Introduction

In the project, it is expected to examine the causal ISI channel with a Gaussian noise. So, the intersymbol interference occurs through the channel and also there exists an AWGN noise ($n_k = N(0, \frac{N_0}{2})$). So the received signal y_k is given in the equation 1.

$$y_k = h_k * x_k + n_k = \sum_{n=0}^{L} h_l x_{k-l} + n_k \tag{1}$$

h_l is the causal ISI channel impulse response which is given as

$$h_l|_{l=-0}^{4} = [0.74 \quad -0.514 \quad 0.37 \quad 0.216 \quad 0.062]$$

, x_k is the binary data symbol at time k ($x_k \in \{-1, 1\}$), and n_k is the noise.

The following two approaches, which deal with the ISI issue, Maximum Likelihood Sequence Estimation (MLSE) with Viterbi, and Frequency Domain Equalization (FDE) are selected to implement.

1 Question 1: MLSE with Viterbi

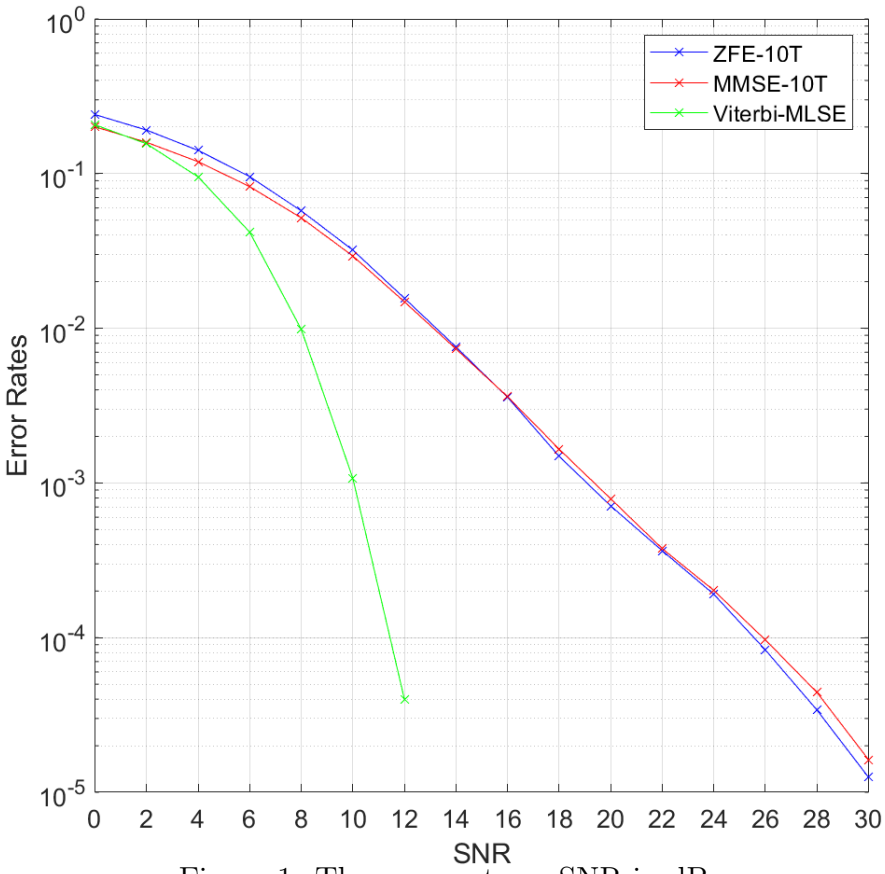

Figure 1: The error rate vs SNR in dB

The MLSE with Viterbi algorithm is implemented as following: Firstly, in order to initialize the process, the estimation of the first 5 received bits are determined by trying all the combinations, 2^5 many, of the bit sequences. 2 candidates of the estimated sequences are decided as the 2 closest sequences which are gone through the channel and compared with the first 5 received bits. After estimating the first 5 bits, the next bit is decided by considering the following 4 possibilities: -1 or 1 can be the next bit of the first candidate, or -1 or 1 can be

the next bit of the second candidate. For each candidate, the one that is close to the received bit are appended to its candidate sequence. After estimating all the received bits, the candidate which is closer to the received sequence is selected as the detected sequence.
In order to compare the performance of the MLSE with Viterbi algorithm with the ZFE and MMSE with 10 taps, they are implemented from the homework 5.

The Figure 1 shows the plot of the bit error rate of the equalizers as a function of signal-to-noise ratio. It can be seen from the figure that the BER performance of the MLSE is better than the other two, as expected, since the MLSE with Viterbi algorihm attemps to detect the bits by considering all the promising combinations of the bit sequences. However, the speed of the MLSE algorithm is slower than the other two due to the consideration of all the promising combinations. And the BER performance of the MMSE is better than the ZFE in low SNR values since it takes into account the noise, but in the high SNR values, their performances get closer to each other.

2 Question 2: FDE

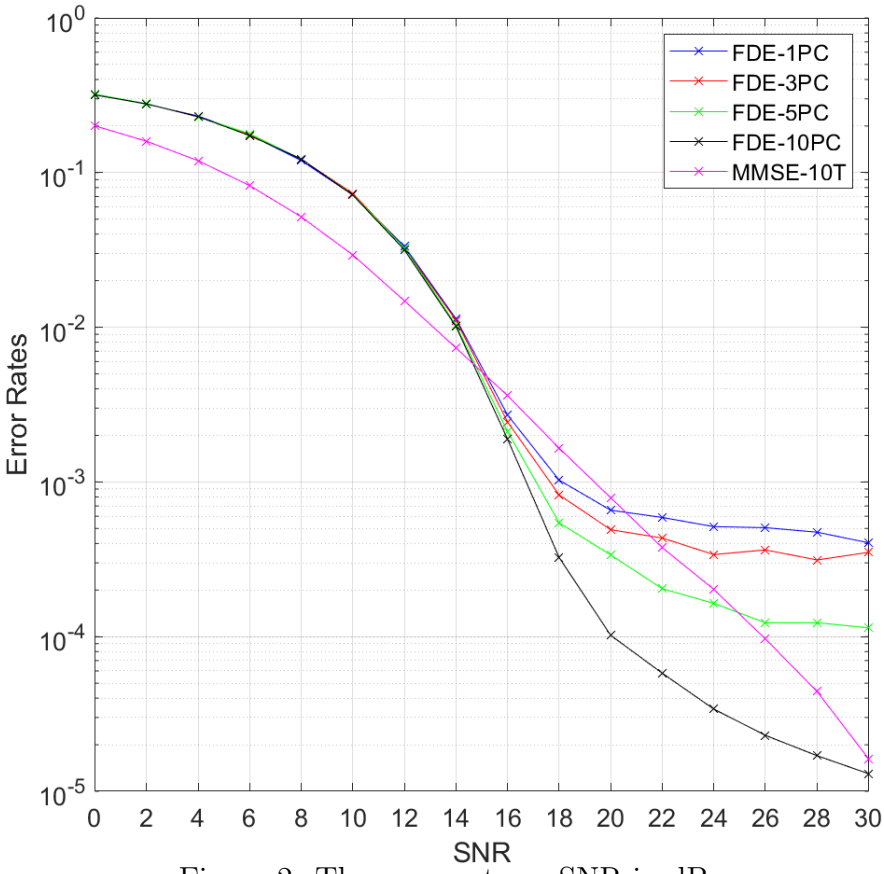

Figure 2: The error rate vs SNR in dB

The FDE is implemented as following: Firstly, the send sequences should have cyclic prefixes (CP) in order to obtain the linear convolution being equal to the circular convolution. Thus, the circular convolution can be represented in the frequency domain as multiplication using Discrete Fourier Transform (DFT), as shown

$$Y[k] = H[k]X[k] + N[k]$$

, where they are the $L+CP$-point DFT of the time domain sequences. The channel coefficients, h_l, are completed with zeros up to $L+CP$ in order to obtain the same length of DFTs as remaining sequences. After transformation to the frequency domain, the estimates of the received se-

quence can be found by taking inverse DFT of $\hat{X}[k] = Y[k]/H[k]$. So, the estimates of the $\hat{x}[n] = IDFT\{X[k]\}$ are given to the regular detector which is used in ZFE and MMSE.

In order to compare the performance of the FDE with the MMSE with 10 taps, it is implemented from the homework 5.

The Figure 2 shows the plot of the bit error rate of the equalizers as a function of signal-to-noise ratio. It can be seen from the figure that the BER performance of the MMSE is better than the FDE in low SNR values yet the performance of the FDE in high SNR values is better than the MMSE. The FDE has low complexity in comparison to the MMSE since in MMSE algorithm, there exists matrix inversion to obtain equalizer coefficients and convolution to estimate the received sequences, and also the SNR value should be known; however, in FDE algorithm, there exist DFT, division and IDFT to estimate the received sequences. And with advantages of the FFT algorithm, the DFT calculations can be calculated very efficiently.
It can also be seen from the figure that at high SNR values, as the number of cyclic prefixes increases, the BER performance of the equalizer increases, since if the number of cyclic prefixes is not sufficient, the linear convolution will not be equivalent to the circular convolution. So, increasing the number of cyclic prefixes makes the linear convolution closer to the circular convolution.