Design of Error Correction and Allusion in Turbo Decoder for Wireless Application

Shivashnkari
PG scholar (VLSI DESIGN),
Arunaicollegengineering,Tiruvannamalai,
TamilNadu.
E-mail:shivanivpm@gmail.com

Victoriya
Assistant Professor,
Department of ECE,
Arunai college of engineering,
E-mail:victoriya.isai@gmail.com

Abstract- Wireless sensor network (WSN) consists of limited energy resources, which reduce the processing capabilities and a radio frequency communication unit with limited transmission power. In wireless communication, Automatic Repeat request (ARQ) technique are used which has power loss in packet retransmission. Then Error correcting code(ECC) can be used to reduce the number of packet retransmission Turbo codes are error correcting codes with at least two dimensions (i.e. each datum is encoded at least twice).The decoding of turbo codes is based on an iterative procedure using the concept of extrinsic information. Highly parallel decoders for convolution turbo codes have been studied by proposing two parallel decoding architectures and a design approach of parallel interleaves. This paper analysis the parallel concatenation turbo encoder and error correction circuit. It detects the error in a given data, if the data should have error then it intimate the error to the user and recover that error.

Keywords:*turbo codes, convolution encoder, wireless sensor networks*

I. INTRODUCTION

Wireless sensor networks are currently being considered for many communications application, including industrial, medical, environment. One of the main applications of Wireless Sensor Networks is monitoring remote and isolated areas, and collecting information about unexpected phenomena like volcano eruptions or enemy movement in the battle field. In these applications the channel state is expected to be continuously varying because of the dynamic changes in environmental factors.Also vehicles and rocksmovements can crash some nodes which can separate parts of the network. In this context it is hard for the network to deliver the collected information, even when transmitting at the maximum power, without strong error correctiontechniques. Using ARQ to tackle erroneous packets in such environment is inefficient, because of the high number of retransmissions needed. Also ARQ techniques introduce high latency, where the repeated retransmission consumes considerable time, which leads to a high delay between detection moment of an event at the sensor nodes and informing the base station about that event. Retransmission also consumes large amount of energy from both transmitting and receiving nodes. Some authors suggest the use of simple Error Correction techniques (like BCH, Reed-Solomon and convolution codes) in Wireless Sensor Networks. The error correction capability of these codes is obtained at the expense of a high redundancy in the transmitted data. Even with simple codes, implementing coding techniques introduces high delay in delivering packets to the base station . The delay comes from the coding and decoding processes that run on each node in the network while routing data to the base station.

Balakrishnan et al. studied the power consumption of different ECC circuitry (BCH, Reed-Solomon and convolutional codes).

Turbo codes constitute major development in the field of Forward Error Correction (FEC). and other applications where designers seek to achieve maximal information transfer over a In electrical engineering and digital communications turbo codes are a class of high performance error correction code developed in 1993 which are finding use in deep satellite communication limited bandwidth communication link in the presence of data corrupting noise. Exhibiting performance approaching the Shannon limit, Turbo Codes (TC) have the TC block set features efficient encoder and decoder designs seen rapid adoption in the design of digital communication systems. Desirable and Designable introduces the basics of turbo codes in their different flavors (more specifically, parallel concatenated convolution turbo codes and block turbo codes). Through the application of systemic design methodology that considers data transfer and storage as top priority candidates for optimization, the authors show how turbo codes can be implemented and the attractive performance results that can be achieved in throughput, latency . the turbo code. This represents a significant development in the field of error- correcting codes. The principle of decoding is to be found in an iterative exchange of information between elementary decoders, called extrinsic information, and it is this principle from which the term turbo originates. The turbo concept is now applied to block codes as well as other parts of a digital transmission system, such as detection, demodulation. Applications that integrate turbo codes into their standards are mobile communications, wireless networks and local radio loops. Future applications could include cable transmission, short-distance communication or data storage includes cable transmission, short-distance communication or data storage. Turbo coding is an advanced error correction technique widely used in the communication industry. Turbo encoder and decoder are key elements in today’s communication system to achieve the best possible data reception with least possible errors. The basis of turbo coding is to introduce redundancy in the data to be transmitted through a channel. The redundant data helps to recover original data from the received data

II. RELATED WORKS

High speed wireless sensor network are currently being considered for a variety of communication application such as environmental, medical, industrial (or) security scenarios, for increased transmission rates given the limited embedded battery lifetime, ultra-low power security is needed in the sensors and processor. Much research in being undertaken in this different area at the device, circuit, system and network levels, although error control coding (ECC) potentially reduces the required transmit power for reliable communication, higher decoder complexity increases the required processing energy. The of ECC results is explored in this paper to find when use of ECC results in the system. Several recently implemented decoder are analyzed, comparing both analog and digital implement. The four most energy circuit decoders are analog decoders, the best analog decoder become energy circuit about ¼ the distance of the best digital implementation. The turbo iterative decoding algorithm only performs well for turbo codes with relatively small memories. Moreover, its decoding complexity becomes prohibitively large when the turbo encoders have very large memories. The sum product and linear programming decoding algorithms are based on the parity-check matrices of the codes. They are widely used to decode low-density parity-check codes. These algorithms do not suffer the limitations of the turbo iterative decoding algorithm. In order to apply them to the turbo codes, the parity check matrices of turbo codes must be found. By treating turbo codes as serially concatenated codes, the generator and parity check matrices of the turbo codes are derived in this paper. Turbo codes with low-density parity-check matrices are then designed based on the derived results. Provided these matrices, turbo codes are decoded using the sum-product algorithms. Preliminary results show that the sum-product decoding of turbo codes performs slightly worse than sum-product decoding of conventional low-density parity-check codes. However, since the encoding of turbo codes has less complexity than the straight forward encoding of low-density parity-check codes,this loss in performance may be justified by the drastically decreased encoding complexity

III. DESIGN

We build our design on the Parallel Concatenated Convolutional Code technique (PCCC) with coding rate 1/3 for its performance and its systematic feature. The turbo decoder implementation is shifted to the base station, where enough processing power and energy are available. Our approach is to use the PCCC circuit on the source node for encoding data packets, while using the PCCC circuits on the routing nodes to the base station as Detect-Correct circuits. At the forwarding nodes, the original data is xtracted from the packet and re encoded using the PCCC circuit. The outputs of the RSC encoders are compared with the parity bits in the received packet. Comparing parity bits enables the forwarding node to detect and correct some of the errors.

The Detect-Correct function, at the forwarding nodes is important, not only it can correct some error bits in the forwarded packets, it also monitors the error pattern occurring in the packets. If bursts of error occurred in the packet, the Detect-Compare circuit detects it and requests the previous routing node or source node to retransmit only the damaged part of the packet. This operation prevents highly corrupted packet from propagating through the network to the base station, where it could be un decodable and lost. Also it reduces the size of the retransmitted packets to be only size of the collided part of the packet when a near node transmit on the same packet transmission duration. Figure shows the updated PCCC encoder circuit to do the function of both encoding and Detect-Correct when needed. .

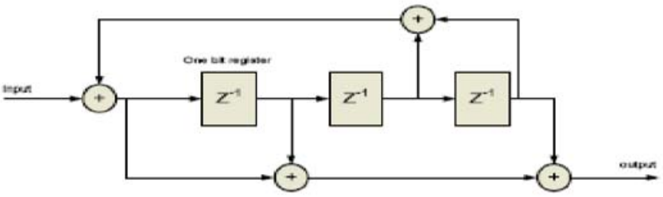

Figure1. Recursive systametic convolution encoder circuit

- each *m*-bitinformation symbol (each *m*-bit string) to be encoded is transformed into an *n*-bit symbol, where *m*/*n* is the code *rate* ($n \geq m$) and each *m*-bitinformation symbol (each *m*-bit string) to be encoded is transformed into an *n*-bit symbol, where *m*/*n* is the code *rate* ($n \geq m$) and the transformation is a function of the last *k* information symbols, where *k* is the constraint length of the code.
- codes are used extensively in numerous applications in order to achieve reliable data transfer, including digital video, radio, mobile communication, and satellite Convolutional communication.

- These codes are often implemented in concatenation with a hard-decision code, particularly Reed Solomon. Prior to turbo codes, such constructions were the most efficient, coming closest to the Shannon limit.

The flowchart of the algorithm running at the compare and correct block. The algorithm runs in the following steps.

Step 1:Fill the interleaver memory with the uncoded data m if the node is a source node.else if the node is a forwarding one receive the packet then extract and store the uncoded data sequence m into the interleaver memory.
Step 2:If the node is the source node, start reading from the memory by rows for the first RSC encoder and interleaver order for the second one and combine output of the encoders(c1,c2)with source data m.
Step 3:If the node is a forwarding one, generate(d1,d2) with the two RSC encoders,then compare with c1,c2
Step 4:If compared bits are identical back to step 3 with new bit from the memiry else
1)If consequent errors detected with length K. which is the length of the tab-delay-line in the RSC encoders. Return k-1 bits in the interleaver memory and flip the data bit.Then back to step 3 with encoding from fipped bit.
2)else if consequent error length smaller then k-1,and
a) C1 and c2 differ from d1 and d2.then flip data bit read from memory and continue.
b) C1 or c2 differ from d1,d2.then flip the different bits between c1,c2 and continue.
3)else if consequent error length equal k-1 and retries limit to correct the exceeded. Ask a retransmission of thepacket from this point to the end of the packet.
Step 5: after running the algorithm on the entire symbol in the memory, transmit the proceed packet to the next in the rougting path.

IV. PROPOSED METHOD

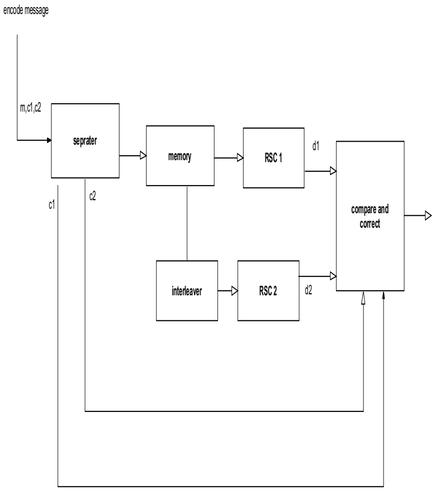

Figure 2.compare and correct circuit

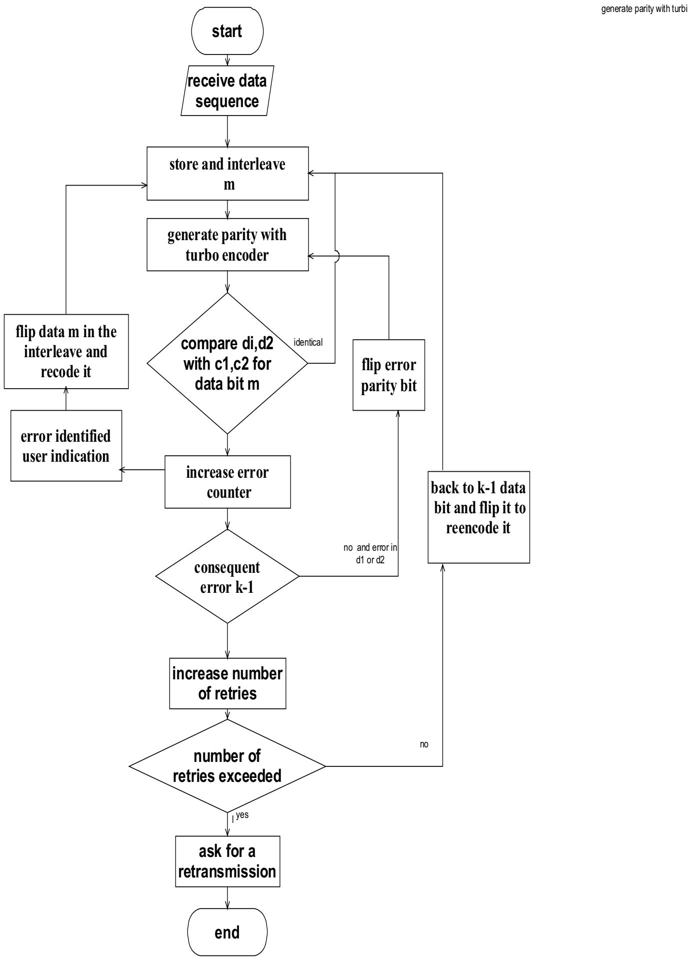
generate parity with turbi
start
receive data sequence
store and interleave m
generate parity with turbo encoder
compare di,d2 with c1,c2 for data bit m
identical
flip error parity bit
flip data m in the interleave and recode it
error identified user indication
increase error counter
no and error in d1 or d2
consequent error k-1
back to k-1 data bit and flip it to reencode it
increase number of retries
no
number of retries exceeded
yes
ask for a retransmission
end

Figure 3.Compare and correct algorithm

In this proposed system error intimation and after to recover error using turbo codes and finally to intimate the recover data to the user. we model the interleaver as a memory block with a variable N, where the encoder can read and correct any bit in it. When the circuit runs the as PCCC encoder at the source node, the data sequence m is serially shifted into the memory. the first Recursive systematic convolutional encoder(RSC) reads serially data bits from the memory row-by-row, with the same order of input sequence m. The second RSC encoder reads the data symbol from the memory in random sequence through the interleaver block. The output of the RSC encoder are combined with the data sequence to form the output coded packet. When the circuit runs as detect-correct circuit, the interleaver memory is filled with received uncoded data sequence m extracted from the received packet. The two RSC encoders run in the same way as in the PCCC encoding mode. The generated sequence (d1,d2) are sent to the compare and correct block. The compare and correct block checks the received packet for errors by comparing(d1,d2) with (c1,c2). The compare and correct block corrects parity bits c1,c2 or data bit m by writing in the interleaver memory if needed.

V. SIMULATION RESULT

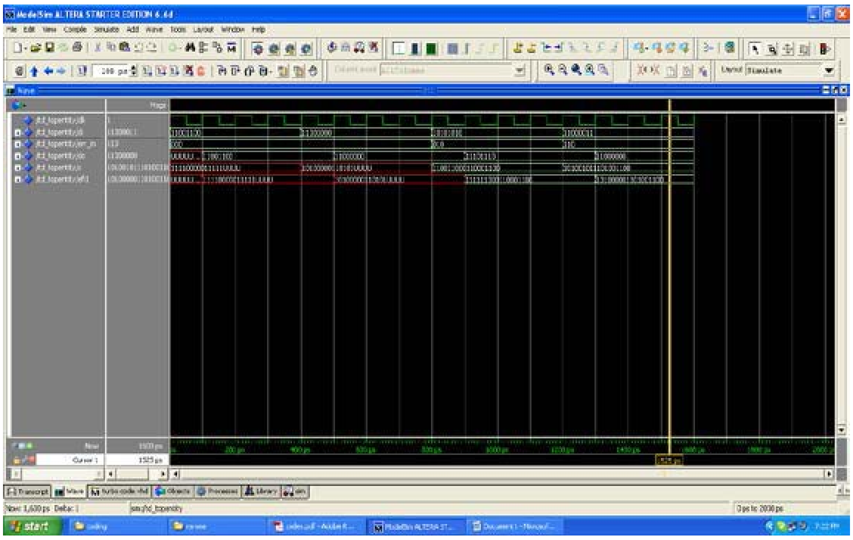

Figure 4.Encoder output

VI. CONCLUSION

In this work, the problem of delivering collected data from sensor nodes to the base station in an energy efficient and delay aware way in a noisy environment. The implementation that uses turbo codes, where the encoding process takes place at the source node. The implementation makes use of low complexity error tracking and error correction at the forwarding nodes. Also consider the trade-off between inter leaver memory size of the Turbo Encoder and error bit probability at the base station. The presented results show the improvement of the performance when using this combined technique

VII. ACKNOWLEDGMENT

The authors would like to thank Ms.D.victoriyaMr.C.A.Sathiyamoorthy, and Mr.T.S.Shivakumaran of the VLSI & ELECTRONICS Division, Arunai college of Engg(AEC), Thiruvannamalai, India, for their contributions to this work.

REFERENCES

1. N. Abughalieh, K. Steenhaut, and A. Nowé, “Low power channel coding for wireless sensor networks,” in Seventeenth Annual Symposium of the IEEE/CVT Benelux Chapter, 2010.
2. G. Balakrishnan, M. Yang, Y. Jiang, and Y. Kim, “Performance Analysis of Error Control Codes for Wireless Sensor Networks,” Fourth International Conference on Information Technology (ITNG’07), pp. 876–879, Apr. 2007
3. S. Benedetto, G. Montorsi, T. Agenzia, and S. Italiana, “Design of parallel concatenated convolutional codes,” IEEE Transactions on Communications, vol. 44, pp. 591–600, 1996.
4. J. Goldsmith and S. B. Wicker, “Design challenges for energy constrained ad hoc wireless networks,” IEEE Wireless Communications Magazine, pp. 8–27, 2002.

5. Fan jiang ,Eric Psota, and Lance c.Perez"Decoding turbo codes based on their parity-check matrices" in IEEE 39thsoutheastem symposium on system theory,2007
6. S. L. Howard, C. Schlegel, and K. Iniewski, "Error Control Coding in Low-Power Wireless Sensor Networks: When Is ECC Energy- Efficient?," EURASIP Journal on Wireless Communications and Networking, vol. 2006, pp. 1–15, 2006.
7. S. Lin and D. J. Costello, Error Control Coding, Second Edition. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 2004.
8. T. K. Moon, Error Correction Coding: Mathematical Methods and Algorithms. Wiley-Interscience, 2005.
9. C. Schlegel and L. Perez, Trellis and Turbo Coding. Wiley-IEEE Press, 2004
10. H. Soude, M. Agueh, and J. Mehat, Towards an optimal reed Solomon codes selection for sensor networks. New York, New York, USA: ACM Press, 2009.