LDPC performance in IEEE802.16e and Telemetry systems

Sumalatha Jatoth

A Thesis Submitted to

Indian Institute of Technology Hyderabad

In Partial Fulfillment of the Requirements for

The Degree of Master of Technology

भारतीय प्रौद्योगिकी संस्थान हैदराबाद

Indian Institute of Technology Hyderabad

Department of Electrical Engineering

June 2016

Declaration

I declare that this written submission represents my ideas in my own words, and where ideas or words of others have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be a cause for disciplinary action by the Institute and can also evoke penal action from the sources that have thus not been properly cited, or from whom proper permission has not been taken when needed.

Sumalatha

(Signature)

SUMALATHA

(Sumalatha Jatoth)

EE13M1031

(Roll No.)

Approval Sheet

This Thesis entitled LDPC performance in IEEE802.16e and Telemetry systems by Sumalatha Jatoth is approved for the degree of Master of Technology from IIT Hyderabad

(Dr Sumohana Channapaya) Examiner
Dept. of Electrical Eng
IITH

(Dr Abhinav Kumar) Examiner
Dept. of Electrical Eng
IITH

(Dr. G.V.V.Sharma) Adviser
Dept. of Electrical Eng
IITH

Acknowledgements

I would like to thank Dr. G.V.V. sharma and friends for their constant help and support during this thesis work. They always gave me valuable comments during this work.Finally, I would like to thank my family and friends who have been always there for me. Without them, this work had not been possible. And thanks to all people who, as much as they possibly could, have supported and guided me through this thesis.

Dedication

I dedicate this thesis to my family and friends without them i wouldn't be here.

Abstract

In this thesis there is a detailed note on LDPC codes history and types. There are many standards such as IEEE802.16e, IEEE802.11ad, IEEE802.11n, Telemetry systems etc. use LDPC codec for their implementation. There are simulation models for the physical layers of IEEE802.16e and Telemetry system are developed. Both the simulated models have been analyzed as per the standard specified rates and code sizes.For IEEE802.16e the codeword length 'N' varies from 576 to 2304 with six different data rates.In this standard physical layer implementation different modulation schemes are considered such as QPSK, 16-QAM and 64-QAM. Telemetry system is been analyzed for message block lengths of 1024, 4096 with three different rates specified. In this standard physical layer implementation the modulation schemes considered are QPSK and 16-QAM. The decoder used for both standards is Bit-flipping algorithm.

Contents

Chapter 1

Introduction to LDPC

Low-density parity-check (LDPC)[2] codes are forward error-correction codes, first proposed in the 1962 PhD thesis of Gallager at MIT, but were not used due to practical constraints. LDPC codes are rediscovered later by McKay and Neal.Design techniques for LDPC codes exist which enable the construction of codes which approach the Shannons capacity to within hundredths of a decibel.LDPC codes have already been adopted in satellite-based digital video broadcasting, long-haul optical communication standards, IEEE wireless local area network standards.

The LDPC code is a linear block code. This type of code maps a block of k information bits together with a codeword (or codeblock) of n bits.LDPC codes are codes with parity-check matrices that contain only a very small number of non-zero entries. It is the sparseness of H which guarantees both a decoding complexity which increases only linearly with the code length and a minimum distance which also increases linearly with the code length.

An LDPC code parity-check matrix is called (W_c, W_r) regular if each code bit is contained in a fixed number,W_c, of parity checks and each parity-check equation contains a fixed number, W_r, of code bits

For an irregular parity-check matrix we designate the fraction of columns of weight i by v_i and the fraction of rows of weight i by h_i. Collectively the set v and h is called the degree distribution of the code.

A regular LDPC code will have,

$$m.W_r = n.W_c$$

ones in its parity-check matrix. Similarly, for an irregular code

$$m\left(\sum_i h_i.i\right) = n\left(\sum_i v_i.i\right)$$

1.1 Different types of LDPC construction:

1.1.1 Gallager's LDPC

The original LDPC codes presented by Gallager are regular and defined by a banded structure in **H**. The rows of Gallagers parity-check matrices are divided into W_c sets with M/W_c rows in each set. The first set of rows contains W_r consecutive ones ordered from left to right across the columns.

Every other set of rows is a randomly chosen column permutation of this first set. Consequently every column of **H** has a 1 entry once in every one of the W_c sets.

A length 12(3,4) regular Gallager parity-check matrix is:

$$H = \left[\begin{array}{cccccccccccc} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ \hline 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ \hline 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \end{array}\right].$$

1.1.2 MacKay and Neal LDPC

Another common construction for LDPC codes is a method proposed by MacKay and Neal. In this method columns of H are added one column at a time from left to right. The weight of each column is chosen to obtain the correct bit degree distribution and the location of the non-zero entries in each column chosen randomly from those rows which are not yet full. If at any point there are rows with more positions unfilled then there are columns remaining to be added, the row degree distributions for H will not be exact. The process can be started again or back tracked by a few columns, until the correct row degrees are obtained.

A length 12(3,4) regular MacKay and Neal parity-check matrix is:

$$H = \left[\begin{array}{cccccccccccc} 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \end{array}\right]$$

1.1.3 Repeat-Accumulate LDPC construction:

A repeat-accumulate (RA) code is an LDPC code with an upper triangular form already built into the parity-check matrix during the code design. An mXn RA code parity-check matrix **H** has two parts:

$$H = [H_1 : H_2]$$

where H_2 is an mXm matrix with the form:

$$H_2 = \begin{bmatrix} 1 & 0 & 0 & & 0 & 0 & 0 \\ 1 & 1 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 1 & 1 & & 0 & 0 & 0 \\ & \vdots & & \ddots & & \vdots & \\ 0 & 0 & 0 & & 1 & 0 & 0 \\ 0 & 0 & 0 & \cdots & 1 & 1 & 0 \\ 0 & 0 & 0 & & 0 & 1 & 1 \end{bmatrix}$$

The parity-check matrix of an RA code is called *(q, a)*-regular if the weight of all the rows H_1 of are the same, a, and the weight of all the columns of H_1 are the same, q. Note that a regular RA parity-check matrix has columns of weight 2, and one column of weight 1, in H_2 and so is not regular in the sense of (j, r)-regular LDPC codes. An irregular RA code will have an irregular column weight distribution in H_1, with H_2 the same as for a regular code.

A(3,2)-regular RA parity-check matrix for length 10 rate $\frac{2}{5}$ code is:

$$H = \begin{bmatrix} 1 & . & 1 & . & 1 & . & . & . & . & . \\ . & 1 & . & 1 & 1 & 1 & . & . & . & . \\ 1 & 1 & . & . & . & 1 & 1 & . & . & . \\ . & . & 1 & 1 & . & . & 1 & 1 & . & . \\ 1 & . & 1 & . & . & . & . & 1 & 1 & . \\ . & 1 & . & 1 & . & . & . & . & 1 & 1 \end{bmatrix}$$

1.1.4 Quasi-cyclic codes

A code is quasi-cyclic if for any cyclic shift of a codeword by c places the resulting word is also a codeword, and so a cyclic code is a quasi-cyclic code with c = 1. The simplest quasi-cyclic codes are row-circulant codes which are described by a parity-check matrix

$$H = [A_1, A_2, .., A_l];$$

where $A_1, ..., A_l$ are binary vxv circulant matrices.

Provided that one of the circulant matrices is invertible (say A_l) the generator matrix for the code can be constructed in systematic form

$$G = \begin{bmatrix} & (A_l^{-1}A_1)^T \\ I_{v(l-1)} & (A_l^{-1}A_2)^T \\ & \vdots \\ & (A_l^{-1}A_{l-1})^T \end{bmatrix},$$

resulting in a quasi-cyclic code of length *vl* and dimension *v(l 1)*. As one of the circulant matrices is invertible, the construction of the generator matrix in this way necessarily leads to a full rank **H**.

1.1.5 Block circulant quasi-cyclic codes

More general quasi-cyclic codes are the block circulant codes. The parity check matrix of a block circulant quasi-cyclic LDPC code is:

$$\begin{bmatrix} I_p & I_p & I_p & \dots & I_p \\ I_p & I_p(p_{1,1}) & I_p(p_{1,2}) & \dots & I_p(p_{1,w_r}) \\ \vdots & & & \ddots & \vdots \\ I_p & I_p(p_{w_c-1,1}) & I_p(p_{w_c-1,2}) & \dots & I_p(p_{w_c-1,w_r-1}) \end{bmatrix},$$

where I_p represents the pxp identity matrix and $I_p(p_{i,j})$ represents the circulant shift of the identity matrix by $r + p_{i,j}(modp)$ columns to the right which give the matrix with the r-th row having a one in the $(r+p_{i,j}modp)$-th column. Block circulant LDPC codes can have better minimum distances and girths than row-circulant codes.

1.2 Encoding

Generator matrix for a code with parity-check matrix **H** can be found by performing Gauss-Jordan elimination on **H** to obtain it in the form

$$H = [A, I_{n-k}]$$

Where **A** is a $(nk)Xk$ binary matrix and I_{nk} is the size $n \;\; k$ identity matrix. The generator matrix is then

$$G = [I_k, A^T]$$

For example let **H** be parity-check matrix of length n=10 and rate=$\frac{1}{2}$ is

$$H = \begin{bmatrix} 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix}$$

The matrix **H** is applied elementary row operations in $\mathbf{GF(2)}$, and is reduced to row-echelon form.

$$H_{rr} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 \end{bmatrix}$$

Finally by using column permutations the matrix is converted into standard format.

$$H_{std} = \begin{bmatrix} 0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Finally, a generator **G** for the code with parity-check matrices H_{std} and **H** is given by

$$G = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 \end{bmatrix}$$

And the codeword can be now generated from the **G** matrix.

$$c = uG$$

where c is the codeword generated and u is the input bit stream.

Chapter 2

LDPC Performance in IEEE802.16e(WiMAX)

2.1 Introduction to WiMax

WiMax (Worldwide Interoperability for Microwave Access) is a wireless broadband technology, which supports point to multi-point (PMP) broadband wireless access.WiMax is basically a new shorthand term for IEEE Standard 802.16[1], which was designed to support the European standards.The IEEE wireless standard has a range of up to 30 miles, and can deliver broadband at around 75 megabits per second.The original version of the standard on which WiMAX is based (IEEE 802.16) specified a physical layer operating in the 10 to 66 GHz range. 802.16a, updated in 2004 to 802.16-2004, added specifications for the 2 to 11 GHz range.There number of applications of WiMax including broadband connections, cellular backhaul, hotspots, etc. It is similar to Wi-Fi, but it can enable usage at much greater distances

2.2 Channel coding in Wimax

Channel coding procedures including randomization, FEC encoding, and modulation.The physical layer design of WiMax is as shown in the figure below:

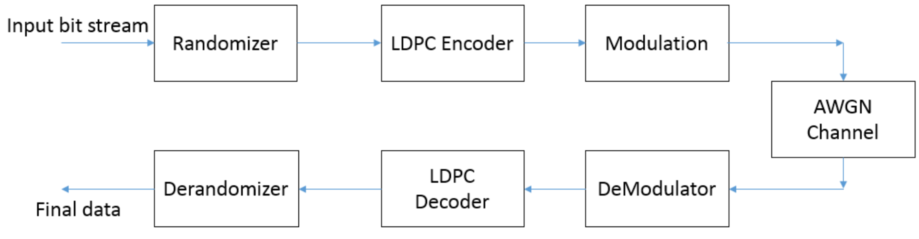

Figure 2.1: Physical layer of IEEE802.16e

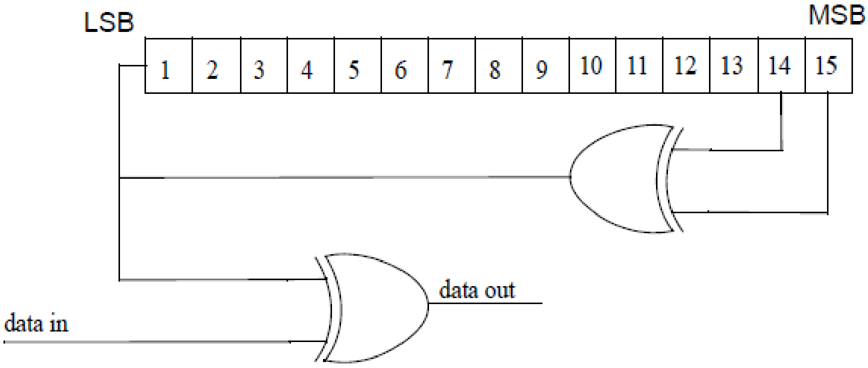

Figure 2.2: Randomizer

2.2.1 Randomizer

The raw data is randomized before getting encoded.The randomizer is shown in the figure below. The randomizer is initialized with the vector [LSB] 0 1 1 0 1 1 1 0 0 0 1 0 1 0 1 [MSB]

2.2.2 LDPC encoder

WiMax 802.16e LDPC code supports six different code classes with four different code rates. All six code classes have same general matrix structure that allows linear encoding scheme which simplifies decoding process significantly.

The parity-check matrix given has 24 colums and(1-rate)*24 rows with each entry describing ZxZ submatix, which is either a permuted identity matrix or a zero matrix.Z varies from 24 to 96 in steps of supporting variable codeword sizes ranging from 576-2304.

$$Codewordlength = Z * 24$$

The parity check matrices of LDPC codes for all six code classes are given and they are examples for RA-LDPC codes.

The given model parity matrices given are for codeword length 2304.This parity check matrix has to be converted to required model matrix according to required codeword length which depends on the expansion factor Z.From expanding the model matrix in binary format we get out sparsely constructed **H**.Since the LDPC code is RA-LDPC code we can generate systematic generator **G** from sparsely spread parity check matrix. Each 0 in the model matrix implies ZxZ identity matrix. -1 implies ZxZ zero matrix.

The Model matrices for N=2304 for all different rates are as follows:

There are two code classes for rate $\frac{2}{3}$ i.e $\frac{2}{3}$A- for highly regular codes, $\frac{2}{3}$B for semi regular codes. Rate $\frac{3}{4}$A and B differ mainly in the maximum variable node degree to be supported.

2.2.3 H matrix generation from model matrix

- Get a model matrix for the given rate.
- For the required codeword length get the expansion factor Zf as per the table given.

Rate 1/2:

-1	94	73	-1	-1	-1	-1	-1	55	83	-1	-1	7	0	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	27	-1	-1	-1	22	79	9	-1	-1	-1	12	-1	0	0	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	24	22	81	-1	33	-1	-1	-1	0	-1	-1	0	0	-1	-1	-1	-1	-1	-1	-1	-1
61	-1	47	-1	-1	-1	-1	-1	65	25	-1	-1	-1	-1	-1	0	0	-1	-1	-1	-1	-1	-1	-1
-1	-1	39	-1	-1	-1	84	-1	-1	41	72	-1	-1	-1	-1	-1	0	0	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	46	40	-1	82	-1	-1	-1	79	0	-1	-1	-1	-1	0	0	-1	-1	-1	-1	-1
-1	-1	95	53	-1	-1	-1	-1	-1	14	18	-1	-1	-1	-1	-1	-1	-1	0	0	-1	-1	-1	-1
-1	11	73	-1	-1	-1	2	-1	-1	47	-1	-1	-1	-1	-1	-1	-1	-1	-1	0	0	-1	-1	-1
12	-1	-1	-1	83	24	-1	43	-1	-1	-1	51	-1	-1	-1	-1	-1	-1	-1	-1	0	0	-1	-1
-1	-1	-1	-1	-1	94	-1	59	-1	-1	70	72	-1	-1	-1	-1	-1	-1	-1	-1	-1	0	0	-1
-1	-1	7	65	-1	-1	-1	-1	39	49	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	0	0
43	-1	-1	-1	-1	66	-1	41	-1	-1	-1	26	7	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	0

Figure 2.3: Parity matrix for N=2304 with rate 1/2

Rate 2/3 A code:

3	0	-1	-1	2	0	-1	3	7	-1	1	1	-1	-1	-1	-1	1	0	-1	-1	-1	-1	-1	-1
-1	-1	1	-1	36	-1	-1	34	10	-1	-1	18	2	-1	3	0	-1	0	0	-1	-1	-1	-1	-1
-1	-1	12	2	-1	15	-1	40	-1	3	-1	15	-1	2	13	-1	-1	-1	0	0	-1	-1	-1	-1
-1	-1	19	24	-1	3	0	-1	6	-1	17	-1	-1	-1	8	39	-1	-1	-1	0	0	-1	-1	-1
20	-1	6	-1	-1	10	29	-1	-1	28	-1	14	-1	38	-1	-1	0	-1	-1	-1	0	0	-1	-1
-1	-1	10	-1	28	20	-1	-1	8	-1	36	-1	9	-1	21	45	-1	-1	-1	-1	-1	0	0	-1
35	25	-1	37	-1	21	-1	-1	5	-1	-1	0	-1	4	20	-1	-1	-1	-1	-1	-1	-1	0	0
-1	6	6	-1	-1	-1	4	-1	14	30	-1	3	36	-1	14	-1	1	-1	-1	-1	-1	-1	-1	0

Figure 2.4: Parity matrix for N=2304 with rate 2/3a

Rate 2/3 B code:

2	-1	19	-1	47	-1	48	-1	36	-1	82	-1	47	-1	15	-1	95	0	-1	-1	-1	-1	-1	-1
-1	69	-1	88	-1	33	-1	3	-1	16	-1	37	-1	40	-1	48	-1	0	0	-1	-1	-1	-1	-1
10	-1	86	-1	62	-1	28	-1	85	-1	16	-1	34	-1	73	-1	-1	-1	0	0	-1	-1	-1	-1
-1	28	-1	32	-1	81	-1	27	-1	88	-1	5	-1	56	-1	37	-1	-1	-1	0	0	-1	-1	-1
23	-1	29	-1	15	-1	30	-1	66	-1	24	-1	50	-1	62	-1	-1	-1	-1	-1	0	0	-1	-1
-1	30	-1	65	-1	54	-1	14	-1	0	-1	30	-1	74	-1	0	-1	-1	-1	-1	-1	0	0	-1
32	-1	0	-1	15	-1	56	-1	85	-1	5	-1	6	-1	52	-1	0	-1	-1	-1	-1	-1	0	0
-1	0	-1	47	-1	13	-1	61	-1	84	-1	55	-1	78	-1	41	95	-1	-1	-1	-1	-1	-1	0

Figure 2.5: Parity matrix for N=2304 with rate 2/3b

Rate 3/4 A code:

6	38	3	93	-1	-1	-1	30	70	-1	86	-1	37	38	4	11	-1	46	48	0	-1	-1	-1	-1
62	94	19	84	-1	92	78	-1	15	-1	-1	92	-1	45	24	32	30	-1	-1	0	0	-1	-1	-1
71	-1	55	-1	12	66	45	79	-1	78	-1	-1	10	-1	22	55	70	82	-1	-1	0	0	-1	-1
38	61	-1	66	9	73	47	64	-1	39	61	43	-1	-1	-1	-1	95	32	0	-1	-1	0	0	-1
-1	-1	-1	-1	32	52	55	80	95	22	6	51	24	90	44	20	-1	-1	-1	-1	-1	-1	0	0
-1	63	31	88	20	-1	-1	-1	6	40	56	16	71	53	-1	-1	27	26	48	-1	-1	-1	-1	0

Figure 2.6: Parity matrix for N=2304 with rate 3/4a

Rate 3/4 B code:

-1	81	-1	28	-1	-1	14	25	17	-1	-1	85	29	52	78	95	22	92	0	0	-1	-1	-1	-1
42	-1	14	68	32	-1	-1	-1	-1	70	43	11	36	40	33	57	38	24	-1	0	0	-1	-1	-1
-1	-1	20	-1	-1	63	39	-1	70	67	-1	38	4	72	47	29	60	5	80	-1	0	0	-1	-1
64	2	-1	-1	63	-1	-1	3	51	-1	81	15	94	9	85	36	14	19	-1	-1	-1	0	0	-1
-1	53	60	80	-1	26	75	-1	-1	-1	-1	86	77	1	3	72	60	25	-1	-1	-1	-1	0	0
77	-1	-1	-1	15	28	-1	35	-1	72	30	68	85	84	26	64	11	89	0	-1	-1	-1	-1	0

Figure 2.7: Parity matrix for N=2304 with rate 3/4b

Rate 5/6 code:

1	25	55	-1	47	4	-1	91	84	8	86	52	82	33	5	0	36	20	4	77	80	0	-1	-1
-1	6	-1	36	40	47	12	79	47	-1	41	21	12	71	14	72	0	44	49	0	0	0	0	-1
51	81	83	4	67	-1	21	-1	31	24	91	61	81	9	86	78	60	88	67	15	-1	-1	0	0
68	-1	50	15	-1	36	13	10	11	20	53	90	29	92	57	30	84	92	11	66	80	-1	-1	0

Figure 2.8: Parity matrix for N=2304 with rate 5/6

n (bits)	*n* (bytes)	*k* (bytes)			Number of subchannels		
		R = 1/2	R = 2/3	R = 3/4	QPSK	16QAM	64QAM
576	72	36	48	54	6	3	2
672	84	42	56	63	7		
768	96	48	64	72	8	4	
864	108	54	72	81	9		3
960	120	60	80	90	10	5	
1056	132	66	88	99	11		
1152	144	72	96	108	12	6	4
1248	156	78	104	117	13		
1344	168	84	112	126	14	7	
1440	180	90	120	135	15		5
1536	192	96	128	144	16	8	
1632	204	102	136	153	17		
1728	216	108	144	162	18	9	6
1824	228	114	152	171	19		
1920	240	120	160	180	20	10	
2016	252	126	168	189	21		7
2112	264	132	176	198	22	11	
2208	276	138	184	207	23		
2304	288	144	192	216	24	12	8

Figure 2.9: Table for Codeblock adjustment and Expansion factor

- Z0=96. since the model matrix is defined for N=2304.
- Now calculate each entry of the model matrix for required rate and codeword length by following steps
  - For code rates $\frac{1}{2}, \frac{3}{4} AandBcode, \frac{2}{3} Bcode, and \frac{5}{6} coderates$, the shift sizes $p(f, i, j)$ for a code size corresponding to expansion factor zf are derived from $p(i, j)$ by scaling $p(i, j)$ proportionally.

$$p(f,i,j) = \begin{cases} p(i,j), p(i,j) \leq 0 \\ \left\lfloor \frac{p(i,j) z_f}{z_0} \right\rfloor, p(i,j) > 0 \end{cases}$$

  - For code rate $\frac{2}{3}A$ model matrix **H** is calculated as

$$p(f,i,j) = \begin{cases} p(i,j), p(i,j) \leq 0 \\ mod(p(i,j), z_f), p(i,j) > 0 \end{cases}$$

- Now Convert the model matrix into binary parity check matrix by applying appropriate shifts as given in the model matrix.
- Now we got our parity check matrix but it is not in systematic form.

2.2.4 Systematic Generator matrix(kxn) construction from H(mxn)

1 The **H** we have is mxn parity check matrix.

2 Divide it into following matrices.

$$H = \begin{bmatrix} A & B & T \\ C & D & E \end{bmatrix}$$

3 A is of size $(m - z)Xk$ B is of size $(m - z)Xz$ T is of size $(m - z)X(m - z)$ C is of size zXk D is of size zXz E is of size $zX(m - z)$

4 Now compute the following

- $P_1' = ET^{-1}A + C$
- $P_2' = T^{-1}(A + BP_1')$
- $P = [P_1, P_2]$

5 Now we can write **G** as

$$G = [I_k, P]$$

The G we got is systematic Generator.
For N=576 and rate 1/2 G and H are

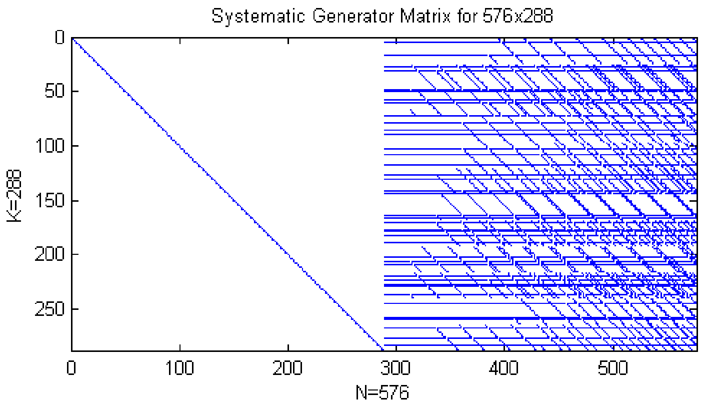

Figure 2.10: Generator matrix for N=2304 with rate 1/2

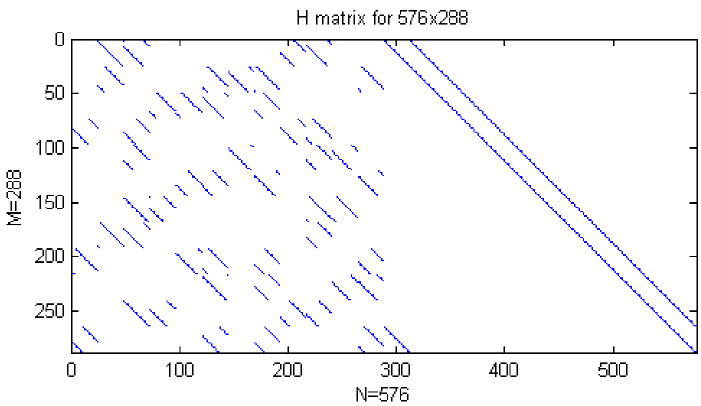

Figure 2.11: Parity matrix for N=2304 with rate 1/2

2.2.5 Modulator and AWGN channel

This block modulates the LDPC coded bits into corresponding modulation scheme. Here we are considering QPSK, 16-QAM and 64-QAM modulation schemes. The channel is just an additive white gaussian noise channel and the factor we are varying is signal to noise ratio(SNR). For different SNRs, the channel effect is different. Better the SNR, better to decode so that the received signal contains less errors.

2.2.6 LDPC Decoder

Hard decision decoder deals with the bit stream. This bit stream is obtained after demodulating the received signal. We consider bit flipping algorithm for hard decision decoding.

For Hard decision decoding, bit flipping algorithm is considered. This algorithm takes the demodu- lated bits from the demodulator. The steps involved in this algorithm are

- Compute the syndrome $r * H^T$
- If the syndrome vector is zero, the code word is without errors.
- For each bit , compute the unsatisfied parity checks.
- Flip the set of bits for which unsatisfied checks are more.
- Compute the syndrome again, if it satisfies stop. otherwise repeat upto certain no. of times and declare as decode failure.

2.2.7 Results

Simulations are done for all rates with **N**=576 and the BER plots are given below.Different Modulation schemes were considered for the simulation.Hard bit-flip decoder works well when compared to SPA.Time taken by Hard bit-flip is less. So we can conclude that Hard BF algorithm is efficient for the given LDPC encoder.

Comparision of Sum-product algorithm and Hard bit-flipping algorithm is also done and the result is as follows

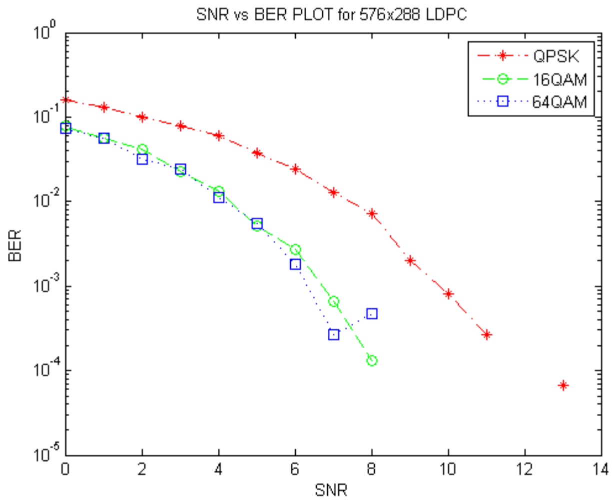

Figure 2.12: SNR vs BER for rate=1/2

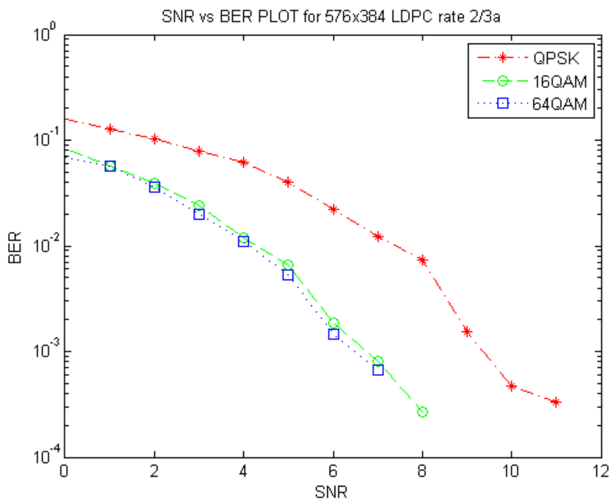

Figure 2.13: SNR vs BER for rate=2/3a

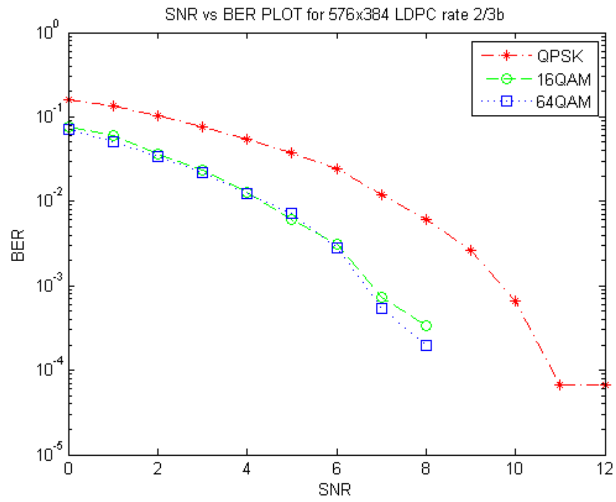

Figure 2.14: SNR vs BER for rate=2/3b

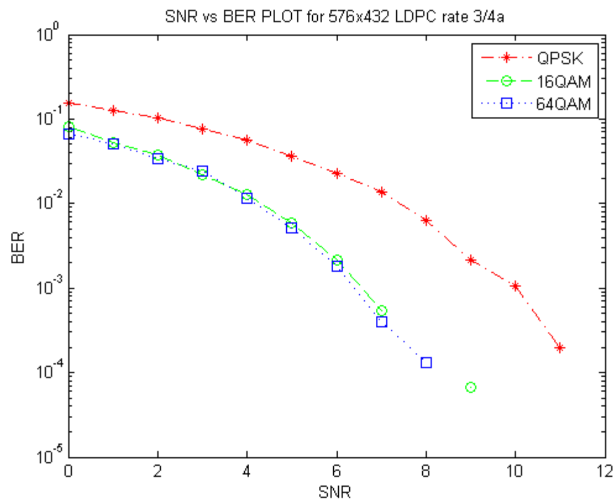

Figure 2.15: SNR vs BER for rate=3/4a

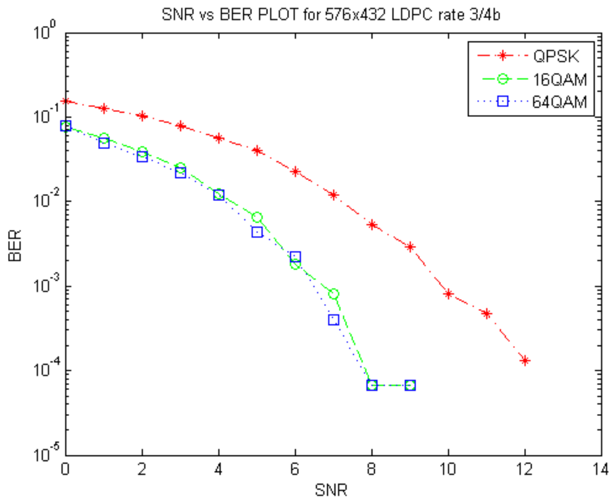

Figure 2.16: SNR vs BER for rate=3/4b

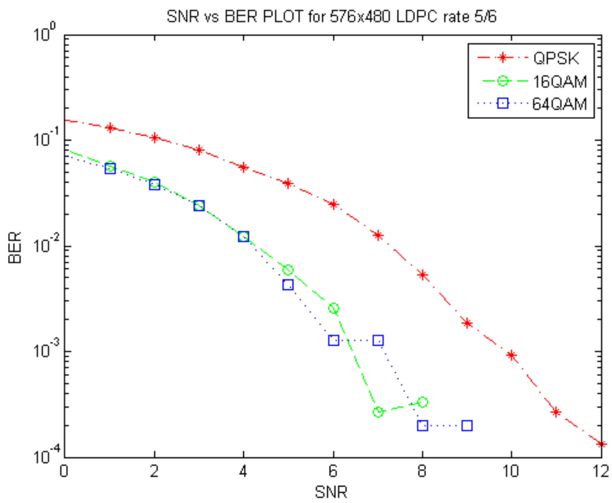

Figure 2.17: SNR vs BER for rate=5/6

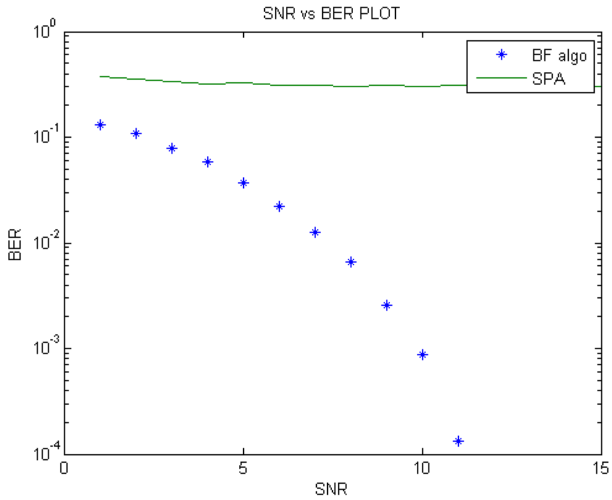

Figure 2.18: Comparision of BF and SPA

Chapter 3

Telemetry Systems

3.1 Introduction

Telemetry[3] is an automated communications process by which measurements and other data are collected at remote or inaccessible points and transmitted to receiving equipment for monitoring.It also encompasses data transferred over other media such as a telephone or computer network, optical link or other wired communications like phase line carriers. Many modern telemetry systems take advantage of the low cost and ubiquity of GSM networks by using SMS to receive and transmit telemetry data.

There are wide variety of application for telemetry and like any other tele communication fields these systems also have standards such as IRIG.

3.2 Modelling of Telemetry system

The Raw data is randomized and then given to LDPC encoder. In the given standard the input block length is 1024 or 4096. and data rates are $\frac{1}{2}, \frac{2}{3} and \frac{4}{5}$. After encoding the data is modulated with different modulation schemes such as QPSK and 16-QAM.Then data is passed through an AWGN channel and received at the receiver.

3.2.1 LDPC encoder

The LDPC codes presented are intended to decrease error probabilities in a primarily noisy transmission channel for use in the aeronautical mobile telemetry (AMT) test environment.

The LDPC code is a linear block code with options for *n,k*, where n is the length of the code block and textitk is the length of the information block. An LDPC code can be entirely defined by its parity check matrix, **H**. The $k \ X \ n$ generator matrix that is used to encode a linear block code can be derived from the parity check matrix through linear operations. Code rates, r, chosen for this AMT application are 1/2, 2/3, and 4/5. Information block sizes (k) are 1024 and 4096 bits. Given the code rate and information block sizes, codeword block sizes are calculated using $n = k/r$.

The LDPC used in telemetry sytem is QC-Block circulant LDPC.

The code block length and information block length for different rates is tabulated in next page

Information Block Length, k	Codeblock Length, n		
	Rate 1/2	Rate 2/3	Rate 4/5
1024	2048	1536	1280
4096	8192	6144	5120

Figure 3.1: Table for Codeblock length for different rates

Information Block Length, k	Submatrix size M		
	Rate 1/2	Rate 2/3	Rate 4/5
1024	512	256	128
4096	2048	1024	512

Figure 3.2: Table for Sub-matrix sizes M

The k X n generator matrix G shall be used to encode a linear block code. The matrix G can be derived from the parity check matrix H.

Parity matrix generation

For each n,k a parity check matrix **H** is constructed from size M X M submatrices.The sizes of M is given in the table below: The H matrices for each code rate are specified below. I_M is the M x M identity matrix and 0_M is the zero matrix.

Parity Check Matrices

$$H_{1/2} = \begin{bmatrix} 0_M & 0_M & I_M & 0_M & I_M \oplus \Pi_1 \\ I_M & I_M & 0_M & I_M & \Pi_2 \oplus \Pi_3 \oplus \Pi_4 \\ I_M & \Pi_5 \oplus \Pi_6 & 0_M & \Pi_7 \oplus \Pi_8 & I_M \end{bmatrix}$$

$$H_{2/3} = \begin{bmatrix} 0_M & 0_M & 0_M & 0_M & I_M & 0_M & I_M \oplus \Pi_1 \\ \Pi_9 \oplus \Pi_{10} \oplus \Pi_{11} & I_M & I_M & I_M & 0_M & I_M & \Pi_2 \oplus \Pi_3 \oplus \Pi_4 \\ I_M & \Pi_{12} \oplus \Pi_{13} \oplus \Pi_{14} & I_M & \Pi_5 \oplus \Pi_6 & 0_M & \Pi_7 \oplus \Pi_8 & I_M \end{bmatrix}$$

$$H_{4/5} = \left[\begin{array}{cccccc|c} 0_M & 0_M & 0_M & 0_M & 0_M & 0_M & \\ \Pi_{21} \oplus \Pi_{22} \oplus \Pi_{23} & I_M & \Pi_{15} \oplus \Pi_{16} \oplus \Pi_{17} & I_M & \Pi_9 \oplus \Pi_{10} \oplus \Pi_{11} & I_M & H_{1/2} \\ I_M & \Pi_{24} \oplus \Pi_{25} \oplus \Pi_{26} & I_M & \Pi_{18} \oplus \Pi_{19} \oplus \Pi_{20} & I_M & \Pi_{12} \oplus \Pi_{13} \oplus \Pi_{14} & \end{array}\right]$$

Permutation matrix Π_k has non-zero entries in row i and column entries are defined by $\Pi_k(i)$ for $i \in 0, 1, ...M-1$

$$\pi_k(i) = \frac{M}{4}\left(\left(\theta_k + \lfloor 4i/M \rfloor\right) \bmod 4\right) + \left(\phi_k\left(\lfloor 4i/M \rfloor\right) + i\right) \bmod \frac{M}{4}$$

whereθ_k and $\Phi_k(j)$ are defined for each submatrix size as follows

Code Rate =1/2, Information Block Size = 1024, M = 512

k	Θ_k	$\phi_k(0,M)$	$\phi_k(1,M)$	$\phi_k(2,M)$	$\phi_k(3,M)$
1	3	16	0	0	0
2	0	103	53	8	35
3	1	105	74	119	97
4	2	0	45	89	112
5	2	50	47	31	64
6	3	29	0	122	93
7	0	115	59	1	99
8	1	30	102	69	94

Code Rate =1/2, Information Block Size = 4096, M = 2048

k	Θ_k	$\phi_k(0,M)$	$\phi_k(1,M)$	$\phi_k(2,M)$	$\phi_k(3,M)$
1	3	108	0	0	0
2	0	126	375	219	312
3	1	238	436	16	503
4	2	481	350	263	388
5	2	96	260	415	48
6	3	28	84	403	7
7	0	59	318	184	185
8	1	225	382	279	328

Code Rate =2/3, Information Block Size = 1024, M = 256

k	Θ_k	$\phi_k(0,M)$	$\phi_k(1,M)$	$\phi_k(2,M)$	$\phi_k(3,M)$
1	3	59	0	0	0
2	0	18	32	46	44
3	1	52	21	45	51
4	2	23	36	27	12
5	2	11	30	48	15
6	3	7	29	37	12
7	0	22	44	41	4
8	1	25	29	13	7
9	0	27	39	9	2
10	1	30	14	49	30
11	2	43	22	36	53
12	0	14	15	10	23
13	2	46	48	11	29
14	3	62	55	18	37

Code Rate =2/3, Information Block Size = 4096, M = 1024

k	Θ_k	$\phi_k(0,M)$	$\phi_k(1,M)$	$\phi_k(2,M)$	$\phi_k(3,M)$
1	3	160	0	0	0
2	0	241	182	35	162
3	1	185	249	167	7
4	2	251	65	214	31
5	2	209	70	84	164
6	3	103	141	206	11
7	0	90	237	122	237
8	1	184	77	67	125
9	0	248	55	147	133
10	1	12	12	54	99
11	2	111	227	23	105
12	0	66	42	93	17
13	2	173	52	20	97
14	3	42	243	197	91

Code Rate =4/5, Information Block Size = 1024, M = 128

k	Θ_k	$\phi_k(0,M)$	$\phi_k(1,M)$	$\phi_k(2,M)$	$\phi_k(3,M)$
1	3	1	0	0	0
2	0	22	27	12	13
3	1	0	30	30	19
4	2	26	28	18	14
5	2	0	7	10	15
6	3	10	1	16	20
7	0	5	8	13	17
8	1	18	20	9	4
9	0	3	26	7	4
10	1	22	24	15	11
11	2	3	4	16	17
12	0	8	12	18	20
13	2	25	23	4	8
14	3	25	15	23	22
15	0	2	15	5	19
16	1	27	22	3	15
17	2	7	31	29	5
18	0	7	3	11	21
19	1	15	29	4	17
20	2	10	21	8	9
21	0	4	2	2	20
22	1	19	5	11	18
23	2	7	11	11	31
24	1	9	26	3	13
25	2	26	9	15	2
26	3	17	17	13	18

Code Rate =4/5, Information Block Size = 4096, M = 512

k	Θ_k	$\phi_k(0,M)$	$\phi_k(1,M)$	$\phi_k(2,M)$	$\phi_k(3,M)$
1	3	16	0	0	0
2	0	103	53	8	35
3	1	105	74	119	97
4	2	0	45	89	112
5	2	50	47	31	64
6	3	29	0	122	93
7	0	115	59	1	99
8	1	30	102	69	94
9	0	92	25	92	103
10	1	78	3	47	91
11	2	70	88	11	3
12	0	66	65	31	6
13	2	39	62	19	39
14	3	84	68	66	113
15	0	79	91	49	92
16	1	70	70	81	119
17	2	29	115	96	74
18	0	32	31	38	73
19	1	45	121	83	116
20	2	113	45	42	31
21	0	86	56	58	127
22	1	1	54	24	98
23	2	42	108	25	23
24	1	118	14	92	38
25	2	33	30	38	18
26	3	126	116	120	62

In the standard directly systematic **G** is been given and Dimensions of G for different code rates are as shown above

Information Block Length, k	Generator Matrix (**G**) Size		
	Rate 1/2	Rate 2/3	Rate 4/5
1024	1024 × 2048	1024 × 1536	1024 × 1280
4096	4096 × 8192	4096 × 6144	4096 × 5120

Figure 3.3: Generator matrix size for different rates

3.2.2 Modulator and AWGN channel

This block modulates the LDPC coded bits into corresponding modulation scheme. Here we are considering QPSK, 16-QAM and 64-QAM modulation schemes. The channel is just an additive white gaussian noise channel and the factor we are varying is signal to noise ratio(SNR). For different SNRs, the channel effect is different. Better the SNR, better to decode so that the received signal contains less errors.

3.2.3 LDPC Decoder

Hard decision decoder deals with the bit stream. This bit stream is obtained after demodulating the received signal. We consider bit flipping algorithm for hard decision decoding.

For Hard decision decoding, bit flipping algorithm is considered. This algorithm takes the demodu- lated bits from the demodulator. The steps involved in this algorithm are

- Compute the syndrome $r * H^T$
- If the syndrome vector is zero, the code word is without errors.
- For each bit , compute the unsatisfied parity checks.
- Flip the set of bits for which unsatisfied checks are more.
- Compute the syndrome again, if it satisfies stop. otherwise repeat upto certain no. of times and declare as decode failure.

3.2.4 Results

The BER is calculated after the stream of bits received are demodulated and decoded and the plots are plotted.Below the plots for SNR vs BER are shown for both block sizes 1024 and 4096 with rates 1/2, 2/3 and 4/5.

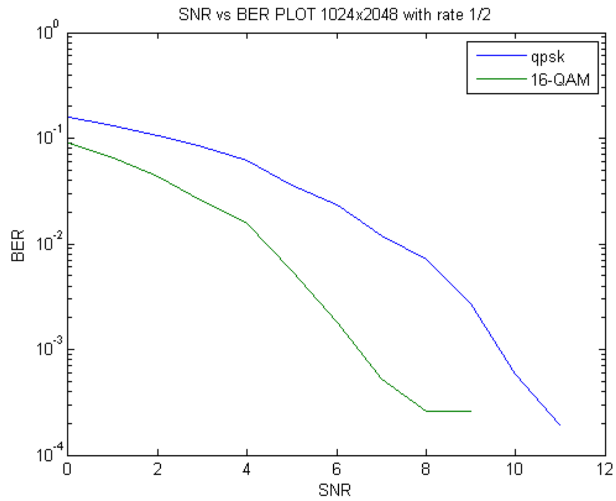

Figure 3.4: SNR vs BER for information block 1024,rate 1/2

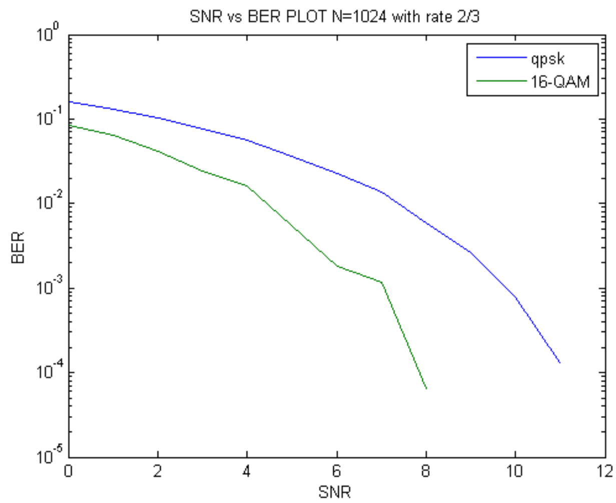

Figure 3.5: SNR vs BER for information block 1024,rate 2/3

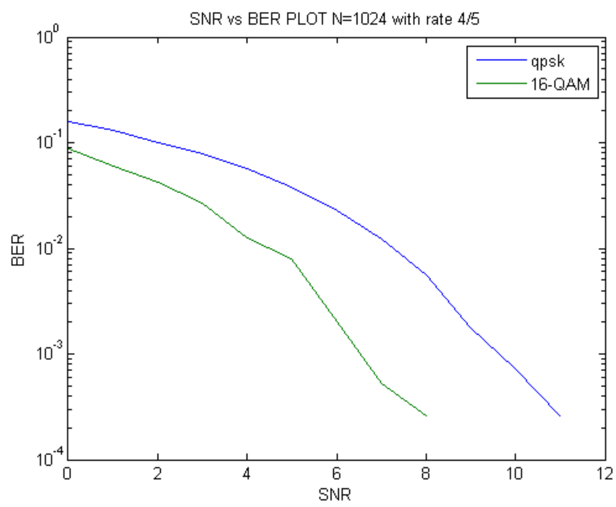

Figure 3.6: SNR vs BER for information block 1024,rate 4/5

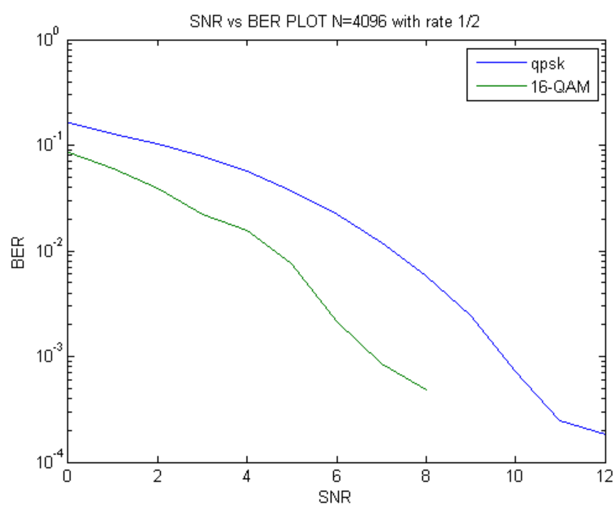

Figure 3.7: SNR vs BER for information block 4096,rate 1/2

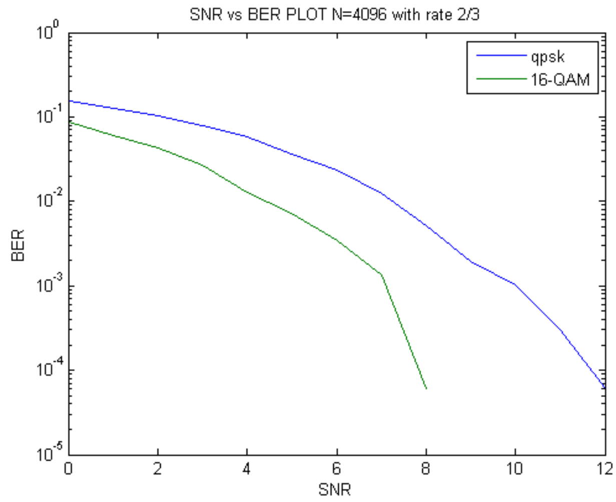

Figure 3.8: SNR vs BER for information block 4096,rate 2/3

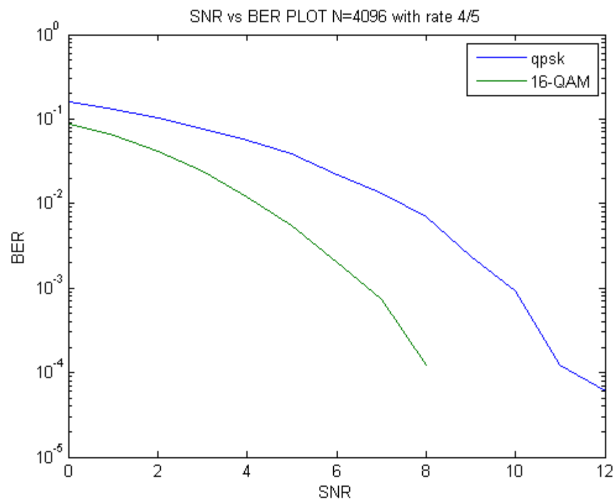

Figure 3.9: SNR vs BER for information block 4096,rate 4/5

References

[1] *IEEE802.16e Standard.*

[2] Introducing Low-Density Parity-Check Codes: *Sarah J. Johnson*

[3] APPENDIX R:Low-Density Parity Check Codes for Telemetry Systems