UCSD MAS WES268A - Lab 5 Report

07JAN2025

PREPARED BY:
Joshua Hoang
Ryan Shimizu

Contents

1 Part 1: Coding Gain for Linear Block Codes

3. - For a range of E_b/N_0 values from 0dB to 10dB (i.e. eleven 1dB steps), measure the Bit Error-Rate for the following "codes":

a. Uncoded

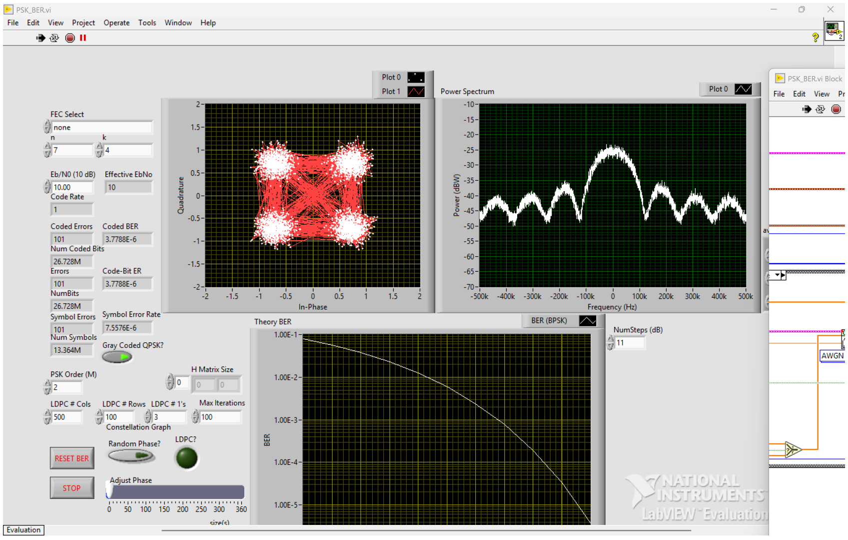


Figure 1: 1.3.a - Uncoded Bit Error Rates under AWGN

b. n-Repetition(3,1,3)

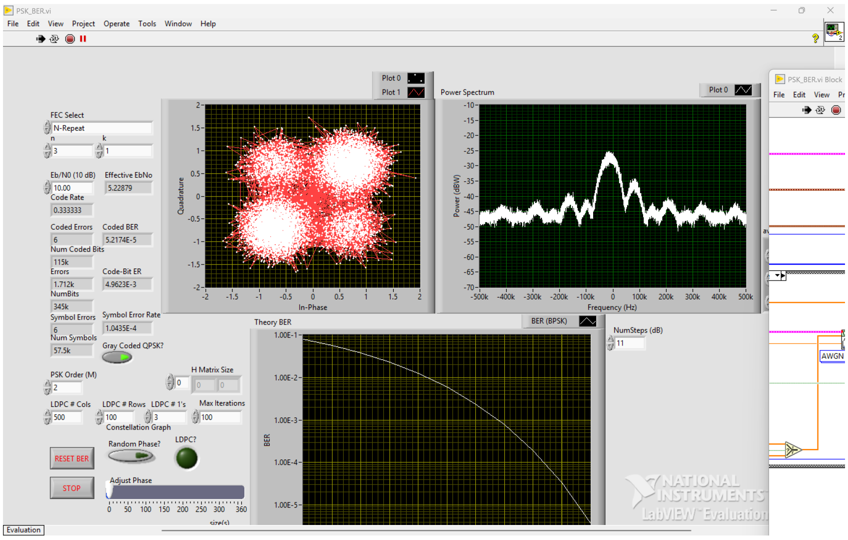


Figure 2: 1.3.b - n-Repetition(3,1,3) Bit Error Rates under AWGN

c. Hamming (7,4,3)

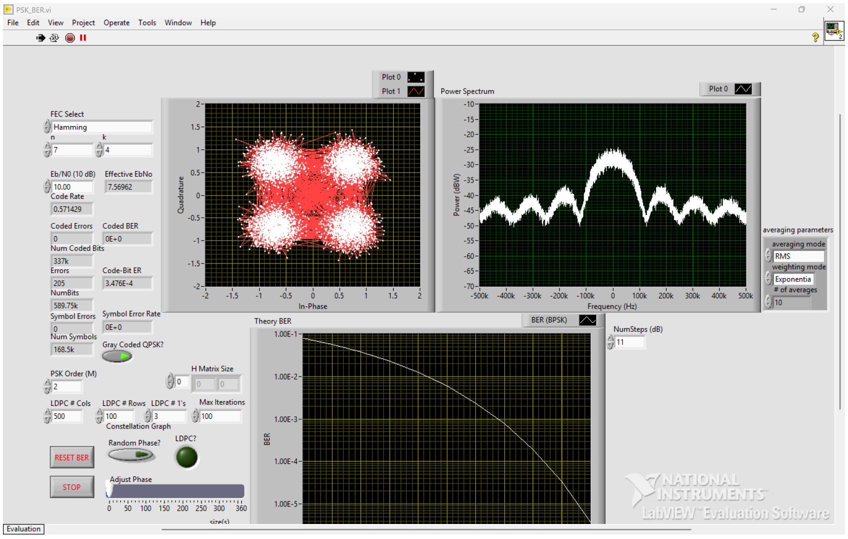


Figure 3: 1.3.c - Hamming (7,4,3) Bit Error Rates under AWGN

d. Golay (24,12,8)

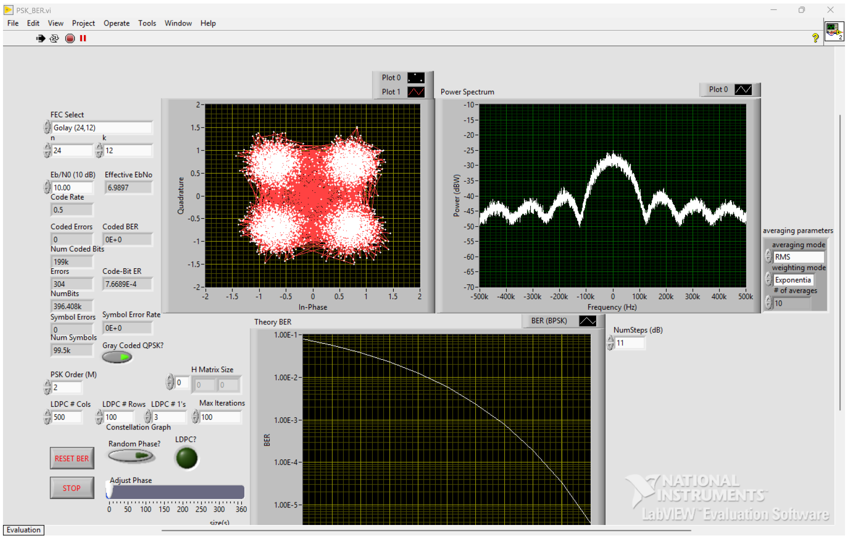


Figure 4: 1.3.d - Golay (24,12,8) Bit Error Rates under AWGN

2 Part 2: Low Density Parity Check Codes

4. - For a range of E_b/N_0 values from 0dB to 10dB (i.e. eleven 1dB steps), measure the Bit Error-Rate for the following "codes":

a. LDPC

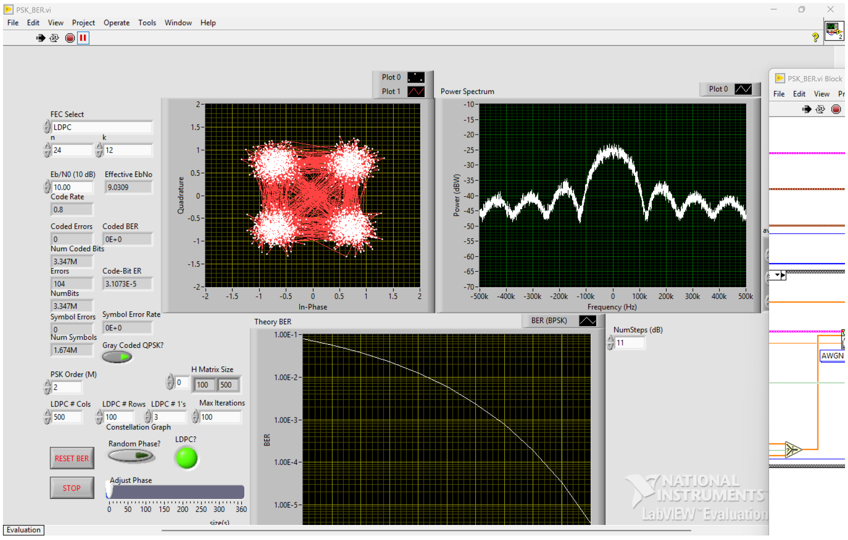


Figure 5: 2.4.a - LDPC Bit Error Rates under AWGN

5. - Repeat step 4 for a range of values of *Max Iterations*. Based on your results, how many iterations *should* we use for this particular LDPC code? Are you able to observe the point of diminshing returns?

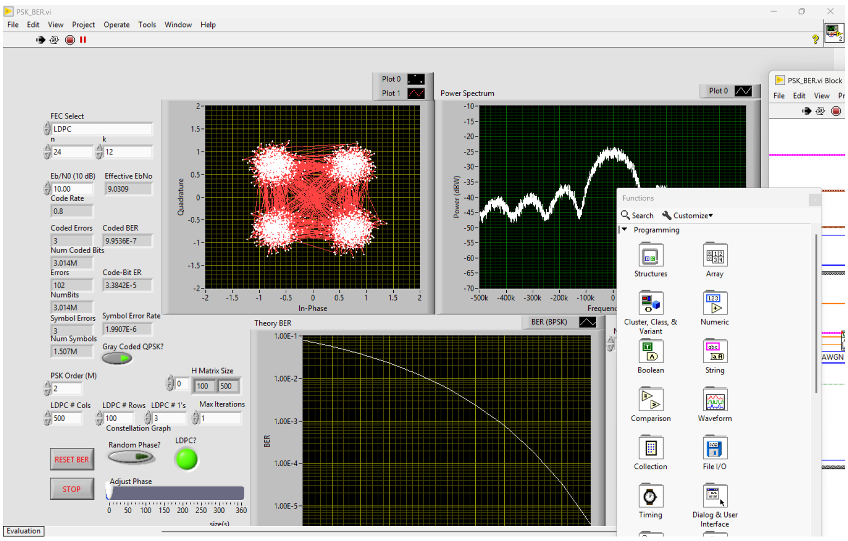


Figure 6: 2.5.1 - LDPC Bit Error Rates under AWGN, Max Iterations 1

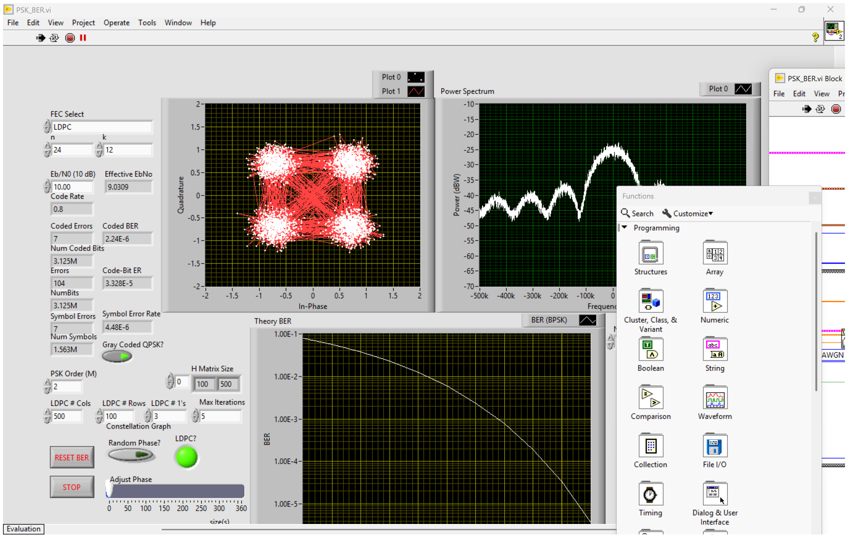


Figure 7: 2.5.2 - LDPC Bit Error Rates under AWGN, Max Iterations 5

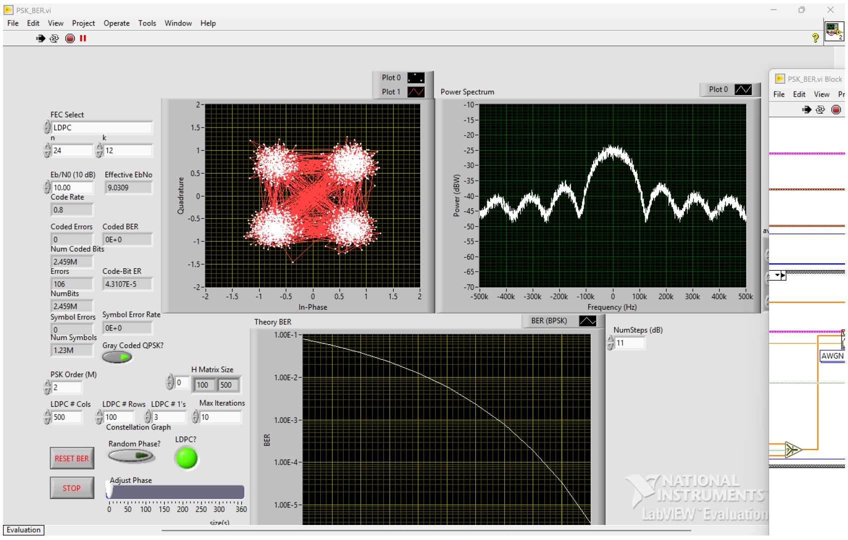


Figure 8: 2.5.3 - LDPC Bit Error Rates under AWGN, Max Iterations 10

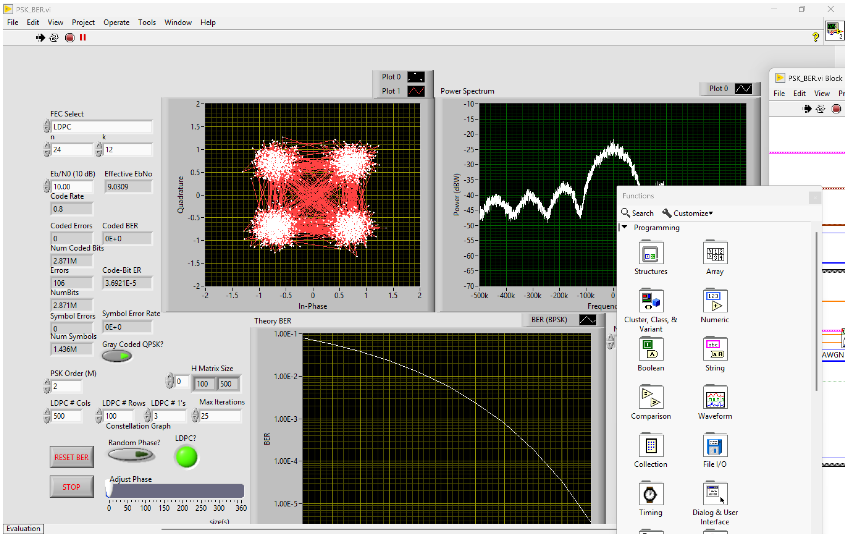


Figure 9: 2.5.4 - LDPC Bit Error Rates under AWGN, Max Iterations 25

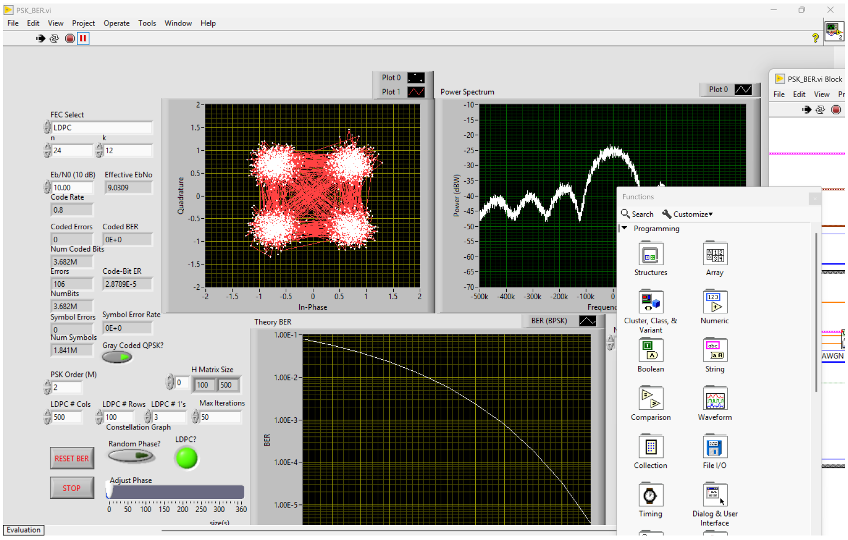


Figure 10: 2.5.5 - LDPC Bit Error Rates under AWGN, Max Iterations 50

Based on our testing, we can see that the Max Iterations appears to have diminishing returns past 10 iterations as the number of coded errors become less frequent. We note that the number of errors as time goes to infinity cannot definitively be 0, as we have only collected measurements after one minute.

3 Part 3: Higher Order Modulation and Gray Coding

2. - For a range of E_b/N_0 values from 0dB to 10dB (i.e. eleven 1dB steps), measure the Bit Error-Rate for the following settings:

a. "Gray Coded" QPSK

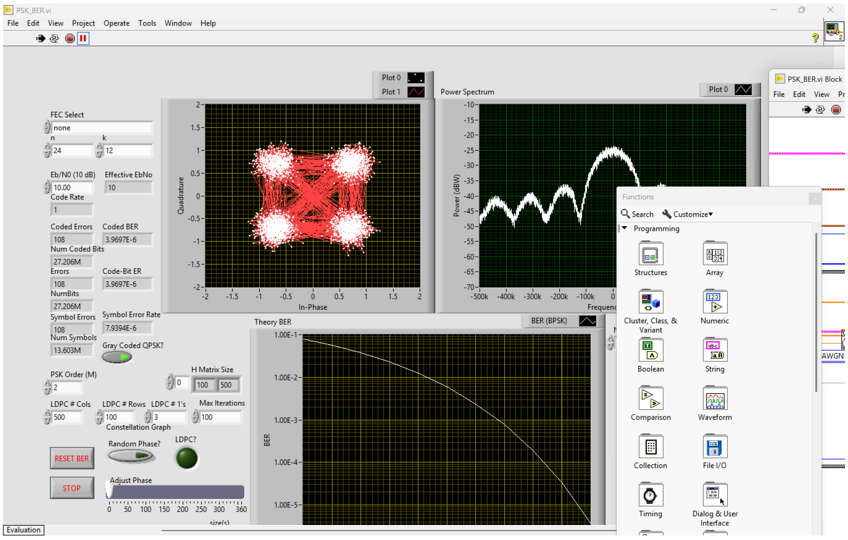


Figure 11: 3.2.a - "Gray Coded" QPSK Bit Error Rates under AWGN

b. "Non-Gray Coded" QPSK

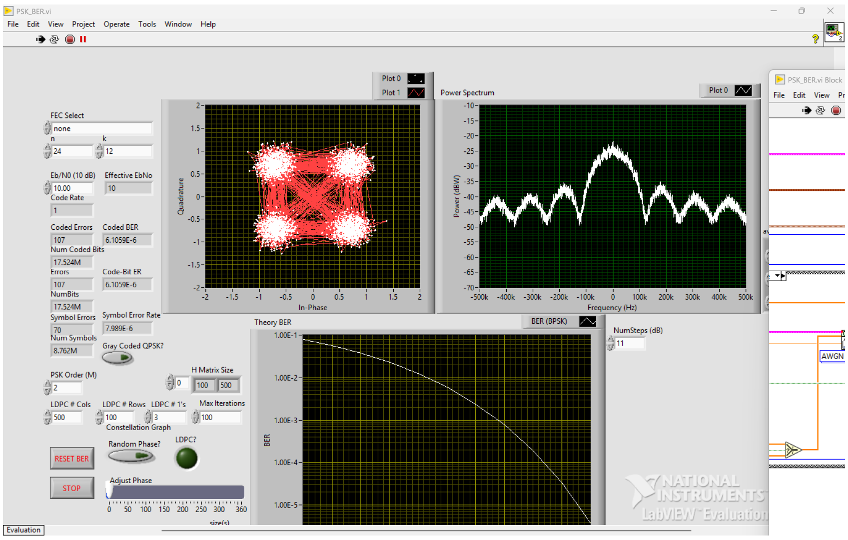


Figure 12: 3.2.b - "Non-Gray Coded" QPSK Bit Error Rates under AWGN

4 Part 4: Post Lab

1. - On the same graph, plot the Uncoded Bit Error Rate and Coded Bit Error Rate for the n-Repetition (3,1) and Hamming (7,4), and Golay (24,12) codes.

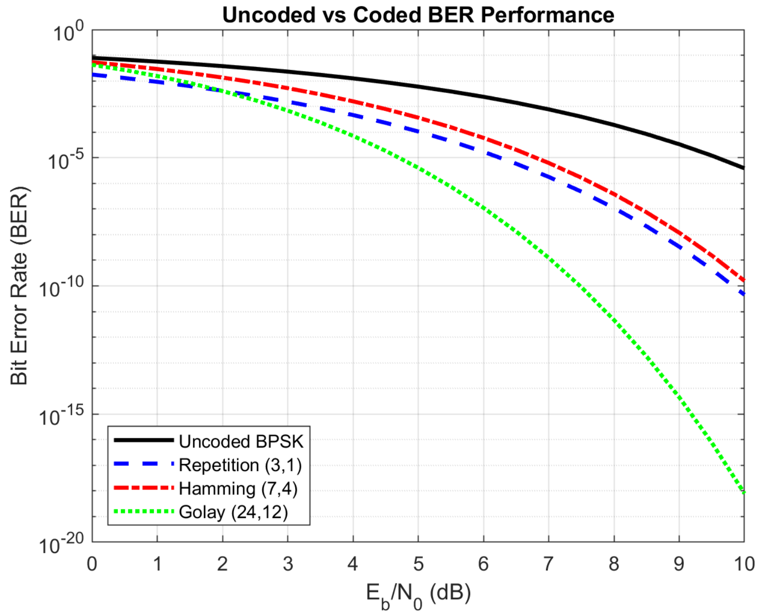


Figure 13: 4.1 - Uncoded Bit Error Rate vs Coded Bit Error Rate

Give an explanation for the cross-over point of the uncoded and coded ber curves.
As we saw during the prelab, the coded BER can actually perform worse than uncoded for low SNRs. This is because the act of coding intentionally introduces redundancy that can act as a source of error. When the errors are spread out across multiple bits and the channel is noisy, there is an increased chance of the decoder to receive an uncorrectable error pattern.

2. - On the same graph, plot the Uncoded Bit Error Rate and Coded Bit Error Rate for the LDPC code for range of values you used for *Max Iterations*. Are there any changes in coding gain for different values of *Max Iterations*?

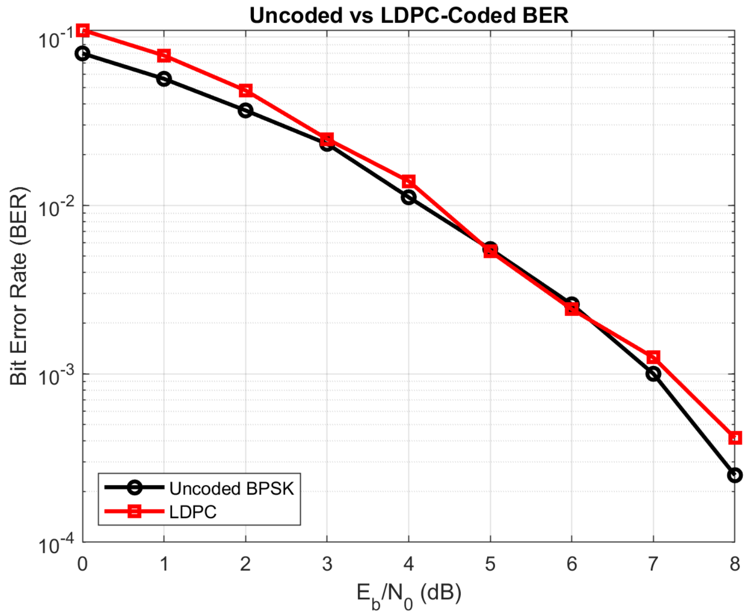


Figure 14: 4.2 - Uncoded Bit Error Rate vs LDPC Bit Error Rate

We can see that, similar to the other coded protocols that we explored earlier, the coding gain varies depending on the SNR. At low SNR (0-2dB), the uncoded BPSK actually performs better for the same reasons discussed earlier. However, at high SNR, the LDPC performs better than the uncoded BPSK, although the coding gain seems to plateau at very high SNR.

3. - On the same graph, plot the Uncoded Bit Error Rate for QPSK with both "gray" and "non gray" codes. Explain the difference between the two curves.

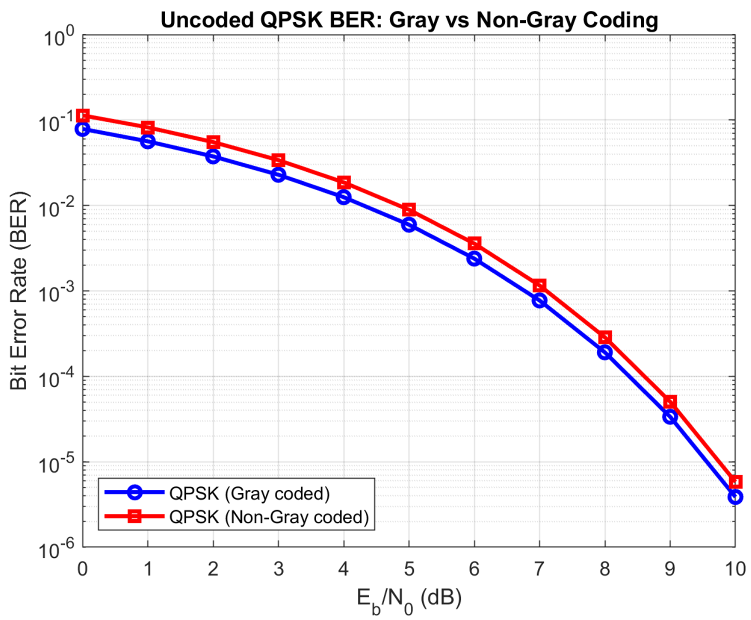


Figure 15: 4.3 - Gray-coded Bit Error Rate vs Non-Gray coded Bit Error Rate

When we use Gray-coding, we are able to minimize the number of bit errors since we know that each symbol is separated by exactly 1 bit. For non-Gray coding, each symbol may differ in distance from the constellation points. Thus, each symbol error can cause multiple bits to be wrong, even if the symbol error rate stays the same.

4. - What is the measured coding gain for the error correction codes from parts 1? Which error correction code is better? Why?
We can obtain the measured coding gain by measuring the vertical distance between each of the BER curves. At high SNRs, Golay outperforms all of the codes because of its ability to correct up to 3 errors per codeword.
5. - Compare the measured coding gains from part 1 to the expression for asymptotic coding gain G, where

$$G \approx R_c(t+1)$$

$$t = \textbf{floor}(\frac{d_{min}-1}{2})$$

Do they match? Why or why not?
No, the asymptotic coding gain G does not match our results because we are only measuring at low to moderate SNR and not in the asymptotic region (high SNR).
6. - What are the consequences of using an even value of n for an n-repetition (n,1) code?
If we have an even value for an n-repetition code, the decoder is faced with the possibility of having a tie and cannot decipher if a 1 or a 0 was transmitted. When n is odd, this never happens.

5 Part 5: Additional Questions

1. - Consider the 8-PSK constellation given by symbol values below:

s_0	s_1	s_2	s_3	s_4	s_5	s_6	s_7
1	$e^{j\frac{2\pi}{8}}$	$e^{j\frac{4\pi}{8}}$	$e^{j\frac{6\pi}{8}}$	$e^{j\frac{8\pi}{8}}$	$e^{j\frac{10\pi}{8}}$	$e^{j\frac{12\pi}{8}}$	$e^{j\frac{14\pi}{8}}$

Figure 3.1: 8-PSK Constellation Values

Can you construct a gray-coded symbol mapper table for this constellation? Justify your answer (i.e. if it's possible, create the symbol mapping table, otherwise explain why it's not possible).
Yes, it is possible to create a Gray-coded symbol mapper table for this constellation because it is uniformly spaced with a phase interval of $\pi/4$ and has exactly 3 nearest neighbors (one for each possible bit flip). We can use a 3-bit mapping since we have 2^3 symbols:

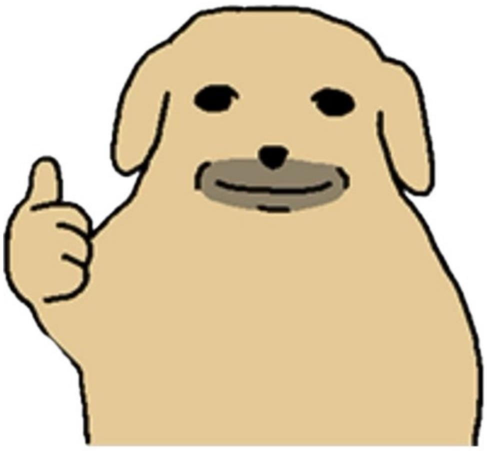
$$000 \rightarrow 001 \rightarrow 011 \rightarrow 010 \rightarrow 110 \rightarrow 111 \rightarrow 101 \rightarrow 100$$

2. - Consider the 8-QAM constellation given by symbol values below:

s_0	s_1	s_2	s_3	s_4	s_5	s_6	s_7
$-3+3j$	$-3-j$	$-1+j$	$-1-3j$	$1+3j$	$1-j$	$3+j$	$3-3j$

Figure 3.2: 8-QAM Constellation Values

Can you construct a gray-coded symbol mapper table for this constellation? Justify your answer (i.e. if it's possible, create the symbol mapping table, otherwise explain why it's not possible)
No, it is not possible to create a Gray-coded symbol mapper table for this constellation. In the above configuration, the symbols in the center of the constellation have 4 nearest neighbors, which is impossible to map with only 3 bits to work with.



feelsgoodman