Design of Protograph LDPC Codes for Partial Response Channels

Yi Fang, Pingping Chen, Lin Wang, *Senior Member, IEEE*, and Francis C. M. Lau, *Senior Member, IEEE*

***Abstract*—We investigate the performance of the protograph low-density parity-check (LDPC) codes, which have been shown to possess simple structures and outstanding error performance over additive white Gaussian noise (AWGN) channels, over partial response (PR) channels using the finite-length extrinsic information transfer (EXIT) algorithm. Due to the intersymbol interference (ISI) caused by the PR channels, we observe that the conventional protograph LDPC codes do not perform well in terms of error rates. We further propose a new design scheme and construct three new types of protograph LDPC codes. Unlike conventional protograph LDPC codes in which the highest-degree variable nodes are punctured, the new protograph LDPC codes have their lowest-degree variable nodes punctured. Moreover, some edge re-connections are made in one of the proposed codes. The EXIT-chart analysis, the convergence analysis and the bit-error-rate simulation have shown that all three new codes outperform the conventional protograph LDPC codes. Moreover, two of the proposed codes are superior to the regular column-weight-3 LDPC code and thus they are good alternatives compared to other error-correction codes for use in data storage systems.**

***Index Terms*—Finite-length EXIT algorithm, intersymbol interference (ISI), partial response (PR) channels, protograph LDPC codes.**

I. Introduction

USING an iterative decoding algorithm, low-density parity-check (LDPC) codes can perform very close to the Shannon limit over an additive white Gaussian noise (AWGN) channel [1], [2]. The error floor of the LDPC codes [3], however, is a major issue that has to be overcome before LDPC codes can be applied to systems that require very low error floors and high throughputs, such as data storage systems and optical communication systems. To overcome this problem, a significant amount of work has been done. Methods based on density evolution (DE) [4], [5] and extrinsic information transfer (EXIT) function [6], [7] have been proposed and used to optimize the degree distributions of the codes. Code construction mechanisms that aim to avoid structures that contribute to the error floors have also been proposed [8], [9], [10], [11]. Moreover, the error-floor problem can be resolved to a certain extent by modifying the iterative decoding algorithms [12], [13], [14].

Different from an AWGN channel, a partial response (PR) channel, such as that used in a magnetic recording system, is modeled as a discrete channel with intersymbol interference (ISI). Turbo equalization [15], [16] has been adopted to improve the error performance of communication systems over PR channels [17]. The main idea of turbo equalization is to treat the PR channel and the error-correction code as the inner code and the outer code of a serial concatenated scheme, respectively. Then one can apply the soft-input-soft-output (SISO) iterative decoding algorithm to deal with them [18], [19], [20]. By allowing an inner detector and an outer decoder to exchange extrinsic soft information iteratively, the ISI can be overcome. As the inner system is not a code but a PR channel, the iterative decoding, which includes the inner detector and the outer decoder, is named as the turbo equalization. To further alleviate the effect of the ISI and to accelerate the convergence of the iterative decoding process, precoding techniques can be employed [21].

Owing to their superior error-correction capability, LDPC codes have been applied in PR channels [22], [23]. To achieve a given bit error rate (BER), LDPC codes require a much lower signal-to-noise ratio (SNR) compared with other channel codes. It has been shown that introducing LDPC codes in PR channels can provide a coding-gain of 5.9 dB over uncoded PR channels [23]. Assuming a PR channel, recent effort has been spent on estimating the error floor of LDPC codes [24] and on researching good decoding algorithms [25], [26], [27] and coding optimization schemes [28]. Further, the modified DE [29] and the EXIT-chart [30] techniques have been proposed for designing LDPC codes in such channels. In particular, LDPC codes concatenated with a Gray mapped quaternary-phase-shift-keying (QPSK) modulator has been investigated in [30].

Recently, a novel class of LDPC codes, called multi-edge type (MET) LDPC codes, has been introduced [31]. In particular, one subclass of MET-LDPC codes, namely the protograph LDPC codes, have been shown to produce excellent error performance with low complexity over an AWGN channel [32]. Moreover, the protograph structure allows high-speed encoding and decoding implementations [33], [34]. The accumulate-repeat-accumulate (ARA) code and the accumulate-repeat-by-4-jagged-accumulate (AR4JA) code, which possess simple protograph representations to realize linear encoding and de-

Paper approved by K. Narayanan, the Editor for Coding and Communication Theory of the IEEE Communications Society. Manuscript received July 14, 2011; revised January 18, 2012.

This work was supported by the NSF of China (Nos. 60972053, 61001073 and 61102134), the European Union-FP7 (CoNHealth, No. 294923), the Fundamental Research Funds for the Central Universities (No. 201112G017), as well as the RGC of the Hong Kong SAR, China (Project No. PolyU 521809). The work was presented in part at the 2011 International Conference on Computer Research and Development (ICCRD), Shanghai, China [38].

Y. Fang, P. Chen, and L. Wang are with the College of Information Science and Technology, Xiamen University, Fujian, 361005, China (e-mail: fangyi1986812@163.com, ppchen.xm@gmail.com, wanglin@xmu.edu.cn).

F. C. M. Lau is with the Department of Electronic and Information Engineering, Hong Kong Polytechnic University, Hong Kong (e-mail: encmlau@polyu.edu.hk).

Digital Object Identifier 10.1109/TCOMM.2012.072412.110464

coding, have also been proposed and studied [35], [36]. In [37], the bit error performance of protograph LDPC codes has been studied in an ultra-wideband (UWB) short-range wireless communication channel with multipath fading. It has been observed that protograph ensembles optimized for AWGN channels degrade in UWB channels with ISI. In [38], the authors have simulated the error performance of the accumulate-repeat-by-3-accumulate (AR3A) code and an improved code over the extended class IV PR (EPR4) channel.

In this paper, we conduct an in-depth investigation on the error performance of protograph LDPC codes in the PR channels. We apply the finite-length EXIT function [39], [40] for analyzing the protograph LDPC codes. The results suggest that when applied in PR channels, the conventional protograph LDPC codes, such as the AR3A code and the AR4JA code, cannot maintain their excellent error performance as in AWGN channels. To overcome this weakness, we propose a novel protograph design scheme with the help of the EXIT-chart analysis. Based on the scheme, we design three new types of protograph LDPC codes for use in the PR channels. Our aim is to improve the error performance of protograph LDPC codes without increasing their complexities. Both the theoretical analyses and the simulated results have illustrated that all three new codes outperform the conventional protograph LDPC codes while two of the proposed codes are superior to the regular column-weight-3 LDPC code [41]. Because of their excellent error performance and low complexities, two of the proposed protograph LDPC codes are very good candidates for use in data storage systems.

The remainder of this paper is organized as follows. In Section II, the system model over PR channels is described. In Section III, we elaborate how the finite-length EXIT algorithm is used in analyzing the protograph LDPC codes in our system. In Section IV, we analyze the performance of the conventional protograph LDPC codes with the EXIT chart. We also propose a new design scheme for the protograph LDPC codes, and construct three new types of protograph LDPC codes. The performance of the codes are compared and presented in Section V, and conclusions are given in Section VI.

II. System Model

The system model being considered is described as follows. The information bits are firstly encoded using the protograph LDPC code, which is treated as an outer code. Then the binary coded bits $v_i \in \{0,1\}$ $(i = 1, 2, \ldots)$ are converted into $x_i = (-1)^{v_i} \in \{+1, -1\}$ by a modulator. The modulated signal is further passed through a PR channel that is modeled as an ISI channel with AWGN. Such a channel can be regarded as the inner code of the concatenated system.

The output of the channel, denoted by y_i, can be expressed as a partial response polynomial plus a Gaussian noise, i.e.,

$$y_i = \sum_{j=0}^{K+1} h_j x_{i-j} + n_i \tag{1}$$

where h_j $(j = 0, 1, \ldots, K+1)$ are the tap coefficients corresponding to the partial response channel, n_i is the Gaussian noise sample with zero mean and variance $N_0/2$, and N_0 denotes the noise power-spectral density. Furthermore, the transfer function of the channel can be expressed as

$$H(D) = (1-D)(1+D)^K = \sum_{j=0}^{K+1} h_j D^j. \tag{2}$$

In this paper, we primarily consider the dicode channel (i.e., $K = 0$) and the EPR4 channel (i.e., $K = 2$) commonly used in magnetic recording research [25], the transfer functions of which are $H(D) = 1 - D$ and $H(D) = 1 + D - D^2 - D^3$, respectively.

At the receiving terminal, the decoder structure is formed by one inner SISO detector and one outer SISO decoder — the former one for the PR channel and the latter one for the outer code. In the concatenated scheme, the detector and the decoder exchange extrinsic log-likelihood-ratio (LLR) messages [4], [5], [6], [7] iteratively so as to increase the accuracy of the decoded messages. In our study, the inner detector and the outer decoder are, respectively, implemented with the Bahl-Cocke-Jelinek-Raviv (BCJR) algorithm [42], [43] and the belief propagation (BP) algorithm [4], [5]. Moreover, we do not consider interleaving or precoding in our system model.

III. Finite-length EXIT Algorithm of Protograph LDPC Codes

A. Background

The EXIT function [6], [7] has been proposed to predict the convergence behavior of the iterative processors used in a variety of communication problems. In particular, the function is very useful in tracing the convergence behavior of the iterative decoding schemes. The EXIT chart, which describes the asymptotic decoding trajectory of a decoder, consists of two EXIT curves. The decoder will converge successfully if the two EXIT curves do not touch or cross each other except at the value of unity. Using this tool, researchers can evaluate the performance of a concatenated code/channel and derive the decoding threshold of an iterative decoding algorithm. The threshold represents the SNR above which an arbitrarily small BER can be achieved as the block length of the code approaches infinity.

Yet, as an asymptotic performance analysis method, the infinite-length EXIT chart [6], [7] can no longer be used to analyze codes with short block length. It is because the properties of typicality and ergodicity cannot be retained. In [39], [40], an EXIT chart consisting of two EXIT bands have been proposed for analyzing codes with finite length and over different types of channels. Each EXIT band is composed of an expected EXIT curve, an upper-bound curve and a lower-bound curve. The EXIT band, bounded by the upper-bound curve and the lower-bound curve, is formed to ensure that the individual EXIT curves of the finite-length blocks lie within the band with a high probability. Then, one can conclude that the turbo decoder will converge successfully with a high probability if the two EXIT bands corresponding to the inner detector and the outer decoder only touch each other at the value of unity.

B. Finite-length EXIT algorithm of protograph LDPC codes in PR channels

One important assumption of the finite-length EXIT analysis is that the output extrinsic LLR messages of each SISO decoder should approximately be Gaussian distributed. In [40], the above assumption has been validated for individual finite-length output LLR values in each coded ISI frame. In the following, we briefly describe how we adopt the finite-length EXIT analysis to our system.

First, we analyze the distribution of the output extrinsic LLR values of the SISO detector/decoder within each coded frame. We use an AR3A code with a code rate of 0.8 and an information length of 4096, and assume a dicode channel with $E_b/N_0 = 4.0$ dB (E_b is the average energy per information bit and $N_0/2$ is the power spectral density of noise). Denote v_i as the i-th bit of an AR3A code frame, L_i as the channel LLR message input to the BCJR detector, and z_i as the corresponding extrinsic LLR value output by the BCJR detector. Using Monte Carlo simulations, we evaluate the values of z_i within each frame for $v_i = 0$ and $v_i = 1$. In Fig. 1, we plot the conditional probability density functions (PDFs) of z_i within each frame which are denoted by $f(z_i|v_i = 0)$ and $f(z_i|v_i = 1)$. Furthermore, we denote u_0 and u_1 as the average extrinsic LLR values output by the BCJR detector corresponding to $v_i = 0$ and $v_i = 1$, respectively. In other words, $u_0 = E(z_i|v_i = 0)$ and $u_1 = E(z_i|v_i = 1)$, where $E(\cdot)$ is the expectation operator. In the same figure, we plot $N(u_0, 2u_0)$ and $N(u_1, 2u_1)$ for comparison where $N(u, 2u)$ denotes a symmetric Gaussian distribution with mean u and variance $2u$. The results in Fig. 1 indicate that the conditional PDFs of the extrinsic LLR values (i.e., $f(z_i|v_i = 0)$ and $f(z_i|v_i = 1)$) agree well with the symmetric Gaussian distributions (i.e., $N(u_0, 2u_0)$ and $N(u_1, 2u_1)$). Simulations have also been performed for the AR4JA code and over the EPR4 channel, and similar observations are obtained. Moreover, we perform simulations on the protograph LDPC decoder and we find that the output extrinsic LLR values within each coded frame follow symmetric Gaussian distributions. Based on the simulation results, we conclude that the finite-length EXIT analysis can be applied to analyze our system.

Our finite-length EXIT analysis is based on the block diagram depicted in Fig. 2. To simplify the analysis, we assume that both the input (*a-priori*) and the output (extrinsic) LLR values of the detector/decoder follow symmetric Gaussian distributions. Given a sequence of transmitted coded bits and channel messages, the mutual information (MI) (i) between the coded bits and the input LLRs for the inner detector; (ii) between the coded bits and the output LLRs for the inner detector; (iii) between the coded bits and the input LLRs for the outer decoder; and (iv) between the coded bits and the output LLRs for the outer decoder; can therefore be computed by [6]

$$I_{\mathrm{A/E}} = 1 - \int_{-\infty}^{\infty} \frac{\exp\left(-\frac{(\xi-\sigma^2/2)^2}{2\sigma^2}\right)}{\sqrt{2\pi\sigma^2}} \log_2\left[1+\exp(-\xi)\right] \mathrm{d}\xi \tag{3}$$

where $\sigma^2/2$ represents the corresponding (normalized) mean of the LLR values obtained through Monte Carlo simulations. The subscript "A" representing "*a-priori*" is used to denote the input of the detector/decoder while the subscript "E" standing for "extrinsic" is used to denote the output of the detector/decoder. The inverse function of (3) is further given by [7]

$$\sigma = \begin{cases} \gamma_1 I^2 + \gamma_2 I + \gamma_3\sqrt{I} & \text{if } 0 \le I \le 0.3646 \\ \gamma_4 \ln[\gamma_5(1-I)] + \gamma_6 I & \text{otherwise} \end{cases} \tag{4}$$

where $\gamma_1 = 1.09542, \gamma_2 = 0.214217, \gamma_3 = 2.33737, \gamma_4 = -0.706692, \gamma_5 = 0.386013$ and $\gamma_6 = 1.75017$. We use "IO" to denote the inner-detector-to-outer-decoder direction and "OI" to denote the outer-decoder-to-inner-detector direction. For example, in Fig. 2, $I_{\mathrm{E,IO}}$ represents the MI between the coded bits and the output extrinsic LLRs passing from the inner detector to the outer decoder and $I_{\mathrm{E,OI}}$ represents the corresponding MI passing from the outer decoder to the inner detector. In addition, we use L to represent a LLR value. Then, for a given E_b/N_0, we compute the two EXIT bands, one for the inner detector and one for the outer decoder, as follows.

Computing the MI of the output extrinsic LLRs of the inner detector

1) For the given E_b/N_0, we generate a sequence of information bits. The information bits are then encoded by the protograph LDPC encoder (and punctured). Afterwards, the coded bits $\{v_i\}$ are converted into appropriate signals $\{x_i = (-1)^{v_i}\}$ by the modulator and sent to the noisy PR channel. Based on the received signals from the channel, we calculate the channel LLR values of the coded bits and we denote these values by $\{L_{\mathrm{ch}}\}$.
2) For a given $I_{\mathrm{A,IO}} \in [0, 1]$, we compute $\sigma_{\mathrm{A,IO}}$ us-

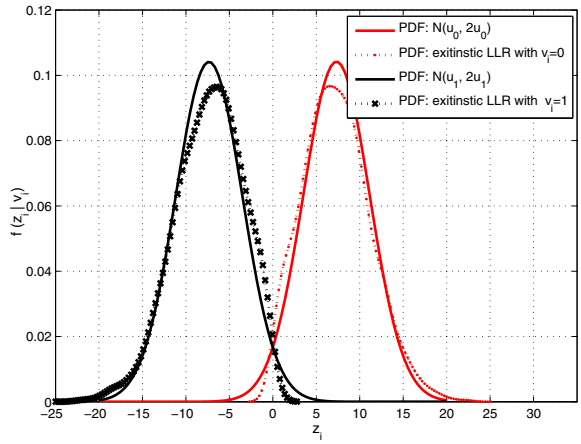


Fig. 1. Conditional probability density functions (PDFs) of the output LLR values z_i of the BCJR detector when the input coded bit $v_i = 0$ and $v_i = 1$. An AR3A code with a code rate of 0.8 and length 4096 is used, i.e., $R = 0.8, k = 4096$. $E_b/N_0 = 4.0$ dB. A dicode channel is assumed.

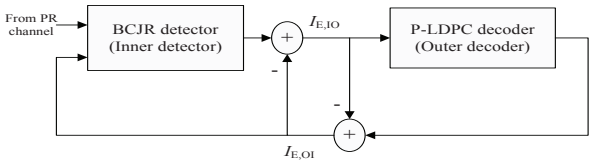


Fig. 2. Block diagram of the finite-length EXIT analysis for the protograph LDPC (P-LDPC) codes over PR channels.

ing (4). Then we generate a sequence $\{\hat{L}_{\mathrm{A,IO}}\} = (\hat{L}^{(1)}_{\mathrm{A,IO}}, \hat{L}^{(2)}_{\mathrm{A,IO}}, \ldots)$ following the symmetric Gaussian distribution $N(\frac{\sigma^2_{\mathrm{A,IO}}}{2}, \sigma^2_{\mathrm{A,IO}})$. Based on the Gaussian-distributed sequence and the coded sequence $\{v_i\}$, we form the sequence $\{L_{\mathrm{A,IO}}\} = (L^{(1)}_{\mathrm{A,IO}}, L^{(2)}_{\mathrm{A,IO}}, \ldots)$ where $L^{(i)}_{\mathrm{A,IO}} = (-1)^{v_i}\hat{L}^{(i)}_{\mathrm{A,IO}}$ $(i = 1, 2, \ldots)$.

3) Passing the sequence $\{L_{\mathrm{ch}}\}$ and the sequence $\{L_{\mathrm{A,IO}}\}$ into the BCJR detector (inner detector), we can measure the output extrinsic LLR sequence $\{L_{\mathrm{E,IO}}\}$ which can be expressed as

$$L_{\mathrm{E,IO}} = F_{\mathrm{inner}}(L_{\mathrm{A,IO}}, L_{\mathrm{ch}}) \tag{5}$$

where $F_{\mathrm{inner}}(\cdot)$ represents the LLR processor of the inner detector.

4) We then calculate the mean $u_{\mathrm{E,IO}}$ of the sequence $\{L_{\mathrm{E,IO}}\}$. Assuming that the sequence $\{L_{\mathrm{E,IO}}\}$ follows a symmetric Gaussian distribution, we evaluate the MI $I_{\mathrm{E,IO}}$ of the output extrinsic LLRs of the inner detector using (3).

5) For the same value of $I_{\mathrm{A,IO}}$, we repeat Step 1 to Step 4 N_A times to obtain a set of $I_{\mathrm{E,IO}}$ values $(I_{\mathrm{E,IO}}(i), i = 1, 2, \ldots, N_A)$. We denote the expected value of $I_{\mathrm{E,IO}}$ by $E[I_{\mathrm{E,IO}}]$, which can be considered as a typical value of all the frames for a given block length. We further compute the variance of $I_{\mathrm{E,IO}}$, represented by $\mathrm{var}[I_{\mathrm{E,IO}}]$, using [44]

$$\mathrm{var}[I_{\mathrm{E,IO}}] = \frac{1}{N_A - 1}\sum_{i=1}^{N_A}\left(I_{\mathrm{E,IO}}(i) - E[I_{\mathrm{E,IO}}]\right)^2. \tag{6}$$

6) We repeat Step 1 to Step 5 for different values of $I_{\mathrm{A,IO}} \in [0, 1]$. We then obtain an EXIT band which includes an expected EXIT curve, an upper-bound curve and a lower-bound curve. In particular, the expected EXIT curve depicts the relationship between $E[I_{\mathrm{E,IO}}]$ and $I_{\mathrm{A,IO}}$ for a given E_b/N_0, i.e.,

$$E[I_{\mathrm{E,IO}}] = f_{\mathrm{inner}}(I_{\mathrm{A,IO}}, E_b/N_0) \tag{7}$$

where $f_{\mathrm{inner}}(\cdot)$ represents the output extrinsic MI processor of the inner detector. The upper-bound curve and the lower-bound curve, which ensure that all the individual EXIT curves of the finite-length blocks be contained with a high probability, can be described by $(I_{\mathrm{A,IO}}, E[I_{\mathrm{E,IO}}]+3\sqrt{\mathrm{var}[I_{\mathrm{E,IO}}]})$ and $(I_{\mathrm{A,IO}}, E[I_{\mathrm{E,IO}}]-3\sqrt{\mathrm{var}[I_{\mathrm{E,IO}}]})$, respectively [40].

Computing the MI of the output extrinsic LLRs of the outer decoder

Considering the protograph LDPC decoder (outer decoder) and using similar procedures as above, we can obtain different expected values and different variance values of the output extrinsic MI, i.e., $E[I_{\mathrm{E,OI}}]$ and $\mathrm{var}[I_{\mathrm{E,OI}}]$, for different $I_{\mathrm{A,OI}}$ values. We express their relationships by

$$E[I_{\mathrm{E,OI}}] = f_{\mathrm{outer}}(I_{\mathrm{A,OI}}) \tag{8}$$

$$\mathrm{var}[I_{\mathrm{E,OI}}] = \frac{1}{N_A - 1}\sum_{i=1}^{N_A}\left(I_{\mathrm{E,OI}}(i) - E[I_{\mathrm{E,OI}}]\right)^2 \tag{9}$$

where $f_{\mathrm{outer}}(\cdot)$ represents the output extrinsic MI processor of the outer (LDPC) decoder. Hence, we can form an EXIT band, which consists of an expected EXIT curve $(I_{\mathrm{A,OI}}, E[I_{\mathrm{E,OI}}])$, an upper-bound curve $(I_{\mathrm{A,OI}}, E[I_{\mathrm{E,OI}}]+3\sqrt{\mathrm{var}[I_{\mathrm{E,OI}}]})$ and a lower-bound curve $(I_{\mathrm{A,OI}}, E[I_{\mathrm{E,OI}}] - 3\sqrt{\mathrm{var}[I_{\mathrm{E,OI}}]})$, for the outer decoder.

Note also that

- N_A is set to 50000 to ensure the accuracy of the finite-length EXIT algorithm;
- the output extrinsic MI function of the inner detector, i.e., $f_{\mathrm{inner}}(\cdot)$, is independent of the type of code, but depends only on E_b/N_0 and the type of channel;
- the output extrinsic MI function of the outer decoder, i.e., $f_{\mathrm{outer}}(\cdot)$ is related only to the type of code;
- the output LLRs of the inner detector becomes the input LLRs of the outer decoder, and *vice versa*; and
- the two EXIT bands can be plotted in the same figure (the axes of the EXIT band of the outer decoder have to be switched) to predict the performance of the code.

IV. Analysis and Design of Protograph LDPC Codes

To begin with, we analyze the conventional protograph LDPC codes in PR channels by means of the EXIT bands. Then, we propose a design scheme for the protograph LDPC codes. Based on the proposed scheme, we design three new types of protograph LDPC codes.

A. Analysis of conventional protograph LDPC codes

A protograph, which was first introduced by [32], can be seen as a Tanner graph containing a relatively small number of nodes. A protograph consists of a variable-node set, a check-node set and an edge set, denoted by V, C, and Ψ, respectively. Moreover, each edge connects a variable node and a check node. However, unlike LDPC codes, a protograph allows parallel edges. As a result, the mapping $\psi \in \Psi \to (v_\psi, c_\psi) \in V \times C$ may not be one-to-one. We denote the adjacency matrix corresponding to a protograph as a *base matrix*. A large protograph can be further obtained by performing a "copy-and-permute" operation on the base matrix [32]. Consequently, a code with an arbitrary block length can be generated by performing the "copy-and-permute" operation repeatedly. The resultant graph is called the *derived graph* and the corresponding LDPC code is called a *protograph code*. Protograph codes not only enable linear encoding and decoding to be implemented easily, but also provide superior error performance at a high code rate in AWGN channels.

We consider two protograph codes, namely the AR3A code and the AR4JA code [35], [36], with a code rate of $R = (n + 1)/(n + 2)$. The corresponding bases matrices, denoted by $\boldsymbol{B}_{A3}$ and $\boldsymbol{B}_{A4}$, are expressed as

$$\boldsymbol{B}_{A3} = \begin{pmatrix} 1\,2\,1\,0\,0\overbrace{0\,0\,\cdots\,0\,0}^{2n} \\ 0\,2\,1\,1\,1\,2\,1\,\cdots\,2\,1 \\ 0\,1\,2\,1\,1\,1\,2\,\cdots\,1\,2 \end{pmatrix} \tag{10}$$

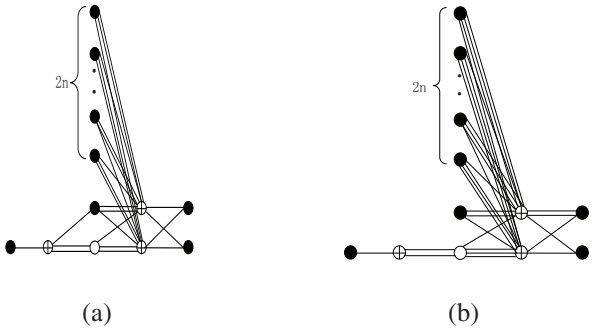


Fig. 3. The protographs of (a) an AR3A code and (b) an AR4JA code.

$$\boldsymbol{B}_{A4} = \begin{pmatrix} 1\,2\,0\,0\,0\,\overbrace{0\,0\,\cdots\,0\,0}^{2n} \\ 0\,3\,1\,1\,1\,3\,1\,\cdots\,3\,1 \\ 0\,1\,2\,2\,1\,1\,3\,\cdots\,1\,3 \end{pmatrix} \tag{11}$$

in which the sixth and the seven columns are repeated in the last $2n$ columns. We assume that the i-th column of the matrix corresponds to the i-th variable node and the j-th row of the matrix corresponds to the j-th check node. The structures of the codes are illustrated in Fig. 3. In the figure, the transmitted nodes and the punctured nodes are shown as dark circles and blank circles, respectively. Further, the check nodes are denoted by circles with a plus sign. The AR3A code is then formed by puncturing the largest-degree variable node (corresponding to the second column of $\boldsymbol{B}_{A3}$); and the AR4JA code is also created in a similar way, i.e., by puncturing the largest-degree variable node (corresponding to the second column of $\boldsymbol{B}_{A4}$) [35], [36].

The AR3A code and the AR4JA code have been shown to possess excellent error performance over an AWGN channel. We also consider the regular column-weight-3 LDPC code [41], which has been shown to provide better error performance than the irregular LDPC codes over PR channels, but is found to be outperformed by the AR3A code and the AR4JA code in AWGN channels [35], [36].

In Fig. 4[1], we plot the EXIT bands of the AR3A code, the AR4JA code and the regular column-weight-3 LDPC code in a dicode channel and in an EPR4 channel. The parameters used are listed as follows.

- Information length k equals 4096.
- Code rate $R = 0.8$.
- $E_b/N_0 = 4.0$ dB.
- The LDPC decoder performs 15 BP iterations.
- In the construction of the AR3A code and the AR4JA code, we use $n = 3$ and apply the "copy-and-permutate" operations 512 times.

We define the region between the two expected EXIT curves (one for the inner detector and one for the outer decoder) corresponding to a code as the *decoding tunnel*. We can observe from Fig. 4 that the decoding tunnel of the AR4JA code is narrower than that of the AR3A code, which is also narrower than that of the regular LDPC code. The results imply that the convergence speeds of the two protograph codes are slower than that of the regular LDPC code. Moreover, the AR4JA code is performing worse than the AR3A code. Fig. 4(a) further reveals that the AR4JA code may not converge to the unity MI value at $E_b/N_0 = 4.0$ dB in a dicode channel. It is because the EXIT band for the inner detector overlaps with that for the AR4JA code when the MI value is small. In an EPR4 channel, Fig. 4(b) shows that the EXIT band for the inner detector overlaps with those for the AR4JA code, the AR3A code and the regular LDPC code. In other words, all these codes may not converge to the unity MI value. According to the simulation results shown in [45], the symmetric information rates (SIRs) for the dicode channel and the EPR4 channel with a code rate of 0.8 are about 2.80 dB and 3.20 dB, respectively. Consequently, there is room for improving the performance of the protograph codes in PR channels because even at 4.0 dB, the AR4JA code may not converge in the dicode channel and in the EPR4 channel.

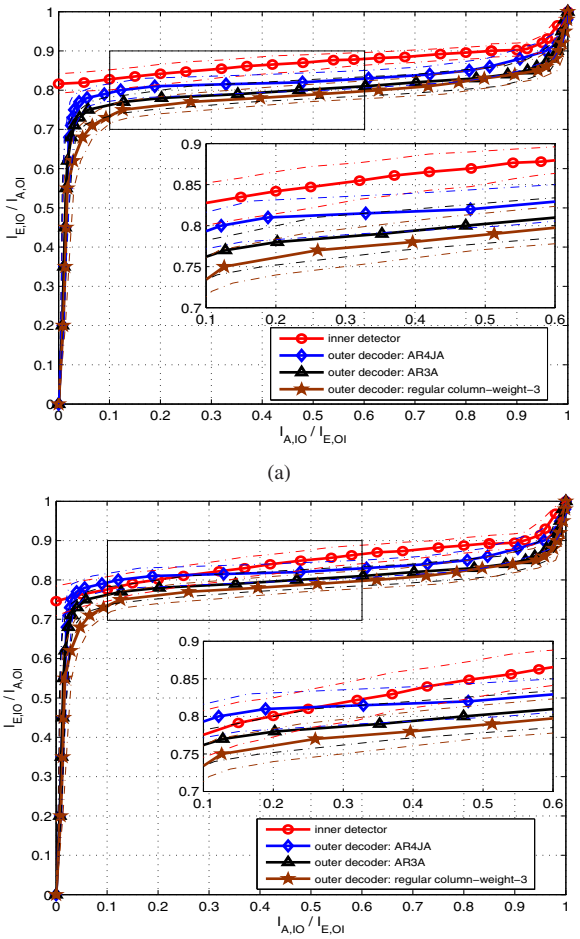


Fig. 4. EXIT bands of the two protograph codes and the regular LDPC code in (a) a dicode channel and (b) an EPR4 channel. Expected EXIT curves are shown by solid lines, upper-bound curves and lower-bound curves are represented by dotted lines. $n = 3, R = 0.8, k = 4096$ and $E_b/N_0 = 4.0$ dB. (The color version of this figure can be downloaded from the IEEE Xplore website or ce.xmu.edu.cn/wanglin/publications/color-figs.pdf or www.eie.polyu.edu.hk/∼encmlau/color-figs.pdf.)

[1]The color version of this figure can be downloaded from the IEEE Xplore website or ce.xmu.edu.cn/wanglin/publications/color-figs.pdf or www.eie.polyu.edu.hk/∼encmlau/color-figs.pdf.

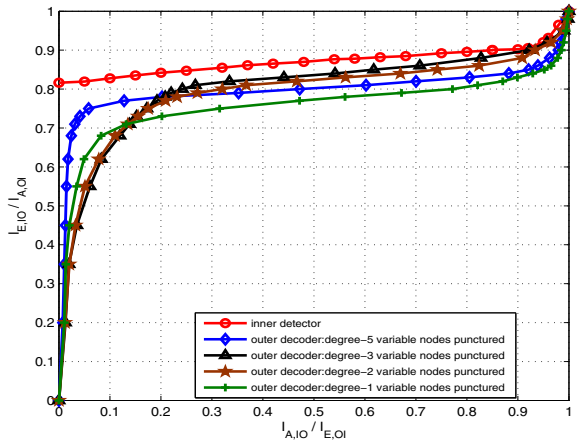


Fig. 5. Expected EXIT curves of four protograph codes with different types of punctured variable nodes in a dicode channel. $n = 3, R = 0.8, k = 4096$ and $E_b/N_0 = 4.0$ dB.

B. Design scheme of the protograph LDPC codes

For an AWGN channel, the decoder is a simple LPDC decoder. Channel LLR messages are first passed into the decoder and then the LLR messages are exchanged iteratively between the variable-node set and the check-node set in the decoder. However, the decoder for a PR channel consists of an inner detector (BCJR detector) and an outer decoder (LDPC decoder). The channel messages distorted by ISI and AWGN are first passed to the inner detector. Subsequently, the extrinsic LLR values are exchanged between the inner detector and the outer decoder in each turbo iteration. While it is known that punctured variable nodes with a higher degree (as in the AR3A code) can be recovered more easily for an AWGN channel, such punctured nodes may produce worse error performance for a PR channel.

To verify the above conjecture, we form four protograph codes using the same base matrix $\boldsymbol{B}_{A3}$ in (10), and the codes differ in the sense that the degree-1, degree-2, degree-3 and degree-5 variable nodes are punctured, respectively. To simplify the analysis, we only consider their expected EXIT curves here. Fig. 5 shows the expected EXIT curves of the four protograph codes in a dicode channel. The parameters used are the same as those listed in Sect. IV-A. It can be seen that the protograph code with punctured degree-1 variable nodes provides the largest decoding tunnel. Since a larger decoding tunnel implies a faster convergence rate and a lower decoding threshold, we deduce that the protograph code with punctured degree-1 variable nodes have the best performance.

We further simulate the error performance of two of the aforementioned protograph codes, in which the largest-degree (degree-5) and the smallest-degree (degree-1) variable nodes are punctured, respectively. We investigate the average number of bit errors per codeword for the two codes in the inner detector e_{inner} and outer decoder e_{outer} at the end of first turbo iteration and the second turbo iteration when $E_b/N_0 = 4.8$ dB[2]. The results are shown in Table I where $T_{\max}$ represents the number of local (BP) iterations performed in the LDPC decoder during one turbo iteration. As seen from the table, the protograph code with punctured degree-1 variable nodes has an average of less than 1 bit error per codeword at the end of the second turbo iteration while the code with punctured degree-5 variable nodes has an average of 9 to 11 bit errors. Thus, the average number of error bits for the code with punctured degree-1 variable nodes decreases much faster than that for the code with punctured degree-5 variable nodes. For the code with punctured degree-1 variable nodes, the error performance hardly changes when $T_{\max}$ increases from 15 to 100. It implies that using 15 local iterations is sufficient for the decoder to converge. For the code with punctured degree-5 variable nodes, moreover, the number of bit errors is reduced from 11 to 9 bits when $T_{\max}$ increases from 15 to 100. It shows that increasing the number of local iterations can help the decoder to converge faster in this case. It also shows that the punctured degree-5 variable nodes can be more readily recovered with a larger number of local iterations. In summary, the results in Fig. 5 and Table I indicate that it is relatively more important to retain the largest-degree variable nodes compared with the smallest-degree variable nodes for a protograph code during puncturing; and that the largest-degree variable nodes play a more important role during the turbo iterations which include a BCJR detector and a BP decoder.

In addition, it has been shown that degree-2 variable nodes helps producing good error performance in the waterfall region but too many degree-2 variable nodes can give rise to an error floor easily in the high SNR region [4], [5], [11], [34]. Hence, we should keep a certain proportion of degree-2 variable nodes but not too large in the protograph code.

Based on the aforementioned observations, we propose a two-step approach to the design of the protograph (LDPC) codes.

Design of the Protograph (LDPC) Codes

1) Puncture the variable nodes with the smallest degree instead of the largest degree in the protograph code.
2) Reduce the proportion of degree-2 variable nodes by adding an extra connection to some of the degree-2 variable nodes. For every edge added, one edge is removed from a variable node with the largest degree so as to keep the complexity (number of connections) unchanged.

C. Three new types of protograph LDPC codes

Based on the proposed design algorithm, we construct three new types of protograph LDPC codes for use in PR channels. The first type of protograph LDPC code is similar to that of the AR3A code, but the variable nodes with the smallest degrees (i.e., degree 1) instead of largest degrees are punctured. We call this code *improved ARA1* (IARA1) code. The protograph of the code is depicted in Fig. 6(a) and the corresponding base matrix, denoted by $\boldsymbol{B}_{IA1}$, is the same as $\boldsymbol{B}_{A3}$ in (10).

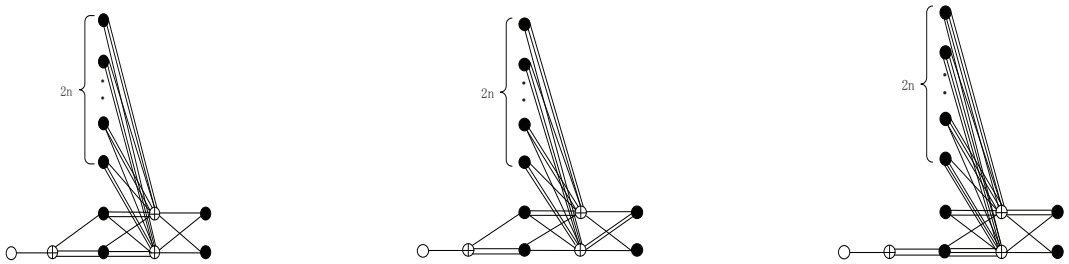
As can be seen in Fig. 6(a), there are two degree-2 nodes for every $2n + 4$ variable nodes in the IARA1 code, i.e., $1/(n + 2)$ of the variable nodes are of degree-2. Next, we reduce the proportion of degree-2 variable nodes in the IARA1 code and form the *improved ARA2* (IARA2) code. To achieve the goal, we perform the following. For every $2n + 4$ variable

[2]When $E_b/N_0 = 4.0$ dB, both codes have large BERs. Thus we need to use a larger E_b/N_0 dB when comparing the number of bit errors for these two codes.

TABLE I

AVERAGE NUMBER OF BIT ERRORS PER CODEWORD AT THE INNER DETECTOR e_{inner} AND AT THE OUTER DECODER e_{outer} AFTER THE FIRST AND SECOND TURBO ITERATIONS. THE NUMBER OF BP (LOCAL) ITERATIONS USED IS $T_{\max} = 15$ AND 100. TWO PUNCTURING SCHEMES (PUNCTURED DEGREE-5 AND PUNCTURED DEGREE-1) ARE USED BASED ON OF THE PROTOGRAPH CORRESPONDING TO $\boldsymbol{B}_{A3}$. THE PARAMETERS USED ARE $n = 3, R = 0.8, k = 4096$ AND $E_b/N_0 = 4.8$ dB. AN EPR4 CHANNEL IS ASSUMED.

Degree Type for Puncturing	$T_{\max} = 15$		$T_{\max} = 100$	
	First Iteration	**Second Iteration**	**First Iteration**	**Second Iteration**
	$e_{\text{inner}}/e_{\text{outer}}$	$e_{\text{inner}}/e_{\text{outer}}$	$e_{\text{inner}}/e_{\text{outer}}$	$e_{\text{inner}}/e_{\text{outer}}$
Degree-5	255.9/152.4	89.1/11.01	255.2/96.4	65.6/9.28
Degree-1	255.0/136.8	77.1/0.94	255.6/134.7	75.7/0.92



(a) (b) (c)

Fig. 6. The protographs of (a) an IARA1 code; (b) an IARA2 code; and (c) an IARA3 code. For the protograph of the IARA1 code, the variable node (dark circle) at the left bottom has 5 edges while the one at the right top has 2 edges. For the protograph of the IARA2 code, the variable node (dark circle) at the left bottom has 4 edges while the one at the right top has 3 edges. The remaining connections for these two codes are the same.

nodes in the IARA1 code, we randomly select one of the two degree-2 variable nodes (corresponding to the fourth and fifth columns of $\boldsymbol{B}_{IA1}$ or $\boldsymbol{B}_{A3}$) and add a new edge to connect it to one of the two highest-degree check nodes (i.e., the check nodes corresponding to the second and third rows of $\boldsymbol{B}_{IA1}$ or $\boldsymbol{B}_{A3}$). At the same time, for the highest-degree check node selected, we remove the edge that connects the check to the variable node with the largest degree such that the total number of edges in the code remains unchanged. Consequently, the encoding complexity and the decoding complexity of the new code are similar to those of the original one. According to the aforementioned construction method, there are totally four different realizations for the IARA2 code. The protograph of one realization is depicted in Fig. 6(b) and the corresponding base matrix, denoted by $\boldsymbol{B}_{IA2}$, is given by

$$\boldsymbol{B}_{IA2} = \begin{pmatrix} 1\,2\,1\,0\,0 \overbrace{0\;0\;\cdots\;0\;0}^{2n} \\ 0\,1\,1\,2\,1\,2\;1\;\cdots\;2\;1 \\ 0\,1\,2\,1\,1\,1\;2\;\cdots\;1\;2 \end{pmatrix}. \tag{12}$$

Note also that the code rates of the two new codes, i.e., IARA1 code and IARA2 code, remain at $R = (n+1)/(n+2)$.

Using the base matrix of the AR4JA code, i.e., $\boldsymbol{B}_{A4}$, we puncture the smallest-degree variable node and form the *improved ARA3* (IARA3) code shown in Fig. 6(c). Thus, the IARA3 and the AR4JA code share the same base matrix. However, the smallest-degree and the largest-degree variable nodes are punctured, respectively, in the construction of the IARA3 and the AR4JA code.

V. PERFORMANCE COMPARISON

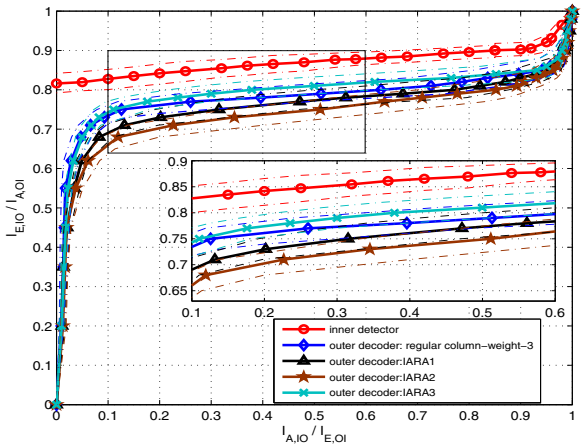
In this section, we compare the performance of the AR3A code, the AR4JA code, the regular column-weight-3 LDPC code, the proposed IARA1 code, the proposed IARA2 code and the proposed IARA3 code. The channels being considered are the AWGN channel, the dicode channel and the EPR4 channel. Unless otherwise stated, the parameters used are the same as those listed in Sect. IV-A.

A. EXIT-chart Analysis

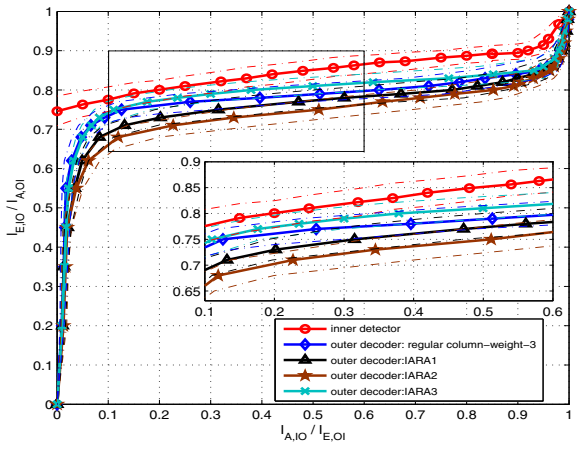
Firstly, we compute the decoding threshold of the regular column-weight-3 LDPC code in an AWGN channel using the infinite EXIT algorithm in [7] and those of the other five punctured protograph LDPC codes using the protograph EXIT algorithm in [46]. The decoding thresholds of the AR4JA code, the AR3A code, the regular column-weight-3 code, the IARA1 code, the IARA2 code, and the IARA3 code with a code rate of 0.8 are, respectively, 2.411 dB, 2.284 dB, 2.586 dB, 2.627 dB, 2.627 dB, and 2.568 dB. Moreover, the channel capacity is 2.040 dB. By comparing the thresholds, we expect the AR4JA code and the AR3A code to outperform the other four codes in an AWGN channel. Furthermore, among the three proposed protograph codes, the IARA3 code possesses the lowest threshold and thus should produce a relatively good error performance.

We then examine the performance of the new codes on PR channels by means of the finite-length EXIT analysis. The EXIT bands of the proposed protograph LDPC codes and the regular column-weight-3 LDPC code in a dicode channel and in a EPR4 channel are plotted in Fig. 7[3]. The results indicate that the decoding tunnels of the proposed IARA1 code and IARA2 code are larger than that of the regular LDPC code while the proposed IARA3 code produces the

[3]The color version of this figure can be downloaded from the IEEE Xplore website or ce.xmu.edu.cn/wanglin/publications/color-figs.pdf or www.eie.polyu.edu.hk/∼encmlau/color-figs.pdf.



(a)



(b)

Fig. 7. EXIT bands of the proposed IARA1 code, the proposed IARA2 code, the proposed IARA3 code and the regular column-weight-3 LDPC code in (a) a dicode channel and (b) an EPR4 channel. Expected EXIT curves are shown by solid lines, upper-bound and lower-bound curves are represented by dotted lines. $n = 3, R = 0.8, k = 4096$ and $E_b/N_0 = 4.0$ dB. (The color version of this figure can be downloaded from the IEEE Xplore website or ce.xmu.edu.cn/wanglin/publications/color-figs.pdf or www.eie.polyu.edu.hk/∼encmlau/color-figs.pdf.)

smallest decoding tunnel. Since the regular LDPC code is found to outperform the conventional protograph LDPC codes in a PR channel (see Fig. 4), the IARA1 code and the IARA2 code should provide faster convergence speeds and lower decoding threshold values compared with the conventional protograph LDPC codes. The IARA1 code and the IARA2 code should also provide a higher coding gain compared with the regular column-weight-3 LDPC code, which has been shown to possess superior error performance in Section IV-A and in [41].

B. Convergence Speed

We also study the convergence behavior of the proposed protograph codes. At the end of the first turbo iteration and the second turbo iteration, we evaluate the average number of bit errors per codeword at the inner detector e_{inner} and at the outer decoder e_{outer}. We use $E_b/N_0 = 4.8$ dB here[4]. Moreover, 15 BP (local) iterations are performed in the outer (LDPC) decoder in each turbo iteration. (Unless otherwise stated, 15 BP local iterations will be used in all subsequent simulations.) We also send a total of 5000 blocks in the simulation. Table II shows the error results of the proposed protograph codes and those of the AR3A code, the AR4JA code and the regular column-weight-3 LDPC code. We can observe that after 2 turbo iterations, the average numbers of error bits of the proposed IARA1 code and IARA2 code decrease much faster than those of the two conventional protograph codes (i.e., AR3A code and AR4JA code) — the average numbers of error bits of the proposed IARA1 code and IARA2 code are both less than 1 while those of the AR3A code and AR4JA code are 11 and 101, respectively. The regular LDPC code is the third-best with an average of 1.65 error bits while the IARA3 code achieves an average of 7.6 error bits. In summary, the convergence speeds of the proposed IARA1 code and IARA2 code are the best among the six codes.

[4]When $E_b/N_0 = 4.0$ dB, all codes have large BERs. Thus we need to use a larger E_b/N_0 dB when comparing the number of bit errors for different codes.

TABLE II
AVERAGE NUMBER OF BIT ERRORS PER CODEWORD AT THE INNER DETECTOR e_{inner} AND AT THE OUTER DECODER e_{outer} AFTER THE FIRST AND SECOND TURBO ITERATIONS. THE PARAMETERS USED ARE $n = 3, R = 0.8, k = 4096$ AND $E_b/N_0 = 4.8$ dB. AN EPR4 CHANNEL IS ASSUMED.

Code Type	First Iteration	Second Iteration
	$e_{\text{inner}}/e_{\text{outer}}$	$e_{\text{inner}}/e_{\text{outer}}$
AR4JA Code	255.7/214.1	177.0/101.6
AR3A Code	255.9/152.4	89.1/11.01
Regular LDPC Code	255.3/144.5	85.4/1.65
IARA1 Code	255.0/136.8	77.1/0.94
IARA2 Code	256.0/134.9	72.4/0.80
IARA3 Code	254.9/150.3	88.2/7.6

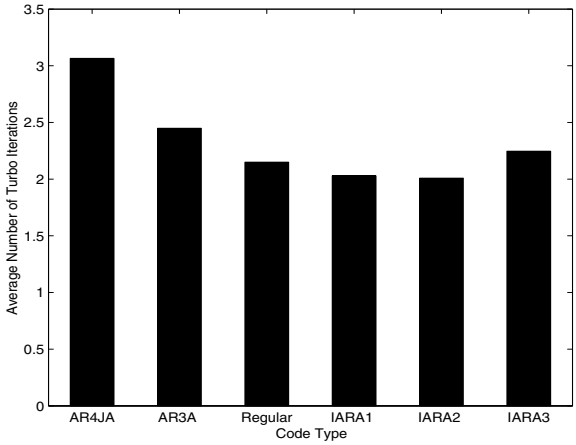


Fig. 8. Average number of turbo iterations required to successfully decode one code frame. The parameters used are $n = 3, R = 0.8, k = 4096$ and $E_b/N_0 = 4.8$ dB. An EPR4 channel is assumed.

We set the maximum number of turbo iterations to 10 and perform the simulations again at $E_b/N_0 = 4.8$ dB. For the successfully decoded frames, we evaluate the average number of turbo iterations used and show the results in Fig. 8. We observe that the convergence speeds of the codes shown in the figure are consistent with those shown in Table II.

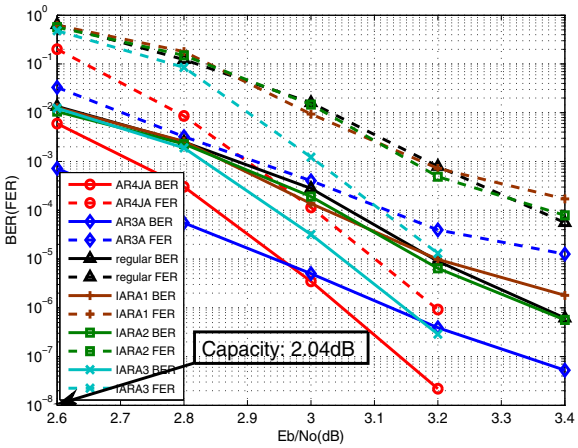


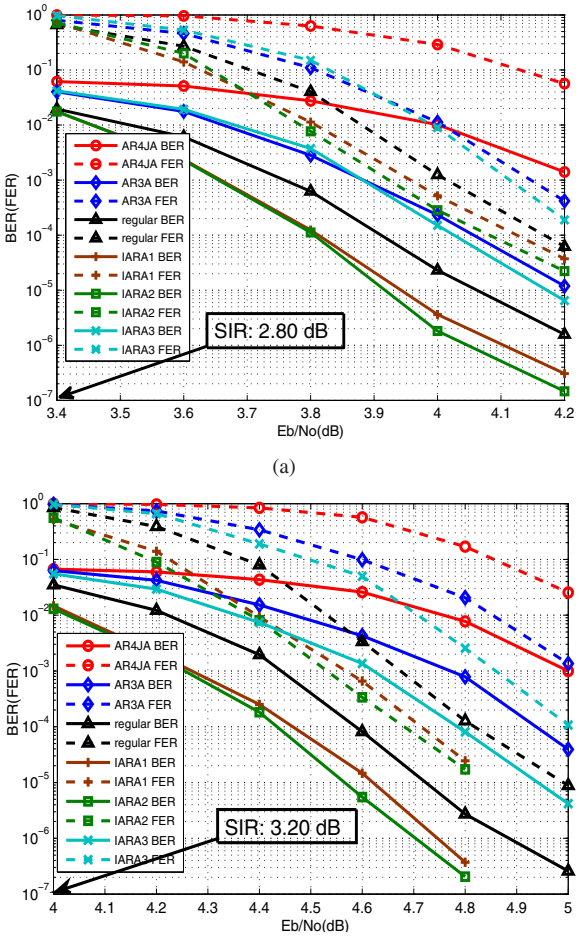
Fig. 9. Simulated BER and FER results of the AR3A, AR4JA, regular LDPC, proposed IARA1, proposed IARA2 and proposed IARA3 codes in an AWGN channel. BERs are shown by solid lines and FERs are represented by dotted lines. The channel capacity is 2.04 dB.

C. Simulated Error Rates

We simulate the bit-error-rate (BER) and frame-error-rate (FER) results of the codes as E_b/N_0 varies. Unless otherwise stated, we terminate the simulations after 500 bit errors are found at each E_b/N_0. Also, in the case of an AWGN channel, there is no inner detector and the LDPC decoder performs a maximum of 100 BP iterations for each code block.

Fig. 9 plots the BER and the FER curves of the codes in an AWGN channel. It can been seen that the error performance of the regular column-weight-3 code is almost the same as those of the IARA1 code and the IARA2 code. At a BER of 10^{-6}, both the AR3A code and the IARA3 code accomplish gains of more than 0.2 dB over the above three codes and the AR4JA outperforms the AR3A code and the IARA3 code by another 0.1 dB. The results verify the superb performance of the AR3A code and the AR4JA code in an AWGN channel [35], [36]. Note that at a $E_b/N_0 = 3.4$ dB, no errors are found out of the 5×10^6 transmitted blocks when the AR4JA code or the IARA3 code is used.

Fig. 10 plots the BER and the FER curves of the codes in a dicode channel and in an EPR4 channel. The maximum number of turbo iterations is set to 4. We can observe that the AR4JA code, the AR3A code and the IARA3 code, which provide the best error performance in an AWGN channel, become the three worst-performing codes in the PR channels. Referring to Fig. 10(a) that shows the error curves for a dicode channel, at $E_b/N_0 = 4.2$ dB, the AR4JA code, the AR3A code and the IARA3 code achieve BERs of 10^{-3}, 10^{-5} and 6×10^{-6}, respectively; while the regular LDPC code, the IARA1 code and the IARA2 code accomplish BERs of 10^{-6}, 3×10^{-7} and 10^{-7}, respectively. In the same figure, we also observe that at a BER of 10^{-5}, the regular LDPC code has a gain of 0.15 dB and 0.12 dB over the AR3A code and the IARA3 code, respectively; while the IARA1 code and the IARA2 code achieve another gain of 0.12 dB over the regular LDPC code. Moreover, a larger gain is expected at a lower BER. Similar observations are found in the results for the EPR4 channel. In addition, referring to Fig. 10(b), at a $E_b/N_0 = 5.0$ dB, no errors are found out of the 10^6 transmitted blocks when the IARA1 code and the IARA2 code are used.



(a)

(b)

Fig. 10. Simulated BER and FER results of the AR3A, AR4JA, regular LDPC, proposed IARA1, proposed IARA2 and proposed IARA3 codes in (a) a dicode channel and (b) an EPR4 channel. BERs are shown by solid lines and FERs are represented by dotted lines. The maximum number of turbo iterations is set to 4. The symmetric information rates (SIRs) for the dicode channel and the EPR4 channel are 2.80 dB and 3.20 dB, respectively.

Fig. 11 shows the BER and FER curves of the codes in an EPR4 channel where the maximum number of turbo iterations is increased from 4 to 8. Referring to this figure, the IARA1 code, the IARA2 code and the IARA3 code also perform better than the ARA3 code and the ARA4 code. Moreover, the error performance of the IARA3 code is still worse than that of the regular column-weight-3 code, which is outperformed by the IARA1 code and the IARA2 code. We observe that the error performance of the codes improves as the number of turbo iterations is increased from 4 to 8. While all codes produce lower BERs compared with using 4 turbo iterations, the relative performance among the codes remains the same. Hence the same conclusion is drawn compared with that using 4 turbo iterations.

In general, the codes, from the best to the worst error performance in a PR channel, are in the following order: (i) IARA2 code, (ii) IARA1 code, (iii) regular LDPC code, (iv) IARA3 code, (v) AR3A code and (vi) AR4JA code.

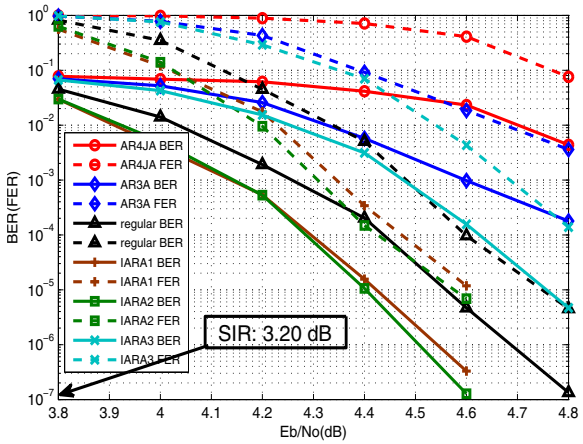


Fig. 11. Simulated BER and FER results of the AR3A, AR4JA, regular LDPC code, proposed IARA1, proposed IARA2 and proposed IARA3 codes in an EPR4 channel. BERs are shown by solid lines and FERs are represented by dotted lines. The maximum number of turbo iterations is set to 8. The SIR is 3.20 dB.

VI. Conclusions

The performance of protograph LDPC codes over partial response (PR) channels has been investigated through the finite-length EXIT algorithm in this paper. We observe that the conventional protograph LDPC codes, namely the AR4JA code and the AR3A code, do not perform well in PR channels and we propose a new design scheme for the protograph LDPC codes. Based on the design scheme, three new protograph LDPC codes, namely IARA1 code, IARA2 code and IARA3 code, are constructed. The EXIT-chart analysis, the convergence analysis and the bit-error-rate simulation have all shown that the IARA1 code, the IARA2 code and the IARA3 code outperform the AR4JA code and the AR3A code. We also conclude that two of the proposed protograph codes, i.e., the IARA1 code and the IARA2 code, provide excellent error performance in PR channels and they outperform (i) the AR4JA code and the AR3A code which are known to achieve very low error rates in AWGN channels and (ii) the regular column-weight-3 code which is one of the best-performing codes in PR channels.

Acknowledgements

The authors would like to thank Sijie Yang, Yong Li and Min Xiao for their valuable discussion and to thank Dr. Henry D. Pfister for providing us with the Shannon limit of PR channels. We would also like to thank the anonymous reviewers for their constructive comments that have helped improving the overall quality of the paper.

References

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

[2] D. MacKay and R. Neal, "Near Shannon limit performance of low density parity check codes," *Electron. Lett.*, vol. 33, no. 6, pp. 457–458, Mar. 1997.

[3] D. J. C. MacKay and M. S. Postol, "Weaknesses of Margulis and Ramanujan-Margulis low-density parity-check codes," *Electron. Notes Theor. Comput. Sci.*, vol. 74, pp. 97–104, 2003.

[4] T. Richardson and R. Urbanke, "The capacity of low-density parity-check codes under message-passing decoding," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 599–618, Feb. 2001.

[5] T. Richardson, M. Shokrollahi, and R. Urbanke, "Design of capacity-approaching irregular low-density parity-check codes," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 619–637, Feb. 2001.

[6] S. ten Brink, "Convergence behavior of iteratively decoded parallel concatenated codes," *IEEE Trans. Commun.*, vol. 49, no. 10, pp. 1727–1737, Oct. 2001.

[7] S. ten Brink, G. Kramer, and A. Ashikhmin, "Design of low-density parity-check codes for modulation and detection," *IEEE Trans. Commun.*, vol. 52, no. 4, pp. 670–678, Apr. 2004.

[8] H. Xiao and A. H. Banihashemi, "Improved progressive-edge-growth (PEG) construction of irregular LDPC codes," *IEEE Commun. Lett.*, vol. 8, no. 12, pp. 715–717, 2004.

[9] X. Y. Hu, E. Eleftheriou, and D. M. Arnold, "Regular and irregular progressive edge-growth Tanner graphs," *IEEE Trans. Inf. Theory*, vol. 51, no. 1, pp. 386–398, 2005.

[10] G. Richter and A. Hof, "On a construction method of irregular LDPC codes without small stopping sets," in *Proc. 2006 IEEE Int. Conf. Commun.*, pp. 1119–1124.

[11] X. Zheng, F. C. M. Lau, and C. K. Tse, "Constructing short-length irregular LDPC codes with low error floor," *IEEE Trans. Commun.*, vol. 58, no. 10, pp. 2823–2834, Oct. 2010.

[12] M. Yazdani, S. Hemati, and A. Banihashemi, "Improving belief propagation on graphs with cycles," *IEEE Commun. Lett.*, vol. 8, no. 1, pp. 57–59, Jan. 2004.

[13] M.-H. Taghavi and P. Siegel, "Adaptive methods for linear programming decoding," *IEEE Trans. Inf. Theory*, vol. 54, no. 12, pp. 5396–5410, Dec. 2008.

[14] N. Varnica, M. Fossorier, and A. Kavcic, "Augmented belief propagation decoding of low-density parity check codes," *IEEE Trans. Commun.*, vol. 55, no. 7, pp. 1308–1317, July 2007.

[15] C. Douillard, M. Jezequel, C. Berrou, A. Picart, P. Didier, and A. Glavieux, "Iterative correction of intersymbol interference: turbo-equalization," *European Trans. Telecommun.*, vol. 6, no. 5, pp. 507–511, Sep. 1995.

[16] D. Raphaeli and Y. Zarai, "Combined turbo equalization and turbo decoding," *IEEE Commun. Lett.*, vol. 2, no. 4, pp. 107–109, Apr. 1998.

[17] W. Ryan, "Performance of high rate turbo codes on a PR4-equalized magnetic recording channel," in *Proc. 1998 IEEE Int. Conf. Commun.*, vol. 2, pp. 947–951.

[18] ——, "Concatenated codes for class IV partial response channels," *IEEE Trans. Commun.*, vol. 49, no. 3, pp. 445–454, Mar. 2001.

[19] H. Alhussien, J. Park, and J. Moon, "Iterative decoding based on error pattern correction," *IEEE Trans. Magn.*, vol. 44, no. 1, pp. 181–186, Jan. 2008.

[20] F. Sun and T. Zhang, "Quasi-reduced-state soft-output Viterbi detector for magnetic recording read channel," *IEEE Trans. Magn.*, vol. 43, no. 10, pp. 3921–3924, Oct. 2007.

[21] K. Narayanan, "Effect of precoding on the convergence of turbo equalization for partial response channels," *IEEE J. Sel. Areas Commun.*, vol. 19, no. 4, pp. 686–698, Apr. 2001.

[22] J. L. Fan, A. Friedmann, E. Kurtas, and S. McLaughlin, "Low density parity check codes for magnetic recording," in *Proc. 1999 Allerton Conf. Commun., Control and Comput.*, pp. 1314–1323.

[23] H. Song, R. Todd, and J. Cruz, "Low density parity check codes for magnetic recording channels," *IEEE Trans. Magn.*, vol. 36, no. 5, pp. 2183–2186, Sep. 2000.

[24] X. Hu, Z. Li, B. Vijaya Kumar, and R. Barndt, "Error floor estimation of long LDPC codes on magnetic recording channels," *IEEE Trans. Magn.*, vol. 46, no. 6, pp. 1836–1839, June 2010.

[25] B. Kurkoski, P. Siegel, and J. Wolf, "Joint message-passing decoding of LDPC codes and partial-response channels," *IEEE Trans. Inf. Theory*, vol. 48, no. 6, pp. 1410–1422, June 2002.

[26] P. Pakzad and V. Anantharam, "Kikuchi approximation method for joint decoding of LDPC codes and partial-response channels," *IEEE Trans. Commun.*, vol. 54, no. 7, pp. 1149–1153, July 2006.

[27] Y. Han and W. Ryan, "Low-floor detection/decoding of LDPC-coded partial response channels," *IEEE J. Sel. Areas Commun.*, vol. 28, no. 2, pp. 252–260, Feb. 2010.

[28] A. Legg and B. Uchoa-Filho, "Graph-matched LDPC codes for partial-response channels," in *Proc. 2008 IEEE Int. Conf. Commun.*, pp. 2066–2070.

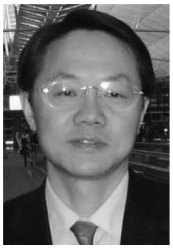
[29] N. Varnica and A. Kavcic, "Optimized low-density parity-check codes for partial response channels," *IEEE Commun. Lett.*, vol. 7, no. 4, pp. 168–170, Apr. 2003.

[30] M. Franceschini, G. Ferrari, and R. Raheli, "EXIT chart-based design of LDPC codes for inter-symbol interference channels," in *Proc. 2005 IST Mobile and Wireless Commun. Summit*.

[31] T. Richardson and R. Urbanke, "Multi-edge type LDPC codes." Available: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.106.7310
[32] J. Thorpe, " Low-density parity-check (LDPC) codes constructed from protographs," in *Proc. 2003 IPN Progress Report*, pp. 42–154.
[33] D. Divsalar, S. Dolinar, C. Jones, and K. Andrews, "Capacity-approaching protograph codes," *IEEE J. Sel. Areas Commun.*, vol. 27, no. 6, pp. 876–888, Aug. 2009.
[34] S. Abu-Surra, D. Divsalar, and W. Ryan, "On the existence of typical minimum distance for protograph-based LDPC codes," in *Proc. 2010 Inf. Theory Appl. Workshop*, pp. 1–7.
[35] D. Divsalar, C. Jones, S. Dolinar, and J. Thorpe, "Protograph based LDPC codes with minimum distance linearly growing with block size," in *Proc. 2005 IEEE Global Commun. Conf.*, Nov. 2005.
[36] A. Abbasfar, D. Divsalar, and K. Yao, "Accumulate-repeat-accumulate codes," *IEEE Trans. Commun.*, vol. 55, no. 4, pp. 692–702, Apr. 2007.
[37] P. Grosa, A. F. dos Santos, M. Lentmaier, W. Rave, and G. Fettweis, "Application of protograph-based LDPC codes for UWB short range communication," in *Proc. 2010 IEEE Int. Conf. Ultra-Wideband*, vol. 2, pp. 1–4.
[38] S. Yang, L. Wang, Y. Fang, and P. Chen, "Performance of improved AR3A code over EPR4 channel," in *Proc. 2011 Int. Conf. Comput. Research Development*, vol. 2, pp. 60–64.
[39] C. Di, D. Proietti, I. Telatar, T. Richardson, and R. Urbanke, "Finite-length analysis of low-density parity-check codes on the binary erasure channel," *IEEE Trans. Inf. Theory*, vol. 48, no. 6, pp. 1570–1579, June 2002.
[40] P. Tan and J. Li, "Finite-length extrinsic information transfer (EXIT) analysis for coded and precoded ISI channels," *IEEE Trans. Magn.*, vol. 44, no. 5, pp. 648–655, May 2008.
[41] J. Li, K. Narayanan, E. Kurtas, and C. Georghiades, "On the performance of high-rate TPC/SPC codes and LDPC codes over partial response channels," *IEEE Trans. Commun.*, vol. 50, no. 5, pp. 723–734, May 2002.
[42] L. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Trans. Inf. Theory*, vol. 20, no. 2, pp. 284–287, Mar. 1974.
[43] D. Fertonani, A. Barbieri, and G. Colavolpe, "Reduced-complexity BCJR algorithm for turbo equalization," *IEEE Trans. Commun.*, vol. 55, no. 12, pp. 2279–2287, Dec. 2007.
[44] A. Papoulis, *Probability, Random Variables, and Stochastic Processes*, 3rd edition. McGraw Hill Higher Education, 1991.
[45] H. Pfister, J. Soriaga, and P. Siegel, "On the achievable information rates of finite state ISI channels," in *Proc. 2001 IEEE Global Commun. Conf.*, vol. 5, pp. 2992–2996.
[46] G. Liva and M. Chiani, "Protograph LDPC codes design based on EXIT analysis," in *Proc. 2007 IEEE Global Commun. Conf.*, pp. 3250–3254.

Yi Fang received the B.Sc. degree in electronic engineering from East China Jiaotong University (ECJTU), Jiangxi, China, in 2008. He is currently working toward the Ph.D. degree in the Department of communication Engineering, Xiamen University, Fujian, China. From May 2012 to July 2012, he visited Hong Kong Polytechnic University, Hong Kong, as a Research Assistant. Mr. Yi Fang is serving as a reviewer of several international journals and conferences, including IEEE SIGNAL PROCESSING LETTERS, IEEE COMMUNICATIONS LETTERS, *Springer Circuits, Systems, and Signal Processing (CSSP)*, *IEEE Global Communications Conference (GLOBECOM)*, etc. His primary research interests are information theory, channel coding, chaotic modulations, UWB and MIMO communications.

Pingping Chen received the B.Sc. degree in Computer Software Engineering from FuZhou University (FZU), Fujian, China, in 2008. Now he is currently working toward the Ph.D. degree in the Department of Electronic Engineering, Xiamen University, Fujian, China. His primary research interests include channel coding, joint source and channel coding, network coding, and UWB communications.



Lin Wang (S'99–M'03–SM'09) received the B.Sc. degree in Mathematics (with first class honors) from the Chongqing Normal University, Chongqing, China, in 1984, the M.Sc. degree in Applied Mathematics from the Kunming University of Technology, Kunming, China, in 1988, and the Ph.D. degree in Electronics Engineering from the University of Electronic Science and Technology of China, Chengdu, China, in 2001.

From 1984 to 1986, he was Teaching Assistant in Mathematics Department of Chongqing Normal University. From 1989 to 2002, he was Teaching Assistant, Lecturer, and then Associate Professor in Applied Mathematics and Communication Engineering in the Chongqing University of Post and Tele-communication, Chongqing, China. From 1995 to 1996, he spent one year with Mathematics Department of the University of New England, Australia. In 2003, he spent three months as visiting researcher in the Center for Chaos and Complexity Networks at the City University of Hong Kong. Since 2003, he has been full-time Professor and Associate Dean in the School of Information Science and Engineering of Xiamen University, Xiamen, China. Recently he has become the editor of *ACTA Electronica Sinica* and Guest Associate Editor of *International Journal of Bifurcation and Chaos*. He holds 8 patents in the field of physical layer in digital communications and has published over 60 international journal and conference papers. His current research interests are in the area of channel coding, chaos modulation, and their applications for wireless communication and storage systems.

Francis C.M. Lau (M'93–SM'03) received the BEng (Hons) degree in electrical and electronic engineering and the PhD degree from King's College London, University of London, UK, in 1989 and 1993, respectively.

He is a Professor and Associate Head at the Department of Electronic and Information Engineering, The Hong Kong Polytechnic University, Hong Kong. He is also a senior member of IEEE. He is the co-author of *Chaos-Based Digital Communication Systems* (Heidelberg: Springer-Verlag, 2003) and *Digital Communications with Chaos: Multiple Access Techniques and Performance Evaluation* (Oxford: Elsevier, 2007). He is also a co-holder of three US patents, one pending US patent and one pending international patent. He has published over 200 papers. His main research interests include channel coding, cooperative networks, wireless sensor networks, chaos-based digital communications, applications of complex-network theories, and wireless communications.

He served as an associate editor for IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS II in 2004–2005 and IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS I in 2006–2007. He was also an associate editor of *Dynamics of Continuous, Discrete and Impulsive Systems, Series B* from 2004 to 2007, a co-guest editor of *Circuits, Systems and Signal Processing* for the special issue "Applications of Chaos in Communications" in 2005, and an associate editor for *IEICE Transactions* (Special Section on Recent Progress in Nonlinear Theory and Its Applications) in 2011. He has been a guest associate editor of *International Journal and Bifurcation and Chaos* since 2010 and an associate editor of *IEEE Circuits and Systems Magazine* since 2012.