Reliability Ratio Based Weighted Bit-Flipping Decoding for LDPC codes

Feng Guo and Lajos Hanzo[1]

School. of ECS, University of Southampton, SO17 1BJ, UK.
Tel: +44-23-8059-3125, Fax: +44-23-8059-4508
Email:lh[1]@ecs.soton.ac.uk, http://www-mobile.ecs.soton.ac.uk

Abstract–**In this contribution, a novel reliability-ratio based weighted bit-flipping(RRWBF) algorithm is proposed for decoding Low Density Parity Check (LDPC) codes. The RRWBF algorithm proposed is benchmarked against the conventional weighted bit-flipping (WBF) algorithm [1] and the improved weighted bit-flipping (IWBF) algorithm [2]. More than 1 and 2 dB coding gain was achieved at an BER of 10^{-5} while invoking the RRWBF algorithm in comparison to the two benchmarking schemes, when communicating over an AWGN and an uncorrelated Rayleigh channel, respectively. Furthermore, the decoding complexity of the proposed RRWBF algorithm is maintained at the same level as that of the conventional WBF algorithm.**

1. INTRODUCTION

The family of Low-Density Parity-Check (LDPC) codes proposed by Gallager [3] has attracted substantial research interest in the information theory community. LDPC codes can be decoded using various decoding schemes [1–4] such as hard-decisions, soft-decisions and hybrid decoding schemes. The high-complexity Sum-Product Algorithm (SPA) was shown to achieve a near-capacity performance [4]. However, the Weighted Bit-Flipping (WBF) algorithm [1] strikes a good trade off between the associated decoding complexity and the achievable performance. The attractive property of the WBF algorithm is that during each iteration the weighted sum of the same values is computed, resulting in a significantly lower decoding complexity in comparison to the SPA. An improved WBF (IWBF) algorithm was proposed by Zhang and Fossorier [2].

An (N, K, j) LDPC code can be uniquely represented by an $M \times N$ parity check matrix (PCM), where $M = N - K$ and each column of the PCM has an average weight of j. Furthermore, each row has an average weight of k, where $j/k = M/N$. By representing the PCM using the Tanner graph [5], each column of the PCM corresponds to a message node in the Tanner graph and each row of the PCM is associated with a check node. We will use the notation **H** for representing the PCM of the LDPC code, and H_{mn} denotes the binary entry in the m_{th} row and the n_{th} column. We denote the set of bits participating in the m_{th} check by $\mathcal{N}(m) = \{n : H_{mn} = 1\}$. The term $\{n : H_{mn} = 1\}$ indicates the specific set of values for the column index n, where the value of the PCM entry H_{mn} at the m_{th} row and n_{th} column is one. Similarly, the set of checks in which the n_{th} bit participates is denoted as $\mathcal{M}(n) = \{m : H_{mn} = 1\}$.

When an information block of size K is encoded by an LDPC encoder, a codeword **c** of length N will be produced, and the coded bits will be mapped using BPSK modulation onto the corresponding constellation point **x**. When the Gaussian noise is added to the transmitted signal, a noise-contaminated received sequence **y** will be obtained. Based on the sequence **y**, an initial hard decision can be made and we arrive at a binary sequence **z** of length N.

Even though the WBF and the IWBF algorithms both strike a reasonable trade-off between the achievable coding gain and the associated decoding complexity, these two algorithms attribute the violation of a particular parity check to only the least reliable bit. However, the violation of a specific parity check could be owing to any message node participating in this check. Furthermore, the conventional WBF algorithm only considers the information provided by the check node, while the IWBF algorithm makes a judgement on the information supplied by both the check nodes and the message nodes with the aid of an optimal weighting factor α. The IWBF algorithm has a better BER performance compared to the WBF algorithm, subject to a properly chosen optimal weighting factor. However, if the factor α is not optimum, the error correction capability of the LDPC code invoking the IWBF algorithm may vary significantly. Hereby some pre-processing has to be carried out off-line for the sake of finding the optimal α. Hence the RRWBF algorithm proposed in this paper requires no pre-processing and outperforms the WBF as well as the IWBF algorithm by 1 and 2 dB at the BER of 10^{-5} when communicating over an AWGN channel and an uncorrelated Rayleigh fading channel, respectively.

2. ALGORITHM

2.1. Conventional Weighted Bit-Flipping Algorithm

The conventional WBF algorithm was proposed by Kou *et al* [1]. The decoding process is as follows:

1. The first step of the conventional WBF algorithm will find the syndrome vector **s** by multiplying the tentatively decoded bit sequence **z** with the transpose of the PCM **H**, i.e. $\mathbf{s} = \mathbf{z}\mathbf{H}^T$. If the syndrome vector **s** is an all-zero vector, the decoder will declare successful decoding and the iterations will be terminated. If not, go on to the second step.

2. During the second step, the conventional WBF algorithm will identify the most unreliable message node associated with each individual check node by computing

$$y_m^{min} = min_{\{n:n \in \mathcal{N}(m)\}} |y_n|. \tag{1}$$

3. When the magnitude of the most unreliable message node has been identified for each check node, the quantity y_m^{min} calculated in Equation 1 will be used to calculate the error-term E_n for each message node as follows:

$$E_n = \sum_{m \in \mathcal{M}(n)} (2s_m - 1)y_m^{min}, \tag{2}$$

where s_m is the syndrome bit associated with the mth check node. The variable s_m will take the value of one, if the mth check is violated, or zero otherwise.

4. When the error-terms E_n have been calculated for all message nodes, the value of the bit associated with the highest E_n value will be inverted. Afterwards, the step 1, 3 and 4 will be repeated, until a valid codeword has been found or the maximum affordable decoding complexity has been exhausted.

2.2. Improved Weighted Bit-Flipping Algorithm

As seen from Equation 2, the error-term is calculated based on only the information provided by the check node. Zhang *et al.* proposed the IWBF algorithm in [2], where the information provided by the message node is also considered. The difference between the conventional WBF algorithm and the IWBF algorithm is the third step in the iterative decoding process, where the error-term E_n is calculated. Similar to Equation 2, the error-term will be calculated as follows, when invoking the IWBF algorithm:

$$E_n = \sum_{m \in \mathcal{M}(n)} (2s_m - 1)y_m^{min} - \alpha \cdot |y_n|. \tag{3}$$

Comparing Equation 3 to Equation 2, it can be observed that there is an extra term in Equation 3, where the information provided by the message node is taken into account. The philosophy of the IWBF algorithm is that it assumes that there are two message nodes having the same error-term formulated in Equation 2, thus there are two message nodes that have an equal probability of being flipped. However, if the magnitudes $|y_n|$ of these two message nodes are different, the one having a lower magnitude is more unreliable and hence should be inverted. Therefore by using Equation 3 and incorporating the extra $-\alpha \cdot |y_n|$ multiplicative term in the evaluation of the error-term, a more accurate decision could be made.

2.3. Reliability Ratio Based Weighted Bit-Flipping Algorithm

As demonstrated in [2], the IWBF algorithm is capable of outperforming the conventional WBF algorithm. However, owing to the extra term at the end of Equation 3, an extra addition operation is needed for each calculation of the error-term E_n of

Equation 3. Furthermore, it has been demonstrated in [2] that the α term applied in Equation 3 should be carefully chosen. The performance of the IWBF algorithm depends significantly on the weighting factor α, hence α has to be pre-computed using off-line processing.

Another drawback of both the WBF algorithm and the IWBF algorithm is that they both attribute the violation of a check to the unreliable bit. However, all the message nodes connected with a check node are liable to the violation of this particular check. In other words, all message nodes might be liable to change, if the check they participate in is violated. However, for two different message nodes participating in the same violated parity check, the probability that the check is violated owing to the message node having a high soft-magnitude is lower than that associated with the message node having a low soft magnitude. Hence, hereby we would like to introduce a new quantity termed as the *Reliability Ratio* (RR) defined as follows:

$$R_{mn} = \beta \frac{|y_n|}{|y_m^{max}|}, \tag{4}$$

where the notation $|y_m^{max}|$ is used to denote the highest soft magnitude of all the message nodes participating in the m_{th} check. The variable β is a normalisation factor introduced for ensuring that we have $\sum_{n:n \in \mathcal{N}(m)} R_{mn} = 1$. Hence, instead of calculating the error-term E_n as in Equation 2 using y_m^{min}, we propose the employment of the following formula:

$$E_n = \sum_{m \in \mathcal{M}(n)} (2s_m - 1)/R_{mn}. \tag{5}$$

The rest of the RRWBF algorithm is the same as the standard WBF algorithm and the iterations will be terminated, when the resultant syndrome vector **s** becomes an all-zero vector or when the maximum number of iterations has been reached.

3. SIMULATION RESULTS

In this section the decoding performance of the RRWBF algorithm advocated will be evaluated against the WBF and IWBF benchmark algorithms. The simulation parameters are summarised in Table 1.

3.1. Performance over AWGN Channels

From Figure 1, we can observe that by using a higher average column weight, the distance properties of the LDPC code were improved, which leads to a faster convergence. We can see that by using the RRWBF algorithm, the decoding performance of LDPC codes may be improved by more than 1dB at a BER of 10^{-5}. When the column weight is increased, the RRWBF algorithm is benefiting more from the improved distance properties in comparison to the other two benchmark schemes. By using a higher column weight, the associated row weight is increased at the same time. Hence the RRWBF algorithm is able to calculate the reliability ratio of Equation 4 more accurately, since there are more entries in a row.

Modem	BPSK
Channel	AWGN
	Uncorrelated Rayleigh Fading
LDPC code	(1000,500,3)
	(1000,500,4)
	(1000,500,5)
Decoder	WBF, IWBF, RRWBF
IWBF optimal α	0.4, 0.5, 0.6
Maximum Number of Iteration	100

Table 1: Simulation parameters used for decoding three different half-rate LDPC codes all having a blocklength of 1000 and communicating over an AWGN channel using BPSK modulation. The three LDPC codes use regular PCM construction invoking an average column weight of three, four and five, respectively. The optimal α used during the IWBF decoding is chosen to be 0.4, 0.5 and 0.6 for the corresponding increased average column weight.
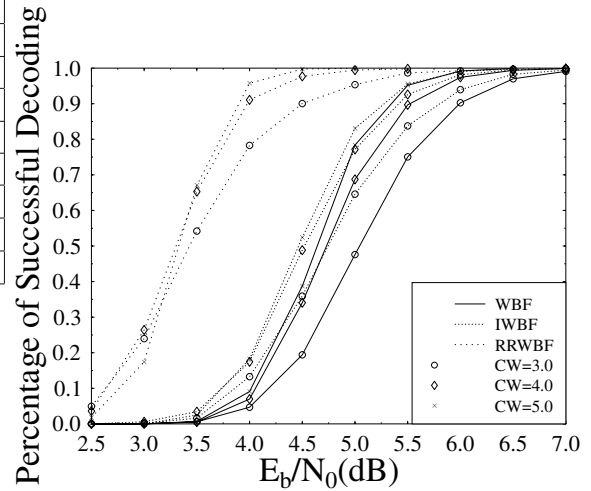

Figure 2: The percentage of successful decoding for the LDPC codes characterised in Table 1 when communicating over an AWGN channel using BPSK modulation.
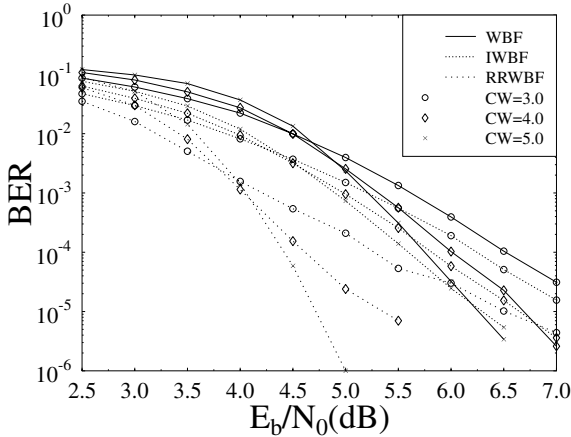
Figure 1: BER performance of the LDPC codes characterised in Table 1, when communicating over an AWGN channel using BPSK modulation.
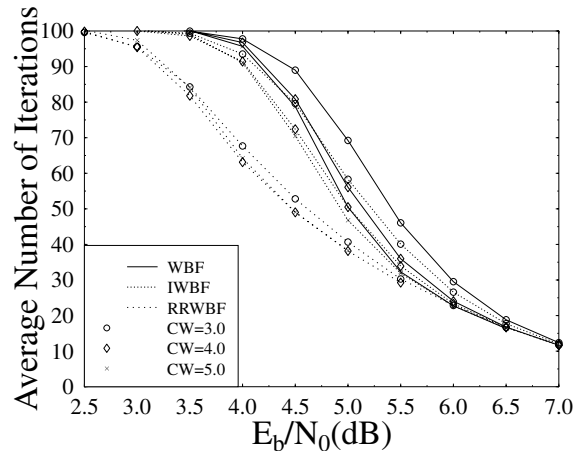

Figure 3: Average number of iterations for the LDPC codes characterised in Table 1, when communicating over an AWGN channel using BPSK modulation.

In Figure 2, we evaluate the percentage of the frames that has been successfully decoded upon invoking the three different bit-flipping algorithms. No undetected errors were recorded during this experiment, in other words, all decoded LDPC codewords were indeed the originally transmitted codewords. In Figure 2, the percentage of successful decoding while using the WBF algorithm and the IWBF algorithm is about the same and codes decoded by the IWBF algorithm are merely slightly superior. However, when the RRWBF algorithm is utilised, a significant fraction of the frames may be successfully decoded. Specifically, in the $E_b/N_0 = 3.5 - 4.5dB$ region there were about eight times more successfully decoded frames for a column-weight three LDPC code when decoded by the RRWBF algorithm, than for the same LDPC code decoded by the IWBF algorithm.

Owing to the fact that the bit-flipping algorithm will terminate its iterations, when a legitimate codeword has been detected, the average number of iterations for the LDPC codes in Table 1 was plotted in Figure 3. As we can see, in the low-SNR region the RRWBF algorithm had an average number of iterations, which was almost 20-30% lower than that of the benchmarking schemes. The average number of iterations for the three different bit-flipping schemes became similar, when the SNR was increased, where the actual number of iterations required was already low. When employing the RRWBF algorithm, we could use a lower maximum number of iterations in

comparison to the WBF and the IWBF algorithms.

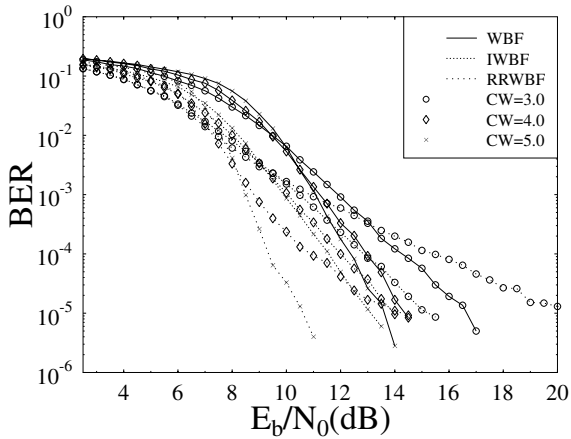## 3.2. Performance over Uncorrelated Rayleigh Fading Channels

Figure 4: BER performance of the LDPC codes characterised in Table 1, when communicating over an uncorrelated Rayleigh fading channel using BPSK modulation.
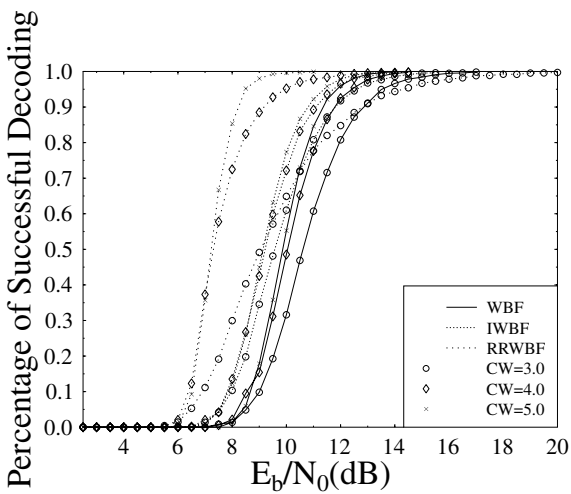

Figure 5: The percentage of successful decoding for the LDPC codes characterised in Table 1, when communicating over an uncorrelated Rayleigh fading channel using BPSK modulation.

Figures 4 to 6 demonstrated the error correction capability for the three different bit-flipping algorithms, when communicating over uncorrelated Rayleigh fading channels. Since now the channel conditions are worse than those in the AWGN channel of Section 3.1, for the column-weight three LDPC code, the performance of the RRWBF algorithm becomes inferior to that of the other two benchmarkers. However, for
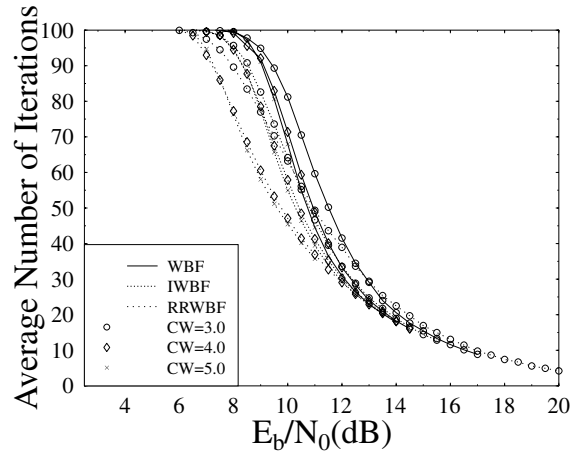

Figure 6: Average number of iterations for the LDPC codes characterised in Table 1, when communicating over an uncorrelated Rayleigh fading channel using BPSK modulation.

an average column-weight higher than three, the advantages of employing the RRWBF algorithm are still explicit. A coding gain of approximately 2dB is achieved at a BER of 10^{-5}, while utilising the RRWBF algorithm compared to the other benchmarking schemes at an average column weight of five. The reason for this phenomenon is that the RRWBF algorithm calculates the reliability ratio using k number of soft channel outputs $|y_n|$, where k denotes the average row weight. When the column weight is increased, the corresponding row weight is increased accordingly. Hence, the RRWBF algorithm is capable of calculating the reliability ratio with the aid of a higher number of values. Statistically speaking, a higher number of values will always results in a more accurate prediction. When transmitting over AWGN channels, the channel conditions were reasonably good and hence the RRWBF algorithm was capable of calculating the reliability ratio sufficiently accurately for the sake of outperforming the other two benchmark schemes. However, when an uncorrelated Rayleigh channel was encountered, the RRWBF algorithm was unable to accurately calculate the reliability ratio for a low average column weight. Hence a poorer BER performance is observed in Figure 4. However, when the average column weight is increased, the RRWBF algorithm still constitutes an attractive scheme, as demonstrated in Figures 4 to 6.

4. COMPLEXITY

The major advantage of the bit-flipping algorithm is that the associated decoding complexity is significantly lower than that of the sum-product algorithm. Hereby, we will characterise the achievable performance of the (1000,500,5) LDPC code decoded by the sum-product algorithm and the proposed RRWBF algorithm. BPSK modulation was used and the signal was

transmitted over an AWGN channel. The maximum number of iterations for the sum-product decoder was set to 10, while the RRWBF decoder used a maximum number of 100 iterations. As seen in Figure 7, the RRWBF algorithm's performance is

Figure 8: Decoding complexity comparison for the (1000,500,5) LDPC code characterised in Figure 7, when invoking the sum-product algorithm and the proposed RRWBF algorithm.

5. SUMMARY

In this contribution, a novel reliability-ratio based weighted bit-flipping algorithm was proposed. This algorithm is capable of maintaining the same decoding complexity as the conventional weighted bit-flipping algorithm, while requiring no off-line pre-processing as needed by the IWBF algorithm. The RRWBF algorithm outperforms the benchmarking schemes for all average column weights used, when communicating over AWGN channels. For the uncorrelated Rayleigh fading channel, the low-column-weight LDPC code decoded by the RRWBF algorithm was found to be inferior in comparison to the benchmarking schemes. However, when the column weight is increased, a significant coding gain of 2dB is achieved by the proposed RRWBF algorithm at a BER of 10^{-5} against the IWBF algorithm at an average column weight of five. The decoding complexity of the RRWBF algorithm was also compared to that of the soft-decision based sum-product algorithm. In Figure 8, the RRWBF algorithm was shown to require a significantly lower processing power compared to the sum-product algorithm, at the cost of 1dB coding gain loss observed at a BER of 10^{-5}.
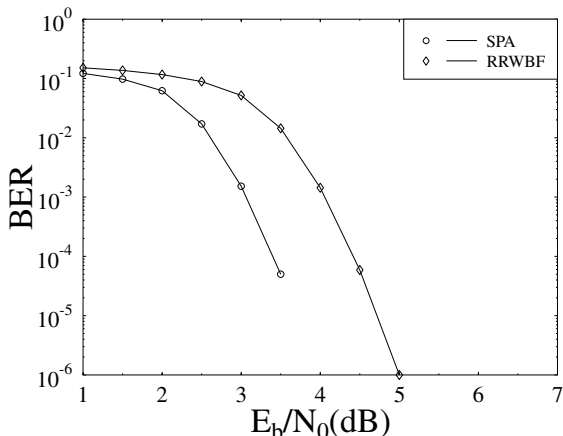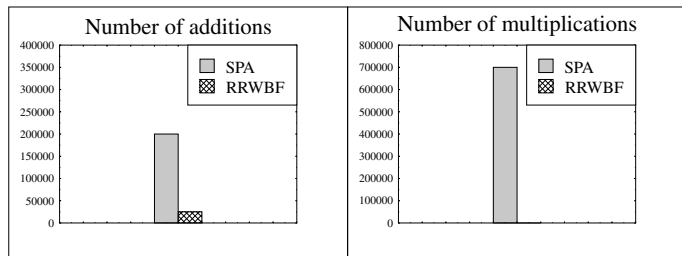

Figure 7: BER performance for the LDPC codes decoded by RRWBF algorithm and the sum-product algorithm when communicating over an AWGN channel using BPSK modulation.

inferior to that of the sum-product algorithm by about 1dB at a BER of 10^{-5}. However, we will quantify the achievable decoding complexity reduction by invoking the RRWBF algorithm, while tolerating an E_b/N_0 performance loss of 1dB.

As seen in Equation 5, during each iteration the E_n term of Equation 5 has to be updated. However, since the bit-flipping aided decoding only changes the state of a particular bit, hence there will be only j number of syndrome bits s_m that are flipped. Consequently, since every check node is associated with k number of message bits, there is an overall maximum of jk number of message nodes requiring the recalculation of the error-term. The evaluation of the error-term E_n in Equation 5 requires j number of additions, hence during each iteration, the maximum decoding complexity imposed will be $j \cdot k \cdot j$ additions. Since the average column weight was five in this experiment, thus the required decoding complexity per iteration is upper bounded by $5 \cdot 10 \cdot 5 = 250$ additions. Because a maximum of 100 iterations was used by the RRWBF decoder, the overall decoding complexity was 25000 additions.

The sum-product algorithm requires $2jq$ number of additions and $7jq/log_2(q)$ multiplications per coded bit per iteration [6], where q is the size of the decoding field, which is two in this binary scenario. For a blocklength of 1000 bits and a maximum of 10 iterations, the required number of arithmetic operations was 200,000 additions and 700,000 multiplications. The resultant complexity comparisons were shown in Figure 8.

6. REFERENCES

[1] Y. Kou, S. Lin and M. Fossorier, "Low-density parity-check codes based on finite geometries: a rediscovery and new results," *IEEE Transactions on Information Theory*, vol. 47, pp. 2711–2736, November 2001.

[2] J. Zhang, M. P. C. Fossorier, "A Modified Weighted Bit-Flipping Decoding of Low-Density Parity-Check Codes," *IEEE Communications Letters*, vol. 8, pp. 165–167, March 2004.

[3] R. Gallager, "Low Density Parity Check Codes," *IEEE Transaction on Information Theory*, vol. 8, pp. 21–28, Jan. 1962.

[4] D. J. C MacKay, and R. M. Neal, "Near Shannon Limit Performance of Low Density Parity Check Codes," *Electronics Letters*, vol. 33, pp. 457–458, 13 March 1997.

[5] M. R. Tanner, "A Recursive Approach to Low Complexity Codes," *IEEE Transactions on Information Theory*, vol. 27, September 1981.

[6] F. Guo, L. Hanzo, "Low complexity symbol-based iterative detection of serially concatenated non-binary LDPC codes and space-time codes," in *Accepted by VTC 2004 Fall*, 2004.