EE6143 Matlab Assignment 3

Pothuganti Harshavardhan
EE17B061

April 11, 2021

OFDM_AWGN.m

5G NR specific constraints

- 1 slot = 14 OFDM symbols
- 1 resource block = 12 resource elements
  1 OFDM symbol = 50 resource blocks
  1 OFDM symbol = 600 resource elements
  1 resource element = 1 M-QAM symbol

Simulation details

Symbols are gray coded to get the least bit error rate for the same symbol error rate. Nearest neighbour decision rule is used for decoding received symbols.

IFFT and FFT

600 Sub carriers are mapped to 4096 sub carriers(during IFFT) using the mapping provided in the 5G spec

$$k\_new = mod(k\_old - 600/2, 4096)$$

Circular shift property of FFT

$$x = [x[0], x[1], x[2], ...x[N-1]]$$

If x is circular shifted by t ($0 \leq t < N$)

$$x\_t = [x[N-t], x[N-t+1], ...x[N-1], x[0], x[1], x[2], ...x[N-t-1]]$$

$$y = fft(x)$$
$$y\_t = fft(x\_t)$$
$$y[k] = y\_t[k] * e^{\frac{2\pi j * timing\_offset * k}{fft\_size}}$$

timing offset

timing offset is corrected using the cyclic(circular) shift property of FFT. So we can correct any timing offset as long as it is less than 288(minimum CP length). After which the circular shift property of FFT can no longer be used as the input to FFT will be from two different OFDM symbols.

timing offset of t_0 is equivalent to circularly shifting FFT input by t_0 (t_0 ¡ 288)

CFO

Offset in frequency domain is equivalent to continuously rotating the constellation in time domain.

let u(t) be the complex message signal transmitted using a carrier frequency of f_1. If the receiver frequency is f_2, then $\hat{u}(t)$ (receiver's estimate of u(t)) is given by

$$\hat{u}(t) = u(t) * e^{2\pi j(f_1 - f_2)t}$$
$$CFO = \frac{\Delta f}{Subcarrierspacing} = \frac{f1 - f2}{SCS}$$
$$\hat{u}(t) = u(t) * e^{2\pi j(CFO*SCS)t}$$
$$SCS = \frac{SamplingRate}{N_{fft}} = \frac{1}{N_{fft} * T_s}$$
$$\hat{u}(t) = u(t) * e^{2\pi j(\frac{CFO}{N_{fft}*T_s})t}$$
$$\hat{u}(nT_s) = u(nT_s) * e^{2\pi j(\frac{CFO}{N_{fft}*T_s})nT_s}$$
$$\hat{u}[n] = u[n] * e^{2\pi j*CFO*\frac{n}{N_{fft}}}$$

This is equivalent to rotating the constellation by an angle of $2\pi\frac{CFO}{N_{fft}}$ for each time step(sampling period).

Plots and Observations

BER curves for AWGN channel and timing offset

In Figure 1, The subplot on the left has timing offset corrected which is equivalent to an AWGN channel with no timing offset (as long as the timing offset is less than minimum CP length(i.e, 288)) and the subplot on the right has uncorrected timing offset.

In Figure 1 4e is empirical BER plot for 4-QAM and 4t is theoretical BER plot for 4-QAM.

BER Timing offset corrected or AWGN channel

we can see that theoretical BER and empirical BER are almost same for high $\frac{E_b}{N_o}$.

theoretical BER is lower than empirical BER when $\frac{E_b}{N_o}$ is low and the difference is high for large M. This is because when $\frac{E_b}{N_o}$ is low, noise variance is significantly large so not all the errors are nearest neighbour symbols which increase bit error rate since non nearest neighbour symbols differ by more than 1 bit. So actual BER is more than our theoretical estimate.

As M increases the minimum distance between symbols decreases for the same $\frac{E_b}{N_o}$ so the deviation from theoretical BER is more for larger M.

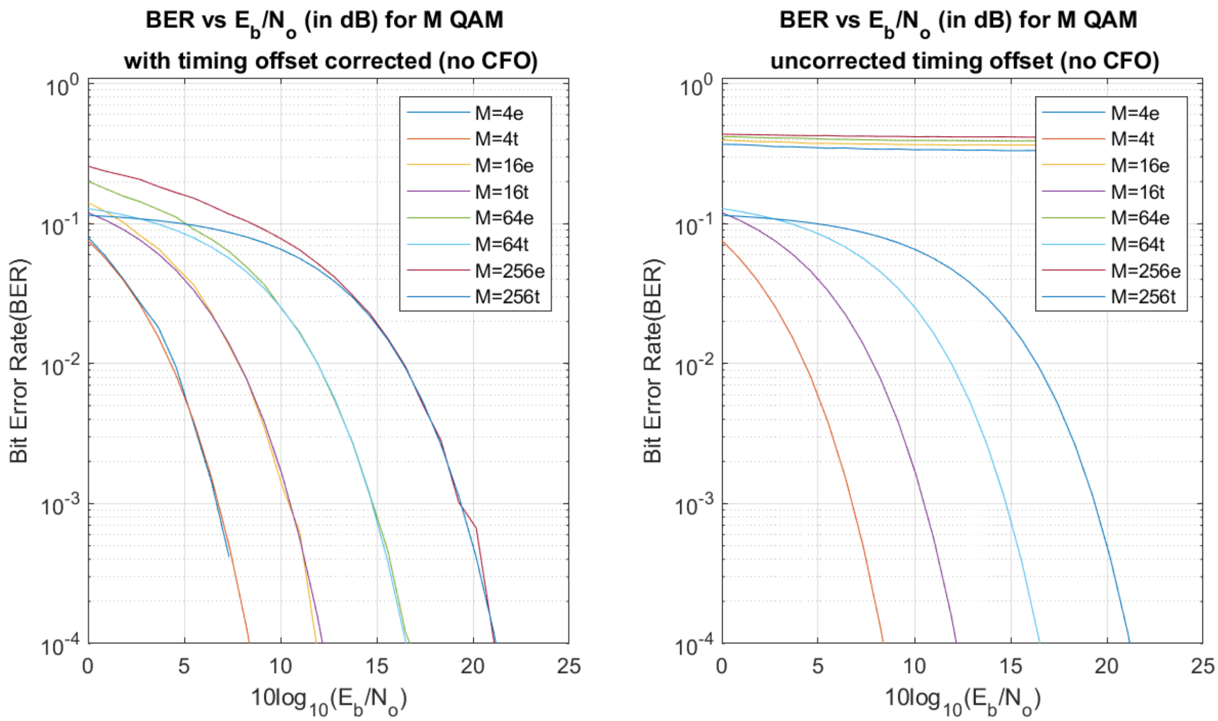

Figure 1: BER_Plots with and without timing offset correction

BER uncorrected timing offset

from Figure 1 the empirical BER is constant (approximately equal to 0.5) for any QAM (irrespective of SNR). This is approximately same as **uniformly random prediction** of bits.

This is because Timing offset has an effect of rotating the constellation(because the circular shift of FFT input adds different phase to different subcarriers) which can be seen from the received symbols scatter plots in Figure 2

So for any SNR the symbol is equally likely to be decoded as any of the symbols which have the same energy(square of distance from origin). which makes the prediction similar to uniform random prediction.

The constant is slightly higher for higher QAM. This could be because higher QAM have lower minimum distance for the same Eb. So even small phase shifts from FFT cyclic shift can result in incorrect decoding/estimation of the symbols.

Effect of timing offset on Received Symbols

In Figure 2, The subplot on the left has timing offset corrected which is equivalent to an AWGN channel with no timing offset (as long as the timing offset is less than minimum CP length(i.e, 288)) and the subplot on the right has uncorrected timing offset.

Timing offset has an effect of rotating the constellation because the circular shift of FFT input (due to timing offset) adds phase randomly (depending on which subcarrier the symbols is in). Each symbol could have been transmitted from any of the subcarrier. So every symbol in the constellation undergoes different(random) shift each time it is transmitted, effectively rotating the constellation.

since we have 3 possible distances of a symbol from the origin (0 + 0i) we have 3 rings in the plot and the thickness(outer radius - inner radius) of the rings is proportional to noise standard deviation.

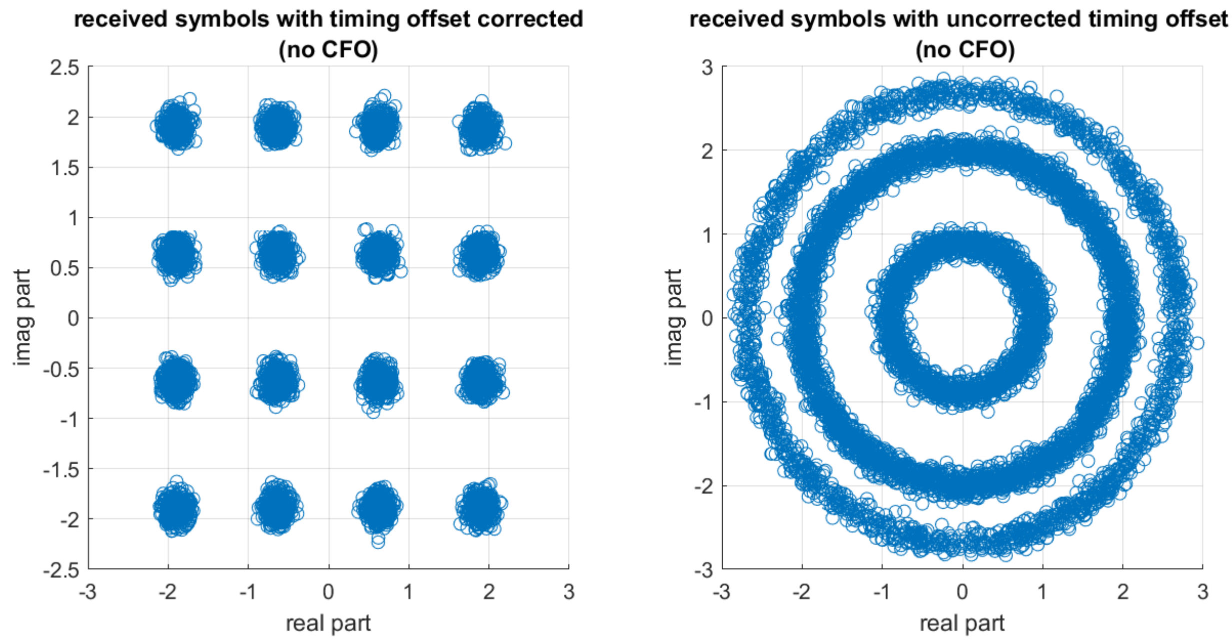

Figure 2: received symbols with and without timing offset correction

effect of CFO

From figure 3 CFO doesn't seem to have much effect on 4-QAM as long as it is less than 0.3 % **BER is increasing for every SNR as CFO increases**. CFO is equivalent to rotating the received constellation and the angle of rotation increases with time since we are not sending many symbols, angle of rotation is less since total message duration is less.

$$
\begin{aligned}
Max\_angle\_of\_rotation &= 180 * MAX\_CFO * \frac{total\_transmitted\_sequence\_length}{N\_fft} \\
&= 180 * 0.003 * 61440/4096 \\
&= 8.1 degrees
\end{aligned}
$$

max angle shift is 8.1 degrees which is not much considering we are using 4-QAM. For 4 - QAM we need about 40 to 45 degree shift(assuming high SNR) for the BER to be too high.

4-QAM has the highest minimum distance so it is robust to small angles of rotation so the BER curves are not flat like the ones we have seen for timing offset. Although these will be flat if we increase the number of OFDM symbols which increases the message duration and max angle of rotation of the constellation. So in a real transmission we should keep periodically estimating the transmitter frequency in order to keep the BER low.

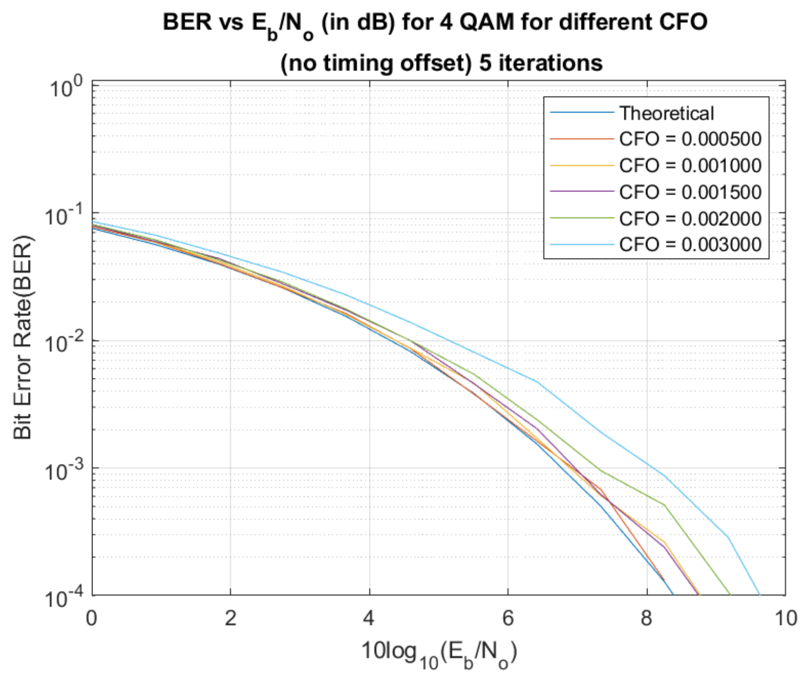

Figure 3: BER for 4-QAM for different CFO

OFDM_TDL.m

channel estimation

OFDM makes each subcarrier equivalent to a single tap channel with white noise addition. we need to estimate the value of this single tap in order to decode transmitted symbols.

even numbered OFDM symbols (2,4,6 ... 14) don't have any pilots in them.
odd numbered OFDM symbols (1,3,5,.... 13) have pilots in every odd numbered sub carrier (1,3,5...599)

Channel estimation for odd numbered OFDM symbols

for RE(Resource element) which are used to transmit pilots we can estimate the channel by dividing the received symbol with pilot value.

$$\begin{aligned} received\_symbol &= channel\_value * pilot\_value + noise \\ channel\_estimate &= \frac{received\_symbol}{pilot\_value} \\ &= channel\_value + \frac{noise}{pilot\_value} \end{aligned}$$

for RE which don't have pilots we can average the channel estimates of the two adjacent RE which are used to send pilots.

$$channelestimate(i^{th}RE) = \frac{channelestimate((i-1)RE) + channelestimate((i+1)RE)}{2}$$

i = 2,4,6.... 598;

for the 600th RE there is only one adjacent RE(599th RE) so the estimate of 599^{th}RE for 600^{th} RE is used.

Channel estimation for even numbered OFDM symbols

even numbered OFDM symbols uses the channel estimate as the most recent ODD numbered OFDM symbol.

channel estimate(2^{nd} OFDM symbol) = channel estimate(1^{st} OFDM symbol)
channel estimate(4^{th} OFDM symbol) = channel estimate(3^{rd} OFDM symbol)
....
channel estimate(14^{th} OFDM symbol) = channel estimate(13^{th} OFDM symbol)

Zero Forcing Equalization

we divide the received symbol with the its corresponding channel estimate to get an estimate of the transmitted symbol.

Plots and Observations

fd = 0

In Figure 4 Empirical BER curves look like up scaled version of theoretical AWGN channel curves.

Assuming we have almost perfect estimate of channel.
channel_estimate ≈ channel_value

$$\begin{aligned}
received_symbol &= channel_value * transmitted_symbol + noise \\
transmitted_symbol_estimate &= \frac{received_symbol}{channel_estimate} \\
&= \frac{channel_value}{channel_estimate} * transmitted_symbol + \frac{noise}{channel_estimate} \\
&\approx transmitted_symbol + \frac{noise}{channel_value} \\
&\approx transmitted_symbol + noise_amplified
\end{aligned}$$

magnitude of channel_value can be ≤ 1. **So TDL channel (fd = 0) with zero forcing equalization is equivalent to AWGN channel with noise amplified(Assuming perfect channel estimate). This explains why the BER curves in Figure 4 are not identical to AWGN channel**.

Effect of Imperfect channel estimate

$$
\begin{aligned}
transmitted\_symbol &= x + yj \\
transmitted\_symbol\_estimate &= \frac{received\_symbol}{channel\_estimate} \\
&= \frac{channel\_value}{channel\_estimate} * transmitted\_symbol + \frac{complex\_noise}{channel\_estimate} \\
&= (a + bj) * transmitted\_symbol + \frac{complex\_noise}{channel\_value} \\
&= (a + bj) * (x + yj) + noise\_amplified \\
&= (a * x - b * y) + (b * x + a * y)j + noise\_inphase + noise\_quadraturej \\
a \approx 1, b &\approx 0(for\_good\_estimates) \\
inphase\_estimate &= a * x + (-b * y + noise\_inphase) \\
quadrature\_estimate &= a * y + (b * x + noise\_quadrature)
\end{aligned}
$$

For imperfect estimates of channel there are two sources of errors

- amplified noise
- interference from other signal component

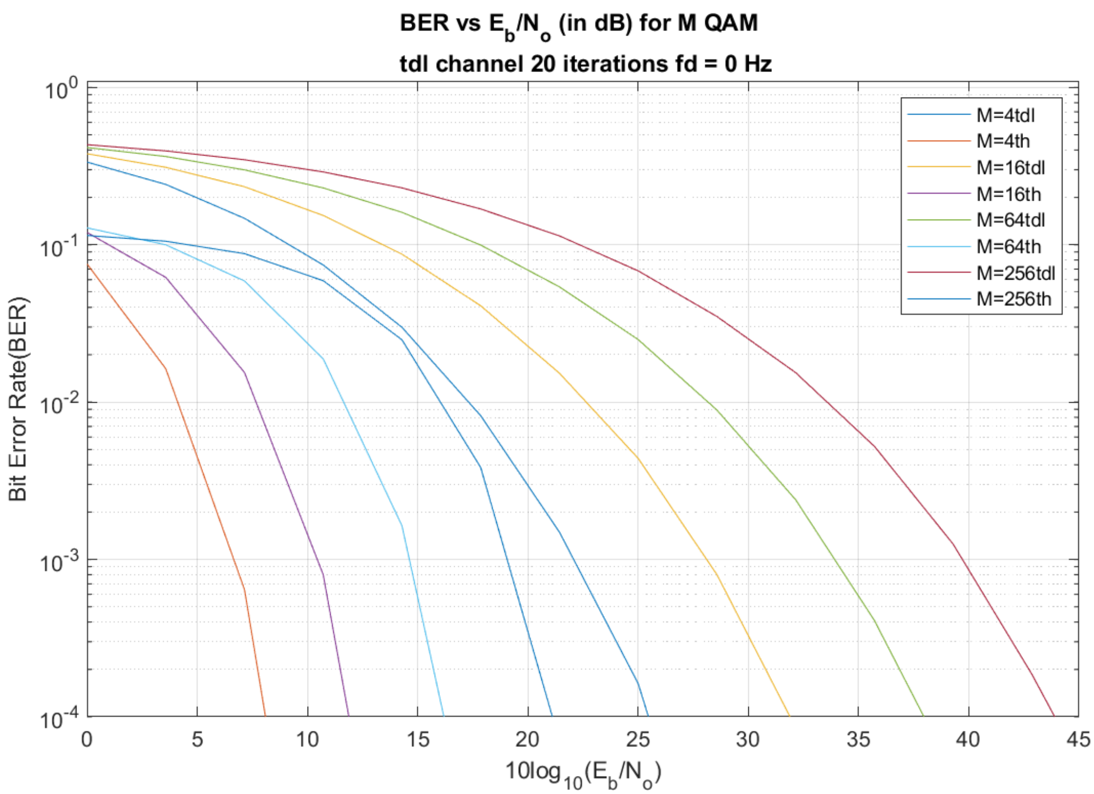

Figure 4: BER for fd = 0 Hz

1 Bonus Simulation fd $\neq$ 0

For non zero Maximum Doppler Shift(fd) the channel keeps changing with time (as symbols are sent through it). This reduces channel estimation accuracy. From figures 5, 6, 7 tdl BER plots seems to flatten as SNR increases.

$$inphase\_estimate = a * x + (-b * y + amplified\_noise\_inphase)$$
$$quadrature\_estimate = a * y + (b * x + amplified\_noise\_quadrature)$$

At low SNR most of the errors are due to Gaussian Noise. At high SNR BER is due to interference from other signal component caused by imperfect estimation of channel. we can improve channel estimation by using pilots in even numbered OFDM too.

BER is higher for higher fd since the channel changes faster, channel estimates get more inaccurate.

BER is higher for higher QAM since magnitude of interfernce (proportional to signal in other component) is higher.

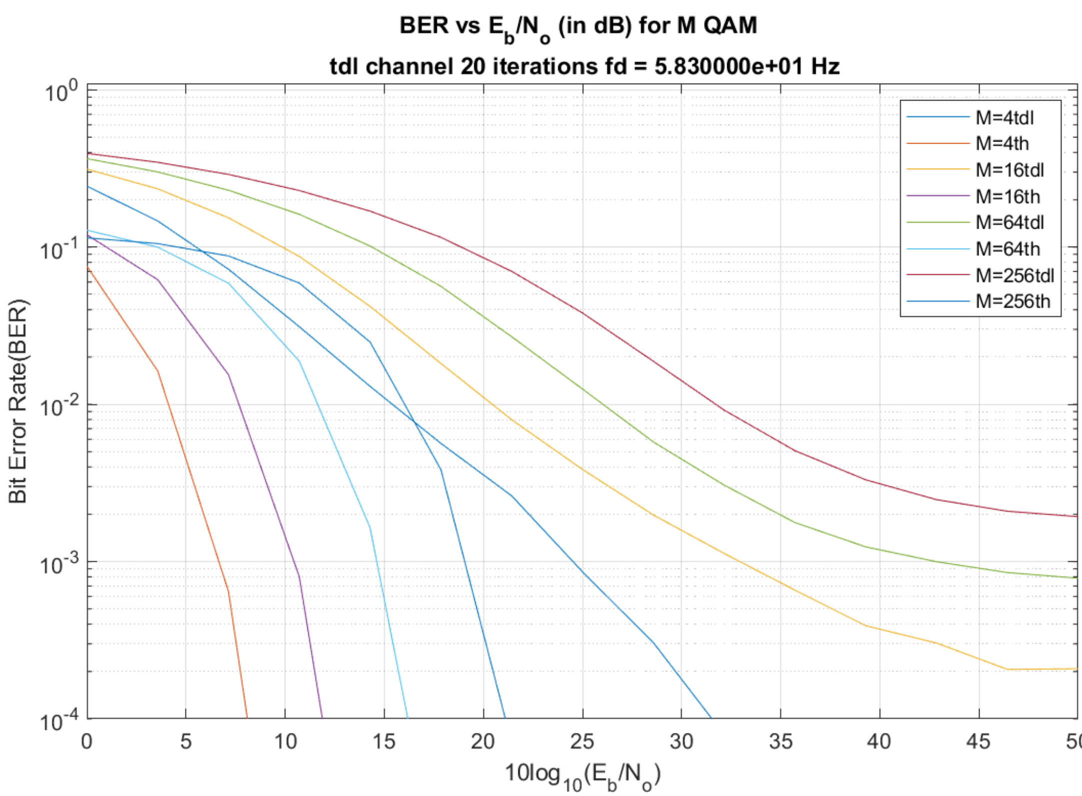

Figure 5: BER for fd = 58.3 Hz

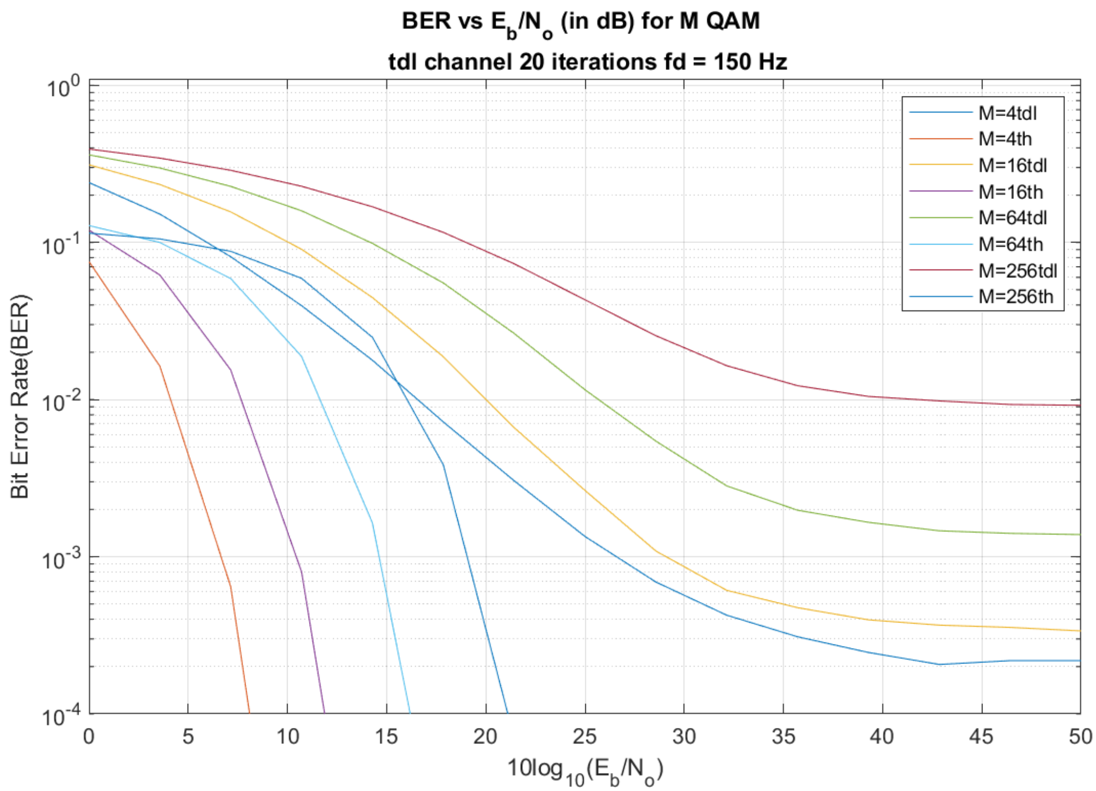

Figure 6: BER for fd = 150 Hz

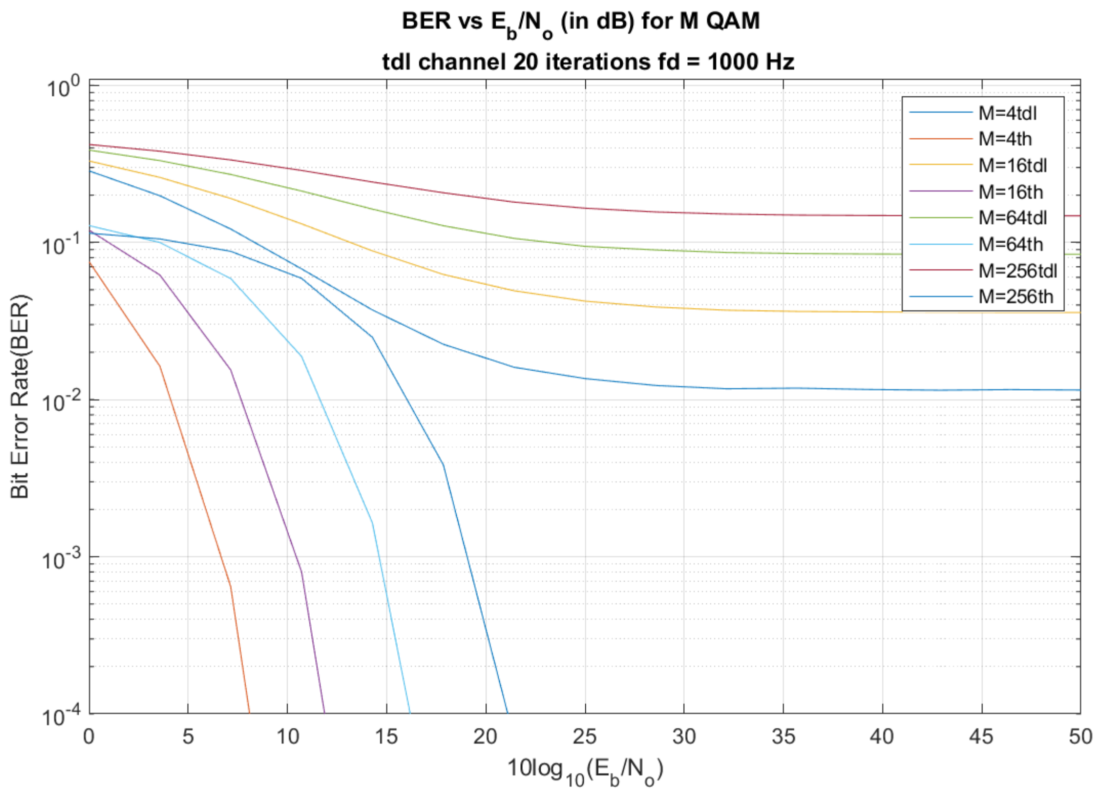

Figure 7: BER for fd = 1000 Hz