LOVELY PROFESSIONAL UNIVERSITY

PHAGWARA (DISTT. KAPURTHALA), PUNJAB

TITLE:

Efficient Encoding by using Compression of Trellis Stages with Constant-LOG Map Algorithm.

A Dissertation Proposal submitted

By

SEEMA JOSHI

To

Department of Electronics and Communication Engineering

In partial fulfilment of the Requirement for the

Award of the Degree of

Master of Technology in Communication

Under the guidance of

MR. Avinash Rajoriya

(MAY 2015)

ACKNOWLEGMENT

An endeavour of a long period can be successful only with the advice of many well-wishers. I take this opportunity to express my deep gratitude and appreciation to all those who encouraged me for successful completion of this thesis work.

I would like to express my heart –felt gratitude to my parents without, whom I would not have been privileged to achieve and fulfill my dreams.

I wish to express my sincere gratitude to Head of Electronics and communication Engineering Department for her consistent help and encouragement to complete the thesis work.

I am grateful to the individuals who contributed their valuable time towards my thesis. I wish to express my sincere and heart full gratitude to my guide **Mr. Avinash Rajoriya** Assistant Professor, Department Electronics Engineering who guided me to take up this thesis in sync with global trends in scientific approach.

Finally, I would like to express my sincere thanks to all who helped me to complete the thesis work successfully.

Seema Joshi

Reg No.:11006233

CERTIFICATE

This is to certify that **Seema Joshi** bearing Registration no. 11006233 has completed objectives formulation of thesis titled, **"Efficient Decoding by using compression in trellis stages with constant-LOG MAP Algorithm"** under my guidance and supervision. To the best of my Knowledge, the present work is the result of her original investigation and study. No part of the thesis has ever been submitted for any other degree at any University

The thesis is fit submission and the partial fulfillment of the conditions for the award of Master of Technology in Electronics and Communication Engineering.

Date**:**

Avinash Rajoriya
Assistant professor
School of Electronics and Communication Engineering
Lovely Professional University
Phagwara, Punjab

DECLARATION

I, **Seema Joshi**, student of Master of Technology under Department of **Electronics and Communication Engineering** of Lovely Professional University, Punjab ,hereby declare that all the information furnished in the this thesis report named **"Efficient Decoding by using compression in trellis stages with constant-LOG MAP Algorithm"** is based on my own intensive research and is genuine.

This thesis does not, to the best of my knowledge, contain part of my work which has been submitted for the award of my degree either of this university or any other university without proper citation.

Date: **Seema Joshi**

RegisterationNo.11006233

ABSTRACT

Over the past few decades, there have been various studies on turbo codes. The main objective of this work is to explore one of the various coding and decoding algorithms to reduce the complexity of hardware implementation by reducing area and increase power and reduced error performance. As MAP algorithm which is consists of so many multiplication and additions increase the computation complexity. To overcome this limitation and attain faster decoding max log algorithm is being used. In this thesis we implement an error correction coding in which we designed a turbo decoder whose speed increase as by reducing the number of trellis stages in the encoding level and then applied different version of decoding algorithm to decode the data. This will increase the speed and save the time .As without compression the trellis stages of high turbo code become to long that it take too much of time and decoding algorithm become inefficient. Hence using bit agglomerated method we reduce the trellis stages and by using different version of constant- log MAP algorithm we find the reduction in the BER.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLE

CHAPTER-1
INTRODUCTION

1.1 INTRODUCTION

Today, we are living in the information age. People talking over cellular phones is a common sight, movies are rented in the form of DVD disk. Information has become the key to success. Behind all this exchange of information 0's and 1's lies that hold information by the way they sit next to one another.

The LTE(Long Term Evolution) is the current standard towards 4G,which is designed to increase the capacity and throughput performance when compared to UMTS () and Wimax.The introduction of turbo code in digital communication system represent a step towards more reliable data transmission which achieve Shannon limit performance. This code has been successfully implemented in satellite and video conferencing system and provision has been made in 3rd generation mobile system. The decoder we used in system are compared for complexity as well as for equal number of iteration. As we use puncturing to increase the rate of turbo code due to which a long stage of trellis is obtained. The turbo decoder takes lots of time and the algorithm like sliding window algorithm become inefficient.

In this thesis we design the turbo code for LTE standard with compressed trellis stages and four different versions of MAP algorithms. The better performance in BER is obtained. Designing the turbo code is a trade-off between energy efficiency, bandwidth efficiency and complexity and error performance.

This report is divided in to five section first section gives the introduction of the objective and scope of study. Chapter 2 gives the literature review of all the work which has been studied during the whole process. Chapter 3 is research methodology which describes the whole research process used till now. Chapter 4 shows the result show the implementation the proposed and conventional architecture. The last Chapter show the result that has been achieved using after implementing the proposed architecture.

1.2 Objective of Study

The general objective of this dissertation work is to implement turbo code of compressed trellis with different version of map algorithm.Seting apart the objective the following are the main goals of this work:-

(1).Make turbo decoding more efficient by reducing the trellis stages.

(2) Reduce the BER, improvement in hardware implementation.

(3) Comparison between the algorithms.

1.3 Scope of Study

Turbo codes have become very flexible codes and have achieved the greatest achievement in the field of communication system. As turbo code remain unaffected even under heavy traffic condition so it is adopted for wireless communication system .Decoding algorithm like MAP algorithm perform its necessary computation in Log domain to reduce the hardware complexity. The recent standard LTE has also use the turbo codes in its architecture. To make the decoding complexity low the trellis compression which reduces the number of stages in the trellis diagram and better decoding algorithm is used. The constant MAP algorithm which reduce the error performance is used .This will help in improvement in hardware implementation which help in reduction in both area and power consumption.

CHAPTER-2

REVIEW OF LITERATURE

2.1 Channel Coding

As, there is tremendous increase in the trends of digital communication in the fields of cellular, satellites and computer communication. The information is represented in the form of sequence of bits. The processing of these bits is done in digital domain. This binary data is modulated and transmitted over communication channel. The signal received after channel is corrupted by noise and interference. At the receiver side the noise bit is demodulated and mapped into binary bits. The BER of the signal depend on the noise and interference induced by channel. In the digital transmission system, error can be control through channel coding scheme. This protect the signal from the effects of channel noise and interference and ensure that it reduce the BER and improve the reliability of information that is transmitted trough channel.

The Communication channel block diagram is:-

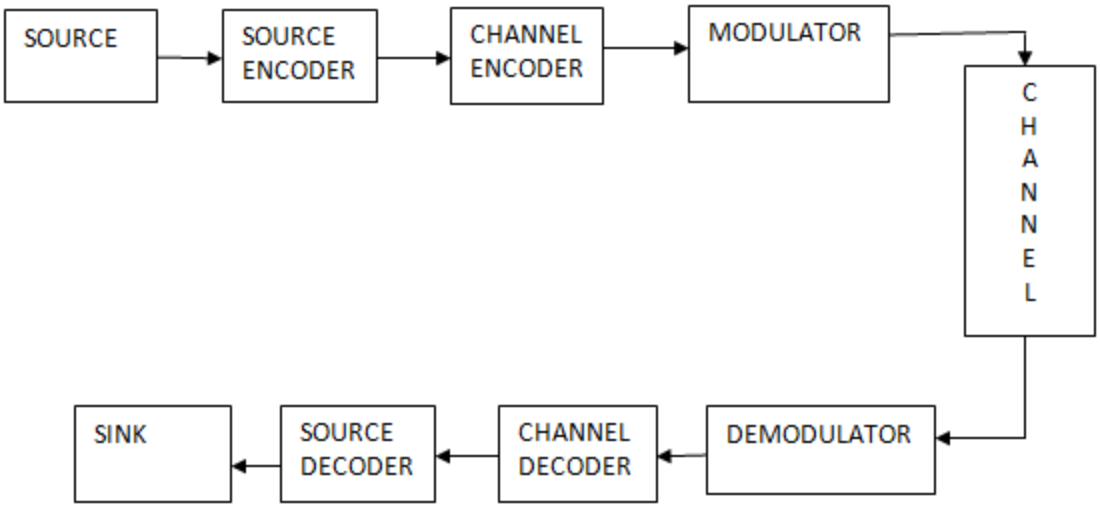

Figure 2.1: Communication channel block diagram

2.2 TYPES OF CHANNEL CODES

The different types of channel codes are:

- Block code.
- Cyclic code.
- Convolutional code.
- Turbo code.

2.2.1 Convolutional Coding:

Block code have very large block lengths which is the main reason of its failure as until then entire block of encoded data is received at the receiver, the decoding process cannot start which cause delays in the result.

Convolutional code is basically defined by two parameters:-

- Code Rate (k)
- Constraint length (K).

Code rate:-The ratio of number of input bits (k) to the number of output bits (n) is known as code rate.

Constraint length: - The length of convolutional encoder (the maximum numbers of input bits that either output can depends on)is known as constraint length.

$$K=m+1$$

where K=constraint length

m = Number of stages in shift register.

The shift register give the information about the state of encoder whereas the constraint length gives the number of bits on which our output depends.

The block diagram of convolutional encoder consists of:

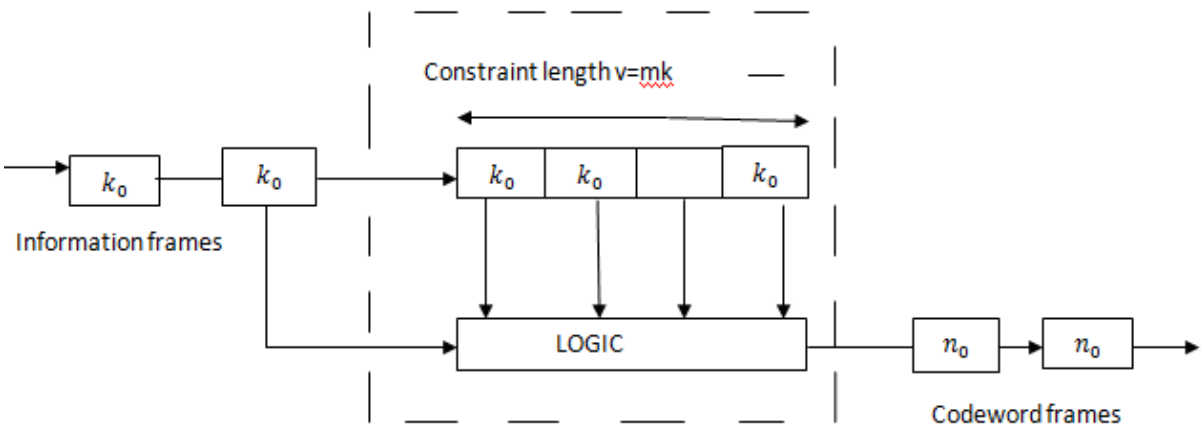

Figure 2.2: Block diagram of convolutional encoder

- Memory, which is basically shift register.
- A logic circuit.

The memory of a shift register can store m information frames. Each time a new information frame is arrive that shifted into a shift register and the oldest one is discarded. When a new frame is arrives, the encoder computes the new codeword frame using new frame and stored previous m frames. The computation of codeword is done with the help of logic circuit. The number of shift register encoder present in the convolutional encoder is known as Constraint length.

2.3 DECODING ALGORITHM

2.3.1 The Viterbi algorithm:

It uses two metrices:

- The branch metrices.
- The path metrices.

Branch matrices: It is the distance between what is transmitted and what is received and defined for each arc in the trellis. In hard decoding we have sequence of digitized parity bits, the branch metric is hamming distance between the expected parity bits and received bits.

Path metrices: It is a value associated with a state in the trellis (a value associated with each node).For hard decision decoding, it corresponds to the hamming distance over the most likely path from the initial state to the current state in the trellis. The path with minimum hamming distance minimizes the total number of bit error.

The operation of extending and pruning that constitute the heart of the viterbi algorithm are :

$$M_t(q)=min_p[\ M_{t-1}(p)+\mu_t(r_t,x'^{(p,q)}]$$

The viterbi algorithm is summarized in three steps:

- For each q state at tine t+1, find the path metric for each path to the state q. then add the path metric$M_{t-1}(p)$ of each survivor path.
- The survivor path to q is selected as that path to state q which has smallest state metric.
- Store the path and path metric to each state q.
- Increase t and repeat until complete.

2.4 TURBO CODES:

In 1993, at a moment, when many people not believe in the practicability of capacity approaching codes, the implementation of turbo code brings revival for channel coding research community.

Turbo codes, a new techniques of error correction coding that tremendous impact on channel coding in last few years. The characteristic feature of turbo codes are:-

(1) The iterative decoding mechanism.
(2) Recursive Systematic Encoders.

Firstly turbo codes are parallel concatenated convolutional encoders separated by a interleaver .The block diagram of turbo coder is:-

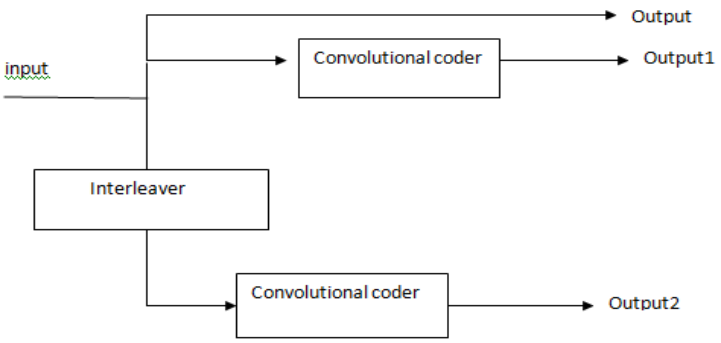

Figure 2.4: Block diagram of turbo encoder

In the nut shell, turbo codes make use of three simple ideas as:

- For simple decoding : Parallel concatenation of codes is used.
- For better weight distribution: Interleaver is used.
- For enhancing decoder decision: Soft decoding is used.

2.5 Recursive convolutional encoders AND Non recursive encoders.

There are two types of convolutional code in which the recursive encoders are better for turbo codes on comparing to non recursive encoders because they produce higher weight code words.

The non-recursive convolutional encoder:-

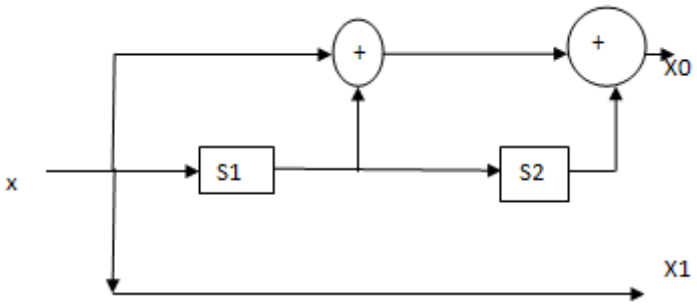

Figure 2.5(a) Non Recursive convolutional coder

The recursive convolutional encoder:-

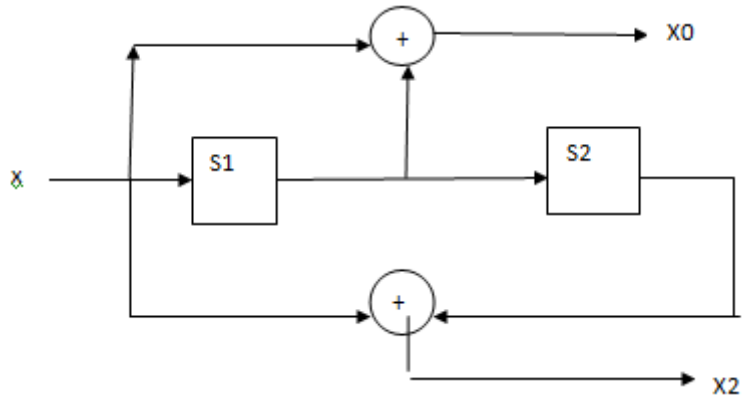

Figure 2.5(b) recursive convolutional coder

As table shown below has output to the same input sequence given to two encoders. The RSC has higher weight code as compared to non recursive which is better for error reduction. In Turbo codes the main purpose of implementing the RSC code is to make use of recursive nature of the encoders.

Encoder type	INPUT	X0	X1	WEIGHT
NON-RSC	1000	1110	1000	4
RSC	1000	1110	1010	5

Table 2.5 Input and Output sequence of convolutional coder

The both the encoder cane be described with same trellis structure as both have same minimum free distance but the BERs will be different for both of them as it depend upon the input-output data of the encoders. RSC encoder is preferred because it gives less BER at low SNR on comparing to non-recursive convolutional coder.

2.6 INTERLEAVER

It is used to change the order of the data sequence of N input sequence. It provides randomness to the input sequences and increases weight of the code words. The basic interleaver design concept is shown by following algorithm:

- Generate a Interleaver of any of the type, generally we use random interleaver.
- Generate all input sequence.

- Determine the resulting code words for all input sequences and then determine the weight of the codeword.
- Determine the weight distribution of the codeword.
- After collecting data, determine the codeword which have minimum weight and the number of codeword having same weight.

2.6.1 DESIGN OF INTERLEAVER:

- **Matrix interleaver**: This is the most common interleaver used in communication system .it writes the data in a matrix form ,cloumnwise from top to bottom and left to right without repeating any of the data bits. Data is read row-wise from top to bottom and left to right.
  Example:
  For an input sequence [S A T M V U], the interleaver matrix is

$$\begin{bmatrix} S & A & T \\ M & V & U \end{bmatrix}$$

- **A Random (Pseudo-random) Interleaver**: In this type of interleaver input data is arranges according to fixed random permutation.

  Example: Consider the data of length l=5 and the random permutation will be [2 5 1 3 4].
  If input data is [P Q R S T], the interleaved sequence is [Q T P R S]

- **Odd even interleaver**: First the bits are left interleaved and encoded, but odd positioned coded bits are stored then bits scrambled and encoded but now only even position coded bits are stored. This type of interleaver guarantees an even protection of each bit of data and provides a uniform distribution of the error correction capability and better performance of the code.

2.7 TURBO DECODING

Previously, in decoding approach, demodulation is based on hard decision of the received data. This discrete value is then passed on the error control decoder which become complex

and has some disadvantages as decoder is not taking any use of certainty of information available to it while decoding.

Hence the similar but better rule of a priori probabilities of the input has come into account. If the +1 has probability of 0.9 and the symbol falls in the negative decision range, The maximum likelihood decoder will decide it as -1 not as +1.This detection method is known as minimum error rule.

The block diagram of turbo decoder is:-

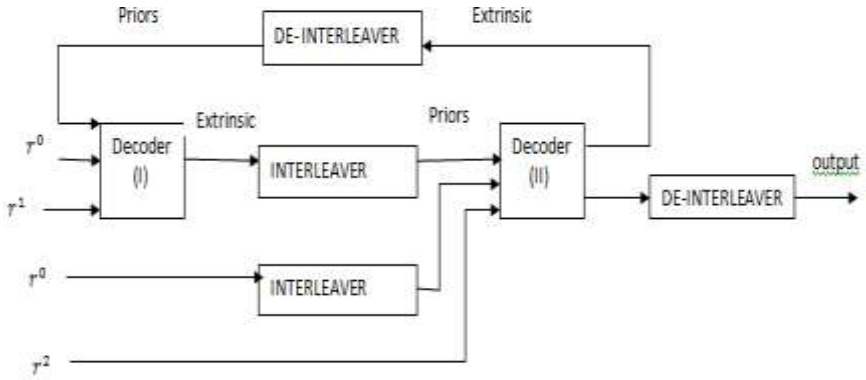

Figure 2.7: Block diagram of turbo decoder

2.7.1 MAP ALGORITHM

This process of decoding involve the use of a posteriori probabilities of each information bit followed by choosing the data bit value corresponding to MAP probabilities for that data bits. During decoding the decoder receive as input a soft value of the signal. In case of turbo codes there are two decoders to decode the output from both encoders. Both of them provide estimates of the same set of data bits but in different order due to presence of encoder. Information received is iterated a number of times in order to enhance the performance. During each iteration the estimates are evaluated by the decoders using information which are obtained from other decoders. In final stage, hard decision is made which provide each bit a value either 1 or 0.

The AAPs are used to find out the likelihood ratio. The logarithmic form of likelihood ratio is:

$$\Lambda_1(d_k) = \log\frac{\Pr(d_k=1|R_N^1)}{\Pr(d_k=0|R_N^1)} \qquad (2.1.1a)$$

Pr $(d_k = 1|R_1^n)$ is a posterior probability (APP) of the information input data at time k(d_k) =1 given the entire received data. The observation block data is R_N^1= {R1... Rk..., RN...} Where R_k= $\{d_r^k, y_r^{ki}\}$ the state of the encoder S_k is represented by a v-tuple

$$S_{k=}\{a_k, a_{k-1}, \ldots\ldots\ldots\ldots\ldots a a_{k-v-1}\} \qquad (2.1.1b)$$

where $a_k$is the output of the first shift Register.

The conditional joint probabilityΓ_k(s) is defines as:

$$\Gamma_k^j(s) = \Pr(d_k=j, S_k=s|R_1^N) \qquad (2.1.1c)$$

The APP of $d_k$is equal to

$$\Pr(d_k=j|R_1^N) = \Sigma\, \Gamma_k^j(s) \quad j=0, 1 \qquad (2.1.1d)$$

The LLR is written by substituting (4) into (1)

$$\Lambda_1(d_k) = \log\frac{\Sigma s\, \Gamma_k^1(s)}{\Sigma s\, \Gamma_k^0(s)} \qquad (2.1.1e)$$

Now, the denominator and numerator of equation (2.11e) is multiplies by Pr $(R_1^N)$and these value will become joint probabilities instead of conditional probabilities.

At k=1 the state is s then equation 5 can be written as

$$\Lambda_1(d_k) = \text{Log}\frac{\sum_s \sum_{s'} \Pr(d_k=1, S_k=s, S_{k-1}=s', R_1^N)}{\sum_s \sum_{s'} \Pr(d_k=0, S_k=s, S_{k-1}=s', R_1^N)} \qquad (2.11f)$$

The BCJR algorithm is defines these probabilities in terms of three parameters

$$\alpha_k(s) = \Pr(S_k=s|R_1^N) \qquad (2.11g)$$

$$\beta_k(s) = \frac{\Pr(R_{k+1|S_k=s}^N)}{\Pr(R_{k+1|R_k}^N)} \qquad (2.11h)$$

And

$$\gamma_j(R_k, s', s) = \Pr(d_K=j, S_{k=}s, R_k|S_{k-1} = s') \qquad (2.1.1i)$$

The LLR in (6)can now be described in terms of (7)(8)(9)

$$\lambda_1(d_k) = \text{Log}\frac{\sum_s \sum_{s'} \gamma_1(R_{k,}s',s)\alpha_{k-1}(s')\beta_k(s)}{\sum_s \sum_{s'} \gamma_0(R_{k,}s',s)\alpha_{k-1}(s')\beta_k(s)} \qquad (2.1.1j)$$

α_k and β_k can be computed using forward and backward recursion

Whereh_α and $h_\beta$are normalized factor. $\gamma_j(R_k$,s',s) consists of transition probabilities of the discrete Gaussian memory less channel and transition probabilities of the encoder trellis.

$$\alpha_k(s) = h_{\alpha \sum_{s'} \sum_{j=0}^{1} \gamma_1} \left(R_{k,}s', s\right)\alpha_{k-1}(s) \qquad (2.1.1k)$$

$$\beta_k(s) = h_{\beta \sum_{s'} \sum_{j=0}^{1} \gamma_1} \left(R_{k+1,}s', s\right)\beta_{k-1}(s) \qquad (21.1L)$$

$(R_k$,s',s)=Pr $(R_k|d_k$=j,S_k=s,S_{k-1}=s')*Pr(d_k=j|S_k=s,S_{k-1}=s)*Pr($S_{k=s}$,S_{K-1}=s')

The second term is divided into two terms :

Pr($R_k|d_k$=j,S_k=s,S_{k-1}=s')=Pr($d_r^k|d_k$=j,S_k=s,S_{k-1}=s')*Pr($y_k^{ri}|d_k$=j,S_k=s,S_{k-1}=s')

d_r^k And y_k^{ri} are two uncorrelated Gaussian variable in R_k based on equation

The MAP decoding algorithm based on following steps:

- Initialize $\alpha_0(s)$ and $\beta_N(s)$ as

$$\alpha_0(s) = \begin{cases} 1 & if\ a = 0 \\ 0 & otherwise \end{cases} \qquad (2.1.1n)$$

$$\beta_N(s) = \frac{1}{M} \text{ For all s} \qquad (2.1.1o)$$

Where M is total number of states

- On receiving each $d_r^k$and corresponding y_k^{ri}, the decoder computes $\gamma_j(R_k$,s',s)for j=0and 1,then compute α_k(s) for all value of s . The computed value are stored for 1<=k<=N.

- The backward recursion for is performed after all the N data sequence and their corresponding parity bits are received for 1<=k<=N-1.
- The soft output $\lambda_1(d_k)$ are computed for 1<=k<=N.

Implementation of the MAP algorithm in term of the likelihood ratio is difficult to compute as it become complex because of multiplication operations. So to reduce this complexity it can be computed in the logarithmic domain.

2.8 Survey of literature

The introduction of turbo decoding in 1993 has been considerable research on the investigation of parallel concatenation coding schemes employing convolutional codes as component codes. It is very important for turbo code to have low code rate to moderate signal-to –noise ratio.

Mrs.K.MBogaware,Ms.shardaMungale, Dr.manishChavan."Implementation of turbo encoder and decoder": the paper consists of main feature of turbo codes. Turbo code has become the coding technique in many communication and storage system due to its near Shannon limit error correction capability. It shows that turbo code has provided 0.8db improvement in Eb/No. The number of iteration depends on the physical channel characteristics.

E.Boutillon,J.Sanchez-Rojas and C.Marchand " Simplified Compression of redundancy free trellis stages in turbo decoder" This paper is simplified form of the paper given below in which the redundancy free trellis sequence of length N is reduced to sequence of m-1(N mod(m-1)).this shows that for an m state turbo decoder. Among the L compressed stage, only m=3 or even m=2 are necessary. This is known m-min algorithm which is used to increase the throughput for decoding as a high rate turbo code reduces the power consumption.

E.Boutillon,J.Sanchez-Rojas and C.Marchand "Compression of redundancy free trellis stages in turbo decoder":This paper reduces the number of trellis stages in turbo decoder with the help of bit agglomerated method due to which decoding become faster .the bit agglomeration is just the xor of bits in redundancy free trellis stage. The xor of two or more stages are will reduced the stages in one particular stage hence the trellis will reduce.

Rohan M.Pednekar “Design and implementation of convolution encoder with viterbi decoder”:As convolutional encoder with Viterbi decoder is more powerful structure for forward error correction techniques. It is used in the communication system for reliability and efficiency of data transmision.The principle behind Viterbi is maximum likelihood.

C.weiss, C.Bettesteller, S.Riedel, D.J Costeello”Turbo decoding with tail biting trellises”:In this paper the calculation of the minimum distance of parallel concatenated code and guidelines to choose the good tail-biting component codes. The state space representation of encoder use in base paper is taken from this paper which helps in finding the state of trellis. In this tail=biting codes are taken as component codes for parallel concatenated codes.

Yassir Nawaz, R.Venkatesan and Paul Gillard “Multiple Bit Releasing Window Turbo Code” The sliding window implementation of decoding algorithm is used to reduce the memory requirement in turbo decoders. Decoding algorithm like MAP is done along with sliding window algorithm thud reducing the computational complexity and increase the decoding speed. These all properties are achieved without any performance degradation. The result obtained denoted that multiple bit release implementation reduces computational complexity and improved performance and faster decoding speed.

Emmanuel Boutillon, Catharine Douillard, and Guido Montorsi”Iterative Decoding of Concatenated Convolutional Codes: Implementation issues”: In this the general structure of iterative decoder and the main feature of the heart of iterative decoder that is soft-input soft-output algorithm are mention. The speed of decoding can be increase by raising decoder clock frequency and limiting the decoding iteration

Ahamad Hasan Khan “BER and FER performance of Interleaver parameter for Small frame size”: In this the interleaver designs problem are discussed about small frame size and the effect of many trellis termination is observed by comparing the performance of the uniform interleaver types with and without termination scheme. Here we also analysed different termination in coding scheme when used with an optimized interleaver type in turbo coding.

E.Boutillon.J.Sanchez-Rojas andC.Marchand “Compression Of redundancy free trellis stages in turbo Decoder”: this is the base paper of mine in which long trellis due to puncturing is reduced with the help of bit agglomerated method which is beneficial in decoding process as it require less time to compute the whole trellis. In this decoding is done with max log map algorithm. It is the best and simple method of decoding.

Kale Ruttik “Decoding of punctured turbo codes using dual codes”:In this paper the construction of trellis for dual code systematic convolutional code is explained, as decoding of high rate turbo code is efficient in dual code as compared to normal code. So for working in dual code we need to know about it trellises structure. The generated dual code is decoded by max-log algorithm.

O.Jearessen and H.Meyr “Terminating the trellis stages”: this paper show how BER effect if we terminate the trellis stages. We always assume the first stage of encoder always zero but last stage depends upon the input sequence due to which decoder has problem to identified the last stage of trellis.wuth the help of termination we make the last stages of the decoder all zero states. Hence decoding become easy and improves in the BER.

Irfan Ali “Bit Error Rate Simulation using Matlab”: This paper shows the simulation of bit error rate in Matlab as BER is the important parameter that is used to assessing the system that is used in transferring the digital data from one to another place.

Mohammad Salim, RP Yadav and S Ravi Kanth. “Performance Analysis of Log-Map, SOVA and Modified SOVA algorithm for turbo decoder”. In this paper the performance of BER is seen by using different decoding algorithm like SOVA, map under AWGN and Rayleigh channel. The modified SOVA algorithm helps in improving BER performance.

Vijaysinh Patil “Implementation of Efficient Turbo code Encoder-Decoder with max log map algorithm”: In this paper the max log map algorithm is used which perform necessary computation in log domain having capacity to reduce hardware complexity. The low complexity in decoder made an efficient coding scheme to achieve improvement in hardware implementation which leads to reduction in both area and power consumption.

CHAPTER-3

RESEARCH METHODOLOGY

The details of turbo code have already given in chapter 2. This chapter will focus on the implementation of turbo encoder and decoder for LTE system using compressed trellis stages at encoder and different version of MAP algorithm.

3.1 LTE TURBO ENCODER

The LTE turbo encoder contains the Parallel Concatenated Code. It contains the constraint length K=4(8-state) RSC encoders which is parallel concatenated. The overall code rate is k=1/3.

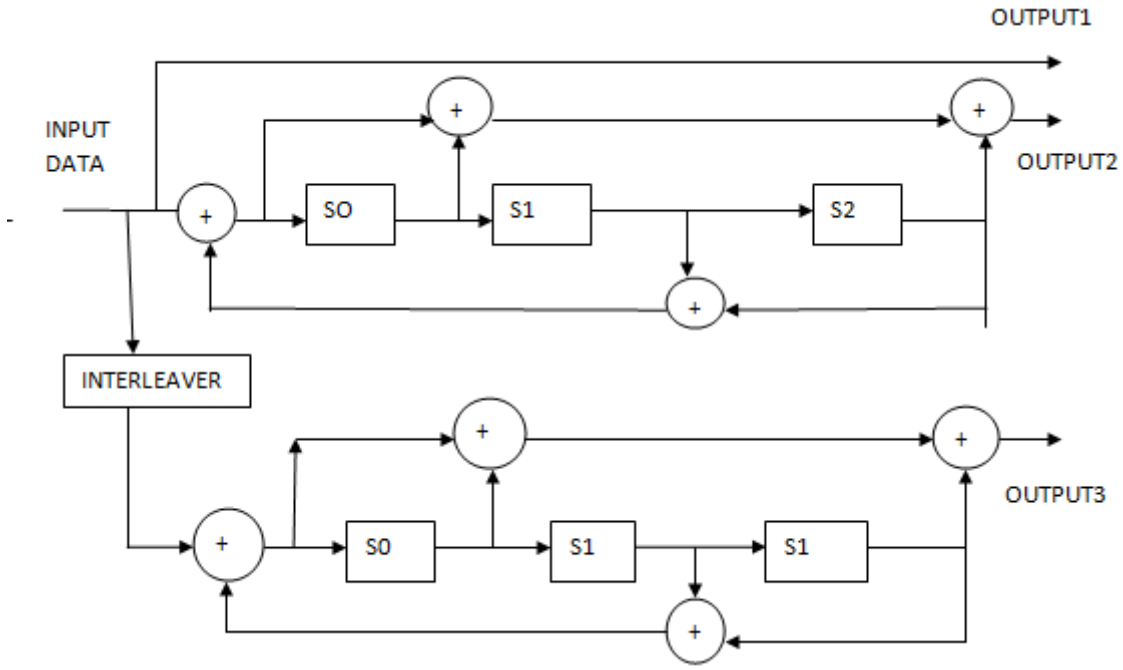

Figure 3.1: The LTE turbo encoder

The input data bits are transmitted with the parity bits generated by the convolutional encoders. Before encoding both encoder are sets into the zero state condition i.e. all the shift register are filled with zeros. It consists of interleaver which interleaves the input sequence. The type of interleaver we used here is Random (Pseudo random) interleaver. The description is given in chapter 2.Let input to the first decoder is O2, second decoder is O3, the final output is written as :[O0,O1,O2,O0,O1,O2...O1O2O3]

3.2 INTERLEAVER

The interleaver used is random (pseudorandom) interleaver the description is given in chapter2

3.3 CHANNEL

After encoding the input data, the data is modulated and transmitted over channel and then pass to decoder. The modulation scheme assumed to BPSK and the channel model is AWGN channel.

3.4 STATE TABLE OF THE ENCODER

PRESENT STATE	INPUT	NEXT STATE
000	0 1	000 100
001	0 1	100 000
010	0 1	101 001
011	0 1	001 101
100	0 1	010 110
101	0 1	110 010
110	0 1	101 011
111	0 1	011 000

TABLE 3.4: State table of convolutional encoder.

3.5 Trellis diagram

The trellis diagram of 8 state convolutional encoder is :

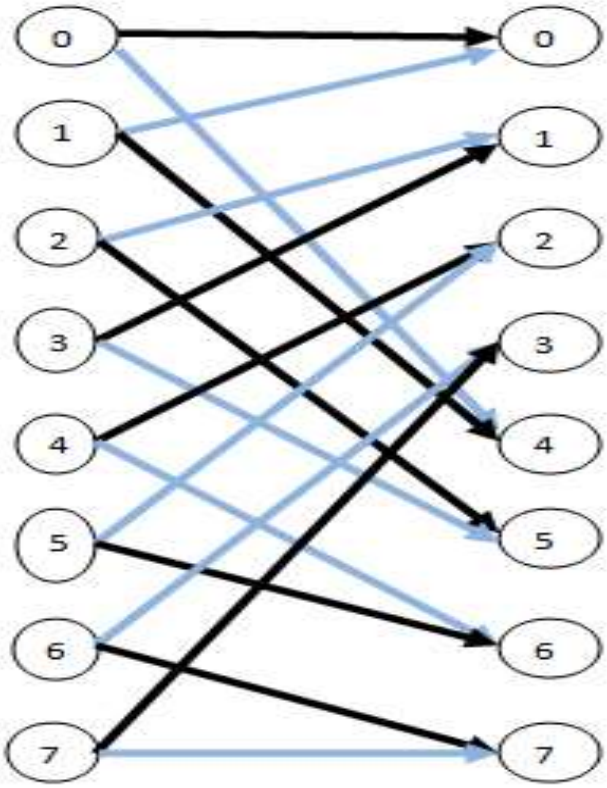

Figure 3.5: Trellis diagram of convolutional encoder

Black bold lines show the o input and blue bold line show the 1 as input.
The branches indicate the next state of the decoder can be reached from a particular present state. Every codeword follow different path in the trellis.

3.6 TRELLIS TERMINATION

As conventional codes always use a stream of zeros as tail bits, the tail bits depend on the state of the encoder .When all the data have been encoded at last the final state of the encoder is different .because of the presence of interleaver the final stage of the both the encoder used in encoding process attain different final states. So the trellis termination bits for the two encoder are different. RSC cannot be brought into the all zero state by simply passing the

zero bits so this can be done with feedback bit at the encoder input. This process is done by using the switch at the input as shown in the figure 3.6

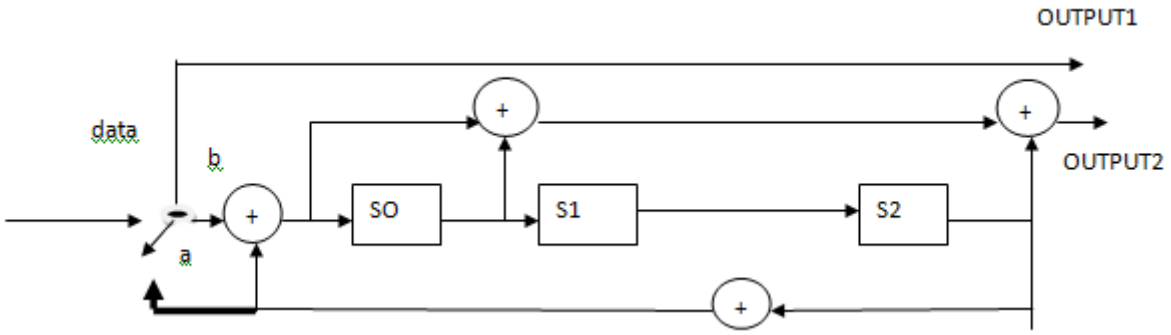

Figure 3.6 Trellis Termination strategy.

The switch shown is in position b while encoding the input data . As the input data completely encode it switched to position a for the termination of trellis. As the XOR of the same bits is zero so encoder will again come back to zero state only. Thus the feedback in the encoder is used for trellis termination as it's become input the encoder and bring the encoder to all zero state.

3.7 BIT Agglomerated method:

This is the method due to which we can reduces the trellis stages . As bit agglomeration consists in doing xor operation on bits which having the same residue m1.Where m1=m-1

$$D_l^a = \sum_{j=0}^{\frac{N}{m1-1}} D_{m1.j+l}$$

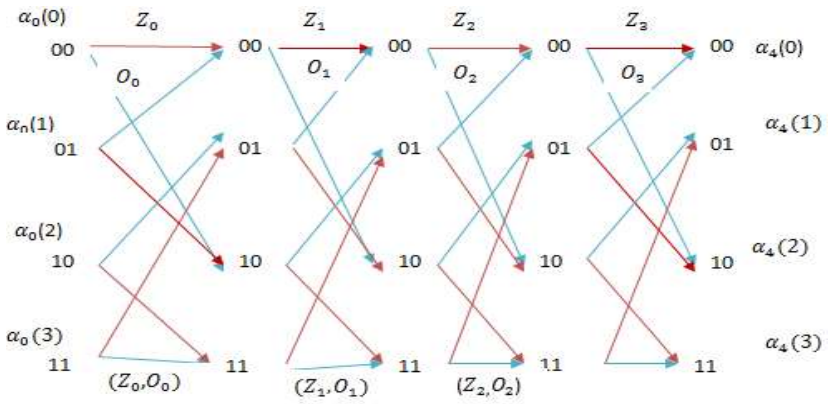

Figure 3.7 trellis diagram for 4 state convolutional coder.

where $\alpha_4(0)=\alpha_0(0).(Z_0Z_1Z_2Z_3 + Z_0O_1O_2O_3 + O_0Z_1Z_2O_3 + O_0O_1O_2Z_3)$

$+\alpha_0(1).(Z_0Z_1Z_2O_3 + Z_0O_1O_2Z_3 + O_0Z_1Z_2Z_3 + O_0O_1O_2O_3)$

$+\alpha_0(2).(Z_0Z_1O_2Z3 + Z_0O_1Z_2O_3 + O_0Z_1O_2O_3 + O_0O_1Z_2Z_3)$

$+\alpha_0(3),(Z_0Z_1O_2O_3 + Z_0O_1Z_2Z_3 + O_0Z_1O_2Z_3 + O_0O_1Z_2O_3)$

On factorizing above equation:

$\alpha_4(0) = \alpha_0(0).(Z_1Z_2(Z_0Z_3+O_0O_3)+O_1O_2(Z_0O_3+O_0Z_3))$

$+\alpha_0(1).(Z_1Z_2(Z_0Z_3+O_0O_3)+O_1O_2(Z_0O_3+O_0Z_3))$

$+\alpha_0(2).(Z_1O_2(Z_0Z_3+O_0O_3)+O_1Z_2(Z_0Z_3+O_0Z_3))$

Now according to bit agglomerated method

$$D_l^a = D_0^a = D_0 + D_3$$

So the probability density function (Z_0^a, O_0^a) of D_0^a is equal to $((Z_0Z_3+O_0O_3),(Z_0Z_3+O_0Z_3))$

Hence the trellis stages N=4 are reduced to m1=3 trellis stages.

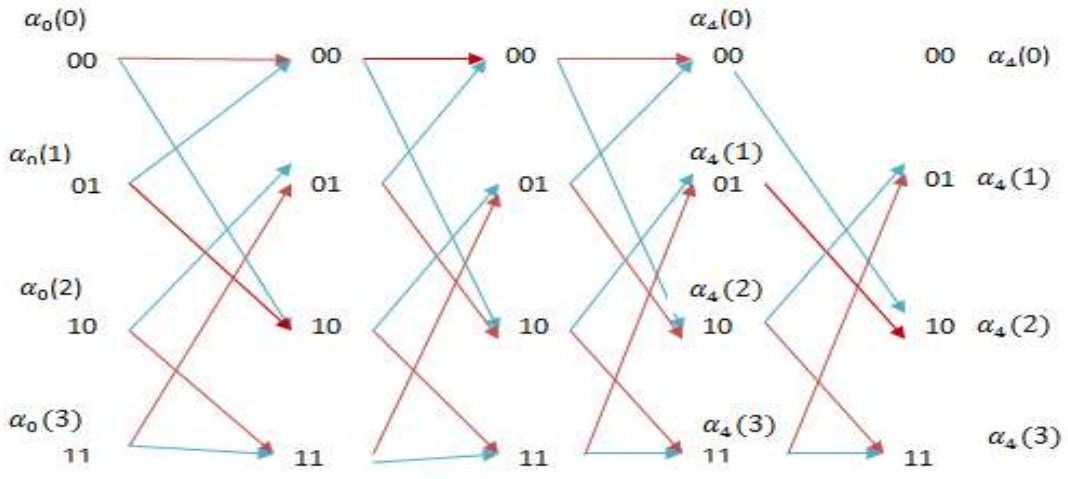

Figure 3.7.1 Trellis Diagram after compression

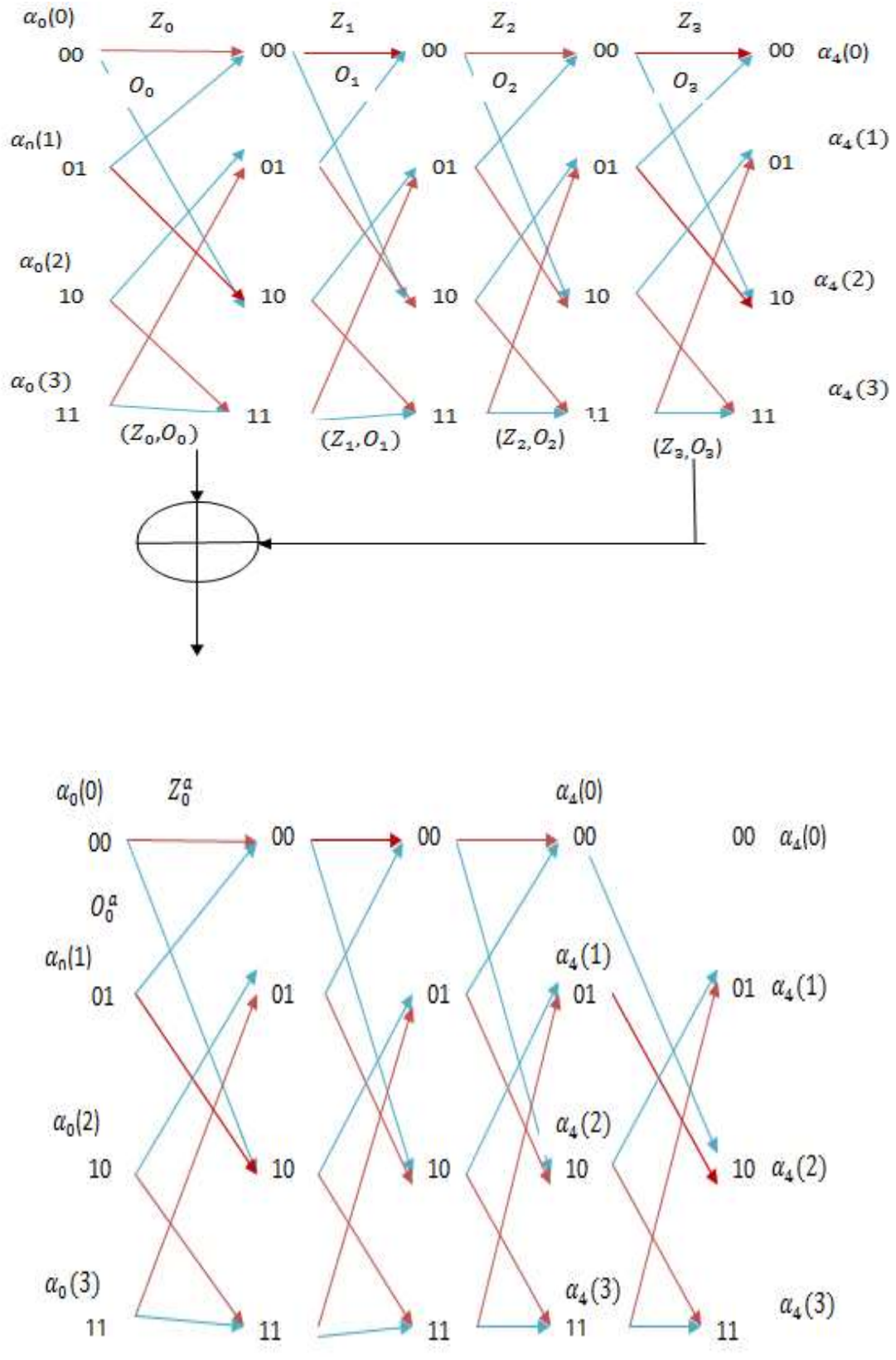

Figure 3.7.2 Trellis diagram after using BIT AGGLOMERATED method

3.8 LTE DECODER ARCHITECTURE

The architecture of the LTE decoder is shown in figure:

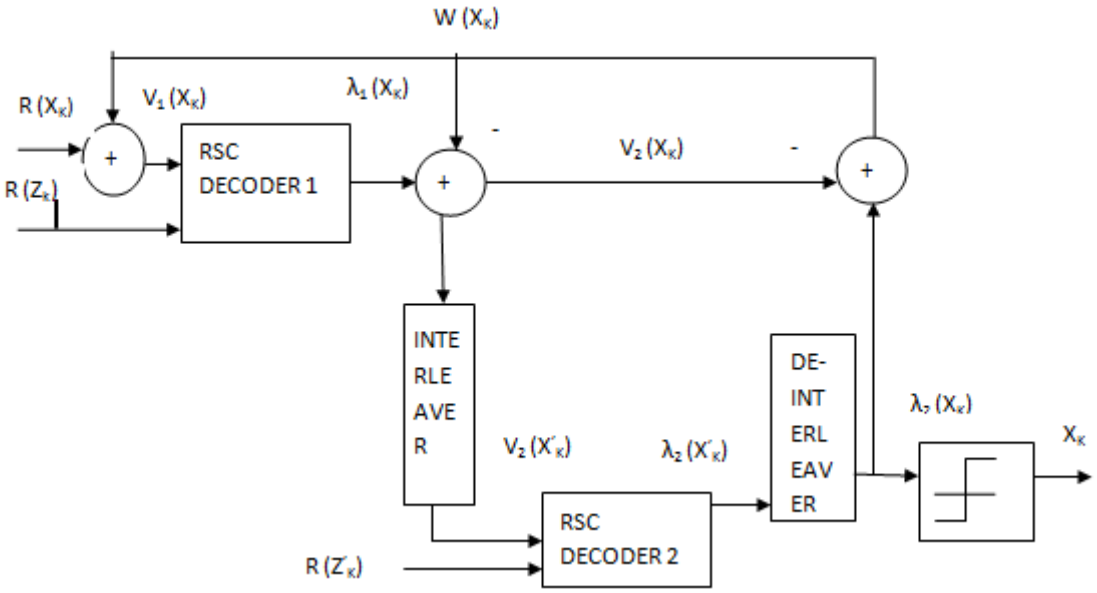

Figure 3.8: The LTE turbo decoder

The performance of decoder is improved by sharing the LLR estimates with each other. The first SISO decoder calculates the LLR and passes the value to the second decoder. The second decoder uses the value again to calculate LLR which is passes to first decoder. These back and forth exchange of information between two turbo decoder is known as iteration.

The steps followed in the decoding are:-

- The extrinsic information $W(X_K)$ is added with received systematic $R(X_K)$.

$$V_1 (X_K) = R (X_K) + W (X_K)$$

The extrinsic information $W (X_K)$ assumed to be zero in the first decoding as decoder 2 cannot work.

- The RSC decoder1 uses this information to decode the data and outputs the LLR of data bits,

$$\Lambda_1(X_k) = \text{RSC decode}[R(Z_k), V_1(X_k)]$$

- The extrinsic information W (X_k) is subtracted from the output of the first encoder.

$$V_2(X_k) = \lambda_1(X_k) - W(X_K)$$

- The output of the decoder is interleaved and then feed to second decoder.

$$V_2(X'_K) = \text{INTERLEAVER}(V_2(X_k))$$

- The input to the second decoder is the interleaved and parity bits of the second encoder.

$$\lambda_2(X'_k) = \text{RSC decode}[R(Z'_K), V2(X'_K)]$$

- Now the LLR bits are deinterleaved to make the sequence of bits to the original bits

$$\lambda_2(X_k) = \text{DEINTERLEAVER}(\lambda_2(X'_K))$$

- The sequence obtained after deinterleaving is feed back to obtain extrinsic information by subtracting $V_2(X_k)$ from it. Then the value obtained is used by the decoder 1 at the time of next iteration.

$$W(X_K) = \lambda_2(X_k) - V_2(X_k)$$

- At last, when the iterations are completed, a hard bit decision is taken to obtain the decoded bits.

$X_K=1$ if $\lambda_2(X_k)>0$

$X_K=1$ if $\lambda_2(X_k) \leq 0$

3.9 DECODING AGLORITHM (RSC DECODER)

The algorithm used in RSC decoder is the heart of the turbo decoder. It uses a trellis diagram to represent the all possible transitions state. As the LTE has three shift register so the number of different possible states is $2^3=8$.

3.9.1 MAP ALGORITHM

The map algorithm has so many technical difficulties due to high number of addition and multiplication. So to reduce these problems MAP algorithm are implemented in the logarithmic domain which reduces the computational complexity, The log map algorithm is implemented twice during each iteration ,once in the forward direction and once in reverse direction.

The four different versions based on four different $\max^*$ operations are described as:

3.9.2 LOG MAP algorithm

This is the logarithm version of the MAP algorithm .It reduces the complexity as multiplication operation are transformed into addition.

$$\max^{*}(a, b)=\ln(e^{a}+e^{b})$$
$$= \max(a, b) +\ln(1+e^{-|b-x|})$$
$$= \max(a, b) +f_c(|b-a|)$$

where a and b are the input argument.

f_c=correction function

The correction function is simply a function of absolute difference between the two input argument of the $\max^{*}$operator .max*operator calculate the maximum value of two input argument. The correction function can be computed by log function or per computed and stored in the lookup table in order to decrease complexity. The log algorithm is the one of the complex of all the four algorithm mention above but it offer best BER performance.

3.9.3 MAX-LOG MAP algorithm

In this algorithm the correction factor use in MAP algorithm is set to zero means it is not used in this algorithm.

$$\max^{*}(a,b)\approx\max(a,b)$$

This become least complex from all of the algorithm but twice complex then viterbi algorithm, which sweep through the trellis once in forward and once in backward. It offers worst BER performance. It has the additional benefit of intolerant of imperfect noise variance estimates during the operation in an AWGN channel.

3.9.4 LINEAR –LOG MAP algorithm

In this algorithm .the correction function f_c was set by look up table which implies the need of a high speed memory. To avoid this need of high –speed memory a linear approximation MAP algorithm is used. It is based on the linear approximation of the Jacobi algorithm

$$\max^*(a,b) \approx \max(a,b) \begin{cases} 0 & if\ |b-x| < T \\ s(|b-x| - T) & if |b-x| \geq T \end{cases}$$

where parameter s and T are used to minimize the total squared error between the exact correction function and its linear approximation.

3.9.5 CONSTANT –LOG MAP algorithm

As by name it is clear that it uses the constant value of correction function from look up table and thus reduces the complexity.

$$\text{Max}^*(a,b) \approx \max(a,b) + \begin{cases} 0 & if\ |b-a| < T \\ S & if |b-a| \leq T \end{cases}$$

The value of T and S in the implementation are already defined as 1.5 and 0.5,it is more susceptible to noise variance estimate errors as compared to Log map algorithm.

CHAPTER-4
IMPLEMENTATION

Since we are targeting to fast decoding ,our intention is to make the decoding fast and reduce ber.As in the base paper the trellis stages is reduced with the help of bit agglomerated method , which reduces the convergence time of long trellis stages. Now our objective is to obtain is not only to reduce the time convergence but also the computation complexity and error rate performance of the system.

In this Chapter, implementation of LTE turbo code with bit agglomerated method and constant log map algorithm. The implementations in this chapter are as follows by using Matlab.

1. The LTE turbo encoder.
2. Channel is AWGN.
3. Termination of trellis.
4. Compression using agglomerated bits.
5. The LTE turbo decoder.
6. Decoding algorithm as constant- LOG MAP algorithm.

Firstly, i implement the LTE turbo encoder in MATLAB. Turbo encoder is parallel Concatenated Convolutional Code which comprise of constraint length K=4.The code rate is r=1/3.

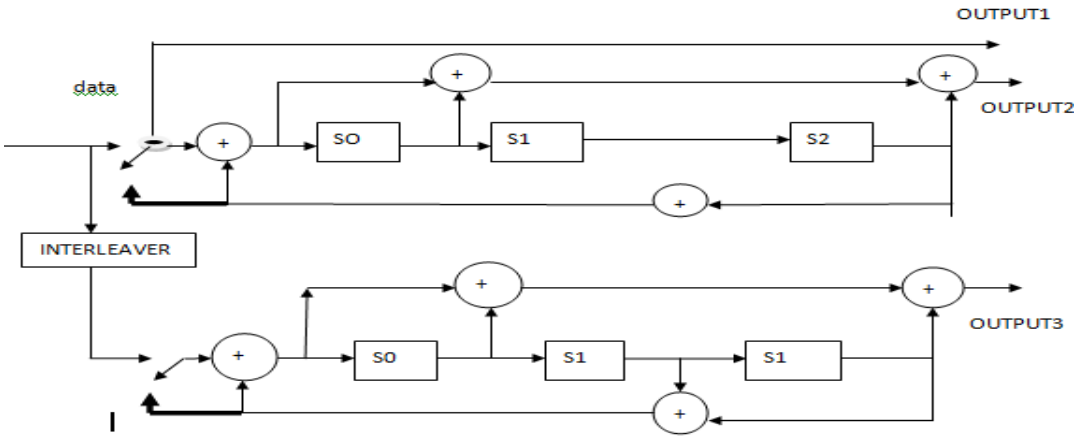

Figure 4.1 LTE turbo encoder architecture

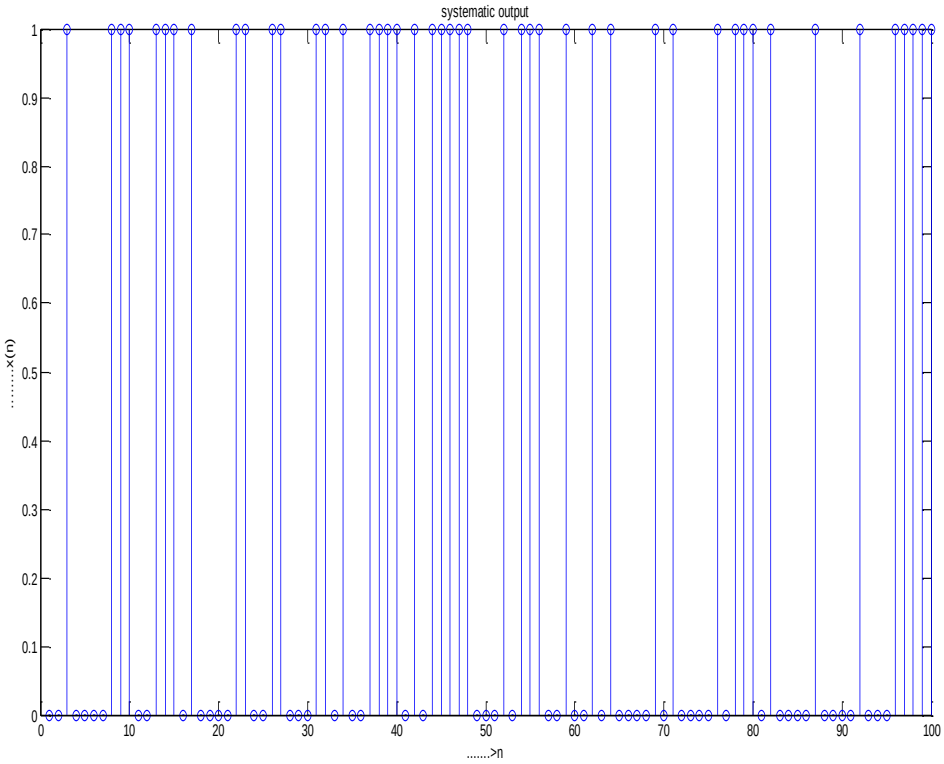

Figure 4.2 Systematic data of encoder

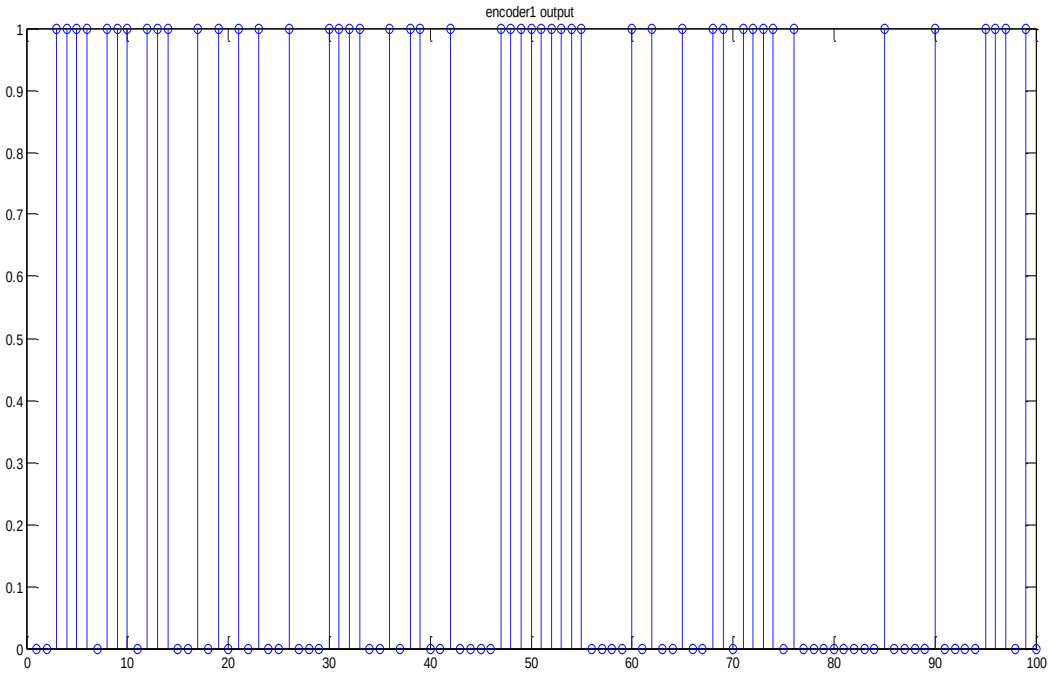

Figure 4.3 Output of the encoder 1

4.1 INTERLEAVER

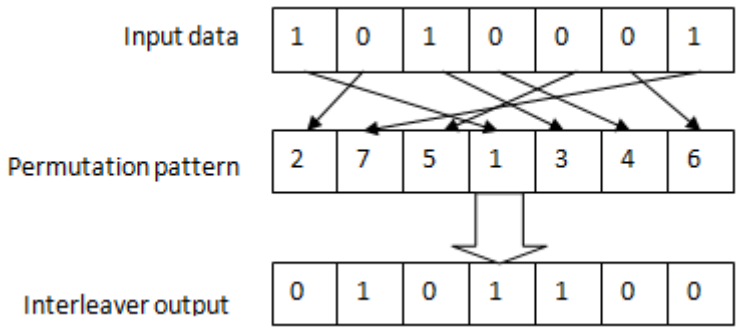

Figure 4.4 Random Interleaver

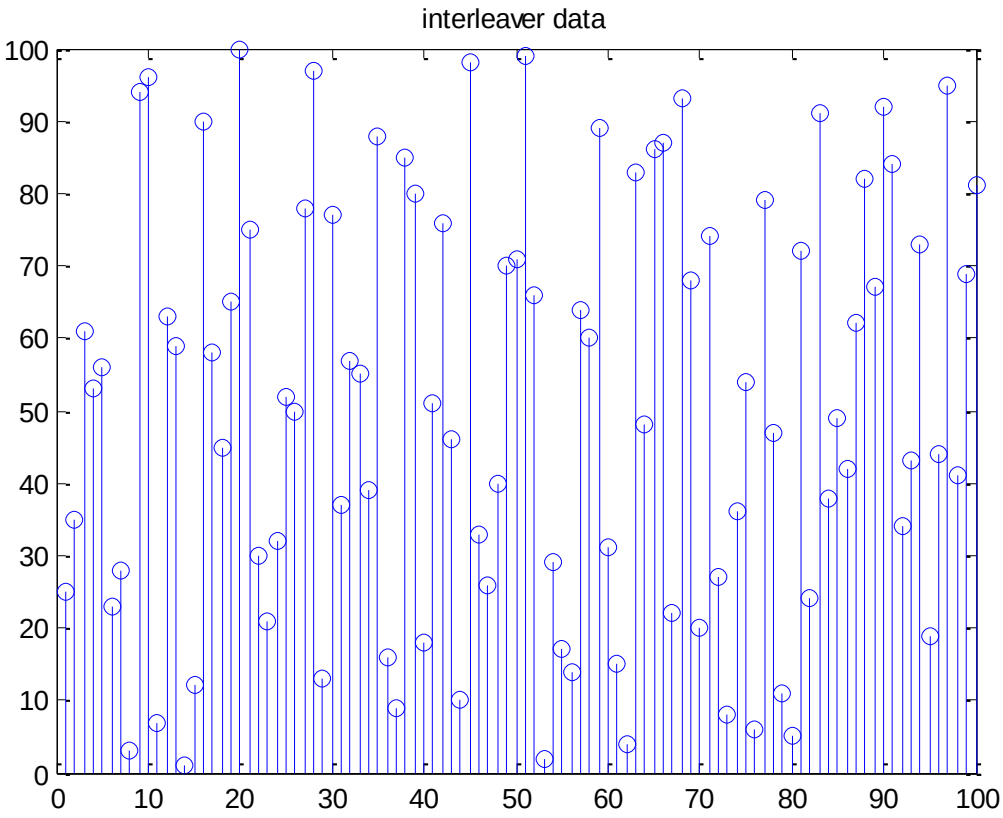

Figure 4.5 Matlab output of random interleaver

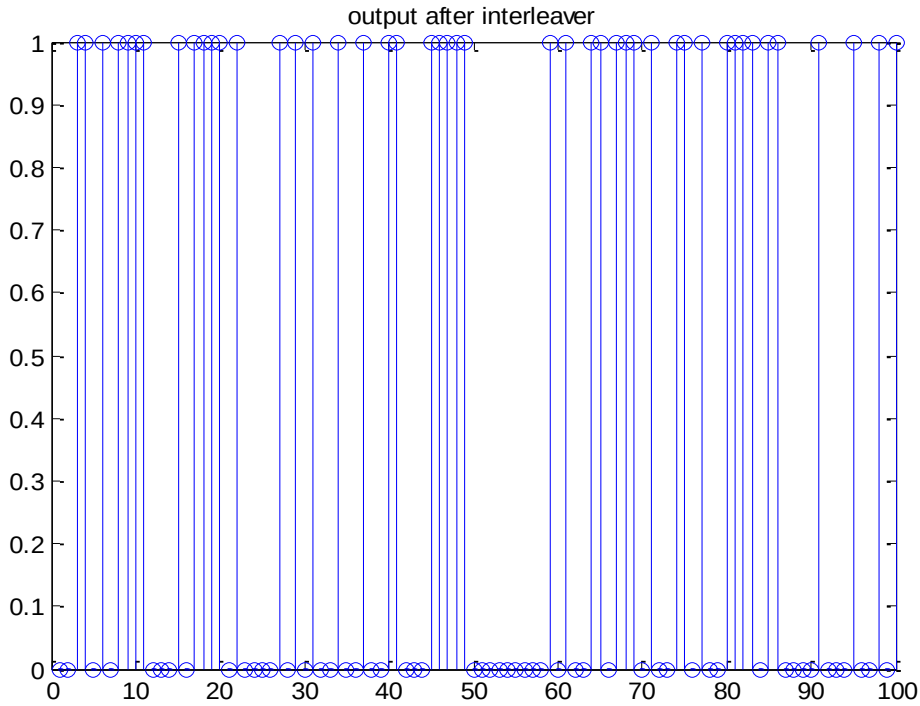

Figure 4.6 Matlab output of bits after interleaving

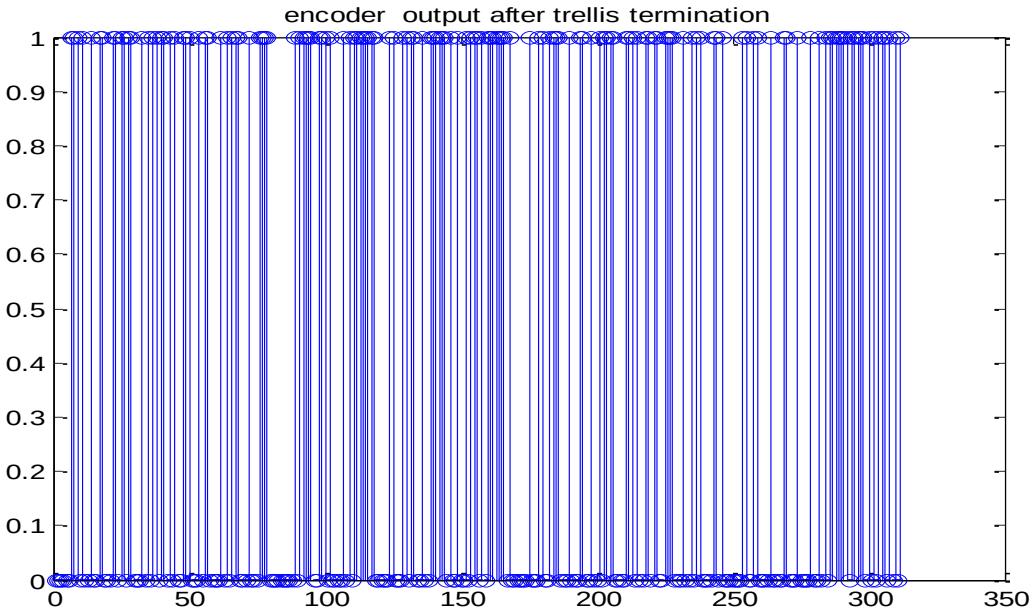

Figure 4.7 Matlab output of encoder 2

4.2 Decoder:

The decoding is with the help of Constant –LOG MAP algorithm. The numbers of iteration are 4.The graph is between BER and SNR and result is shown after each iteration. The number of iteration may be increase for better improvement but after a specific number of iteration, the BER values remain constant and it will neither increase nor not decrease. For lower BER the numbers of bits are increased or we can increase the SNR, the signal power increase and hence the apriori information data improved and decoding of data become better.

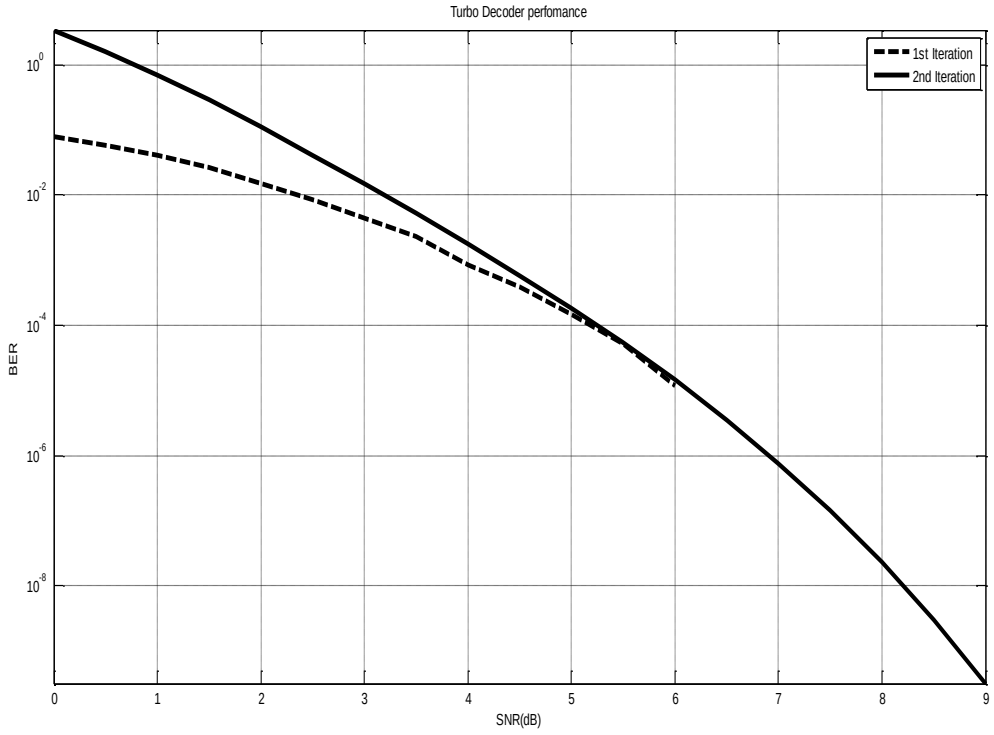

Figure 4.8 BER after 1st iteration

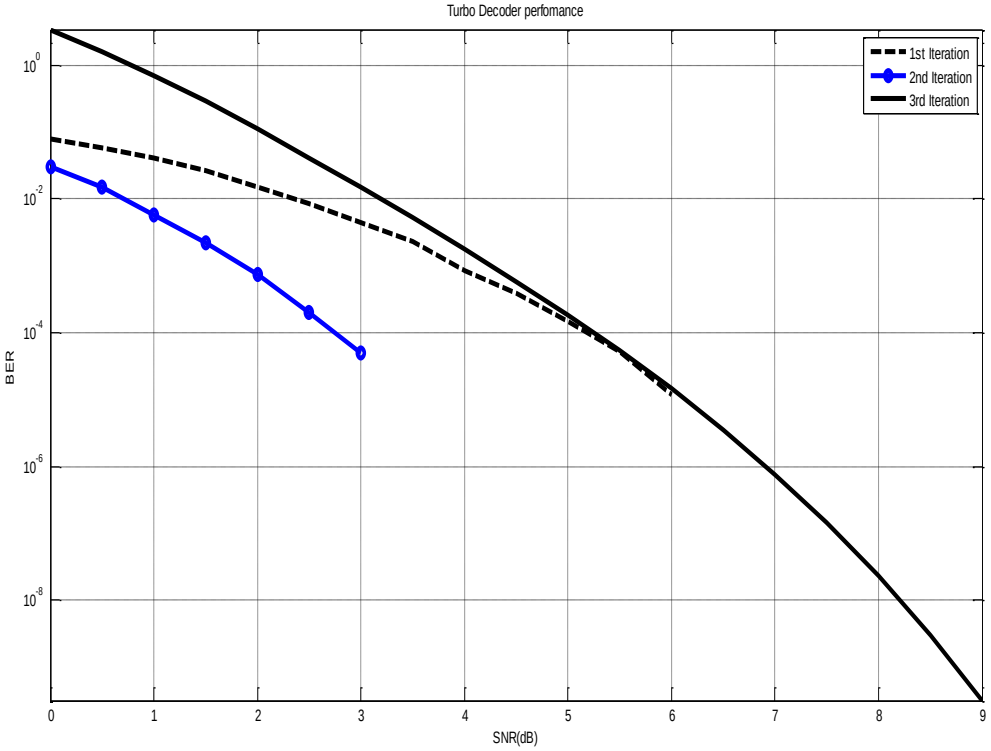

Figure 4.9 BER after second iteration

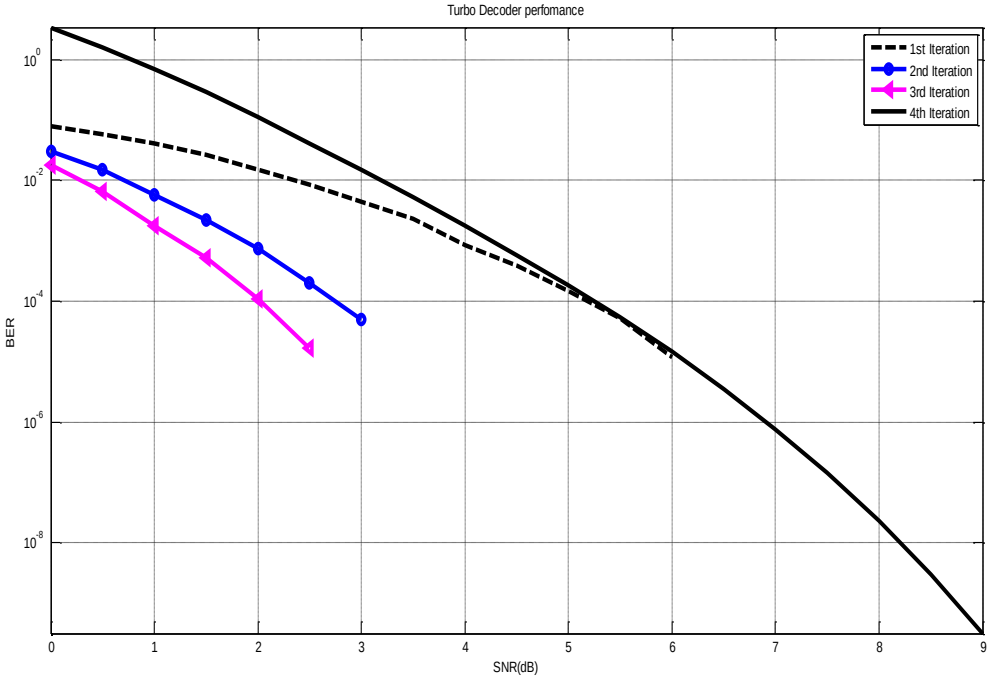

Figure 4.10 BER after third iteration

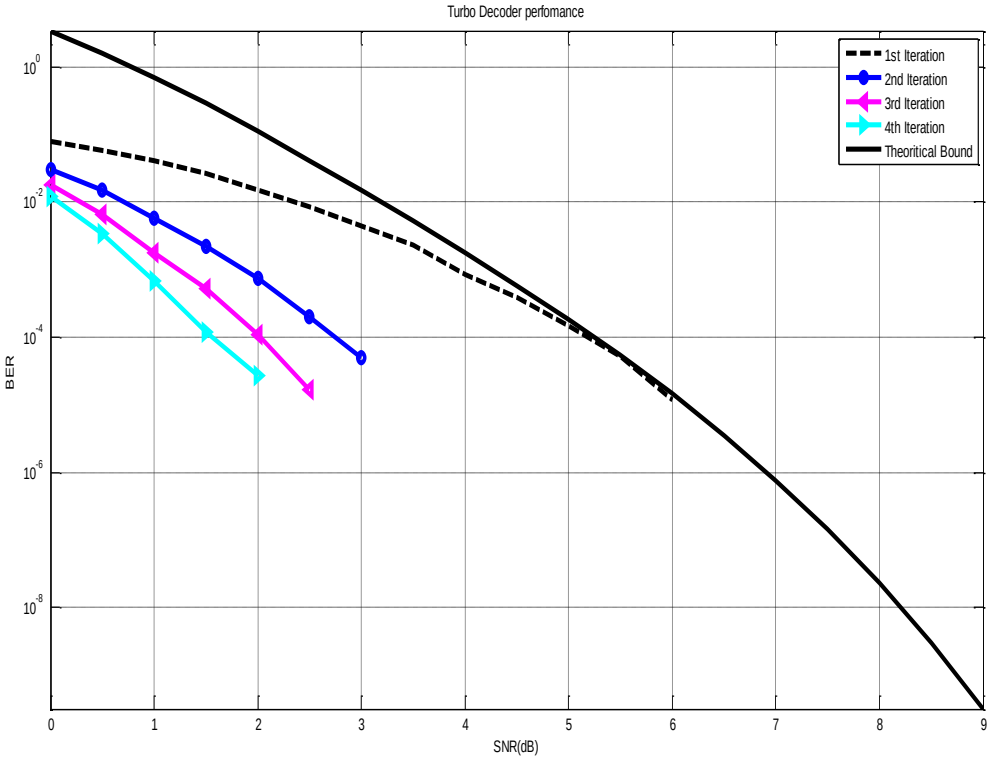

Figure 4.11 BER after fourth iteration

CHAPTER -5
RESULT AND DISCUSSION

In chapter 4 we have implemented the turbo coder and encoder using constant –log map algorithm. In this chapter we are comparing the results of implemented Turbo code with the base paper .In base paper after doing compression max log map algorithm is used whereas we use the constant log map algorithm which calculation is faster than max log map algorithm which increase the speed of decoding. As using compression the numbers of trellis stages reduce which increase the decoding as well as using constant max log algorithm further increase the speed of the decoder .hence the combination of both gives us an efficient decoder.

The comparisons of both the results are shown on the MATLAB.

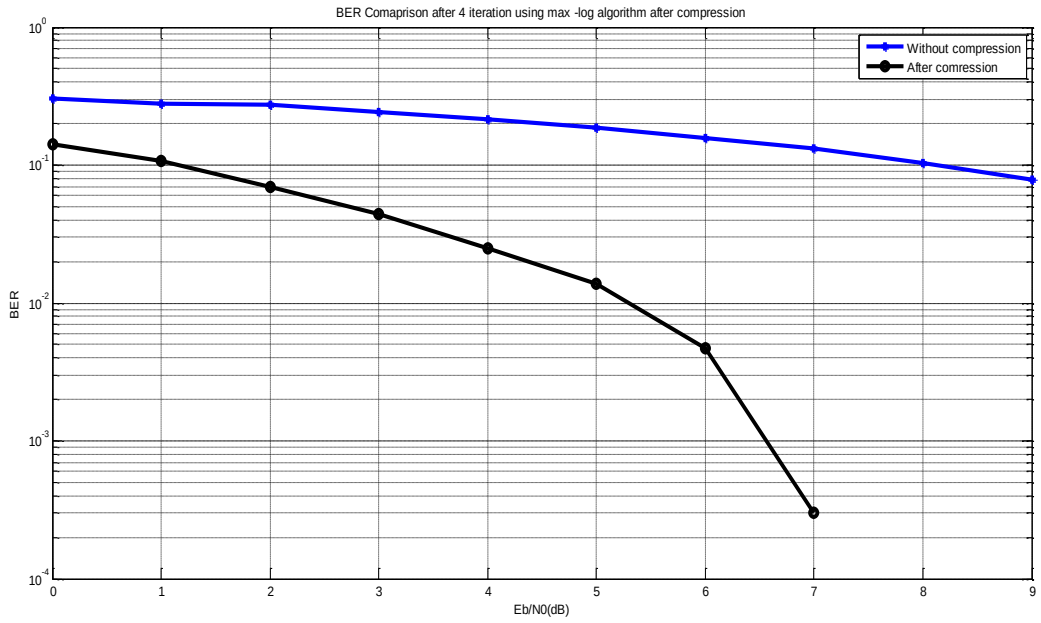

Figure 5.1 BER graph of compression of trellis with max log map algorithm after 4rth iteration

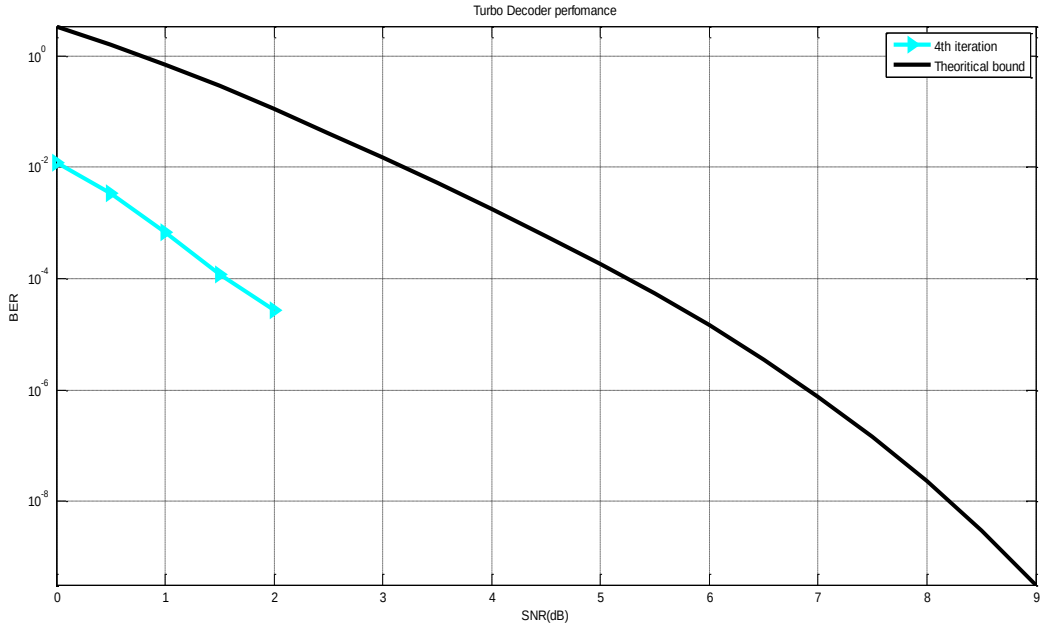

Figure:5.2 BER graph of compression of trellis stages with constant max log algorithm after 4rth iteration

SNR values (db)	USING MAX LOG MAP Algorithm	USING Constant MAX LOG MAP Algorithm
0	0.910	0.0129
0.5	0.0585	0.0034
1	0.0310	6.67*10^(-4)
1.5	0.0262	1.17*10^(-4)
2	0.0154	3.6*10^(-5)
2.5	0.0090	0
3	0.0045	0
3.5	0.0023	0
4	8.35*10^(-4)	0
4.5	6.22*10^(-4)	0
5	3.78*10^(-4)	0
5.5	8.22*10^(-5)	0
6	5.6*10(-5)	0
6.5	1.0*10^(-5)	0
7	0	0
7.5	0	0
8	0	0
8.5	0	0
9	0	0

TABLE 5.1 Comparison between MAX log map and constant max log map algorithm.

Hence it may be conclude that compression and constant log map algorithm give follwing improvement in the results:

- Better BER.
- High speed

CHAPTER 6
CONCLUSION & FUTRURE WORK

Comparative analysis of both turbo codes with compression and max-log map algorithm and turbo coding with compression and constant log map algorithm method has done in MATLAB. From the result taken from MATLAB for both the design it is clear that out of both algorithm the constant log map algorithm is best one as it is more susceptible to noise errors as compared to max log map algorithm. And combination of Bit agglomerated method for compression and constant log map algorithm give a fast and efficient decoding. In future as in communication system we are using turbo codes by using this combination will make error correction technique fast and efficient.

REFERENCES

[1].L.Bahl, J.Cocke, F.Jelinek, .Raviv, "Optimal decoding of linear codes for minimizing symbol error rate", IEEE Trans.Inf.Theory, vol.IT-20(2) pp.284-287, March 1974.

[2].C. Berrou and A. Glavieux, "Near optimum error correcting coding and decoding: turbo codes,"IEEE Trans. Comm., vol. 44, no. 10, pp. 1261–1271, Oct. 1996.

[3].A.J.Viterbi, "An intuitive justification and a simplified implementation of the MAP decoder for convolutional codes", IEEE J.Sel.Areas Commun, vol.16, Feb.1998, pp, 260-264.

[4].C.Weiss,C.Bettstetter,S.Riedel,D.J. Constello "Turbo decoding with tail bitting trellises" International Symposium on Signal System andElectronice,Pisa,Italy,1998,pp,343-348.

[5]. A.S.Barbulescu, S.S.Pietrobon, "Terminating the trellis of turbo code in the same state', Electronic letter, vol, 31, no.1, Jan.1995, pp.22-23

[6].O.Jearessen and H.Meyr, "Terminating the trellis of turbo code" Electronic letter 4th August 1994, vol.30.No.16 pp.1285-1286.

[7].J. Vogt and A. Finger, "Improving the max-log-MAP turbo decoder,"Elec. Letters, vol. 36, no. 23, pp. 1937–1939, Nov. 2000.

[8].C.Douillard, M.Jezequel, and C. Berrou, "The turbo code standard" Proc. 2nd Int. Symp. Turbo codes, pp. 551-554, Sept. 2000.

[9].J. P. Woodard and L. Hanzo, "Comparative study of turbo decoding techniques: an overview," IEEE Trans. Vehicular Tech., vol. 49, no. 6,pp. 2208–2233, Nov. 2000.

[10].M. Bickerstaff, L. Davis, C. Thomas, D. Garrett, and C. Nicol, "A24Mb/s radix-4 log MAP turbo decoder for 3GPP-HSDPA mobile wire-less," in IEEE ISSCC dig. tech. papers, vol. 1, San Francisco, CA, USA,Feb. 2003, pp. 150–484.

[11].E .Boutillon, C. Douillard, G. Montorsi, "Iterative Decoding of Concatenated Convolutional Codes: Implementation Issues'.IEEE transaction, vol.95, no.6, June2007.

[12].Norfeshab Ab.Wahab, Ir.Muhammad Ibrahim, Suzi Sereja Sarnim, Naimah Mat Isa, "Performance of hybrid automatic repeat request scheme with turbo codes "Proceeding of the International Scientists 2010 vol,2,March 17-19.2010.

[13].Mohammad Salim,R.P.Yadav,S Ravi Kanth, “Performance Analysis Of Log-Map,SOVA and Modified SOVA algorithm for Turbo decoder” International Journal of Computer Application vol,9,No.11 Nov2010.

[14].Christop Studer, Christan Benkeser Member, “Design and Implementation of a Parallel Turbo decoder ASIC for 3GPP-LTE “proceeding IEEE Journal 2011.

[15].Patel Sneha Bhanubhai Mary Grace S hajan, Upena D. Dalal “Performance of Turbo encoder and Turbo decoder for LTE” IJETT vol2, issue, 6, October 2012.

[16].Irfan Ali , “Bit Error Rate simulation using MATLAB “ vol,3, issue1,January-Feb 2013.pp.706-711.

[17].E.Boutillon.J.L. Sanchez-Rojas.C.Marchand, “Compression of redundancy free trellis stages in turbo decoder”, Elctronics Letter Vol.49, no.7, Feb.2013, pp.460-462.

[18].Rohan M.Pednekar, Dayanand B M “Design and Implementation of Convolutional encoder and Viterbi Decoder”, International Journal of Emerging Technologies in Computational and Applied Sciences, Nov 2013, pp.84-89.

[19].Vijaysinh Patil, P.C Latane “Implementation of efficient turbo code encoder-decoder with max-log-map algorithm”IJATER, 2014

[20].http://www.3gpp.org/ftp/Specs/html-info/36212.htm,version 10.0.0.

[21].“Information Theory, Coding and Cryptography”, “Ranjan Bose, second edition pp.191, 220.

[22].Moon, Todd K., Book Error correction coding : mathematical Methods and algorithms /Todd K ISBN 0-471-64800-0.