A novel soft reliability-based iterative majority-logic decoding algorithm with uniform quantization*

YUAN Jian-guo (袁建国)**, **WANG Zhe** (汪哲), **HE Chang-wei** (何昌伟), **LIN Jin-zhao** (林金朝), **and PANG Yu** (庞宇)

Chongqing Key Laboratory of Photoelectronic Information Sensing and Transmitting Technology, Chongqing University of Posts and Telecommunications, Chongqing 400065, China

(Received 27 February 2017 ; Revised 24 March 2017)

In this paper, a novel soft reliability-based iterative majority-logic decoding algorithm with uniform quantization is proposed for regularly structured low density parity-check (LDPC) codes. A weighted measure is introduced for each check-sum of the parity-check matrix and a scaling factor is used to weaken the overestimation of extrinsic information. Furthermore, the updating process of the reliability measure takes advantage of turbo-like iterative decoding strategy. The main computational complexity of the proposed algorithm only includes logical and integer operations with the bit uniform quantization criterion. Simulation results show that the novel decoding algorithm can achieve excellent error-correction performance and a fast decoding convergence speed.

Document code: A **Article ID:** 1673-1905(2017)03-0217-4

DOI 10.1007/s11801-017-7046-1

Low density parity-check (LDPC) codes are currently the most promising channel coding technique to approach the Shannon capacity limitation for a wide range of channels. Due to the excellent error-correction performance and the low structure complexity, LDPC codes have become a hot research topic, and a great deal of research effort has been expended in design, construction, encoding, decoding and applications of LDPC codes[1-4]. Now LDPC codes have been constructed by different methods and a variety of decoding algorithms have been proposed for LDPC codes to make a balance between decoding performance and complexity.

The algorithms for decoding LDPC codes can be classified into three general categories[5-8]: soft-decision decoding, hard-decision decoding, and hybrid decoding (also called the reliability-based decoding). From an implementation point of view, the soft-decision decoding algorithms, such as belief propagation (BP) decoding algorithm and its simplified versions, retain all the information provided by the channel. So they can provide the best error-correction performance and the highest decoding convergence speed, but need high computational complexity among all decoding algorithms. The hard-decision decoding, such as the bit-flipping (BF) algorithm and the one-step majority-logic decoding (OSMLG) algorithm, quantizes the received symbols to 0 or 1 before decoding process and loses most of the channel information[8]. It has the lowest complexity and is easy for hardware implementation, but the low complexity results in serous performance degradation.

In contrast to the three decoding methods, the reliability-based decoding algorithm offers efficient trades-off between computational complexity and decoding performance. The well-known reliability-based decoding ones are weighted bit-flipping (WBF) and soft reliability-based iterative majority-logic decoding (SRBI-MLGD) algorithms. The SRBI-MLGD algorithm is a binary message-passing reliability-based algorithm and only requires logical operations and integer additions, thus it has significantly low decoding complexity while maintaining good performance[9]. But in the decoding process, all check-sums have the same reliability. However, the reliability of the received symbol is different, which means that the reliability of check-sums is also different. So an improved version of the SRBI-MLGD was presented by Ngatched, which introduces the weighted measure for each check-sum[10,11].

This paper presents a novel SRBI-MLGD algorithm for constructing binary LDPC codes with uniform

* This work has been supported by the National Natural Science Foundation of China (Nos.61472464, 61671091 and 61471075), the Natural Science Foundation of Chongqing Science and Technology Commission (No.cstc2015jcyjA0554), the Program for Innovation Team Building at Institutions of Higher Education in Chongqing (No.J2013-46), and the Undergraduate Science Research Training Project for Chongqing University of Posts and Telecommunications (No.A2016-61).

** E-mail: yuanjg@cqupt.edu.cn

quantization criterion. A weighted measure is introduced for each check-sum of the parity-check matrix. To avoid the mistaken assessments of reliability measure in the iterative updated process, a scaling factor is used to weaken the overestimation of extrinsic information, and the reliability measure updating expression is modified with a turbo-like iterative decoding strategy. This proposed algorithm results in a significant error-correction performance improvement and high decoding convergence speed in the case of requesting very little additional computation.

A structured binary LDPC code can completely be described by a sparse parity-check matrix of $\boldsymbol{H}=[H_{mn}]$, whose dimension is $m\times n$. For a regular LDPC code, $\boldsymbol{H}$ is associated with a Tanner graph which contains the variable nodes v_j ($0<j<n$) that correspond to j columns of $\boldsymbol{H}$ and the check nodes c_i ($0<i<m$) that correspond to i rows of $\boldsymbol{H}$. Then we define the position of 1 at the jth column as $N(j)=\{i:0\leq i<m, H_{i,j}=1\}$, and the position of 1 at the ith row as $M(i)=\{j:0\leq j<n, H_{i,j}=1\}$.

Assume that a regular binary LDPC code has a sparse parity-check matrix $\boldsymbol{H}$, and $\boldsymbol{c}=(c_0, c_1..., c_{n-1})$ denotes an encoded LDPC codeword with the length of n. The codeword $\boldsymbol{c}$ is mapped into the bipolar sequence $\boldsymbol{x}=(x_0, x_1..., x_{n-1})$ by using binary phase shift keying (BPSK) signaling with unit energy per signal, where the jth component is $x_j=2c_j-1$ ($0\leq j<n$), $2c_j-1=+1$ stands for $c_j=1$ and $2c_j-1=-1$ stands for $c_j=0$. Then it is transmitted over the additive white Gaussian noise (AWGN) channel, and the received sequence is denoted by $\boldsymbol{y}=(y_0, y_1..., y_{n-1})$, $y_j=x_j+e_j$ ($0\leq j<n$), where e is a Gaussian random variable with the mean of 0 and the variance of σ^2.

Suppose the received sequences are symmetrically clipped at a threshold of y_{th} with b bits quantization. So the uniformly quantized range is $[-y_{\text{th}}, y_{\text{th}}]$, and there are 2^b-1 intervals in total. Each interval has a length of $\Delta=2y_{\text{th}}/(2^b-1)$. Let q_j denote the quantized value which is an integer of one of the 2^b-1 intervals as follows

$$q_j=\begin{cases}-(2^{b-1}-1), & \dfrac{y_j}{\Delta}\leq-(2^{b-1}-1)\\ \left\lfloor\dfrac{y_i}{\Delta}\right\rfloor, & -(2^{b-1}-1)<\dfrac{y_j}{\Delta}<2^{b-1}-1\\ 2^{b-1}-1, & \dfrac{y_j}{\Delta}>2^{b-1}-1\end{cases}, \tag{1}$$

where $\lfloor x\rfloor$ is a rounding symbol, indicating the nearest integer from x.

The SRBI-MLGD decoding algorithm can be described as a message passing between variable nodes and check nodes over the Tanner graph like the BP decoding algorithm. Assume that the n variable nodes $v_0, v_1..., v_{n-1}$ and m check nodes participate in message passing in Tanner graph. Syndrome vector can be calculated as $\boldsymbol{s}=\boldsymbol{z}\boldsymbol{H}^{\text{T}}=(s_0, s_1..., s_{m-1})$, where $\boldsymbol{z}$ means hard-decision sequence received from the AWGN channel. The algorithm can be divided into four main parts. **Step1** Initialization: The receiving terminal can get a received sequence of $\boldsymbol{y}=(y_0, y_1..., y_{n-1})$ after the signal transmitting in the channel. With the quantization standard of Eq.(1), the received sequence can switch into a sequence of integers like $\boldsymbol{q}=(q_0, q_1..., q_{n-1})$. Therefore, the initial reliability measure can be described as

$$R_j^{(0)}=q_j\ ,\ \ (0\leq j<n)\ . \tag{2}$$

Step2 Hard-decision process and the syndrome vector computation (Parameter k represents the kth iteration.):

$$z_j^{(k)}=\begin{cases}1, \text{ if } R_j^{(k)}\geq 0\\ 0, \text{ otherwise}\end{cases},\ (0\leq j<n), \tag{3}$$

$$s_i^{(k)}=\sum_{i=0}^{m-1}z_j^k H_{i,j}=\sum_{i\in M(i)}z_j^{(k)}\ (\text{mod}\,2),\ (0\leq j<n)\cdot \tag{4}$$

If all the parity check equations are satisfied, which means all check-sums are $s_i=0$, the output hard-decision sequence $\boldsymbol{z}$ is the decoded codeword. Otherwise, continue the iterative process until the correct codeword is got or the maximum number of iterations is reached.

Step3 Check-node updating process: Update the extrinsic information. Each check-sums s_i can be composed of self-information z_j and extrinsic check information $\sigma_{i,j}$ which is expressed as

$$\sigma_{i,j}^{(k)}=\sum_{j\in N(j)/n}z_j^{(k)}H_{i,j}=s_{i,j}^{(k)}+z_{i,j}^{(k)}(\text{mod}\,2),(0\leq i<m)\ . \tag{5}$$

Step4 Variable-node updating process:

The total extrinsic information is

$$e_j^{(k)}=\sum_{j\in N(j)}(2\sigma_{i,j}^{(k)}-1),\ (0\leq i<m)\,. \tag{6}$$

The reliability measure updating process is

$$R_j^{(k+1)}=R_j^{(k)}+e_j^{(k)},\ (0\leq j<n)\ . \tag{7}$$

It is important to note that due to the introduction of bits quantization criterion, the reliability measure of the received bit and the total extrinsic information are integers. As a result, only logical operations and integer additions are required to carry out the SRBI-MLGD algorithm[12,13].

For SRBI-MLGD algorithm, it treats every check-sum with the same reliability, but this situation could be true when the received symbols are equal. In fact, the received symbols are different with the influence of additive noise after channel transmission[14]. In this section, a novel SRBI-MLGD algorithm is proposed. The proposed algorithm utilizes the minimum received quantization integer which participates in the mth parity-check equation as the weighted measure χ for each check-sum. And then taking the iterative process into account may overestimate the extrinsic information, which results in performance degradation. A scaling factor θ is introduced to reduce the overestimate of the extrinsic information which is harmful for the decoding process. The value of the optimal scaling factor θ can be obtained through the density evolution theory and the simulation[5]. It shows that different LDPC codes have different optimal scaling factors, but this will not influence the signal to noise (*SNR*) significantly. So θ is optimized for an *SNR* and kept unchanged for all *SNRs* for each code. At last, a turbo-like iterative decoding strategy is applied for the reliability measures updating, which can avoid the impact of last updating process. The novel algorithm can be summarized as follows.

Step1 Initialization: Quantize the vector y_j into integer q_j according to Eq.(1), and set the initial reliability measure as $R^{(0)}= q_j$. For each check-sum, the weighted measure equals $\chi_{m,n} \cdot \min_{j\in N(m)/n} |q_j|$, where $\chi_{m,n}$ represents the minimum bit quantization value that is orthogonal to the check equation. The iteration loop counter is k=0<$I_{\max}$.

Step2 Hard-decision based on the following decision rule:

$$z_j^{(k)} = \begin{cases} 1, \text{ if } R_j^{(k)} \ge 0 \\ 0, \text{ otherwise} \end{cases}, \ (0 \le j < n) . \tag{8}$$

For 0≤i<m−1, compute the syndrome vector as

$$s_i^{(k)} = \sum_{j=0}^{n-1} z_j H_{i,j} = \sum_{j\in N_i} z_j^{(k)} \ (\text{mod}\, 2), \ (0 \le i < m) . \tag{9}$$

Stop iterative process until all the parity check-sums are 0 or the maximum number of iterations is reached, then $z_j^{(k)}$ is output as the decoded codeword.

Step3 For 0≤j<n−1, compute the total extrinsic information as

$$e_j^{(k)} = \left\lfloor \theta \sum_{i\in M(j)} (2\sigma_{i,j}^{(k)} - 1)\chi_{m,n} \right\rfloor , \ (0 \le j < n) . \tag{10}$$

Step4 Update the reliability measures of all the received bits as

$$R_j^{(k+1)} = R_j^{(0)} + e_j^{(k)}, \ (0 \le j < n) , \tag{11}$$

where $R_j^{(0)}$ indicates the initialization reliability of the received codeword, which does not change during the iteration. So the channel information is retained as much as possible.

If the computation complexity of initialization is ignored, the total computational complexity of the conventional SRBI-MLGD per iterative process requires δ integer additions and $2\delta+n-m$ logical operations, where $\delta=\rho m=\gamma n$ (ρ presents the row weight and γ presents the column weight). The proposed decoding algorithm needs $\delta-m$ more integer comparisons in $\chi_{m,n}$ and a register to storage $R^{(0)}$ in the initialization part. Each iteration process for the presented algorithm requires $2\delta+n-m$ logic operations, δ integer additions and n rounding operations to find the nearest integer. Thus the novel algorithm just costs a slightly higher computation to weaken extrinsic information overestimation, and the turbo-like strategy to update the reliability measures can have an improvement in performance.

Next, we use the simulation results to analyze the error-correction performance and the average iteration number for decoding. Set the simulation parameters as follows: QC-LDPC(961,721) code[15] is used; The maximum iteration number is set to be 30; The signal sequence adopts BPSK modulation and is transmitted in AWGN channel; The optimal factor is set to be 0.5; The 8-bits quantization is used.

The simulation results of error-correction performance for QC-LDPC(961,721) code are shown in Fig.1. We come to the following conclusion that compared with conventional SRBI-MLGD and ISRB-MLGD[11] algorithms, the proposed SRBI-MLGD decoding algorithm has almost 0.5 dB and 0.3 dB performance gains at bit error rate (*BER*) of 10^{-5}. The performance difference between BP algorithm and the presented algorithm is only 0.6 dB.

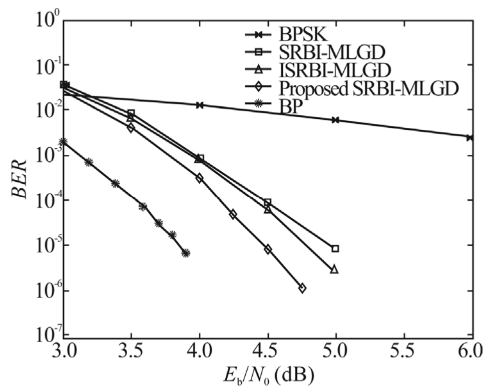

Fig.1 Error-correction performance of the QC-LDPC(961,721) codes with different decoding algorithms

From the simulation results in Fig.2, we can see that the decoding convergence speed of the proposed SRBI-MLGD is higher than those of the ISRBI-MLGD and conventional SRBI-MLGD algorithms. When the E_b/N_0 is around 3.0 dB, the presented decoding algorithm requires about 17.5 iterations on average for decoding successfully, but SRBI-MLGD and ISRB-MLGD algorithms need respectively almost 20 and 23 iterations. When the E_b/N_0 is more than 4.0 dB, the convergence speed of the novel algorithm is very close to that of BP algorithm and just needs about 2.5 iterations on average for decoding.

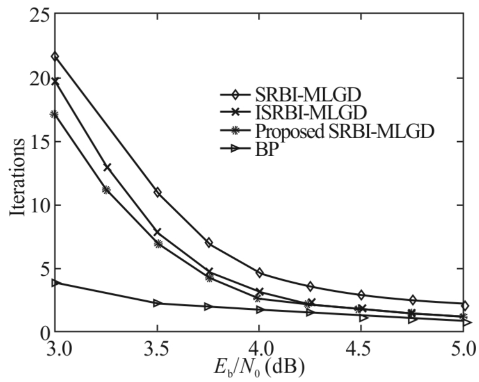

Fig.2 Average numbers of iterations of the QC-LDPC(961,721) codes with different decoding algorithms

In this paper, a novel SRBI-MLGD decoding algorithm with uniform quantization is proposed for regularly structured LDPC codes. The algorithm introduces a turbo-like iterative decoding strategy to update the reliability measure and uses weighted measure to calculate the reliability of each check-sum. At last, an optimal scaling factor is utilized to compensate the performance degradation by weakening overestimation of the extrinsic information. Simulation results show that the novel SRBI-MLGD algorithm has better error-correction performance than SRBI-MLGD and reduces the average number of iterations for decoding, which indicates that the algorithm has a higher decoding convergence speed than the original one with slightly increased complexity. In conclusion, the novel SRBI-MLGD is more suitable for high speed data transmission communication systems.

References

[1] Wojciech Sułek, IEEE Communications Letters **20**, 1705 (2016).

[2] Lei Tang, Fengfan Yang and Shunwai Zhang, IET Journals and Magazines **25**, 179 (2016).

[3] Zhang P, Yu S and Liu C, Electronics Letters **50**, 320 (2014).

[4] WEI Xia and ZHANG Wenjun, Journal of Chongqing University of Posts and Telecommunications (Natural Science Edition) **28**, 60 (2016). (in Chinese)

[5] Andre G. D. Uchoa, Cornelius T. Healy and Rodrigo C. de Lamare, IEEE Transactions on Vehicular Technology **65**, 2735 (2016).

[6] Yuan Jianguo, He Changwei, Gao Wenchun and Wang Yong, Journal of Chongqing University of Posts and Telecommunications (Natural Science Edition) **27**, 770 (2015). (in Chinese)

[7] Yuan Jian-guo, Tong Qing-zhen, Huang Sheng and Wang Yong, Optoelectronics Letters **9**, 0469 (2013).

[8] Saedong Yeo and In-Cheol Park, IEEE Communications Letters **21**, 230 (2017).

[9] S. Khittiwitchayakul, W. Phakphisut and P. Supnithi, Weighted Bit-flipping Decoding for Product LDPC Codes, 13th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology 1 (2016).

[10] Jiaying Wen, Guojun Han and Xusheng Lin, Performance of the Modified SRBI-MLGD Algorithm for LDPC Codes in MLC NAND Flash Memory, 10th International Conference on Information, Communications and Signal Processing, 1 (2015).

[11] Ngatched T. M. N., Alfa A. S. and Jun Cai, Near Optimum Majority-Logic Based Decoding of Low-Density Parity-Check Codes, IEEE Intentional Conference on Communications, 1 (2011).

[12] Haiqiang Chen, Lingshan Luo and Youming Sun, Communications and Networks **17**, 339 (2015).

[13] Youming Sun, Haiqiang Chen and Xiangcheng Li, Communications and Networks **17**, 213 (2015).

[14] Cat. J. M. and Gar. F. Valls, Communication Letters **18**, 2065 (2014).

[15] L Lan, L Zeng, YY Tai and L Chen, IEEE Transactions on Information Theory **53**, 2429 (2007).