ALaRI: Digital Signal Processing

Exercise Session 4: Orthogonal Frequency Division Multiplexing (OFDM) modulation

Deepaknath Tandur, Paschalis Tsiaflakis, and Toon van Waterschoot[1]

May 2013

Goal: to create m-files for modulating and demodulating a binary signal using the OFDM technique, and for evaluating the transmission performance in an ideal channel.

Requirements: Matlab in Windows or Linux, 4 m-files (qam_experiment.m, qam_mod.m, qam_demod.m, ber.m).

Outcome: 3 m-files (ofdm_experiment.m, ofdm_mod.m, ofdm_demod.m).

Exercise 4-1: OFDM – modulation, transmission, and demodulation

Create an m-file "ofdm_experiment.m" in which the following five steps are implemented (and try to reuse as much as possible the code you have produced in Exercise 3-1 to obtain the Matlab script qam_experiment.m):

1. Generate an M-ary QAM frame OFDM packet of length N_l as shown in Fig. 1. Each frame consists of N complex numbers, of which two are equal to zero, and the other ones are equal to the successive QAM symbols taken from a QAM symbol sequence (which is generated, e.g., in the function qam_mod.m that you created in Exercise 3-1), or to the complex conjugate (*) of these QAM symbols. See Fig. 1 for the exact structure of one frame.

2. Create a Matlab function "ofdm_mod.m" that simulates an OFDM-based transmitter. In the transmitter an IFFT operation is applied to each of the M-ary QAM data frames and then each resulting frame is parallel-to-serial converted.

3. Create a Matlab function "ofdm_demod.m" that simulates an OFDM-based receiver. In the receiver, a serial-to-parallel conversion is performed, followed by an FFT operation, to regenerate the transmitted QAM frames and symbols.

4. Test your OFDM-based transmitter and receiver design by simulating a transmission over an ideal channel (i.e., $H(z) = 1$ and SNR $= \infty$ dB). Use the ber.m function implemented in Exercise 3-1 to check if your design is correct.

[1]For questions and remarks: toon.vanwaterschoot@esat.kuleuven.be

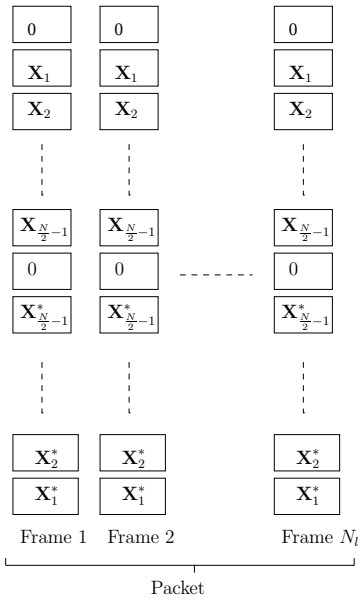

Figure 1: M-ary QAM OFDM packet

5. Extend the transmitter and receiver functions ofdm_mod.m and ofdm_demod.m with Matlab code that serves to add and remove a cyclic prefix to every OFDM data frame. The cyclic prefix length is user-defined, and is usually much shorter than the actual frame length. Again check the BER to see of the design is correct.