Cross-Layer Iterative Decoding of Irregular LDPC using Cyclic Redundancy Check Codes

Zhimin Yang, Shiju Li, Hao Feng and Thomas Honold
Dept. of Information Science and Electronic Engineering, Zhejiang Univ.
ZheDa Road #38, Hangzhou, P.R.China, 310027
Email: bestyang@gmail.com, leesj@zju.edu.cn, fenghao666@126.com, honold@zju.edu.cn

***Abstract*—This paper presents a cross layer iterative decoder for irregular Low-Density Parity-Check (LDPC) coded system by using Cyclic Redundancy Check (CRC) codes. The key idea of the decoder is to use the correctly decoded frame to help correcting the left erroneous ones. To accomplish this, the decoder digs the useful information between layers by using the cross layer design method and an iterative decoding architecture. Moreover, the Unequal-Error Protection (UEP) of irregular LDPC is exploited and both multiple-error detection and single-error correction capability of CRC code are used. Simulation results show that the proposed decoder outperforms the pure Sum-Product Algorithm (SPA) decoder by a considerable gain while the complexity increase is moderate. Furthermore, the error floor of irregular LDPC in high Eb/N0 can be lowered effectively. The proposed cross layer iterative decoder can be used for any irregular LDPC coded wireless system to boost the performance and lower the error floor.**

***Index Terms*—Low-Density Parity-Check, Cyclic Redundancy Check, Iterative Decoder, Forward Error Correction, Cross-Layer Design**

I. Introduction

Low-Density Parity-Check (LDPC) codes, which are originally invented by Gallager [1] and then rediscovered by Mackay and Neal [2], are widely used in wireless systems to achieve extremely low error rate transmission in physical layer [3] [4]. Irregular LDPC achieves better performance than its regular counterpart under the same code rate and block length [5]. However, it exhibits higher error floor than that of the regular codes. So extra methods are used to reduce the error floor of irregular LDPC coded system in wireless communications, e.g. BCH codes are used in DVB-S2 to obtain low error rate in high SNR [3].

Cyclic Redundancy Check (CRC) codes—-one of the most powerful error detection codes—-are used in Data Link Layer (DLL) to detect erroneous frames. The service data, which is delivered by the physical layer in the form of frame after channel decoding, is usually fed to the CRC error detecting circuit. And the system accepts the frame if it passes the check. Otherwise, a frame retransmission is requested by the receiver.

Conventionally, the LDPC decoding in physical layer and the CRC error detecting in data link layer are performed separately. But this processing strategy does not fully exploit the information that could be used for error recovery. In contrast, cross-Layer Design (CLD) methodology, which gives up designing protocols at different layers independently, aims digging as much useful information as possible by exchanging information between layers and optimizing the parameters jointly [6].

Unlike the conventional system in which the channel decoding in physical layer and the error detecting in data link layer are independent, this paper presents a novel cross layer iterative decoding architecture for irregular LDPC coded wireless system. The demodulated signal is firstly decoded by Sum-Product Algorithm (SPA) in physical layer. The decoded bits are then delivered to the data link layer frame by frame. Later CRC processing in DLL is performed and the corrected frames are fed back to the SPA decoder and then another LDPC decoding try is turned on. The whole decoder iterates until the termination condition is met.

The cross layer iterative decoder is based on the following theoretical backgrounds:

Firstly, any code that has the minimum hamming distance $d_{min} \geq 3$ can at least correct a single bit error. And most of the standardized CRC codes have $d_{min} \geq 4$ with a wide range of block length [7]. Therefore, the CRC codes can correct single-bit error besides detecting multiple-error.

Secondly, irregular LDPC under SPA decoding exhibits Unequal Error Protection (UEP) property. According to the Gaussian approximation analysis of SPA decoder, the average bit error probability after ℓ iterations is formulated by [8, p.664]

$$P_\ell = \sum_{i=2}^{d_l} \lambda_i^{'} Q\left(\sqrt{\frac{s + i \cdot t_\ell}{2}}\right) \tag{1}$$

where $\lambda_i^{'}$ is the fraction of degree-i variable nodes; s is equivalent to the mean of the channel LLR supposing all-zero codeword is transmitted; t_ℓ is equivalent to the mean of the external LLR about the variable node; $Q(\cdot)$ is the Q function and d_l is the maximal degree of the variable node in the factor graph [9, p.619]. (1) indicates that the bit error probability of degree-i nodes is $Q\left(\sqrt{(s + i \cdot t_\ell)/2}\right)$, i.e., the bits corresponding to higher degree variable nodes are better protected by the codes.

According to the above two properties of CRC and irregular LDPC, the DLL frame that consists of higher degree variable nodes can be corrected by CRC codes with higher probability. The proposed decoder uses the correctly decoded frames to help to correct the left ones by using cross layer design and an iterative architecture.

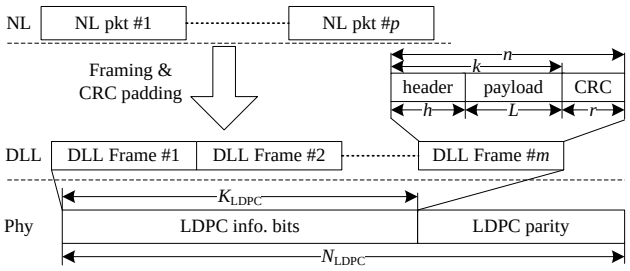


Fig. 1. Data structure used in the irregular LDPC coded system

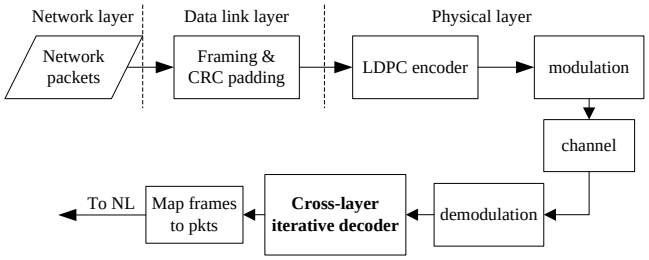


Fig. 2. End-to-End diagram of the irregular LDPC coded system

Simulation results show that the cross layer iterative decoder outperforms the conventional one (pure SPA decoder) considerably in most SNR at the expense of moderately increasing the complexity. Moreover, the error floor in high SNR is lowered markedly. The frame retransmission rate is reduced due to the reduction of frame error rate and therefore the overall bandwidth efficiency is improved by the proposed decoder. The proposed decoder does not add extra redundancy but exploits the useful information effectively.

The rest of the paper is organized as follows: in Section II, the end-to-end diagram and the data structure is firstly described. Then the keynotes of the coded system is addressed. In Section III, both the details of the proposed cross layer iterative decoder and the decoding complexity analysis are presented. And then the simulation results are shown in Section IV. Finally, discussions are carried out and some conclusion remarks are drawn in Section V.

II. System Model and Keynotes

A. End-to-End Diagram and Data Structure

Figure.1 depicts the data structure. According to the figure, n, k, h, L and r denote the code length, information length, header length, payload length and parity length of the DLL frame, respectively. The $(N_{\text{LDPC}}, K_{\text{LDPC}})$ LDPC codeword contains m DLL frames.

Figure.2 shows the end-to-end diagram of the irregular LDPC coded system. According to the seven-layer open systems interconnect (OSI) model [10, p.20], parts of the data link layer's functionality are to serve the Network Layer (NL) and deliver the structured data to the physical layer (Phy). To accomplish these, Data link layer maps the network packets to DLL frames by adding headers and appending CRC checksums. Then the resulted frames are delivered to physical layer. Total m DLL frames make up of a LDPC encoding block and the block is encoded according to the irregular parity check matrix. Before being fed into the channel, the LDPC codeword is modulated. At the receiver, the received signal is firstly demodulated and then the resulted Logarithm Likelihood Ratio (LLR) is injected to the cross layer iterative decoder. The decoder performs LDPC decoding and CRC error correcting/detecting jointly and iteratively. It outputs the decoded bits frame by frame and the resulted DLL frames are mapped to network packets.

B. Selection of CRC and DLL Frame Length

The criterion for CRC code selection is to ensure the undetected-error probability sufficiently low. And almost all of the standardized CRC codes with parity length $r \geq 16$ can meet this requirement.

Another concerned issue of the CRC code is about the optimal length of DLL frame. Suppose the error detection based on CRC is perfect and the error probability of each bit P_b within a DLL frame is equal. Let $n = h + L + r$ be the frame length (Figure.1), Given h, r, the frame error probability can be written as $P_f = 1-(1-P_b)^n$. The average bandwidth efficiency η in link layer is formulated by:

$$\eta = \frac{L}{n}(1 - P_f) \tag{2}$$

Take the partial derivative of η with respect to n:

$$\frac{\partial \eta}{\partial n} = \frac{(h+r)(1-P_f) - nL\frac{\partial P_f}{\partial n}}{n^2} \tag{3}$$

$$= \frac{1-P_f}{n^2}(h + r + nL\ln(1-P_{eb})) \tag{4}$$

$$\approx \frac{1-P_f}{n^2}(h + r - nLP_b) \tag{5}$$

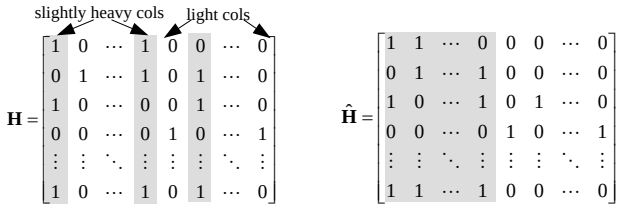
where $\frac{\partial P_f}{\partial n} = (P_f - 1)\ln(1-P_b)$. $\ln(1+x) \approx x$ for small x is used in (5). Set $\frac{\partial \eta}{\partial n} = 0$, then one can find the optimal frame length n as the function of P_b:

$$n_{opt}(P_b) = \text{round}\left(\sqrt{\left(\frac{h+r}{2}\right)^2 + \frac{h+r}{P_b}} + \frac{h+r}{2}\right) \tag{6}$$

where round(x) is the function of getting the nearest integer of x. (6) shows that n_{opt} decreases with P_b monotonously.

C. Parity Check Matrix of Irregular LDPC

Given a degree distribution pair (λ, ρ) of irregular LDPC, where λ and ρ are polynomials defined in [9, p.620], the parity check matrix (PCM) can be generated randomly or by specified method, e.g., the Progressive-Edge Growth (PEG) algorithm [11]. If the PCM obtained looks like the form in Figure.3(a), in which the columns with relative high weight are called slightly heavy col while the other columns are called light col, then we perform columns permutation according to column weight to obtain a new PCM in which the column weight is descending, i.e., it looks like the form in Figure.3(b).



(a) Random parity check matrix (b) Desired parity check matrix

Fig. 3. Original and desired parity check matrix of irregular low-density parity-check codes

The column permutation corresponds to the variable node permutation in the factor graph and does not modify the edges connectivities. It does not alert the code performance, but it makes each of the resulted DLL frame containing the variable nodes having almost the same degree. Thus, we finally obtain some elite DLL frames which consist of slightly heavy degree variable nodes. The elite frames have lower frame error probability than the others and can be used to help to correct other erroneous frames if they are correctly decoded.

III. Proposed Cross Layer Decoding Method

Figure.4 presents the architecture of the cross layer iterative decoder. It mainly consists of a CRC sub-decoder in data link layer, a SPA sub-decoder and a bit LLR updating module in physical layer. The whole decoder works as:

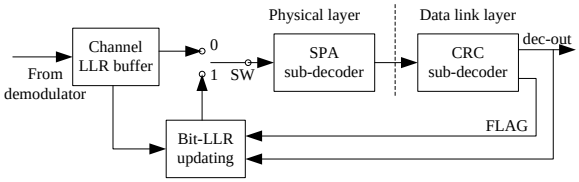


Fig. 4. Proposed corss layer iterative decoder for irregular LDPC coded system

1) Initialization: connect SW to 0;
2) SPA sub-decoder starts working as described in Section III-A, then the decoded bits is delivered to the data link layer frame by frame;
3) CRC sub-decoder starts to decode the DLL frames as described in Section III-B. Then it outputs the decoded frame and the associated FLAG, which indicates whether the frame is successfully decoded or not. After all the m frames are decoded, it performs the termination conditions checking. If any of the condition is met, go to step *5)*, otherwise go to step *4)*;
4) The bit-LLR updating module updates the channel LLR as described in Secion III-C. Connect SW to 1 and the new LLR is fed into the SPA sub-decoder, go to *step 2)* to start the next iteration;
5) Exit the decoding procedure.

A. SPA sub-decoder

The standard SPA [12] (also known as Belief Propagation (BP) algorithm) is used for irregular LDPC decoding. Although the SPA is well known, it is briefly described in order to make this paper self-contained.

Let $\hat{\mathbf{H}} = [\hat{H}_{mn}]$ be the desired PCM obtained in Section II-C. The set of check nodes connected to variable node n is denoted by $\mathcal{M}(n)$, and $\mathcal{M}(n) \backslash m$ is denoted as the set $\mathcal{M}(n)$ with check node m excluded. Similarly, the set of variable nodes that participate in the m-th parity check equation is denoted by $\mathcal{N}(m)$, and $\mathcal{N}(m) \backslash n$ is denoted as the set $\mathcal{N}(m)$ with variable node n excluded. Let $u_0(n)$ be the demodulator output LLR of the n-th symbol, then the SPA decoder works as:

initialization: for each index pair (m, n) such that $\hat{H}_{mn} = 1$, set $u_{mn} = u_0(n)$, where u_{mn} is the LLR sent from the bit node n to the check node m.

Horizontal step: for each m, n, compute

$$v_{mn} = 2 \tanh^{-1} \left(\prod_{n' \in \mathcal{N}(m) \backslash n} \left(\frac{u_{mn'}}{2} \right) \right) \tag{7}$$

where v_{mn} is the LLR of bit n which is sent from the m-th check node to the n-th bit node.

Vertical step: for each m, n, compute

$$u_{mn} = u_0(n) + \sum_{m' \in \mathcal{M}(n) \backslash m} v_{m'n} \tag{8}$$

decision step: for each n, compute the LLR of bit n

$$z_n = u_0(n) + \sum_{m \in \mathcal{M}(n)} v_{mn} \tag{9}$$

Make hard decision and obtain $\hat{\mathbf{x}} = [\hat{x}_1, \hat{x}_2, \cdots, \hat{x}_n]$ which satisfies $\hat{x}_i = 1$ if $z_i < 0$ and $\hat{x}_i = 0$ otherwise. If the parity check equation $\hat{\mathbf{H}}\hat{\mathbf{x}}^{\mathbf{T}} = \mathbf{0}$ is satisfied, then the SPA terminates. Otherwise go to the *Horizontal step* to continue the next iteration until the check equation is satisfied or the maximum number of iterations is reached.

B. CRC sub-decoder

Conventionally, CRC codes are only used for error detection [13] [14]. In fact, most of the CRC codes have the minimum hamming distance $d_{min} \geq 4$ under a wide range of block length [15]. According to the coding theory, these codes can correct at least 1-bit error. Very rare system employs the CRC for error correction except the ATM [16].

Let $g(x)$ be the generator polynomial of a CRC code C with parity length r, usually it can be decomposed by:

$$g(x) = (1 + x) g_1(x) \tag{10}$$

where $g_1(x)$ is a primitive polynomial of GF(2^M) and $M = r - 1$. [15, Theorem 4] shows that C can detect all single, double and triple errors if the code length $n \leq 2^M - 1$. Therefore, $d_{min}(C) \geq 3$ can be satisfied, which results that C can correct all single-bit error by some decoding method.

Let $V(x)$, $c(x)$ and $e(x)$ be the polynomial of received vector, codeword and error pattern, respectively. Denote $S(x)$ as the remainder of $V(x)$ divided by $g(x)$. According to $V(x) = c(x) + e(x)$, we have $S(x) = e(x) \mod g(x)$.

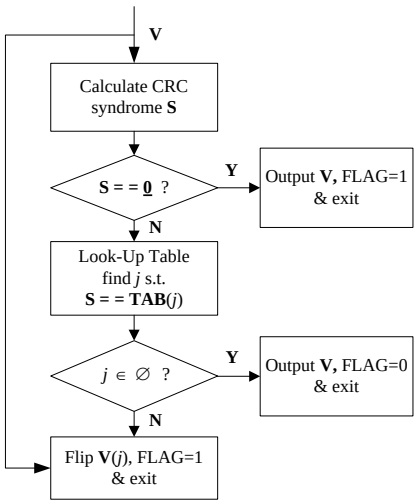


Fig. 5. CRC sub-decoder architecture

Suppose $e_1(x) = x^i$ and $e_2(x) = x^j$, where $0 \le i, j < n \le 2^M - 1$. Then

$$x^i = q_1(x)g(x) + S_1(x) \quad (11)$$

$$x^j = q_2(x)g(x) + S_2(x) \quad (12)$$

$$x^i + x^j = [q_1(x) + q_2(x)]g(x) + S_1(x) + S_2(x) \quad (13)$$

$S_1(x) = S_2(x)$ if and only if $i = j$. Because if $S_1(x) = S_2(x)$ under $i \neq j$, then (13) becomes $[x^i + x^j] \mod g(x) = 0$, which indicates C cannot detect any double errors. This conflicts with the conclusion of [15, Theorem 4]. So every $S(x)$ corresponds to an unique single-bit error pattern. The decoder produce a table **TAB** which maps S to the error location, then it can correct any single-bit error very fast by using Look-Up Table (LUT).

In order to indicate whether a frame is correctly decoded or not, the CRC decoder designate a FLAG to each frame. It is set to 1 if the associated frame is correctly decoded and 0 otherwise.

The diagram of the CRC sub-decoder is depicted in Figure.5. The decoder firstly compute the syndrome S of the input V. If $S = 0$, which indicates V is correct, it outputs V and sets FLAG=1 before exit. Otherwise, it finds the error location j by LUT. If j is empty, which indicates the error pattern contains more than one bit error, it outputs V and sets FLAG=0 before exit. If the error location j is successfully found, it flips $V(j)$ and outputs the resulted vector and sets FLAG=1 before exit. Essentially, the CRC decoder is of the coset-leader decoder [17, p.42-43] or the Minimum Distance Decoder (MDD)

Admittedly, error pattern $e(x)$ containing multiple errors may also produce the same $S(x)$ as that of the single-bit error. In this case, it cannot be corrected by CRC code and the MDD treat it as the single-bit error pattern.

C. Bit LLR updating

The bit LLR updating module updates the channel LLR according to the decoded DLL frame and its FLAG. Let $u_0^{new}(i)$ be the new LLR of the i-the bit b_i in the frame and $u_0(i)$ the original one, then the updating rule is formulated by:

$$u_0^{\text{new}}(i) = \begin{cases} (-1)^{b_i} \cdot \text{MAX}, & \text{if FLAG=1} \quad (14) \\ u_0(i), & \text{if FLAG=0} \quad (15) \end{cases}$$

where MAX is a large positive number (e.g., MAX=20). (14) reveals that the bit LLR is set to be large enough to indicate highly reliable if the DLL frame is correctly decoded. Otherwise, it is not updated according to (15).

D. Complexity Reduction Strategies

In order to reduce the computational complexity, the cross layer iterative decoder performs termination conditions checking in *step 3)*. The decoder exits if any of the following conditions is met:

i) The maximum number of iterations is reached;
ii) All of the DLL frames are correctly decoded;
iii) None of the DLL frame is corrected, i.e., FLAG=0 for all frames;
iv) No new frame is corrected in the current iteration.

The reason for condition iii) is that the bit LLR would not be updated according to (14) and (15), so there would not be any improvement if the iteration goes on.

E. Complexity Analysis

Compared with SPA, the amount of computation of CRC decoder is neglectable since it only needs XOR based logical operation. (14) and (15) show the bit LLR updating module is extremely simple, so SPA dominates the amount of computations needed. Define the average number of iterations ($\bar{\ell}$) taken by SPA to decode one LDPC block by:

$$\bar{\ell} = \frac{\#\text{total iterations taken by SPA}}{\#\text{decoded LDPC block}} \quad (16)$$

where one decoded LDPC block contains K_{LDPC} bits. $\bar{\ell}$ is used to evaluate the computational complexity of both the proposed cross layer iterative decoder and the pure SPA decoder.

IV. Numeric Results

A. Simulation Setups

Two irregular LDPC coded systems (code-A and code-B) are simulated in this paper. Their parameters are listed in Table.I. The header length h and CRC parity length r are set to 48 and 16, respectively. The CRC length are optimized for $P_b = 1 \times 10^{-3}$, which results $n_{opt} = 287$ according to (6) and finally we select $n = 300$ to ensure K_{LDPC} can be divided by n. The maximum number of iterations in SPA is set to 50 and the maximum number of iterations for the cross layer iterative decoder (denoted as **iter** in the following figures) is set to 3. The CCITT CRC-16 code, whose generator polynomial is 0x1021, is used for error detection/correction. The degree distribution pair (λ, ρ) of the custom LDPC in

TABLE I
PARAMETERS OF CODE-A AND CODE-B, THE MEANING OF EACH PARAMETER CAN BE FOUND IN FIGURE.1

Code Name	R_{LDPC}	$(N_{\text{LDPC}}, K_{\text{LDPC}})$	m	comment
code-A	$4/9$	$(16200, 7200)$	24	DVB-S2 LDPC
code-B	$2/3$	$(7200, 4800)$	16	custom LDPC

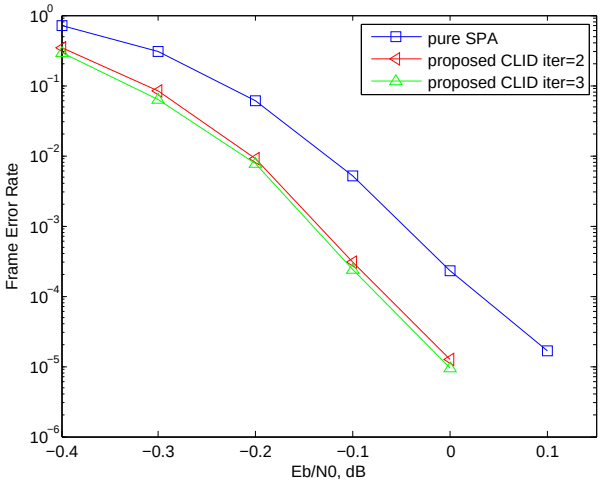


Fig. 6. Frame error rate of code-A over AWGN channel

code-B are optimized according to AWGN channel by using the online program [18]:

$$\begin{aligned}\lambda(x) &= 0.2295x + 0.3171x^2 + 0.0672x^5 + 0.0486x^6 \\ &\quad +0.0266x^7 + 0.3110x^9 \\ \rho(x) &= x^{10}\end{aligned}$$

And the PCM is generated via the PEG algorithm [11]. Note that the identified code rate of the DVB-S2 LDPC in code-A is $\frac{1}{2}$, but the real code rate is $\frac{4}{9}$. The LDPC coded bits are modulated by BPSK before being fed to the channel. AWGN channel is considered during the simulation.

B. Simulation Results and Discussions

Frame error rate (FER) is used for performance evaluation, because the frame error rate is more directly than the bit error rate to reflect the bandwidth efficiency according to (2).

Figure.6 shows the FER of code-A over AWGN channel. Compared with the pure SPA decoder, the proposed Cross-Layer Iterative Decoder (CLID) achieves an additional gain of 0.1 dB in most Eb/N0 after up to 2 iterations. The curves also reveal that the performance of CLID saturates after up to 3 iterations.

Figure.7 shows the average number of iterations (defined by (16)) taken by SPA to decode one LDPC block for code-A over AWGN channel. The curves indicate that the complexity of the CLID is adaptive. The $\bar{\ell}$ of CLID is moderate in low Eb/N0, compared with that of pure SPA. And they approaches to the same level when Eb/N0 goes high.

Figure.8 presents the FER of code-B over AWGN channel. It is shown that code-B with pure SPA decoding exhibits

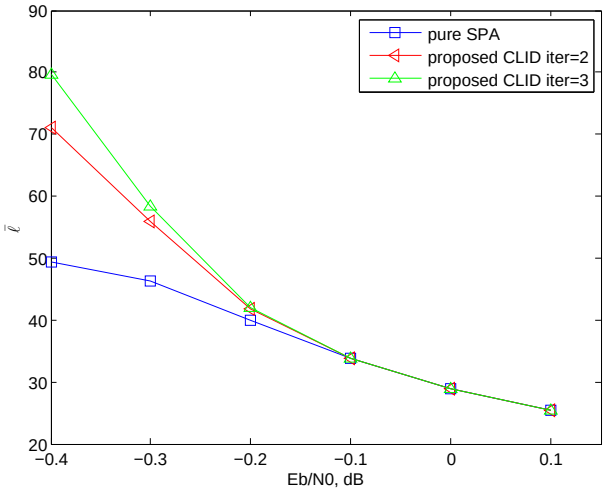


Fig. 7. Average number of iterations per decoded LDPC block of code-A over AWGN channel

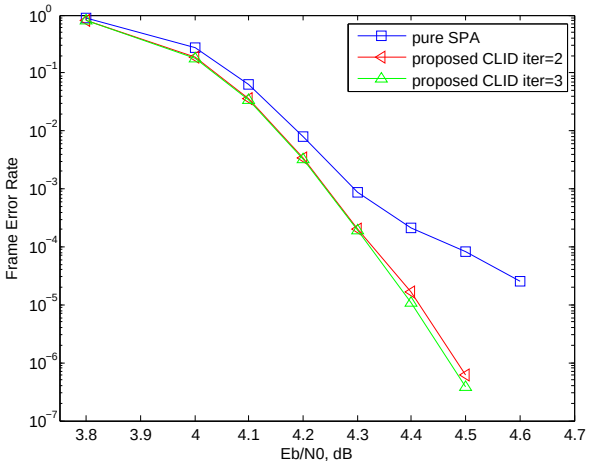


Fig. 8. Frame error rate of code-B over AWGN channel

high error floor. However, it can be lowered effectively by the CLID after up to 2 iterations and the performance gain is considerable in high Eb/N0. Figure.9 gives the average number of iterations needed to decode one LDPC block for code-B over AWGN channel. The computational complexity of the CLID after 2 and 3 iterations are moderate in low Eb/N0 and become almost the same as that of the pure SPA.

The numeric results support the advantages of the proposed cross layer iterative decoder for irregular coded LDPC system. The key idea of the CLID is to use the correctly decoded frames to help to correct the left erroneous ones by the iterative decoding architecture. And this is well done by employing the cross layer design method. The CLID does not add extra redundancy but exploits the useful information in the data link layer. Moreover, the reduction of the FER improves the system bandwidth efficiency. Furthermore, the new decoder can lower the error floor of irregular LDPC effectively.

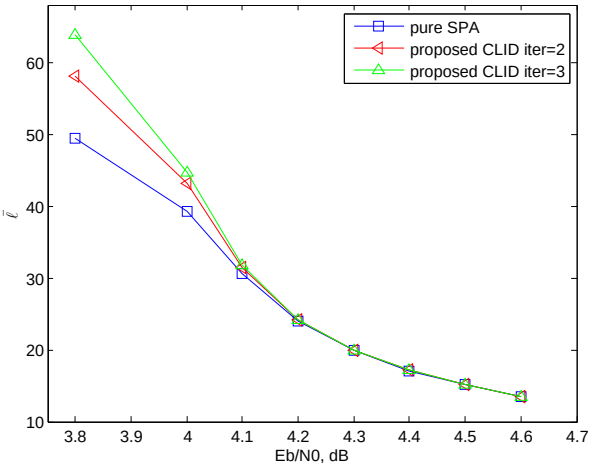


Fig. 9. Average numbber of iterations per decoded LDPC block of code-B over AWGN channel

V. Conclusion

In this paper, a novel cross layer iterative decoder for irregular LDPC coded system is presented and evaluated. The proposed decoder uses the correctly decoded link layer frames to help to correct the left erroneous ones to obtain better performance. And this is implemented by using cross layer design method and an iterative decoding architecture. The proposed decoder outperforms the pure sum-product algorithm decoder by a considerable gain at the expense of increasing the complexity moderately in low Eb/N0. Moreover, the error floor of irregular LDPC in high Eb/N0 is lowered effectively. The reduction of the frame error rate results the improvement of bandwidth efficiency. Considering the above merits, we can use the the proposed cross layer iterative decoder in irregular LDPC coded wireless system to improve the overall performance.

Acknowledgment

This work was supported by the Industrial Key Research & Development Program of Zhejiang Province, China (No.2006C21009) and the National Nature and Science Foundation of China (No.60872063).

References

[1] R. Gallager, "Low-Density Parity-Check Codes," *IRE Transactions on Information Theory*, vol. 8, no. 1, pp. 21–28, January 1962.

[2] D. MacKay and R. Neal, "Near Shannon Limit Performance of Low Density Parity Check Codes," *Electronics Letters*, vol. 33, no. 6, pp. 457–458, Mar 1997.

[3] A. Morello and V. Mignone, "DVB-S2: the Second Generation Standard for Satellite Broad-Band Services," *Proceedings of the IEEE*, vol. 94, no. 1, pp. 210–227, 2006.

[4] IEEE, "Draft Standard for Information Technology-Telecommunications and Information Exchange Between Systems–Local and Metropolitan Area Networks–Specific Requirements– Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Amendment 4: Enhancements for Higher Throughput," *IEEE Draft Std P802.11n D3.00*, Sep 2007.

[5] T. Tian, C. Jones, J. Villasenor, and R. Wesel, "Construction of Irregular LDPC Codes with Low Error Floors," *IEEE International Conference on Communications*, vol. 5, pp. 3125–3129, May 2003.

[6] V. Srivastava and M. Motani, "Cross-Layer Design: a Survey and the Road Ahead," *Communications Magazine, IEEE*, vol. 43, no. 12, pp. 112–119, Dec. 2005.

[7] G. Castagnoli, J. Ganz, and P. Graber, "Optimum Cycle Redundancy-Check Codes with 16-Bit Redundancy," *Communications, IEEE Transactions on*, vol. 38, no. 1, pp. 111–114, Jan 1990.

[8] S.-Y. Chung, T. Richardson, and R. Urbanke, "Analysis of Sum-Product Decoding of Low-Density Parity-Check Codes Using a Gaussian Approximation," *Information Theory, IEEE Transactions on*, vol. 47, no. 2, pp. 657–670, Feb 2001.

[9] T. Richardson, M. Shokrollahi, and R. Urbanke, "Design of Capacity-Approaching Irregular Low-Density Parity-Check Codes," *IEEE Transactions on Information Theory*, vol. 47, no. 2, pp. 619–637, 2001.

[10] D. Bertzekas and R. Gallager, *Data Networks*, 2nd ed. Prentice-Hall, 1992.

[11] X.-Y. Hu, E. Eleftheriou, and D. Arnold, "Regular and Irregular Progressive Edge-Growth Tanner Graphs," *IEEE Transactions on Information Theory*, vol. 51, no. 1, pp. 386–398, Jan. 2005.

[12] F. Kschischang, B. Frey, and H. Loeliger, "Factor Graphs and the Sum-Product Algorithm," *IEEE Transactions on Information Theory*, vol. 47, no. 2, pp. 498–519, 2001.

[13] Y. Kwon, M. Oh, and D. Park, "A New LDPC Decoding Algorithm Aided by Segmented Cyclic Redundancy Checks for Magnetic Recording Channels," *IEEE Transactions on Magnetics*, vol. 41, no. 7, pp. 2318–2320, 2005.

[14] S. GOUNAI and T. OHTSUKI, "Lowering Error Floor of Irregular LDPC Codes by CRC and OSD Algorithm," *IEICE Transactions on Communications*, vol. E89-B, no. 1, pp. 1–10, 2006.

[15] W. Peterson and D. Brown, "Cyclic Codes for Error Detection," *Proceedings of the IRE*, vol. 49, no. 1, pp. 228–235, Jan. 1961.

[16] S. Aikawa, Y. Motoyama, and M. Umehira, "Error Correction and Error Detection Techniques for Wireless ATM Systems," *Wireless Networks*, vol. 3, no. 4, pp. 1–6, 1997.

[17] J. Hall, "Notes on Coding Theory," 2003. [Online]. Available: http://www.mth.msu.edu/~jhall/classes/codenotes/Notes.pdf

[18] LTHC, "Efficient Optimization Program for LDPC Codes," visited at Sep. 2008. [Online]. Available: http://lthcwww.epfl.ch/research/ldpcopt/