underline{v}_2^{\dagger} \lambda_2^{*} \underline{v}_1$$

$$\Longrightarrow \underline{v}_2^{\dagger}\underline{v}_1 = \frac{\lambda_2^{*}}{\lambda_1} \underline{v}_2^{\dagger} v_1$$

$$\Longrightarrow \underline{v}_2^{\dagger}\underline{v}_1 \left[1 - \frac{\lambda_2^{*}}{\lambda_1}\right] = 0$$

So if $\lambda_1 \neq \lambda_2$ then $\underline{v}_2^{\dagger}\underline{v}_1 = 0$

$= \langle v_2, v_1 \rangle = 0$

Of course, in a practical OFDM transmitter we cannot have pulse basis as $e^{j2\pi f_n t}$ as the complex exponential has an infinite time width. As a result, we choose the more practical

$$p_n(t) = e^{j2\pi f_n t} \mathbb{1}_{[0,T]}$$

where T is called the "signaling interval".

Effect of finite signaling interval on LTI orthogonality:

consider $p_n(t) = e^{j2\pi f_n t} I_{[0,T]}$, we have $\hat{P}_n(f) = \mathcal{F}(p_n(t))$

as $\hat{P}_n(f) = T \operatorname{sinc}((f-f_n)T) e^{-j\pi f T}$

A well known fact is that $\operatorname{sinc}(ax)$ decays rapidly (envelope wise) as $x \sim \propto a^{-1}$ for large α.

For a dispersive or a frequency selective channel, we denote B_c as the "channel coherence bandwidth" which is the length of the frequency interval over which the channel's freq response is approximately constant. Therefore if $\frac{1}{T} << B_c$ then over a dispersive channel $h(t)$ we have

$$p_n(t) * h(t) \longleftrightarrow P_n(f) H(f) \simeq H(f_n) P_n(f)$$

That is, the channel frequency characteristic can be taken to be approximately constant (Note that making $\frac{1}{T}$ too small will adversely affect the data rate)

Orthogonality wise, we have

$$\int_0^T e^{j2\pi(f_n - f_m)t} dt = \frac{e^{j2\pi(f_n-f_m)T} - 1}{j2\pi(f_n - f_m)}$$

So if $f_n - f_m \neq 0$ we have orthogonality maintained (weakly) as

$$(f_n - f_m)T \in \mathbb{Z}.$$

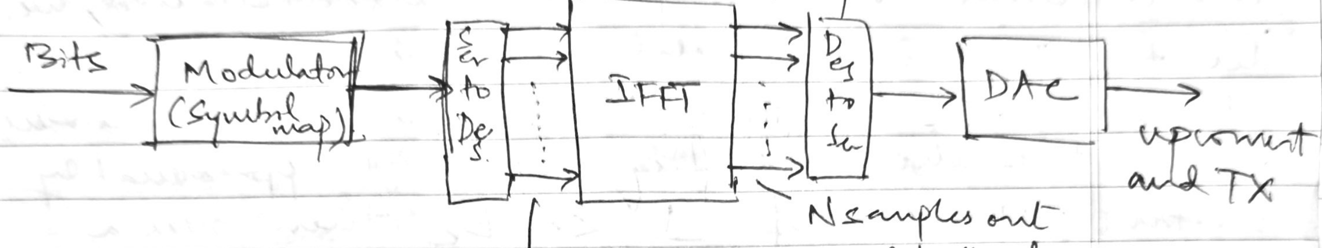
Therefore, with this scheme we can send N symbols at once over N subcarriers that leads to an effective bandwidth of $\frac{N}{T}$ for the entire transmission.

If we can pick f_i, f_j ST $(f_i - f_j)T$ is a non zero integer and $T >> \frac{1}{B_c}$ (channel delay spread) we can ensure spectral orthogonality in principle

Practical Implementation of OFDM TX:

As alluded to earlier, the FDM closely resembles that of a FT when the multiplexing subcarriers live on an evenly spaced spectrum.

This fact is even more pronounced when encoding and mapping bits for transmission as one could use an IFFT ~~block~~ based DSP block to parallelize the FDM.



Basically, we wish to feed the DAC the time domain signal $x(t)$ that is sampled at a rate of $f_s > \frac{N}{T}$ (why?)

Thus $b[k] = x(kT_s) = \sum_{n=0}^{N-1} B[n] e^{\frac{j2\pi nk}{N}}$

where $B[n]$ is actually the symbol sequence (after the modulator performs constellation mapping to produce complex valued symbols). The number of subcarriers is usually picked to be a power of 2.

Thus, at the receive end, after downconversion, we can perform

$$\boxed{B[n] = \frac{1}{N}\sum_{k=0}^{N-1} b[k] e^{-j\frac{2\pi nk}{N}}}$$

This enables us to retrieve $B[n]$ with little to no ISI. While the frequency selective nature of the channel can be countered by lowering the data rate, the maintenance of the orthogonality property is more crucial, and can be made exact.

The cyclic Prefix:

Prior to the DAC stage in transmission, adding a cyclic prefix to close the loop on the FFT can enable strong orthogonality among subcarriers even with finite signaling time duration.

Assume that $h(t) = \text{DAC} * \text{Tx filter} * \text{channel} * \text{Rx filter}$

Then we have $r(t) = \sum_{k=0}^{N-1} b[k]\, h(t - kT_s)$

$$\Rightarrow r[m] = \sum_{k=0}^{N-1} b(k)\, h[m-k] \quad \left(\text{sampling } r(t) \text{ @ } f_s\right)$$

Now, in general, the discrete time channel impulse response $h[n] = h(nT_s)$ need not be non zero for the entire duration of N symbols, and for some reason it is assumed that WLOG $h[n] = 0$ for $n < 0$ and $n \geq L$

where $\underline{L \leq N}$. (This is a design parameter)

So if we take the N-point DFT of $h[n]$

$$H[k] = \sum_{n=0}^{N-1} h[n] e^{-\frac{j 2\pi k n}{N}} = \sum_{n=0}^{L-1} h[n] e^{-\frac{j 2\pi k n}{N}}$$

Note that, $r[m] = \sum_{k=0}^{L-1} h[k]\, b[m-k]$ is a linear convolution, whose DFT/FFT spectrum is inconvenient if the ~~bit~~ ~~stream~~ OFDM stream $b[k]$ is an N point sequence.

Instead, if we consider the circular convolution

$$\hat{V}[m] = (h \odot b)[m] = \sum_{k=0}^{N-1} h[k \bmod N]\, b[(m-k) \bmod N]$$

Then $\hat{V}[k] = \hat{H}[k] \hat{B}[k]$ for $k = 0, 1, \ldots, N-1$

(This is the discrete time analogue of Fourier convolution thm)

Proof:
$$\hat{V}[k] = \sum \hat{v}[n] e^{-j\frac{2\pi nk}{N}}$$

$$= \cancel{\sum h[k \bmod N]\, b[(n-k) \bmod N]}$$

$$= \sum_{n=0}^{N-1}\left(\sum_{q=0}^{N-1} h[q \bmod N]\, b[(n-q) \bmod N]\right) e^{-j\frac{2\pi nk}{N}}$$

$$= \sum_{q=0}^{N-1} h[q \bmod N] \sum_{n=0}^{N-1} b[(n-q) \bmod N]\, e^{-j\frac{2\pi nk}{N}}$$

Thus summation holds q fixed

Then w goes from $-q$ to $N-q-1$ so if $n-q = w$

$$\sum_{q=0}^{N-1} h[q \bmod N] \sum_{w=-q}^{N-q-1} b(w \bmod N)\, e^{-j\frac{2\pi(q+w)k}{N}}$$

$$= \sum_{q=0}^{N-1} h[q]\, e^{-j\frac{2\pi qk}{N}} \sum_{w=-q}^{N-q-1} b(w \bmod N)\, e^{-j\frac{2\pi wk}{N}}$$

$= \tilde{H}[K]\, \tilde{B}[K]$ after rearranging indices!

Consequently, if we expand the circular convolution

$$\hat{v}[m] = (h \circledast b)[m] = \sum_{K=0}^{\min(L-1,m)} h[K]\, b[m-K]$$
$$+ \sum_{K=m+1}^{L-1} h[K]\, b[m-K+N]$$

We note that $h[K]$ for $K > L$ is zero so the second summation component is the difference b/w linear and cyclic convolution

Notice that $b[(m-k) \bmod N]$ was rewritten as $b[m-k+N]$ in the second term of the sum as the range of k, m is restricted to $L-1 < N$ in the first term making the modulus irrelevant.

Consequently, to transition from linear to circular convolution one needs to pad $b[k]$. ~~[illegible]~~

<u>Note that</u>: $b[m-k] = 0$ if $k > m$ but $\leq L-1$

Thus, if we send

$$b[N-(L-1)], b[N-(L-2)], \ldots$$
$$\ldots, b[N-1], b[0],$$
$$b[1], b[2], \ldots, b[N-1]$$

instead of the classic $b[0], b[1], \ldots, b[N-1]$ for linear convolution, we obtain an equivalent circular convolution

Of course this is at a transmission cost of L extra prefix bits which do not include any new information thus adding redundancy and degrading channel capacity.

Now, the receiver implementation is very simple, we run the N point DFT on $r[m]$ to obtain $\tilde{R}[k]$

$$\tilde{R}[k] = H[k]B[k] + N[k]$$

And using ML estimation one can evaluate the signal component from which $B[k]$ can be read off via a single stage of channel equalization

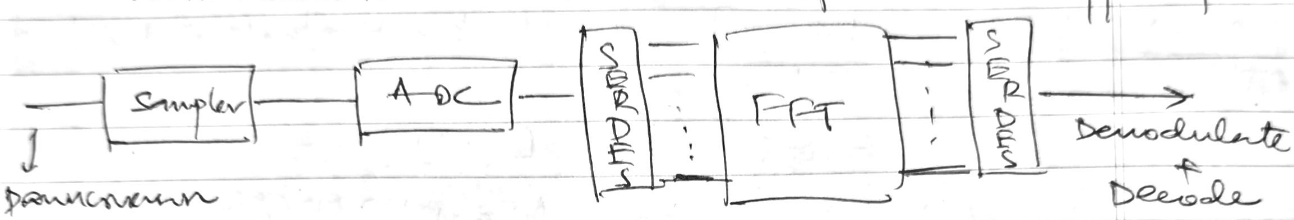
<u>Pros</u>: N fold TX gain via multiplexing (cost of L ignored)
Simplified TX/RX implementation.

<u>Cons</u>: PAPR is very high, needs FFT blocks,
guard / cyclic prefix might have to be larger for heavy fading channels.

How does OFDM get information about the channel?

It uses specialized OFDM "frames" called pilot tones prior to payload frames, for channel estimation. Pilot frames are OFDM frames which are pre-registered on both TX and RX sides which have no symbol decoding uncertainty

→ As a result they only carry information relevant to synchronization errors, timing errors, and channel non-idealities such as multipath and Doppler fading



An OFDM frame has $N' = N + L$ symbols ~~abc~~ where $N' = \#$ of subcarriers and L = size of cyclic prefix

Thus, both the IFFT and FFT blocks are <u>N' point DFT operations</u>

For fine tuned frame synchronization one can use the cyclic prefix information to ~~align~~ align the RX frame and match the samples before decoding (use autocorrelation, find the lag amount at the peak and adjust accordingly). This is only useful for fine timing as the prefix cannot exceed the length of the frame in principle.