ENEE630 DSP Project 1

Young-Shiuan Hsu, Henok Gebeyehu

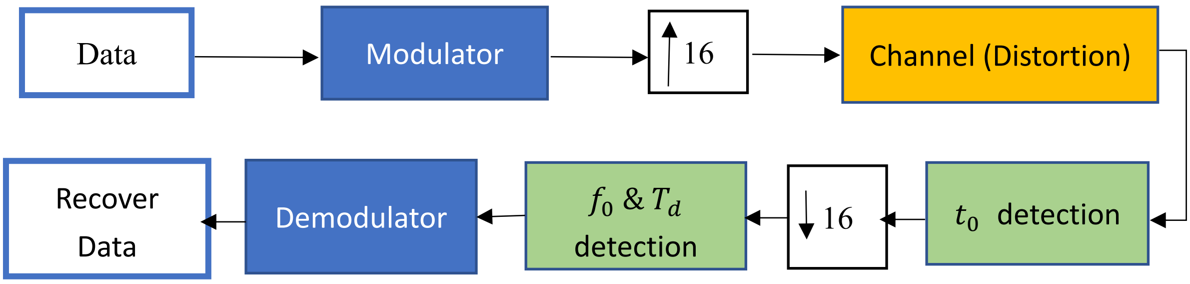
1. Introduction

The objective of the project is to simulate the process of transmitting and receiving data (in the form of a packet). The project consists of three parts: a transmitter, channel and receiver. By adding noise, frequency distortion, time delay and phase distortion, we can analyze the relationship between the transmitted data and the received data.

(Notes on how to run code can be found in the readme)

2. Methodology

I. System



II. Data (Packet) Design

We specifically designed the data so that the receiver could identify the time delay and frequency distortion. By detecting the time delay, the receiver could start down sampling at the precise position that the data is located. By analyzing the frequency distortion, we could demodulate the data to minimize the distortion and decrease our bit error rate.

The first 128 bits of the packet (piolet) is the cosine wave form. We set it all to ones (frequency 0) which means that the peak of its DTFT will be at zero. The next 8 bits is the key, which we set to all zeros. The remaining 664 bits is the data that we wish to transmit.

III. BPSK Modulation

BPSK is a modulation technique that maps all zeros to negative ones and ones remain as ones. This helps the receiver to differentiate between zeros and ones clearly since the 0-mapped negative ones have a phase of -π and the ones have a phase of 0. In this case we also rotate the data ($\pi/4\ BPSK$) which could also protect the data from high SNR.

$$X(k) = e^{j\pi k/4}\,(1 - 2 * Data[k])$$

IV. Channel

The channel is added to all the uncertainties that could cause distortion and the designed receiver should be able to detect the frequency distortion and time delay. The frequency distortion is -1500Hz ~ +1500Hz and the time delay range is -2.5ms ~ +2.5ms (corresponding to -40bits ~ +40bits of the received data). The formula of the designed channel is as follow

$$R(k) = e^{i\varphi_0}\, e^{j2\pi f_0 k/f_s}\, S(k - k_0) + n(k)$$

Where $n(k)$ is the added AWGN noise, φ_0 is the phase error, f_0 and k_0 is the frequency distortion and time delay (generated randomly). We have omitted the random phase error from our experiments by setting φ_0 to 0.

V. Receiver

As the receiver receives the data, it is essential to identify the position from which we begin down sampling. This is because down sampling at an arbitrary starting position may result in loss of data due to the unknown time shift of the received data. After the received data has been down sampled, we have to identify the time delay in order to extract the transmitted data (800 bits). When we find the true data that was transmitted, we still have to compensate for the frequency distortion added in the channel so that we can minimize the error when we demodulate.

We first have to find a precise position (denoted as t_0) to begin down sampling the data since the data was up sampled 16 times in the transmitter. To find t_0, we have to take every sixteen samples of the received data and calculate their DFT. The one that has the maximum value of the DFT (not the power of the data's DFT) is the position where we want to start down sampling because the cosine in the data contains a peak. After calculating the maximum value of the first 15 bits, we are able to find the perfect position to down sample the data.

However, this t_0 that we compute is not the actual time delay of the received data. The time delay is what we have to find after down sampling. Since we know that the data must begin somewhere in the first 80 bits (time delay is -2.5ms ~ +2.5ms), we analyze frames of size 128 bits, shifting the frame one bit at a time, and compute the DFT at each shift of the frame to find the maximum value. Therefore, within the 80 iterations, we are able to find time delay (denoted as T_d).

Last but not least, we would have to find the frequency distortion (denoted as f_0). We will only have to take the DFT of the cosine wave form of the data and check how much the peak is shifted. Each interval represents $\frac{16000}{128} = 125\ hz$.

VI. Demodulator

The demodulator is similar to modulator but instead rotate it in a different direction. After we extract the data (800 bits) from the delayed and frequency distorted signal, we multiply it with the frequency distortion and demodulator.

$$\widetilde{Data}\,(k) = \ e^{(-2\pi\tilde{f}k/f_s)}\ T\big(k - \tilde{k}\ \big)\, e^{-j\pi k/4}$$

$\widetilde{Data}\,(k)$ is the demodulated signal, $\tilde{f}$ is the frequency distortion, f_s is the sampling rate and $\tilde{k}$ is the time delay.

3. Simulation

I. Condition 1

SNR	Time Offset	Frequency Offset	Simulations
100 dB	-2.5ms ~ +2.5ms	0 Hz	100

With every step size is 0.0625ms, the mean of the time delay is 2.46875ms (39.5 samples) and standard deviation is 23.0922. Since we have a clean channel, our system performs perfect reconstruction (BER=0, FER=0), therefore, we are also able to perfectly estimate the value of time delay. To verify our mean of time delay, as time offset = -2.5ms, the time delay returns 0, which is the number of samples of delays and if time offset = 2.5ms, the time delay returns 79 samples of delay. Every simulation has the same result, hence, we can conclude that the mean of the time delay is 39.5 samples, corresponding to 2.46875 ms.

II. Condition 2

SNR	Time Offset	Frequency Offset	Simulations
100 dB	0	-1500Hz ~ +1500Hz	100

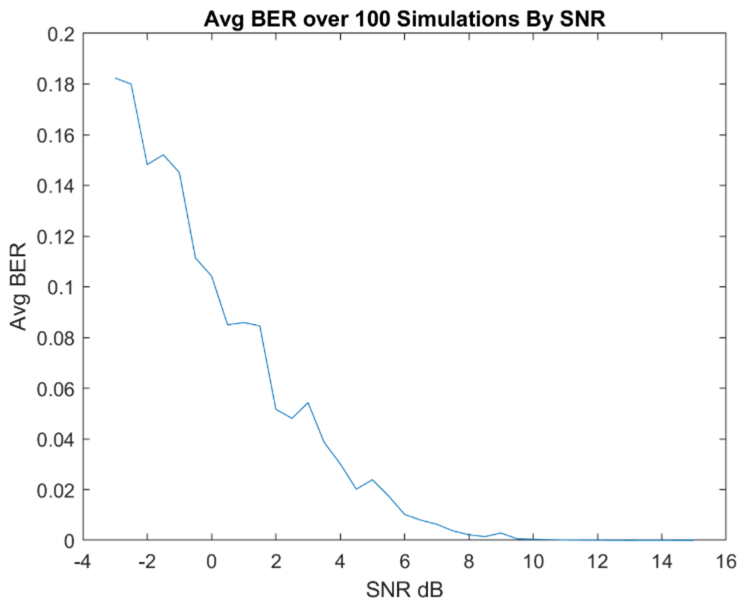
With step size 125Hz, the mean of the estimated frequency distortion is 0Hz and standard deviation is 209.3878Hz. Since we have a clean channel, our system performs perfect reconstruction (BER=0, FER=0), therefore, we are also able to perfectly estimate the value of frequency distortion.

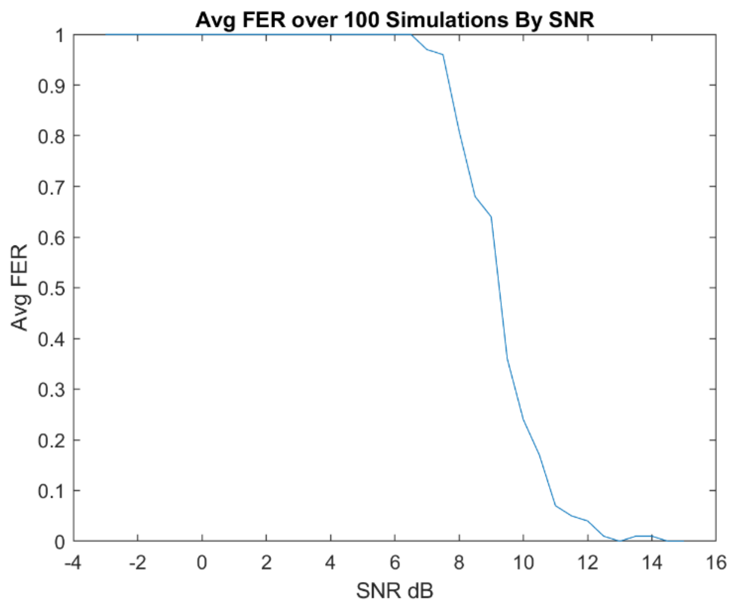
III. Condition 3

SNR	Time Offset	Frequency Offset	Step Size
-3dB ~ 15dB	0	0 Hz	0.5dB

Mean of Time Delay(samples)	STD of Time Delay	Mean of Frequency distortion	STD of Frequency Distortion
39.8530	0.8144	0	0

With only noise interference, no time and frequency shift. The bit error rate and frame error rate is the following figure. When SNR is low, which means that it is extremely hard for the receiver to sperate the signal and the noise, therefore, the bit error rate would be high. On the other hand, as SNR increase, receiver will approach perfect reconstruction, which is obvious to observe in the figure of frame error rate.

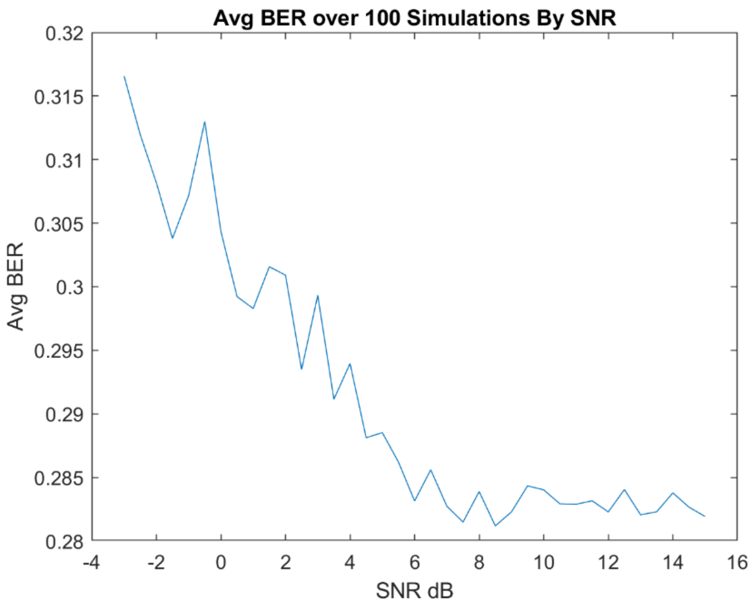


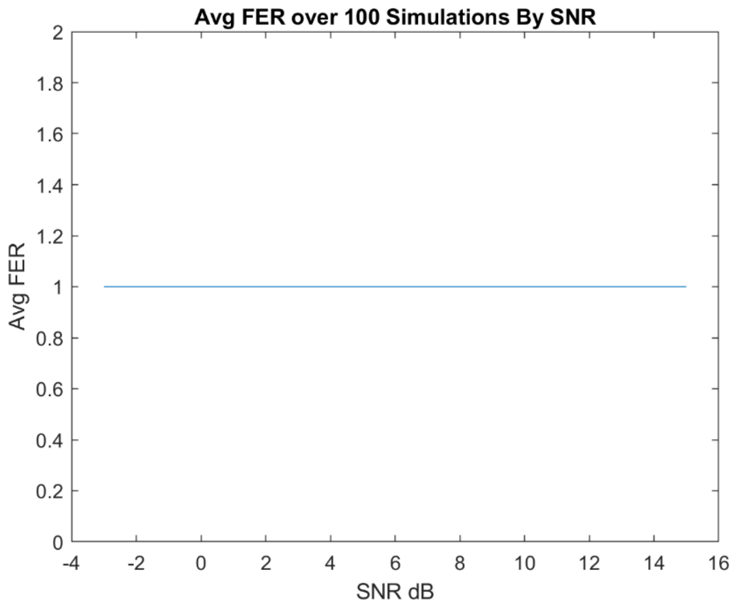


SNR	Time Offset	Frequency Offset	Simulations
-3dB ~ 15dB	2.5/10ms	600 Hz	100

Mean of Time Delay(samples)	STD of Time Delay	Mean of Frequency distortion	STD of Frequency Distortion
43.8003	0.8144	625	0

In this case, the time delay is not a multiple of 16 and frequency distortion not multiple of 125Hz, but since the difference is small so our system is still able to detect the transmitted data. Therefore, we can still estimate the frequency distortion but not the time delay. Therefore, it is impossible to obtain a 0 bit error rate no matter how high the SNR is (shown in the figure of bit error rate) and also the reason why frame error rate would be always 1.

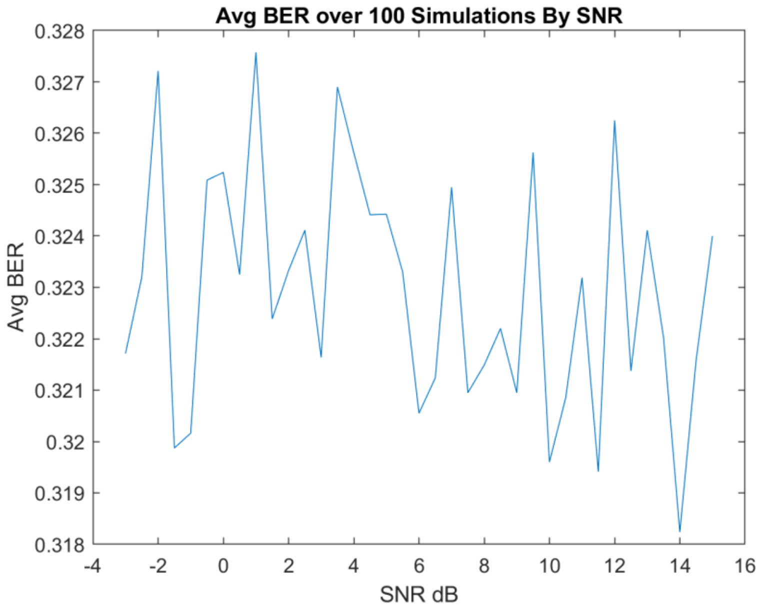


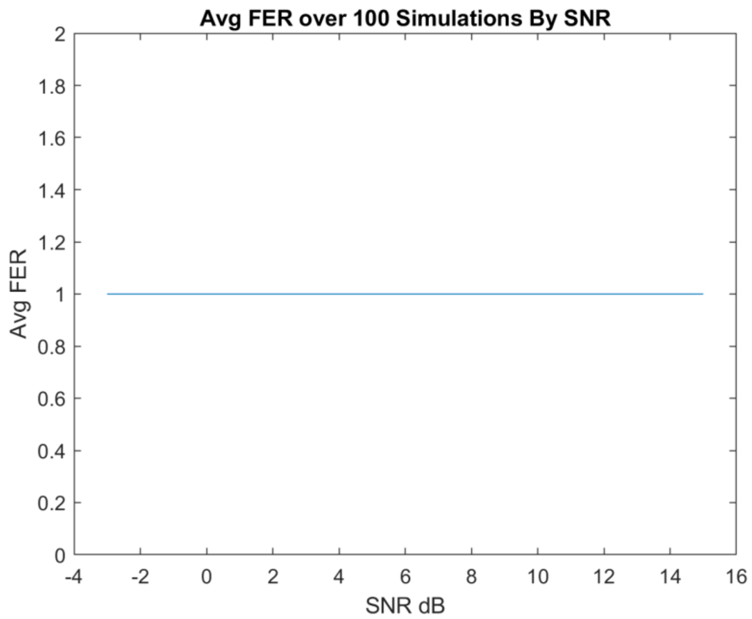


SNR	Time Offset	Frequency Offset	Simulations
-3dB ~ 15dB	2.5/80ms	62.5 Hz	100

Mean of Time Delay(samples)	STD of Time Delay	Mean of Frequency distortion	STD of Frequency Distortion
39.79	9.2767	66.8919	62.3894

In this case, the time delay is not a multiple of 16 and frequency distortion not a multiple of 125Hz. With our system, it is impossible to estimate the frequency distortion. Therefore, in bit error rate will remain high no matter what the SNR is and the frame error rate will always be 1 since bit error rate will never approach to 0 (within this SNR range).


Avg BER over 100 Simulations By SNR
Avg BER
SNR dB
0.318
0.319
0.32
0.321
0.322
0.323
0.324
0.325
0.326
0.327
0.328
-4
-2
0
2
4
6
8
10
12
14
16


Avg FER over 100 Simulations By SNR
Avg FER
SNR dB
0
0.2
0.4
0.6
0.8
1
1.2
1.4
1.6
1.8
2
-4
-2
0
2
4
6
8
10
12
14
16

IV. Computation Complexity

COST OF RX Algorithm	Number of Additions	Number of Multiplications	Number of Sin() and Cos() functions	Number of divisions	Any non linear operation
LPF RX Filter	85360	85360	0	0	0
Down Sampling	2640	1760	0	0	0
DFT of finding To	12404480	86831360	10853920	10853920	14080
DFT of finding Time Delay	655360	4587520	1310720	1310720	5120
DFT of finding Frequency delay	16384	114688	32768	32768	128
Time & Frequency compensation	0	8000	1600	800	0
Total	13164224	91628688	12199008	12165440	19328