İYTE - Department of Electrical and Electronics Engineering

EE 451 – Communication Systems II Project

Project: P18

Comparison of uncoded and coded linear block codes (Hamming) QPSK for channel coding

Final Report

30 December 2021

Group Members

Utku Acar – 250206062

Fatih Tuncay – 240206016

Egemen Uluçam – 240206034

Table of Contents

1. INTRODUCTION

The goal of this project is to show effects of different types of Hamming Coding on Bit Error Rate (BER) and compare BER of Hamming-coded sequences (after error correction) and the uncoded sequence in AWGN channel with using Quadrature Phase Shift Keying (QPSK) modulation and demodulation technique.

Hamming (7,4) and Hamming (15,11) are implemented and the range of SNR per bit for AWGN channel is from 0 to 15 in this project.

1.1 Hamming Code

Hamming Code is one of the linear error-correction coding methods. This method has been found by Richard W. Hamming while he is intending to correct errors in punch cards automatically. It has the capability that detect and correct 1-bit errors. The number of bit errors which can be detected by Hamming can be increased to 2 by adding 1 more parity bit to the codeword, but the correction capability does not change.

Hamming Coding has a block length of $n = 2^r-1$, where r is the number of parity bits, and $r \geq 2$, the message length of $k = 2^r-1 - r$ [1]. Parity bits are calculated by XORing specific data bits; the location of the data and parity bits are important. After calculations of parity bits, they should be located to the specific locations which are power of 2 in binary form (001,010,100). Venn Diagram for parity bit XORing operations for Hamming (7,4) can be seen in Figure 1 and for Hamming(15,11) can be seen in Figure 2.

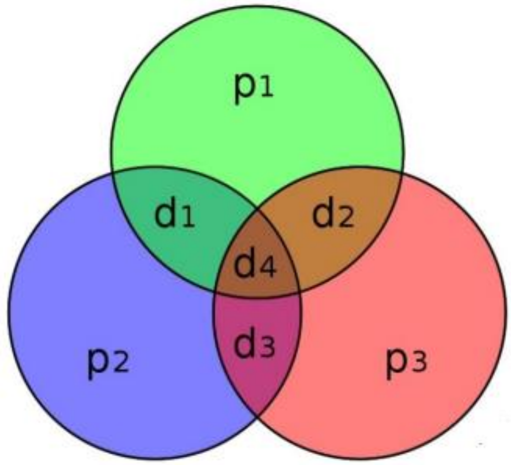

Figure 1: Hamming (7,4) parity XOR Venn Diagram [2]

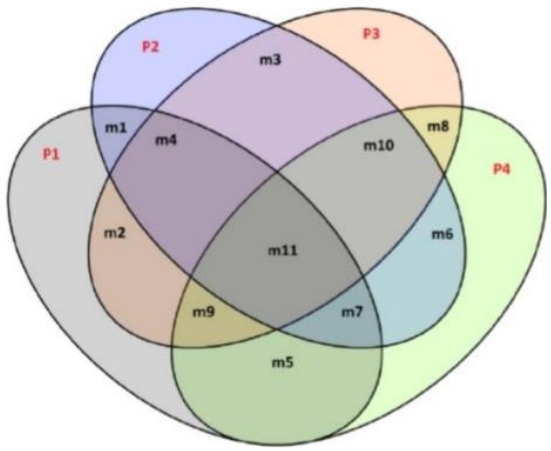

Figure 2: Hamming (15,11) parity XOR Venn Diagram [3]

The parity bit calculation equations for Hamming (7,4) are given below:

$p_1 = d_1 \oplus d_2 \oplus d_4$

$p_2 = d_1 \oplus d_3 \oplus d_4$

$p_3 = d_2 \oplus d_3 \oplus d_4$

Location of the bits for Hamming (7,4) can be seen in Table 1.

Location of Bits	1	2	3	4	5	6	7
Transmitted Bit	p_1	p_2	d_1	p_3	d_2	d_3	d_4

Table 1: Location of Bits for Hamming (7,4)

The parity bit calculation equations for Hamming (15,11) are given below:

$p_1= d_1 \oplus d_2 \oplus d_4 \oplus d_5 \oplus d_7 \oplus d_9 \oplus d_{11}$

$p_2= d_1 \oplus d_3 \oplus d_4 \oplus d_6 \oplus d_7 \oplus d_{10} \oplus d_{11}$

$p_3= d_2 \oplus d_3 \oplus d_4 \oplus d_8 \oplus d_9 \oplus d_{10} \oplus d_{11}$

$p_4= d_5 \oplus d_6 \oplus d_7 \oplus d_8 \oplus d_9 \oplus d_{10} \oplus d_{11}$

Location of the bits for Hamming (15,11) can be seen in Table 2.

Number of bits	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Transmitted bits	p_1	p_2	d_1	p_3	d_2	d_3	d_4	p_4	d_5	d_6	d_7	d_8	d_9	d_{10}	d_{11}

Table 2: Location of Bits for Hamming (15,11)

2. PROJECT DESIGN AND MODELLING

In our project we compare uncoded and hamming coded versions of a bit sequence with Bit Error Rate (BER) performance (Figure 3). To do that after generating a random N-bit sequence, we parse it into 4 and 11 pieces. We use these pieces in hamming coding (7,4) and (15,11). To obtain Hamming (7,4) we add 3 redundant parity bits to the pieces with four bits, and to obtain Hamming (15,11) we add 4 redundant parity bits to the pieces with eleven bits. After the encoding part, all three-bit sequences (Uncoded, Hamming (7,4) and Hamming (15,11)) pass through Quadrature Phase Shift Keying (QPSK) modulation. In this part two bits are transferred into one symbol, with this data rate being doubled. More detail about QPSK will be given at QPSK Modulation - QPSK Demodulation part. After modulation, the bit sequences are transmitted with additive white gaussian noise (AWGN) from 0 SNR to 15 SNR added as channel effect. The noisy bit sequences are demodulated with Euclidean distance. Then the demodulated noisy bit sequences are parsed into 7 and 15 according to be hamming encoded or not. From these pieces 1-bit error correction is done with parity bits. After error correction data bits are extracted from bit sequences with redundant bits. Error correction and Hamming decoding will be explained with more details at Error Correction and Hamming Decoding part. In the last part, to do BER analysis we check all three received bit sequences with the original generated bit sequence and compare the results.

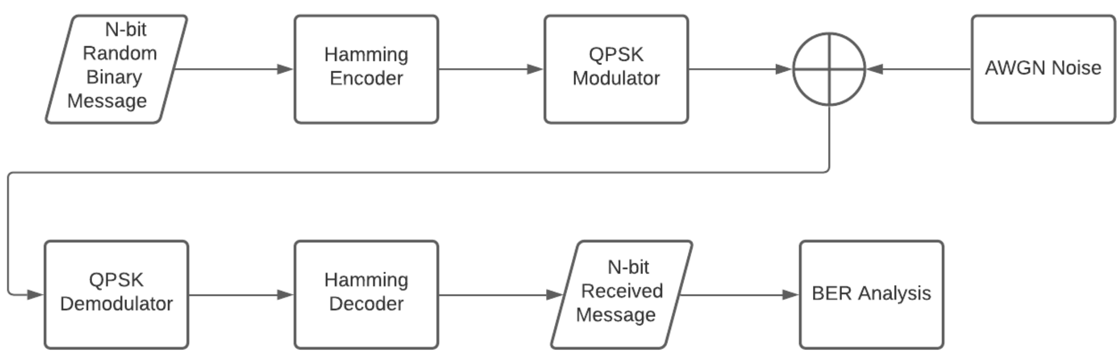

Figure 3: Block Diagram of the Project

2.1 QPSK Modulation - QPSK Demodulation

Phase shift keying is a digital modulation process that data can represent by changing the phase of a carrier wave. There are different phase shift keying types based on numbers of phases for using areas. The simplest phase shift keying is BPSK. One bit can be transferred with one symbol with 2 phase values, that 180-degree phase difference is used. Quadrature Phase Shift Keying (QPSK) Modulation is another method that two bits can be transferred with one symbol, and it has 4 phase values that 90 degrees phase difference is used. According to Table 3, there are 4 cases as 00, 01, 11, 10 depending on phase-shift degree. In this Project QPSK modulation is used instead of BPSK. QPSK has double the bit rate of BPSK for the same bandwidth, so bandwidth efficiency can be provided thanks to QPSK [4].

	m(t) Input Sequence	QPSK Modulated Sequence
Case 1	00	0.5+0.5i
Case 2	01	-0.5+0.5i
Case 3	11	-0.5-0.5j
Case 4	10	0.5-0.5j

Table 3: QPSK Modulation Cases

The decisions of modulated sequence have been given in Figure 4 below:

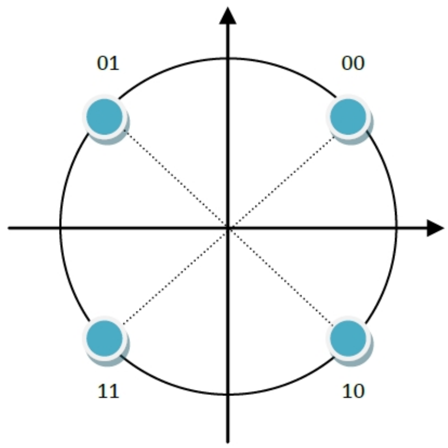

Figure 4: Location of Bits for Modulation Scheme [5]

At demodulation of QPSK, Euclidean distance method has been used to find and choose the shortest path between all paths. Possible paths for given examples are given below in color coded at Figure 5 and decision examples for each case can be seen in Table 4.

Cases (m=message)	Example Input (/w AWGN)	Demodulated Output
Case 1 (Real(m) >= 0 && Imj(m) >= 0)	0.7+j0.3	00
Case 2 (Real(m) < 0 && Imj(m) >= 0)	-0.22+j0.28	01
Case 3 (Real(m) <= 0 && Imj(m) < 0)	-0.65-j0.25	11
Case 4 (Real(m) >= 0 && Imj(m) < 0)	0.87-j0.48	10

Table 4: QPSK Demodulation Cases

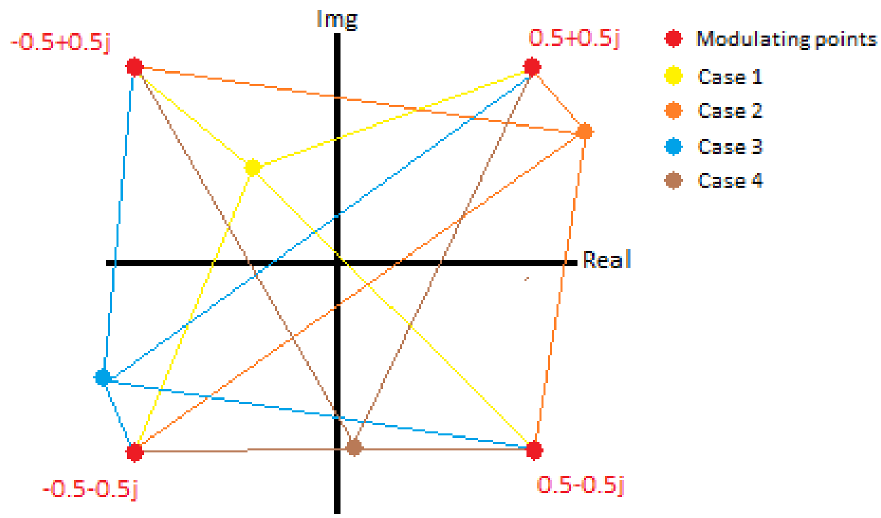

Figure 5: Example for QPSK Demodulation

2.2 Error Correction and Hamming Decoding

Error correction is done by checking parity bits that were calculated and sent in the encoding part. Controlling the parity bits is done in a similar way with encoding, XORing the same specific data bits and with themselves [6].

The parity bit checking equations for Hamming (7,4) are given below:

$p_1 = d_1 \oplus d_2 \oplus d_4 \oplus p_1$

$p_2 = d_1 \oplus d_3 \oplus d_4 \oplus p_2$

$p_3 = d_2 \oplus d_3 \oplus d_4 \oplus p_3$

The parity bit checking equations for Hamming (15,11) are given below:

$p_1 = d_1 \oplus d_2 \oplus d_4 \oplus d_5 \oplus d_7 \oplus d_9 \oplus d_{11} \oplus p_1$

$p_2 = d_1 \oplus d_3 \oplus d_4 \oplus d_6 \oplus d_7 \oplus d_{10} \oplus d_{11} \oplus p_2$

$p_3 = d_2 \oplus d_3 \oplus d_4 \oplus d_8 \oplus d_9 \oplus d_{10} \oplus d_{11} \oplus p_3$

$p_4 = d_5 \oplus d_6 \oplus d_7 \oplus d_8 \oplus d_9 \oplus d_{10} \oplus d_{11} \oplus p_4$

If no bits are corrupted, the new parity bits will be calculated as zero (0). But if one bit is corrupted, parity bits that include the errored bit will be calculated as one (1).

After new parity bits are calculated, detecting the errored bit is found by:

Detecting errored bit equation for Hamming (7,4):

$$(p3\ p2\ p1)_2 = \ (Errored\ bit)_{10} = p3 \times 2^2 + \ p2 \times 2^1 + \ p1 \times 2^0$$

Detecting errored bit equation for Hamming (15,11):

$$(p4\ p3\ p2\ p1)_2 = \ (Errored\ bit)_{10} = p4 \times 2^3 + \ p3 \times 2^2 + \ p2 \times 2^1 + p1 \times 2^0$$

The correction done by altering the detected errored bit. If there is more than one errored bit, parity bits can't detect the errored bits.

After the correction of the errored bit, extracting the data bits from the bit sequence is the reverse of encoding. Bits are ordered in a specific order in encoding. In decoding data bits are pulled according to that sequence. The new data sequences extracted from Hamming (7,4) (Table 1) and Hamming (15,11) (Table 2) can be seen in Table 5 and Table 6.

Location of Bits	1	2	3	4
Transmitted Bit	d_1	d_2	d_3	d_4

Table 5: Decoded Location of Bits for Hamming (7,4)

Number of bits	1	2	3	4	5	6	7	8	9	10	11
Transmitted bits	d_1	d_2	d_3	d_4	d_5	d_6	d_7	d_8	d_9	d_{10}	d_{11}

Table 6: Decoded Location of Bits for Hamming (15,11)

3. TEST RESULTS and DISCUSSION

When the SNR was lower than 7 dB, we are getting better results for uncoded sequence rather than both hamming-coded signals but after 9 dB we are getting much better results for both coded transmissions compared to uncoded transmissions. This can be seen in Figure 6 and Table 7. The reason why we get worse results while SNR is lower is when SNR gets lower, there could be more than 1 corrupted bit in codewords and Hamming cannot correct these extra corrupted bits. We normalized the Eb/N0 to make a fair comparison between coded and uncoded transmissions. We also determined coding gain to compare performance of uncoded and coded transmissions after the normalization process. Coding gain can be found by drawing a horizontal line at 10^{-6}, which is the minimum required BER for data communications, and then determining the points which correspond to 10^{-6} points for each coded and uncoded SNR values. Then we took the difference between the uncoded and each coded SNR values for the same BER. This difference is called Coding Gain. We found the coding gain of Hamming (7,4) as 0.5 dB and the coding gain of Hamming (15,11) as 1.5 dB. This explanation has been visualized in Figure 6.

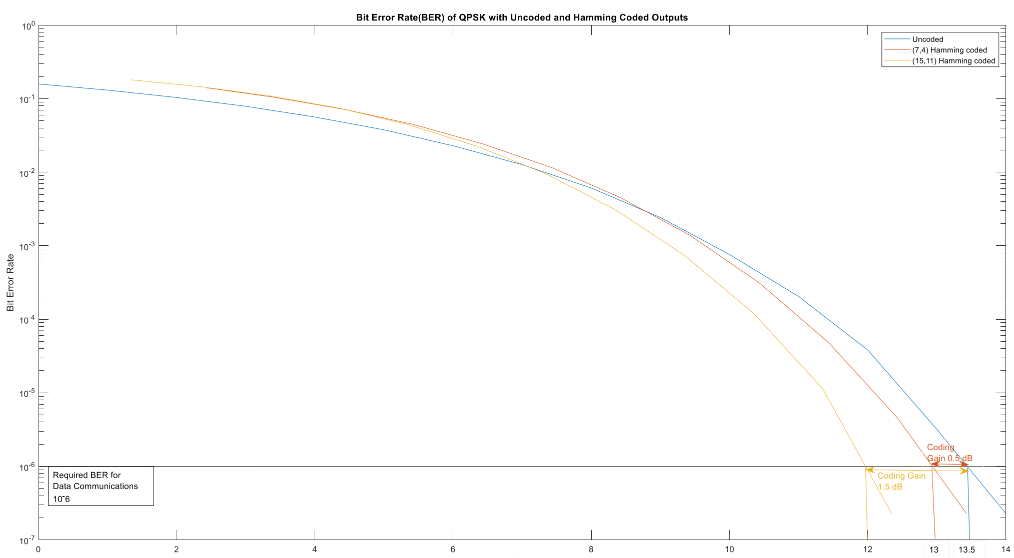

Figure 6: BER and Coding Gain of Uncoded and Coded Sequences

SNR (dB) (NORMALIZED) (UNCODED/ HAMMING (7,4)/ HAMMING (15,11))	UNCODED	HAMMING (7,4)	HAMMING (15,11)
0/2.43/1.35	0.1586	0.1408	0.1806
1/3.43/2.35	0.1310	0.1041	0.1457
2/4.43/3.35	0.1039	0.0714	0.1093
3/5.43/4.35	0.0789	0.0445	0.0745
4/6.43/5.35	0.0563	0.0242	0.0444
5/7.43/6.35	0.0377	0.0115	0.0225
6/8.43/7.35	0.0228	0.0044	0.0094
7/9.43/8.35	0.0126	0.0013	0.0030
8/10.43/9.35	0.0061	0.0003	0.0007
9/11.43/10.35	0.0024	$4.84\text{x}10^{-5}$	0.0001
10/12.43/11.35	0.0008	$4.55\text{x}10^{-6}$	$1.14\text{x}10^{-5}$
11/13.43/12.35	0.0002	$2.27\text{x}10^{-7}$	$2.27\text{x}10^{-7}$
12/14.43/13.35	$3.82\text{x}10^{-5}$	0	0
13/15.43/14.35	$3.18\text{x}10^{-6}$	0	0
14/16.43/15.35	$2.27\text{x}10^{-7}$	0	0
15/17.43/16.35	0	0	0

Table 7: Bit Error Rate (BER)

The bit error rate of Hamming (15,11) was higher than Hamming (7,4) because while we are using Hamming (7,4) we are having more bits in codeword due to higher data over parity bit ratio than Hamming (15,11). So, Hamming (7,4) requires more power in total than Hamming (15,11). This problem is solved by shifting/normalization of Eb/N0. After this normalization operation BER of Hamming (15,11) became lower than Hamming (7,4) which is consistent with theoretical data like in Table 7 . If we have enough power to use Hamming (7,4) we can choose it, but if we have excessive data to transmit we should use Hamming (15,11).

Also there is a way to improve the BER values; this way is called “Soft Decoding”. Soft Decoding differs from Hard Decoding by using Euclidean Distance instead of Hamming Distance. For example, if we use hard decoding we could have same lowest hamming distances with different codewords for same received message parts but if we use Euclidean distance instead, then we have lower probability that obtaining same lowest distance values this results with lower error rates[7].

4. CONCLUSION

To summarize, we started this project with the aim of comparing uncoded and coded linear block codes (Hamming) QPSK for channel coding. After our research, we learned that there are different types of Hamming codes. We worked on Hamming (7.4) and Hamming (15.11) and performed simulations. We observed how different hamming codes affect the BER result.

We learned that Hamming code can detect and correct 1-bit errors easily. However, we observed that Hamming code is insufficient in correcting multiple errors; we can see that in low SNR values because there are too many errors in low SNR values in QPSK applications. BER values for coded sequences at higher SNR values for Hamming (15,11) and Hamming (7,4) were not consistent with theoretical data, so we normalized SNR values to make a fair comparison between coded sequences. After this normalization we get consistent results with theoretical data.

Eventually, we got results which we were expecting. We did not use cyclic operations or possible codewords approaches. Because XOR operation makes this process without requiring cyclic operation or possible codewords approaches. Thus, we are not using neither hard decoding nor soft decoding, but since we use Euclidean distances in the demodulation part of the project, our results are more likely soft decoding.

5. REFERENCES

[1] Wikimedia Foundation, " Hamming code.," Wikipedia, 13-Dec-2021.[Online]. Available: https://en.wikipedia.org/wiki/Hamming_code. [Accessed 27 December 2021].

[2] PNGWing, " Hamming kodu hamming (7,4) Hata Tespiti ve Düzeltmesi Eşlik Biti,," PNGWing,[Online]. Available: https://www.pngwing.com/tr/free-png-srhop. [Accessed 27 December 2021].

[3] Mstafa, R. J., & Elleithy, K., "Fig. 2 Venn diagram of the hamming codes (15, 11).," ResearchGate, 11-Oct-2020.[Online]. Available: https://www.researchgate.net/figure/enn-diagram-of-the-hamming-codes-15-11_fig2_283461950. [Accessed 27 December 2021].

[4] All About Circuits, " Digital Phase Modulation: BPSK, QPSK, DQPSK: Radio Frequency Modulation: Electronics Textbook.," All About Circuits,.[Online]. Available: https://www.allaboutcircuits.com/textbook/radio-frequency-analysis-design/radio-frequency-modulation/digital-phase-modulation-bpsk-qpsk-dqpsk/. [Accessed 27 December 2021].

[5] Delta, " PSK modulation and its types.," Delta,.[Online]. Available: https://shopdelta.eu/print.php?page=portal%2Fdesc_page&id=986. [Accessed 27 December 2021].

[6] Pandey, H, " Hamming code in Computer Network..," GeeksforGeeks,4-May-2020.[Online]. Available: https://www.geeksforgeeks.org/hamming-code-in-computer-network/. [Accessed 27 December 2021].

[7] N. Den, "Hard and Soft Decision Decoding," *Hard and soft decision decoding*, 03-Jan-2019. [Online]. Available: https://www.tutorialspoint.com/hard-and-soft-decision-decoding. [Accessed: 27-Dec-2021].