1 OFDM Channel Estimation

In the first part of the course, we discussed the principles of OFDM systems. The main result was that an (idealized) OFDM transmission can be written as

$$\mathcal{Y}[n,m] = \mathcal{H}[n,m]\,\mathcal{S}[n,m] + \mathcal{N}[n,m]\,, \tag{1}$$

where $\mathcal{Y}$ and $\mathcal{S}$ are the frequency-domain received and transmitted symbols, $\mathcal{H}$ is the complex channel gain, and $\mathcal{N}$ is AWGN. The indices n and m denote the time and frequency index, respectively.

The result (1) is important because it shows that the OFDM technique effectively transforms an ISI-channel into a channel without intersymbol interference. Equalization in OFDM systems simply means multiplying the received symbols by the inverse channel gain:

$$\hat{\mathcal{S}}[n,m] = \mathcal{Y}[n,m]\,\mathcal{H}[n,m]^{-1}. \tag{2}$$

This is also called a *one-tap equalizer*, because the multiplication with the single complex number $\mathcal{H}[n,m]^{-1}$ can be seen as an FIR filter with only one tap.

Since the channel gain is a random process in wireless communication systems, the receiver does not know the true value $\mathcal{H}[n,m]$. Instead, it has to estimate the channel based on the received signal, which leads us to the topic of this assignment. For simplicity, we assume the channel to be constant during the transmission, which means that the channel transfer function reduces to $\mathscr{h}[m]$. In practice, however, the receiver must usually also track the channel variations over time.

2 Channel Estimation Sequence and the Least Squares Estimator

Since we want to focus on the basic concepts of OFDM transmission, we will only consider *training aided* channel estimation in this lab. In our system, the channel estimation is based on one OFDM symbol, the *channel estimation sequence*, which is completely known to the receiver. It is located between the frame synchronization sequence and the data carrying OFDM symbols, as shown in Figure 1.

Since throughout this chapter we are only concerned with the channel estimation, we drop the symbol index n and write the received channel estimation sequence (after CP removal and FFT) as

$$\mathscr{y}[m] = \mathscr{h}[m]\,\mathscr{s}[m] + \mathscr{n}[m]\,, \tag{3}$$

Figure 1: Frame Structure including the Channel Estimation Sequence

where $s[m]$ are QPSK symbols that are known to the receiver. (In the lab, you will get a file `channel_estimation_sequence.mat` that contains these symbols.)
How can we get an estimate of $h[m]$ from $y[m]$? An obvious approach is to simply divide the received symbols by the known transmitted symbols. Since the symbols $s[m]$ belong to a QPSK constellation ($|s[m]| = 1$), their inverses are equal to their complex conjugates, $s^{-1}[m] = s^*[m]$, and the channel estimate is given as

$$\begin{aligned} \hat{h}[m] &= y[m]\, s^*[m] \\ &= h[m] + n[m]\, s^*[m] \,. \end{aligned} \tag{4}$$

Since $n[m]$ are zero-mean noise samples, the products $n[m]\, s^*[m]$ are still zero-mean, and (4) is an unbiased estimator of the channel transfer function,

$$E\{\hat{h}[m]\} = h[m] \,. \tag{5}$$

In the Appendix, we show that (4) is actually the so-called *least squares* (LS) channel estimator, because it fulfills the least squares optimality criterion.

3 Channel Correlation and the MMSE Estimator

We have found out that (4), in the following denoted as $\hat{h}_{\mathrm{LS}}[m]$, is an unbiased channel estimator. Can we get any better than that?
Actually, we can. Up to now, we have not exploited the correlation between the channel gains $h[m]$. However, we know that OFDM systems are designed such that the channel is approximately constant within each subband, and therefore we can assume that the channel gains of two adjacent subbands are quite similar, i.e.,

$$h[m+k] \approx h[m] \,, \tag{6}$$

where k is a small number. This is illustrated in Figure 2, which shows the transfer functions (in dB) of two exemplary channels. One channel has a delay spread (the length of the impulse response) of $T_{\mathrm{d}} = 2\,\mu\mathrm{s}$, and the other one of $T_{\mathrm{d}} = 10\,\mu\mathrm{s}$. As we can see, both functions change smoothly over the frequency, and the one with the longer delay spread changes more rapidly than the other one.
For the following discussion, it is instructive to introduce the so-called *coherence bandwidth* W_{c}, which is a measure for the channel variability over the frequency. It denotes the maximum bandwidth over which two frequencies of a signal are likely to experience similar fading. The coherence bandwidth should be understood in an order-of-magnitude-sense, and not as a

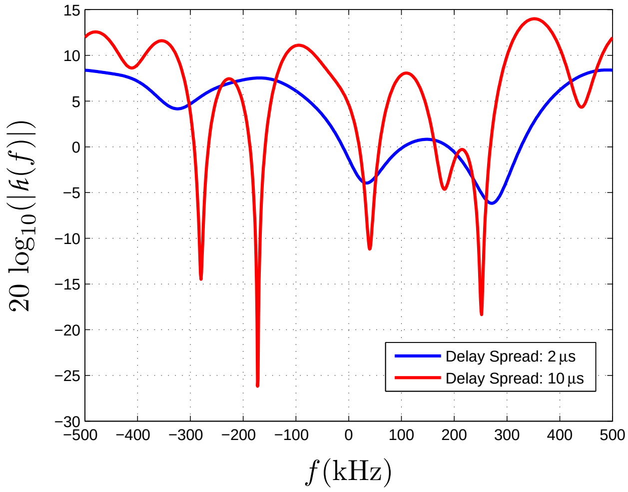

Figure 2: Typical transfer functions of bandlimited channels; $B = 1\,\mathrm{MHz}$

precise physical quantity. One way to define it is

$$W_{\mathrm{c}} = \frac{1}{2T_{\mathrm{d}}}\,. \tag{7}$$

For example, the channels from Figure 2 have coherence bandwidths of $W_{\mathrm{c}} = 250\,\mathrm{kHz}$ and $W_{\mathrm{c}} = 50\,\mathrm{kHz}$. The AWGN channel is an extreme case with $T_{\mathrm{d}} = 0$, and thus $W_{\mathrm{c}} = \infty$, which means that the channel is constant over the whole spectrum. By exploiting the a-priori knowledge that we have about the channel correlation, we can improve the LS estimates. This operation is also called *smoothing*.

The *minimum mean square error* (MMSE) estimator is a standard estimator that takes such knowledge into account. The derivation of the MMSE estimator is also given in the Appendix. In the following, we summarize the results. The vector of MMSE estimates is given as

$$\hat{\mathcal{h}}_{\mathrm{MMSE}} = \mathbf{Q}_{\mathrm{MMSE}}\hat{\mathcal{h}}_{\mathrm{LS}} \tag{8}$$

with the matrix

$$\mathbf{Q}_{\mathrm{MMSE}} = \mathbf{C}_{\mathcal{h}}(\mathbf{C}_{\mathcal{h}} + \sigma_n^2\mathbf{I})^{-1}\,. \tag{9}$$

Here, finally, the matrix $\mathbf{C}_{\mathcal{h}}$ is the channel correlation matrix and σ_n^2 is the noise variance (in the lab, this is the inverse SNR, because the signal power is normalized to one).

Under the assumption of an exponentially decaying channel impulse response with time constant τ, the entries of $\mathbf{C}_{\mathcal{h}}$ are

$$\mathbf{C}_{\mathcal{h}}[m_1, m_2] = \frac{1}{1 + j2\pi\tau F(m_1 - m_2)}\,, \tag{10}$$

where F is the width of the subbands. A detailed derivation of (10) is also given in the Appendix. Combining (9) and (10), we see that the MMSE channel estimator depends on two values: The time constant of the exponentially decaying channel impulse response, and the SNR. In practical systems, these values are not known a-priori and therefore must also be estimated. However, we will not go into further detail here. The MMSE estimator yields very good results, but it has a major drawback: its computational complexity. The calculation

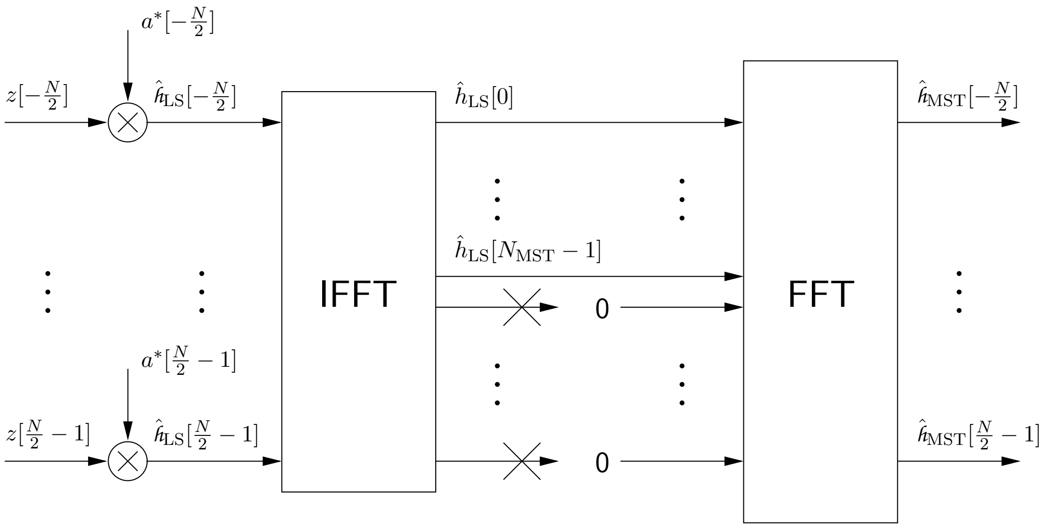

Figure 3: Block diagram of the Most Significant Taps algorithm

(8) involves a matrix-vector multiplication, whose dimension can be larger than 1024 in practice, and the calculation (9) of the MMSE estimator matrix even involves a matrix inversion of the same dimension. Therefore, practical receivers use suboptimal estimators with lower computational complexity.

A large variety of proposals for channel estimators can be found in the literature. In the next section, we will present one example of a low-complexity estimator.

4 Smoothing in the Time Domain: The MST Approach

The idea of the estimator that we present in this section is to improve the LS estimates in the time domain instead of the frequency domain. The first step is thus to calculate the LS estimate of the channel impulse response:

$$\hat{h}_{\mathrm{LS}} = \mathrm{IFFT}\{\hat{\hbar}_{\mathrm{LS}}\}\,. \tag{11}$$

By design of an OFDM system, the channel impulse response is considerably shorter than the OFDM symbol duration. Therefore, the energy of the channel is concentrated in the first N_{MST} taps of $\hat{h}_{\mathrm{LS}}$, with $N_{\mathrm{MST}} \ll N$, while the rest of the LS CIR estimate consists mainly of noise. The first channel taps are therefore called the *most significant taps* (MST). A straightforward way to improve the channel estimates is now to set the non-MST taps, which consist only of noise, to zero. The resulting CIR estimate is then converted back into the frequency domain, yielding an improved estimate of the channel transfer function. Figure 3 illustrates this algorithm.

The MST algorithm can be seen as lowpass filtering of the LS channel estimates, with exchanged roles of the time and frequency domain: Setting the non-MST taps to zero is effectively a multiplication of the CIR estimate with a rectangular window in the time domain. In the frequency domain, this multiplication corresponds to an ideal lowpass filter. The MST estimator basically consists of two concatenated FFT operations and is therefore of significantly lower computational complexity than the MMSE estimator. The remaining question is how N_{MST}

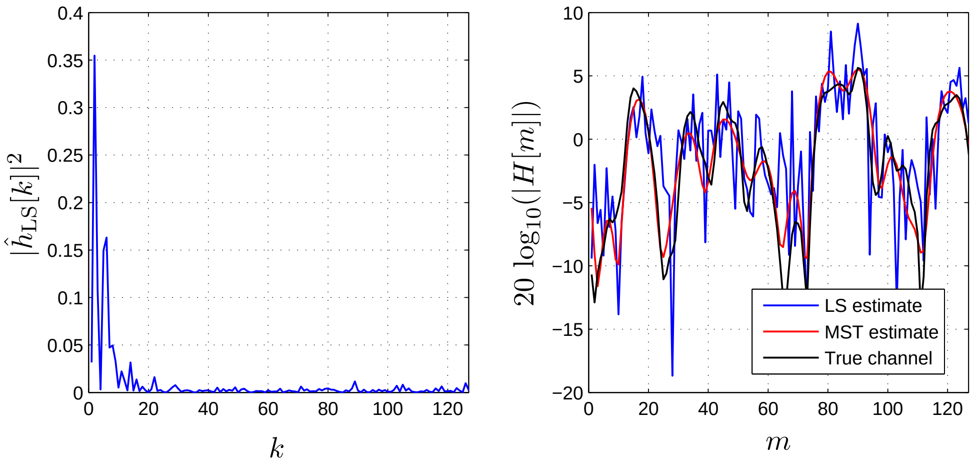

Figure 4: Left: LS estimate of an examplary CIR. The MST algorithm chooses $N_{\mathrm{MST}} = 16$. Right: Comparison of LS estimate, MST estimate and true channel in the frequency domain.

should be chosen. If it is chosen too small, some taps that belong to the channel impulse response are deleted, which would cause a severe degradation of the channel estimation. On the other hand, if N_{MST} is chosen too large, some noise-only taps are left over and less noise is removed from the channel estimate. A simple option is to select $N_{\mathrm{MST}} = N_{\mathrm{cp}}$, because the length of the cyclic prefix should, by design of the OFDM system, be at least as long as the channel impulse response.

5 Your Tasks

You are given an implementation of an OFDM receiver with the channel estimation part missing. You will get signals that include a channel estimation sequence as in Figure 1, and that are affected by multipath channels.

1. Expand the OFDM receiver with a channel estimator based on the channel estimation sequence. Implement the LS, the MMSE, and the MST estimation algorithms in the `ofdm_channel_estimator` function. The width of the subbands in our system is $F = 50\,\mathrm{kHz}$, and the time constant of the CIR is approximately $\tau = 0.7\,\mu\mathrm{s}$.

2. Plot the estimated channel transfer functions obtained by the three algorithms into one figure and compare them at different SNRs.

3. Implement the one-tap equalizer according to (2) in the `ofdm_demodulator` function.

4. Display the received image at various SNRs. Can you explain the pattern of the distortion? Can you think of a way to make the pattern less apparent?

5. Use the PN-sequence to compare the BER performance of the three channel estimation algorithms over a reasonable range of SNRs.

Algorithms

OFDM Channel Estimation

In this section, we derive the *least squares* (LS) and the *minimum mean square error* (MMSE) channel estimators. The estimation is based on the signal model

$$y[m] = s[m]\, h[m] + n[m]\,, \tag{12}$$

where $y[m]$ are the observations, $s[m]$ are known QPSK symbols, $h[m]$ are the channel gains that we want to estimate, and $n[m]$ is AWGN with variance σ_n^2.
For the following derivation, it is convenient to write (12) in matrix-vector notation:

$$y = \mathcal{S} h + n\,, \tag{13}$$

where $\mathcal{S} = \mathrm{diag}(s[-\frac{N}{2}], \dots, s[\frac{N}{2} - 1])$.

LS Estimator

The least squares estimator yields the estimate $\hat{h}_{\mathrm{LS}}$ that minimizes the metric $\|y - \mathbf{AH}\|^2$, where $\|\cdot\|$ is the Euclidean vector norm. In other words,

$$\begin{aligned}
\hat{h}_{\mathrm{LS}} &= \arg\min_{h} \|y - \mathcal{S}h\|^2 \\
&= \arg\min_{h} (y - \mathcal{S}h)^H (y - \mathcal{S}h) \\
&= \arg\min_{h} (y^H y - h^H \mathcal{S}^H y - y^H \mathcal{S} h + h^H \mathcal{S}^H \mathcal{S} h) \\
&= \arg\min_{h} (y^H y - h^H \mathcal{S}^H y - y^H \mathcal{S} h + h^H h)\,,
\end{aligned} \tag{14}$$

where in the last equation we have used that $\mathcal{S}^H \mathcal{S} = \mathbf{I}$ (because $|a[m]| = 1$). In order to find the vector h that minimizes this metric, we set its partial derivative to zero:

$$\begin{aligned}
\mathbf{0} &\stackrel{!}{=} \frac{\partial}{\partial h^H} (y^H y - h^H \mathcal{S}^H y - y^H \mathcal{S} h + h^H h) \\
&= -\mathcal{S}^H y + h\,.
\end{aligned} \tag{15}$$

The channel vector that minimizes (15) is simply

$$\hat{h}_{\mathrm{LS}} = \mathcal{S}^H y\,, \tag{16}$$

or, going back to sample notation,

$$\hat{h}_{\mathrm{LS}}[m] = z[m]\, a^*[m]\,. \tag{17}$$

MMSE Estimation

The MMSE estimator improves the LS estimates by also taking the correlation between adjacent channel gains into account.
With (13) and (16), the LS estimates can be written as

$$\begin{aligned}
\hat{h}_{\mathrm{LS}} &= h + \mathcal{S}^H n \\
&= h + n'\,.
\end{aligned} \tag{18}$$

Since $\mathcal{S}$ is a diagonal matrix of QPSK symbols, n' is an AWGN vector with the same statistics as n.
Since we have a linear system distorted only by Gaussian noise, the MMSE estimator is equivalent to the *linear MMSE* estimator, which is given as

$$\hat{\mathit{h}}_{\mathrm{MMSE}} = \mathbf{Q}_{\mathrm{MMSE}} \hat{\mathit{h}}_{\mathrm{LS}} \,. \tag{19}$$

We determine the matrix $\mathbf{Q}_{\mathrm{MMSE}}$ with the *orthogonality principle*, which says that the error vector $\hat{\mathit{h}}_{\mathrm{MMSE}} - \mathit{h}$ must be orthogonal to the data $\hat{\mathit{h}}_{\mathrm{LS}}$:

$$E\left\{(\mathbf{Q}_{\mathrm{MMSE}} \hat{\mathit{h}}_{\mathrm{LS}} - \mathit{h})\hat{\mathit{h}}_{\mathrm{LS}}^H\right\} = \mathbf{0} \tag{20}$$

$$E\left\{\mathbf{Q}_{\mathrm{MMSE}} \hat{\mathit{h}}_{\mathrm{LS}}\hat{\mathit{h}}_{\mathrm{LS}}^H\right\} = E\left\{\mathit{h}\hat{\mathit{h}}_{\mathrm{LS}}^H\right\} \tag{21}$$

$$\mathbf{Q}_{\mathrm{MMSE}} E\left\{(\mathit{h}+\mathit{n}')(\mathit{h}+\mathit{n}')^H\right\} = E\left\{\mathit{h}(\mathit{h}+\mathit{n}')^H\right\} \tag{22}$$

$$\mathbf{Q}_{\mathrm{MMSE}}(\mathbf{C}_{\mathit{h}} + \sigma_W^2 \mathbf{I}) = \mathbf{C}_{\mathit{h}} \tag{23}$$

and finally

$$\mathbf{Q}_{\mathrm{MMSE}} = \mathbf{C}_{\mathit{h}}(\mathbf{C}_{\mathit{h}} + \sigma_{\mathit{n}}^2 \mathbf{I})^{-1} \,. \tag{24}$$

Here, $\mathbf{C}_H = E\{\mathit{h}\mathit{h}^H\}$ is the channel covariance matrix. The entries of $\mathbf{C}_{\mathit{h}}$ are determined by the spaced-frequency correlation function:

$$R_{\mathit{h}}(\Delta f) = E\{\mathit{h}(f+\Delta f)\mathit{h}^*(f)\}\,, \tag{25}$$

evaluated at integer multiples of the subcarrier spacing F. The correlation function (25) is the Fourier transform of the so-called *power delay profile*:

$$R_h(\tau) = E\{|h(\tau)|^2\}\,. \tag{26}$$

The power delay profile is usually unknown, but in many scenarios, it can be well approximated by an exponential function

$$R_h(\tau) \approx \frac{1}{\tau_{\mathrm{rms}}}\, e^{-\frac{\tau}{\tau_{\mathrm{rms}}}}\,, \tag{27}$$

where τ_{rms} is the *root mean square* (RMS) delay spread of the channel. Notice the normalization

$$\int_0^\infty R_h(\tau)d\tau = 1\,. \tag{28}$$

Taking the Fourier transform of (27), we find the spaced-frequency correlation function as

$$R_{\mathit{h}}(\Delta f) = \frac{1}{1 + j2\pi\tau_{\mathrm{rms}}\Delta f}\,. \tag{29}$$

Since the difference of the center frequencies of two subbands m_1 and m_2 is $\Delta f = F(m_1 - m_2)$, the entries of the channel correlation matrix are finally given as

$$\mathbf{C}_{\mathit{h}}[m_1, m_2] = \frac{1}{1 + j2\pi\tau_{\mathrm{rms}}F(m_1 - m_2)}\,. \tag{30}$$