ECE 461: Digital Communications

Lecture 14: Transmitter-Centric ISI Harnessing: Orthogonal Frequency Division Modulation (OFDM)

Introduction

In the previous lecture, we took a first order transmitter-centric approach to dealing with ISI: the focus was on eliminating the effects of ISI. In this lecture we take a more balanced view: harnessing the benefits of ISI instead of just treating it as interference, continuing our transmitter-centric approach. Our goal is to get the full benefit of the fact that multiple delayed copies of the transmit symbols appear at the receiver while still employing a receiver no more complicated than the one used over an AWGN channel. While this seems to be a tall order, we will see a remarkable scheme that achieves exactly this. In a sense, it is a natural culmination of the various ISI mitigation techniques we have seen in the course of the past several lectures. The scheme that converts the frequency selective ISI channel into a plain AWGN channel is known as *orthogonal frequency division modulation* (OFDM) and is the main focus of this lecture.

An Ideal Situation

Consider the frequency selective model that we have been working with as a good approximation of the wireline channel:

$$y[m] = \sum_{\ell=0}^{L-1} h_\ell x[m-\ell] + w[m], \quad m \geq 1. \tag{1}$$

Since we know how to communicate reliably over an AWGN channel (cf. Lecture 7), it would be ideal (not to mention, easy) if the channel with ISI can *somehow* (by appropriate transmitter and reciever operations) be converted into an AWGN one: say,

$$y[m] = \hat{h} x[m] + w[m], \quad m \geq 1. \tag{2}$$

In such a case, we could simply and readily use the transmitter and receiver techniques developed already for the AWGN channel (available "off-the-shelf", so to say).

While this is asking for a bit too much, we will see that we can get somewhat close: indeed, we will convert the ISI channel in Equation (1) into a *collection* of AWGN channels, each of different noise energy level:

$$\hat{y}[N_c k + n] = \hat{h}_n \hat{x}[N_c k + n] + \hat{w}[N_c k + n], \quad k \geq 0, \quad n = 0 \ldots N_c - 1. \tag{3}$$

The idea is that the time index m is replaced by $N_c k + n$. The inputs are voltages $\hat{x}$. The additive noise $\hat{w}[\cdot]$ is white Gaussian (zero mean and variance σ^2). We observe that there are N_c different AWGN channels, one for each $n = 0, \ldots, N_\mathrm{c} - 1$. We can make two further observations:

- each of the N_c AWGN channels has a *different* operating SNR: the n^{th} channel has an SNR equal to $\hat{h}_n^2\mathsf{SNR}$ where SNR is, as usual, the ratio of the transmit energy to the noise energy;
- each of the N_c AWGN channels is available for use only a *fraction* $\frac{1}{N_\mathrm{c}}$ of the time.

Such a collection of non-interfering AWGN channels is called a *parallel* AWGN channel. The individual AWGN channels within the collection are known as *sub-channels.* Our understanding of efficient reliable communication over the AWGN channel suggests a natural strategy to communicate over the parallel AWGN channel as well: we can split the information bits so that we communicate over each sub-channel *separately.* The only choice remaining is how to split the total power budget amongst the sub-carriers, say power P_n to the n^{th} sub-carrier, so that

$$\sum_{n=0}^{N_\mathrm{c}-1} P_n = P, \tag{4}$$

where P is the total power budget for reliable communication. With a transmit power constraint of P_n, the overall SNR of the n^{th} sub-channel is

$$\frac{P_n \hat{h}_n^2}{\sigma^2}. \tag{5}$$

Thus, with an appropriate coding and decoding mechanism reliable communication is possible (cf. Lecture 7) on the n^{th} AWGN sub-channel at rate

$$R_n = \frac{1}{2N_\mathrm{c}} \log_2 \left(1 + \frac{P_n \hat{h}_n^2}{\sigma^2} \right), \tag{6}$$

measured in bits/symbol. The factor of $1/N_\mathrm{c}$ in the rate appears because each of the sub-channels is available for use only a fraction $1/N_\mathrm{c}$ of the time. The total rate of reliable communication is

$$\sum_{n=0}^{N_\mathrm{c}-1} R_n = \frac{1}{2N_\mathrm{c}} \sum_{n=0}^{N_\mathrm{c}-1} \log_2 \left(1 + \frac{P_n \hat{h}_n^2}{\sigma^2} \right). \tag{7}$$

We can now split the power to *maximize* the rate of reliable communication over the parallel AWGN channel:

$$\max_{P_n \geq 0, \sum_{n=0}^{N_\mathrm{c}-1} P_n = P} \frac{1}{2N_\mathrm{c}} \sum_{n=0}^{N_\mathrm{c}-1} \log_2 \left(1 + \frac{P_n \hat{h}_n^2}{\sigma^2} \right). \tag{8}$$

The optimal power split can be derived explicitly and is explored in a homework exercise. The main property of this optimal power split is that

> the larger the "quality" of a sub-channel, the more the power that is allocated to it, and hence the larger the corresponding data rate of reliable communication.

In the rest of this lecture, we will see how to get from the frequency-selective channel we have (Equation (1)), to the parallel AWGN channel (Equation (3)) we find easy to work with. We will be able to make this transition by some very simple signal processing techniques. Interestingly, these signal processing techniques are *universally* applicable to every wireline channel, i.e., they do not depend on the exact values of channel coefficients $h_0, \ldots, h_{L-1}$. This makes OFDM a very *robust* communication scheme over the frequency-selective channel.

Cyclic Prefix

Suppose we have mapped our information bits into N_c voltages. We will revisit the issue of how these voltages were created from the information bits at a slightly later point in this lecture. For now, we write them as a vector:

$$\mathbf{d} = [d[0], d[1], \ldots, d[N_\mathrm{c}-1]]^t.$$

We use these N_c voltages to create an $N_\mathrm{c}+L-1$ block of *transmit* voltages as:

$$\mathbf{x} = [d[N_\mathrm{c}-L+1], d[N_\mathrm{c}-L+2], \ldots, d[N_\mathrm{c}-1], d[0], d[1], \ldots, d[N_\mathrm{c}-1]]^t, \tag{9}$$

i.e., we add a *prefix* of length $L-1$ consisting of data symbols rotated cyclically (Figure 1). The first $L-1$ transmitted symbols contain the "data" symbols $d[N_\mathrm{c}-(L-1)], \ldots, d[N_\mathrm{c}-1]$. The next N_c transmitted voltages or symbols contain the "data" symbols $d[0], d[1], \ldots, d[N_\mathrm{c}-1]$. In particular, for a 2-tap frequency-selective channel we have the following result of cyclic precoding:

$$\begin{aligned} x[1] &= d[N_\mathrm{c}-1] \\ x[2] &= d[0] \\ x[3] &= d[1] \\ &\vdots \\ x[N_\mathrm{c}+1] &= d[N_\mathrm{c}-1] \end{aligned}$$

With this input to the channel (1), consider the output

$$y[m] = \sum_{\ell=0}^{L-1} h_\ell x[m-\ell] + w[m], \qquad m = 1, \ldots, N_\mathrm{c} + 2(L-1).$$

The first $L-1$ elements of the transmitted vector $\mathbf{x}$ were constructed from circularly wrapped elements of the vector $\mathbf{d}$, which are included in the last $N_c - 1$ elements of $\mathbf{x}$. The receiver hence ignores the first $L-1$ received symbols $y[1], \ldots, y[L-1]$. The ISI extends over the first $L-1$ symbols and the receiver ignores it by considering only the output over the time interval $m \in [L, N_c + L - 1]$. Let us take a careful look at how the N receive voltages (received at times L through $N_c + L - 1$) depend on the transmit voltages $d[0], \ldots, d[N_c - 1]$:

$$y[m] = \sum_{\ell=0}^{L-1} h_\ell d[(m - L - \ell) \text{ modulo } N_c] + w[m]. \tag{10}$$

See Figure (1).

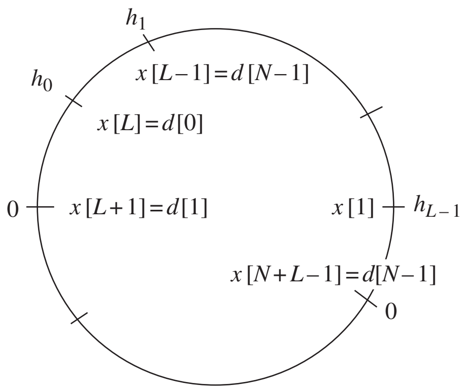

Figure 1: Convolution between the channel ($\mathbf{h}$) and the input ($\mathbf{x}$) formed from the data symbols ($\mathbf{d}$) by adding a cyclic prefix. The output is obtained by multiplying the corresponding values of $\mathbf{x}$ and $\mathbf{h}$ on the circle, and outputs at different times are obtained by rotating the x-values with respect to the h-values. The current configuration yields the output $y[L]$.

Denoting the received voltage vector of length N_c by

$$\mathbf{y} = [y[L], \ldots, y[N_c + L - 1]]^t,$$

and the channel by a vector of length N_c

$$\mathbf{h} = [h_0, h_1, \ldots, h_{L-1}, 0, \ldots, 0]^t, \tag{11}$$

(10) can be written as

$$\mathbf{y} = \mathbf{h} \otimes \mathbf{d} + \mathbf{w}. \tag{12}$$

Here we denoted

$$\mathbf{w} = [w[L], \ldots, w[N_c + L - 1]]^t, \tag{13}$$

as a vector of i.i.d. $\mathcal{N} \sim (0, \sigma^2)$ random variables. The notation of $\otimes$ to denote the *cyclic convolution* in (12) is standard in signal processing literature. The point of all this manipulation will become clear when we review a key property of cyclic convolution next.

Discrete Fourier Transform

The discrete Fourier transform (DFT) of a vector (such as $\mathbf{d}$) is also another vector of the same length (though the entries are in general, complex numbers). The different components of the discrete Fourier transform of the vector $\mathbf{d}$, denoted by DFT($\mathbf{d}$), are defined as follows:

$$\tilde{d}_n := \frac{1}{\sqrt{N_\mathrm{c}}} \sum_{m=0}^{N_\mathrm{c}-1} d[m] \exp\left(\frac{-\mathrm{j}2\pi nm}{N_\mathrm{c}}\right), \qquad n = 0, \dots, N_\mathrm{c} - 1. \tag{14}$$

Even though the voltages $d[\cdot]$ are real, the DFT output $\tilde{d}_n$ are complex. Nevertheless, there is *conjugate symmetry*:

$$\tilde{d}_n = \tilde{d}^*_{N_\mathrm{c}-1-n}, \quad n = 0, \dots, N_\mathrm{c} - 1. \tag{15}$$

DFTs and circular convolution are crucially related through the following equation (perhaps the most important of all in discrete time digital signal processing):

$$\mathrm{DFT}(\mathbf{h} \otimes \mathbf{d})_n = \sqrt{N_\mathrm{c}}\mathrm{DFT}(\mathbf{h})_n \cdot \mathrm{DFT}(\mathbf{d})_n, \qquad n = 0, \dots, N_\mathrm{c} - 1. \tag{16}$$

The vector $[\tilde{h}_0, \dots, \tilde{h}_{N_\mathrm{c}-1}]^t$ is defined as the DFT of the L-tap channel $\mathbf{h}$, multiplied by $\sqrt{N_\mathrm{c}}$,

$$\tilde{h}_n = \sum_{\ell=0}^{L-1} h_\ell \exp\left(\frac{-\mathrm{j}2\pi n\ell}{N_\mathrm{c}}\right). \tag{17}$$

Thus we can rewrite (12) as

$$\tilde{y}_n = \tilde{h}_n \tilde{d}_n + \tilde{w}_n, \qquad n = 0, \dots, N_\mathrm{c} - 1. \tag{18}$$

Here we have denoted $\tilde{w}_0, \dots, \tilde{w}_{N_\mathrm{c}-1}$ as the N_c-point DFT of the noise vector $w[1], \dots, w[N_\mathrm{c}]$. Observe the following.

- Even though the received voltages $y[\cdot]$ are real, the voltages at the output of the DFT $\tilde{y}_n$ are complex. Thus it might seem odd that we started out with N real numbers and ended up with $2N$ real numbers. But there is a redundancy in the DFT output $\tilde{y}[\cdot]$. Specifically,

$$\tilde{y}_n = \tilde{y}^*_{N_\mathrm{c}-1-n}. \tag{19}$$

  In other words, the real parts of $\tilde{y}_n$ and $\tilde{y}_{N_\mathrm{c}-1-n}$ are the same. Further, the imaginary parts of $\tilde{y}_n$ and $\tilde{y}_{N_\mathrm{c}-1-n}$ are negative of each other.

- Even though the noise voltages $w[\cdot]$ are real, the noise voltages at the output of the DFT $\tilde{w}_n$ are complex. Just as before, there is a redundancy in the noise voltages:

$$\tilde{w}_n = \tilde{w}^*_{N_\mathrm{c}-1-n}, \quad n = 0, \dots, N_\mathrm{c} - 1. \tag{20}$$

We know that the noise voltages $w[m]$ were white Gaussian. What are the statistics of the DFT outputs? It turns out that they are *also* white Gaussian: the real and imaginary parts of $\tilde{w}_0, \ldots, \tilde{w}_{\frac{N_c}{2}-1}$ are all independent and identically distributed as Gaussian with zero mean and variance σ^2 (here we supposed for notational simplicity that N_c is even, so $\frac{N_c}{2}$ is an integer).

- Even though the channel coefficients h_ℓ are real (and zero for $\ell = L, \ldots, N_c - 1$), the values at the output $\tilde{h}_n$ of the DFT are complex (and, in general, non-zero for all values $n = 0, \ldots, N_c - 1$). Again, observe that

$$\tilde{h}_n = \tilde{h}[N_c - 1 - n]^*, \quad , n = 0, \ldots, N_c - 1. \tag{21}$$

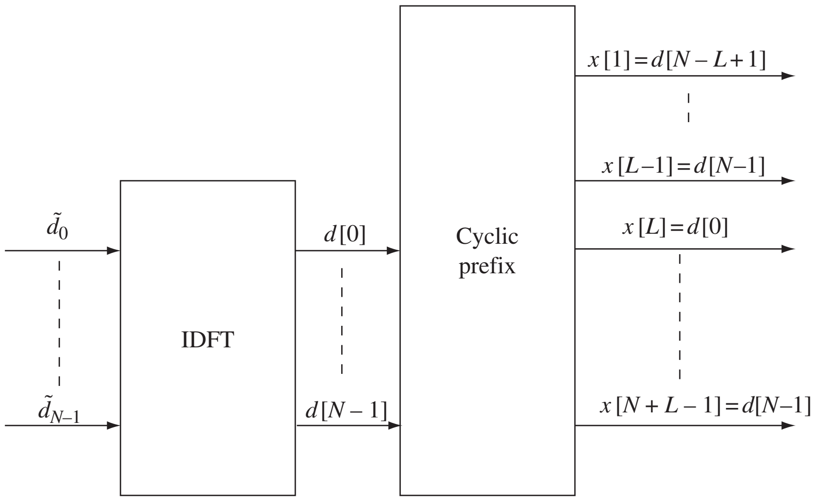

Figure 2: The cyclic prefix operation.

The result that we get from this precoding is the following: the DFT of the received vector $\mathbf{y}$ and the DFT of our initial "data" vector $\mathbf{d}$ have the relationship that the received and transmitted vectors have in an AWGN channel with no ISI (given a suitable definition for the noise vector and it's DFT as given earlier). This seems to suggest that if we put the actual data that we want to transmit on the DFT, and take the DFT of what we receive, then we can perform something similar to traditional AWGN style decoding. Note that this scheme uses $L-1$ extra time instants. This yields the block diagram in Figure (3). A careful and detailed derivation of this step is carried out next. At the end of that calculation, we will have also shown how we arrive at the parallel AWGN channel (cf. Equation (3)).

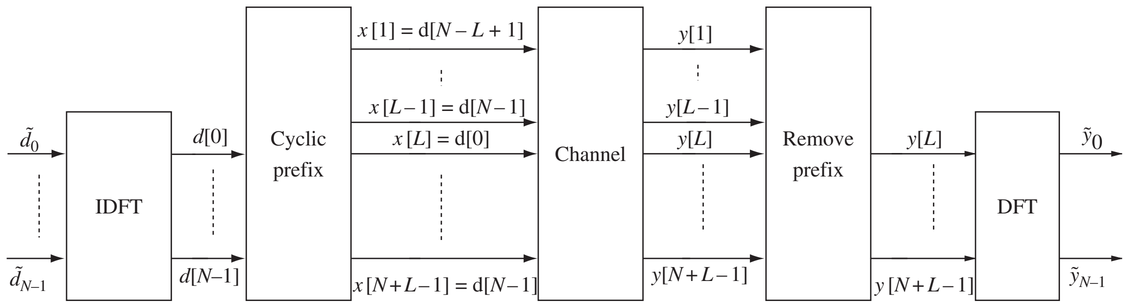

Figure 3: The OFDM transmission and reception schemes.

Packaging the Data

In this section we will see how to connect the AWGN modulation techniques with the OFDM transmission scheme. Suppose we start out with N_c real voltages $\hat{x}[0], \ldots, \hat{x}[N_c-1]$. These are the transmit voltages on the N_c sub-channels using the separate communication architecture (they are generated by efficient coding techniques – such as LDPC codes – for the AWGN channel). Let us suppose that N_c is an even number. This will simplify our notations. We generate *half* of the data vector $\tilde{\mathbf{d}}$ as follows:

$$\Re\left[\tilde{d}_n\right] \stackrel{\text{def}}{=} \hat{x}[2n] \tag{22}$$

$$\Im\left[\tilde{d}_n\right] \stackrel{\text{def}}{=} \hat{x}[2n+1], \quad n = 0, \ldots, \frac{N_c}{2} - 1. \tag{23}$$

The second half is simply conjugate symmetric of the first part (so as to respect Equation (15)):

$$\tilde{d}_n = \tilde{d}^*_{N_c-1-n}, \quad n = \frac{N_c}{2}, \ldots, N_c - 1. \tag{24}$$

Since $\tilde{\mathbf{d}}$ is conjugate symmetric by construction, the inverse discrete Fourier transform (IDFT) vector $\mathbf{d}$ is composed only of real numbers. The cyclic prefix is then added on and the transmit voltages $x[m]$ are generated. Observe that we need an extra $L-1$ time instants to send over the N_c voltages $\hat{x}[0], \ldots, \hat{x}[N_c-1]$.

Unpacking the Data

At the output of the DFT of the received voltage vector $\mathbf{y}$ we have the complex vector $\tilde{\mathbf{y}}$. Taking complex conjugate operation on both sides of Equation (16):

$$\begin{aligned} \tilde{y}^*_n &= \tilde{h}^*_n \tilde{d}^*_n + \tilde{w}^*_n && (25) \\ &= \tilde{h}_{N_c-1-n} \tilde{d}_{N_c-1-n} + \tilde{w}_{N_c-1-n} && (26) \\ &= \tilde{y}_{N_c-1-n}. && (27) \end{aligned}$$

Here we used Equations (15),(21), (20) to verify Equation (19). This means that *half* the DFT outputs are *redundant* and can be discarded. Using the first half, we arrive at the following N_c received voltages $\hat{y}[0], \ldots \hat{y}[N_\mathrm{c}-1]$:

$$\begin{aligned}
\hat{y}[2n] &\stackrel{\mathrm{def}}{=} \Re\left[\frac{\tilde{h}_n^*}{|\tilde{h}_n|}\tilde{y}_n\right] && (28)\\
&= |\tilde{h}_n|\hat{x}[2n] + \hat{w}[2n] && (29)\\
\hat{y}[2n+1] &\stackrel{\mathrm{def}}{=} \Im\left[\frac{\tilde{h}_n^*}{|\tilde{h}_n|}\tilde{y}_n\right] && (30)\\
&= |\tilde{h}_n|\hat{x}[2n+1] + \hat{w}[2n+1], \quad n = 0, \ldots, \frac{N_\mathrm{c}}{2} - 1. && (31)
\end{aligned}$$

Here $\hat{w}[\cdot]$ is also white Gaussian with zero mean and variance σ^2 (explored in a homework exercise). Putting together Equations (29) and (31), we can write

$$\hat{y}[n] = \hat{h}_n \hat{x}[n] + \hat{w}[n], \quad n = 0, \ldots, N_\mathrm{c} - 1 \tag{32}$$

where we have written

$$\hat{h}_n \stackrel{\mathrm{def}}{=} \begin{cases} |\tilde{h}_{\frac{n}{2}}| & n \text{ even} \\ |\tilde{h}_{\frac{n-1}{2}}| & n \text{ odd.} \end{cases} \tag{33}$$

We can repeat the OFDM operation over the next block of N_c symbols (taking up an extra $L-1$ time instants, as before) and since the wireline channel stays the same, we have the end-to-end relation (as in Equation (32)):

$$y[N_\mathrm{c} + n] = \hat{h}_n \hat{x}[N_\mathrm{c} + n] + \hat{w}[N_\mathrm{c} + n], \quad n = 0, \ldots, N_\mathrm{c} - 1. \tag{34}$$

By repeating the OFDM operation over multiple N_c blocks, we have thus created the parallel AWGN channel promised in Equation (3). This packaging and unpacking of data as appended to the basic OFDM scheme (cf. Figure 3) is depicted in Figures 4 and 5.

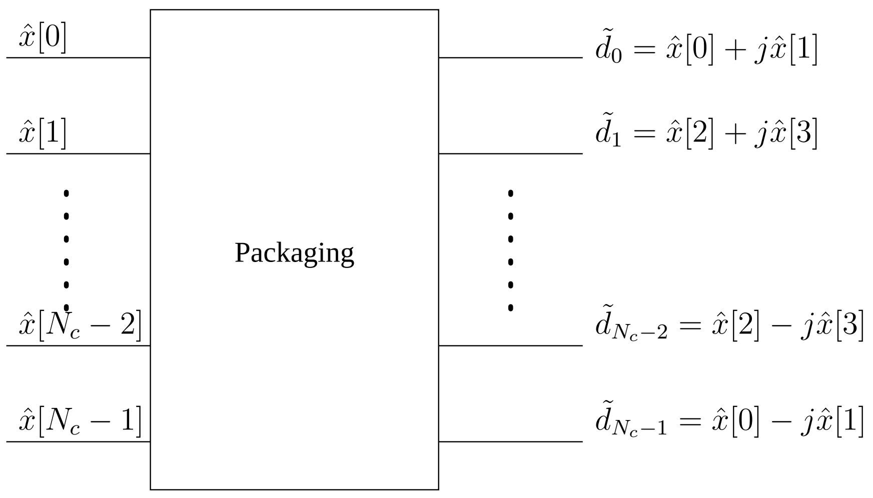
$\hat{x}[0]$
$\hat{x}[1]$
$\hat{x}[N_c-2]$
$\hat{x}[N_c-1]$
Packaging
$\tilde{d}_0 = \hat{x}[0] + j\hat{x}[1]$
$\tilde{d}_1 = \hat{x}[2] + j\hat{x}[3]$
$\tilde{d}_{N_c-2} = \hat{x}[2] - j\hat{x}[3]$
$\tilde{d}_{N_c-1} = \hat{x}[0] - j\hat{x}[1]$

Figure 4: The packaging at the transmitter maps AWGN coded voltages of the sub-channels to the transmit voltages on the ISI channel.

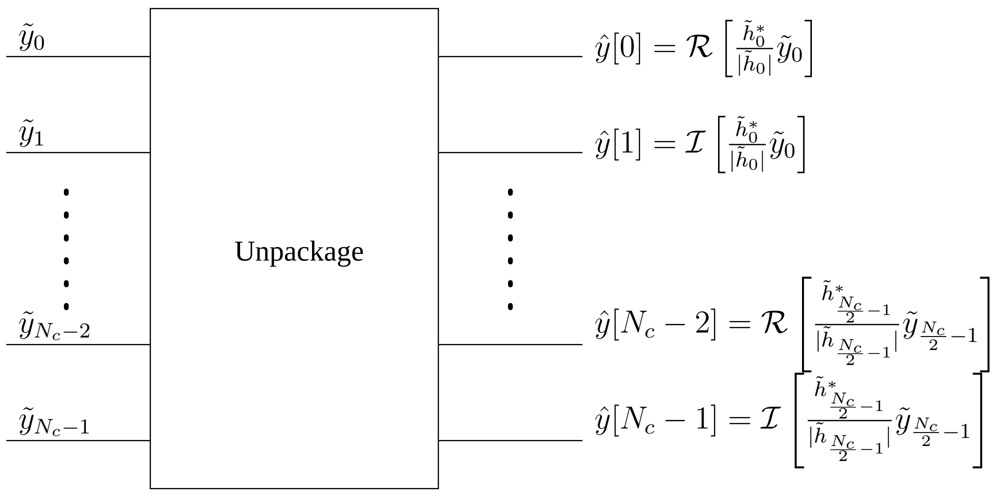
$\tilde{y}_0$
$\tilde{y}_1$
$\tilde{y}_{N_c-2}$
$\tilde{y}_{N_c-1}$
Unpackage
$\hat{y}[0] = \mathcal{R}\left[\frac{\tilde{h}_0^*}{|\tilde{h}_0|}\tilde{y}_0\right]$
$\hat{y}[1] = \mathcal{I}\left[\frac{\tilde{h}_0^*}{|\tilde{h}_0|}\tilde{y}_0\right]$
$\hat{y}[N_c-2] = \mathcal{R}\left[\frac{\tilde{h}^*_{\frac{N_c}{2}-1}}{|\tilde{h}_{\frac{N_c}{2}-1}|}\tilde{y}_{\frac{N_c}{2}-1}\right]$
$\hat{y}[N_c-1] = \mathcal{I}\left[\frac{\tilde{h}^*_{\frac{N_c}{2}-1}}{|\tilde{h}_{\frac{N_c}{2}-1}|}\tilde{y}_{\frac{N_c}{2}-1}\right]$

Figure 5: The unpacking at the receiver maps the DFT outputs into the outputs suitable for decoding the coded voltages of the sub-channels.