Uniport Journal of Engineering and Scientific Research (UJESR)

Vol. 6, Issue 2, 2022, Page 120-130
ISSN: 2616-1192

(www.ujesr.org)

Modelling and Examining the Behaviour of Convolutional Encoders with Viterbi Decoders Over an Additive White Gaussian Noise Channel as a Function of Bit Error Rate and Signal-to-Noise Ratio

Agwu, O.E[1], Amadi, C.C[2] and Enebe, C.C[3]
[1&3]Department of Electrical/Electronic Engineering, Michael Okpara University of Agriculture Umudike, Nigeria.
[2]Department of Computer Engineering, Michael Okpara University of Agriculture, Umudike, Abia state.
*Corresponding author's email: agwu.ekwe@muouau.edu.ng

Abstract
This study examined the behaviour of convolutional encoders with Viterbi decoders over an Additive White Gaussian Noise (AWGN) channel as a function of Bit Error Rate (BER) and signal-to-noise ratio (SNR). This was done by employing compatible rate convolutional codes to achieve high data rate while keeping the bit error probability at its barest minimum. The study was centred on channel coding technique as stated by the IEEE 802.16 – 2009 standard for the next generation Broadband Wireless Access (BWA) system. Using MATLAB tool, the study investigated the performance of convolutional codes using Binary Phase Shift Keying (BPSK) in its model implementation. For rate ½ and subsequent rates, an input sequence of 1 million bits ranging from 0 to 10dB SNR values and 0.5 line spacing was used in other to obtain a good performance curve. Observation from the hard decision decoding curve showed that the coding gain in SNR at a BER of 10^{-5} rated about 3.2dB indicating a decrease in the amount of transmit power up to a factor of 2 when compared with the theoretical signal. Rate 2/3 had the best gain of 4.28dB. Rate 3/4 showed a coding gain of 3.4dB for 2-bit quantization width, 4.0dB for both 3-bit and 4-bit quantization, while rate 5/6 for 3-bit quantization measured a coding gain of 3.4dB and that of 4-bit quantization has 3.3dB coding gain. The results were validated using some already established error performance bound standards and obtained results showed a tighter upper bound for the model.
Keywords: *Additive White Gaussian noise, Bit error rate, Signal-to-noise ratio, Binary phase shift keying, Convolutional encoders*

Received: 3rd May, 2022 *Accepted: 29th June, 2022*

1. Introduction

Convolutional encoders with Viterbi decoders are techniques used in correcting errors which are greatly deployed in communication systems to better the BER performance (Chen, 2016). To understand and appreciate this algorithm, there is a need to describe the stages involved in the communication channel. Fig. 1 is a comprehensive block diagram of a complete classic communication structure.

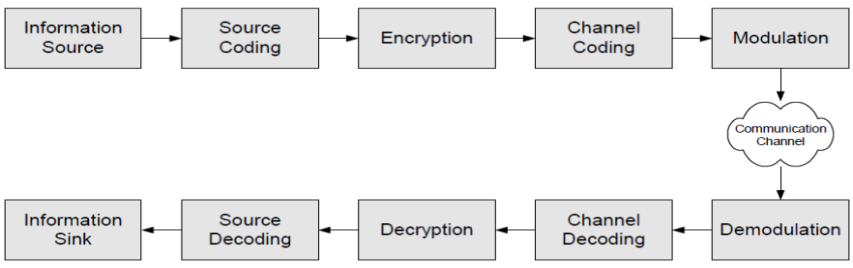

Fig. 1: Block diagram of a communication system (Zhu, 2011)

The primary function of the communication system is to transmit messages and data from a source to a designated point. The source encoder converts the signals meant to be transmitted from analogue to digital format. Redundancy in the signal is removed here by source coding and the

information is then further compressed or converted into a sequence of binary digits for onward storage or transmission. Encryption on its own is the process of adding redundancy for security purposes. 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 (KE-Lin et.al., 2010). By modulating the channel-encoder, data stream for transmission coming from the channel encoder are converted into waveforms of time duration t-seconds making it suitable to be transmitted over the communication channel which can either be a wired or wireless system. One function of a modulator is BPSK which is used for modulating high carrier frequency codeword. On entering the channel which is a physical medium used for information transmission, the waveform is affected by noise existing in the channel which could be in form of thermal, crosstalk and switch impulse. This waveform is then reduced to a sequence of numbers by the demodulator in which the acquired channel errors are processed and an error free output sequence produced which matches the estimated transmitted data symbols and termed the received sequence. The channel encoder tries to reproduce the main signal sequence by using the knowledge of code employed by the channel encoder and the redundancy present in the received signal. The source decoder receives the output information from the channel encoder and translates this sequence of binary digits into an estimate of the output source which it then forwards to the information sink. Information sink stand as the final destination to the main signal that was transmitted (Anderson, 2013).

This study concentrated basically on dealing with channel coding and decoding taking cognisance of convolution encoders and Viterbi decoding of the transmitted information.

2. Materials and methods

2.1 Method adopted

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 Fig. 2.

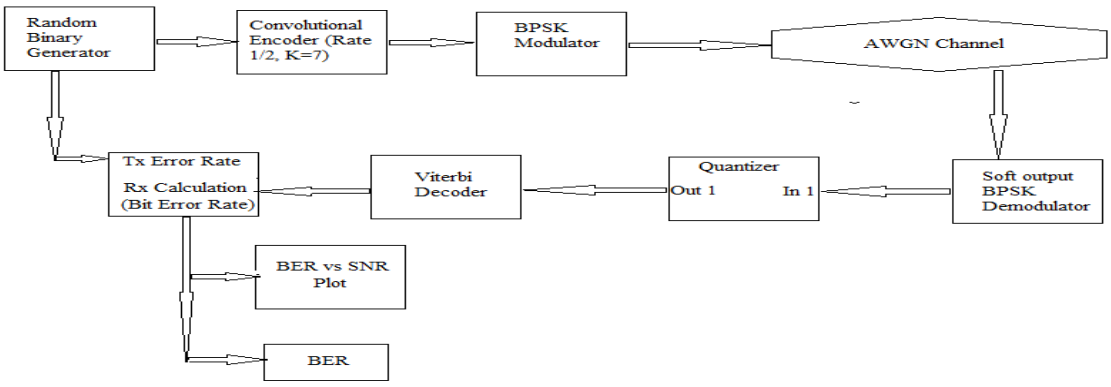

Fig. 2: A communication system model block exhibiting the Convolutional Encoder and Viterbi Decoder (Kai *et.al.*, 2011)

2.2 Specifications for modelling

To ensure proper modelling implementation, the following materials were employed in the order as enlisted below:

a. A binary convolutional encoder of rate 1/2 code, 6 memory storage units, constraint length K of 7, 2/3, 3/4 and 5/6 configurable rates with generator polynomial of [171, 133] octal.
b. A soft input Viterbi decoder to take care of the convolutional encoder in 'I'.
c. BPSK modulation technique.
d. A binary random data generator as information production unit which should be able to hand in at least 5 million bits of information so as to account for a useful BER data.
e. An SNR bit E_b/N_o of 0dB to 10dB.

2.3 Analysis of modelling

The random data generator was modelled using a simple MATLAB random number generation

functional code known as 'rand ()'. This was used to generate the information sequences sent into the convolutional encoder. The maximum value of the generated data was specified as '1' such that any value greater than half or equals the maximum value was tagged as '1' and values equal to or less than half the maximum was tagged as '0' making it possible to generate an equal probability of 'zeros' and 'ones'. The maximum generated data length was made up of 1 million bits so as to obtain a good performance and smooth curve. The steps involved in the coding of the IEEE 802.16 – 2009 standard channel included an application in a sequential order mode of Randomization, Forward Error Correction (FEC) and Interleaving at transmission and equivalent operations applied equally at the receiving end. The FEC adopted for this standard comprised concatenation of a Reed-Solomon outer code and a rate-compatible convolutional inner code. Increasing both the code rate 'k' and constraint length 'K' automatically increased the complexity of the encoders. This was taken care of by puncturing the mother code rate (e.g. rate ½ for this study).

The Convolutional data encoding as shown in Figure 3 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 went 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. Here, the best generation polynomial of [171, 133] octal for a convolutional encoder with rate 1/2 and constraint length K=7 was determined. The constraint length 'K' here represents the number of shift registers that make up delay elements and the encoders present input. For the convolutional encoder of Figure 3 to be made configurable in other to obtain from it the higher code rates of 2/3, 3/4 and 5/6 as required in this study, code puncturing was employed. Puncturing is a technique employed to generate higher code rates from a mother code rate which has an advantage of allowing us to make use of the same hardware in the implementation of several code rates thereby reducing the production cost of both encoders and decoders. This technique has a way of dropping some output bits based on the desired rate because the encoder has been configured to output 2 symbols for every single input bit. This made it possible to obtain the rates exhibited in the form of (n-1)/n. Table 1 shows a standard puncturing matrix for a rate 1/2, K=7 convolutional encoder.

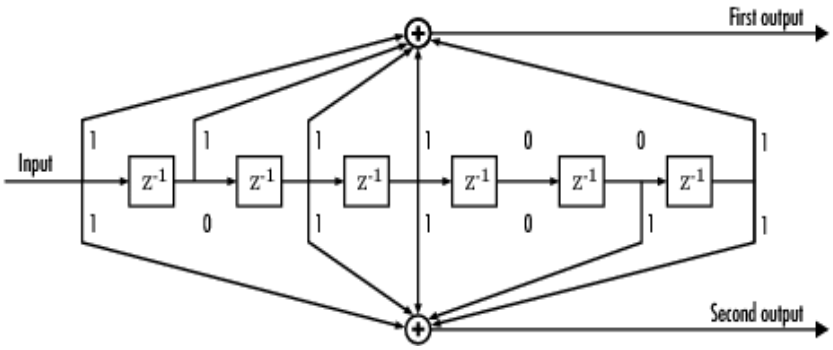

Fig. 3: A convolutional encoder of rate 1/2, constraint length 7 (Branka, 2019)

Table 1: Puncturing matrixes and free distance for the code rates used (Ying, 2016)

Code Rate	Puncturing Matrix	Free Distance
Rate 1/2	1 1	10
Rate 2/3	1 0 1 1	6
Rate 3/4	1 0 1 1 1 0	5
Rate 5/6	1 0 1 0 1 1 1 0 1 0	4

Table 1 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. 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. This means that the 'zero' output values of the encoders are converted to 'ones (1)' and the corresponding 'ones' converted to 'negative ones (-1)'. This was actualised by carrying out a simple MATLAB iteration process involving the use of 'Modulated = 1 - 2*Code' equation on the encoders output. In modelling the AWGN channel, generation of Gaussian random numbers was done 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 shown in Equation (1) (Abou-El-Azm, 2019).

$$E_s/N_o = E_b/N_o + 10log_{10}(k/n) \quad (1)$$

For the code rate of an uncoded channel, $E_s/N_o = E_b/N_o$, making it equivalent to unity. Based on this, 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 as shown in Equation (2) (Hagenauer, 2018).

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

The AWGN 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 (Christopher, 2001). 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 (Fu-hua 2017). Since quantisation level can change from 1-signal bit to infinity, therefore, 1-bit (for hard decision), 2-bit, 3-bit, 4-bit (for soft decision) and unquantized level has been chosen for this study. 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.

2.4 Viterbi decoding the encoded data

Viterbi decoder modelling involves some major stages which include: De-puncturing, Branch Metric Computation BMU, Add-Compare and Select ACS, and Trace Back Decoding TBD. The block diagram of Fig. 4 shows the processes. 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. 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. 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**.**

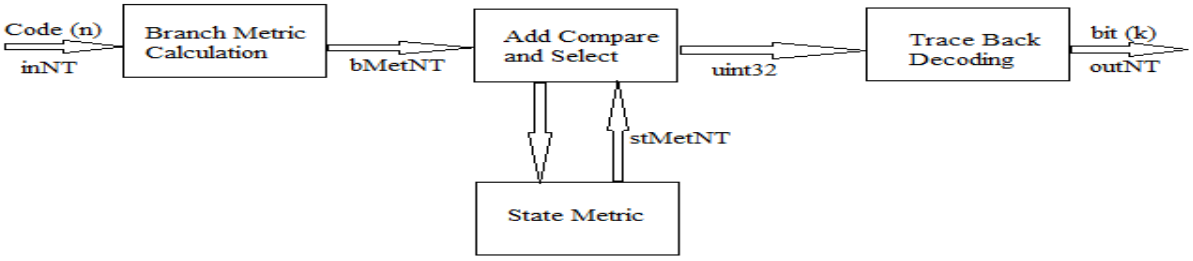

Fig. 4: Viterbi decoding algorithm (Forney, 2000).

3. Results and discussion

3.1 Performance of rate 1/2 exhibiting soft and hard decision decoding for different quantization widths

In Fig. 5, convolutional encoder data simulation was carried out on an input sequence of 1 million bits ranging from 0 to 10dB SNR values and 0.5 line spacing in other to obtain a good performance. It is observed from the hard decision decoding result that the coding gain in SNR at a BER of 10^{-5} rated about 3.2dB presenting a decrease in the amount of transmit power up to a factor of 2 in comparison with the theoretical signal. Observation also showed 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.

When 2-bit soft decision decoding which involves 4-levels (00, 01, 10, 11) of quantization was implemented, an additional gain of 1dB SNR at a BER of 10^{-5} was got in comparison to when hard decision decoding was used. This also improved to 2.1dB when 3-bit soft decision decoding was implemented and compared with that of hard decision decoding. On increasing to a 4-bit soft decision decoding, little or no significant change was recorded when compared with its 3-bit counterpart. So, from our observations, one can justify that there is a huge reduction in transmit power by a factor of 3 for 3-bit soft decision quantization even though operations were carried out at the same BER of 10^{-5}.

3.2 Performance of rate 2/3 convolutional encoder for different quantization widths using soft decision decoding

Using an input random sequence of 1 million bits for a range of 0 – 10dB, the curves obtained are shown in Figure 6. It is observed that BER for each quantization width decreased exponentially with the increase in SNR. The coding gain of each of them at 10^{-5} BER showed some slight differences with the 4-bit quantization having the best gain of 4.28dB, though not showing much significant difference with the gain 4.25dB 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. This can be justified using the 3-bit quantization width for both rates where rate 1/2 '3-bit' quantization exhibited a coding gain of 5.3dB while the configured rate 2/3 '3-bit' quantization exhibited that of 4.25dB. On the other hand, the percentage rate of bandwidth usage was seen to increase with the decrease in coding rate.

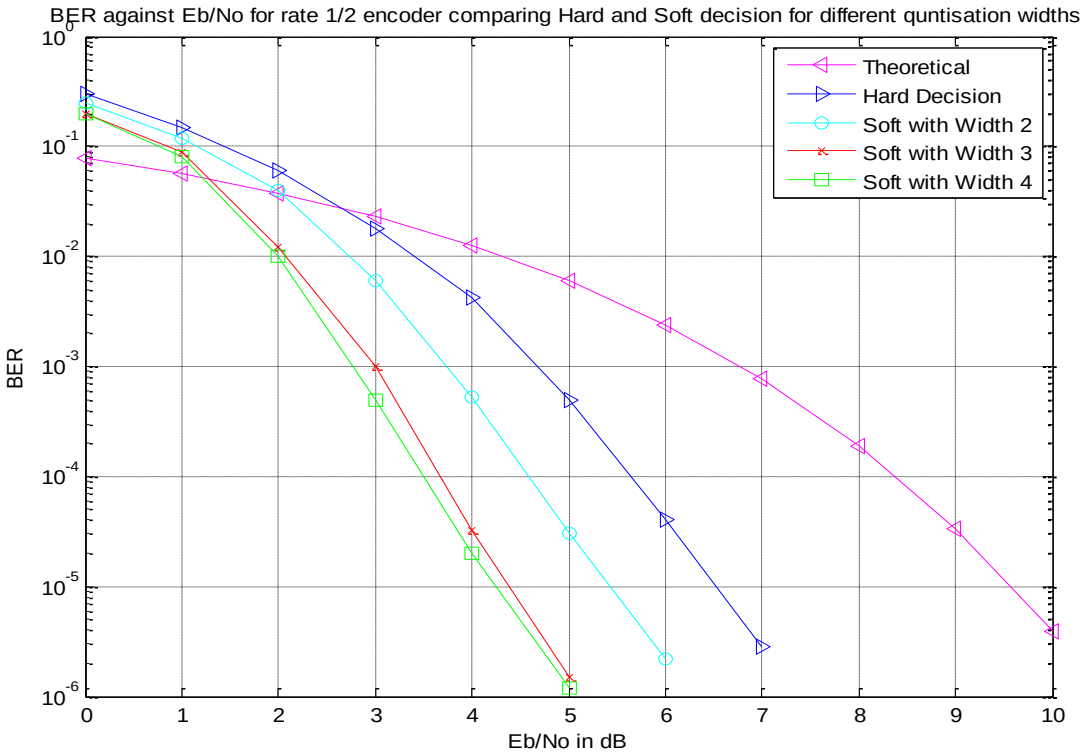

Fig. 5: Performance of different quantization widths for rate 1/2 for both hard and soft decision decoding using BPSK modulation.

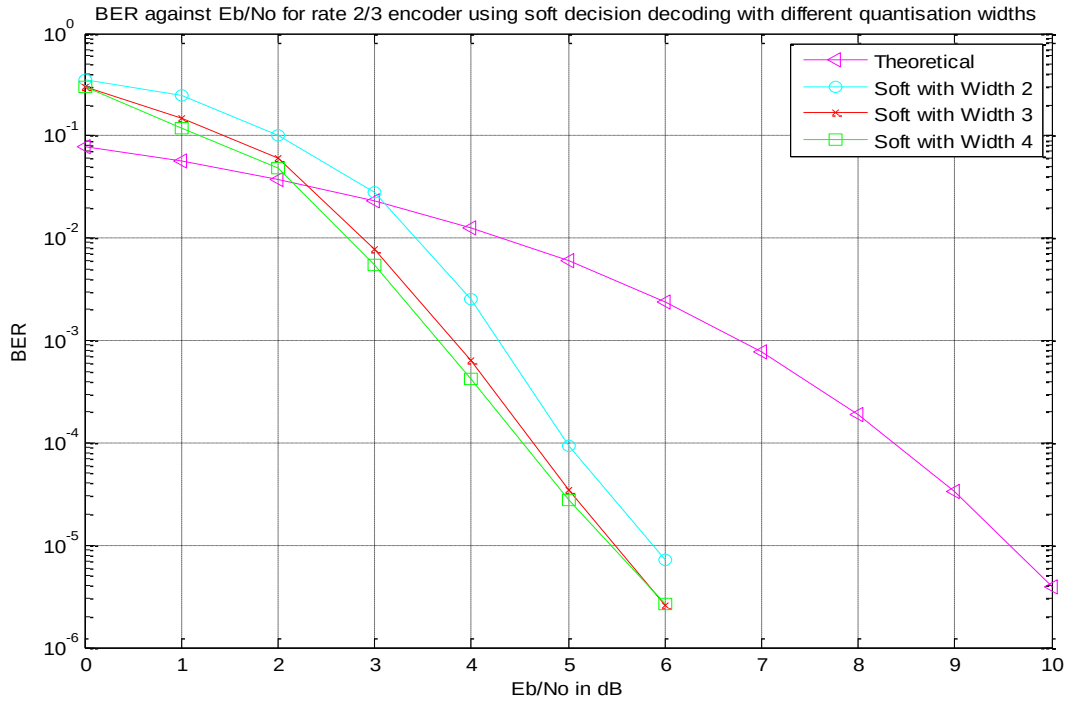

Fig. 6: Performance of rate 2/3 convolutional encoder showing different quantization widths with soft decision decoding.

3.3 Performance of configurable rate 3/4 convolutional encoder for different quantization width using soft decision decoding

In Fig. 7, the coding gain showed that the 2-bit quantization width exhibited a coding gain of 3.4dB at a BER of 10^{-5}, 3-bit quantization width showed a 4.0dB value at the same BER of 10^{-5}, while the 4-bit quantization gave the same value as that of the 3-bit quantization. This indicates that an increase in the quantization bit after the third bit does not really have a significant effect or much better BER performance but rather takes a longer computational time to simulate and produce results. Comparing the result got for this rate with the previous mother rate 1/2 and configurable rate 2/3, it is noticed that the BER performance of each individual curve in comparison with its counterpart in the previous rates was far much better. This was measured from the points the corresponding curve cuts the theoretical curve. It showed that as the rate 'k/n' increases, the plots for the coded data falls faster indicating better error correction properties which mean a better BER performance.

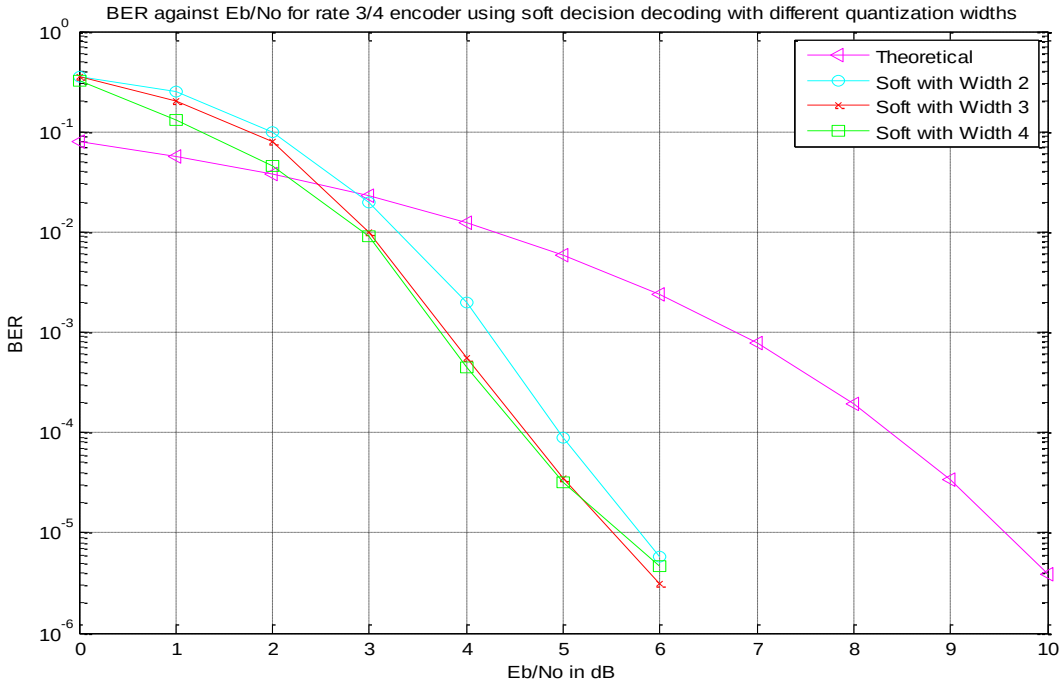

Fig. 7: Performance of rate 3/4 convolutional encoder for different quantization width with soft decision decoding.

3.4 Performance of rate 5/6 convolutional encoder for 3-bits and 4-bits quantization widths using soft decision decoding

Fig. 8 presents us with the best performance compared with the previous rates discussed earlier. The Bit Error Rate curves at 10^{-5} BER for 3-bit quantization measured a coding gain of 3.4dB while that of 4-bit quantization has 3.3dB coding gain. This does not show any significant difference. Still leaving the data sequence at 1 million bits and a range of 0 – 10dB SNR to get a better BER performance, a simulation was carried out using a less number of bit sequence of a hundred thousand (100,000) bits, though graph is not shown here, observation showed that reducing the data sequence while keeping the SNR range constant reduces the BER performance when taken at the same value though the computational speed tends to increase. Therefore, in order to get a better BER performance, the computational speed is traded off for both lower and higher code rates. This final rate results confirmed all our previous observations as it still indicates that an increase in code rate 'k/n' brings about an increase in the BER performance and as SNR increases, the BER drops at each BER point taken into consideration.

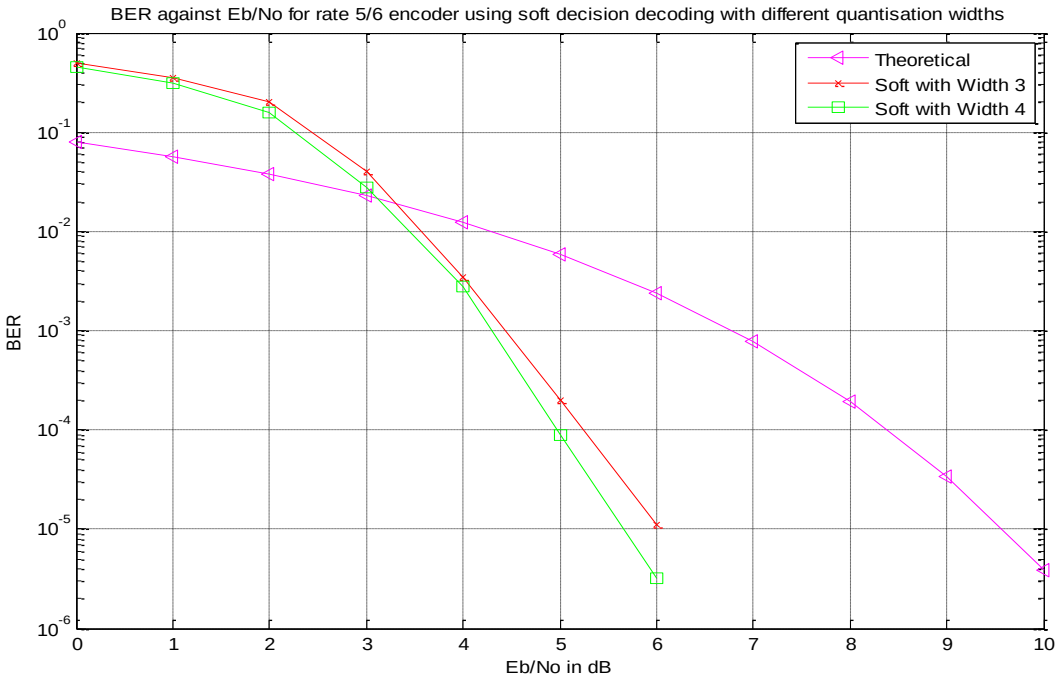

Fig. 8: Performance of rate 5/6 convolutional encoder for 3-bit and 4-bit quantization widths with soft decision decoding

3.5 BER performance for rates 1/2 and punctured rates 2/3, 3/4 and 5/6 using soft decision Viterbi decoder

Presented in Fig. 9 is the error bound plots of the code rates 1/2, 2/3, 3/4 and 5/6 for an un-quantized AWGN channel. From the error bound plots, we can certify that the significance of performance deterioration as we move from rate 1/2 to rate 5/6 was not well noticed. This we can justify clearly by plotting the performance bound alongside the BER on the same graph for the individual code rates.

A glance at the results of Fig. 10 - 13 show that the BER for our model strictly adheres to the theoretical performance bound. BER values that are lower than 10^{-3} gave results that hardly exceeded the bounds for the un-quantized decoding which validated the result as bounds are generally made specific for infinitely fine quantization level. Moreover, the best performance is got from the un-quantized decoding for convolutional encoder with Viterbi decoding though of very small increase in performance. This singular reason gives the 3-bits quantization decoding an edge over other quantization bits hence its wide application in mobile phone, digital television, VSAT, to mention but a few. Having proved from the obtained result that a minute difference in SNR exists between the 3-bit soft decision decoding and the un-quantized decoding, it has on that note been suggested to be a preferable choice to actualizing a next peak performance.

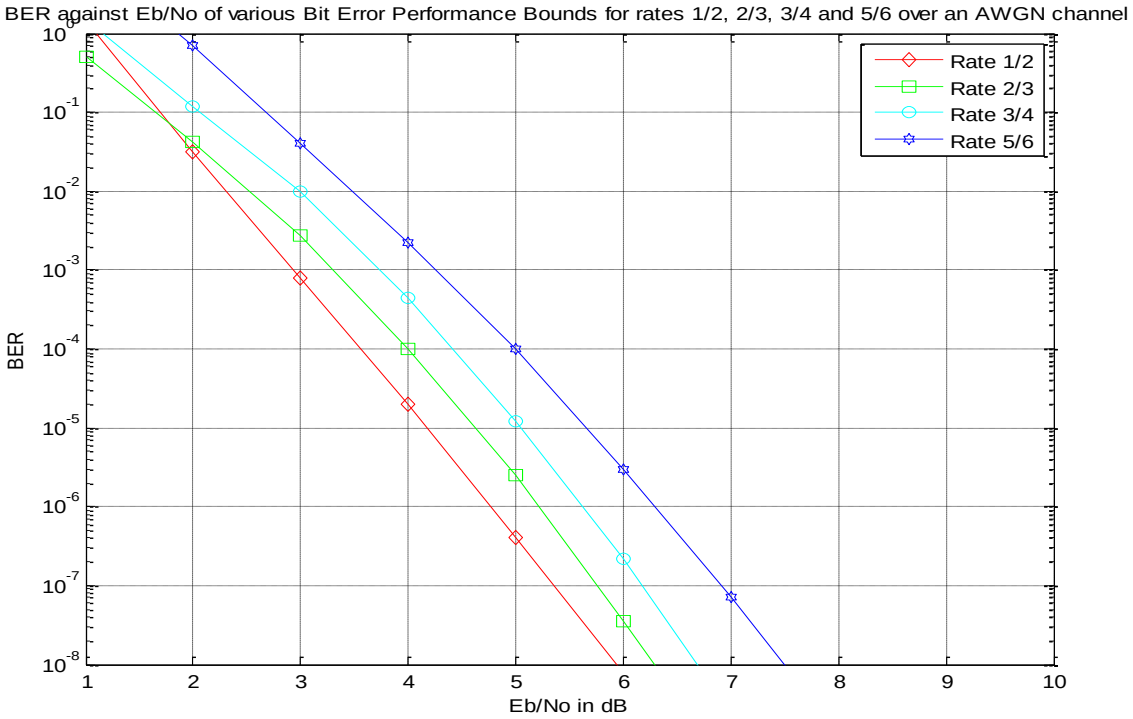

Fig. 9: Bit Error performance bounds for mother rate 1/2 and punctured rates 2/3, 3/4 and 5/6 convolutional encoder

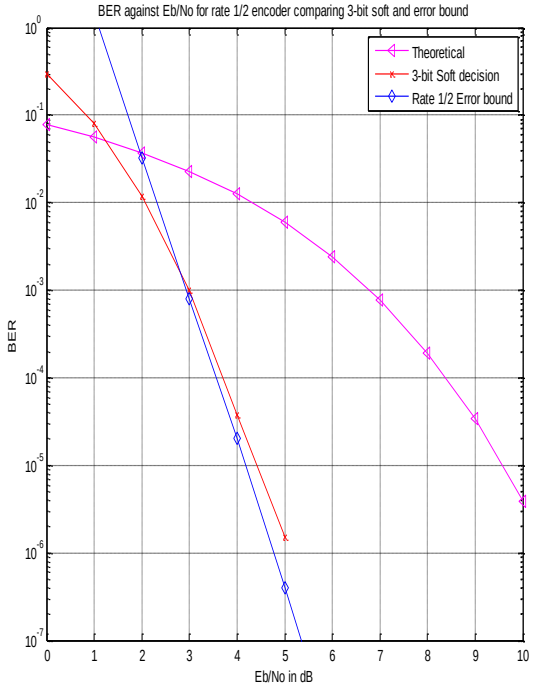

Fig. 10: Performance error bound for rate ½

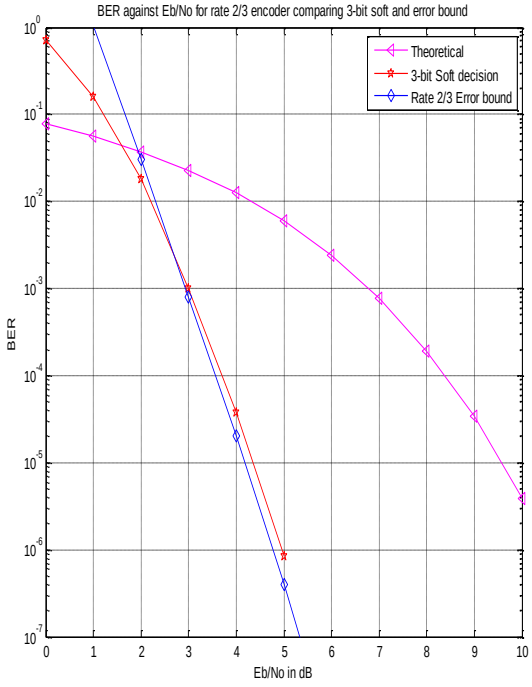

Fig. 11: Performance error bound for rate 2/3

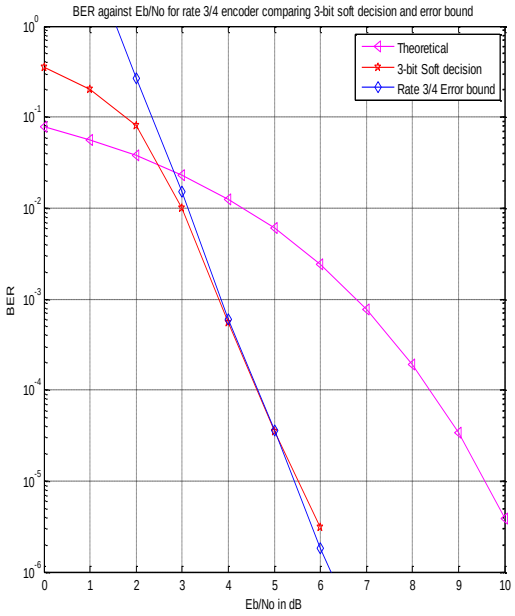

Fig. 12: Performance error bound for rate ¾

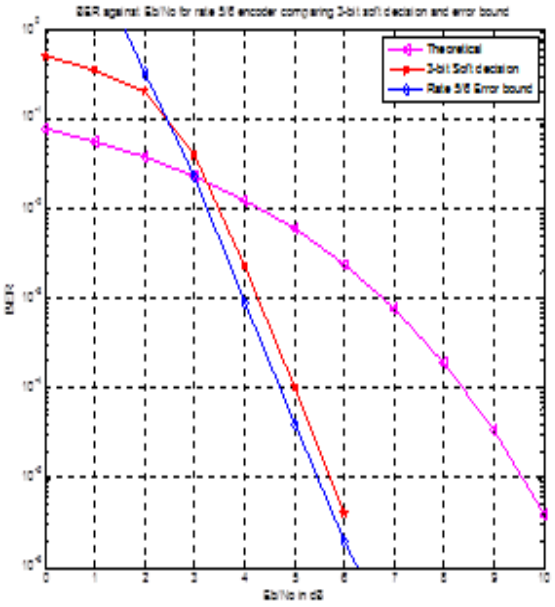

Fig. 13: Performance error bound for rate 5/6

4. Conclusion

This study has carefully covered the 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 rates of 2/3, 3/4 and 5/6 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-compactible punctured codes as against the normal mother rate code have been explained in which the justification of using the punctured codes performed better than their normal code counterparts when examined at the same rate and memory by comparing their degree of computation and duration taken for each decoding to stimulate at a BER value of 10^{-5}. 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 the demands or set-up standard handed in by the IEEE 802.16 – 2009 for the next generation BWA system. Based on this fact, analysis of other CC schemes was made and results showed 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.

References

Abou-El-Azm, A. and Mohammed, F.S. (2019) Variable-rate punctured convolutional coding over fading mobile communication channels., Circuits and Systems. ISCAS '99. Proceedings of the 2019 Institute of Electrical and Electronic Engineers International Symposium on Volume 3.

Anderson, H.R., (2013) Fixed Broadband Wireless System Design. New York, Wiley.

Branka, V. (2019) An Adaptive Coding Scheme for Time-Varying Channels. Institute of Electrical and Electronic Engineers Trans. Communication, vol. COM-39, pp. 653-663, May 2019.

Chen, W. (2016) RTL Implementation of Viterbi Decoder. MSc Dissertation, Linköpings University.

Christopher, L.T. (2001) Punctured convolutional coding scheme for multi-carrier antenna wireless systems. MSc. Dissertation, Electrical Engineering and Computer Sciences, University of California, Berkeley.

Forney Jr., G.D. (2000) The Viterbi algorithm. Procedure of Institute of Electrical and Electronic Engineers Trans. Communication, vol. 61, no. 3, pp. 268-278, March 2000.

Fu-hua, H. (2017) Evaluation of Soft Output Decoding for Turbo Codes. Dissertation (MSc) Virginia Polytechnic Institute and State University.

Hagenauer, J. (2018) Rate-compatible punctured convolutional codes (RCPC codes) and their applications. Institute of Electrical and Electronic Engineers Trans. Communication, vol. 36, no. 4, pp. 389-400, April 2018.

Kai, H. and Gert, C. (2011) An Area-Efficient Analog VLSI Architecture for State-parallel Viterbi Decoding. Retrieved from www.ieeexplore.ieee.org. on 16/02/2022.

KE-Lin, D. and Swamy, M. (2010) Wireless Communications Systems: From RF Subsystems to 4G Enabling Technologies. Cambridge University Press.

Ying, F. and Kar-Ming, C. (2016) Seamless data-rate change using punctured convolutional codes for time-varying signal-to-noise ratio. Institute of Electrical and Electronic Engineers Trans. Communication, international conference on communications, pp 342 – 346, vol. 1.

Zhu, Y. (2011) Modelling of Convolutional Encoders with Viterbi decoders for Next Generation Broadband Wireless Access system. University of Nottingham MSc. Project tutorial note.