© 2014 IJIRT | Volume 1 Issue 6 | ISSN : 2349-6002

Error Detection and Correction: An Introduction

Vineet Godara, Prashant Solanki, Raja Solanki,
Dronacharya College Of Engineering

Abstract– **In most communication system whether wired or wireless convolutional encoders are used and AWGN introduces errors during transmission. Various error correcting and controlling mechanisms are present. In this paper all mechanisms are studied and best mechanism on the basis of accuracy, complexity and power consumption is selected. There should be trade off between complexity of hardware and power consumption in decoder.**

Index Terms- **FEC, Block Codes, Convolutional Codes, ARQ, HARQ, Viterbi Mechanism.**

I. INTRODUCTION

Unlike wired digital networks, wireless digital networks are much more prone to bit errors. Packets of bits that are received are more likely to be damaged and considered unusable in a packetized system. Error detection and correction mechanisms are vital and numerous techniques exist for reducing the effect of bit-errors and trying to ensure that the receiver eventually gets an error free version of the packet. The major techniques used are error detection with Automatic Repeat Request (ARQ), Forward Error Correction (FEC) and hybrid forms of ARQ and FEC (H-ARQ).

Forward Error Correction (FEC) is the method of transmitting error correction information along with the message. At the receiver, this error correction information is used to correct any bit-errors that may have occurred during transmission. The improved performance comes at the cost of introducing a considerable amount of redundancy in the transmitted code. There are various FEC codes in use today for the purpose of error correction. Most codes fall into either of two major categories: block codes [11] and convolutional codes [6]. Block codes work with fixed length blocks of code. Convolutional codes deal with data sequentially (i.e. taken a few bits at a time) with the output depending on both the present input as well as previous inputs. In terms of implementation, block codes become very complex as their length increases and are therefore harder to implement.

Convolutional codes, in comparison to block codes, are less complex and therefore easier to implement. In packetized digital networks convolutionally coded data would still be transmitted as packets or blocks. However these blocks would be much larger in comparison to those used by block codes. The design of error correcting codes and their corresponding decoders is usually done in isolation. The code is often designed first with the goal of minimizing the gap from Shannon capacity [1] and attaining the target error probability. To reflect the concerns of implementation, the code is usually chosen from a family of codes that can be decoded with low ―complexity‖ [2]. On the implementation side, decoders are carefully designed (see e.g. [3]) for the chosen code with the goal of consuming low power while achieving the required decoding throughput2. This ―division of labor‖ has been extremely successful and forms the paradigm behind many modern long-distance communication system designs.

Shannon-theoretic limits, complemented by modern coding-theoretic constructions [3], have provided codes that are provably good for minimizing transmit power. Can we develop a parallel approach in order to minimize the total system power?

With simplistic encoding/decoding models, the issue of fundamental limits on total (transmit +encoding+ decoding) power has been addressed in some recent works [4], [5], [6], [7]. These fundamental limits abstract power consumed in computational nodes [5], [6] and wiring in the encoder/decoder implementation and can provide insights into the choice of the code and its corresponding decoding algorithm. While such theoretical insights can serve to guide the choice of the code family, the simplicity of these theoretical models, which (to an extent) is needed in order to be able to obtain fundamental bounds, also limits their applicability. Even if the models are refined further, the large-deviations techniques used [5], [9] are usually tight only in asymptopia. Thus, at reasonably high error probability (e.g. 10-6) and small distances (e.g. less than five meters), it is unlikely that the bounds themselves can be used to give precise answers on what codes to use. Given the limitations of the fundamental bounds, how do we search for a total-power-efficient code & decoder? After all, for a given block length, there are super-exponentially many possible codes. Further, for each code, there are many possible decoding algorithms. Even when the code and its corresponding decoding algorithm are fixed, there are many possible implementation architectures. Even today, the design and optimized implementation of just a single decoder requires significant effort. It is therefore infeasible to implement and measure the power consumption of every code and decoder in order to determine the best combination.

II. FORWARD ERROR CORRECTION [FEC]

Forward error correction (FEC) is a digital signal processing technique used to enhance data reliability. It does this by introducing redundant data, called error correcting code, prior to data transmission or storage. FEC provides the receiver with the ability to correct errors without a reverse channel to request the retransmission of data.
The first FEC code, called a Hamming code, was introduced in the early 1950s. It is a method adopted to obtain error control in data transmission where the transmitter sends redundant data. Only a portion of the data without apparent errors is recognized by the receiver. This allows broadcasting data to be sent to multiple destinations from a single source.

Forward error coding is also known as channel coding.

BLOCK CODE:-

A block code is a code in which k bits (or, more generally, symbols) are input and n bits (or, more generally symbols) are output. We designate the code as an (n, k) code. We will start with bits, elements from the field GF(2); later we will consider elements from a field GF(q) (after we know what this means). If we input k bits, then there are 2k distinct messages (or, more generally q k). Each message of n symbols associated with a with each input block is called a codeword. We could, in general, simply have a lookup table with k inputs and n outputs. However, as k gets large, this quickly becomes infeasible. (Try k = 255, for example.) We therefore restrict our attention to linear codes.

Convolutional Codes

Convolutional codes are codes that are generated sequentially by passing the information sequence through a linear finite-state shift register. A convolutional code is described using three parameters k, n and K. The integer k represents the number of input bits for each shift of the register. The integer n represents the number of output bits generated at each shift of the register. K is an integer known as constraint length, which represents the number of k bit stages present in the encoding shift register [11].
Each possible combination of shift registers together forms a possible state of the encoder. For a code of constraint length K, there exist 2K-1 possible states. Since convolutional codes are processed

sequentially, the encoding process can start producing encoded bits as soon as a few bits have been processed and then carry on producing bits for as long as required. Similarly, the decoding process can start as soon as a few bits have been received. In other words, this means is that it is not necessary to wait for the entire data to be received before decoding is started. This makes it ideal in situations where the data to be transmitted is very long and possibly even endless, e.g.: phone conversations.

In packetized digital networks, even convolutional codes are sent as packets of data. However, these packet lengths are usually considerably longer than what would be practical for block codes. Additionally, in block codes, all the blocks or packets would be of the same length. In convolutional codes the packets may have varying lengths. There are alternative ways of describing a convolutional code. It can be expressed as a tree diagram, a trellis diagram or a state diagram. For the purpose of this project, trellis and state diagrams are used. These two diagrams are explained below.

A. State Diagram

The state of the encoder (or decoder) refers to a possible combination of register values in the array of shift registers that the encoder (or decoder) is comprised of. A state diagram shows all possible present states of the encoder as well all the possible state transitions that may occur. In order to create the state diagram, a state transition table may first be made, showing the next state for each possible combination of the present state and input to the decoder. The following tables and figures show how a state diagram is drawn for a convolutional encoder.

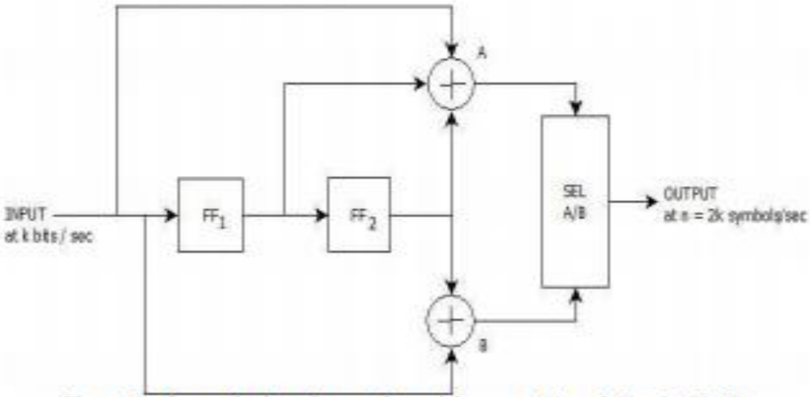
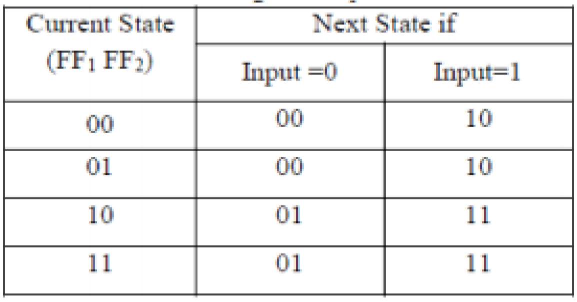
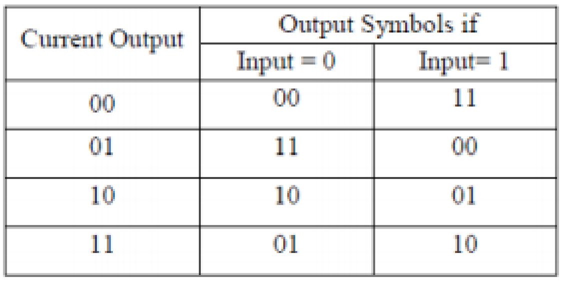


Figure1 : Convolutional encoder with a rate ½ and K =3, (7, 5).



Current State (FF_1 FF_2)	Next State if	
	Input =0	Input=1
00	00	10
01	00	10
10	01	11
11	01	11

Table 1: State Transition Table



Current Output	Output Symbols if	
	Input = 0	Input= 1
00	00	11
01	11	00
10	10	01
11	01	10

Table 2: Output Table

Finally, using the information from Table 1 and Table 2, the state diagram is created as shown in Figure 2. The values inside the circles indicate the state of the flip flops. The values on the arrows indicate the output of the encoder.

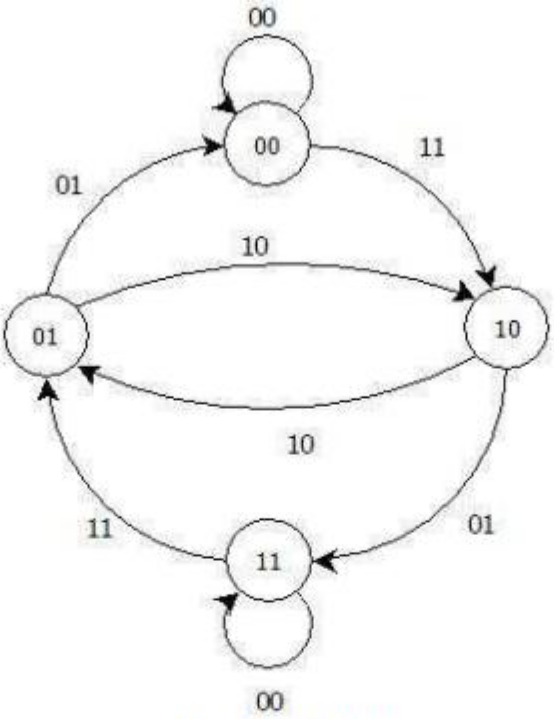


Figure2: State Diagram

B. Trellis Diagram

In a trellis diagram the mappings from current state to next state are done in a slightly different manner as shown in Figure 3.

Additionally, the diagram is extended to represent all the time instances until the whole message is decoded. In the following Figure 3, a trellis diagram is drawn for the above mentioned convolutional encoder. The complete trellis diagram will replicate this figure for each time instance that is to be considered

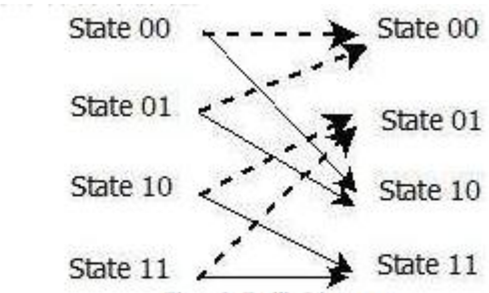


Figure3: Trellis Diagaram

III. CONCLUSION

All error detection, correction controlling mechanisms has been studied. But it has been found that viterbi is most efficient error correction mechanism in long distance communication. Following points justify my view: 1. Now a days convolutional encoders are used in all communication at the transmitter and the transmitter channel is more prone to Additive White Gaussian Noise (AWGN) which introduces error in data. To correct errors either sequential decoding (Fano coding) or most likelihood mechanism (viterbi decoder) is used. But viterbi decoder corrects error exactly.

2. Viterbi decoder assumes that errors occur infrequently, the probability of error is small and errors are distributed randomly.

REFERENCES

[1]. Sklar, B., 2001. Digital Communications – Fundamentals and Applications. 2nd ed. New Jersey: Prentice Hall

[2]. C. E. Shannon, ―A mathematical theory of communication,‖ Bell Sys. Tech. Jour., vol. 27, pp. 379–423, 623–656, Jul./Oct. 1948.

[3]. T. Richardson and R. Urbanke, Modern Coding Theory. Cambridge University Press, 2007.

[4]. Z. Zhang, V. Anantharam, M. Wainwright, and B. Nikolic, ―An efficient 10 GBASE-T ethernet LDPC decoder design with low error floors,‖ IEEE Journal of Solid-State Circuits, vol. 45, no. 4, pp. 843 –855, Apr. 2010.

[5]. P. Grover, K. Woyach, and A. Sahai, ―Towards a communicationtheoretic understanding of system-level power consumption,‖ IEEE J. Select. Areas Commun., vol. 29, no. 8, pp. 1744 – 1755, Sept. 2011.

[6]. P. Grover, A. Goldsmith, and A. Sahai, ―Fundamental limits on complexity and power consumption in coded communication,‖ in extended version of paper presented at ISIT‘12, Feb. 2012.

[7]. P. Grover and A. Sahai, ―Fundamental bounds on the interconnect complexity of decoder implementations,‖ in Proc. of the 45th Annual Conference on Information Sciences and Systems (CISS), March 2011, pp. 1 – 6.

[8]. ——, ―Green codes: Energy-efficient short-range communication,‖ in Proceedings of the 2008 IEEE Symposium on Information Theory, Toronto, Canada, Jul. 2008.

[9]. Shannon, C.E., 1948. A Mathematical Theory of Communication. Bell SystemTechnical Journal, vol. 27, pp.379-423.

[10]. Proakis, J.G., 2003. Digital Communications. 3rd ed. New York: McGraw-Hill, Inc.

[11]. Wikipedia. General Algorithm - Hamming Codes. [Online]. Available at: <http://en.wikipedia.org/wiki/Hamming_code#General_algorithm> [Accessed 15 May 2010]

[12]. Ranpara, S.; Dong Sam Ha, 1999. A low-power Viterbi decoder design for wireless communications applications. IEEE Proceedings of the Twelfth Annual IEEE International Int. ASIC Conference 1999, Washington, DC, 15-18 Sept. 1999, pp. 377-381

[13]. Wikipedia. Viterbi Decoder. [Online]. Available at: http://en.wikipedia.org/wiki/Viterbi_decoder#Euclidean_metric_computation [Accessed 26 August 2010]

[14]. Chip Fleming, 'A tutorial on convolutional coding with Viterbi algorithm',2003.

[15]. K. S. Arunlal1 and Dr. S. A. Hariprasad,‖ AN EFFICIENT VITERBI DECODER‖ International Journal of Advanced Information Technology (IJAIT) Vol. 2, No.1, February 2012