ISSN (Online): 2455-7838
SJIF Impact Factor (2017): 5.705

EPRA International Journal of

Research & Development

(IJRD)

Monthly Peer Reviewed & Indexed
International Online Journal

Volume: 3, Issue:8, August 2018

Published By :
EPRA Journals

SJIF Impact Factor: 5.705 Volume: 3 | Issue: 8 | August | 2018 ISSN: 2455-7838(Online)
EPRA International Journal of Research and Development (IJRD)

NOISE REDUCTION IN WIRELESS COMMUNICATION SYSTEM USING HIGH CONSTRAINT LENGTH CONVOLUTIONAL ENCODER AND VITERBI DECODER IN MATLAB

Nwangwu E.C. [1]
[1]Department of Electrical/Electronic Engineering, Akanu Ibiam Federal Polytechnic, P.M.B. 1007, Unwana. Ebonyi State, Nigeria

Abioye A. E. [2]
[2] Department of Electrical/Electronic Engineering, Akanu Ibiam Federal Polytechnic, P.M.B. 1007, Unwana. Ebonyi State, Nigeria

Oluwe M. O. [3]
[3] Department of Electrical/Electronic Engineering, Akanu Ibiam Federal Polytechnic, P.M.B. 1007, Unwana. Ebonyi State, Nigeria

Egonwa B. O. [4]
[4] Department of Electrical/Electronic Engineering, Akanu Ibiam Federal Polytechnic, P.M.B. 1007, Unwana. Ebonyi State, Nigeria

ABSTRACT

This Research focuses on modelling of convolutional encoder and viterbi decoder for the purpose of noise reduction in wireless communication system. In this work a convolutional encoder and viterbi decoder of constraint length of 7 and code rate of ½ was simulated using matlab software. A stipulated rate-compatible punctured Convolutional Codes (RCPC) from the usual mother rate of ½ with constraint length K = 7 were employed to obtain a higher rate of 2/3. The matlab software was used in the simulation of the wireless communication channel exbiting the convolutional encoder and viterbi decoder which was carried out over an additive White Gaussian Noise (AWGN) using Binary Phase Shift Keying (BPSK) modulation technique. The simulation was done across Signal-to-Noise Ratio (SNR) value between 0dB-6dB with one million input bits gotten from the data generator which is basically the least yardstick used by researchers to obtain a better Bit Error Rate (BER) performance. It was observed that the RCPC performed better than the normal code rate as when examined at the same rate and memory. Also the degree of computation for the two codes and the time taken for decoding each differ in favour of RCPC. In conclusion, the analysis showed that there was a huge reduction in transmit power by a factor of 4 for 2-bit soft decision quantization even though operations were carried out at the same BER of 0.12.

KEYWORDS: *MATLAB, Convolutional Encoder, Viterbi Decoder, Constraint Length, SNR, BER, AWGN, BPSK*

1.0 INTRODUCTION

1.1 Background

Communication system plays a major role in our daily life; people use cell phones, satellites, internet, and data transmission. All those appliances are used in an environment exposed to noise, also data might be transmitted for long distances. These effects could cause changes in data values causing data corruption and loss. It is therefore necessary for the telecommunication provider to reduce the data corruption by providing a suitable solution to the errors in the communication process [1].

Errors can occur in the form of fading, Inter-signal interference (ISI) or noise can also occur when data is transmitted across an impaired channel. Therefore, for an efficient and reliable data communication, the use of a method which can efficiently and effectively locate and correct errors; so as to help forward the standard established by IEEE must be applied [2].

The operations involved in locating and correcting errors in a communication system is called Channel Coding (CC).

According to [3], Convolutional encoders with Viterbi decoders are techniques used in correcting errors which are greatly deployed in communication systems to better the Bit Error Ratio (BER) performance.

Convolutional codes are linear codes over the field of one sided infinite sequences. Its usage is regularly seen in the correction of errors existing in a badly impaired channel due to their high affinity to error correction. These codes are majorly used in place of block codes when Forward Error Correction (FEC) is needed and have been registered to perform exceptionally well when run with Viterbi decoder which can be in the form of soft decision decoding or probabilistic decoding algorithm [4].

In the convolutional encoding techniques, the source encoder converts the signals meant to be transmitted from analogue to digital format. Redundancy in the signal is removed by source coding and the information is then further compressed or converted into a sequence of binary digits for onward storage or transmission

The information sequence is transformed by the Channel Encoder into encoded sequence. Also, the redundant information incorporated into the generated binary data at encoder for the purpose of removing noise could be accurately recovered at the receiving end. These binary data are generated by the source encoder from the source. Therefore, the information sequence stored in the source encoder is changed by the channel encoder to a discrete encoded sequence known as a codeword. By modulating the channel encoder, data stream for transmission coming from the channel encoder are converted into waveforms of time duration [5].

This research will focus on the modeling of convolutional encoder (in the absence of Reed-Solomon outer code) with a Viterbi Decoder for constraint length of 7 and bit rate of ½ using MATLAB as well as investigating its performance when exposed to an impaired channel like the Additive White Gaussian Noise (AWGN) channel.

There are always errors in every wireless communication channels due to noise which can come inform of fading, inter-signal interference (ISI) or errors [6]. These errors alter the transmitted signal which causes changes in the information and distorts the data transmitted over the channel. The problem arises when the received data has been corrupted which makes the information incorrect or incomplete, which may need retransmission.

However retransmission is no longer acceptable in some application especially with the advanced new technologies in communication systems, thus the need for convolutional encoder and viterbi decoder which can detect and correct errors in communication system came up. In this research, MATLAB shall be used in the modeling of convolutional encoder with viterbi decoder.

Using the MATLAB software as required and employing the knowledge of the analytical theory of the coding fundamental principles, the convolutional encoder and viterbi decoder was modeled.

1.2 Aim and Objective of the Study

The main objective of this research is to model convolutional encoder and viterbi decoder of constraint length of 7 and bit rate of ½ using matlab. Specifically, this study sought to:

(i) Analyze the basics of ½ rate and 2/3 rate convolutional encoder with Viterbi decoders as a channel error control technique.

(ii) Design a convolution encoder that can reduce or eliminate noise in a digital communication system using mat lab software.

(iii) Design a viterbi decoder that can decode the convolutionally encoded data to give the replica of the input data.

(iv) Developing a mathematical model for the system as well as optimizing the design for maximum output accuracy and high performance with minimum distortion.

2.0 MATERIALS AND METHODS

2.1.Materials

In this research, we explored the use of MATLAB in modeling of convolutional encoder with Viterbi decoders for next generation broadband wireless access systems.

Using the MATLAB software as required and employing the knowledge of analytical theory of the coding fundamental principles, the convolutional encoder and Viterbi decoder was modelled as shown in Figure 1.

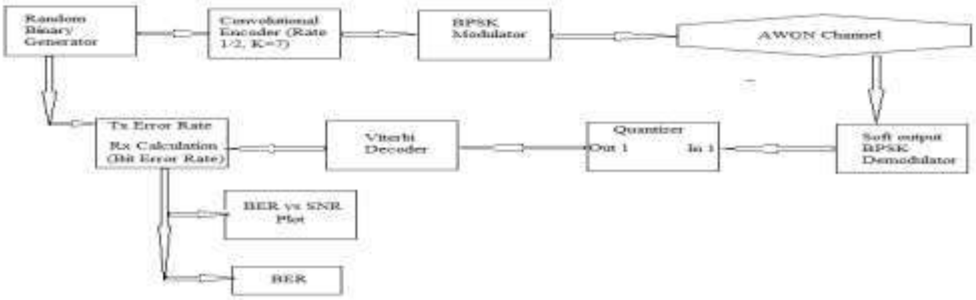

Figure 1: A communication system model block diagram exhibiting the Convolutional Encoder and Viterbi Decoder

The steps involved in simulating a communication channel using convolutional encoding and Viterbi decoding are as follows:

2.1.1 Generating the data

The data to be transmitted through the channel is generated using *randn* function of Matlab in combination with the sign function. We have generated 1000000 bits.

Below is a piece of MATLAB functional code cut out from the full code that performs this action. The multiple zeros sent in at the end of the sequence are used to flush out the bits.

```
clc;
clear;
Rate = 1/2;        %Rate can be 1/2, 2/3
N_Data_bits = 1000000;
Data_Input = [randi([0,1],[1,N_Data_bits-7]), 0, 0, 0, 0, 0, 0, 0];
```

2.1.2 Convolutionally encoding the data

Our convolutional encoder as shown in Figure 2 is made up of a data input generator, a pair of modulo-2 adder with corresponding pair of outputs (first and second) and 6 memory shift registers. A 'k' number of bits/second goes into the input and an 'n' output bits equivalent to '2k' symbols/second got for each output, thus giving a code rate value of 'k/n' = 1/2.

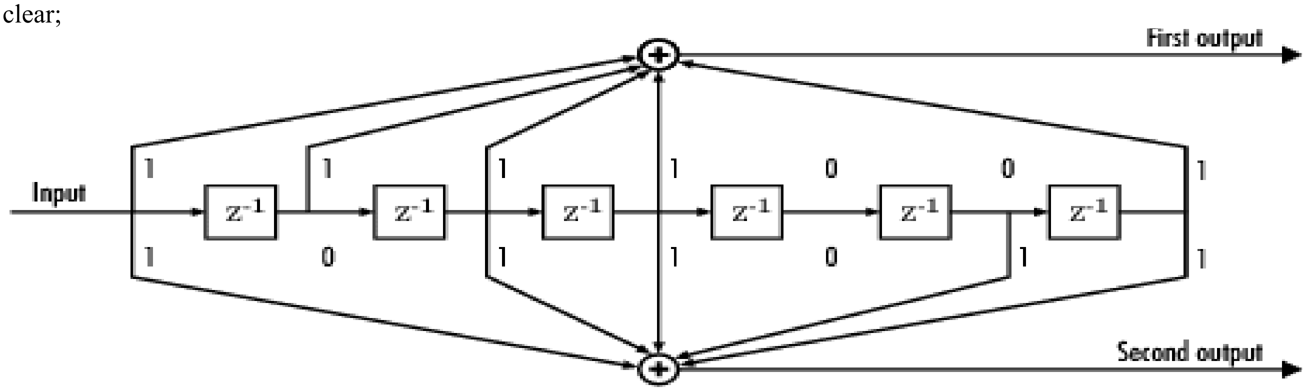

Figure 2: A convolutional Encoder of Rate 1/2, Constraint length 7.

Here, the best generation polynomial of [171, 133] octal for a convolutional encoder with rate 1/2 and constraint length K=7 has been determined. The constraint length 'K' here represents the number of shift registers that make up delay elements and the encoders present input.

Converting the generator polynomial of [177, 133] octal to binary, we have;

First output (g1) = 1 1 1 1 0 0 1

Second output (g2) = 1 0 1 1 0 1 1

For the convolutional encoder of Figure 2 to be made configurable in other to obtain from it the higher code rate of 2/3, as required in this project, a code puncturing technique was employed. This technique has a way of dropping some output bits based on the desired rate due to the fact that the encoder has been configured to output 2 symbols for every single input bit. This makes it possible to obtain the rates exhibited in the form of (n-1)/n.

2.1.3. The BPSK Modulator

The BPSK modulation technique is utilised here in modulating the transmitted data sequence. The 'zeros' and 'ones' got from the encoders output are mapped onto the antipodal baseband signalling scheme using the BPSK block maps, that is the 'zero' output values of the encoders are converted to 'ones (1)' and the corresponding 'ones' converted to 'negative ones (-1)'. This was actualized by carrying out a simple MATLAB iteration process involving the use of 'Modulated = 1 - 2*Code' equation on the encoders output. 'Code' represents the convolutional encoders output and 'Modulated' being the result of the modulation.

2.1.4. The AWGN Channel

In modelling the AWGN channel, we first of all generated Gaussian random numbers which were further scaled based on the transmitter energy per symbol in comparison to the noise density ratio, i.e. E_s/N_o. This is a function of SNR per bit, E_b/N_o and code rate, k/n which can be represented mathematically as:

$$E_s/N_o = E_b/N_o + 10\log_{10}(k/n) \quad [2.1]$$

For the code rate of an uncoded channel, $E_s/N_o = E_b/N_o$, making it equivalent to unity. Based on this finding, the rate 1/2 encoder exhibits an energy per symbol to noise density ratio of $E_b/N_o + 10\log_{10}(1/2) = E_b/N_o - 3.01\text{dB}$.

The uncoded signal over the AWGN channel has its theoretical BER written as

$$P_b = 1/2 \text{ erfc}\sqrt{\left(\frac{E_s}{N_o}\right)} \quad [2.2]$$

2.1.5. Demodulation

The Additive White Gaussian Channel gives out its sequence in a complex form ranging from 'negative ones' to 'positive ones' (-1 to +1) but this is not in the form the Viterbi decoder can act on it. Therefore, the function of the BPSK demodulator as employed here is to convert these complex data sequence to real data so it can be acted upon by the Viterbi decoder. The demodulator simply carries out on the complex data an operational function 'y = real(x) > 0' for the case of hard decision decoding and 'y = real(x)' for both cases of soft decision and un-quantized decoding.

2.1.6. Quantization

A perfect Viterbi decoder should be able to operate perfectly well with an infinitely quantized sequence, but unfortunately, this has a way of increasing the complexity of the Viterbi algorithm and data sequence decoding time, so a few bits of precision in practice is employed in the quantization of the channel symbol to checkmate this.

2.1.7 Viterbi Decoding the Encoded Data

Viterbi decoder modelling among the other elements in the whole system is the most tasking. Their modelling process involved some major stages which include: - De-puncturing, Branch Metric Computation BMU, Add-Compare and Select ACS, and finally the TraceBack Decoding TBD.

The block diagram of Figure 3 shows the processes.

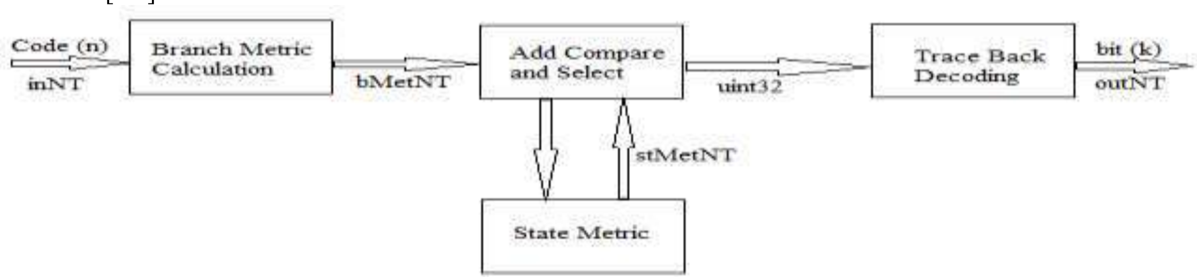

Figure 3: Viterbi Decoding Algorithm

Starting with de-puncturing, it makes use of the same puncturing matrixes used in the puncturing of data sequence for each code rate in the convolutional encoder to direct the Viterbi decoder on where to put 'dummy' (i.e. zeros) when decoding.

The Viterbi decoder implementation can be represented for easy understanding using a flow chart diagram as shown in Figure 4.

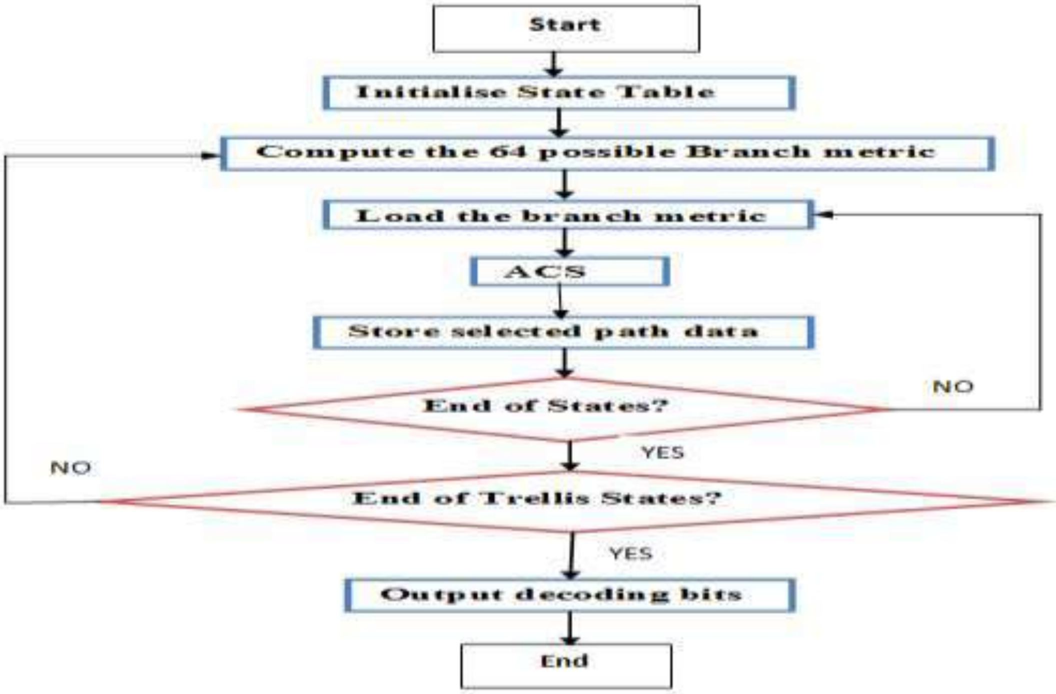

Figure 4: Flow diagram of Viterbi decoder implementation

2.2 Calculating the Bit Error

This calculation as handled by the responsible block compared the data sequence given out by the Viterbi decoder bit by bit with the sequence sent by the data generator such that if it discovered any bit from the decoded sequence to be different from the data sent in, it marks that particular bit as an error. Having done this for all the bit sequences, the whole cases of encountered errors are added up. The division of the total error summation by the total summation of the sent bits gives us our Bit Error Rate. Therefore,

BER = Total number of errors/ Total number of bits sent.

2.2.1 Model Testing

The testing of our model requires that all the system modelling steps shown in our communication system model block diagram be simulated using MATLAB software and the BER result plotted against SNR input. This model simulation was done across an SNR value between 0dB – 10dB with one million input bits got from the data generator which is basically the least yardstick used by many authors to obtain a10^{-5} BER performance.

Due to the numerous generated input bits, which lead to so many number of iterations taking place before simulating and sending out result, it was some how impossible to test for a BER above 10^{-8} as this is capable of taking several hours just to compute a single rate.

2.2.2 Error Performance Bound

We can determined the error performance bound for any rate of un-punctured 1/n convolutional code by calculating our estimation for the BER probability, P_b, of convolutional code for the un-quantised decision decoding given by;

$$P_b \leq= \sum_{d=f}^{\infty}(C_d P_d) \quad [2.3]$$

Such that $d = distance, C_d =$ simulation of bit errors at d, $f =$ codes free distance and $P_d =$ pairwise error probability. P_d is calculated using equation 3.4

$$P_d = \frac{1}{2}\operatorname{erf}\left(\sqrt{dR\,\frac{Eb}{N0}}\right) \quad [2.4]$$

With R as the convolutional encoder code rate, E_b/N_o as SNR and erf as the complementary error function,this has its equations as:

$$\operatorname{erf}(x) = \frac{2}{\sqrt{x}}\int_x^{\infty} e^{-t^2}\,dt \quad [2.5]$$

For the case of compactible rates of punctured convolutional code having rate r, given as r = (n-1)/n, the BER performance is bounded above by this Equation,

$$P_b \leq \frac{1}{2(n-1)}\sum_{d=f}^{\infty}(C_d P_d) \quad [2.6]$$

With P_d still retaining its value as shown in Equation 2.3

By computing the bit error probability, P_b for the values of Signal-to-Noise Ratio between 1dB to 10dB, the result acquired from simulating the 1/2 rate convolutional encoder and Viterbi decoder was plotted and analysis fully made and presented in the next section.

2.3. Implementation

2.3.1 Generating the data

The data to be transmitted through the channel was generated using randn function of Matlab in combination with the sign function.

Number of bits or symbols, N_Data_bits = 1000000

2.3.2 Data Encoding

ip = rand(1,N) [2.7]

Where ip is the code equation

2.3.3 BPSK modulator

The BPSK modulation technique was utilized in modulating the transmitted data sequence. We use the relation given below to achieve BPSK modulation

s = 2*ip-1 [2.8]

2.3.4 Additive white Gaussian noise.

The equation for Additive Gaussian noise is given as:

y = s + 10^(- Eb/No)*n [2.9]

2.3.5 Receiver

The demodulator simply carries out on the complex data an operational function 'y = real(x) > 0' for the case of hard decision decoding and 'y = real(x)' for both cases of soft decision and un-quantized decoding.

2.3.6 Theory of Ber

The equation for the theory of Ber is given by

theory Ber = Pb = 1/2 erfc$\sqrt{\left(\frac{Es}{N0}\right)}$ [2.10]

Eb/No as SNR and erf as the complementary error function

3.0 ANALYSIS OF RESULTS AND DISCUSSION

This section presents to us the whole results of simulations and findings encountered in the convolutional encoder modelling in which the Viterbi decoding algorithm was implemented as modelled in section 2.

Figure 5 presents the theoretical graph of the Convolutional Encoder which shall as well form the basics of our comparison.

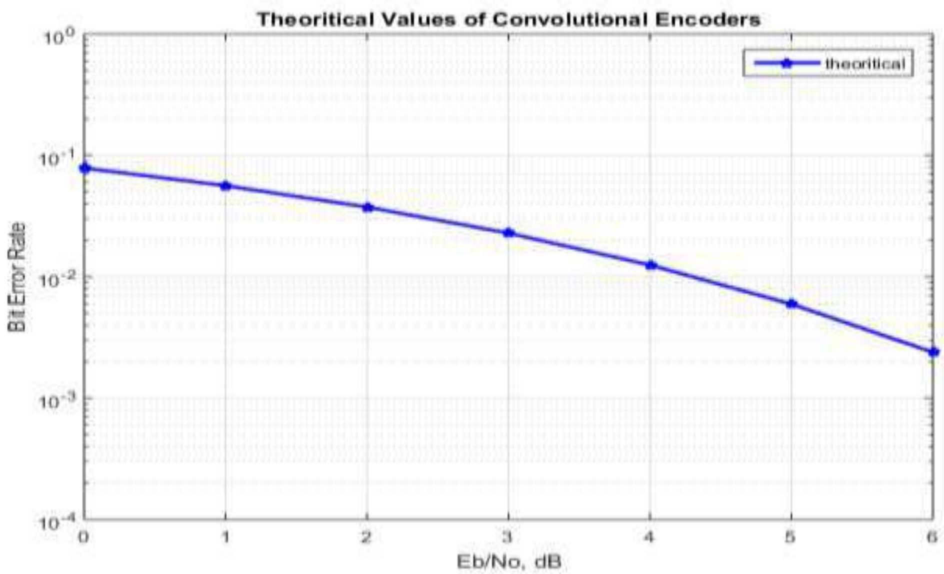

Figure 5: Theoretical graph of the Convolutional Encoder

3.1 Performance Analysis of Rate 1/2 with Consraint Length 7 Convolutional Encoder Exhibiting Soft and Hard Decision Decoding for different Quantization Widths

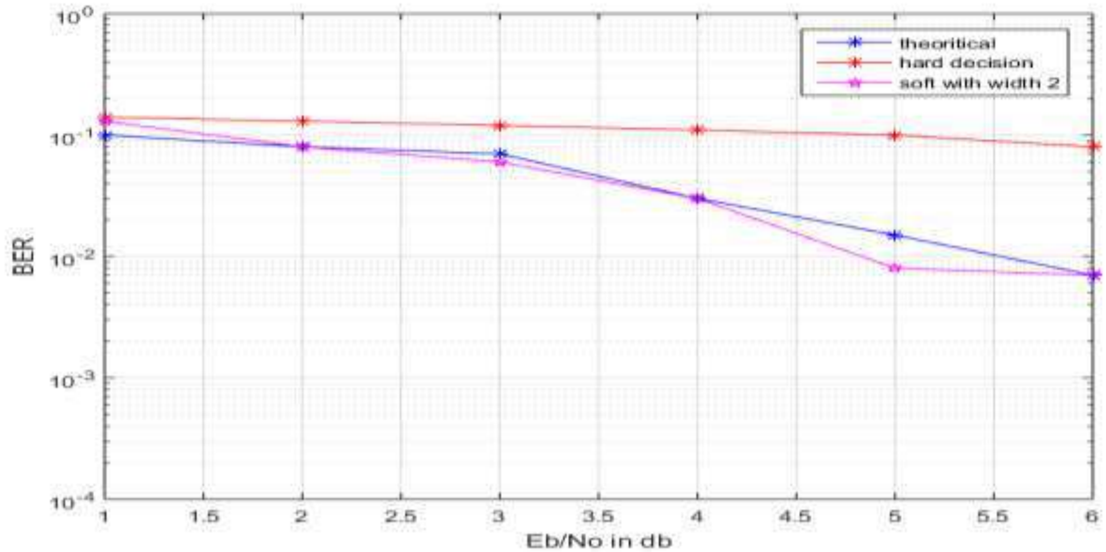

Figure 6: BER vs SNR curve of different quantization widths for rate 1/2 Binary Convolutional Encoder

In Figure 6, convolutional encoder data simulation was carried out on an input sequence of 1 million bits ranging from 0 to 6dB SNR values and 2.0 line spacing in other to obtain a good performance curve.

Measurement of the convolutional encoder and Viterbi decoder performance is anchored on the Bit Error Rate (BER) against Signal-to-Noise Ratio (E_b/No) in decibels. As can be seen from the graph label, the curve of the convolutional encoder of rate 1/2 and K=7, with Viterbi decoder using hard decision decoding of two-level quantization signals which is converted to only 'ones' and 'zeros' over an AWGN channel is marked with blue in Figure 5. Subsequently, curves of 2-bits soft decision and hard decision decoding are presented in the Figure 6 for comparison. The reference curve being the theoretical BER 'un-coded' is also present for use in the verification, comparison and analysis of the differences in the coding gain of the individual curves.

From the hard decision decoding curve, the coding gain in SNR at a BER of 0.12 presenting a decrease in the amount of transmitted power up to a factor of 4 in comparison with the theoretical signal. This transmit power decrease is recognized and implemented in communication systems to curb the excess cost encountered in the assembling of hardware, in effect to make room for a positive move towards the miniaturization of communication devices.

From Figure 6, it was also observed that when soft decision decoding was implemented, which involved the quantization of signals into levels order than just 'zeros' and 'ones', the gain received increased which meant that there was an improvement in the reduction of transmit power required. But one major set-back inferred here is that its implementation demands a more complex algorithm and sophisticated hardware.

In conclusion, we can justify from our observation that there was a huge reduction in transmit power by a factor of 4 for 2-bit soft decision quantization even though operations were carried out at the same BER of 0.12. It can be stated also that in this particular rate 1/2 convolutional encoder, there was a trade-off in the quantity of bandwidth needed to transmit the theoretical information in which it needs about a double amount of bandwidth at the same BER to do this, though the benefits of encoding the information bits before being transmitted far much exceeds this required bandwidth trade-offs.

3.2 Performance Analysis of Configurable Rate 2/3 Convolutional Encoder With Viterbi Decoder For different Quantization Widths Using Soft Decision Decoding

Using an input random sequence of 1 million bits for a range of 0 – 12dB, the curves obtained are as shown in Figure 7. It was observed that BER for each quantization width decreased exponentially with the increase in SNR. The coding gain of each of them at 0.12 BER showed some slight differences with the 4-bit quantization, though not showing much significant difference with the gain 2dB as exhibited by the 3-bit quantization width. There is also no doubt from the results obtained that the coded data curves exhibited a sharp fall unlike that of the un-coded, suggesting a better performance for the coded signals. Comparing the coding gain achieved for this configured rate with that of rate 1/2, it showed that an increase in the coding rate 'k/n' brings about a decrease in SNR gain. On the other hand, the percentage rate of bandwidth usage was seen to increase with the decrease in coding rate.

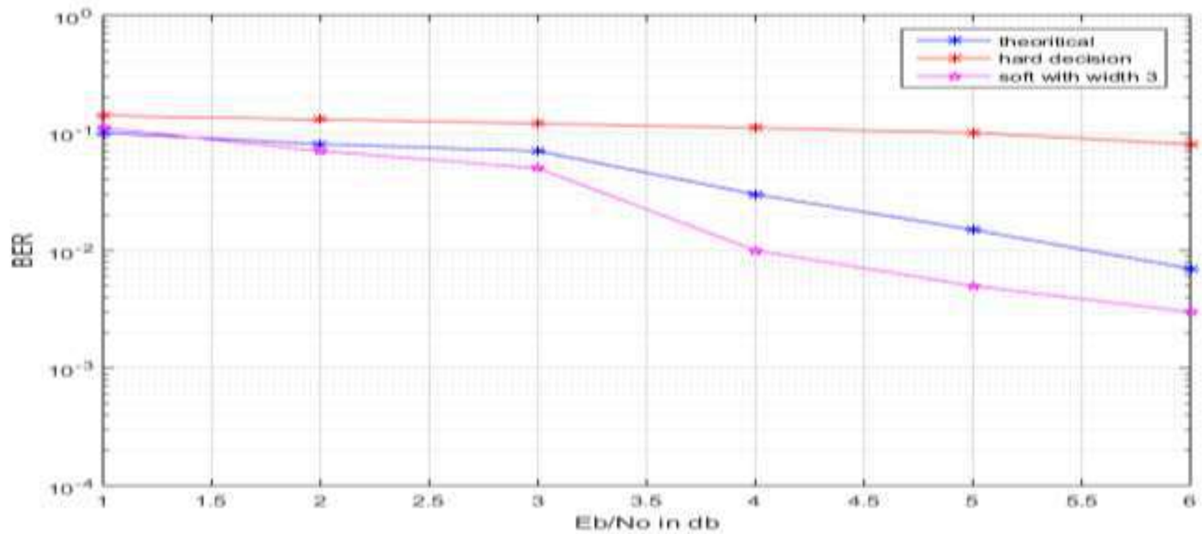

Figure 7: BER performance of rate 2/3 convolutional encoder showing curves for different quantization widths with soft decision decoding.

Also, Comparing soft decision coding with width of 2 and soft decision coding with width of 3, it can be seen from the graph that an increase in the coding rate 'k/n' resulted in a decrease in SNR gain of both. Also soft decision coding with width of 2 tends to have a better coding gain as compared to soft decision coding with width of 3.

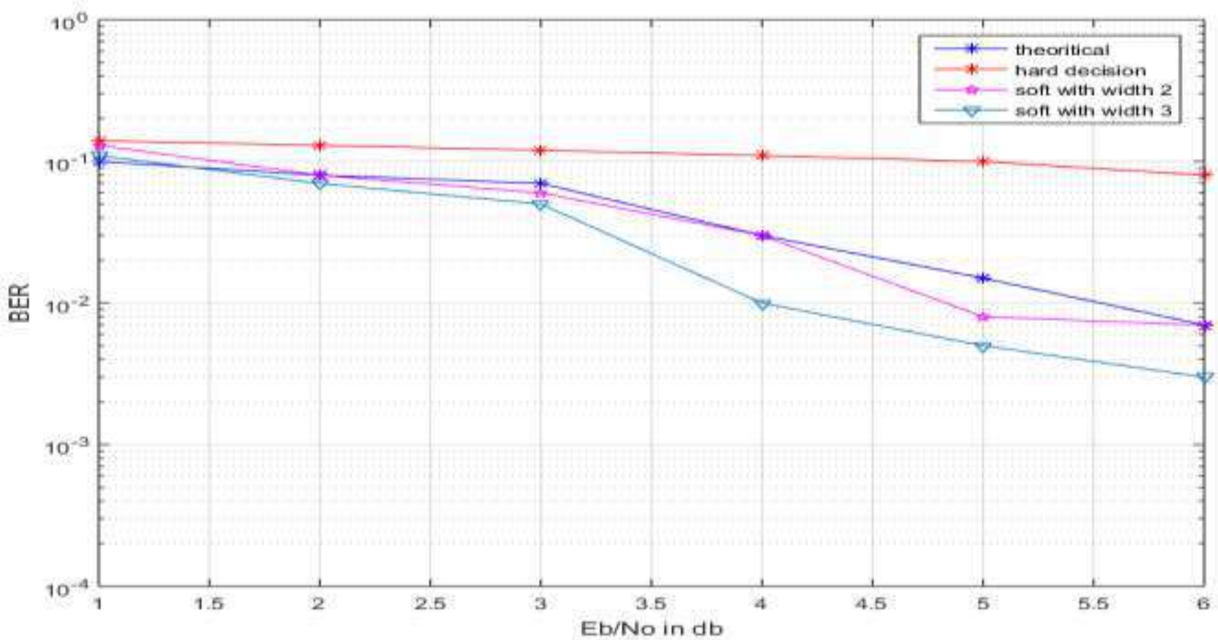

Figure 8: Comparison soft decision coding with width 2 and soft decision coding with width 3

4.0 CONCLUSION AND RECOMMENDATION

4.1 Conclusion

This research has carefully covered a modelling of configurable rate compactable convolutional encoder with Viterbi decoder from a mother code rate 1/2 and a constraint length 7 convolutional code from which other higher rate of 2/3 were further obtained with each exhibiting a low performance degradation when compared with the mother code. This modelling success was anchored on complementing the use of standard code puncturing matrixes in the convolutional encoder and using the 3-bits soft decision decoding as a yardstick in the Viterbi decoder that was modelled. The whole system performance results were proved using some already established error performance bounds standard in which the achieved results exhibited a tighter upper bound for the model.

The benefits of making use of rate-compatible punctured codes as against the normal mother rate code in which the justification of using the punctured codes have been proved to perform more than their normal code counterparts when examined at the same rate and memory having compared their degree of computation and duration taken for each decoding to stimulate. These established benefits were ascertained to increase with both the increase in SNR (E_b/N_o) and coding rates.

Based on this fact, though other channel Code schemes were analyzed, it was concluded that the Viterbi decoding algorithm still stands out when it involves the decoding of convolutional encoder which is very powerful in random error correction. The AWGN channel was used in the presence of BPSK modulation because of its characteristic nature of offering the best BER performance with a requirement of low transmitting energy.

4.2 Suggestion for Further Work

From the whole analysis of this work, the observations have proved that in using Viterbi decoder to decode the normal '1/n' code rate with K constraint length, a trace-back length of 'Kx5' or 'Kx6' will be fully enough for the Viterbi decoder to comfortably handle the received data symbol decoding without any noticeable performance degradation as against when comparison is made with a Viterbi decoder with an infinite memory. On the other hand, the punctured code rates demands a greater trace-back depth but one major set-back here is that no standard metric of calculation has been proved in determining the trace-back depth which will give complete information of the Viterbi decoder to fully decode the data sequence while keeping the decoding complexity at its barest minimum. Trace-back length have been discovered in the cause of this thesis to be a tool that figures out the amount of bit error rate that goes out of the performance bounds in the system. Therefore, based on these above observations, further studies is being suggested here to find a means of estimating the actual needed trace-back length that will produce an optimum performance of a convolutional encoder with Viterbi decoder for the punctured convolutional code rates.

REFERENCES

1. *Mike Rude (2003). Wireless Broadband Access and the Last Mile, A white paper, May 1, 2003*
2. *Anader Benyamin-Seeyar (2001). SC-FDE PHY Layer System Proposal for Sub 11 GHz Broadband Wireless Access (An OFDM Compatible Solution). Proposal for the 802.16 PHY, IEEE 802.16 Broadband Wireless Access working group, 2001*
3. *Ekwe, A.O. (2014). Performance Evaluation of convolutional Encoders with Different Constraint Rates. International Journal for Research in Applied Science and Engineering Technology (IJRASET), Vol. 2 Issue 8, August 2014.*
4. *Keattisak Sripimanwat (2005). Error Control Coding and its Applications, Lecture notes, Electrical Engineering, King's Mongkut Institute of Technology. 2005.*
5. *Lawrence, O. and Okomba, B. (2015) Analysis of convolutional Encoder with Viterbi Decoder for Next Generation Broadband wireless Access Systems. International Journal of Engineering and Technical Research (IJETR) Vol. 3, issue 4,*
6. *Abou-El-Azm, A (2010). 'Implementation of Viterbi Decoder' IEEE Trans. Communication., vol. 36, no. 4, pp. 389-400.*
7. *Wikipedia (2010). 4G [Available at: http://en.wikipedia.org/wiki/4G]*
8. *Wikipedia,(2011). The Future of Mobile Media Communication, (2011) 4G [Available at: http://www.business-intractive.com]*