UTKU ACAR 250206062

LAB-9 REPORT

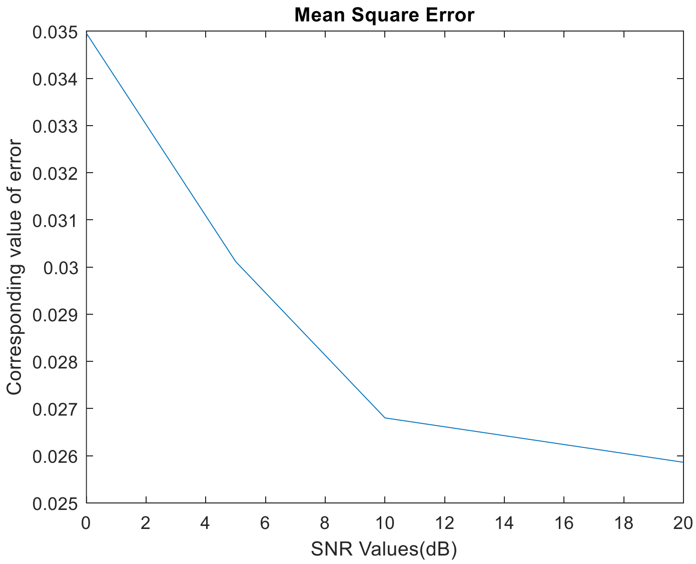

Figure 1:9.4. b (cumsum(.) with message length n)

dB to Magnitude (Energy)→ 1dB ==$10^{0.1}$ Logic is pow (10, dB/10)

SNR (Signal to Noise Ratio) → Message Power/Noise power

We want to increase this ratio for quality

We can see in figure 1 that corresponding error values are decreasing while SNR values are increasing. The ratio of change(slope) also decreases with increasing SNR.

When we listen each sample the quality of the gong voice getting better while SNR increasing. Since SNR means ratio between signal power and noise power the high SNR means total power composed of our signal power more than noise power. Lower SNR value means noisier signal.

The choice of low pass filter order(10) was satisfactory due to error values and cutoff was Fc(2kHz)

Trial Notes:

I have not expected error values to be so close while SNR increases. We have learned that FM is more durable to the noise, but I think there was a mistake in kf. Because we have carrier frequency as 2kHz but for constant Am value (1) we have also 10000 kf so freqdev=10000*1=10kHz which is higher than our carrier frequency (higher even sampling frequency). This is like cutting 100-meter rope into 1000-meter parts. Which has no meaning to me. However, if we divide the kf by 10 we get 1000 and this is less than fc so kf*Am(1kHz) can divide Fc into 2 and get 1kHz parts.

My first technique requiring multiplication the demodulated signal with Am, but I think if we multiply FM signal's phase part (2*pi*kf*cumsum(mt)/fs) with Am first, then we do not need to multiply Am at the demodulation part. Because we are already giving the modulated signal with frequency deviation to the demodulator(fmdemod). I did some proof of this idea actually same with using fmmod() function on previous sixth lab.
My modifications:

1) Am→1→0.1 or kf→10000→1000 (changing kf is more logical since we do not know Am exactly (21st row))
2) xfm→2*pi*kf*cumsum(mt)/fs→2*pi*Am*kf*cumsum(mt)/fs(24th row)
3) Also, at rows between 33 and 36 Am should be discarded from equations
4) For correction comment out 25th row fmmod(.) function and comment 23rd row.
5) For more detailed inspection we can comment out rows from 62 to 79 to look for frequency domain and time domain.

Modification results:

Old

MSE1	0.0370
MSE2	0.0347
MSE3	0.0326
MSE4	0.0319

→→→→

Modified

MSE1	1.0778
MSE2	0.5360
MSE3	0.1766
MSE4	0.0639

After modifications, the slope became like below:

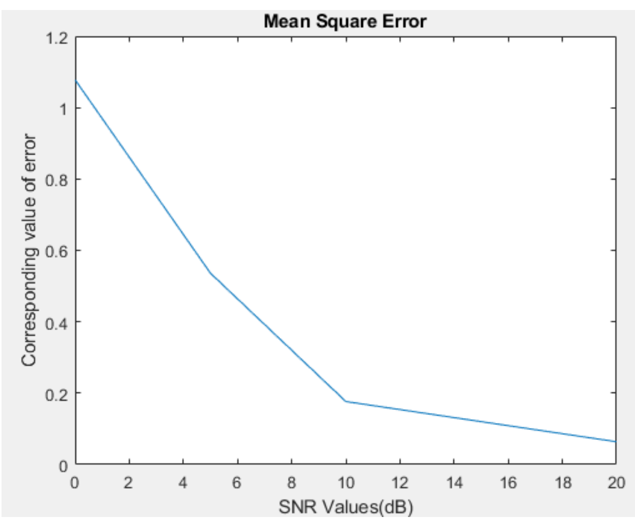

Figure 2(cumsum(.) with sampling frequency Fs)

But I have noticed that the sound is better when I use length of the message(n) instead of sampling frequency in phase part of the xfm (row 23) instead of doing all the modifications above. This change has increased my precision since I decrease each width of integral element. So, the error is lower but also close to each other and I think this is because of durability of FM to the noise.