ELEG5663 Project 3
Zero Forcing Equalization

I. Objectives

1. Learn to perform simulation of bandpass digital communication systems.
2. Understand the concept of equalization.
3. Learn to design a zero forcing (ZF) equalizer.
4. Learn to implement ZF equalizer.

II. Procedures

A. ZF Equalizer Training

1. Pass a unit impulse through the channel. The received signal after matched filter and sampler is h = [0.0094, 0.0188, -0.0282, 0.0940, 0.9396, 0.1879, -0.0940, -0.0470, 0.0188], where h(0) = 0.9396. Design a 5-order ZF equalizer. Find the coefficients of the equalizer, and list your results in a table (show procedure).
2. Repeat the above step for a 3-order ZF equalizer.

B. ZF equalization for BPSK system in ISI channel

3. The simulation of BPSK can be performed with orthogonal signal representation. Since baseband bipolar signal has similar orthogonal signal representation with BPSK, the modulation and demodulation of BPSK system simulation is exactly the same as the simulation of bipolar baseband system in Project 2. With the constellation diagram shown in Fig. 1 and the procedure in Project 2, generate one slot (200) BPSK modulated symbols. Let's denote the BPSK symbol vector as x.

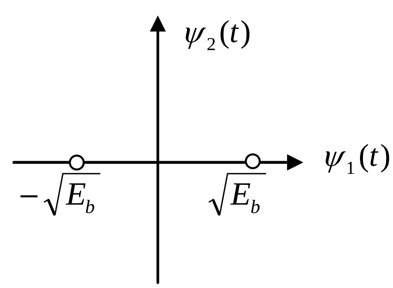

Fig. 1, Orthogonal representation of bipolar signaling

4. Pass the BPSK symbol through the ISI channel. The channel is a LTI system with input being the BPSK symbols. Therefore, the output of the ISI channel is the convolution between the BPSK symbol (the x vector in step 3) and the channel impulse response (the h vector in step 1). The convolution can be performed by using the Matlab command 'conv', e.g. data_isi = conv(x, h);
5. Add AWGN to the symbols distorted by ISI.
6. Pass the received signal vector through the 5-tap ZF equalizer. The operation can be performed with convolution since the ZF equalizer is a LTI system with impulse response being the ZF coefficients. Let's denote the vector after the ZF equalizer as y.

7. The convolution operations during ISI channel and ZF equalization add additional symbols to the slot. If the length of vector x is n1, and the length of vector y is n2, the length of conv(x, y) is n1 + n2 – 1. What should be the length of the vector after ZF equalization? (Note two convolutions are performed). Verify your results by using the length( ) command.
8. Due to convolution, the output of the equalization has more than 200 symbols. Since we have 200 data symbols, we only need the 200 symbols in the middle of y. Discard the redundant symbols at the beginning and the tail of y such the 200 symbols in the middle of y are extracted. Let's denote the vector as r.
9. Perform BPSK demodulation over r, and compare your results with the original data. Test your equalizer at high Eb/N0, e.g. Eb/N0 = 20 dB. You should have almost zero error at high SNR.
10. Find the BER at Eb/N0 = [0, 2, 4, 6, 8, 10] dB. Use 1000 slots for each value of Eb/N0.
11. Plot the BER curve.

12. Repeat the above procedures for the 3-tap equalizer. Plot the BER curve at the same figure.

13. Plot the theoretical BER curve of BPSK in the same curve. Compare the three BER curves.