View metadata, citation and similar papers at core.ac.uk
brought to you by CORE
provided by e-Prints Soton

Construction of Regular Quasi-Cyclic Protograph LDPC Codes Based on Vandermonde Matrices

Nicholas Bonello, Sheng Chen, and Lajos Hanzo

***Abstract*—In this paper, we investigate the attainable performance of quasi-cyclic (QC) protograph low-density parity-check (LDPC) codes for transmission over both additive white Gaussian noise and uncorrelated Rayleigh channels. The presented codes are constructed using the Vandermonde matrix and thus have a girth of at least six in their corresponding Tanner graph. Furthermore, they also benefit from both low-complexity encoding and decoding, low memory requirements, as well as hardware-friendly implementations. Our simulation results demonstrate that the advantages offered by this family of QC protograph LDPC codes accrue with no compromise in the attainable bit error ratio (BER) and block error ratio (BLER) performances. In fact, it is also shown that despite their implementational benefits, the proposed codes exhibit slight BER/BLER gains when compared to some of their more complex counterparts of the same length.**

***Index Terms*—Low-complexity low-density parity-check (LDPC) codes, protograph LDPC codes, quasi-cyclic (QC) LDPC codes, Vandermonde matrix (VM).**

I. Introduction

Following more than three decades of neglect, low-density parity-check (LDPC) codes [1], [2] are nowadays at the center of attention of the coding research community. This rekindled interest has been motivated by the outstanding performance demonstrated by turbo codes [3] that employ a similar soft-input–soft-output iterative decoding strategy [4].

In the context of LDPC codes, the relationship between the information bits and the redundant parity-check bits is described by a sparse parity-check matrix (PCM) or by the corresponding bipartite Tanner graph [5]. The design of an LDPC code is characterized by a range of contradictory design factors such as their bit error ratio (BER), their mathematical construction attributes, and their hardware complexity. Of prime concern is the BER performance exhibited by the code in both the "waterfall" and "error-floor" region. The mathematical construction attributes are related to the specific design of the PCM, which, generally speaking, can be constructed in either a pseudorandom [2] or a structured manner [6] (see also the references in [6]). It has been shown that pseudorandom codes [2], [7] exhibit excellent error-correction capabilities and, thus, are capable of operating close to the Shannon limit, particularly for high codeword lengths. However, such codes typically exhibit complex hardware implementations due to their high-complexity descriptions, and generally, their encoding complexity quadratically grows (or slower [8]) with the block length.

In this paper, we will pursue a more holistic LDPC code design approach and, thus, search for good LDPC codes, which strike an attractive tradeoff between the range of contradictory design factors. More explicitly, we investigate novel structured PCMs, which are designed based on Vandermonde-like block matrices [9]. The employment of Vandermonde block matrices was first proposed for classic Reed–Solomon codes and was also adopted for array codes in a conference paper by Fan [9]. Both Yang and Helleseth [10], as well as Mittelholzer [11], investigated the minimum distance bounds of array codes, whereas the rank of various LDPC code constructions based on Vandermonde matrices (VMs) was analytically determined by Gabidulin and Bossert in [12]. In [13], Pandya and Honary constructed variable-rate codes using VM-based LDPC codes having rates compliant with the DVB-S2 standard.

The aforementioned construction has the benefit of having a quasi-cyclic (QC) form [14]–[17] and, thus, significantly reduces the nonvolatile memory storage requirements. In addition, the encoding procedure can be implemented with the aid of shift registers, thus rendering the encoding complexity linear in the block length [18]. We further reduce the associated decoding complexity by invoking a so-called projected graph construction, which is also referred to as a "protograph" by Thorpe [19]. Protograph codes may also be considered as a subclass of Richardson's multiedge-type construction [20]. As a benefit of imposing a structural regularity, these codes can be decoded by means of a semiparallel architecture, as suggested by Lee *et al.* in [21], thus facilitating high-speed decoding. A number of optimized protographs have been designed for the additive white Gaussian noise (AWGN) channel by Thorpe [19] and Dolinar [22], all of which achieve high performance.

Against this backdrop, the contribution of this paper is to propose a PCM construction that is based on Vandermonde-like block matrices for the first time in the context of protograph LDPC arrangements. This results in the implementation-related advantages of combining the benefits of having a low-complexity QC encoder structure with a readily parallelizable protograph decoder structure. More explicitly, the resultant QC protograph LDPC codes have a girth of at least six in their corresponding Tanner graph and exhibit a low encoding and decoding complexity, as well as reduced memory requirements while facilitating hardware-friendly parallel implementations. We will compare our performance results to those attained by MacKay's codes [23] and to the codes generated using the extended bit filling (EBF) [24], as well as to the progressive edge-growth (PEG) [25] algorithms. Simulation results are provided for both AWGN and uncorrelated Rayleigh (UR) channels. It is demonstrated that the achievable performance is comparable to or slightly better than that exhibited by the higher complexity benchmarker codes of [2], [24], and [25] having the same lengths.

The structure of this paper is as follows. Sections II and III introduce the basic principles of LDPC codes and the protograph codes' construction. Our discourse continues with a description of the VM construction. The original PEG algorithm of [25] is then further developed in Section IV. Our simulation results are presented in Section V. Finally, Section VI is devoted to our conclusions.

II. Preliminaries

We consider a binary LDPC code defined by the null space of a low-density PCM matrix $\mathbf{H}$ constructed over GF(2). Then, assuming a full-rank PCM composed of M rows and N columns, the rate of this code becomes $R = 1 - M/N$. This can also be represented by means of a bipartite Tanner graph [5] consisting of M check nodes and N variable nodes. More explicitly, we consider a regular construction code having a uniform degree of edges emerging from each check and variable nodes. The variable and check nodes' degrees will be denoted by γ and ρ, which also correspond to the row and column weight of the PCM, respectively.

Manuscript received March 29, 2007; revised July 25, 2007, October 6, 2007, and November 5, 2007. This work was supported in part by the Engineering and Physical Sciences Research Council, U.K., and in part by the European Union under the auspices of the Newcom and Phoenix projects. The review of this paper was coordinated by Prof. H. Nguyen.

The authors are with the University of Southampton, SO17 1BJ Southampton, U.K. (e-mail: nb06r@ecs.soton.ac.uk; sqc@ecs.soton.ac.uk; lh@ecs.soton.ac.uk).

Digital Object Identifier 10.1109/TVT.2007.913180

0018-9545/$25.00 © 2008 IEEE

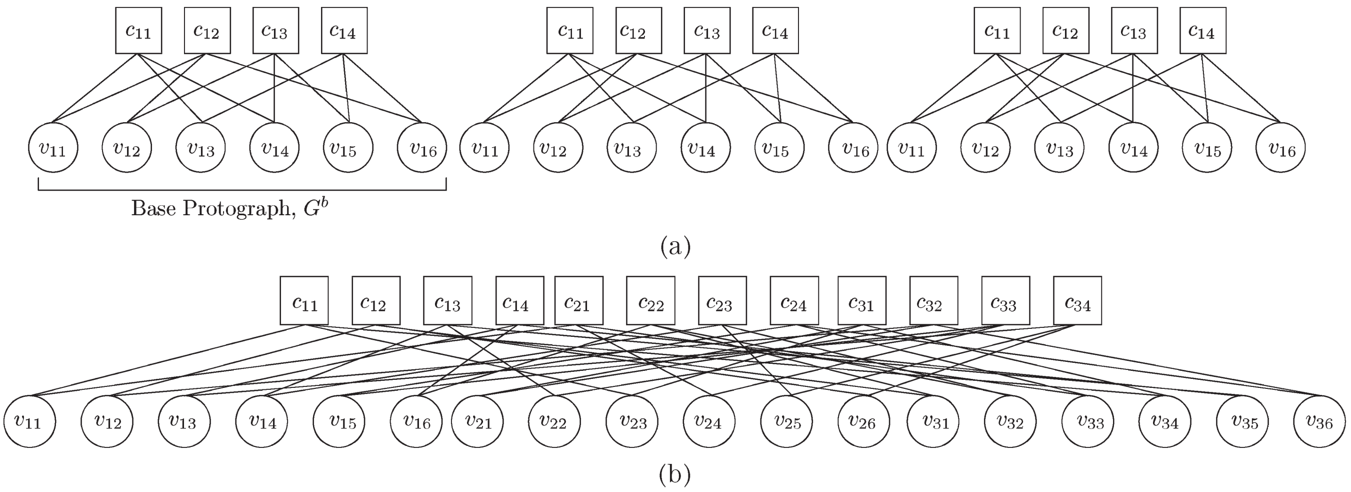
c_{11} c_{12} c_{13} c_{14}
v_{11} v_{12} v_{13} v_{14} v_{15} v_{16}
Base Protograph, G^b
(a)
c_{11} c_{12} c_{13} c_{14} c_{21} c_{22} c_{23} c_{24} c_{31} c_{32} c_{33} c_{34}
v_{11} v_{12} v_{13} v_{14} v_{15} v_{16} v_{21} v_{22} v_{23} v_{24} v_{25} v_{26} v_{31} v_{32} v_{33} v_{34} v_{35} v_{36}
(b)

Fig. 1. (a) Base protograph is replicated by a factor J, in this case, $J = 3$. (b) Construction of the derived graph is obtained by permuting the edges between the check and variable nodes of the J copies of the base protograph. The permutations are performed in a way to maximize the girth while exhibiting a QC construction constrained by the VM-based protograph.

LDPC codes are typically decoded using the sum–product algorithm (SPA) [26], where messages or "beliefs" [27] are exchanged between the nodes residing at both sides of the graph. The independence of these messages is characterized by the length of the shortest cycle on the graph, which is typically referred to as the girth g. Specifically, Gallager demonstrated in [1] that the number of independent iterations T, i.e., the iterations that provide valuable extrinsic information and, hence, a useful iteration gain, is bounded by $T < g/4 \leq T+1$. Clearly, for the girth to be high, the block length also has to be sufficiently high. The loose lower bound on the required block length is given by [1]: $N \geq 1 + \sum_{k=2}^{x+1} \gamma(\gamma-1)^{k-2}(\rho-1)^{k-1}$ for a specific girth $g = 4x+2$, where x is an integer. By contrast, we have [1] $N \geq \sum_{k=1}^{x+1} \rho[(\gamma-1)(\rho-1)]^{(k-1)}$ for $g = 4x$. Furthermore, we only consider codes having $\gamma \geq 3$, and hence, the resultant minimum distance grows linearly, instead of logarithmically, with the block length [1].

III. Protograph LDPC Code Construction

The construction of a protograph code, which is illustrated in Fig. 1, can be described in two main steps [19].

1) Determine the base protograph, which is typically a graph with a relatively low number of nodes, and replicate this graph J times.
2) Permute the edges of the nodes in the J replicas of the base protograph to obtain the resultant graph.

Consider the base protograph G^b described by the set of check nodes $C^b = \{c_{ji} : j = 1; i = 1, \ldots, M^b\}$, the set of variable nodes $V^b = \{v_{ji} : j = 1; i = 1, \ldots, N^b\}$, and the set of edges E^b, where $|E^b| = M^b\rho = N^b\gamma$. We denote the number of check and variable nodes on the base protograph by M^b and V^b, respectively. The value of $j = 1$ refers to the base protograph. The base protograph will therefore have the corresponding base PCM of size $(M^b \times V^b)$. After replicating G^b J times, we obtain the resultant graph of the protograph code G' defined by the sets C', V', and E', where each set has a size J times larger than the corresponding sets in the base protograph. The permutations of the nodes' edges in the graph derived obey certain constraints, which will be discussed in more detail in Section IV.

A. VM-Based LDPC Code Construction

Because we want to impose a QC structure on our protograph code, we opt for constructing the QC base protograph from the VM [9] construction. Let $\mathbf{I}_q$ represent a $(q \times q)$ identity matrix where q is either larger than the row, as well as the column weight, and it is a relative prime with respect to all the numbers less than ρ or else obeys $q > (\rho-1)(\gamma-1)$. We also construct the permutation matrix $\mathbf{P}_q$ having elements of p_{mn}, $0 \leq m < q$ and $0 \leq n < q$, which is defined by [28]

$$p_{mn} = \begin{cases} 1, & \text{if } m = (n-1) \bmod q \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

where $a \bmod b$ represents the modulus after the division of a by b. For the sake of simplifying our analysis, we consider the example of $q = 4$, where the permutation matrices $\mathbf{P}_q$, $\mathbf{P}_q^2$, and $\mathbf{P}_q^3$ are given by

$$\begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

and

$$\mathbf{P}_q^x = \begin{cases} \mathbf{I}_q, & \text{if } x \bmod q = 0 \\ \mathbf{P}_q^{x \bmod q}, & \text{otherwise.} \end{cases} \tag{2}$$

Then, the constructed VM-based sparse PCM for the base protograph is formulated by [28]

$$\mathbf{H}^b = \begin{bmatrix} \mathbf{I}_q & \mathbf{I}_q & \mathbf{I}_q & \cdots & \mathbf{I}_q \\ \mathbf{I}_q & \mathbf{P}_q & \mathbf{P}_q^2 & \cdots & \mathbf{P}_q^{\rho-1} \\ \mathbf{I}_q & \mathbf{P}_q^2 & \mathbf{P}_q^4 & \cdots & \mathbf{P}_q^{2(\rho-1)} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{I}_q & \mathbf{P}_q^{(\gamma-1)} & \mathbf{P}_q^{2(\gamma-1)} & \cdots & \mathbf{P}_q^{(\gamma-1)(\rho-1)} \end{bmatrix}. \tag{3}$$

The PCM $\mathbf{H^b}$ of size $(\gamma q \times \rho q)$ will describe the null space for a base protograph LDPC code defined by the block length $N^b = \rho \times q$

and rate $R \geq 1 - \gamma/\rho$. The permutation matrix $\mathbf{P}_q$ is essentially constructed from an appropriate cyclic shift of the identity matrix $\mathbf{I}_q$. The restrictions imposed on the parameters q, ρ, and γ ensure that no permutation matrix $\mathbf{P}_q^x$, $0 \leq x \leq (\gamma-1)(\rho-1)$ is repeated in the same row or column of the permutation matrices.

IV. Modifications of the PEG Algorithm

The permutation pattern of the nodes' edges in the derived graph was determined using a modified version of the PEG algorithm. Whilst we still maintain the elegant characteristics of the PEG with regard to maximizing the girth of the graph and the minimum distance of the code [25], we impose two additional constraints. The first constraint ensures that the derived graph has the same structure as the base protograph, whereas the second ascertains that the derived graph is also QC. The procedure that was used is summarized in Algorithm 1.

```
Algorithm 1. The modified PEG.
    input: $M^b$, $N^b$, $J$, $q$, $\gamma$
    output: $C'^{ji}$ for $j = 1, \ldots, J$ and $i = 1, \ldots, M^b, G'$
1  Lines 2–21 determine the forbidden set of check nodes based on the VM PCM of the base protograph (Constraint 1).
2  for kth variable node ← 1 to $N^b J$ do
3      $k \leftarrow$ (kth variable node) mod $N^b$, $n \leftarrow 0$, $C'_{\text{tmp3}} = \emptyset$
4      if $k \leq q$ then
5          $C'^{ji} = \{c_{ji} : j = 1, \ldots, J; i = k, k+q, k+2q, \ldots, M^b\}$
6      else
7          $x \leftarrow$ (integer value of) $[(k-1)/q]$, $r \leftarrow 1$
8          $C'_{\text{tmp1}} = \{c_{ji} : j = 1, \ldots, J; i = n+1\}$
9          for $y \leftarrow x$ to $x(\gamma-1)$, (step: $y \leftarrow 2 \times$ previous value of $y$) do
10             $C'_{\text{tmp2}} = \{c_{ji} : j=1, \ldots, J; i=(rq+1)+(n-y) \bmod q\}$
11             $C'_{\text{tmp3}} = C'_{\text{tmp2}} \cup$ (previous $C'_{\text{tmp3}}$)
12             $r \leftarrow r+1$
13         end
14         $C'^{ji} = C'_{\text{tmp1}} \cup C'_{\text{tmp3}}, \overline{C'^{ji}} = C' \backslash C'^{ji}$
15         if $x >$ previous value of $x$ then
16             $n \leftarrow 0$
17         else
18             $n \leftarrow n+1$
19         end
20         foreach $c_{ji} \in \overline{C'^{ji}}$ do Store the number of connections under the current graph construction and then set their number of connections to $\rho$
21     end
22     if $j > 1$ then
23         Set the number of connections of the check nodes connected with variable nodes $v_{ji}$, with $1 \geq j \leq$ (current $j$) $-1$ and $i = k$ to $\rho$
24     end
25     Starting the modified PEG algorithm.
26     for connection ← 1 to $\gamma$ do
27         if connection = 1 then
28         Similar to PEG [25] with the chosen $c_{ji} \in C'^{ji}$
29         else
30             Similar to PEG [25] but the chosen $c_{ji} \in C'^{ji}$ must have the lowest degree (under the current graph construction) and be the nearest to the selected $c_{j(i-1)}$ for the same connection (Constraint 2).
31         end
32     end
33     foreach $c_{ji} \in C'$ do Restore the original number of connections.
34 end
```

It can be observed from Fig. 1(b) that the permutations of the node's edges follow a particular pattern, which is governed by the PCM of the base protograph. For example, the edges emerging from the variable nodes v_{j1}, $j = 1, \ldots, 3$, are only connected to the check nodes c_{ji} associated with $i = 1, 2$ and $j = 1, \ldots, 3$. This effectively imposes the structure of the base protograph on the graph derived. For each variable node v_{ji}, $j = 1, \ldots, J$ and $i = 1, \ldots, N^b$, we define the set of "allowed" checks C'^{ji} and the set of "forbidden" checks by the complementary set $\overline{C'^{ji}} = C' \backslash C'^{ji}$, i.e., the set of elements in C' but not in C'^{ji}. It is only necessary to calculate N^b different sets because the sets repeat every N^b variable nodes. Then, for each v_{ji}, the algorithm selects that check node in the specific C'^{ji} set having the lowest number of edges emerging from it under the current graph construction. On the other hand, we set the number of edges of every check node in $\overline{C'^{ji}}$ equal to ρ, which corresponds to the maximum number of connections a check node is allowed to have. In this manner, it is guaranteed that no connection between a variable node and a check node in the corresponding set $\overline{C'^{ji}}$ will be established.

However, by imposing only this constraint on the original PEG, the resultant graph will be acyclic (AC). This is due to the fact that the PEG [25] will randomly select the check nodes[1] if multiple choices are available. Therefore, we further restrict the algorithm to choose a check node $c_{ji} \in C'^{ji}$, which is the closest to the previously selected $c_{j(i-1)}$, for the same connection.[2] Because the base protograph was chosen to be QC, the algorithm is always capable of choosing that check node, which still retains the structural characteristics of the base, and so, the resulting protograph code will also be QC. This modification will lead to similar results to those attained by the QC-PEG proposed by Li and Kumar in [29], where, in our case, the "QC constraint" [29] is imposed by the base protograph PCM. When compared to the PEG algorithm, as originally proposed by Hu *et al.* [25], the modified algorithm is capable of reducing the size of the set of allowed checks from being governed by the binomial coefficient $\binom{N}{\gamma}$, to $\binom{J\gamma}{\gamma}$, where $N = JM^{bn}$.

V. Results and Discussion

The results presented in this section were obtained using binary phase-shift keying (BPSK) modulation when communicating over the AWGN and UR channels and using a maximum of $I = 100$ decoding iterations of the SPA. We will consider codes having $\gamma = 3$, a block length of N ranging from 200 to 3060, and code rates R spanning from 0.4 to 0.8.[3] We compare both the achievable block error ratio (BLER) and BER performances for transmission over both AWGN and UR channels for five different code constructions, namely those of MacKay [23], the EBF [24], the PEG [25], and the AC, as well as of QC protograph codes. The AC protograph code was constructed by considering only the first constraint in the modified PEG. We will appropriately distinguish between the codes using the notation (N, K). The error bars shown on the BLER curves are

[1] For the modified PEG, these check nodes will be members of the set C'^{ji}.

[2] The total number of connections for each variable node is equal to γ.

[3] The row weight of the LDPC codes having rates 0.4, 0.5, 0.625, and 0.8 are 5, 6, 8, and 15, respectively.

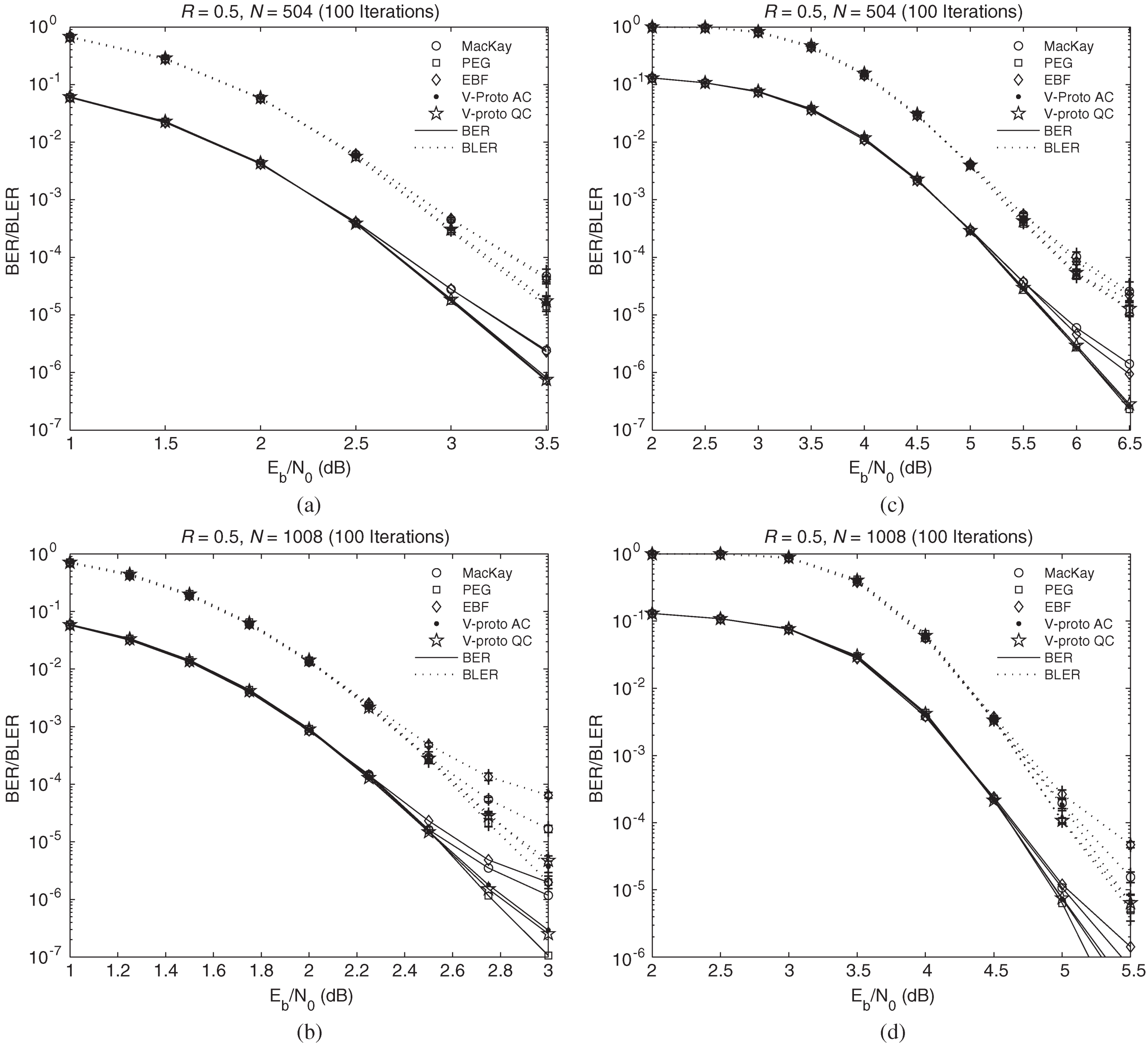

Fig. 2. BER and BLER performance comparison of $R = 0.5$ LDPC codes with $N = 504$ and $N = 1008$ and a maximum of $I = 100$ decoder iterations when transmitting over the AWGN and UR channels using BPSK modulation. The error bars shown on the BLER curves are associated with a 95% confidence level. (a) $N = 504$, AWGN channel. (b) $N = 1008$, AGWN channel. (c) $N = 504$, UR channel. (d) $N = 1008$, UR channel.

associated with a 95% confidence level, and it was ensured that at least 100 block errors were collected for every point on the simulation curve.

The BLER and BER performance results over the AWGN channel recorded for the (504, 252) and (1008, 504) codes are illustrated in Fig. 2(a) and (b), respectively. The (504, 252) protograph codes were constructed from 12 replicas of VM-based protographs using $q = 7$. In a similar manner, 14 replicas of VM-based protographs having a permutation matrix of size (12×12) were used for the protograph LDPC codes having a length of $N = 1008$. It can be observed that the proposed QC protograph code still exhibits a performance gain of about 0.2 dB over the randomly generated MacKay code at a BER of 10^{-6}. There is only a 0.06-dB loss in the performance of the QC protograph code when compared to the significantly more complex unstructured PEG code, which is deemed to have the best performance for the transmission of short blocks over the AWGN channel. Therefore, our results demonstrate that the proposed QC codes having a protograph structure and low-complexity hardware-friendly implementations exhibit a BER/BLER performance that is comparable to or even slightly better than that of their more complex unstructured counterparts. Similar BLER and BER performance trends were observed for the UR channel, as demonstrated in Fig. 2(c) and (d).

For the sake of completeness, we also investigated the performance of QC protograph codes having rates of 0.4, 0.625, and 0.8, as well as both shorter and longer blocklengths. Our simulation results, which are not shown in this correspondence owing to space limitations, showed that the performance of the protograph codes is always comparable to that exhibited by the other benchmarker codes. A slight degradation was manifested by the QC protograph codes for high code rates and very short block lengths because the constraints described in Section III-A could not be satisfied.

TABLE I
SUMMARY OF THE CHARACTERISTICS OF THE CONSIDERED CODES. THE COMPUTATIONAL DECODING COMPLEXITY Δ (MESSAGE UPDATES/DECODED BIT) IS MEASURED FOR THE (1008, 504) CODES

Complexity/Performance Criteria		MacKay	PEG	EBF	Proto AC	Proto QC
Desirable Encoder Characteristics	Simple description and MAG					■
	Complexity linear with N		■			■
	Simple hardware implementation					■
Desirable Decoder Characteristics	Reduced logic depth	■	■	■	■	■
	Simple parallel architecture				■	■
	Simple MAG and on-chip interconn.					■
Δ	AWGN at E_b/N_0 = 3 dB with I = 50	40	39	41	40	39
	UR at E_b/N_0 = 4.5 dB with I = 50	58	56	59	57	57

A. Encoder and Decoder Complexity

In this section, we provide a more comprehensive comparison of the different code constructions that were considered by taking into account the encoder and decoder complexity. We employ a similar benchmarking technique to that used in [30], where the metrics used for comparison are based on an amalgam of the desirable encoder and decoder characteristics. The former include a low-complexity description due to structured row–column connections and simple memory address generation (MAG), the linear dependence of the encoding complexity on the codeword length, and a hardware implementation based on simple components.

With regard to attractive decoder characteristics, we are concerned with the reduction of MAG and on-chip wire interconnections, the reduced logic depth,[4] and the ability to use parallel decoding architectures for systolic-array-type implementations. We also evaluate the decoder's computational complexity expressed in terms of the number of message-passing updates per decoded bit, which is given by $\Delta = \bar{i}|E'|/K$ [30], where $\bar{i}$ represents the average number of iterations required for finding a legitimate codeword at a particular E_b/N_0 value.

A summary of these measures recorded for each considered code are summarized in Table I. It can be observed in Table I that the encoder structure is quite complex for the majority of the five codes considered. Only the PEG and the QC protograph codes have linearly increasing complexity as a function of the codeword length.[5] The QC protograph's encoder can also be implemented using a simple linear shift-register circuit of length K, and therefore, the encoder only requires $r(N - K)$ binary operations, where r is one less than the row weight of the generator matrix. By contrast, the remaining codes must be encoded by means of sparse matrix multiplications that require $(N - K)(2K - 1)$ binary operations [31].

As far as the decoder's complexity is concerned, all the five code constructions score at least one point due to their low logic depth that accrues from using small values of ρ and γ. However, the lowest decoding complexity can only be attained using QC protographs codes. The AC protograph code does benefit from facilitating parallel hardware implementations, but it suffers from having a high-complexity description due to the random PEG permutations. Therefore, its implementation still relies on inflexible hard-wired connections or on lookup tables that require a large amount of memory. By contrast, memory shifts corresponding to the cyclic PCM structure can be used to address the messages exchanged between the nodes of QC protograph. Several decoders for QC codes have been proposed, particularly that of Chen and Parhi [32], which is capable of doubling the decoding throughput (assuming a dual-port memory) when compared to the decoding of randomly constructed codes by overlapping the variable and check node updates.

VI. SUMMARY AND CONCLUSION

In this paper, we have proposed the construction of protograph LDPC codes based on QC VMs. These codes benefit from low-complexity encoding and decoding implementations due to their semiparallel architectures. We have investigated their BLER and BER performances for transmission over both AWGN and UR channels and for various rates and block lengths. Explicitly, our experimental results demonstrate that the performance of these protograph codes is comparable to or slightly better than that exhibited by the higher complexity benchmarker codes. Therefore, it can be concluded that the advantages offered by the family of QC protograph LDPC codes accrue without any compromise in the attainable BLER and BER performances.

[4] The logic depth is directly related to the depth of the graph tree spreading from a variable node v_{ji}, $j = 1, \ldots, J$ and $i = 1, \ldots, N^b$.

[5] The PEG codes that were simulated cannot be decoded in linear time; however, linear-time encoding for PEG codes is possible using "zigzag" [25] connections. On the other hand, the MacKay and EBF codes can only be encoded using the near-linear encoding scheme, as proposed by Richardson and Urbanke [8].

REFERENCES

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

[2] D. J. C. MacKay, "Good error-correcting codes based on very sparse matrices," *IEEE Trans. Inf. Theory*, vol. 45, no. 2, pp. 399–431, Mar. 1999.

[3] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: Turbo-codes. 1," in *Proc. Int. Conf. Commun., Geneva Tech. Program*, Geneva, Switzerland, May 23–26, 1993, vol. 2, pp. 1064–1070.

[4] R. J. McEliece, D. J. C. MacKay, and J.-F. Cheng, "Turbo decoding as an instance of Pearl's belief propagation algorithm," *IEEE J. Sel. Areas Commun.*, vol. 16, no. 2, pp. 140–152, Feb. 1998.

[5] R. M. Tanner, "A recursive approach to low complexity codes," *IEEE Trans. Inf. Theory*, vol. IT-27, no. 5, pp. 533–547, Sep. 1981.

[6] J. M. F. Moura, J. Lu, and H. Zhang, "Structured low-density parity-check codes," *IEEE Signal Process. Mag.*, vol. 21, no. 1, pp. 42–55, Jan. 2004.

[7] S.-Y. Chung, G. D. J. Forney, T. J. Richardson, and R. Urbanke, "On the design of low-density parity-check codes within 0.0045 db of the Shannon limit," *IEEE Commun. Lett.*, vol. 5, no. 2, pp. 58–60, Feb. 2001.

[8] T. Richardson and R. Urbanke, "Efficient encoding of low-density parity check codes," *IEEE Trans. Commun.*, vol. 47, no. 6, pp. 808–821, Feb. 2001.

[9] J. L. Fan, "Array codes as low density parity check codes," in *Proc. 2nd Int. Symp. Turbo Codes*, Brest, France, 2000, vol. 3, pp. 543–546.

[10] K. Yang and T. Helleseth, "On the minimum distance of array codes as LDPC codes," *IEEE Trans. Inf. Theory*, vol. 49, no. 12, pp. 3268–3271, Dec. 2003.

[11] T. Mittelholzer, "Efficient encoding and minimum distance bounds of Reed–Solomon-type array codes," in *Proc. IEEE Int. Symp. Inf. Theory*, 2002, p. 282.

[12] E. M. Gabidulin and M. Bossert, "On the rank of LDPC matrices constructed by Vandermonde matrices and RS codes," in *Proc. IEEE Int. Symp. Inf. Theory*, Seattle, WA, Jul. 2006, pp. 861–865.

[13] N. Pandya and B. Honary, "Variable-rate LDPC codes based on structured matrices for DVB-S2 applications," in *Proc. 8th Int. Symp. Commun. Theory Appl.*, Ambleside, U.K., 2005, pp. 368–373.

[14] R. M. Tanner, "On quasi-cyclic repeat-accumulate codes," in *Proc. 37th Annu. Allerton Conf. Commun., Control Comput.*, Monticello, IL, Sep. 1999, pp. 249–259.

[15] R. M. Tanner, "Spectral graphs for quasi-cyclic LDPC codes," in *Proc. IEEE Int. Symp. Inf. Theory*, Washington, DC, Jun. 2001, p. 226.

[16] L. Chen, J. Xu, I. Djurdjevic, and S. Lin, "Near-Shannon-limit quasi-cyclic low-density parity-check codes," *IEEE Trans. Commun.*, vol. 52, no. 7, pp. 1038–1042, Jul. 2004.

[17] M. P. C. Fossorier, "Quasicyclic low-density parity-check codes from circulant permutation matrices," *IEEE Trans. Inf. Theory*, vol. 50, no. 8, pp. 1788–1793, Aug. 2004.

[18] Z. Li, L. Chen, L. Zeng, S. Lin, and W. H. Fong, "Efficient encoding of quasi-cyclic low-density parity-check codes," *IEEE Trans. Commun.*, vol. 54, no. 1, pp. 71–81, Jan. 2006.

[19] J. Thorpe, "Low-density parity-check (LDPC) codes constructed from protographs," JPL, Pasadena, CA, IPN Progress Rep. 42-154, Aug. 2003.

[20] T. Richardson and V. Novichkov, "Methods and apparatus for decoding LDPC codes," U.S. Patent 6 633 856, Oct. 14, 2003.

[21] J. K. S. Lee, B. Lee, J. Thorpe, K. Andrews, S. Dolinar, and J. Hamkins, "A scalable architecture of a structured LDPC decoder," in *Proc. IEEE Int. Symp. Inf. Theory*, Jun. 27–Jul. 2, 2004, p. 292.

[22] S. Dolinar, "A rate-compatible family of protograph-based LDPC codes built by expurgation and lengthening," in *Proc. IEEE Int. Symp. Inf. Theory*, Sep. 4–9, 2005, pp. 1627–1631.

[23] D. MacKay, *Online Database of Low-Density Parity-Check Codes*. [Online]. Available: wol.ra.phy.cam.ac.uk/mackay/codes/data.html

[24] J. Campello and D. S. Modha, "Extended bit-filling and LDPC code design," in *Proc. IEEE Global Telecommun. Conf.*, San Antonio, TX, Nov. 25–29, 2001, vol. 2, pp. 985–989.

[25] 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, Jan. 2005.

[26] F. R. Kschischang, B. J. Frey, and H. A. Loeliger, "Factor graphs and the sum–product algorithm," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 498–519, Feb. 2001.

[27] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. San Mateo, CA: Morgan Kaufman, 1988.

[28] B. Ammar, "Error protection and security for data transmission," Ph.D. dissertion, Univ. Lancaster, Lancaster, U.K., 2004.

[29] Z. Li and B. V. K. V. Kumar, "A class of good quasi-cyclic low-density parity check codes based on progressive edge growth graph," in *Proc. 38th Asilomar Conf. Signals, Syst. Comput.*, Nov. 7–10, 2004, vol. 2, pp. 1990–1994.

[30] D. D. K. Andrews and S. Dolinar, "Design of low-density parity-check (LDPC) codes for deep-space applications," JPL, Pasadena, CA, IPN Progress Rep. 42-159, Nov. 2004.

[31] S. J. Johnson and S. R. Weller, "A family of irregular LDPC codes with low encoding complexity," *IEEE Commun. Lett.*, vol. 7, no. 2, pp. 79–81, Feb. 2003.

[32] Y. Chen and K. K. Parhi, "Overlapped message passing for quasi-cyclic low-density parity check codes," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 51, no. 6, pp. 1106–1113, Jun. 2004.

Performance Analysis of Transmit Diversity Systems With Antenna Replacement

Seyeong Choi, *Member, IEEE*,
Hong-Chuan Yang, *Senior Member, IEEE*,
and Young-Chai Ko, *Senior Member, IEEE*

***Abstract*—We propose a new closed-loop transmit diversity scheme for multiple-input–multiple-output (MIMO) diversity systems based on orthogonal space–time block coding (OSTBC). The receiver of the proposed scheme checks the output signal-to-noise ratio (SNR) of the space–time decoder against an output threshold and requests the transmitter to replace the transmit antenna resulting in the poorest path with an unused antenna if the output SNR is below the threshold. We provide some interesting statistical analysis and obtain closed-form expressions for the cumulative distribution function (cdf), the probability density function (pdf), and the moment-generating function of the received SNR. We show through numerical examples that the proposed scheme offers a significant performance gain with a very minimal feedback load over existing open-loop MIMO diversity systems, and for a properly chosen threshold, its performance is commensurate with a more complicated generalized-selection-combining-based transmit diversity system while requiring a much smaller feedback load.**

***Index Terms*—Diversity techniques, fading channels, multiple-input–multiple-output (MIMO), performance analysis, switched diversity, transmit diversity.**

I. Introduction

Future wireless communication systems should support not only high spectral efficiency but good link reliability as well. Antenna diversity systems with multiple transmit and/or receive antennas can significantly increase the reliability of wireless fading channels [1]. Well-known receive diversity combining techniques include maximal ratio combining (MRC), equal-gain combining, selection combining (SC), and switch-and-stay combining [2], [3]. Meanwhile, the main advantage of transmit diversity is that diversity gain can be obtained for downlink transmission without implementing multiple antennas at the mobile station. Transmit diversity systems based on an orthogonal space–time block code (OSTBC) have received considerable interest [4]–[7]. Its two-antenna special case, i.e., the so-called Alamouti scheme [8], has been incorporated into third-generation standards.

In general, OSTBC systems can achieve full diversity gain with simple linear processing at the receiver and without any knowledge of the fading channels at the transmitter side. However, when the number of transmit antennas is greater than two, the OSTBC will suffer a rate loss. Moreover, the more the transmit antennas are used, the larger the signal-to-noise ratio (SNR) loss occurs due to power spreading over

Manuscript received August 17, 2006; revised February 7, 2007, March 20, 2007, and March 21, 2007. This work was supported by the Ministry of Knowledge Economy, Korea, under the Information Technology Research Center (ITRC) support program supervised by the Institute of Information Technology Advancement (IITA) under Grant IITA-2008-C1090-0801-0037. This paper was presented in part at the IEEE International Symposium on Wireless Pervasive Computing, Phuket, Thailand, January 2006. The review of this paper was coordinated by Prof. N. Arumugam.

S. Choi is with the Department of Electrical Engineering, Texas A&M University at Qatar, Doha, Qatar (e-mail: seyeong.choi@qatar.tamu.edu).

H.-C. Yang is with the Department of Electrical and Computer Engineering, University of Victoria, BC V8W 3P6, Canada (e-mail: hyang@ece.uvic.ca).

Y.-C. Ko is with the School of Electrical Engineering, Korea University, Seoul 136-701, Korea (e-mail: koyc@korea.ac.kr).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TVT.2007.912957

0018-9545/$25.00 © 2008 IEEE