ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 1.852

IJESRT

INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

Decoding Techniques of Error Control Codes called LDPC

Mini Manpreet, Prabhdeep Singh
ECE Deptt. ,RPIIT Bastara, Haryana, India
Manpreetsaggu87@gmail.com

Abstract

This paper deals with the design and decoding of an extremely powerful and flexible family of error-control codes called low-density parity-check (LDPC) codes. LDPC codes can be designed to perform close to the capacity of many different types of channels with a practical decoding complexity. It is conjectured that they can achieve the capacity of many channels and, indeed, they have been shown to achieve the capacity of the binary erasure (BEC) channel, under iterative decoding. With help of this paper LDPC codes and their decoding techniques are explained with overview of LDPC.

Keywords: Chanel Coding, LDPC codes and its overview, decoding techniques.

Introduction

The source encoder/decoder and the channel coder/decoder are added to a basic communication model to avoid noise during transmission. Source encoder removes the redundancy of the source information, while the source decoder retrieves the full source information from the encoded data. On the other hand, channel encoder introduces redundancy for reliable transmission of the data through the noisy channel (or storage medium), and the channel decoder retrieves of course depending on the capabilities of the channel encoding/decoding blocks and the noise- the source coded data from the received data. While transmission some errors are still present in transmitted codes.Error correcting codes are the most important tools for reliable communication in any noisy communication channel. Noisy Channel Coding Theorem proves that, probability of decoding error can be made to approach zero exponentially with the code length, if properly coded information is transmitted at a rate below the channel capacity. The use of parity-check codes makes coding relatively simple to implement, however decoding of parity-check codes is not simple. With ordinary parity-check matrices, decoding complexity of parity-check codes grows quadratically with the block length. Therefore decoding algorithms with reasonable decoding complexity are needed for the parity-check codes of large block lengths.

Channel coding

Channel coding is a way of introducing controlled redundancy into a transmitted binary data stream in order to increase the reliability of transmission and lower power transmission requirements. Channel coding is carried out by introducing redundant parity bits into the transmitted information stream. The requirement of a channel coding scheme only exists because of the noise introduced in the channel. Simple channel coding schemes allow the received transmitted data signal to detect errors, while more advanced channel coding schemes provide the ability to recover a finite information about of corrupted data. This result in more reliable communication, and in many cases, eliminates the need for retransmission. Although channel coding provides many benefits, there is an increase in the number of bits being transmitted. This is important when selecting the best channel coding scheme to achieve the required bit error rate for a system. There are two main types of channel coding techniques. The first type is called Automatic Repeat Request (ARQ), in which the receiver requests retransmission of unreliable data frames. A data frame can be declared unreliable if an erasure or an error is detected in that frame. The second type is Forward Error Correction (FEC) where the channel decoder estimates a codeword from the received codeword. An error correction/detection scheme can be evaluated by three important properties; the reliability of the scheme, the complexity of the scheme, and the efficiency of the scheme. The reliability of the scheme stands for the reliability of

ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 1.852

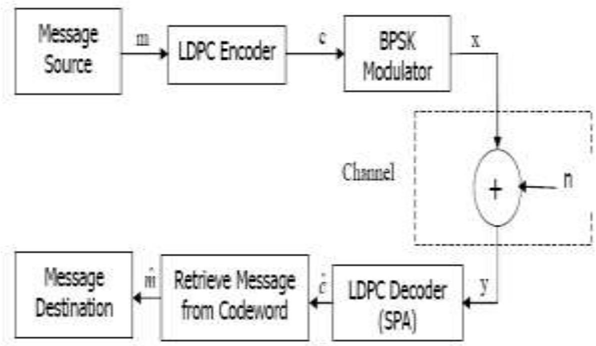
the decoded words in the receiver, which can be measured by Bit Error Rate (BER) or Frame Error Rate (FER).

LDPC codes

LDPC codes are low density parity check codes or linear block codes, that are defined by their sparse parity check matrices. By density, we mean the ratio of the number of ones in the matrix to the number of all elements in the matrix. If for each row (or column) ratio of the number of ones to the length of that row (or column) is equal, then the code is called a regular LDPC code. The low-density condition can be satisfied especially for larger block lengths. The code words of a parity-check code are formed by combining a block of binary information digits with a block of check digits. Low-density parity-check codes are the codes specified by a matrix containing mostly 0's and only a small number of 1's. In particular (n, w_c , w_r) low density code is a code of block length n with a parity-check matrix, where each column contains a small number, w_c of 1's and each row contains a small number w_r of 1's. However, simple decoding schemes exist for low-density codes and this compensates for their lack of optimality. The analysis of a low-density code of long block length is difficult because of the immense number of code words involved. It is simpler to analyze a whole ensemble of such codes because the statistics of an ensemble lets us average over quantities that are not tractable in individual codes.

An LDPC code is a linear error-correcting code that has a parity check matrix H with small number of nonzero elements in each row and column. The code is the set of vectors x such that Hx' = 0. The main advantage of LDPC codes is that they provide a performance which is very close to the capacity for a lot of different channels and linear time complex algorithms for decoding. They are also suited for implementations that make heavy use of parallelism. Low-density parity-check (LDPC) codes are used by high-speed communication systems[3,8].

LDPC overview



LDPC system block diagram

- m Message
- c Codeword
- x Modulated signal
- n AWGN noise
- y Received signal
- $\hat{c}$ Estimated codeword
- $\hat{m}$ Estimated message

Message Source:

The message source is the end-user transmitting the data. In terms of mobile communications, the message source would be the end-user transmitting his/her voice information.

LDPC Encoder:

The LDPC encoder is implemented at the end-user transmitting the data. In terms of simulation implementation, encoding is done via a generator matrix.

BPSK Modulator:

The BPSK (Binary Phase Shift Keying) modulator maps the input binary signals, to an analog signal for transmission. In simulation, the BPSK signal is represented by the mapping: $\{0, 1\} \rightarrow \{E_b, -E_b\}$.

***Channel*:**

The channel is the medium of which information is transmitted from the transmitter to the receiver. In mobile communication, this is a wireless channel, and for other applications this could be copper or fibre optics. The addition of noise normally occurs in the channel. In the simulations, the channel is modeled as an AWGN (Additive White Gaussian Noise) channel. The resulting noise added to the system follows the zero-mean normal distribution, with variance $N_O/2$, and N_O is the single-sided noise power spectral density.

LDPC Decoder:

The decoder is implemented at the end-user receiving the information. In terms of simulation implementation, decoding is a process that loops through passing messages back and forth along the tanner graph under certain conditions are satisfied, or a maximum number of passes have occurred. It is obvious that in mobile communications, the handset would require both the encoder and decoder as a pair to allow for bi-directional communication.

***Retrieve Message From Codeword*:**

This simple process retrieves the estimated message from the estimated codeword. In the simulation this is done via a simple function call following estimating the codeword.

Message Destination:

The message destination is the end-user receiving the data. In a mobile communications environment, this would be the user receiving the voice information of the other user. In

the simulations, there is no message destination; rather the estimated message is compared to the transmitted message in order to detect whether a transmission error occurred.

Decoding techniques

Hard Decision Decoding

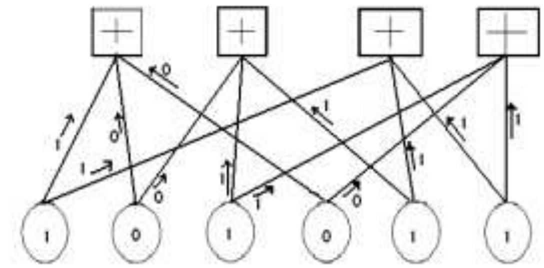
Hard decision decoding involves making a decision on the value of a bit at the point of reception, such as a MAP (Maximum A Posteriori Probability) decoder. Such a decoder forms a decision based on a boundary that minimizes the probability of bit error shows the likelihood functions for BPSK modulation over an AWGN (Additive White Gaussian Noise) channel. The hard-decision decoding algorithm is simple, fast and their hardware implementations are easy. Major drawback of this algorithm is that it operates on hard decisions made by the decoder at the channel output. This throws away valuable information coming from the channel especially when we are dealing with continuous-output channels[5].

Bit Flip Decoding- Bit-flipping (BF) decoding algorithm is a hard-decision decoding algorithm which is much simpler than the soft decision decoding techniques like the Sum-Product Algorithm (SPA) or its modifications but does not perform as well. In the hard decision decoding we analyzed the decoding procedure of Bit-flip decoding.

The bit-flip algorithm is as follows:

Step 1 *Initialization:* Each variable node is assigned the bit value received from the channel, and sends messages to the factor nodes to which it is connected indicating this value.

Step1: *Initial values of the variable nodes are sent from variable nodes to factor nodes.*

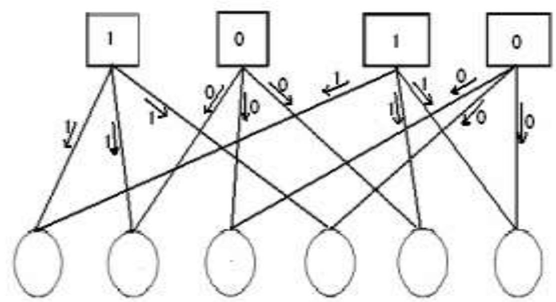


Step 1 of Bit Flip decoding

Step 2 *Parity Update:* Using the messages from the variable nodes, each factor node calculates whether or not its parity-check equation is satisfied. If all parity-check are satisfied the algorithm terminates, otherwise each factor node sends messages to variable nodes to which it is connected indicating whether or not the parity-check equation is satisfied.

Step 3 *Bit Update:* If the majority of the messages received by each variable node are “not satisfied”, the variable node flips its current value, otherwise the value is retained. If the maximum number of allowed iterations is reached, the algorithm terminates and a failure to converge is reported; otherwise each variable node sends new messages to the factor nodes to which it is connected, indicating its value and the algorithm returns to Step 2.

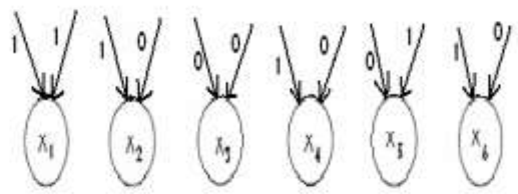
Step 2: *Each factor node calculates whether or not its parity-check equation is satisfied,using the messages from variable nodes.*



Step 2 of Bit Flip decoding

Step 3: *If all parity-check equations are satisfied, that is, all factor nodes send message 0 to variable nodes, algorithm stops*

ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 1.852



Step 3 of Bit Flip decoding

Since this algorithm works on sparse parity-check matrices, each check equation will contain either one transmission error or no transmission errors. If the number of parity-check equations is small then this procedure is reasonable. When most of the parity-check equations containing a bit are unsatisfied, this strongly indicates that bit is in error[6,9].

Soft decision decoding

A decision for a soft-decision decoder is not so clear. Soft-decoding requires processing of the received codeword vector prior to making a decision on the value of the bits. Soft decision decoding algorithms have better BER performance. The soft decision message passing algorithms exchange the extrinsic messages along the edges of the factor graph. At the nodes, local decoding operations update the extrinsic messages according to the message update rules. The messages can be either in terms of probabilities or L-values. Sum-product is a general name for a class of maximum likelihood decoding algorithms. The sum-product algorithm (SPA) is symbol-wise decoding. While decoding LDPC, we are interested in computing the *a posteriori probability* (APP) that a given bit in the transmitted codeword c= $[c_0, c_1, \ldots c_{n-1}]$ given the received word $y[y_0, y_1, \ldots y_{n-1}]$.[5,9] Hence, while encoding of bit c_i ,the interest lies in computing the APP

$$\Pr(c_i = 1|y) \tag{1.1}$$

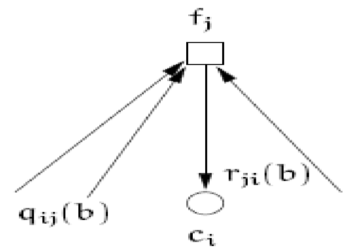
or the APP ratio (also known as likelihood ratio, LR)

$$l(c_i) \triangleq \frac{\Pr(ci = 0y)}{\Pr(ci = 1|y)} \tag{1.2}$$

This result can also be extended to the more numerically stable computation of the log-likelihood ratio (LLR):

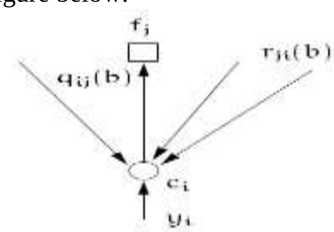
$$L(c_i) \triangleq log\frac{\Pr(ci = 0y)}{\Pr(ci = 1|y)} \tag{1.3}$$

The decoding algorithms based upon the computation of $P_r(c_i = 1|y)$, $l(c_i)$, $L(c_i)$ involves iterations based upon the code's Tanner graph. In one half iteration, each variable node processes its input messages and passes its resulting output messages up to neighbouring check nodes. We observe that the information passed to check node f_j is all the information available to v- node c_i from the channel and through its neighbours, excluding c-node f_j; that is only extrinsic information is passed .such extrinsic information is computed for each connected v-node/c-node pair c_i/f_j at each half-iteration.



Calculation of r_{ji} ***(b)***

In the other half iteration, each c-node processes its input messages and passes its resulting output messages down to its neighbouring v-nodes. This is shown in figure below:



calculation of q_{ij} ***(b)***

The MPA assumes that the messages passed are statistically independent throughout the decoding process. For a girth of λ, the independence assumption is true only up to λ/2-th iteration, after which messages start to loop back to themselves in the graph's various cycles. Simulations have shown that the message passing algorithm is very effective provided length-four cycles are avoided. In this paper we compare the bit error rate performance of the hard decision algorithm and versions of sum-product (soft-decision) message-passing algorithms such as probability domain SPA, log-domain SPA and simplified log-domain SPA[4,7,5,9].

Conclusion

Low-Density Parity-Check Codes (LDPC) were discovered in the early 1960's by Robert Gallager. These codes had been largely forgotten after their invention until their rediscovery in the mid-nineties. This was due in part to the high complexity of decoding the messages and the lack of computing power when these codes were originally invented. LDPC performance improves as block

ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 1.852

length increases, so they can theoretically achieve Shannon's limit as block length goes to infinity. LDPC allows for the reliable transmission, or storage, of data in noisy environments. Even short codes provide a substantial coding gain over uncoded, or low complexity coded systems. These results allows for lower transmission power, transmission over noisier channels, with the same, if not better reliability. . In terms of complexity, the bit flip technique, being the simplest of all is considered the best.

References

1. *Voicila, A., Declercq, D., Verdier, F., "Fossorier, M., Urard,P.,"Low Complexity Decoding for Non-Binary LDPC Codes in Higher Order Fields", IEEE Transactions on Communication, Vol.58,May2010*
2. *Meng-Ying Tsai,"Yousefi, S. "Dynamic-list joint detection and decoding of LDPC-coded V-BLAST systems", Canadian Conference on Electrical and Computer Engineering, 2008*
3. *Jorge Castiñeira Moreira and Patrick Guy Farrell ; Essentials of Error-Control Coding; John Wiley & Sons Ltd;2006*
4. *Enrico Paolini, Michela Varrella and Marco Chiani, "Low-Complexity LDPC Codes with Near-Optimum Performance over the BEC", 2005*
5. *Ben Lu, Guosen Yue and Xiaodong Wang, "Performance Analysis and Design Optimization of LDPC-Coded MIMO OFDM Systems" IEEE Transactions on Signal Processing, Vol. 52, No. 2, February 2004*
6. *Rathnakumar Radhakrishnan, Sundararajan Sankaranarayanan, and Bane Vasi," Analytical Performance of One-Step Majority Logic Decoding of Regular LDPC Codes",IEEE Transactions on Information Theory, Vol. 51, No. 2, February 2003*
7. *Gianluigi Liva and Balazs Matuz,"Pivoting Algorithms for Maximum Likelihood Decoding of LDPC Codes over Erasure Channels", IEEE Transactions on Information Theory, Vol. 49, no. 3, February 2003*
8. *Sae-Young Chung , G. David Forney, Jr., Thomas J. Richardson, and Rüdiger Urbanke, "On the Design of Low-Density Parity-Check Codes within 0.0045 dB of the Shannon Limit", IEEE Communications Letters, Vol. 5, no. 2, February 2001*
9. *Thomas J. Richardson and Rüdiger L. Urbanke,"The Capacity of Low-Density Parity-Check Codes Under Message-Passing Decoding", IEEE Transactions on Information Theory, Vol. 47, February 2001*
10. *D. J. C. MacKay, "Good error-correcting codes based on very sparse matrices," IEEE Trans. Inform. Theory, vol. 45, pp. 399–432, Mar. 1999.*