Analysis of Convolutional Encoder with Viterbi Decoder for Next Generation Broadband Wireless Access Systems

Lawrence O., Okonba Brown

Abstract— **Due to the incessant demand for bandwidth by bandwidth application and digital communication equipment miniaturization, the need to design a good encoders and decoders for the next generation wireless communications system became very important. This research focuses on designing of convolutional encoder with Viterbi decoder for next generation broadband wireless access systems. We employed the stipulated rate-compactible punctured convolutional codes from the usual mother rate 1/2, constraint length K= 7 and generator polynomial [171, 133], to obtain higher rate of 2/3. Also, Matlab software was used in the Simulation of the model which was carried out over an Additive White Gaussian Noise (AWGN) channel using the Binary Phase Shift Keying (BPSK) modulation technique. These established benefits were ascertained to increase with both the increase in SNR (E_b/N_o) and coding rates. Also, we 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.**

Index Terms— **Convolutional Encoder, Viterbi Decoder, Matlab Software, Communication**

I. INTRODUCTION

Broadband Wireless Access (BWA) system is one technology that provides the users with an option to wired access such as Digital Subscriber Lines (DSL), fibre optic link and coaxial cable system with regards to coverage, speed, and capacity. It is suggested that Broadband Wireless Access system is capable of working efficiently in the 2 gigahertz – 11 gigahertz spectrum frequency aiming at 1000Mbit/s data rate for a fixed or slow dynamic user and 100Mbit/s for a high accelerating vehicle.

In the present scenarios, data transferring between the systems plays a vital role as the technologies are increasing day-by-day the number of users is simultaneously increasing. This wide usage leads to major issues in the digital communication systems and results in data corruptions. It's very necessary for the telecommunication to reduce the data corruption by providing a suitable solution to the errors occurred in the communication process [1]. Errors can occur in the form of fading, Inter-signal interference, ISI or noisy when data is transmitted across an impaired channel. Therefore, for the next generation BWA system to obtain an efficient and reliable data communication, it must employ the use of a method which can efficiently and effectively locate and correct errors; so as to help forward the standard established by IEEE for broadband wireless access systems[2].

The operations involved in locating and correcting errors in a Broadband Wireless Access (BWA) system is called Channel Coding (CC).

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. 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 [3]. The information sequence is transformed by the Channel Encoder into encoded sequence and redundant information incorporated into the generated binary data at encoder for the purpose of removing noise such that the sequential data can 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[4].

This research will focus on the analysis of convolutional encoder (in the absence of Reed-Solomon outer code) with a Viterbi Decoder for next generation BWA system as well as investigating its performance when exposed to an impaired channel like the Additive White Gaussian Noise (AWGN) channel.

Manuscript received April 24, 2015.

Lawrence O. Phd, Department of Electrical/Electronics Engineering, Michael Okpara University of Agriculture, Umudike.
Okonba Brown B.Eng, Department of Electrical/Electronics Engineering, Michael Okpara University of Agriculture, Umudike.

II. METHODOLOGY

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

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 3.1.

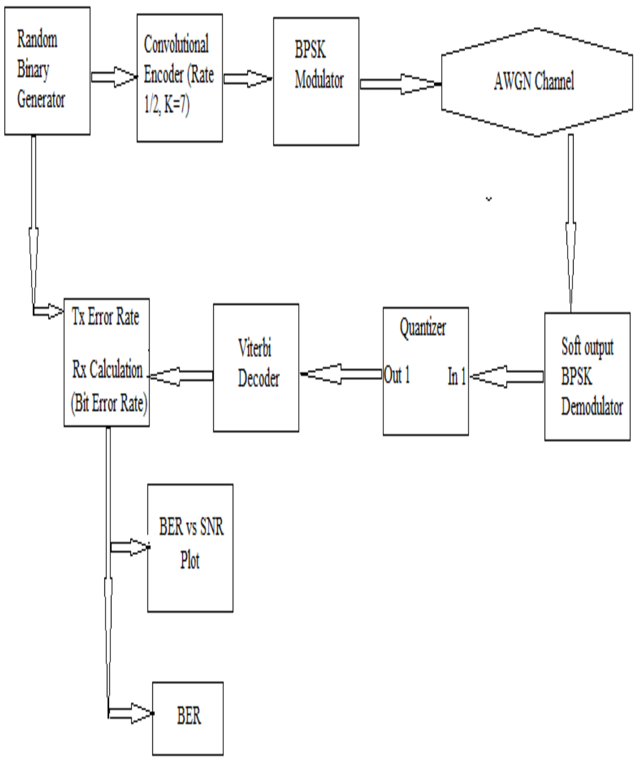

Figure1: A communication system model block diagram exhibiting the Convolutional Encoder and Viterbi Decoder

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

A. *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.Due to the randomness in the data generation, a different data array is got for each different simulation of the code, giving us a somewhat different plot though with each of the curves maintaining the same plotting trend due to the evenly distribution of the overall data [5].

B. *Convolutionally encoding the data:*

Our convolutional encoder as shown in Figure 3.2 below 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.

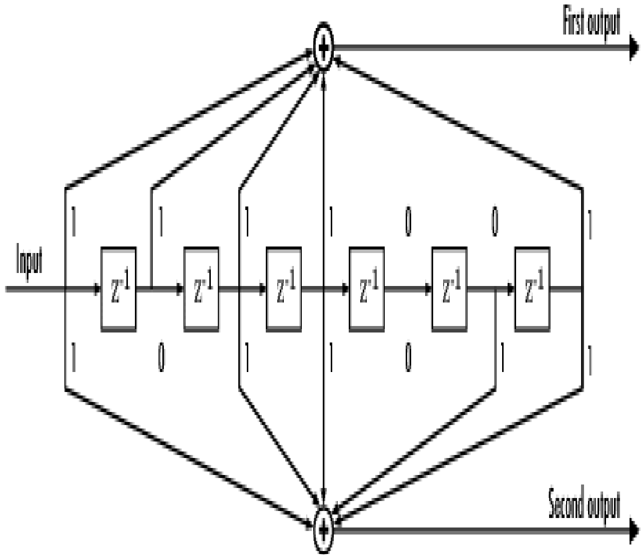

Figure2: A convolutional Encoder of Rate 1/2, Constraint length 7.

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 3.2 to be made configurable in other to obtain from it the higher code rate of 2/3, as required in this project, we had to employ the use of a technique known as code puncturing. 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.

The table above displays how a desired rate can be got from the mother rate 1/2 by simply using the puncturing matrix of each rate in the puncturing block. An exhibition of '1' means that the particular bit that corresponds to that '1' in a data sequence is sent while an exhibition of '0' is the opposite meaning that the corresponding bit has been punctured or discarded. A complete MATLAB function that performs these actions has been written and presented as appendix in the box with 'ConvEnc.m' for the convolutional encoderand 'Depuncture.m' for the puncturing block.

C. *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. By this we mean that the 'zero' output values of the encoders are converted to 'ones (1)' and the corresponding 'ones' converted to 'negative ones (-1)'. This is actualized by carrying out a simple MATLAB iteration process involving the use of 'Modulated = 1 - 2*Code' equation on the encoders output as shown in the box below. 'Code' represents the convolutional encoders output and 'Modulated' being the result of the modulation.

D. The AWGN Channel:

In modelling the AWGN channel, we first of all generated Gaussian random numbers which was 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:

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

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 $E_b/N_o + 10\log_{10}(1/2) = E_b/N_o - 3.01$dB.

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

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

E. 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.

F. 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. Since quantisation level can change from 1-signal bit to infinity, we have chosen 1-bit (for hard decision), 2-bit, 3-bit, 4-bit (for soft decision) and unquantized level for this work. Any bit less than or equal to zero is mapped to '0' and ones greater than zero mapped to '1' for the case of '1-bit' quantization level.

The input values for the '2-bit', '3-bit' and '4-bit' quantization is being set by the block from '0 to 2^n-1' where 'n' takes the values of '2', '3' and '4' for the respective bit decision decoding, making the numbers range from '0 – 3', '0 – 7' and '0 – 15' respectively. For '3-bit', the Viterbi decoder interprets '0' as the most confident '0' (strongest) and '7' as the most confident '1', while decision values lying between '0 – 7' are at extreme of the respective values.

G. Viterbi Decoding the Encoded Data:

Viterbi decoder modelling among the other elements in the whole system is the most tasking. Their modelling process involves 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 3.3 below shows the processes.

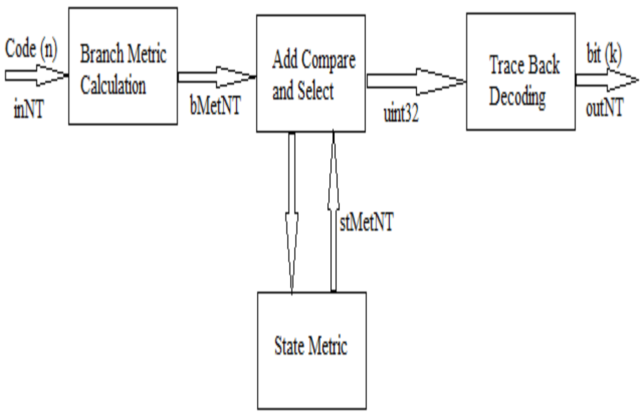

Figure3: Viterbi Decoding Algorithm

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 space between the inputs affected by noise and the ideal symbols are being calculated by the BMU. The ACS unit takes care of the state metric computation and transfers any of its decision or its chosen path into the trellis to the survival memory unit where it is stored [6].

In deciding which of the branch to choose, the ACS unit makes use of the maximum Euclidean decision metric to choose the right branch metric which must be the bigger branch metric between the two that shows up at every state. The TBD which often has a depth about 5 – 7 times (5K – 7K) the constraint length determines the survival memory unit length. Due to the fact that a lot of time is required to achieve the maximum likelihood path when inserting dummy bits, puncturing maintains on having a very large trace-back depth to achieve this.

The Viterbi decoder implementation can be represented for easy understanding using a flow chart diagram as shown below. This is self-explanatory.

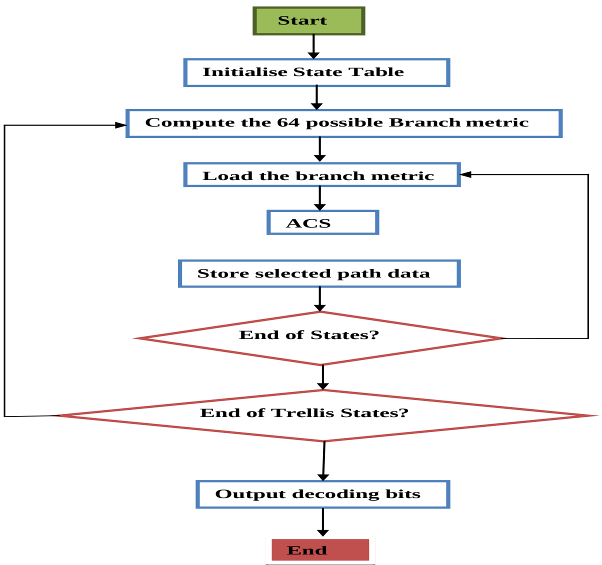

Figure4. Flow diagram of Viterbi decoder implementation

Calculating the Bit Error:

This calculation as handled by the responsible block compares the data sequence given out by the Viterbi decoder bit by bit with the sequence sent by the data generator such that if it discovers 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.

H. 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 somewhat impossible to test for a BER above 10^{-8} as this is capable of taking several hours just to compute a single rate.

I. Error Performance Bound:

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

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

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 an equation like

$$P_d = \frac{1}{2}\operatorname{erf}\left(\sqrt{dR\frac{Eb}{N_o}}\right)$$

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:

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

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,

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

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.

III. ANALYSIS OF RESULTS AND DISCUSSION

This result presents to us the whole results of simulations and findings encountered in the convolutional encoder analysis in which the Viterbi decoding algorithm have been implemented as modelled in the immediate preceding section.

Figure 5 below presents the theoretical graph of the Convolutional Encoder which shall as well form the basics of our comparison, while figure 6 presents Performance Analysis of Rate 1/2 with Consraint Length 7 Convolutional Encoder Exhibiting Soft and Hard Decision Decoding for different Quantization Widths.

In Figure 6 below, convolutional encoder data simulation was carried out on an input sequence of 1 million bits ranging from 0 to 14dB 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 the Figure 5. Subsequently, curves of 2-bits soft decision and hard decision decoding are presented in the same 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 0.14 presenting a decrease in the amount of transmit power up to a factor of 4 in comparison with the theoretical signal. This transmit power decrease is recognized and implemented in wireless 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 below, we 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 means 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 is 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 0.14. It can be stated also that in this particular rate 1/2 convolutional encoder, there is 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.

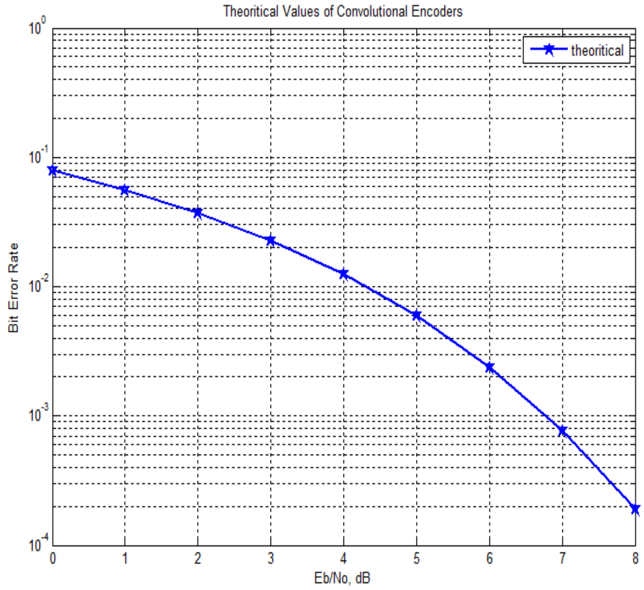

Figure5: Theoretical graph of the Convolutional Encoder

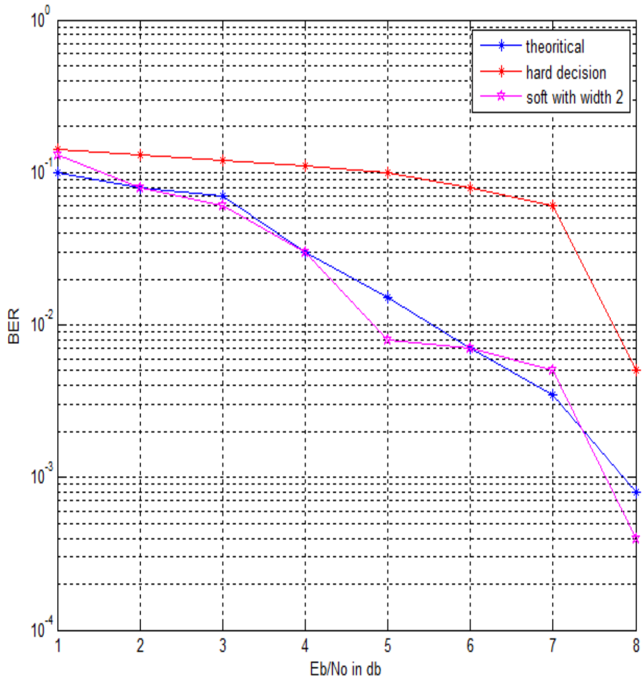

Figure 6: BER vs SNR curve of different quantization widths for rate 1/2 Binary Convolutional Encoder

Using an input random sequence of 1 million bits for a range of 0 – 14dB, the curves obtained are shown below in Figure 7. It is observed that BER for each quantization width decreased exponentially with the increase in SNR. The coding gain of each of them at 0.14 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 shows 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.

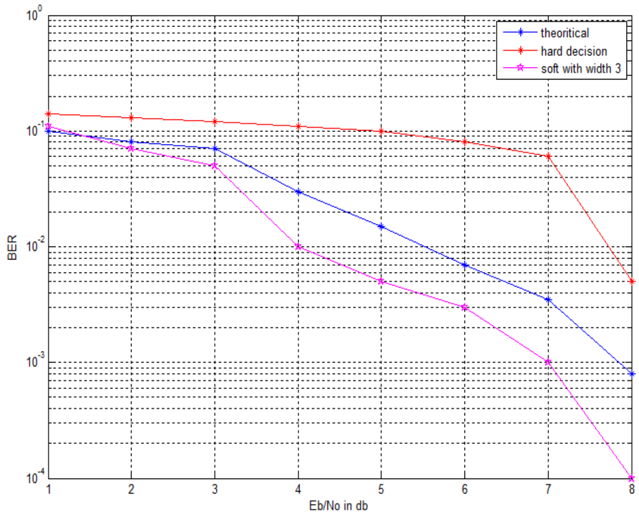

Figure7: BER performance of rate 2/3 convolutional encoder showing curves for different quantization widths with soft decision decoding.

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’ brings about 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

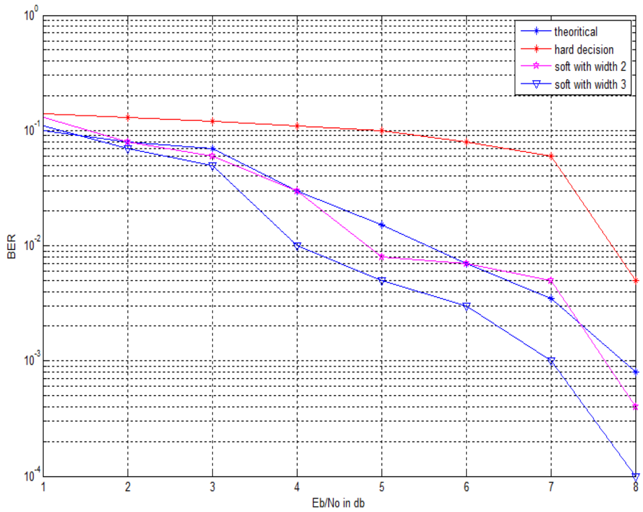

Figure8: Comparison soft decision coding with width 2 and soft decision coding with width 3

IV. CONCLUSION AND SUGGESTION FOR FURTHER WORK

A. Conclusion

This research have carefully covered analysis 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.

We have also penned down in this work 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.

All these processes were carefully followed in order to design a model that will checkmate the channel noise which constitute a barrier to achieving a the demands or set-up standard handed in by the IEEE 802.16 – 2009 for the next generation BWA system. Based on this fact, I analysed other CC schemes but came to a conclusion that 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.

B. Suggestion for Further Work

From my whole analysis of this work, my 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. Implementing a hardware aspect of this thesis modelled work can also spearhead a good area for further research work.

REFERENCE

[1]. Mike Rude, MMDS, Wireless Broadband Access and the Last Mile, A white paper, May 1, 2003.

[2]. Anader Benyamin-Seeyar, 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]. Keattisak Sripimanwat, Error Control Coding and its Applications, Lecture notes, Electrical Engineering, King's Mongkut Institute of Technology. 2005.

[4]. W. Chen. (2006), 'RTL Implementation of Viterbi Decoder', MSc. Thesis, Linköpings University

[5]. H. R. Anderson, "Fixed Broadband Wireless System Design". New York,
Wiley 2003

[6]. B Sklar, Digital Communications, Prentice-Hall International Editions, 1988.