A Modified Weighted Bit-Flipping Algorithm for LDPC Codes

Haigang Zhang†, Dongfeng Yuan†, and Cheng-Xiang Wang††

†School of Information Science and Engineering, Shandong University, Jinan, Shandong, 250100, P. R. China.
E-mail: hgzhang@sdu.edu.cn dfyuan@sdu.edu.cn
†† Joint Research Institute for Signal and Image Processing, School of Engineering & Physical Sciences, Heriot-Watt University.
Edinburgh EH14 4AS, UK. E-mail: cheng-xiang.wang@hw.ac.uk

Keywords: LDPC codes, SPA, WBF algorithm

Abstract

In this paper, a modified weighted bit-flipping (WBF) algorithm is proposed for finite geometry Low-Density Parity-Check (FG-LDPC) codes. The calculation for the flipping function in the new method includes all the elements in sum product algorithm (SPA) instead of the minimum one in most existing methods. Simulation results show that the proposed method exhibits better performance than the best method available in the literature so far. The performances of different algorithms are also considered in random constructed LDPC codes. The reliability ratio based WBF (RRWBF) method performs the best in this kind of codes and the proposed method performs very closely to the RRWBF method. The decoding complexities for both FG-LDPC and random constructed LDPC are also analyzed.

1 Introduction

Low-Density Parity-Check (LDPC) codes are one class of codes which exhibit near Shannon limit performance. It was first introduced by Gallager in his thesis in 1960' [1] and rediscovered by D. J. C. Mackay [2] after the debut of Turbo codes. Both Turbo codes and LDPC codes utilize the soft-in soft-out (SISO) iterative decoding, which is an important reason for their Shannon limit approaching ability. But the high complexity of SISO methods restricts their implementations. For LDPC codes, soft decoding can be realized by iterative decoding based on the SPA or belief propagation (BP) algorithm. Gallager also proposed the so-called bit-flipping (BF) method based on hard decision with lower complexity [1] but at the expense of large performance degradation. To bridge the performance gap between BF and SPA decoding, different WBF methods were proposed [3]-[9], which have a good tradeoff between the performance and complexity. The WBF method is very effective for finite geometry codes which usually have relatively large column or row weight. The WBF methods in [7] and [8] perform the best so far.

The WBF method was first proposed in [3] together with the construction of finite geometry codes. In [3] the minimum magnitude of the received symbols in a check sum was used as the reliability of the check sum and further be used to calculate the reliability of bit. A modified version was proposed in [4] in which intrinsic message was added on check constraint messages. A different bit metric using both the minimum and maximum magnitudes of the symbols in the check sum was presented in [5]. In [6], an improved method was given through adjusting the weighting factors of different check sums. A method called RRWBF was given in [7] and its implementation-efficient version was shown in [8]. The RRWBF method is effective for LDPC codes with smaller row or column weight. But for finite geometry codes the method in [9] performs better than all the WBF methods mentioned above. The improved modified WBF (IMWBF) method in [9] modifies the method in [4] and constructs the connection between the WBF method and the normalized uniformly most powerful (UMP) BP algorithm [10], [11]. Also, it derives the adjusting factor theoretically instead of just by simulations.

The contributions of the IMWBF method lie in two aspects. First, it excludes the intrinsic information of the symbol from check reliability which receives the reliability value. Second, it adopts the method in [11] for the normalized UMP BP algorithm to get the weighting factor in WBF decoding. The weighting factor is used to adjust the simplified result for approaching the real result in SPA decoding. But the result in SPA decoding includes all the symbols in a check sum except the symbol receiving the result, while the simplified result is only based on the symbol with the minimum magnitude. The aim of this paper is to propose a novel method, which utilizes a linear combination of all the elements in the operation of SPA decoding to approach the real result. The coefficients in the combination can be obtained in terms of the minimum mean square error between the linear combination result and real result. The novel method is extended from the IMWBF method. Simulation results show that the new method achieves better performance than both IMWBF and RRWBF methods in PG-LDPC codes. Also, the novel method is applied to random constructed LDPC codes which have relatively low row or column weight. Simulation results show that the RRWBF method performs best in this kind of codes. The proposed method has worse performance than the RRWBF method but better than the IMWBF method. The decoding complexity is considered through average decoding iteration numbers and some conclusions will be given. The idea in the proposed WBF method can also be adapted to BP based decoding method [12].

The paper is organized as follows. In Section 2, the RRWBF and IMWBF algorithms are reviewed. The novel scheme is introduced in detail in Section 3. Simulation results and decoding complexity analysis for PG-LDPC codes and random constructed codes are given in Section 4. Finally, Section 5 concludes the paper.

2 RRWBF and IMWBF methods

For finite geometry LDPC codes, the IMWBF method performs the best. The RRWBF method is also a good WBF method. However, there are only a few researches comparing the performance of IMWBF and RRWBF methods. In this paper, we will use the two methods as benchmarking for our new scheme. Some other WBF methods can be found in [3]-[6]. In this section we only outline the principles of RRWBF and IMWBF methods.

H=$\left[H_{mn}\right]$ denotes the parity check matrix which has ρ 1s in each row and γ 1s in each column. The matrix usually is square for finite geometry codes. The set of bits participating in check m is denoted by $N(m)=\{n : H_{mn}=1\}$ while the set of checks that bit n participates in are denoted by $M(n)=\{m : H_{mn}=1\}$. Also, we denote M(n)\m and N(m)\n as the set M(n) with check m excluded and the set N(m) with bit n excluded, respectively. The codeword c= $\left(c_1, c_2, \cdots, c_N\right)$ is BPSK modulated to a transmit a sequence $\left(x_1, x_2, \cdots, x_N\right)$. The received sequence through an additive white Gaussian noise (AWGN) channel is denoted by y= $\left(y_1, y_2, \cdots, y_N\right)$ with $y_n=x_n+w_n$, where w_n represents a Gaussian random variable with zero mean and variance $N_0/2$. The hard decision result is deduced by y=$\left(y_1, y_2, \cdots, y_N\right)$.

The RRWBF method gives a new quantity from a check sum for calculating the weight for a symbol [7]:

$$R_{mn}=\beta \frac{\left|y_n\right|}{\left|y_m^{\max}\right|}. \tag{1}$$

Here, y_n represents the symbol which receives the quantity, $y_m^{\max}$ denotes the symbol which has the maximum magnitude in the check, and β is a normalization factor for ensuring

$$\sum_{n \in N(m)} R_{mn}=1. \tag{2}$$

Then the error term [7] or weight in each iteration for each symbol is given by

$$E_n=\sum_{m \in M(n)}\left(2 s_m-1\right) / R_{mn}. \tag{3}$$

In (3), s_m is the syndrome bit for check m. The bit with the largest E_n will be flipped in each decoding iteration. In [8] a more implementation-efficient form for the RRWBF method was presented where

$$E_n=\frac{1}{\left|y_n\right|} \sum_{m \in M(n)}\left(2 s_m-1\right) T_m. \tag{4}$$

Here,

$$T_m=\sum_{n \in N(m)}\left|y_n\right| \tag{5}$$

represents the sum for all the magnitudes of symbols in check m.

In the IMWBF method [9] the reliability value from check m to its member bit n is termed as

$$w_{n,m}^{'}=\min_{i \in N(m) \backslash n}\left|y_i\right|, m \in[1, M], n \in N(m). \tag{6}$$

The weight for each bit in k iteration is calculated by

$$e_{IMWBF,n}^k=\sum_{m \in M(n)}\left(2 s_m^k-1\right) w_{n,m}^{'}-\alpha\left|y_n\right|. \tag{7}$$

It also flips the largest weighting bit. The factor α in (7) is very important for performance. It is first obtained by simulation in [4]. In [9], (7) is rewritten as

$$e_{IMWBF,n}^k=\frac{1}{\alpha} \sum_{m \in M(n)}\left(2 s_m^k-1\right)\left|L_2\right|-\left|2 y_n / \sigma^2\right|. \tag{8}$$

L_2 is the result of horizontal step in the UMP BP based algorithm [10] and

$$\left|L_2\right|=\min_{i \in N(m) \backslash n}\left|2 y_i / \sigma^2\right|=2 w_{n,m}^{'} / \sigma^2. \tag{9}$$

Then the question is very similar as that in the simplified BP algorithm. The factor α is used to adjusting $\left|L_2\right|$ to approach L_1 in BP or SPA.

$$\left|L_1\right|=\left|\ln \frac{1-T}{1+T}\right|, \quad T=\prod_{i \in N(m) \backslash n} \frac{1-\exp\left(2 y_i / \sigma^2\right)}{1+\exp\left(2 y_i / \sigma^2\right)} \tag{10}$$

Just like in [11], the IMWBF method gives the factor α under the constraint of the minimum mean square error between $\frac{\left|L_2\right|}{\alpha}$ and $\left|L_1\right|$, i.e.,

$$\alpha=\frac{\gamma}{\rho} \times \frac{E\left(\left|L_1 L_2\right|\right)}{E\left(\left|L_1\right|^2\right)}. \tag{11}$$

3 The novel WBF algorithm

$S=\left\{s_1, s_2, \cdots, s_{\rho-1}\right\}=\left\{\left(4 / N_0\right) y_{n^{'}},\left|n^{'} \in N(m) \backslash n\right.\right\}$ is defined in which $s_i \leq s_j$ for $i<j$. We focus on the calculation of factor α in (7). It is obvious that the IMWBF method in [9] only uses s_1 to approach L_1 from (11). But L_1 is related to all the elements in S. Therefore, an improved algorithm is proposed by defining the following variable:

$$L_3=\alpha_1 s_1+\alpha_2 s_2+\cdots+\alpha_k s_k \tag{12}$$

where $k=\rho-1$. The k coefficients are calculated by minimizing the mean square error $E\left[\left(\left|L_3\right|-\left|L_1\right|\right)^2\right]$.

Let
$$\frac{d E\left[\left(\left|L_3\right|-\left|L_1\right|\right)^2\right]}{d \alpha_1}=0. \tag{13}$$

$$E\left[s_1^2\right] \alpha_1+E\left[s_1 s_2\right] \alpha_2+\cdots+E\left[s_1 s_k\right] \alpha_k=E\left[\left|s_1 L_1\right|\right] \tag{14}$$

A group of equations can be easily derived calculating the differential coefficient for each coefficient:

$$\begin{vmatrix} E[s_1^2] & E[s_1 s_2] & \cdots & E[s_1 s_k] \\ E[s_2 s_1] & E[s_2^2] & \cdots & E[s_2 s_k] \\ \vdots & \vdots & \vdots & \vdots \\ E[s_k s_1] & E[s_k s_2] & \cdots & E[s_k^2] \end{vmatrix} \cdot \begin{vmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_k \end{vmatrix} = \begin{vmatrix} E[|L_1 s_1|] \\ E[|L_1 s_2|] \\ \vdots \\ E[|L_1 s_k|] \end{vmatrix} \quad (15)$$

All the elements, except the unknown coefficients, in the above equation group can be obtained through the samples in simulations.

The procedure of the proposed algorithm is summarized as follows:

Step 1: hard decision and computing the syndrome vector s. If the syndrome vector is a zero vector, finish decoding. The hard decision sequence is taken as the decoding result.

Step 2: For $m = 1,2,\cdots,M$, compute the reliability for each member bit n:

$$w_{n,m} = \alpha_1 s_1 + \alpha_2 s_2 + \cdots + \alpha_k s_k . \quad (16)$$

Step 3: For $n = 1,2,\cdots,N$, compute the bit weights:

$$e_n^k = \sum_{m \in M(n)} \left(2 s_m^k - 1\right) w_{n,m} - |y_n| . \quad (17)$$

Step 4: Flip the bit with the largest weight and update the syndrome vector. Go to step 2 or stop the decoding iteration if the maximum number has been reached.

The coefficients are dependent on the signal-to-noise ratio (SNR). We can use different coefficients at different SNRs. In this paper, for simplicity reasons, we only adopt the coefficients obtained at the SNR of 3dB and apply to all SNR values in simulations.

The above method is adapted to PG-LDPC codes. When applied into random constructed LDPC codes, the method should be changed just like in [9]. The coefficients equations group (15) is rewritten as

$$\begin{vmatrix} \eta E[s_1^2] & E[s_1 s_2] & \cdots & E[s_1 s_k] \\ E[s_2 s_1] & \eta E[s_2^2] & \cdots & E[s_2 s_k] \\ \vdots & \vdots & \vdots & \vdots \\ E[s_k s_1] & E[s_k s_2] & \cdots & \eta E[s_k^2] \end{vmatrix} \cdot \begin{vmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_k \end{vmatrix} = \begin{vmatrix} \eta E[|L_1 s_1|] \\ \eta E[|L_1 s_2|] \\ \vdots \\ \eta E[|L_1 s_k|] \end{vmatrix} \quad (18)$$

Note that $\eta = \frac{\rho}{\gamma}$ is a modification to the bit-based reliability value as in [9]. Now we can see (15) again. There is also a factor η in (15). But η equals to 1 since the row weight and column weight are usually identical for PD-LDPC codes. For random constructed codes, η is not identical to 1. For example, $\eta = \frac{10}{5} = 2$ for the (816,408) code used in later simulations.

4 Simulation results

The codes used in the simulation are (273,191) projective geometry code with $\rho = \gamma = 17$ and (816,408) code with $\rho = 10, \gamma = 5$. Note that the (816,408) code is random constructed. In Figure 1, we demonstrate the error performance simulation results for SPA, RRWBF, IMWBF algorithms and the proposed scheme on the (273,191) PG-LDPC code in AWGN channel. Both bit error rate curves and block error rate curves are given. The IMWBF method performs better than the RRWBF method. The proposed scheme is the nearest to the performance of SPA in the view of both the bit error rate and block error rate. The proposed scheme is derived from the IMWBF method. So at low SNR area, the two performance curves almost overlap. But when the SNR is larger than 2dB, the performance of new scheme outperforms that of the IMWBF method. The advantage becomes more obvious with the increasing SNR.

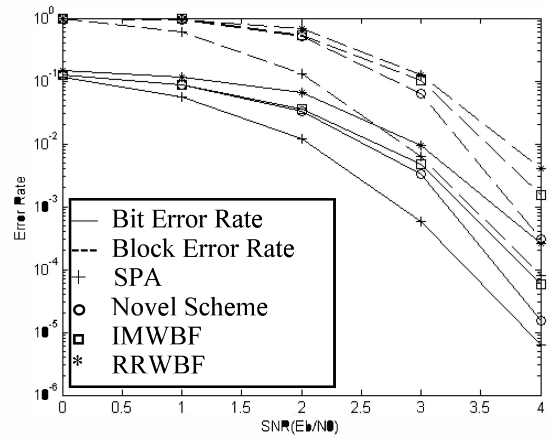

Figure 1: Bit error rate (solid line) and block error rate (dashed line) performances for different WBF methods on the (273,191) PG-LDPC code

Table 1 gives the average decoding iteration numbers for the three WBF methods on different SNRs for the (273,191) PG-LDPC code. The maximum iteration number is set at 100 in simulations. At a low SNR, the RRWBF method has the lowest iteration number. At a high SNR, the novel scheme performs the best and the IMWBF method also outperforms the RRWBF method. All the decoding numbers are lower than 9 at 4dB. At the initial decoding step, the proposed scheme needs $M * \rho * (\rho - 1)$ more multiplications and additions than the IMWBF method. The IMWBF method has N more multiplications in each decoding iteration. Considering the average decoding number difference, the total decoding complexity of the novel scheme is a little larger than the IMWBF method. However, it provides better performance.

Iteration Number	RRWBF	IMWBF	Novel Scheme
1dB	59.3	98.41	94.34
2dB	48.22	60.71	58.59
3dB	18.6386	21.585	18.315
4dB	8.5666	8.4579	8.33

Table 1: The average decoding iteration numbers for different WBF methods on different SNRs for the (273,191) code.

Figure 2 shows the performance of three WBF methods and SPA on (816,408) random constructed LDPC code. It is different from PD-LDPC code. RRWBF is nearest to SPA curve. The proposed method is close to RRWBF but a little worse. IMWBF is worst among the three WBF methods and the gap to other two methods is large. At the same time, the gap between SPA and best WBF method is much larger than the corresponding one in PG-LDPC code. So WBF method is more adapted to PG-LDPC code with larger row or column weight. For smaller row or column weight, the bit flipping has greater effect on the check sum. Through the average decoding iteration numbers below the conclusion will be more clear.

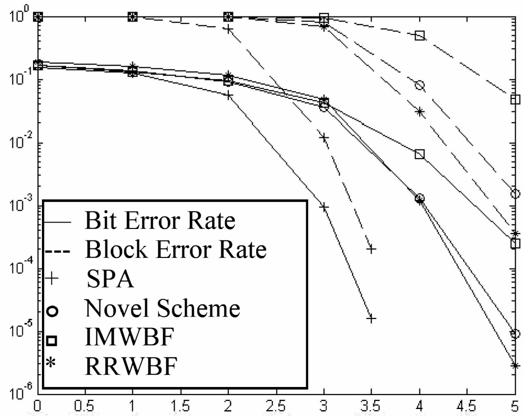

Figure 2: Bit error rate (solid line) and block error rate (dashed line) performances for different WBF methods on the (816,408) LDPC code

Iteration Number	RRWBF	IMWBF	Novel Scheme
1dB	100	100	100
2dB	100	99.73	100
3dB	92.21	98.4	94.1
4dB	52.7842	72.505	50.38
5dB	31.9226	36.0604	30.9523

Table 2: The average decoding iteration numbers for different WBF methods on different SNRs for the (816,408) code.

Table 2 gives the average decoding iteration numbers for the three WBF methods on different SNRs for the (816,408) random constructed codes. The maximum iteration number is also limited to 100. We can see that the decoding numbers decrease much slower with the increasing SNR than PG-LDPC codes. At the high SNR area, such as 5dB, the bit error performance is approaching 10^{-5}. But the decoding iteration number is still larger than 30. The proposed method is the lowest and the IMWBF method is the largest at 5dB. The RRWBF method is a little larger than proposed method.

The flipping selection is not always precise in each decoding iteration. The wrong bit flipping must be flipped back through more decoding iteration if possible. Otherwise, there will be decoding failure for the word. For lower row or column weight, the wrong bit selection occurs with larger possibility. The effect of wrong bit flipping is twofold: one is more decoding iteration number, the other is worse error performance. The two effects are both testified by the simulation results. Although more 1s in parity check matrix means more decoding operations. But from Table1 and Table 2, we can see that the decoding iteration number for random constructed LDPC code is too large compared with that for PG-LDPC code. So the WBF method is not proper for random constructed LDPC codes in the view both of performance and complexity.

5 Conclusions

In general, WBF decoding methods are effective for PG-LDPC codes since they can provide a good tradeoff between the performance and decoding complexity. However, the existing WBF methods cannot be adapted to random constructed LDPC codes with low row or column weight. For such random constructed LDPC codes, the RRWBF method performs the best but the proposed scheme also has good performance. The performance of the IMWBF method is better than that of the RRWBF method on PG-LDPC codes, while the proposed new WBF method performs the best through the simulation results. The proposed scheme achieves the smallest average decoding iteration number at the high SNR area both on random constructed and PG LDPC codes. Compared to all the existing WBF methods, our scheme has a better tradeoff between the complexity and performance even the total complexity is a little larger than the IMWBF method.

Acknowledgements

Thanks are expressed to Dr. Ming Jiang in Southeast University, Nanjing, China. This paper is partly supported by: National Scientific Foundation of China (No. 60672036), Key Project of Provincial Scientific Foundation of Shandong (No. Z2006G04).

References

[1] R. G. Gallager. "Low-Density Parity-Check Codes", Cambridge, MA: MIT Press, 1963.

[2] D. J. C. MackKay and R. M. Neal. "Near Shannon limit performance of low density parity check codes", Electronic Letters, vol. 32, no. 18, pp. 1645—1646, Aug. 1996.

[3] Y. Kou, S. Lin, and M. Fossorier. "Low Density Parity Check Codes based on Finite Geometries: A Rediscovery and New Results", IEEE Trans. on Inform. Theory, vol. 47, no. 5, pp. 2711-2736, Nov. 2001.

[4] J. Zhang and M. P. C. Fossorier. "A Modified Weighted Bit-Flipping Decoding of Low-Density Parity-Check Codes", IEEE Communications Letters, vol. 8, no. 3, pp. 165-167, Mar. 2004.

[5] Z. Liu and D. A. Pados. "Low Complexity Decoding of Finite Geometry LDPC Codes", IEEE Trans. on Communications, vol. 53, no.3, pp415-421, Mar. 2005.

[6] M. Shan, C. M. Zhao and M. Jiang. "Improved Weighted Bit-Flipping Algorithm for Decoding LDPC Codes", IEE Proc.- Commun., vol. 152, no. 6, pp.919-922, Dec. 2005.

[7] Guo F., and Hanzo, L.. "Reliability ratio based weighted bit-flipping decoding for low-density parity-check codes", Electronic Letters, 2004, 40, (21), pp. 1356-1358.

[8] C. –H. Lee and W. Wolf. "Implementation-efficient reliability ratio based weighted bit-flipping decoding for LDPC codes", Electronic Letters, Vol. 40, No.13, June 2005.

[9] Ming Jiang, Chunming Zhao, Zhihua Shi, and Yu Chen. "An Improvement on the Modified Weighted Bit Flipping Decoding Algorithm for LDPC Codes", IEEE Communications Letters, vol. 9, no. 9, pp. 814-816, Sep. 2005.

[10] J. Chen, and M. P. C. Fossorier. "Near optimum universal belief propagation based decoding of low-density parity check codes", IEEE Trans. On Commun., vol. 50, no. 3, pp. 406—414, Mar. 2002.

[11] N. Kim, and H. Park. "Modified UMP-BP decoding algorithm based on mean square error", Electronic Letters, vol. 40, pp. 816—817, Jun. 2004.

[12] Haigang Zhang, Dongfeng Yuan, Cheng-Xiang Wang, "An Improved Normalized BP Based Decoding Algorithm for LDPC Codes", to be published in Proceedings of the Second IET International Conference on Wireless, Mobile and Multimedia Networks-2008 (ICWMMN 2008), 12-15 October, 2008, Beijing China.