Journal of Electrical Engineering, Electronics, Control and Computer Science
JEEECCS, Volume 3, Issue 7, pages 13-20, 2017

Combined Unequal Error Protection and Optimized Scaling for IEEE 802.11n Low Density Parity Check Codes

Madhavsingh Indoonundon
Dept of Electrical and Electronics Engineering
University of Mauritius
Reduit, Mauritius
madhavsingh.indoonundon@umail.uom.ac.mu

Tulsi Pawan Fowdur
Dept of Electrical and Electronics Engineering
University of Mauritius
Reduit, Mauritius
p.fowdur@uom.ac.mu

***Abstract* – Low Density Parity Check codes are among the most successful forward error correction codes which are capable of providing near channel capacity performance. However, several new schemes are still being developed to improve their performance and reduce their complexity. In this paper, the performances of the IEEE 802.11n Low Density Parity Check codes are evaluated by combining three techniques: Unequal Error Protection, Optimized Scaling Factor and Failed Check Node. Unequal Error Protection is employed by mapping systematic bits onto prioritized constellation points in 16-QAM and 64-QAM constellations. Optimized Scaling Factor is performed by obtaining scaling factors for each E_b/N_0 values which are then used in both the check node and bit node update steps in the Min-Sum decoding algorithm. Finally, Failed Check Node is incorporated to select the best decoded sequence among the different sequences obtained at each iterations. Simulation results showed that maximum gains of 0.9 dB and 1.35 dB could be achieved as compared to conventional Low Density Parity Check codes decoding with 16-QAM and 64-QAM respectively in the range BER≤10^{-2}.**

Keywords-component; LDPC; UEP; OSF; FCN; IEEE; 802.11n

I. INTRODUCTION

Low density parity check (LDPC) codes are forward error-correction codes which were originally introduced by Gallager in his doctoral dissertation in 1960 [1]. However, due to their computational complexity which was too high at that time, LDPC codes were ignored. Later, in 1996, LDPC codes were rediscovered by MacKay and Neal and since then, LDPC codes have become subject to various new researches. Since LDPC codes can achieve near Shannon limit performances [2], they are considered to be one of the most powerful classes of error correcting codes developed to date. Consequently, several communications standards such as WiMax [3], DVB-T2 [4] and IEEE 802.11n [5] have adopted LDPC codes. The 802.11n standard combines QAM with LDPC codes and uses several code lengths ranging from 648 to 1944, with code rates of 1/2, 2/3, 3/4 and 5/6 [[5],[6]]. Several new techniques such as Unequal Error Protection (UEP), Optimized Scaling Factor (OSF) and Failed Check Nodes (FCN) have been developed to enhance the performance of LDPC codes. An overview of these techniques is given next.

The QAM constellation has an interesting characteristic that allows UEP to be performed as demonstrated by the bit-reordering scheme proposed in [7]. The authors [7] combined LTE Turbo codes with QAM and UEP was used to provide greater protection to the systematic bits. Consequently, significant performance gains were obtained [7]. In [8], the authors extended the work of [7] with joint source channel decoding for LTE Turbo codes. The same UEP principle was applied to IEEE 802.11n LDPC codes along with a modified hybrid ARQ scheme in [9]. An interesting scheme in [10] performed UEP by mapping the more important bits of an image to the variable nodes with higher degrees in irregular LDPC codes. After LDPC encoding the systematic bits were mapped onto a power efficient QAM constellation and the parity check bits onto a spectrally efficient 16-QAM constellation. The scheme provided an SNR gain of 1.2 dB in the range $10^{-1} \leq BER \leq 10^{-5}$. In [11], a FCN scheme for LDPC codes was proposed to reduce the error floor phenomenon but the scheme also slightly increased the decoder's complexity. An OSF scheme was used with MSA decoding in [12]. This scheme outperformed the conventional MSA decoder by 1.2 dB and the Normalized MSA proposed in [13] by 0.2 dB.

II. TRANSMITTER AND RECEIVER SYSTEMS

The complete transmission system is shown in Fig. 1.

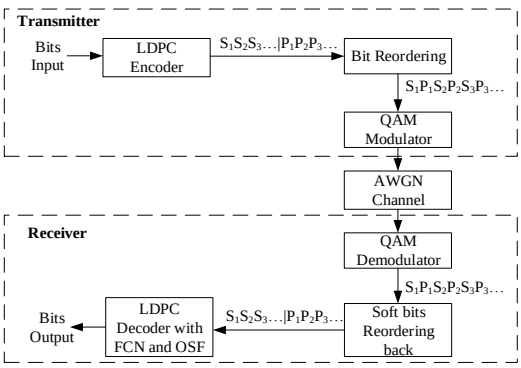

Figure 1. Complete transmission system.

A random sequence of N_s message bits is generated and fed to the LDPC Encoder. An LDPC encoded code-word c for a binary message u can be obtained using the generator matrix G in the following matrix equation [14]:

$$c = u.G \quad (1)$$

In this work, the generator matrix used is that of the IEEE 802.11n LDPC codes [5].

The bits in the LDPC code-word are then reordered so that the systematic bits are mapped onto prioritized constellations points to give them better protection than the parity bits in order to improve the overall error performance [7].

For any code-rate used, if the number of systematic bits and parity bits are N_s and N_p respectively, the last N_p systematic bits in the LDPC code-word are reordered with all the parity bits such that the former are placed onto prioritized bit positions in each QAM symbols. For example, if a code-word of code-rate 3/4 and code-length 648 bits is used, the last 162 systematic bits are reordered with the 162 parity bits. The remaining N_s - N_p systematic bits are not included in the reordering process. After reordering, the reordered sequence is multiplexed with the N_s - N_p unused systematic bits and then sent to the QAM modulator. The idea is illustrated in Fig. 2 and the reordering process is explained next.

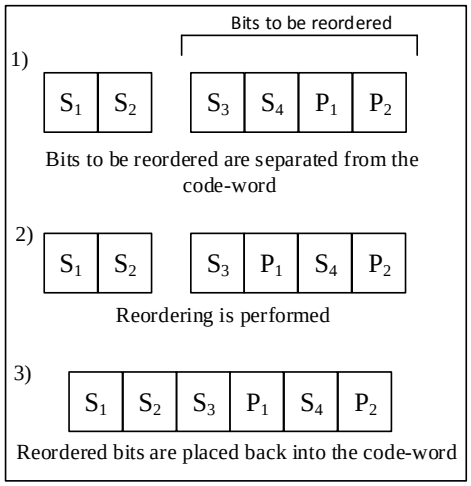

Figure 2. Bit reordering.

It is observed from the IEEE 802.11n 16-QAM constellation, in Fig. 3, that in each quadrant, the 1st and 3rd bits are the same for all the four points of the quadrant. For example in the upper right quadrant the 1st and 3rd bits are 11 for all the four points. Hence, the last N_p systematic bits are reordered with the parity bits such that the former are placed on the 1st and 3rd bit positions in each QAM symbol whereas the latter are placed on the 2nd and 4th bit positions. If the receiver correctly detects the quadrant of the received 16-QAM symbol, these systematic bits will always be correctly detected, hence leading to an improved performance [8], [9].

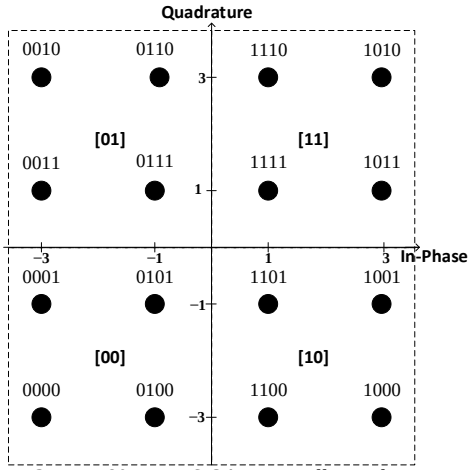

Figure 3. IEEE 802.11n 16-QAM constellation diagram.

When 64-QAM is used, the systematic bits are placed at the 1st, 2nd and 4th bit positions in each QAM symbol. Parity bits are placed in 3rd, 5th and 6th bit position in the QAM symbols. These bit orders are based on the 64-QAM constellation of the IEEE 802.11n standard as shown in Fig. 4. This constellation has four major quadrants. In each quadrant the 1st and 4th bits are same for all the 16 points. For example in the upper right quadrant the 1st and 4th bits are 11 for all the 16 points. Hence, by placing two systematic bits at the 1st and 4th positions, if the receiver correctly detects the major quadrant of the received symbol, these two systematic bits will be correctly detected. Each major quadrant is subdivided into four minor quadrants. In each minor quadrant the 2nd and 5th bits are common for all the four points found in it. For example in the upper right minor quadrant the 2nd and 5th are 00 for all the four points. Hence if the receiver correctly detects the minor quadrant of the received symbol, these 2nd and 5th bits will be correctly detected.

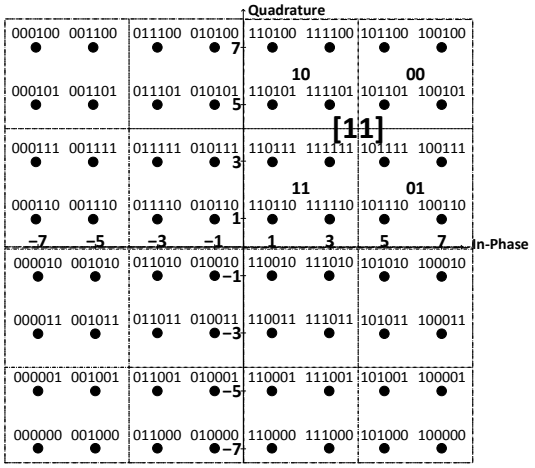

Figure 4. IEEE 802.11n 64-QAM constellation diagram.

After performing bit reordering, the reordered bits are first modulated using either IEEE 802.11n 16-QAM or 64-QAM and are then transmitted over the AWGN channel.

At the receiver's end, the received QAM symbols are demodulated to obtain soft bits. The soft bits are reordered back into their original positions and then sent to the LDPC decoder.

Binary LDPC decoding is then performed by the LDPC Decoder block in Fig. 1, which uses the Min-Sum Algorithm (MSA) with OSF and FCN, as summarized as follows [14]:

Step 1

The a-priori bit probabilities are expressed in terms of log-likelihood ratios (LLRs). For the AWGN channel, the LLR of the received noisy vector y is approximated as follows:

$$LLR = -y \tag{7}$$

Step 2

A matrix M of same size as the parity check matrix, H, is initialized such that it contains the LLR values at all the positions where there is a 1 in H. Matrices A and B are obtained from matrix H such that A stores the position of each bit node connected to each check node and B stores the position of each check node connected to each bit node.

Step 3

Matrix E which contains the extrinsic message is computed by computing each value in its j^{th} row and i^{th} column as follows:

$$E_{j,i} = \left(\alpha \prod_{i' \in B_j, i' \neq i} sign(M_{j,i'}) \right) Min|M_{j,i'}| \tag{8}$$

Where,

α is the optimized scaling factor.

Step 4

The total LLR of the i^{th} bit, L_i, is computed. It is the sum of input a priori LLR, r_i, and the LLRs from every check nodes connected to the bit as follows:

$$L_i = r_i + \sum_{j \in A_i} E_{j,i} \tag{9}$$

Step 5

The code-word is finally decoded by using the sign of the total LLR of each bits (hard decision). A positive LLR implies a bit being decoded as a 0 while a negative LLR implies a bit being decoded as a 1.

To check whether all the parity-check constraints have been satisfied, the syndrome, *s*, is calculated:

$$s = Hz^T \tag{10}$$

Where,

z^T is the transpose of the vector of the decoded LDPC code-word.

If the syndrome is zero, it implies that all the parity check constraints are satisfied and the decoding process is stopped. Else the decoder proceeds with the following step.

Step 6

Matrix M is updated with the message received by the check nodes from the bit nodes, using the optimized scaling factor as follows:

$$M_{j,i} = L_i - \alpha E_{j,i} \tag{11}$$

The decoder then proceeds with the next iteration which begins with the calculation of the extrinsic message at Step 3 using the updated matrix M.

Basically, in the OSF scheme, the MSA was modified such that a scaling factor α is used in the check node and the bit node information update steps as shown in equations 8 and 11 [12].

The values of α for the OSF scheme need to be experimentally determined by performing the decoding process with different values of α, in the range of 0-1, for different E_b/N_0 values with random LDPC code-words [12]. For each value of E_b/N_0, the value of α providing the minimum BER is selected and stored in a lookup table. Finally, when decoding needs to be performed on a code-word, the lookup table is referred to for obtaining the value of α corresponding to the E_b/N_0 value of the signal which needs to be decoded. The FCN scheme is explained next.

Check nodes are able to detect odd number of errors in variable nodes connected to them. Essentially, if a check node value of "1" is obtained in the syndrome test, this implies that there are errors in the variable nodes connected to the check node. Such check nodes are called failed check nodes (FCNs) [11]. The authors in [11] also showed that the number of erroneous bits in a decoded LDPC code-word follows the same pattern as the number of FCNs. Hence by choosing the decoded code-word at the iteration whereby the number of FCNs is minimum, the number of erroneous bits is minimized.

To implement the FCN scheme, initially, the minimum number of FCNs, FCN_{min}, is set to be equal to the number of check nodes in the code-word, N_p. Then at the n^{th} iterations, the number of FCNs, FCN_n, is counted. FCN_{min} is updated with the value of FCN_n and the decoded code-word at that iteration, C_{FCNmin}, is stored in memory X only if FCN_n is less than FCN_m.

If the last iteration, I_{max}, is performed and the syndrome does not becomes zero, the decoder outputs the code-word stored in X. This can be summarized with equation 12 and 13:

$$FCN_{min} = \min\{FCN_1, FCN_2, \ldots, FCN_{Imax}\} \tag{12}$$

$$\overline{C} = C_{FCNmin} \tag{13}$$

Where,

$\overline{C}$ is the decoder's output.

The systematic bits are extracted from $\overline{C}$ to obtain the corrected message bits.

III. Simulation Results and Analysis

The performances of the following schemes with binary LDPC codes with 16-QAM and 64-QAM are compared:

Scheme 1: LDPC codes with FCN, UEP and OSF.

Scheme 2: LDPC codes with FCN.

Scheme 3: LDPC codes with UEP.

Scheme 4: LDPC codes with OSF.

Scheme 5: Conventional LDPC codes.

The simulations were performed using the IEEE 802.11n LDPC matrices and simulation parameters are as follows:

Number of decoding iterations, T = 20.

Channel Model: Complex AWGN.

Modulation: 16 QAM and 64 QAM.

Code-rates: R = 1/2, 2/3 and 3/4.

Code-lengths: G = 648 and 1296.

The optimized scaling factors, α, for the schemes employing OSF were experimentally determined and obtained as in Tables I-IV.

A. Simulation Results when using G= 648 with 16-QAM

The simulation results obtained for the five schemes using G=648 with 16-QAM at different code-rates are given in Fig. 5-7.

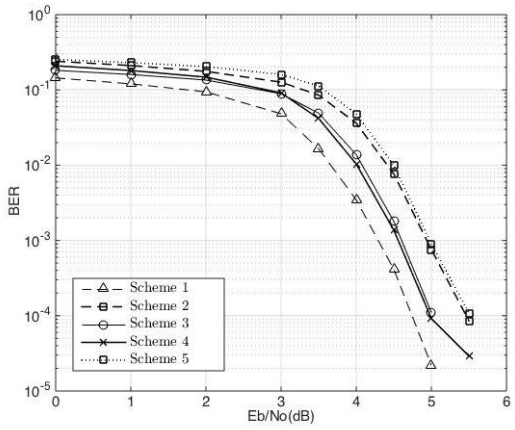

Figure 5. BER performance using 16-QAM with R=1/2 and G=648.

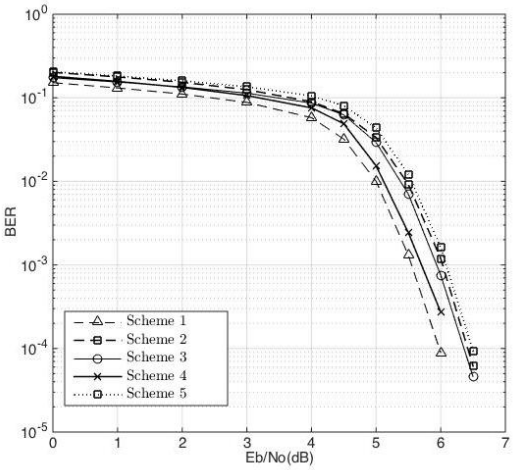

Figure 6: BER performance using 16-QAM with R=2/3 and G=648.

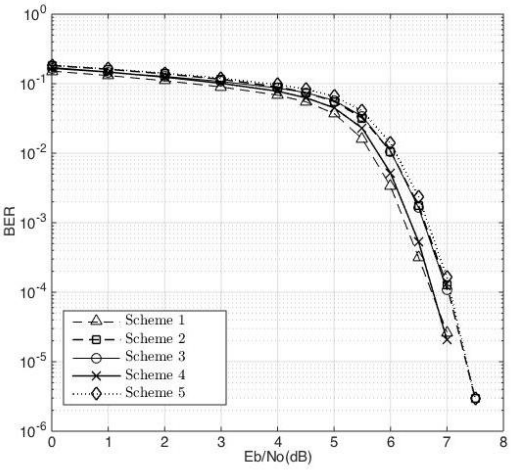

Figure 7: BER performance using 16-QAM with R=3/4 and G=648.

It can be observed from Fig. 5-7 that the proposed scheme, scheme 1, outperforms all the other schemes almost over the whole range of tested E_b/N_0 values. It gives important gains of 1.8 dB, 1.65 dB and 1.37 dB for code-rates 1/2, 2/3 and 3/4 respectively over scheme 5 at a BER of 10^{-1}. It also gives significant gains of 0.9 dB, 0.5 dB and 0.38 dB over scheme 5 for code-rates 1/2, 2/3 and 3/4 respectively in the range $10^{-2} \leq BER \leq 10^{-5}$. In the same range, scheme 2 provides the least significant gain over scheme 5 for all the tested code-rates with a maximum of 0.09 dB obtained with code-rate 3/4.

Scheme 4 was able to outperform scheme 2, 3 and 5 with every code-rate and provided E_b/N_0 gains of 0.42 dB, 0.41 dB and 0.32 dB over scheme 5 for code-rates 1/2, 2/3 and 3/4 respectively in the range $10^{-2} \leq BER \leq 10^{-5}$. Scheme 3 is found to give larger gains with code-rate 1/2 than with the other code-rates.

TABLE I. OSFS FOR E_B/N_0 VALUES USED FOR LDPC CODE-LENGTH OF 648 BITS WITH 16-QAM

		E_b/N_0											
Schemes	**Code-rate**	***0***	***1***	***2***	***3***	***3.5***	***4***	***4.5***	***5***	***5.5***	***6***	***6.5***	***7***
1	1/2	0.3	0.3	0.5	0.8	0.9	0.9	0.9	0.9	0.9	-	-	-
	2/3	0.1	0.2	0.3	0.3	0.3	0.6	0.9	0.8	0.9	0.9	0.9	-
	3/4	0.1	0.2	0.2	0.2	0.2	0.4	0.4	0.6	0.9	0.9	0.9	0.9
4	1/2	0.2	0.2	0.3	0.8	0.9	0.9	0.9	0.9	0.9	-	-	-
	2/3	0.1	0.2	0.2	0.3	0.3	0.5	0.8	0.8	0.9	0.9	0.9	-
	3/4	0.1	0.1	0.1	0.2	0.2	0.3	0.4	0.5	0.8	0.8	0.8	0.9

TABLE II. OSFS FOR E_B/N_0 VALUES USED FOR LDPC CODE-LENGTH OF 1296 BITS WITH 16-QAM

		E_b/N_0										
Schemes	**Code-rate**	***0***	***1***	***2***	***3***	***3.5***	***4***	***4.5***	***5***	***5.5***	***6***	***6.5***
1	1/2	0.2	0.3	0.4	0.9	0.9	0.9	0.9	0.9	-	-	-
	2/3	0.1	0.2	0.2	0.3	0.4	0.8	0.9	0.8	0.9	0.9	-
	3/4	0.1	0.1	0.2	0.3	0.3	0.3	0.3	0.8	0.9	0.9	0.9
4	1/2	0.2	0.3	0.4	0.9	0.9	0.9	0.8	1	-	-	-
	2/3	0.1	0.2	0.2	0.3	0.4	0.4	0.8	0.9	0.9	0.9	-
	3/4	0.1	0.1	0.2	0.2	0.3	0.3	0.4	0.5	0.8	0.9	0.9

TABLE III. OSFS FOR E_B/N_0 VALUES USED FOR LDPC CODE-LENGTH OF 648 BITS WITH 64-QAM

		E_b/N_0											
Scheme	**Code-rate**	***0***	***2***	***4***	***6***	***8***	***10***	***11***	***11.5***	***12***	***12.5***	***13***	***13.5***	***≥14***
1	1/2	0.1	0.1	0.2	0.4	0.8	0.9	0.8	0.9	0.9	0.9	-	-	-
	2/3	0.1	0.1	0.2	0.2	0.3	0.3	0.8	0.7	0.8	0.8	0.9	0.9	0.9
	3/4	0.1	0.2	0.2	0.2	0.3	0.3	0.3	0.3	0.8	0.8	0.9	0.9	0.9
4	1/2	0.1	0.2	0.2	0.3	0.9	0.9	0.9	0.9	0.9	0.9	0.8	-	-
	2/3	0.1	0.1	0.1	0.2	0.3	0.3	0.7	0.8	0.8	0.8	0.9	0.8	0.8
	3/4	0.1	0.1	0.1	0.2	0.3	0.3	0.3	0.3	0.7	0.7	0.7	0.8	0.8

TABLE IV. OSFS FOR E_B/N_0 VALUES USED FOR LDPC CODE-LENGTH OF 1296 BITS WITH 64-QAM

		E_b/N_0												
Schemes	**Code-rate**	***0***	***2***	***4***	***6***	***8***	***10***	***11***	***11.5***	***12***	***12.5***	***13***	***13.5***	***14***
1	1/2	0.1	0.1	0.3	0.4	0.7	0.9	0.9	0.9	0.9	0.9	-	-	-
	2/3	0.1	0.1	0.1	0.2	0.3	0.7	0.8	0.8	0.9	0.9	-	-	-
	3/4	0.1	0.1	0.1	0.2	0.3	0.3	0.3	0.8	0.9	0.8	0.8	0.9	0.9
4	1/2	0.1	0.1	0.2	0.3	0.8	0.3	0.9	0.9	-	-	-	-	-
	2/3	0.1	0.1	0.2	0.2	0.3	0.3	0.8	0.9	0.8	0.9	0.9	-	-
	3/4	0.1	0.1	0.1	0.2	0.2	0.3	0.3	0.3	0.7	0.7	0.7	0.7	0.9

B. Simulation Results when using G=1296 with 16-QAM

The simulation results obtained for the five schemes using G=1296 with 16-QAM at different code-rates are given in Fig. 8-10.

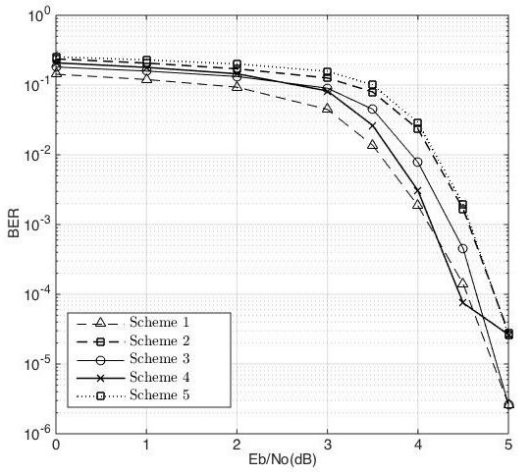

Figure 8. BER performance using 16-QAM with R=1/2 and G=1296.

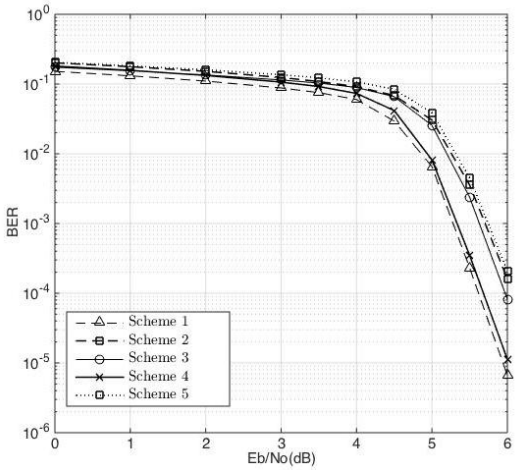

Figure 9. BER performance using 16-QAM with R=2/3 and G=1296.

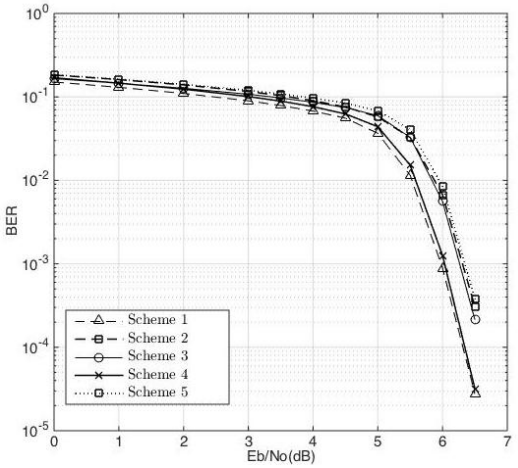

Figure 10. BER performance of schemes for uniformly distributed data transmission using 16-QAM with R=3/4 and G=1296.

The results from Fig. 8-10 show that scheme 1 provides E_b/N_0 gains of 0.5 dB, 0.44 dB and 0.38 dB over scheme 5 when using code-rates 1/2, 2/3 and 3/4 respectively over the range $10^{-2} \leq BER \leq 10^{-5}$. At a BER of 10^{-1}, scheme 1 provides gains of 2.3 dB, 1.9 dB and 1.39 dB over scheme 5 when using code-rates 1/2, 2/3 and 3/4 respectively. These results follow the same trend as when G=648. Scheme 2 provides the smallest gain and scheme 1 outperforms all the other schemes over most of the E_b/N_0 range.

C. Simulation Results when using G=648 with 64-QAM

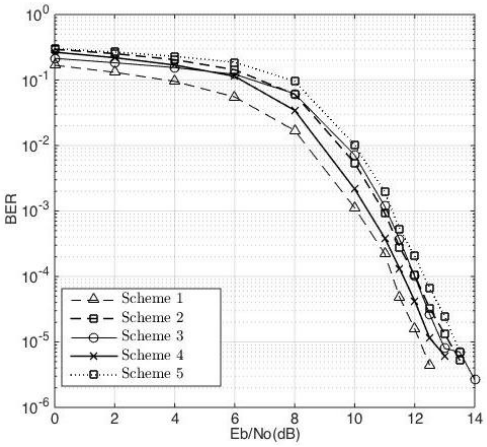

Figure 11. BER performance using 64-QAM with R=1/2 and G=648.

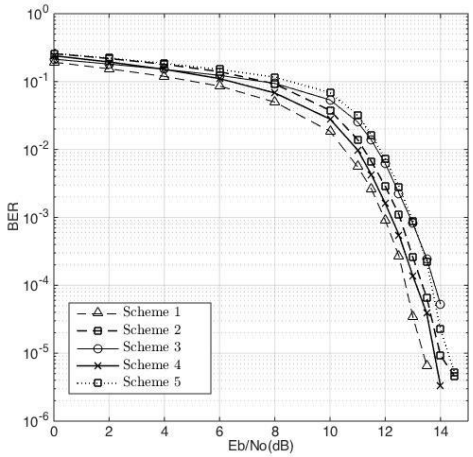

Figure 12. BER performance of schemes using 64-QAM with R=2/3 and G=648.

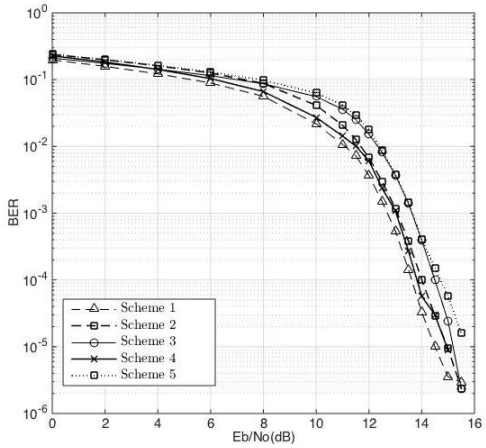

Figure 13. BER performance using 64-QAM with R=3/4 and G=648.

From Fig. 11-13, it is observed that scheme 1 provides the best overall performance. In the range of $10^{-2} \leq BER \leq 10^{-5}$, scheme 1 gives striking E_b/N_0 gains of 1.35 dB, 1.06 dB and 0.9 dB over scheme 5 for code-rates 1/2, 2/3 and 3/4 respectively. Scheme 4 outperforms schemes 2, 3 and 5 like observed with 16-QAM and scheme 3 gives the smallest gain over scheme 5 and even converges with scheme 5 at some points.

D. Simulation Results when using G=1296 with 64-QAM

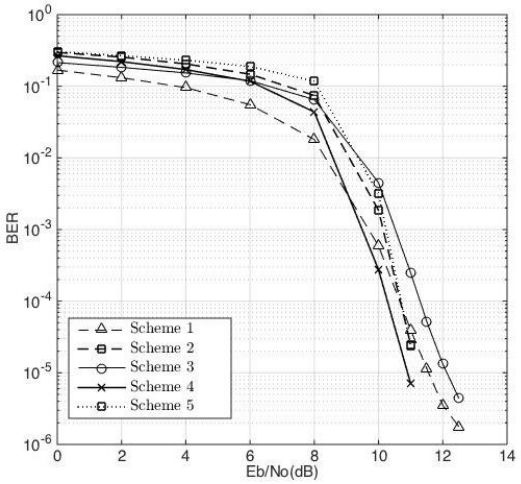

Figure 14. BER performance using 64-QAM with R=1/2 and G=1296.

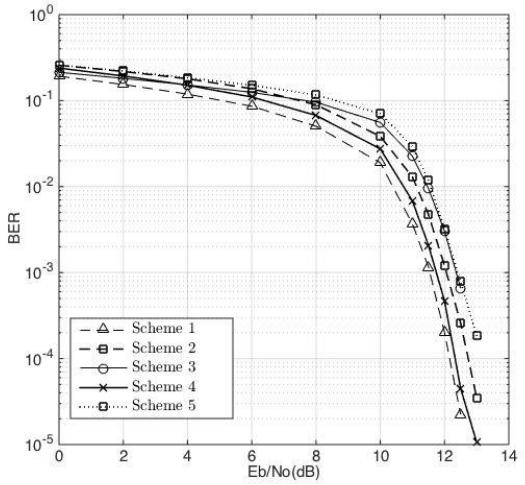

Figure 15. BER performance using 64-QAM with R=2/3 and G=1296.

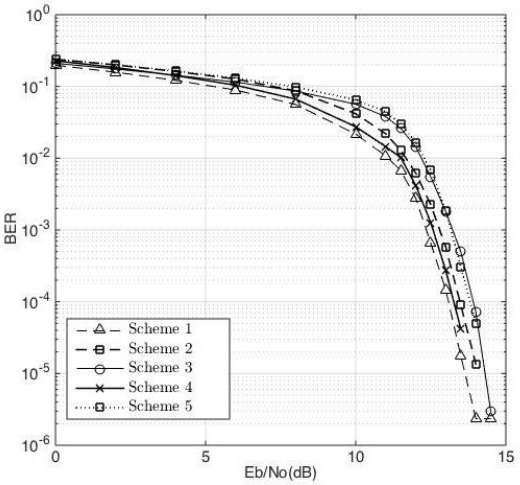

Figure 16. BER performance using 64-QAM with R=3/4 and G=1296.

With G=1296 and R=1/2, scheme 1 provides gains of 0.35 dB, 0.85 dB and 0.78 dB over scheme 5 when using code-rates 1/2, 2/3 and 3/4 respectively. Scheme 3 is found to be less effective with 64-QAM. This may be due to the fact that with 64-QAM, three priority levels are used for UEP.

IV. CONCLUSION

This paper proposed a hybrid scheme which combines the FCN, OSF and UEP techniques with the IEEE 802.11n LDPC codes. A bit reordering technique is incorporated prior to the QAM modulator with a view to provide better protection to the systematic bits. Moreover, the MSA LDPC decoder was enhanced by using a scaling factor in the check node and bit node information update steps. FCN was used to select the decoded sequence which has the least number of failed check nodes among the decoded sequences for each iterations. Simulations were performed using binary LDPC codes with 16 and 64 QAM. The proposed scheme provided the best gains when using LDPC code-rate ½. With 16-QAM, a maximum gain of 0.9 dB in E_b/N_0 was obtained for

BERs lower than 10^{-2} and with 64-QAM, a maximum gain of 1.35 dB in E_b/N_0 was obtained in the same range. An interesting future work would be to extend the performance analysis with other modulation schemes and LDPC decoding algorithms.

REFERENCES

[1] R. Gallager, "*Low-density parity-check codes*", IRE Transactions on Information Theory, vol. 8, no. 1, pp. 21-28, 1962.

[2] S.Y. Chung, "On the Design of Low-Density Parity-Check Codes within 0.0045 dB of the Shannon Limit", *IEEE Comm Letters*, vol.5, No. 2, 2001.

[3] IEEE 802.16e Task Group, "Mobile Wireless MAN", [Online] Available: http://www.ieee802.org/16/tge/

[4] DVB, 2012. Digital Video Broadcasting (DVB); "Implementation guidelines for a second generation digital terrestrial television broadcasting system (DVB-T2)", *ETSI TS 102 831 V1.2.1 (*2012-08), Available: http://www.etsi.org/deliver/etsi_ts/102800_102899/102831/01.02.01_60/ts_102831v010201p.pdf

[5] IEEE 802.11n-2009 Standard, Available from: http://standards.ieee.org/getieee802/download/802.11n-2009.pdf

[6] IEEE 802.11n, "Wireless LAN Medium Access Control and Physical Layer specifications: Enhancements for Higher Throughput", *IEEE P802.16n/D1.0*, Mar 2006.

[7] H. Lüders, A. Minwegen, and P. Vary, "Improving UMTS LTE Performance by UEP in High Order Modulation", *7th International Workshop on Multi-Carrier Systems & Solutions (MC-SS 2009)*, Herrsching, Germany, 2009, pp.185-194.

[8] T.P. Fowdur, Y. Beeharry and K.M.S Soyjaudah, "Performance of LTE Turbo Codes with Joint Source Channel Decoding, Adaptive Scaling and Prioritised QAM Constellation Mapping", *International Journal on Advances in Telecommunications*, vol 6 no 3 & 4, 2013, pp.143-152.

[9] T.P. Fowdur and B.N. Furzun, "Performance of IEEE 802.11n LDPC Codes with Modified Reliability Based Hybrid ARQ and Unequal Error Protection", *Proceedings of IEEE Eurocon 2015 Conference*, 8-11th September 2015, Salamanca, Spain.

[10] Y. Zhang, X. Li and H. Yang, "Unequal Error Protection in Image Transmission Based on LDPC Codes", *International Journal of Signal Processing, Image Processing and Pattern Recognition Vol.9*, No.3, 2016, pp.1-10.

[11] S. Yu and E. Joo, "LDPC Decoding by Failed Check Nodes for Serial Concatenated Code", *ETRI Journal*, vol. 37, no. 1, pp. 54-60, 2015.

[12] M. Rakibul, D. Siam, M. Mostafa and I. Rahman, "Optimized Min-Sum Decoding Algorithm for Low Density Parity Check Codes", *International Journal of Advanced Computer Science and Applications*, vol. 2, no. 12, 2011.

[13] J. Chen and M. Fossorier, "Near optimum universal belief propagation based decoding of low-density parity check codes", *IEEE Transactions on Communications*, vol. 50, no. 3, pp. 406-414, 2002.

[14] S. J. Johnson, "*Introducing Low Density Parity- Check Codes*", The University of Newcastle, http://sigpromu.org/sarah/SJohnsonLDPCintro.pdf [Accessed 20 December 2016].