

Intensity Reflection Coefficient based Min-Sum Decoding for Low Density Parity Check Codes

Md. Moshir Rahman FARAZI¹, Mohammad Rakibul ISLAM², Khandaker Sultan MAHMOOD³, Md. Farhan Tasnim OSHIM⁴

Dept. of Electrical and Electronic Engineering, Islamic University of Technology
Board Bazar, Gazipur-1704, Bangladesh

E-mail: ¹moshiur22@gmail.com, ²rakibultowhid@yahoo.com, ³ksm_pro@hotmail.com, ⁴farhan_oshim@yahoo.com

Abstract — Low Density Parity Check Codes (LDPC) give groundbreaking performance which is known to approach Shannon's limits for sufficiently large block length. Historically and recently, LDPC have been known to give superior performance than concatenated coding. In the following paper, a proposal to modify the standard Min-Sum (MS) algorithm for decoding LDPC codes is presented. This is done by introduction of a factor, intensity reflection coefficient (IRC) in the check to bit node updating process. Simulation results demonstrate that the proposed algorithms are effective in imparting a better performance in terms of a lower bit error rate (BER) at low to medium signal to noise ratio (SNR) when compared to the traditional MS or Belief Propagation (BP) algorithm while adding a minimum amount of complexity.

Keywords — Low-Density Parity-Check codes, Min-Sum algorithm, Belief Propagation algorithm, IRC, bit error rate (BER), computational complexity.

I. INTRODUCTION

Low density parity-check (LDPC) codes were first proposed by Gallager in 1963 in [1] and after more than 30 years of research, it was rediscovered by Mackay and Neal [2] and Sipser and Spielman [3]. An extensive research focused on binary LDPC codes have shown to achieve a rate very close to the Shannon's limit. Such codes have already been considered in communication standards such as digital video broadcasting (DVB-S2), Wi-Fi (802.11n), WiMax (802.16e) and 10 Gigabit Ethernet (10GBASE-T). Codes of various schemes, with a wide range of complexity and fairly effective to highly accurate performance have been achieved.

Since rediscovery of Mackay and Neal, several researchers have proposed various algorithms which are known to achieve performance with LDPC code near Shannon limit. M. Fossorier, et al. [5] proposed a new algorithm to reduce the complexity of LDPC code based on belief propagation. Improvement of belief propagation decoding for LDPC code was done by K. Chung, et al. [6], Yuan-Mao Chang, et al. [7], N. Varnica, et al. [8], S. Gounai, et al. [9] and Nedeljko Varnica, et al. [10]. For the Binary Input Additive White Gaussian noise (BIAWGN) channel, an LDPC code of length one million was constructed by Richardson, et al. [11] achieved a bit-error probability of 10^{-6} less than 0.13 dB away from capacity, surpassing the best (Turbo) codes known hitherto. On that

basis, Chung, et al. [12] constructed another LDPC code, which achieved within 0.04 dB of the Shannon limit (a theoretical capacity of any channel set by Shannon for randomly constructed code) at a bit error rate of 10^{-6} using a block length of 107.

The paper is organized as follows: Low density parity check code is discussed in Section II. Section III and IV reviews standard Belief Propagation (BP) and Min-Sum (MS) decoding algorithms respectively. Our proposed algorithm Min Sum algorithm with Intensity Reflection Coefficient (IRC) is discussed in Section V. Analysis of the computational complexity is done in Section VI. Simulation results and their interpretation are given in Section VII. Section VIII presents some concluding remarks on the paper.

II. LOW DENSITY PARITY CHECK CODES

LDPC codes are block codes with parity-check matrices H that contain only a very small number of non-zero entries. This sparseness of H is essential for an iterative decoding complexity that increases only linearly with the code length. The biggest difference between LDPC codes and classical block codes is how they are decoded. Classical block codes are generally decoded with Maximum Likelihood (ML) like decoding algorithms and so are usually short and designed algebraically to make this task less complex. However, LDPC codes are decoded iteratively using a graphical representation of their parity-check matrix and so are much longer, less structured, and designed with the properties of H .

A sparse binary parity check matrix $H = \{M \times N\}$ can be used to completely describe a binary (N, K) (W_c, W_r) LDPC code where N denotes the length and K denotes the dimension and $M \geq N - K$ due to redundant parity check sums. W_c denotes the number of ones in each column and W_r represent the number of ones in each row. If W_c and W_r are fixed, then code is termed as regular. An LDPC code may be represented by a bipartite graph, called Tanner graph. Each row of the parity check matrix represents a parity check constraint, v_i where $1 \leq i \leq M$ and each column represents the variable nodes, c_j where $1 \leq j \leq N$. A sample parity check matrix from [13] is given in (1).

$$H = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 1 \end{bmatrix} \quad (1)$$

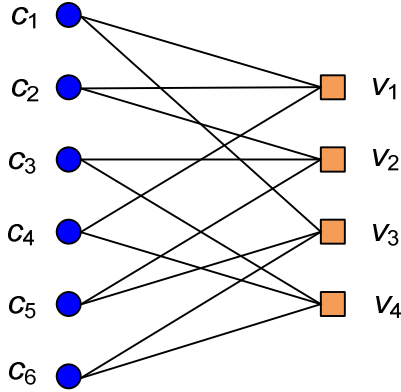


Figure 1. Tanner graph representation of a ($W_c = 2, W_v = 3$) LDPC code with code length $N = 6$ bits.

The corresponding Tanner Graph representation of the parity check matrix, H is shown Fig.1. For a given row, if the coded bits c_j are 1 then a connection is made between the coded bits, c_j and parity check equation, v_i in the tanner graph. Tanner graph is useful in representing links through which all the check nodes and variable nodes are updated in the decoding process.

Some of the extensively used LDPC decoders employ Belief Propagation (BP) [4] and Min-Sum (MS) [6] algorithms which are able to achieve highly efficient bit-error rate (BER) performance. The main decoding for LDPC codes is the BP algorithm but this requires complex computation of check nodes and hence it is difficult in hardware implementation. In the MS algorithm, the decoding performance is sacrificed by trading off with computational complexity. In order to obtain better decoding performance, an algorithm is devised that modifies the Min-Sum algorithm introducing a unique factor, Intensity Reflection Coefficient (IRC) to give a BER performance better than both MS and BP algorithm. This factor helps reduce the probability of sign changes in the decoded vector from which the codeword is obtained more efficiently.

III. STANDARD BELIEF PROPAGATION (BP) DECODING

BP decoding is a soft decoding approach to decode a transmitted message. Since log likelihood ratios are utilized, this algorithm is often termed as log likelihood decoding. Let a binary message $(x_1, x_2, \dots, x_N)$ is transmitted with N number of message bits and is transmitted over an Additive White Gaussian Noise (AWGN) channel. Let the received message be $(y_1, y_2, \dots, y_N)$ and binary hard decision obtained from the received message be $(b_1, b_2, \dots, b_N)$.

$$b_i = \begin{cases} 1, & y_i \leq 0 \\ 0, & y_i > 0 \end{cases} \quad \text{where } i = 1 \text{ to } N$$

For BP decoding, the extrinsic information passed between nodes, is given as probabilities rather than as hard decisions. In this algorithm, LDPC codes are decoded iteratively in two sequential steps: check node updating & variable node updating. During check node update, parity check operations are performed after receiving information from the neighboring variable nodes and results are sent back the neighboring variable nodes. For variable node update, the decoded bits and its corresponding soft information are updated from the check nodes and the results are sent back to the check nodes.

Notations:

LLR_j the information delivered by the log-likelihood ratio of received symbol y_i

$$LLR_i = \ln \left(\frac{p(x_i=0|y_i)}{p(x_i=1|y_i)} \right) \quad (2)$$

σ^2 channel variance

γ_{ij} message from check node i to variable node j .

δ_{ij} message from variable node j to check node i .

V_{ij} all variable nodes in V_i except node j .

C_{ij} all check nodes in C_j except node i .

The steps for decoding by BP algorithm is briefly discussed below:

a) *Initialization:* Set the iteration counter, $itr = 0$ and maximum number of iteration, $itr = l_{max}$. Set LLR of the channel output $LLR_{ch} = 2y_i/\sigma^2$ for Additive White Gaussian Noise (AWGN) channel. For each i and j , all δ_{ij} are initialized to the value of the extrinsic LLR of the received value y_j , which is LLR_j . Then the values of γ and δ are calculated and exchanged between the variable and check nodes until the parity-check equation is satisfied.

b) *Check node computation:* Compute γ_{ij} from the δ messages originating from all other variable nodes connected to check node C_i . The nonlinear function, φ is defined as,

$$\varphi(x) = -\log(\tanh \frac{|x|}{2}) \quad (3)$$

The φ function is used to update the check node using the following equation

$$\gamma_{ij}^{itr} = \prod_{j'' \in V_{ij}} \text{sign}(\delta_{ij''}^{itr}) \times \varphi(\sum_{j'' \in V_{ij}} \varphi(|\delta_{ij''}^{itr}|)) \quad (4)$$

- c) *Variable node computation:* Variable nodes are updated after completion of check node computation. δ_{ij} message are computed using the channel information, LLR_j and other check node values that are connected to that particular variable node V_j .

$$\delta_{ij}^{itr} = LLR_j + \sum_{i'' \in C_{ji}} \gamma_{i''j}^{itr} \quad (5)$$

After computing δ_{ij} , all the message bits in the j^{th} column are again updated by adding the channel information, LLR_j .

The syndrome check, ζ is defined as

$$\zeta_j = LLR_j + \sum_{i'' \in C_{ji}} \gamma_{i''j}^{itr} \quad (6)$$

- d) *Message computation:* From the updated vector ζ , the new estimated message bit is extracted. As this computation is done by the BP decoding, the decision criteria for the new estimated message x' will be

$$x'_j = \begin{cases} 1, & \text{if } \zeta_j \leq 0 \\ 0, & \text{if } \zeta_j > 0 \end{cases} \quad (7)$$

- e) *Validation:* The received code word is now x' . The stopping criterion is

$$\text{if } H \cdot x'^T = \begin{cases} 0, & x' \text{ is a valid code word} \\ \text{otherwise}, & x' \text{ is not a valid code word} \end{cases}$$

If the valid code word is obtained then the decoding stops otherwise the decoder again repeats iteration from step (b) onwards.

IV. MIN-SUM (MS) DECODING

The Min-Sum (MS) algorithm is a soft decoding approach to decode a transmitted message with lower complexity compared to BP Algorithm [14][15]. Let a binary message $(x_1, x_2, \dots, x_N)$ is transmitted with N number of message bits and is transmitted over an additive white Gaussian noise (AWGN) channel. Let the received message be $(y_1, y_2, \dots, y_N)$ and binary hard decision obtained from the received message be $(b_1, b_2, \dots, b_N)$.

$$b_i = \begin{cases} 1, & y_i \leq 0 \\ 0, & y_i > 0 \end{cases} \quad \text{where } i = 1 \text{ to } N$$

In the MS algorithm, LDPC codes are decoded iteratively in two sequential steps similar to BP algorithm: check node updating & variable node updating. During check node update, parity check operations are performed after

receiving information from the neighboring variable nodes and results are sent back the neighboring variable nodes. For variable node update, the decoded bits and its corresponding soft information are updated from the check nodes and the results are sent back to the check nodes.

The steps for this decoding algorithm are as follows:

- a) *Initialization:* Initialize δ_{ij} with the channel reliability (CR)

$$\delta_{ij}^{itr=0} = CR_j = \frac{2y_j}{\sigma^2}, \quad j = 0, \dots, N-1, i \in M(j) \quad (8)$$

- b) *Check node computation:* The check node update equation is given by

$$\gamma_{ij}^{itr} = \left(\prod_{j'' \in V_{ij}} \text{sign}(\delta_{ij''}^{(itr-1)}) \right) \times \left(\min_{j'' \in V_{ij}} |\delta_{ij''}^{(itr-1)}| \right) \quad (9)$$

- c) *Variable node update:* The variable node update equation is given by

$$\delta_{ij}^{itr} = CR_j + \prod_{i'' \in C_{ji}} \gamma_{i''j}^{itr} \quad (10)$$

The message is then computed to an estimated code word.

$$\zeta_j = CR_j + \sum_{i'' \in C_{ji}} \gamma_{i''j}^{itr} \quad (11)$$

- d) *Message Computation:* The decision criterion for this algorithm is given as follows,

$$x_j = \begin{cases} 0, & \text{if } \zeta_j > 0 \\ 1, & \text{otherwise} \end{cases}$$

- e) *Validation:* The received codeword is x and the stopping criterion is

$$\text{if } H \cdot x^T = \begin{cases} 0, & x \text{ is a valid code word} \\ \text{otherwise}, & x \text{ is not a valid code word} \end{cases}$$

If the valid code word is obtained then the decoding stops otherwise the decoder again repeats iteration from step (b) onwards.

V. PROPOSED ALGORITHM

A. Idea of the Proposed Algorithm

Our proposed algorithm introduces a factor, κ in the check to bit node updating process in the MS algorithm described in step (b). This factor κ lies between zero 0 and 1 i.e. $\kappa \in [0,1]$ and is computed as the square of the ratio of the difference of the estimated codeword to the sum of the estimated codeword in present and preceding iteration and is termed as the factor IRC.

The idea of IRC egresses from phenomena of propagation of sound waves e.g. ultrasonic wave between two mediums having discontinuity in densities [16]. In classical engineering physics, when a sound wave is incident on an interface between two mediums of different densities, part of it is reflected back into the original

medium. The amount of energy reflected back depends on impedance (*i.e.* acoustical impedance) mismatch between the mediums on each side of the interface.

The coefficient of reflection is given by the percentage of energy that is reflected also termed as Intensity Reflection Coefficient (IRC) and is calculated by the ratio of the intensity of the reflected wave to the incident wave.

$$\Gamma = \left(\frac{Z_1 - Z_2}{Z_1 + Z_2} \right)^2$$

Where,

- Γ = Intensity reflection coefficient (IRC)
- Z_1 = Impedance of medium 1
- Z_2 = Impedance of medium 2

If we analogize the phenomenon of impedance (Z) with the syndrome check (ζ) computation in the variable node update, then the Intensity Reflection Coefficient (IRC) can be expressed as

$$\kappa = \left(\frac{\zeta_1 - \zeta_2}{\zeta_1 + \zeta_2} \right)^2$$

Where,

- κ = Intensity reflection coefficient
- ζ_1 = Syndrome check of the previous iteration
- ζ_2 = Syndrome check of the present iteration

B. Min Sum algorithm with Intensity Reflection Coefficient (IRC)

Check to bit node update per iteration may be regarded as a particular medium. As the number of iteration changes, the characteristics of the media changes which in turn affects the probability of error in the transmitted bit. The factor κ characterizes the computational error in check to bit node updating process. The factor is introduced to minimize the probability of sign changes during check node update in later iterations. Similar to classical physics, a value of $\kappa = 0$ signifies the absence of errors during check to bit node update whereas a value of $\kappa = 1$ signifies that the modified algorithm is equivalent to the MS algorithm. The modification is described as:

if

$$\prod_{j'' \in V_{ij}} \text{sign}(\delta_{ij}^{(itr)} \delta_{ij''}) \neq \prod_{j'' \in V_{ij}} \text{sign}(\delta_{ij}^{(itr-1)} \delta_{ij''})$$

then

$$\gamma_{ij}^{itr} = \kappa \times \left(\prod_{j'' \in V_{ij}} \text{sign}(\delta_{ij}^{(itr-1)} \delta_{ij''}) \right) \times \left(\min_{j'' \in V_{ij}} |\delta_{ij}^{(itr-1)} \delta_{ij''}| \right)$$

else

$$\gamma_{ij}^{itr} = \left(\prod_{j'' \in V_{ij}} \text{sign}(\delta_{ij}^{(itr-1)} \delta_{ij''}) \right) \times \left(\min_{j'' \in V_{ij}} |\delta_{ij}^{(itr-1)} \delta_{ij''}| \right)$$

Variable node update, message computation and validation step essentially remains the same as MS algorithm.

VI. COMPUTATIONAL COMPLEXITY

The computational complexity of the decoding algorithms is analyzed in this section. The analysis is done by determining the number of arithmetic operations involved in each algorithm per iteration and the corresponding time delays associated with it. From Table – I, it is possible to calculate the relative complexity of each algorithm. MS decoding algorithm being the least complex is assigned a relative complexity of 1, while the complexity BP decoding algorithm is calculated 7578.92, signifying its complex hardware implementation. This is, however, offset by the better convergence characteristics and higher BER performance of the algorithm.

It should be noted that the relative complexity of IRC-MS has nearly the same complexity as MS algorithm, while IRC-MS-Constant yields an even lower complexity compared IRC-MS. This is to be expected as the value of the factor, κ is a constant in IRC-MS-Constant and thus requires fewer numbers of arithmetic operations. The delay for the proposed algorithm MS-IRC lies between the delay of BP and the delay of MS algorithm.

VII. SIMULATION

In this section the effectiveness of the proposed algorithm are verified by considering a regular (7,11) LDPC code which is a popular code for IEEE 802.16e standard. The LDPC code has a codeword length, $N = 1944$ and information length, $K = 972$ which corresponds to a code rate of 0.5. The LDPC code has no short cycle of length 4. All the figures were obtained with a maximum of 20 iterations.

Table – I.
COMPUTATIONAL COMPLEXITY PER DECODING ITERATION

Algorithm	Addition	Subtraction	Multiplication	Division	Log(tanh(x))	Relative Complexity	Average Iteration Completion Time per frame per SNR dB
Min-Sum(MS)	1936386	139320	40821408	0	0	1	2.6110 seconds
Belief Propagation(BP)	40451400	2.6325e+011	40825296	2.6325e+011	278640	7578.92	2.9703 seconds
Min-Sum with IRC factor(MS-IRC)	1938330	176256	40897479	36936	0	1.003	2.7367 seconds
Min-Sum with constant IRC factor	1936386	139320	40862731	0	0	1.001	2.6899 seconds

Fig. 2 depicts four curves: BP, MS, IRC-MS and IRC-MS-Constant. It is observed that the proposed algorithm IRC-MS algorithm gives a better BER performance than both MS and BP algorithm for the SNR range of -1 to 1.5 dB. The graph of IRC-MS-Constant was obtained by choosing an arbitrary constant value of $\kappa = 0.0125$. The proposed algorithm, IRC-MS achieves an approximately 0.4 dB improvement over MS algorithm at BER of 3.755×10^{-2} . This is because IRC-MS takes into account the probability of error in each check to bit node update.

Fig. 3 was obtained for the same algorithms as mentioned above but $2/3$ rate (1944,648) LDPC code was used. Hence a comparative analysis may be done for the two rates ($1/2$ and $2/3$) for different algorithms. In fig. 3, it is noticed that significant fall in the BER curves were obtained in the range -1 to 3 dB. The IRC-MS algorithm still yields a better performance over MS algorithm and approaches very close to the BP algorithm.

In fig. 4, the performance of IRC-MS algorithm is obtained over LDPC codes of various rates. A lower code rate *i.e.* more redundancy, serves the purpose of improving BER performance by trading off with throughput. Among all the rates, a $1/2$ rate LDPC code yields the best performance for the proposed algorithm.

Fig. 5 displays a graph in which various fading channels were incorporated with the proposed IRC-MS algorithm. The classical AWGN channel which is free of any inter-symbol interference produced the best result in terms of BER performance. Small-scale fading is represented by Rayleigh fading and large-scale fading by Log-Normal fading. Weibull fading yields the poorest BER performance for IRC-MS among all the fading channel models used.

X. CONCLUSION

The LDPC codes are capable of nearing Shannon capacity performance when decoded with iterative message passing decoders such as MS or BP algorithm which are soft decoding approach. In this paper, a decoding algorithm has been proposed by modifying MS algorithm. We have introduced a new factor named IRC, κ which was constantly updated during each iteration. The factor is computed as a vector rather than a single valued constant. This modification allowed us to minimize the effect of sudden sign changes in the decoded vector during the check

to bit node update in successive decoding iterations. Simulation results reflect on the fact, modifications meliorate BER performance up to 1.5 dB SNR value over AWGN channel as well as improve the decoder's convergence behavior while keeping the computational complexity minimal. The proposed algorithm may be regarded as an application specific design where improvement in BER performance is desired for low to medium SNR values with low level complexity as our SNR range was from -1 to 1.5 dB when a $1/2$ rate code was used.

Analyses of the proposed decoding techniques are required for irregular LDPC codes and for non-binary instances. It has been proven that non-binary LDPC codes have shown performance which is advantageous over its binary counterpart; however, this is also known to give rise to a much higher computational complexity. Development of these algorithms is still an open challenge for non-binary cases.

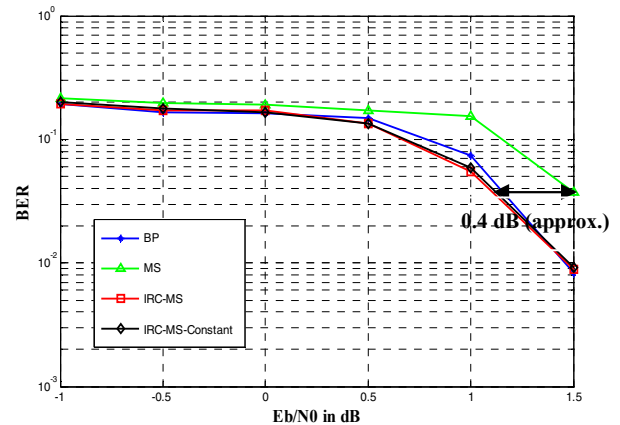


Figure 2. BER performance of proposed algorithm with $1/2$ rate (1944, 972) symbol (7, 11)-regular LDPC codes with iteration number = 20.

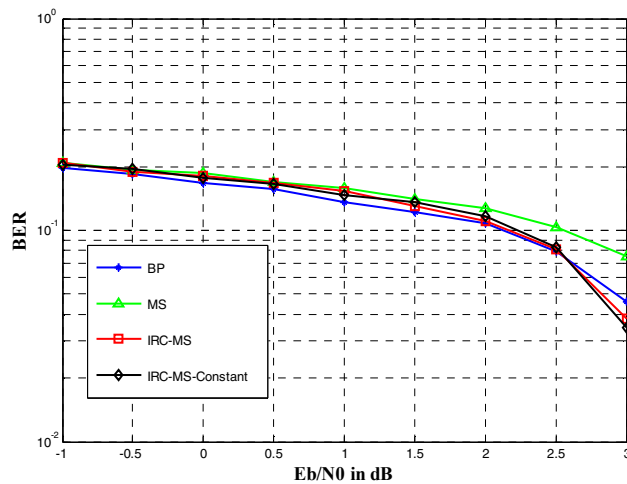


Figure 3. BER performance of proposed algorithm with $2/3$ rate (1944, 648) symbol (8, 11)-regular LDPC codes with iteration number = 20.

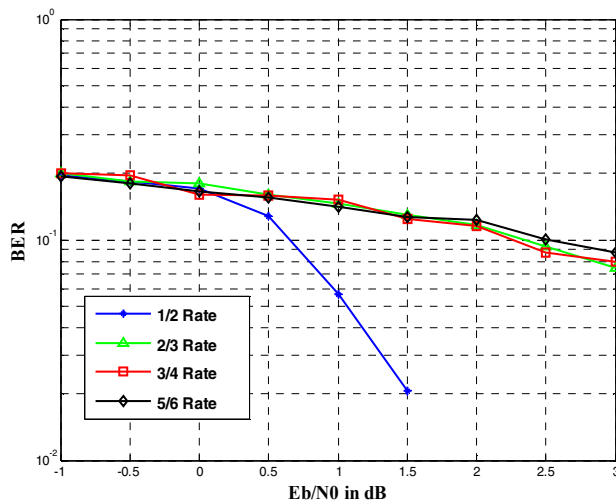


Figure 4. BER performance of proposed algorithm for various rate LDPC codes

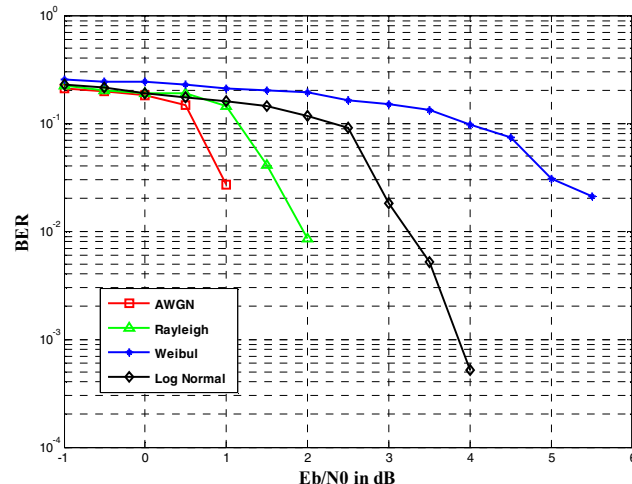


Figure 5. BER performance of the proposed algorithm for different fading channels.

REFERENCES

- [1] R. G. Gallager, "Low-Density Parity-Check Codes". Cambridge, MA: MIT Press, 1963
- [2] J. C. Mackay and R. M. Neal, "Near Shannon limit performance of low-density parity-check codes," *Electron. Lett.*, vol. 32, pp. 1645-1646, Aug. 1996.
- [3] S. Lin and D. J. Costello, Jr., *Error Control Coding: Fundamentals and Applications*. Englewood Cliffs, NJ: Prentice Hall, 1983.
- [4] J. C. MacKay, "Good error-correcting codes based on very sparse matrices," *IEEE Trans. Inform. Theory*, vol. IT-45, pp. 399-432, Mar.1999.
- [5] M. Fossorier, M. Mihaljevic, and H. Imai, "Reduced complexity iterative decoding of low-density parity-check codes," *IEEE Trans. Commun.*, vol. 47, pp. 673-680, May 1999.
- [6] Chung, k. and Huo, J. "Improved Belief Propagation (BP) Decoding for LDPC Codes with a large number of short cycles" *IEEE 63rd Vehicular Technology Conference*, 2006. VTC 2006-Spring.
- [7] Yuan-Mao Chang; Vila Casado, A.I.; Chang, M.-C.F.; Wesel, R.D. "Lower-Complexity Layered Belief-Propagation Decoding of LDPC Codes" *IEEE International Conference on Communications*, 2008. ICC'08.
- [8] Varnica, N.; Fossorier, M.P.C.; Kavcic, A.; , "Augmented Belief Propagation Decoding of Low-Density Parity Check Codes," *Communications, IEEE Transactions on* , vol.55, no.7, pp.1308-1317, July 2007
- [9] Gounai, S.; Ohtsuki, T.; Kaneko, T.; "Modified Belief Propagation Decoding Algorithm for Low-Density Parity Check Code Based on Oscillation" *IEEE 63rd Vehicular Technology Conference*, 2006. VTC 2006-Spring. Volume: 3 Page(s): 1467 – 1471.
- [10] Varnica Nedeljko; Fossorier Marc; , "Improvements in belief-propagation decoding based on averaging information from decoder and correction of clusters of nodes," *Communications Letters, IEEE* , vol.10, no.12, pp.846-848, December 2006
- [11] Richardson, T. J.; Shokrollahi, A.; and Urbanke R., "Design of capacity-approaching low-density parity-check codes," *IEEE Trans. Inform. Theory*, vol. 47, Feb. 2001, pp. 619-637.
- [12] S.Y. Chung, J. G. D. Forney, T. Richardson, and R. Urbanke, "On the design of low-density parity-check codes within 0.0045 dB of the Shannon limit," *IEEE Commun. Lett.*, vol. 5, Feb. 2001, pp. 58-60.
- [13] R. A. Carrasco, M. Johnston, "Non-binary error control coding for Wireless Communication and Data Storage", Chichester, WS: John Wiley & Sons, Ltd., 2008.
- [14] N. Wiberg, *Codes and Decoding on general graphs*, Ph.D. dissertation, Linkoping University, Sweden, 1996.
- [15] M.C.Davey, *Error-correction using Low-Density Parity-Check Codes*, University of Cambridge, British, 1999.
- [16] Joseph J. Carr, John M. Brown, "Medical Ultrasonography" in *"Introduction to Biomedical Equipment Technology"* 3rd ed., NJ, Prentice Hall, 1998, ch. 17, sec. 17.4-6, pp. 482-488.