Project 1: Design of a Communication Link – Part I

William Chen, Jongoh (Andy) Jeong, Yilun (Eric) Jiang

ECE-300 Fall 2018

Professor Keene

5 December 2018

TABLE OF CONTENTS

APPROACH 3

ADAPTIVE EQUALIZER STRUCTURES EXPLORED 3-5

LINEAR EQUALIZER - 1 ....
NONLINEAR EQUALIZER - 1 ....
NONLINEAR EQUALIZER - 2 ....

CONSIDERATIONS 5

BIT ERROR RATE ....
TIME AND BER PERFORMANCE TRADEOFF ....
BIT RATE ....
VARY TRAINING SEQUENCE LENGTH ....

EQUATIONS 5

BIT RATE ....
SNR SCALE ADJUSTMENT FOR M-ARY MODULATION SCHEMES ....

DATA – PART I 6

MATCH THEORETICAL CURVES ....
EFFECT OF MODERATE ISI CHANNEL ....
EFFECT OF EQUALIZER ON MODERATE ISI CHANNEL ....
GRAPHICAL ANALYSIS: EQUALIZER COMPARISON ....
LIMITS FOR BETTER BER – WITH MLSE EQUALIZER ....

CONCLUSION 6

APPENDIX/GRAPHS 7-125

PART I

- Approach

1. Match the theoretical (uncoded) performance curves derived using the MATLAB function berawgn() for 2-, 4- and 16- QAM modulation schemes with proper noise scale factor through no channel
2. Equalize the channel to reduce the bit error rate to the order of 10^{-4} through the moderate ISI channel
3. General Block Diagram of the employed Transmission System

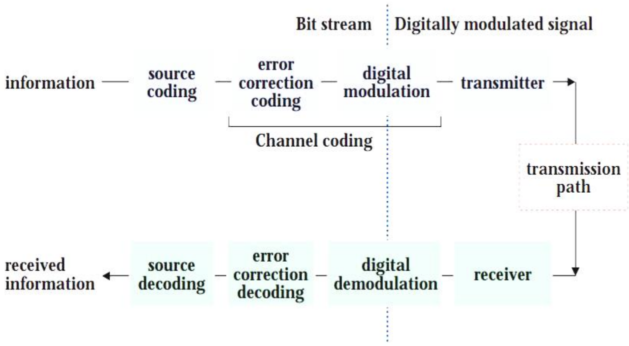


Figure 1: Digital transmission system

The main processing loop includes:

- Data generation
- Data encoding (Part II)
- QAM modulation
- Channel filtering
- Signal equalization
- QAM demodulation
- Data decoding (Part II)
- Error computation

- Adaptive Equalizer Structures explored

1. Linear Equalizer
   - LMS Linear Equalizer (no feedback, simple, but suboptimal)

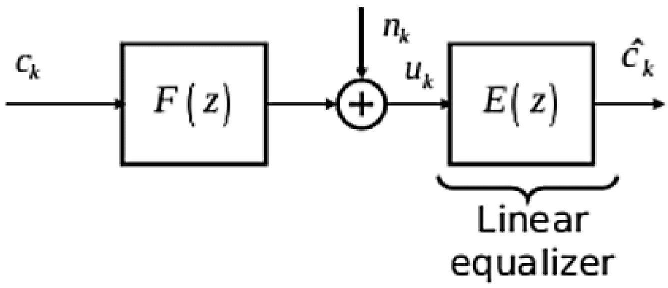


Fig. 1: Block Diagram of a Simple Linear Equalizer

- Can eliminate ISI when there is a null in the frequency response of a channel
- Properties
  - `nWeights: 4,`
    `Stepsize: 0.01 for M = 2, 0.001 for M > 2`
    `(experimentally determined)`

2. Nonlinear Equalizer - 1

 - LMS Decision-Feedback Equalizer (DFE; feedback, feedforward filter, not optimal in minimizing error in the detection of the information sequence from the received signal samples)

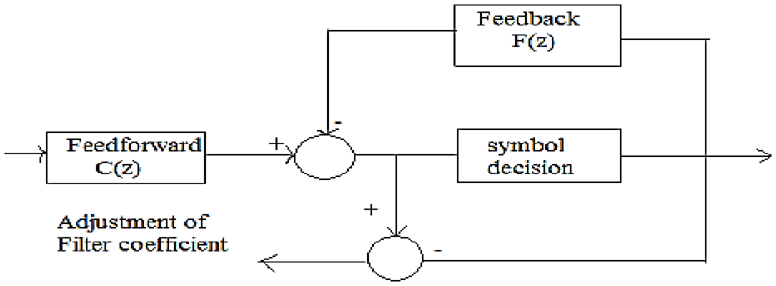


Fig 2: Block Diagram of a Decision-Feedback Equalizer

 - Uses previous decisions in estimating the current symbol with a symbol-by-symbol detector; any tailing ISI from previous symbol is reconstructed and then subtracted
 - Contains a forward and feedback filter; forward filter is similar to the linear equalizer, while feedback contains a tapped delay line
 - Can reduce ISI in frequency selective channels when a null is present without noise enhancement
 - Can be symbol-spaced or fractionally spaced
 - Properties
   - `Feedforward weights: 2, Feedback weights: 3, RefTap = 1, ResetBeforeFiltering = 0 Stepsize: 0.01 for M=2, 0.001 for M > 2 (experimentally determined)`

3. Nonlinear Equalizer - 2

 - Maximum-likelihood Sequence Estimation (MLSE; optimal in error detection by testing all possible data sequences and chooses the data with the maximum probability as the output)

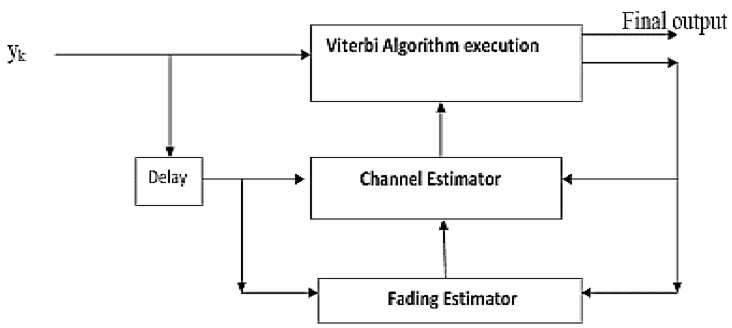


Fig 3: Block Diagram of a MLSE Equalizer

 - First trains itself to build up the channel statistic, and estimates symbols based on the Viterbi algorithm (which uses trellis diagram consisting of M^nstates with M^{n+1}state transitions, where M = length of modulation format, N = length of channel)

- Difference from the other two: does not require training sequence but needs an estimate of the channel for estimation
- For this specific simulations, they are performed with the assumption of perfect knowledge of the channel coefficients; ideal and imperfect knowledge of the channel gives a slight variation in BER values, as Fig 4 demonstrates.
- Properties
    - `TracebackDepth = 1, 10, 100, channel = moderate ISI channel, constellation = M-ary QAM constellation`

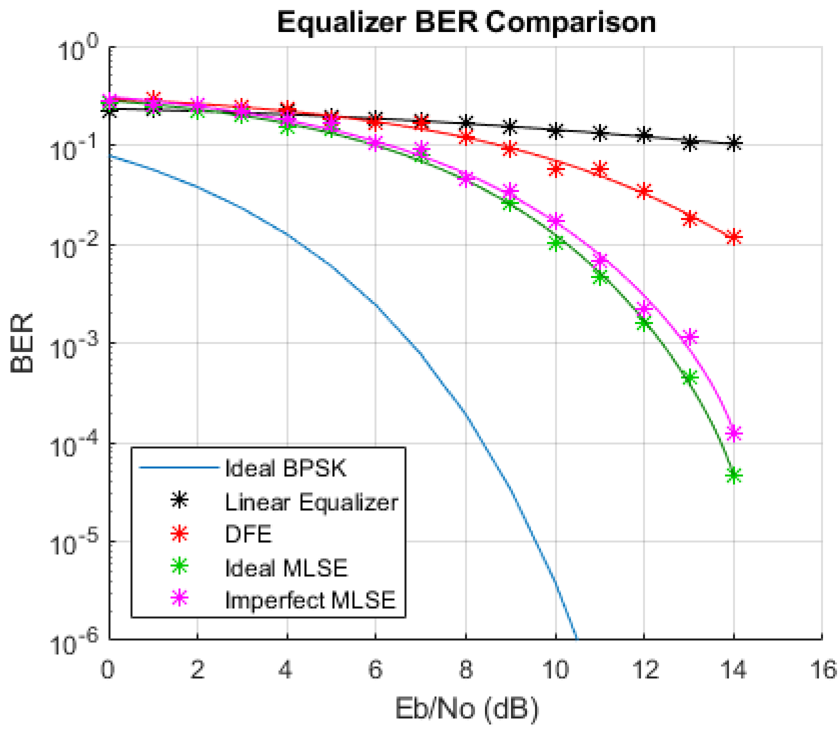


Fig 4: Mathworks Documentation – "ber-performance-of-different-equalizers" (source: https://www.mathworks.com/help/comm/examples/ber-performance-of-different-equalizers.html)

- Considerations

1. Achieve better (non-zero) BERs
    - With the constraint of ~1,000 symbols to transmit per packet, the number of iterations needed to be increased to produce a BER result of orders up to 10e-4 and 10e-6. However, trade-off between efficiency in time and BER outputs is observed.
    - MLSE equalizer required a significantly greater size of symbols for better (non-zero) BERs at higher Eb/No values, so a more time-efficient, yet performance-efficient scheme—DFE—is employed as a measure for comparing error correction code results.
2. Maintain as high bit rate as possible by minimizing training symbols used in the equalizer
    - Vary training sequence length for applicable equalizers and observe bitrates and BER

- Equations

1. $Bit\ rate = (sampling\ rate) \times \frac{\#\ symbols\ Tx}{\#\ total\ symbols\ sent} \times \frac{bits}{symbol} = [\frac{bits}{symbol} Tx]$
2. $SNR\ (scaled, dB) = \frac{Eb}{No}(dB) + 10\log_{10} k + 10\log_{10}\frac{K}{N} - 10\log_{10}(sampling\ rate),$ $where\ N = codeword\ length, K = \ message\ length\ if\ encoded\ (Part\ II);$ $k = 2\ for\ M = 2, otherwise\ \log_2 M$

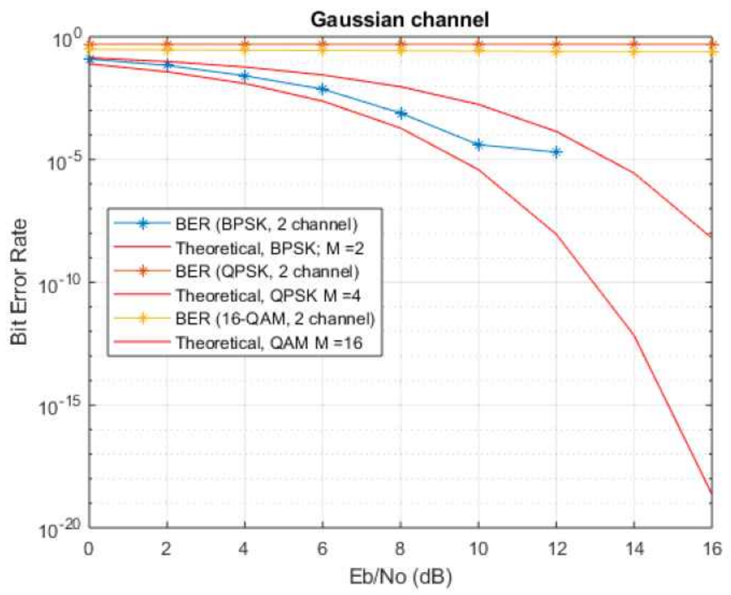
- Data – Part I (graphs in Appendix)

  1. Matching of empirical BER to the theoretical curves for M = 2, 4, 16 with no channel on
     - BER vs. Eb/No (dB) Curve
     - Scatterplots of the Noisy Signal at various SNR values
     - Bit rate
  2. Empirical BER curves for M = 2, 4, 16 with moderate ISI channel, without equalizer
     - BER vs. Eb/No (dB) Curve
     - Scatterplots of the Noisy Signal at various SNR values
     - Bit rate
  3. Empirical BER curves for M = 2, 4, 16 with moderate ISI channel and equalizer with various training sequence lengths
     - BER vs. Eb/No (dB) Curve
     - Scatterplots of the Equalized Signal at various SNR values
     - Bit rate
  4. Performance results
     - Graph: Performance in BER comparison among the different types of equalizers (linear and nonlinear)
     - Graph: Evidence for time complexity limiting the choice of equalizer to use
     - Result: BER performance and time complexity (high → low):
       MLSE → DFE → Linear equalizer, through moderate ISI channel

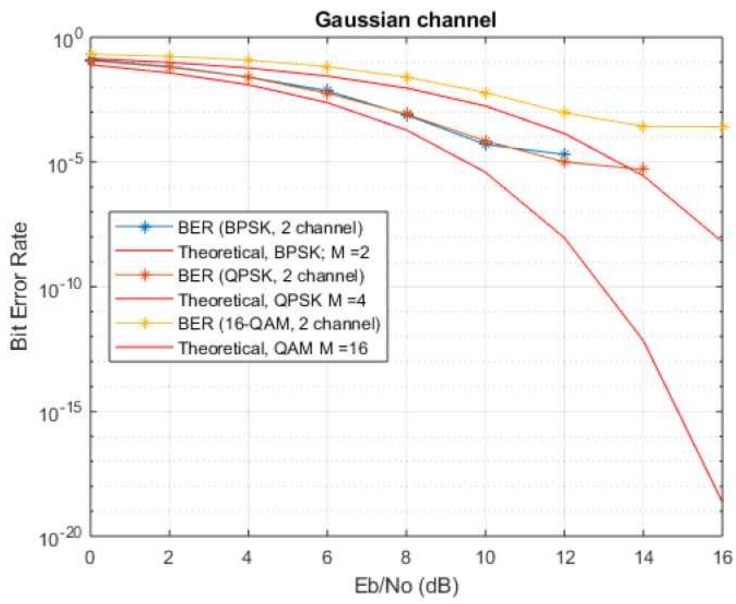
- Conclusion Remarks

  1. From DFE simulations with different training sequence lengths, the ideal model with highest bit rate performance is one with training length of 100 (given `nIter = 100, nSymbols = 1,000`). Above the length gives similar or identical BER, and below gives poorer performance, so 100 is approximately the threshold at which its performance is maximized (bit rate and BER)
  2. For MLSE equalizer simulations, it requires a significantly greater number of symbols sent for more precise BER at higher Eb/No values. In addition, a traceback depth length is directly proportional precision, while not increasing time complexity as much.

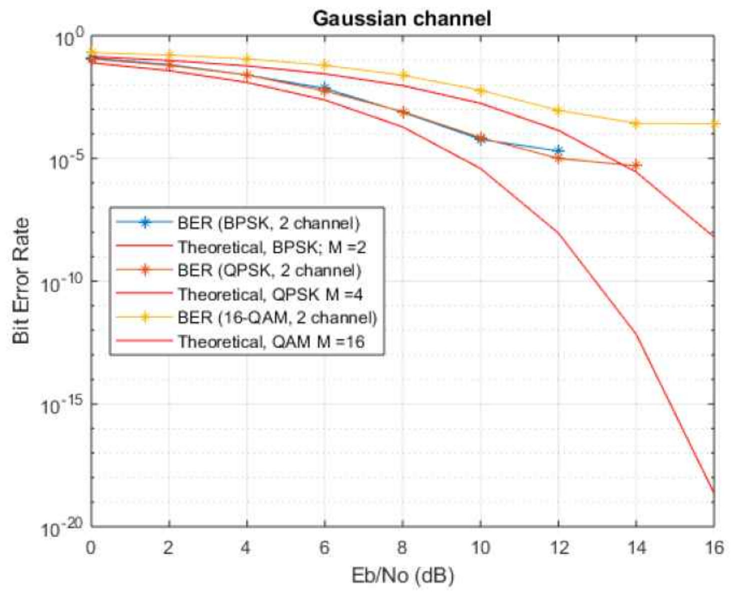
Decision-Feedback Equalizer – training sequence length and BER comparison



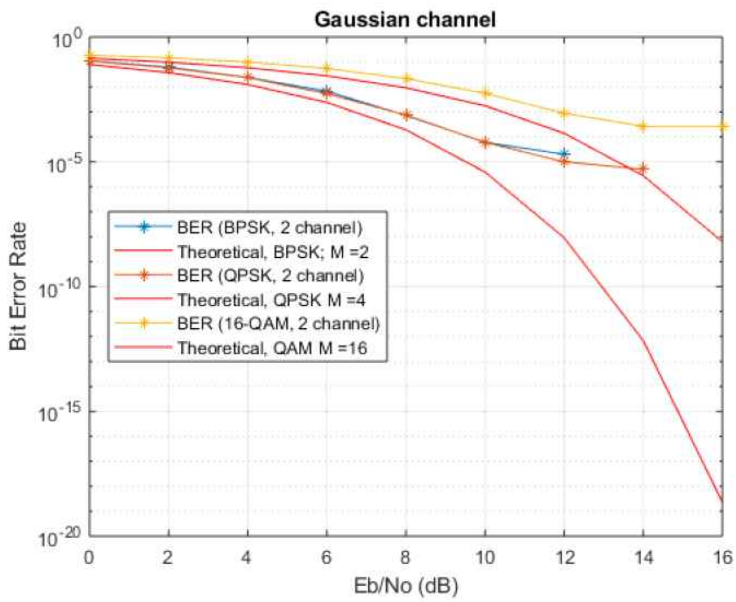
DFE, trainlen = 10, nIter = 100, nSym = 1,000



DFE, trainlen = 100, nIter = 100, nSym = 1,000



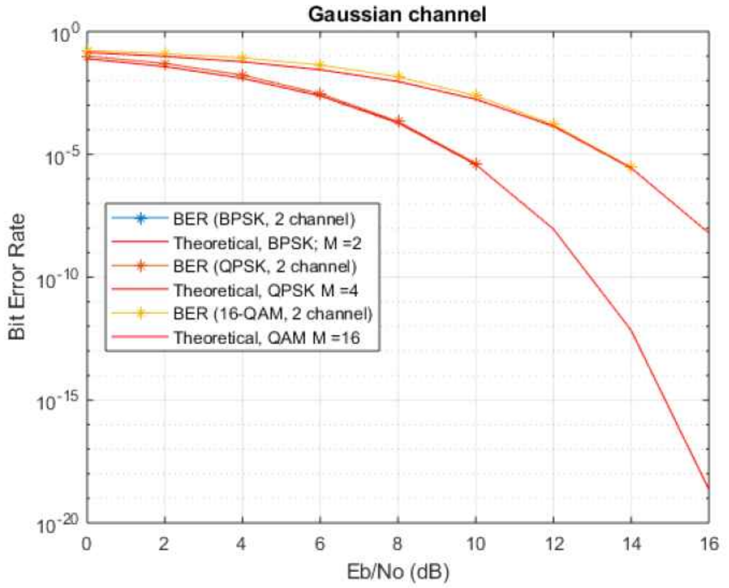
DFE, trainlen = 200, nIter = 100, nSym = 1,000



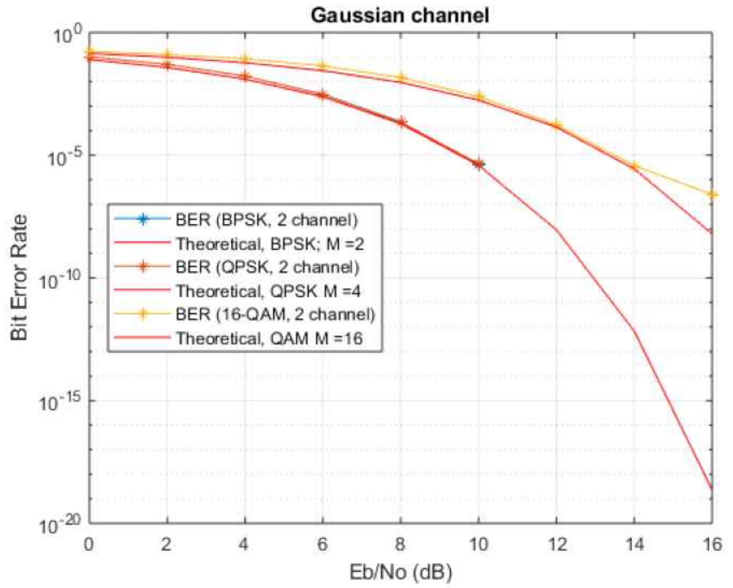
DFE, trainlen = 500, nIter = 100, nSym = 1,000

Comment: for small training sequence, the equalizer fails to equalize the noisy signal for M > 2; however, for the sequence length beyond a threshold (~ 100), the BER performances are almost identical

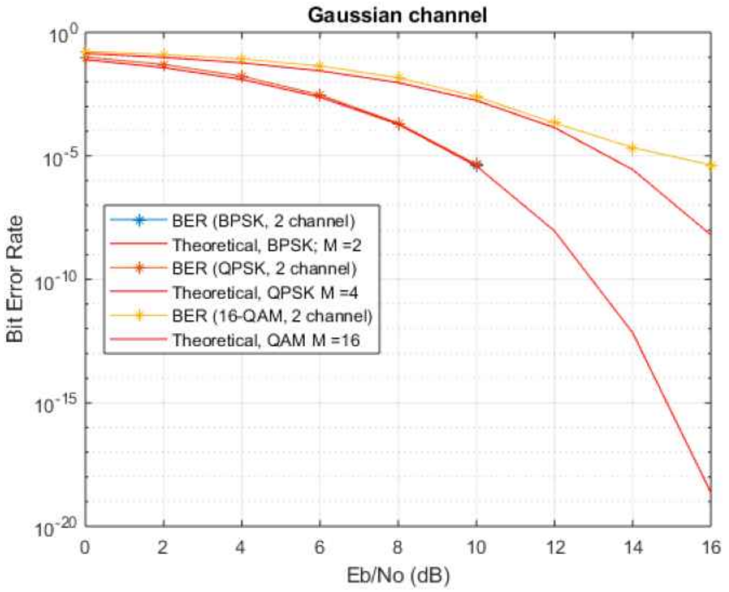
Maximum-Likelihood Sequence Estimator equalizer – BER performance and time complexity trade-off



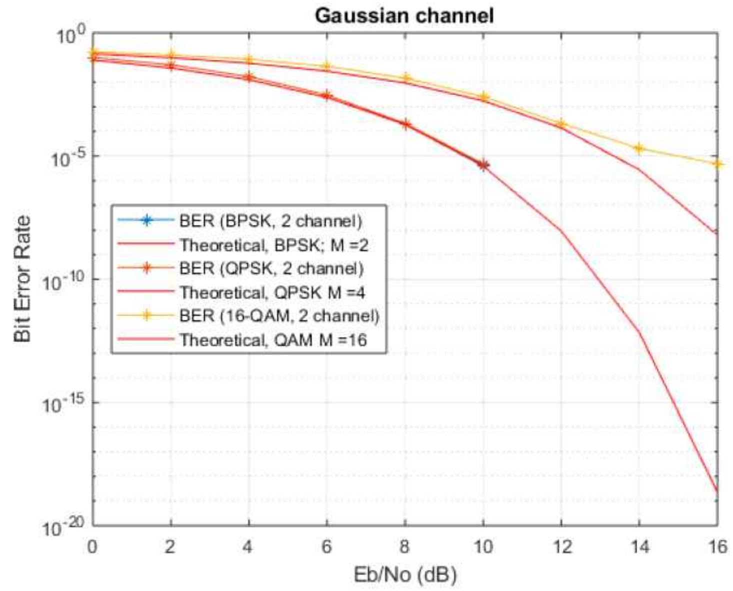
MLSEeq, tbl = 10, nIter = 10, nSym = 1,000,000
Elapsed time ~ 2,950 sec



MLSEeq, tbl = 10, nIter = 100, nSym = 10,000
Elapsed time ~ 308 sec



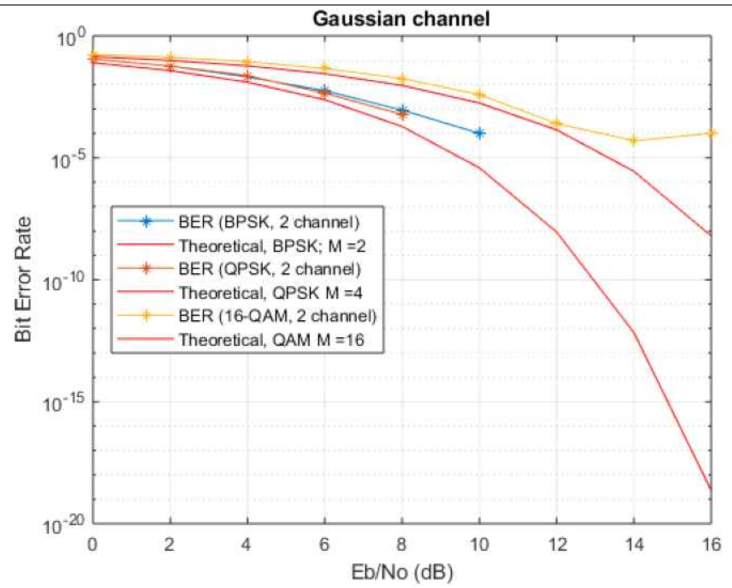
MLSEeq, tbl = 10, nIter = 5,000, nSym = 1,000
Elapsed time ~ 1,642 sec



MLSEeq, tbl = 10, nIter = 10,000, nSym = 1,000
Elapsed time ~ 3,254 sec

Comment: comparison between the two graphs on top shows that for higher number of symbols sent, the better BER performance, but with signficant growth in running time, given the same traceback depths. The bottom two graphs are simulated for the same number of symbols sent (1,000), with different number of iterations performed; the time complexity linearly grows with the number of iterations.

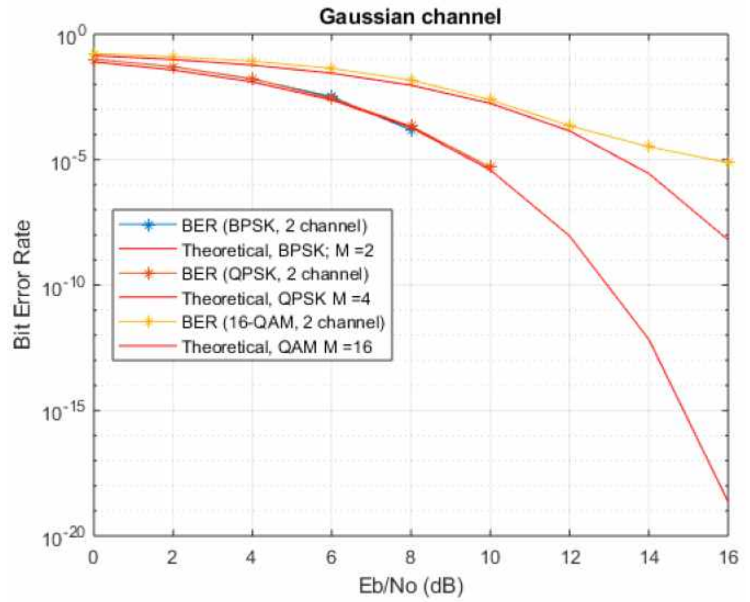
Maximum-Likelihood Sequence Estimator equalizer – traceback depths and time complexity limitation



MLSEeq, tbl = 10 nIter = 10, nSym = 1,000

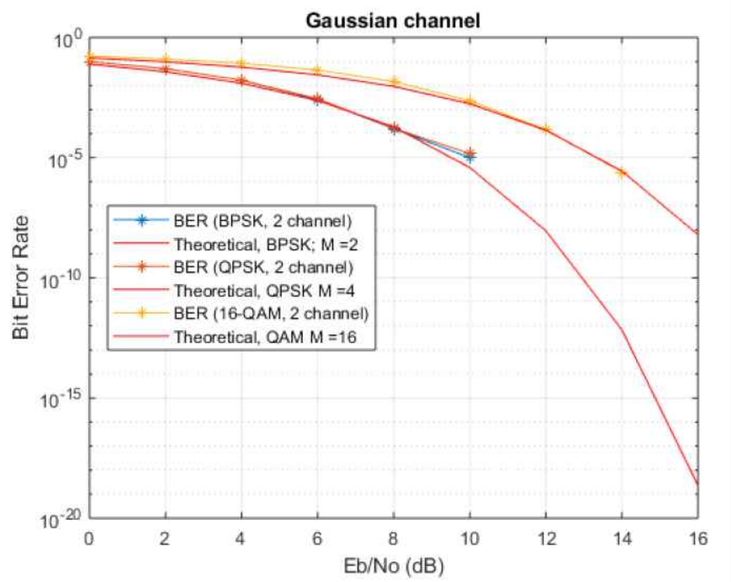
Elapsed time ~ 3.58 sec

-



MLSEeq, tbl = 100 nIter = 100, nSym = 1,000

Elapsed time ~ 35.70 sec



MLSEeq, tbl = 100 nIter = 1, nSym = 100,000

Elapsed time ~ 38.31 sec

Comment: with the experimental result from the previous page (time grows proportionally with iterations) in mind, the bottom two graphs have iteration and symbol numbers decreased and increased by the same factor, 100, and have similar time complexities— this implies that time grows proportionally with the number of symbols as well. With these in considerations, the graphs on top and bottom left have traceback depth differed by a factor of 10, and yet have similar running times, implying the depths don't really increase time complexities for MLSE equalizer.

In general, MLSE equalizers as graphs show don't go as far as 12 dB SNR for BPSK and QPSK; non-zero BER values could be found for these by increasing the number of symbols, but time complexity will also need to grow, which limits its uses in other applications (such as for Part II).