

Research of Multi-Rate LDPC Decoding in Optical Communication System

Wenhao Jin, Chunwu Liu

College of Artificial Intelligence, National University of Defense Technology, Changsha, China

Email: @nudt.edu.cn

How to cite this paper: Jin, W.H. and Liu, C.W. (2020) Research of Multi-Rate LDPC Decoding in Optical Communication System. *Optics and Photonics Journal*, 10, 174-181. <https://doi.org/10.4236/opj.2020.106019>

Received: June 12, 2020

Accepted: June 26, 2020

Published: June 29, 2020

Abstract

Low-density parity-check code (LDPC) not only has good performance approaching the Shannon limit, but also has low decoding complexity and flexible structure. It is a research hot-spot in the field of channel coding in recent years and has a wide range of application prospects in optical communication systems. In this paper, the decoding aspects and performance of LDPC codes are analyzed and compared according to the bit error rate (BER) of LDPC codes. The computer simulation was carried out under additive white Gaussian noise (AWGN) channel and binary phase shift keying (BPSK) modulation. Through theoretical analysis and simulation results, this paper explores the way of multi-rate LDPC decoding.

Keywords

LDPC, Multi-Rate, Decoding, Simulation

1. Introduction

Since the end of the 20th century, with the development and growth of the Internet, people's demand for data services is increasing, such as online video, audio, game entertainment and other multimedia services. How to improve the capacity and speed has become an urgent task. There are two effective methods to improve the data bandwidth: one is through high-order phase modulation; the other is through channel multiplexing technology. However, as the modulation order increases and the channel multiplexing density increases, the transmission distance of high-speed optical fiber communication systems will also be severely limited, which is manifested in the increase of the BER in the information. One solution is to use better forward error correction (FEC) coding techniques such as LDPC code [1].

LDPC code is a kind of linear block code invented by R. Gallager in 1962. Its decoding is based on sparse matrix and has low computational complexity. It is relatively easy to implement in hardware, and can perform parallel iterative operation to save decoding time. Therefore, LDPC code is more suitable for large capacity communication system. Besides, the rate of LDPC code is adjustable and flexible in practical application. Therefore, multi-rate LDPC decoding in optical communication system has important theoretical and application value [2].

2. LDPC Code and Decoding Implementation

2.1. Principle and Structure of LDPC Code

LDPC code is a kind of (n, k) linear block code defined by sparse parity check matrix H . The parity check matrix H is an $m \times n$ matrix, most of its elements are "0", only a few "1", $m = n - k$. A binary vector $c = (c_1, c_2, \dots, c_n)$ is a codeword when $H * c^T = 0$. Usually (n, w_c, w_r) is used to represent a binary regular LDPC code. Among them, n represents the code length; w_c represents the column weight which means the number of "1" contained in each column; w_r represents the row weight which means the number of "1" contained in each row. For the check matrix, each row of it corresponds to a check equation, each column corresponds to one bit of the codeword, n represents the code length, m represents the check bit length, and $n - m$ represents the information bit length [3].

LDPC codes can also be represented by Tanner graph. This method was first proposed by Tanner. In Tanner graph, there are two kinds of nodes: variable nodes and check nodes, which correspond to the columns and rows of the check matrix respectively. Any variable node in the Tanner graph has the number of branches consistent with the column weight of check matrix; similarly, any check node has the number of branches consistent with the row weight of the check matrix. **Figure 1** shows an example of a Tanner graph of (8, 2, 4) regular LDPC codes on GF (2), where v_i represents a variable node, c_i represents a check node, and the line connecting v_i and c_i is called an edge [3].

2.2. LDPC Decoding Algorithm

The LDPC decoding algorithm has developed into many variants so far. Based on the decision method, it can be divided into two categories. The first category is the hard decision decoding algorithm: Bitflipping (BF); the second category is the soft decision decoding. Algorithm: Message Passing (MP). The calculation unit and storage unit requirements of the BF algorithm are very low, but the performance is poor and the error correction capability is limited. The hard decision decoding algorithm has the characteristics of low decoding complexity and easy hardware implementation. It is simpler in hardware design, but its decoding performance is poor in bit error rate and has limited error correction capabilities. Therefore, hard decision Decoding algorithms are gradually replaced by soft-decision decoding algorithms.

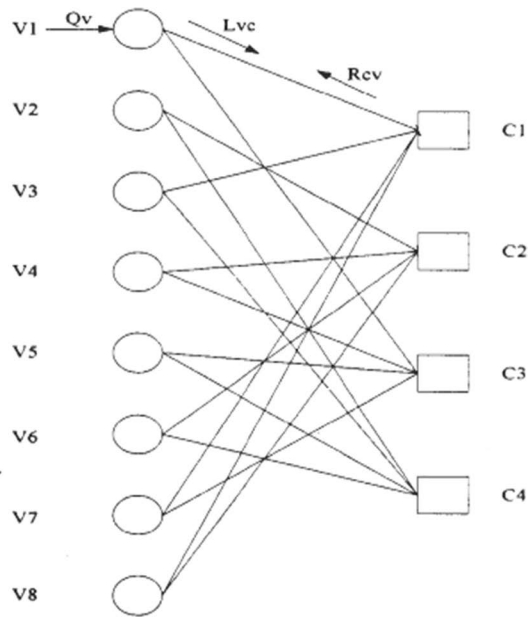


Figure 1. Tanner graph.

The Belief Propagation (BP) algorithm is proposed on the basis of the hard decision decoding algorithm, but exponential and logarithmic functions need to be used in the decoding process, which is complicated in operation and difficult to implement in hardware. In order to facilitate hardware implementation, the BP algorithm can be simplified into a minimum sum (MS) algorithm.

LDPC iterative decoding is updated and corrected according to the log-likelihood ratio (LLR) information received by the channel and the check relationship between channel information $\{f_n\}$. v_n represents the variable node corresponding to the n th column, c_m represents the check node corresponding to the m th row. Define $M(n) = \{m : h_{mn} = 1\}$ as the set of check nodes connected to the variable node, $N(m) = \{n : h_{mn} = 1\}$ as the set of variable nodes connected to the check node. C_{mn}^{iter} and V_{mn}^{iter} respectively represent the LLR transferred from v_n to c_m and the LLR transferred from c_m to v_n during the i -th iteration [4].

The corresponding algorithm is described as follows:

1) Initialization: set the initial values of all C_{mn}^{iter} , V_{mn}^{iter} and iteration counter iter to 0, and the initial value of v_n to the received likelihood information.

2) Check node update: update the information of v_n connected with c_m according to the principle of information transmission. The update formula is as follows: sdadasd

$$V_{mn}^{iter} = \prod_{n' \in N(m) \setminus n} \text{sign}(C_{m'n'}^{iter}) \cdot \partial \cdot \min_{n' \in N(m) \setminus n} |C_{m'n'}^{iter}| \tag{1}$$

3) Variable node update: Pass the information of c_m connected to v_n to the variable node, and update with the information carried by v_n itself:

$$C_{mn}^{iter} = f_n + \sum_{m' \in M(m)} V_{m'n}^{iter} \tag{2}$$

4) Result decision verification: the data in the updated data node is judged, and the judgment result is verified according to the check matrix. If the verification result is satisfied, or the iteration counter iter reaches the predetermined maximum value, the iteration is stopped and the result is output as the final result; otherwise, the iteration is continued according to steps 2) and 3), and the value of ITER counter is increased by 1. The flowchart is shown in **Figure 2**.

2.3. Multi-Rate LDPC Code Decoding

In the actual application process, in order to meet the needs of different scenarios, most communication systems choose different types and parameters of LDPC codes according to different scenarios. Generally, except for some special LDPC codes, LDPC codes with different types and parameters have different decoder structures. In this case, the system often needs to integrate multiple LDPC decoders, which limits the utilization of hardware resources. Therefore, according to the actual needs of users, it is necessary to design a LDPC decoder which can flexibly configure the code length and code rate according to the actual channel conditions.

In 2.1, we introduced Tanner graph. There are two types of nodes in Tanner graph, namely variable node and verification node. According to the decoding algorithm introduced in 2.2, LDPC code decoding is realized by iteratively transferring information between the two types of nodes. Since the calculation process of each variable node is independent of other variable nodes, and the calculation process of each check node is also independent of other verification nodes, we naturally think of mapping the Tanner diagram of LDPC code directly

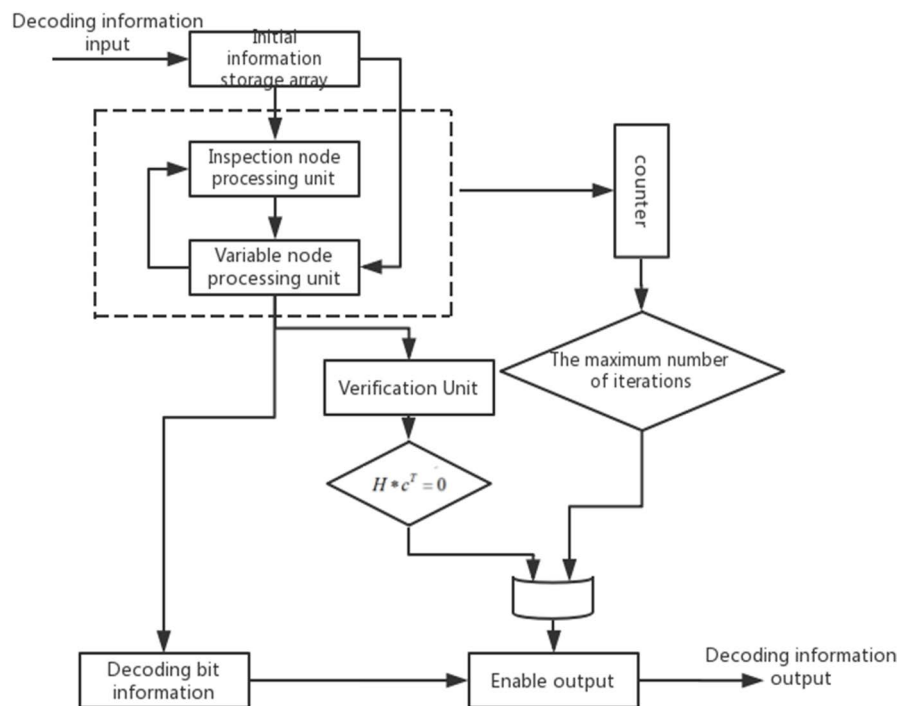


Figure 2. Decoding flowchart.

to the hardware structure.

The hardware structure mapped is shown in **Figure 3**, in which the functions of check node module and variable node module (CNFUs and VNFUs, collectively referred to as NFUs module) correspond to steps 2) and 3) of the above steps; $M(r,c)$ is the information storage module corresponding to the sub block in row r and column c of LDPC code parity check matrix. The capacity of each module is the same as that of check matrix sub block. The starting position of access depends on the offset of each sub block.

For an LDPC code with a certain code length and code rate, the minimum value module and the sum module in the decoder structure have a fixed number of data involved in the operation. When the code length or code rate of the LDPC code is smaller, it corresponds to fewer storage units in the hardware structure, that is to say, the total number of storage units in the figure is $r \cdot c$, and the storage units involved in the actual decoding process is not more than $m \cdot n$ ($m < r, n < c$). At this time, the number of input data ports preset by the corresponding NFUs corresponds to r and c , respectively. Therefore, in the implementation process, the decoder with a larger memory array can be configured by invalidating some input terminals in the NFUs module to achieve the purpose of decoder multiplexing [5].

3. Simulation Analysis

For matrices, especially sparse matrices, MATLAB has more function instructions and extremely rich drawing functions, which can show the simulation results more vividly, so we rely on MATLAB for research.

The process is shown in **Figure 4**. The input information of LDPC code encoder

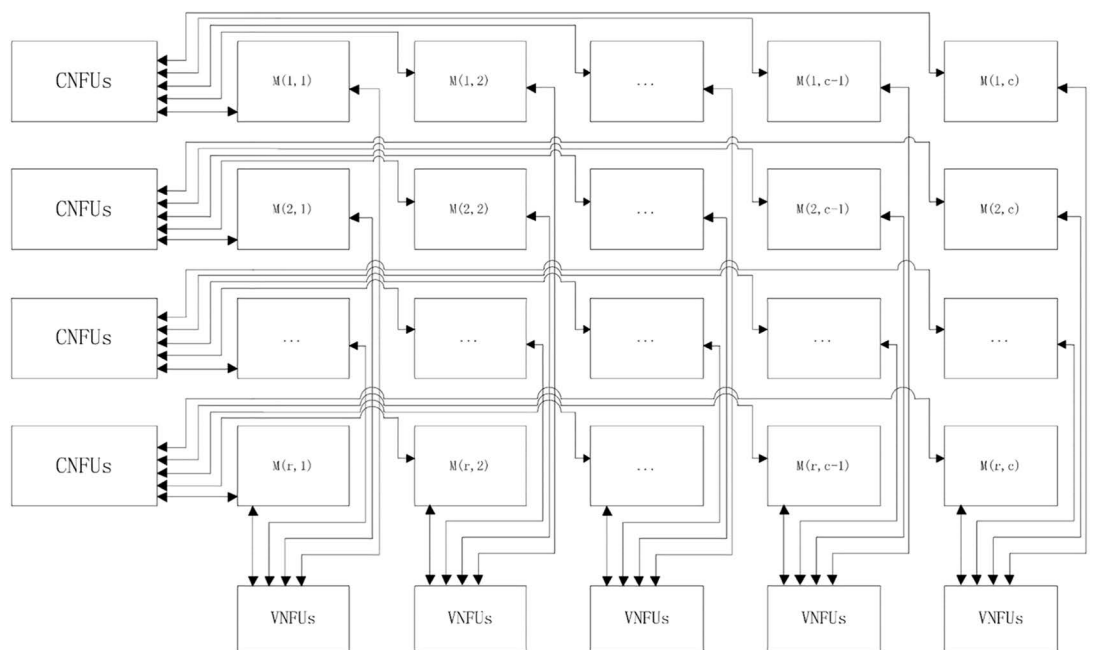


Figure 3. Hardware structure.

is $C = (c_1, c_2, \dots, c_k)$, and the output code word of LDPC code encoder is $X = (x_1, x_2, \dots, x_n)$. Then, through BPSK modulation, the transmission code word is $S = (s_1, s_2, \dots, s_n)$. The transmitted code word S passes through the AGWN channel with mean value of 0 and variance of σ , and then demodulated by BPSK demodulator and decoded by LDPC decoder. The code word received by LDPC decoder is $y_n = s_n + w_n$. w_n is the additive white Gaussian noise in the channel, and the decoded code word obtained by LDPC decoder is \hat{X} . By comparing C and \hat{X} , the BER of transmission is obtained [6].

The simulation results under different iterations are shown in **Figure 5**. Corresponding to the same SNR, the BER performance of LDPC code has been significantly improved with the increase of iteration times. The decoding performance has been significantly improved with different iterations starting from 1.5 dB. The 10 iterations are about 1 dB higher than that of 3 iterations and 0.5 dB higher than that of 5 iterations. The performance of the first few iterations is improved obviously, and the change decreases and tends to be stable as the number of iterations increases. With the increase of the correct rate of information obtained by each node, the decoding accuracy will be higher, and the final

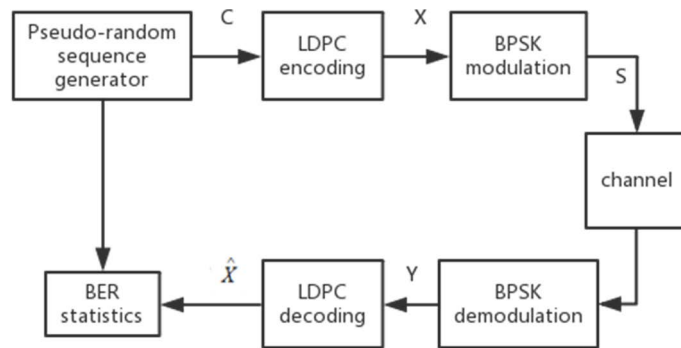


Figure 4. Simulation block diagram.

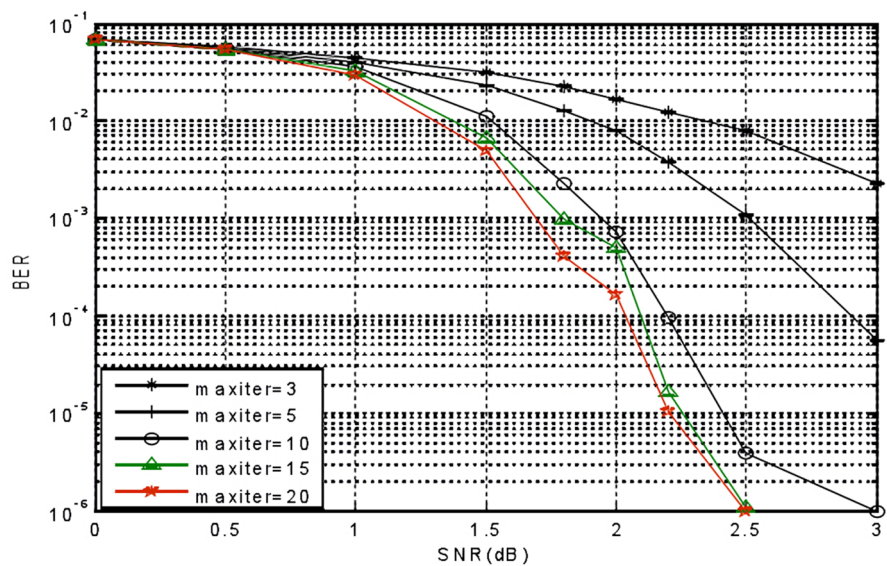


Figure 5. Simulation with different iterations.

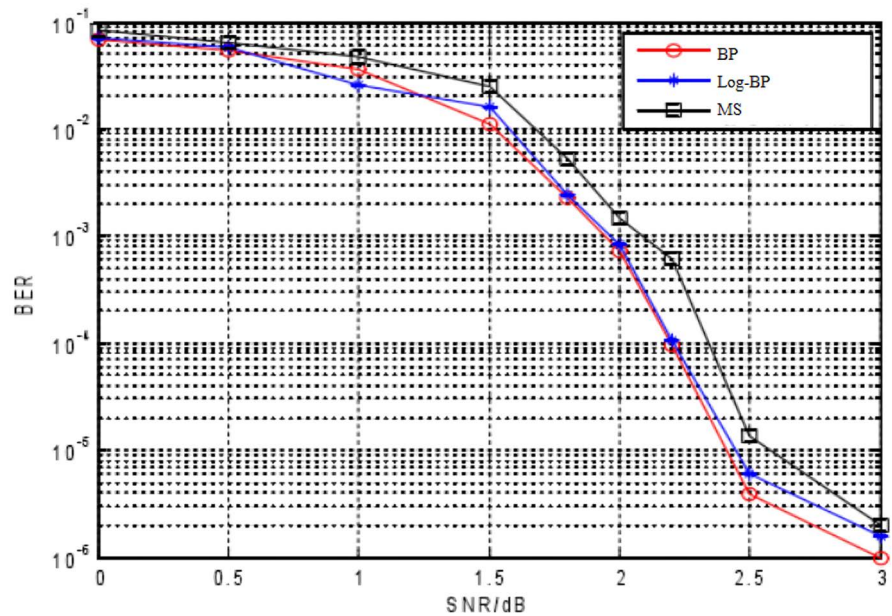


Figure 6. Simulation of different algorithms

bit error rate will be correspondingly lower.

The simulation results of different algorithms under the same number of iterations are shown in **Figure 6**. Judging from the decoding performance in the figure, the belief propagation algorithm is the best decoding algorithm. The BP algorithm in the logarithmic domain is a simplification of the BP multiplication operation, which can simplify its large number of multiplication operations into addition, which simplifies the decoding time complexity. The minimum sum algorithm decoding is relatively poor because, based on the Log-BP, the minimum value operation is used to simplify the function calculation and approximate, so the decoding performance has a certain degree of degradation. It can be seen from the figure that the decoding performance of the MS algorithm is about 0.1 dB lower than that of the BP algorithm. Nevertheless, since only two operations of addition and comparison are used in the minimum sum algorithm, it is still more suitable for hardware implementation.

4. Conclusion

In this paper, the principle of the multi-rate decoding of LDPC codes in optical communication system is analyzed in depth. On this basis, the simulation based on MATLAB shows that the corresponding multi-rate decoding can be realized by using the MS algorithm and selecting the appropriate iteration times.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

- [1] Yuan, J.G., Sun, X.M., Wang, Z., Zeng, L., Hu, X.Y. and Wu, Y.D. (2018) Research

Progress on Decoding Algorithm of LDPC Codes in Optical Communication Systems. *Laser Journal*, **39**, 1-7.

- [2] Zhao, D.F., Zhao, H. and Xu, Y.Z. (2012) Design and Implementation of Configurable LDPC Decoder Based on FPGA. *Journal of Natural Science of Heilongjiang University*, **29**, 259-264.
- [3] Wen, H., Fu, C.S. and Zhou, L. (2006) Principle and Application of LDPC Code. University of Electronic Science and Technology Press, Chengdu.
- [4] Chen, F.T., Liu, Y.F. and Tang, C. (2020) A Low-Complexity Normalized Min-Sum Decoding Algorithm for LDPC Codes. *Journal of Chongqing University of Posts and Telecommunications (Natural Science Edition)*, **32**, 92-98.
- [5] Zhai, X.F., Li, M.Q. and Wang, C. (2019) Design of a Reusable QC-LDPC Decoder. *Video Engineering*, **43**, 115-120.
- [6] Zhang, X.H. and Peng, S.F. (2012) Research and Implementation of LDPC Code Based on MATLAB. *Public Communication of Science & Technology*, **4**, 234-235.