An HARQ based Optimized Error Correction Technique

Kaustuv Kunal
Indian Institute of Information Technology,
Allahabad,
India

R. C. Tripathi
Indian Institute of Information Technology,
Allahabad,
India

Vrijendra Singh
Indian Institute of Information Technology,
Allahabad,
India

ABSTRACT
Errors during data communication are inevitable. Noise in the channel leads to bit error. The paper proposes a matrix based novel bits encoding technique, aim to achieve error correction capability with optimize redundancy. Furthermore an efficient software based decoding algorithm to detect and correct transmission errors is introduced. Here errors include single bit error, multiple bits error and burst errors. The proposed technique maintains high code rate, provides multiple bit error correction capability and can best be implemented as hybrid automatic repeat request (HARQ).

General Terms
Error correcting codes, bits encoding, decoding

Keywords
Error detection, error correction, forward error correction, automatic repeat request, hybrid automatic repeat request

1. INTRODUCTION
Data communication over the data networks comprising of various carriers like repeater, hub, gateways etc is prone to errors because of reasons such as traffic congestion, delay, network components getting down, packet drop, non receipt of acknowledgements and signaling factors. Automatic repeat request (ARQ) and forward error correction (FEC) are strategies to combat error. ARQ proposes retransmission. Retransmission is not a desirable option particularly in long distance and wireless communication such as through satellite. In FEC, redundancy is added for error prevention. Redundant bits are encapsulated with data bits to form encoded information. However this increases the payload for transmission. Addition of redundant bit is known as channel coding. Error correcting codes (ECC) are used for this purpose. Reliability can be enhanced by combining FEC and ARQ as Hybrid-ARQ (HARQ).

First significant ECC as a way of systematic codes was introduced by Hamming [1].He corrected one error per block of code using parity and adjusted it to detect up to two bit error. Elias [2] improved the efficiency with his second order check product codes (figure 1) and further emphasized that it is not necessary to use the same kind of code at each stage of iterative process. BCH code[3,4,5],a multiple bit ECC, is a further generalization of hamming code where *2m-1* bits comprising of *mt* redundant bits enable correction up to *t*-bits error. Other significant ECC include Reed Muller code [6, 7], Low density parity check code (LDPC) [8, 9, 10] and turbo code [11, 12]. FEC has reliability issues and ARQ suffers with degrading throughput. A solution is to combine both. Such a scheme is termed as hybrid ARQ [13, 14].

The proposed technique achieves multiple bit correction capability after encoding with high code rate. It uses outlines of product codes and encode message in the form of matrix. Row and column encoding method are dissimilar with an objective to achieve maximum correction capability with minimum redundancy. Our proposed technique when used as HARQ is able to archive high throughput as well as high reliability.

Proposed technique has been described under next two sub sections information encoding and information decoding. Performance analysis, limitations and assumptions are discussed in section III followed by conclusion. Theoretical and software level implementation are discussed.

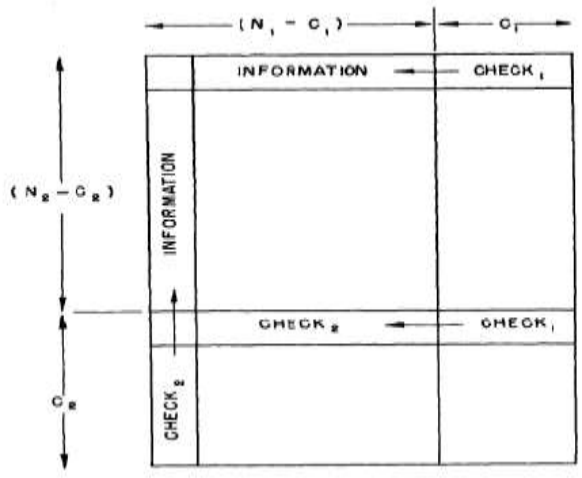

Figure 1 : Organization of first and second order check digits by Elias

2. PROPOSED TECHNIQUE

2.1 Information Encoding
Data bits are encoded into fixed size matrix. This size should be agreed upon and known to both sender as well as receiver. Matrix structure outlines figure1 and is shown in figure 2. A

novel encoding technique is introduced here, check bits-I represent the number of ones in the data bits. This has two major advantages, first the number of check bits required will be bare minimum for example 31 data bits require just 5 check bits and second, less overhead in extraction of original message [15].

Last row of matrix contains only redundant information i.e. check bits-II, They are column parity bits of matrix and can be constructed using XOR. The check bit –II spans only across single row. In this scenario they comprise of least redundant size in comparison to the scheme of Elias[2]shown in figure 1.Thus in our case encoded matrix has clearly less redundancy than any other multiple bit error correction versions of product codes.

Matrix size is chosen based on channel efficiency and the level of correction require. Our scheme provides best possible tradeoff between redundancy and efficiency of retransmission. An example of encoded 8 X 19 bit structure is depicted in figure 2. Here 105 data bits are encoded with 47 check bits. The code rate is 0 .69. Code rate exponentially increases with increase in matrix size. Check bits-II are even parity bits of respective columns.

Data															Check -I			
1	0	0	0	1	0	1	1	0	0	0	1	0	0	1	0	1	1	0
0	1	1	1	1	1	1	0	0	0	1	0	1	1	0	1	0	0	1
1	1	1	0	0	0	0	1	1	0	1	0	1	1	0	1	0	0	0
1	0	0	1	1	1	0	1	0	0	0	0	1	0	1	0	1	1	1
0	0	0	0	1	1	1	0	0	1	1	0	0	0	0	0	1	0	1
1	1	1	1	1	1	1	0	1	1	0	1	1	1	0	1	0	1	0
0	1	0	1	1	0	1	0	1	1	0	0	0	1	0	0	1	1	1
Check-II (Whole last row)																		
0	0	1	0	0	0	1	1	1	1	1	0	0	0	0	1	0	0	0

Figure2: Encoded matrix of 19 bits row and 8 bits column

2.2 Information Decoding

Receiver receives encoded message and decodes it to get correct message. Software based hard decision decoding [16] steps are discussed. Decoding process is designed to support HARQ.

Step 1: Record all erroneous columns of received matrix. A column is erroneous if bits of that column are not in parity.

Step 2: Record erroneous rows with their difference values. A row is erroneous if number of data bits having value as one is found to be different than decimal conversion of binary checks bits. Difference is subtraction value of decimal conversion of check bits to number of ones in data bits of respective row. Clearly difference is positive if the number represented by check bits is more than number of ones in data bits and negative in the opposite case. The last row which contains check bits-2 is excluded here and is certainly not a candidate for erroneous row calculation.

Step 3: If no column or row is found as erroneous, message will be accepted as correct otherwise correction algorithm, as explained in step 4, will be applied.

Step 4: The flow chart of correction algorithm is depicted in figure 3. Recorded erroneous row and erroneous columns are input to correction algorithm. For each erroneous row, matrix entries corresponding to every erroneous column position is checked. A match is said to be found if matrix content is zero for positive difference row, and one for negative difference row. Total number of matches found, in the erroneous row is counted; if this is equal to modulus of row difference then matched matrix entries are inverted. Respective columns are no longer erroneous and are removed from list of erroneous column. The respective row is corrected so it is also removed from erroneous row list. In case of match count not equal to row difference, the same process is repeated with next erroneous row. Initially algorithm traverses through all erroneous rows sequentially. In case of correction in any matrix entry this traversal continues in cyclic manner and revisit previously visited rows which are still erroneous. Reason is that due to deletion of an erroneous column it may possible that revisited row now has equal value of match entries and row difference. If in a whole cycle there is no single matrix entry changed the process exists from loop.

Step 5: Finally if no erroneous row or column remains, the received matrix is said to be corrected and receiver accepts the packet, otherwise retransmission is requested.

A C code has been written for decoding process simulation. Such a code can be embedded into a decoder device. Few variation of decoder is possible. Figure 2 can be transmitted column wise, received column is checked for parity. If the data bit columns are in parity, receiver accepts data as correct and sender refrain itself from sending check bits columns (i.e. column 16, 17, 18, 19). Only when data columns are not in parity receiver asks for check bits column. Drawback of this scheme is that it will correct only in cases when errors per columns are odd.

3. ANALYSIS

3.1 Redundancy

Number of check bits-I should be enough to represent data bits of the row. Table 1 shows check bits for a given range of data bits per row. The relation between check bits-I (c) and data bits (k) of any row is mathematically expressed as:

$$C = [\log_2 k] + 1$$

Where [ ] is floor function, [x] will give the largest integer <= x.

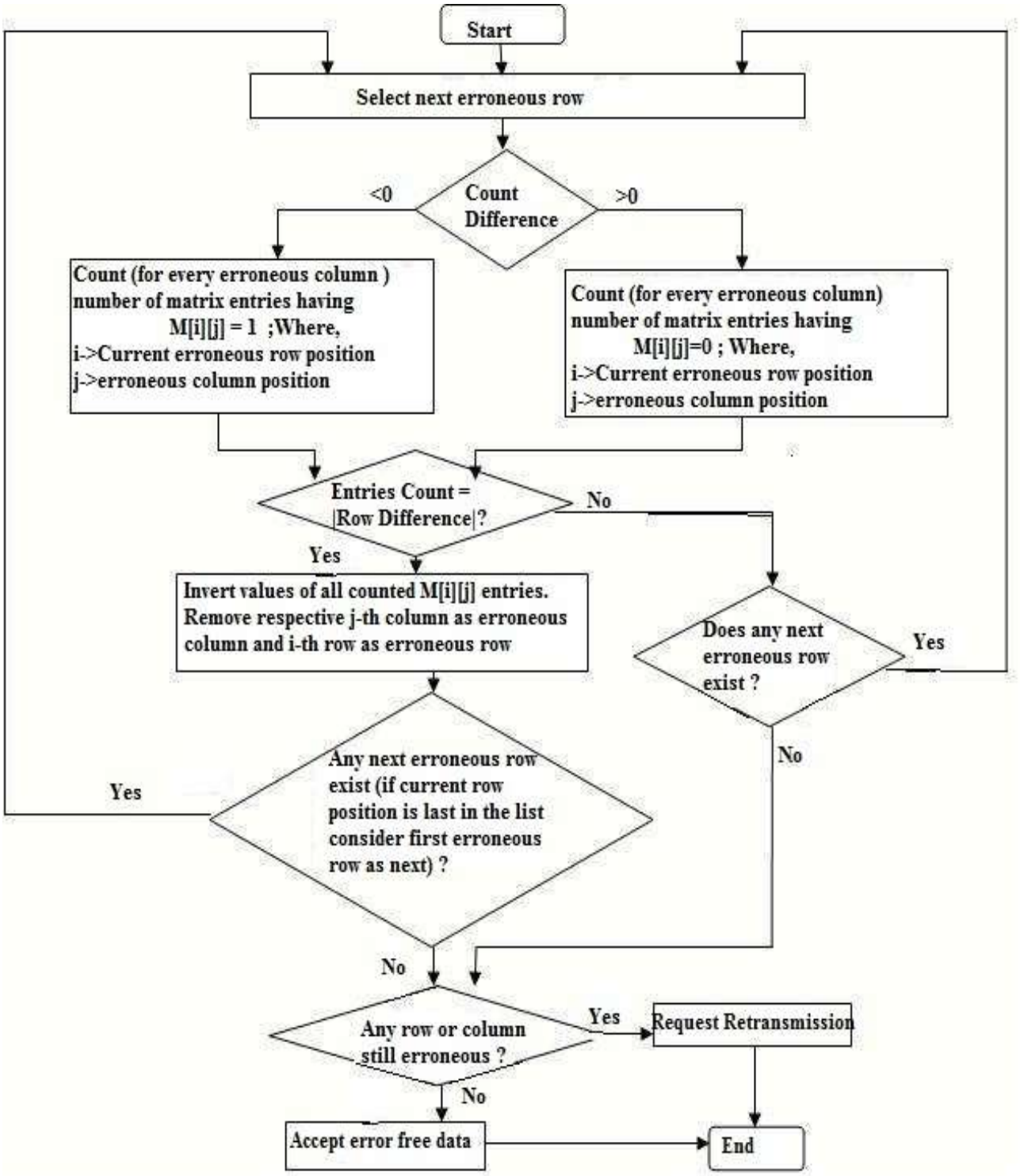

Figure 3: Flowchart of correction algorithm

Table 1: Number of Check bits-1 for different data bits value range

Data bits(k)	Check bits(c)
1	1
2-3	2
4-7	3
8-15	4
16-31	5
32-63	6
64-127	7
128-255	8
256-511	9
512-1023	10
1024-2047	11
2^{n-1}–(2^n-1)	n

Appending redundant bits after data bits eliminates the overhead of interspersing the redundancy bits at the sender end and their removal at the receiver end after checking for error and consequent correction.

Through its implementation, it emerges that in our case, encoded matrix has high code rate. Code rate versus column length graph for matrix of fixed row size of length eight bits is plotted in fig 4. The code rate increases exponentially initially and thereafter attains almost a constant value. In this case the constant value is 0.85. Higher value can be achieved by using row of length 16 or 32. Drawback obviously is compromise against correction efficiency.

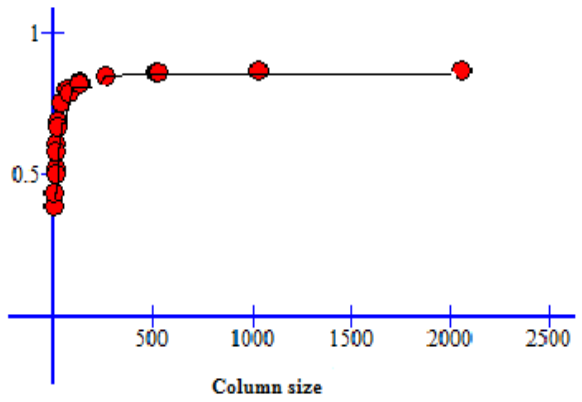

Figure 4: Code rate vs. column size graph for row size of 8 bits

3.2 Error detection and correction

Through various test cases examined, proposed technique has been found capable of detecting almost any single or multiple bit errors including burst errors with higher efficiency than CRC or checksum [17]. Figure 5 depicts a comparison between error detecting techniques and proposed technique. Proposed technique efficiently detects errors which are undetected by CRC and Checksum. It corrects single data bit error with 100 percent efficiency. Typical multiple bit and burst error patterns can also be corrected. Technique is suitable for randomly located errors in data streams. As there is far more probability for data bits to be erroneous than check bits. If any column of check bits is erroneous the correction algorithm is not applied and retransmission request is sent directly.

Proposed correction algorithm adheres to few limitations. The algorithm is designed to correct single error per column. The algorithm will be applicable if corrupted bits of a row are of same value. However in all above cases it will detect the error and send retransmission request. Clearly scheme is best implemented when used as HARQ.

Checksum	CRC	Proposed Technique
Error detection only Undetected Error example 1: Sent message : 1 0 0 1 1 1 0 1 1 0 0 0 ->checksum Received erroneous message 0* 0 0 0* 0* 1 0 1 1 0 1* 0 As complement of addition is all zeroes, received data is assumed correct and accepted.	Error detection, single bit error correction (requires large memory overhead) Undetected error example 2: Data: 1001010 Divisor: CRC 4 – 10011 Sending info : 10010101010 Received erroneous message 1 0 1*1 0 1 0 1 1* 1 1* Received message is completely divisible by CRC-4, though it contains 3 erroneous bits.	Error detection ,single and multiple bit error correction Can detect all generic errors. Will Successfully detect error for given undetected error example 1 &2. Detects the errors in case of corrupted redundant bits and request retransmission directly.

Figure 5: Comparison with error detection methods

4. CONCLUSION

A new optimized bit encoding technique is implemented. Here check bits represent number of ones in data bit. This significantly minimizes redundancy besides allowing extraction of original message with low over head. The matrix size affects the multiple bit correction efficiency and is chosen based on channel efficiency. A receiver side error detection technique along with a software based algorithm for error correction is described. We verified that algorithm can correct any single data bit error. We also verified that its error detection capability is better than most of the existing ones. Typical

multiple bit and burst error patterns can also be corrected and the technique is best implemented as HARQ.

5. REFERENCES

[1] R.W. Hamming," Error correction and error detection coding, "Bell System technical journal, Vol XXIX, No.2, April 1950.

[2] Peter Elias," Error Free coding," IRE Professional Group on Information Theory, 1954

[3] Peterson W,"Encoding and Error-Correction Procedures for the Bose-Chaudhuri Codes," IRE Transactions on information theory, Volume: 6, Issue: 4, 1960.

[4] J Cho, W Sung,"Efficient Software-Based Encoding and Decoding of BCH Codes" IEEE Transactions on Computers, 2009.

[5] EH Lu, T Chang," New decoder for double-error-correcting binary BCH codes," IEEE Proceedings-Communications, Vol. 143, No. 3, June 1996.

[6] Ben Cooke, "Reed-Muller Error Correcting Codes," MIT Undergraduate Journal of Mathematics, pp. 21-26, 1999.

[7] P Gopalan, AR Klivans, D Zuckerman, "List-decoding reed-muller codes over small fields" Proceedings of the 40th annual ACM symposium on Theory of computing, 2008.

[8] R Gallager, "Low density parity check codes," IRE Transactions on Information Theory, 1962.

[9] Casado, A.I.V, Wen-yen Weng, Valle, S, Wesel R, "Multiple-rate low-density parity-check codes with constant blocklength", IEEE Transactions on Communications, Volume: 57, 2009.

[10] Kaufman, A Wigderson, "Symmetric LDPC codes and local testing" Proceedings of ICS, 2010.

[11] S. Adrian Barbulescu, Steven S. Pietrobon, "Turbo Codes: a tutorial on a new class of powerful error correcting coding schemes Part- I," Journal of Electrical and Electronics Engineering, Australia, 19(3):129–152, Sept. 1999.

[12] Louay Bazzi , Mohammad Mahdian ,Daniel A. Spielman," The Minimum Distance of Turbo-Like Codes" IEEE Transactions on Communications, volume 57 , Issue 5, May 2009.

[13] Shu Lin, Philip S. Yu," A Hybrid ARQ Scheme with Parity Retransmission for Error Control of Satellite Channels," IEEE Transactions on Communications, Vol. Com-30, NO. I, July 1982.

[14] Jianming Wu, Jinyin Xue, and Tomohiko Taniguchi, "A Novel HARQ Using Grouping and Encoding Technique" IEEE 70th Vehicular Technology Conference, Fall 2009.

[15] U. K. Kumar, and B. S. Umashankar,"Improved Hamming Code for Error Detection and Correction," 2nd International Symposium on Wireless Pervasive Computing, 2007.

[16] A J Viterbi, "Convolutional Codes and Their Performance in Communication Systems," IEEE Transactions on Communications, 1971.

[17] Norman Matloff, "Cyclic Redundancy Checking,"2001.