

Multilevel-Structured Low-Density Parity-Check Codes for AWGN and Rayleigh Channels

Nicholas Bonello, Sheng Chen, *Fellow, IEEE*, and Lajos Hanzo, *Fellow, IEEE*

Abstract—We propose a novel class of protograph low-density parity-check (LDPC) codes having a combinatorial rather than a random structure, which are termed multilevel-structured (MLS) LDPC codes. It is demonstrated that they possess a strikingly simple structure and, thus, benefit from reduced storage requirements, hardware-friendly implementations, and low-complexity encoding. Our simulation results provided for both additive white Gaussian noise (AWGN) and uncorrelated Rayleigh (UR) channels demonstrate that these advantages accrue without compromising the attainable bit error ratio (BER) and block error ratio (BLER) performance, when compared with their previously proposed more complex random-construction-based counterparts, as well as with other structured codes of the same length.

Index Terms—Low-complexity low-density parity check (LDPC), multilevel structured (MLS), protograph LDPC codes, quasi-cyclic (QC).

I. INTRODUCTION

LOW-DENSITY parity-check (LDPC) codes [1], [2] have attracted substantial interest in the coding research community. It is widely recognized that their soft-input–soft-output iterative decoding strategy is capable of exhibiting a performance close to the Shannon limit [3]–[5] when sufficiently high codeword lengths are considered. Moreover, the sparseness of their parity-check matrix (PCM) ensures that this performance is achieved at an acceptable decoder complexity.

The pseudorandom allocation of the logical one values in the PCM was considered to be an important feature in LDPC design, since it was demonstrated in [2]–[6] that these codes exhibit excellent error-correction capabilities. Other algorithmic constructions tend to focus on a particular attribute of the associated graph, such as the girth [7], [8] or the employment of cycle conditioning [9]–[11]. However, the resultant PCM structure remains random and, therefore, possesses no compact description that would facilitate their efficient implementation. For this reason, various structured constructions have been investigated [12], such as those using geometric approaches [13] or combinatorial designs [14]. The latter family includes different balanced incomplete block design [15] classes such as the Steiner and Kirkman triple systems [16], [17], Bose designs [18], and the so-called anti-Pasch [19] techniques. Most of these

structured constructions are cyclic or quasi-cyclic (QC) [20]–[23], and therefore, their encoding can be implemented with the aid of linear shift registers, thus rendering the encoding complexity to be a linear function of the block length [24].

The iterative decoder of LDPC codes can be regarded as a serial concatenation of two constituent decoders separated by an edge interleaver, which defines the edge interconnections between the nodes involved in the parity-check equations, as governed by the code's PCM or by the corresponding bipartite Tanner graph [25]. This effectively means that each nonzero position in the PCM or, equivalently, each edge of the Tanner graph represents an entry either in a large lookup table (LUT) or in a large-area hardwired mesh of interconnections on a chip. The complexity of the code's description tends to linearly increase with the block length, and again, it is essentially determined by the specific design of the PCM.

Multilevel-structured¹ (MLS) LDPC codes attempt to strike a balance between two contradictory factors in the design of LDPC codes, i.e., that of having a pseudorandom versus a structured PCM. In actual fact, MLS LDPC codes are capable of favoring either of these factors; however, we are particularly interested in how far the pseudorandom structure of the PCM can be restricted in favor of becoming more structured, without adversely affecting either the bit error ratio (BER) or the block error ratio (BLER) performance. The novel contribution of this paper is that we propose a class of structured protograph LDPC codes having a combinatorial nature, which benefits from reduced storage requirements, hardware-friendly implementations, and low-complexity encoding. Our simulation results provided for both additive white Gaussian noise (AWGN) and uncorrelated Rayleigh (UR) channels demonstrate that these advantages accrue without compromising the attainable BER and BLER performance when compared with their previously proposed more complex counterparts of the same length.

The structure of this paper is described as follows. Sections II and III introduce the preliminaries of LDPC codes and the basic principles of protograph LDPC codes. Then, Section IV describes the general construction and the necessary constraints of MLS codes. Our discourse continues with the characterization of the code's description complexity in Section V. The structure of MLS codes is then detailed in Section VI. Section VII describes the additional constraints, which were introduced to aid the efficient hardware implementation of MLS codes even further. Then, in Section VIII, we present an efficient search

Manuscript received October 5, 2009; revised January 15, 2010, March 16, 2010, and April 26, 2010; accepted April 29, 2010. Date of publication May 18, 2010; date of current version September 17, 2010. This work was supported in part by the European Union under the auspices of the Newcom and Phoenix projects and in part by the Engineering and Physical Sciences Research Council, U.K. The review of this paper was coordinated by Prof. W. A. Hamouda.

The authors are with the School of Electronics and Computer Science, 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.2010.2050708

¹Although they almost have the same nomenclature, MLS LDPC codes bear no resemblance to the previously proposed multilevel coding [26] and its relatives. The word “multilevel” is used here to emphasize the point that MLS LDPC codes are characterized by a PCM constructed of a number of levels.

method designed for graphs having a large girth, which is based on exploiting the isomorphism of edge-colored bipartite graphs. The corresponding simulation results are then detailed in Section IX before offering our conclusions in Section X.

II. PRELIMINARIES

We consider LDPC codes defined by regular bipartite graphs $G(\mathbf{H})$ associated with a PCM \mathbf{H} , whose rows span the null space of the code constructed over $\text{GF}(2)$. Then, the graph $G(\mathbf{H})$ consists of the nonempty set of elements $\{V(G), C(G), E(G)\}$, where $V(G)$ and $C(G)$ represent the disjoint vertex sets of the variable nodes and check nodes, while $E(G)$ is the set representing the edges.² Furthermore, we assume that the degree of the variable nodes $v \in V(G)$ and that of the check nodes $c \in C(G)$ are γ and ρ , respectively. If \mathbf{H} is of full rank, then it has a dimension of $(M \times N)$ elements, where $N = |V(G)|$ represents the LDPC block length, and the number of parity bits becomes $M = N - K = |C(G)|$, where K represents the number of original information bits. Consequently, $|E(G)|$ becomes equal to γN or ρM , and the code rate becomes $R = K/N = 1 - M/N$. We also point out that there exist numerous LDPC codes' constructions represented by rank-deficient PCMs, and therefore, they have rates that are higher than the rate of $1 - M/N$.

LDPC codes are typically decoded using the sum-product algorithm (SPA) [27], where messages or "beliefs" are exchanged between the nodes residing at both sides of the graph. The grade of independence of these messages is characterized by the length of the shortest cycle found in their Tanner graph, which is typically referred to as the girth g .

III. PROTOGRAPH LOW-DENSITY PARITY CHECK CODES

Let us proceed by briefly describing the construction of conventional protograph LDPC codes [28]. The construction of a protograph code involves three main steps; we first determine a base protograph, which typically consists of a graph having a relatively low number of nodes, and then replicate this graph J times. Finally, we permute the edges of the nodes in the J replicas of the base protograph to obtain the resultant graph. The code represented by this (final) graph is typically referred to as a protograph code [28]. Let a base protograph $G(\mathbf{H}_b)$ be described by the set of check nodes $C(\mathbf{H}_b) = \{c_{mj} : m = 1, \dots, M_b; j = 0\}$ and variable nodes $V(\mathbf{H}_b) = \{v_{nj} : n = 1, \dots, N_b; j = 0\}$, as well as by the set of edges $E(\mathbf{H}_b)$, where \mathbf{H}_b , M_b , and N_b represent the PCM of the base protograph, the number of check nodes, and the number of variable nodes in the base protograph, respectively. We also note that the index $j = 0$ is being assigned to the base protograph. After replicating $G(\mathbf{H}_b)$ J times, we obtain the Tanner graph $G(\mathbf{H})$ of the protograph code, which is defined by the sets $C(\mathbf{H})$, $V(\mathbf{H})$, and $E(\mathbf{H})$, where each set has a cardinality, which is J times higher than that of the corresponding set in the base protograph. The matrix \mathbf{H} denotes the PCM of the graph derived, which has $(JM_b \times JN_b)$ elements.

²We will interchangeably use the notation of $\{V(G), C(G), E(G)\}$ and $\{V(\mathbf{H}), C(\mathbf{H}), E(\mathbf{H})\}$.

An LDPC code is considered to be a protograph code if and only if the interconnection of the edges in each of the J replicas obeys the constraints governed by the base protograph, i.e., the interconnections between the nodes on both sides of the graph derived follow the same specific permutation pattern of the base protograph [28]. Typically, protograph LDPC codes are constructed using a variant of the progressive edge growth (PEG) [7] algorithm, which exploits the attractive characteristics of the PEG algorithm with regard to maximizing the girth of the corresponding graph, as well as the minimum distance, while satisfying the constraints governed by the base protograph. By the term "constraints," we imply that the connection of the edges in each of the J replicas must follow the specific permutation pattern of the base protograph.

Consider the example of a variable node v_{n0} , $n = 1, \dots, N_b$, located on the base protograph in a position adjacent to the check nodes c_{x0} , c_{y0} , and c_{z0} , where the three indices x , y , and z are within the integer interval $[1, M_b]$. Then, a PEG-based algorithm will randomly³ connect every variable node v_{nj} , $n = 1, \dots, N_b$, to one of the check nodes c_{xj} , c_{yj} , and c_{zj} , where $j = 0, \dots, J - 1$. This "randomness" introduced by the PEG algorithm will render the resultant PCM \mathbf{H} unstructured, hence slightly complicating its implementation.⁴ Indeed, it was argued in [30] that although protograph codes do obey an internal structure, they still suffer from a relatively high-complexity description due to the random PEG permutations, and thus, they still require a considerable amount of memory to store the addresses to which each input bit is mapped.

IV. GENERAL CONSTRUCTION METHODOLOGY OF MULTILEVEL STRUCTURED LOW-DENSITY PARITY CHECK CODES

On the other hand, MLS codes are a class of protograph codes that are always structured regardless of whether we opt for a pseudorandom or structured base protograph. Naturally, every structured code is governed by a set of constraints, and the larger the number of constraints satisfied is, the more structured the code's construction becomes. For the case of MLS LDPC codes, we distinguish between two types of constraints: the *necessary* constraints that must be satisfied by every MLS code and the *additional* constraints. In addition to the necessary constraints, we later impose a number of additional ones in Section VII to generate code constructions, which facilitate more efficient hardware implementations. For the sake of simplifying our discourse, we introduce the following three definitions:

Definition 4.1: The *base matrix*, which is represented by \mathbf{H}_b , is a sparse matrix defined over $\text{GF}(2)$ having $(M_b \times N_b)$ elements and containing exactly ρ and γ nonzero entries in each of its row and column, respectively. The base matrix of the MLS code will correspond to the PCM of the base protograph.

Definition 4.2: The *constituent matrices* are represented by the set $\Omega = \{\mathbf{Q}_0, \mathbf{Q}_1, \dots, \mathbf{Q}_{J-1}\}$, where each nonzero

³This is subject to the optimization criterion of maximizing the local girth of the variable node.

⁴A possible solution was proposed in [29], which uses a structured base protograph and then accordingly modifies the PEG algorithm to retain the same structure of the base.

constituent matrix \mathbf{Q}_j , $j = 0, \dots, J - 1$ is a distinct sparse matrix over GF(2) having the same dimensions as the base matrix. The parameter J denotes what we refer to as the number of *levels* of the MLS code in the resultant PCM, which also corresponds to the number of replicas of the base protograph constituting the derived graph.

Definition 4.3: The *adjacency matrix* is a $(J \times J)$ -element array matrix represented by \mathbf{P}_J , whose row blocks constitute a sharply transitive set of J permutations within Ω .

Clearly, there are a number of possible adjacency matrices that satisfy Definition 4.3, and their number increases with the number of levels J . For the sake of simplifying our discourse, let us consider the simple example of having an adjacency matrix given by

$$\mathbf{P}_J = \begin{pmatrix} 0 & 1 & 2 & \cdots & J-1 \\ J-1 & 0 & 1 & \cdots & J-2 \\ J-2 & J-1 & 0 & \cdots & J-3 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 1 & 2 & \cdots & J-1 & 0 \end{pmatrix}. \quad (1)$$

Each element in the adjacency matrix will then describe the position of the constituent matrix $\mathbf{Q}_j \in \Omega$ with respect to the PCM of the code. For this specific example, the $(JM_b \times JN_b)$ -element PCM \mathbf{H} of the J -level MLS code constructed on the adjacency matrix of (1) will be given by

$$\mathbf{H} = \begin{pmatrix} \mathbf{Q}_0 & \mathbf{Q}_1 & \mathbf{Q}_2 & \cdots & \mathbf{Q}_{J-1} \\ \mathbf{Q}_{J-1} & \mathbf{Q}_0 & \mathbf{Q}_1 & \cdots & \mathbf{Q}_{J-2} \\ \mathbf{Q}_{J-2} & \mathbf{Q}_{J-1} & \mathbf{Q}_0 & \cdots & \mathbf{Q}_{J-3} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \mathbf{Q}_1 & \mathbf{Q}_2 & \cdots & \mathbf{Q}_{J-1} & \mathbf{Q}_0 \end{pmatrix} \quad (2)$$

which is also sparse, and its null space represents an LDPC code having a rate of $R \geq 1 - M_b/N_b$.

We can now proceed to explain the aforementioned necessary constraints, which are outlined in the points below.

- **Constraint 1:** Each of the sparse constituent matrices $\mathbf{Q}_j \in \Omega$ must avoid having pairs of nonzero entries that are symmetrically repeated in two or more rows (or columns). It may readily be shown that this ensures that the girth of each constituent matrix is at least six.
- **Constraint 2:** All the nonzero entries of all the sparse constituent matrices $\mathbf{Q}_j \in \Omega$ must occur in the same position as in the base matrix. Furthermore, a nonzero entry in a particular location $\mathbf{Q}_j \in \Omega$, implies that the entries in the corresponding locations of $\mathbf{Q}_i \in \Omega$ are zero, where $i \in [0, J-1]$, and $i \neq j$. The number of nonzero elements in \mathbf{H}_b is equal to the sum of those in the J constituent matrices.

It may readily be demonstrated that the first and second constraints are closely related; in fact, any base matrix having a girth of $g > 4$ will produce a set of constituent matrices \mathbf{Q}_j , $j = 0, \dots, J - 1$ satisfying the first constraint. Naturally, a girth higher than four requires that the base matrix has a sufficiently large dimension. The loose lower bound on the required number of columns of \mathbf{H}_b , i.e., N_b , was given by Gallager in [1]: $N_b \geq 1 + \gamma(\rho - 1)$ for a specific girth of $g = 6$. Furthermore, we also note that if both the first and second constraints are satisfied, then the girth of the graph $G(\mathbf{H})$ associated with the

PCM of the MLS code is definitely larger than $g = 4$, since the adjacency matrix will avoid positioning any constituent matrix in the same row or column block.

MLS codes can also be considered as protograph codes since their adjacency matrix ensures that the permutations of edges incident to every N_b th variable node at each level of the graph $G(\mathbf{H})$ are determined using the same J constituent matrices [see the column blocks in the example shown in (2)], where the latter have nonzero entries occurring in the same position of the base matrix (by the second constraint). However, it is important to emphasize that, while all MLS codes constitute protograph codes, the reverse is not necessarily true. The reason for this lies in the previously described technique used for the construction of conventional protograph LDPC codes, which prohibits the generation of structured PCMs such as the one shown in the example in (2). In this light, we may also interpret MLS codes as specific protograph codes having more compact descriptions. Despite the aforementioned construction constraints, MLS codes still benefit from inheriting implementationally attractive semiparallel architectures such as those suggested by Lee *et al.* [31].

Previously, we have mentioned that the PCM construction of MLS codes simultaneously exhibits both pseudorandom and deterministic structural characteristics. The pseudorandom PCM structure of MLS codes is attributed to the fact that no constraints are imposed on the actual base matrix selected, and therefore, any previously proposed pseudorandom PCM construction can be utilized as a base matrix. The base matrix chosen may obey a structured construction. However, we emphasize that all our results were obtained using base matrices having pseudorandom constructions. Our decision was based on the fact that the resultant construction of \mathbf{H} will definitely be structured due to the *necessary* constraints imposed. This can be verified with the aid of the example in (2). The positions of the nonzero entries in each of the constituent matrices \mathbf{Q}_j in Ω are also chosen at random while obeying the previously described first and second constraints.

In this paper, we have assumed both randomly and uniformly distributed positions for the nonzero entries in the constituent matrices \mathbf{Q}_j of the set $\Omega = \{\mathbf{Q}_0, \mathbf{Q}_1, \dots, \mathbf{Q}_{J-1}\}$. For the case of uniformly distributed positions, we have introduced *additional* constraints, which enhance the code's structure and, thus, improve the associated implementational aspects even further. The additional constraints will be discussed in Section VII.

V. COMPLEXITY OF THE CODE DESCRIPTION

It is quite easy to recognize the reduced code description complexity that accrues from having a PCM obeying (2). Increasing the number of levels J will automatically imply that the size of both the base matrix and the constituent matrices $\mathbf{Q}_j \in \Omega$ will be decreased, and consequently, the grade of randomness in the resultant MLS code's construction will become less pronounced.

Using the terminology introduced in Section II, we denote the two vertex sets belonging to the regular bipartite Tanner graph $G(\mathbf{H})$, representing an MLS code by the variable node set $V(G) = \{v_{nj} : n = 1, \dots, N_b; j = 0, \dots, J-1\}$ and the check

node set $C(G) = \{c_{mj} : m = 1, \dots, M_b; j = 0, \dots, J - 1\}$. Furthermore, we assume that $E(\mathbf{H}_j)$ denotes the nonempty set of edge interconnections that uniquely and unambiguously describe the connections between the check nodes c_{mj} , $m = 1, \dots, M_b$, and the variable nodes v_{nj} , $n = 1, \dots, N_b$. In effect, the edges represented by this set correspond to the nonzero entries of the constituent matrix $\mathbf{Q}_j \in \Omega$, $j = 0, \dots, J - 1$. In this light, the complete bipartite graph represented by the PCM \mathbf{H} of an MLS code can be interpreted as a specific partition of an edge set $E(\mathbf{H})$ constituted by the following union:

$$E(\mathbf{H}) = E(\mathbf{H}_0) \cup E(\mathbf{H}_1) \cup E(\mathbf{H}_2) \dots E(\mathbf{H}_{J-1}) \quad (3)$$

where $E(\mathbf{H}_j)$, $j = 0, \dots, J - 1$, are all disjoint (as required by the second constraint) nonempty sets of edges. Let the parameter ϵ denote the effective number of edges that must be stored to represent the corresponding Tanner graph or, equivalently, the total number of entries in the memory LUT storing the code description. Following (3), a J -level MLS code can be represented by $\epsilon = |E(\mathbf{H})|$, which—by the second constraint—is identical to $|E(\mathbf{H}_b)| = N_b \gamma$. Therefore, it is only necessary to enumerate the edges present within each of the constituent matrices $\mathbf{Q}_j \in \Omega$ to describe an entire MLS code. Due to the second necessary constraint introduced in Section IV, these edges will be represented by the nonzero entries found in the same positions of the base matrix \mathbf{H}_b . On the other hand, an LUT that stores a pseudorandom PCM description must enumerate $\epsilon = N \gamma$ edges, where $N = N_b J$. Hence, we may argue that the complexity of an MLS code's description is effectively reduced by a factor, which is proportional to the number of levels J , when compared with other pseudorandom codes.

VI. EXTERNAL CONSTRUCTION OF MULTILEVEL STRUCTURED CODES

MLS codes possess both an internal and an external structure, where the latter is based on the adjacency matrix \mathbf{P}_J chosen for the specific implementation, which is essentially what makes them different from the protograph codes originally proposed by Thorpe [28]. The adjacency matrix will then appropriately position each (internally structured) constituent matrix $\mathbf{Q}_j \in \Omega$ with respect to the (externally structured) PCM of the MLS code \mathbf{H} . This implies that the adjacency matrix must also be stored, and therefore, it is equally desirable that it has a compact description. Hence, we may identify two classes of MLS codes, which are distinguished by their adjacency matrices and by the complexity of their descriptions, as described in the following sections.

A. Class-I MLS Codes Based on an HCC

We will introduce the following definition to define the family of Class-I MLS codes.

Definition 6.1: A *homogeneous coherent configuration* (HCC) is identified by the set of binary matrices $\mathbf{A} = \mathbf{A}_0, \dots, \mathbf{A}_{J-1}$ having a sum equal to the all-one matrix and that is closed under transposition. In addition, the set \mathbf{A} has the property that one of the matrices is the identity matrix and that the product of any two matrices is a linear combination of the matrices in the set.

Class-I MLS codes are those codes whose adjacency matrix describes the adjacency algebra of an HCC [32]. The adjacency

matrix and the corresponding PCM of a J -level Class-I code are, in fact, shown in (1) and (2), respectively, which represent the adjacency matrix of a nonsymmetric association scheme [32] on J points. Elaborating slightly further, we will use the example of a five-level Class-I MLS code having an adjacency matrix \mathbf{P}_5 given by

$$\mathbf{P}_5 = \begin{pmatrix} 0 & 1 & 2 & 3 & 4 \\ 4 & 0 & 1 & 2 & 3 \\ 3 & 4 & 0 & 1 & 2 \\ 2 & 3 & 4 & 0 & 1 \\ 1 & 2 & 3 & 4 & 0 \end{pmatrix} \quad (4)$$

where each element in the matrix corresponds to a subscript and thus defines the position of a constituent matrix $\mathbf{Q}_j \in \Omega$. The compact description of \mathbf{P}_J can readily be demonstrated in two different ways. First, it can be recognized that each of the J zero-one-valued matrices $\mathbf{A}_j \in \mathbf{A}$, $j \in [0, J - 1]$ is a circulant matrix of size $J = 5$. It can be observed in both (2) and (4) that the matrix $\mathbf{A}_0 = \mathbf{I}_J$, where \mathbf{I}_s corresponds to the identity matrix having a size of s , while the remaining binary matrices \mathbf{A}_j , $j \in [1, J - 1]$ have a binary one entry in column $(r + j) \bmod J$, where r is the row index of the circulant matrix, $0 \leq r \leq J - 1$, and $(a \bmod b)$ represents the modulus after division of a by b . Alternatively, it can also be argued that a cyclic shift obeying $x \mapsto x + 1$, $x \in \mathbb{Z}_5$, with \mathbb{Z} being the set of integers, is an automorphism of this scheme, and therefore, its basic relations can be simply described by

$$R_i = \{(x, y) \in \mathbb{Z}_5 \times \mathbb{Z}_5 | y - x = i\}, \quad i \in [0, 4] \quad (5)$$

where R_i is a binary relationship on the group \mathbb{Z}_5 , and “ \times ” denotes the Cartesian product.

B. Class-II MLS Codes Based on Latin Squares

The adjacency matrix \mathbf{P}_J can also be interpreted as a Latin square [33] of order J , consisting of row and column blocks described by the sets \mathbf{Q}_j , $j = 0, \dots, J - 1$ that generate the symmetric group S_Ω on Ω having order $J!$. Fig. 1 depicts this representation of an adjacency matrix for a six-level Class-II MLS code, where the J rows and columns of the Latin square correspond to the respective *multicheck* node $C_{mj} \subset C(G)$ and *multivariable* node $V_{nj} \subset V(\mathbf{H})$, $|C_{mj}| = |V_{nj}|$, where we have $m = 1, \dots, M_b$, $n = 1, \dots, N_b$, and $j = 0, \dots, J - 1$.

A Latin square is also equivalent to a 1-factorization of a bipartite graph, and hence, we can also regard a J -level MLS code as an edge-colored complete bipartite graph of degree J . Equation (3) shows that the edge set $E(\mathbf{H})$ of the graph $G(\mathbf{H})$ is partitioned into J disjoint nonempty sets $E(\mathbf{H}_j) \subset E(\mathbf{H})$, $j = 0, \dots, J - 1$. This brings us to the notion of what is known as coloring [34] of edges, where $E(\mathbf{H})$ is said to be an edge coloring of $G(\mathbf{H})$ if any two edges on the graph containing the same vertex have different colors. Correspondingly, each symbol of the Latin square will create a monochromatic 1-factor of the Tanner graph and, thus, represents a *multiedge* on the degree- J edge-colored graph. Fig. 1 also illustrates the corresponding edge-colored graph for a six-level Class-II MLS code having an adjacency matrix represented by a reduced Latin square. The different “edge colors” on the Tanner graph in Fig. 1 are represented using different line types.

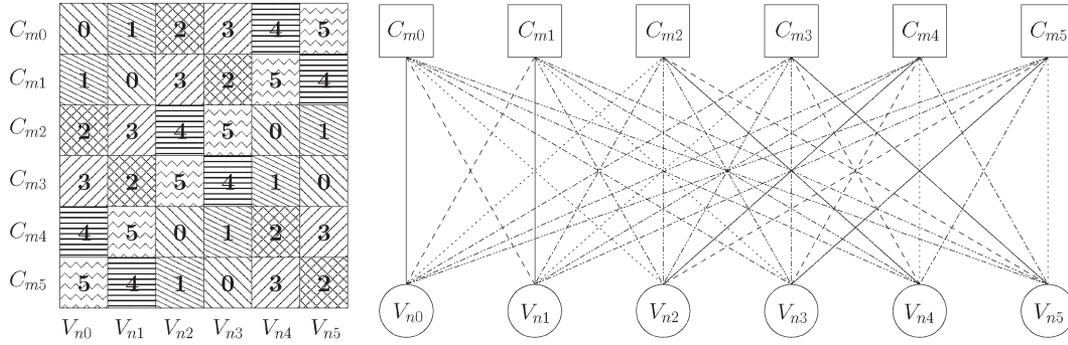


Fig. 1. Latin square representation of the adjacency matrix of a six-level Class-II MLS code. The J rows and columns of the Latin square correspond to the respective *multicheck* node C_{m_j} and *multivariable* node V_{n_j} , where $m = 1, \dots, M_b$, $n = 1, \dots, N_b$, and $j = 0, \dots, J - 1$. Each of the J symbols (or patterned box) in the Latin square represent the disjoint nonempty set (*multiedge*) $E(\mathbf{H}_j)$, $j = 0, \dots, J - 1$ [see (3)]. The corresponding *edge-colored* complete bipartite graph is shown on the right, having a degree of J , where the different “edge color” (corresponding to a multiedge) is represented by using a different line type.

VII. ADDITIONAL CONSTRAINTS

We impose the additional constraints over the necessary constraints mentioned in Section IV to aid the efficient hardware implementation of MLS codes even further. The constraints are described as follows.

- Constraint 3: Starting from any base matrix having $(M_b \times N_b)$ elements, uniformly distribute the nonzero entries across the constituent matrices so that each row and column of any $\mathbf{Q}_j \in \Omega$ contains a single nonzero entry. This constraint can only be applied in the scenario when the number of levels J is at least equal to the column weight ρ of the PCM.
- Constraint 4: Replace each nonzero entry in each constituent matrix by a circulant matrix of size q from the set \mathbf{I}_q , $\mathbf{I}_q^{(1)}$, $\mathbf{I}_q^{(2)}$, \dots , $\mathbf{I}_q^{(q-1)}$, where $\mathbf{I}_q^{(s)}$ represents a right cyclic shift by s positions for each row of the identity matrix \mathbf{I}_q .

The third constraint will facilitate the parallel processing of messages exchanged over the interconnections between the check and variable nodes. Since each nonzero entry in each row or column of the base matrix is positioned in a different constituent matrix, each memory block will only access (read or write) each location once per clock cycle. Furthermore, it becomes possible to simultaneously process the ρ edges incident on each check node by the J memory blocks.

By the fourth constraint, the resultant PCM having $(qJM_b \times qJN_b)$ elements will be composed of only circulant matrices of weight zero or one, and thus, the code effectively becomes QC. The amount of memory required to store the code’s description is then reduced by factor of $1/qJ$, when compared with other pseudorandom constructions, since memory shifts corresponding to the QC PCM structure can be used to address the messages exchanged between the nodes. The encoding process can be implemented using simple shift registers, thus considerably reducing the associated encoding complexity.

VIII. EFFICIENT SEARCH FOR GRAPHS HAVING A LARGE GIRTH

We have selected MLS codes based on the optimization criterion of maximizing the average girth, using an approach similar to that of Mao and Banihashemi in [35]. However, the differentiating feature of our search is that it is now possible

to avoid the inspection of isomorphic (edge-colored) graphs based on their corresponding Latin square representation, and hence, our search is much more efficient. Formally, we have the following definitions.

Definition 8.1: Two Latin squares S and S' are said to be *isotopic* if there exists a triple (α, β, χ) (referred to as an *isotopy*), where α , β , and χ correspond to a respective row, column, and symbol permutation, which carries the Latin square S to S' . Effectively, this implies that if we consider any particular row and column position of the Latin square specified by the check and variable nodes (c_{m_j}, v_{n_j}) containing entry e , where $m = 1, \dots, M_b$, $n = 1, \dots, N_b$, and $j = 0, \dots, J - 1$, then the entry at position $(\alpha(c_{m_j}), \beta(v_{n_j}))$ of the Latin square S' will be equal to $\chi(e)$. Subsequently, an *isotopy class* comprises the set of all the Latin squares isotopic to a given Latin square.

Definition 8.2: Two Latin squares S and S'' are said to be *conjugates* (or *parastrophes*) if S'' is obtained from S by simply permuting the “roles” of the rows, columns, and symbols of S . Therefore, there will be six conjugate Latin squares⁵ that can be obtained from S . With the aid of the following claim, we can effectively avoid searching through the isomorphic edge-colored graphs.

Claim 7.1 [36]: Two Latin squares S_1 and S_2 will give rise to isomorphic edge-colored complete bipartite graphs if and only if S_1 is isotopic to either S_2 or $(S_2)^T$, where the superscript T denotes the transpose operation.

The transpose of Latin square S is actually one of its conjugates, which is obtained by exchanging the roles of the columns with that of the rows. For the sake of illustrating the efficacy of the aforementioned technique, let us consider the example of a Class-II MLS code having $J = 6$ levels in its underlying PCM. According to McKay and Rogoyski [37], the total number of distinct Latin squares of order J is given by $\mathcal{X}_J = J! \times (J - 1)! \times L(J, J)$ [37], where $L(J, J)$ is the number of normalized $(J \times J)$ -element Latin squares. Consequently, the full search will involve the examination of a staggering number of $\mathcal{X}_6 = 6! \times 5! \times 9408$ [37] Latin squares. By contrast, using the aforementioned Claim 7.1, we can avoid

⁵If we use the orthogonal array representation of a Latin square having order J , thus representing the square by J^2 triples in the form of (row, column, symbol), we can obtain a conjugate of the same Latin square by changing the roles in each triple.

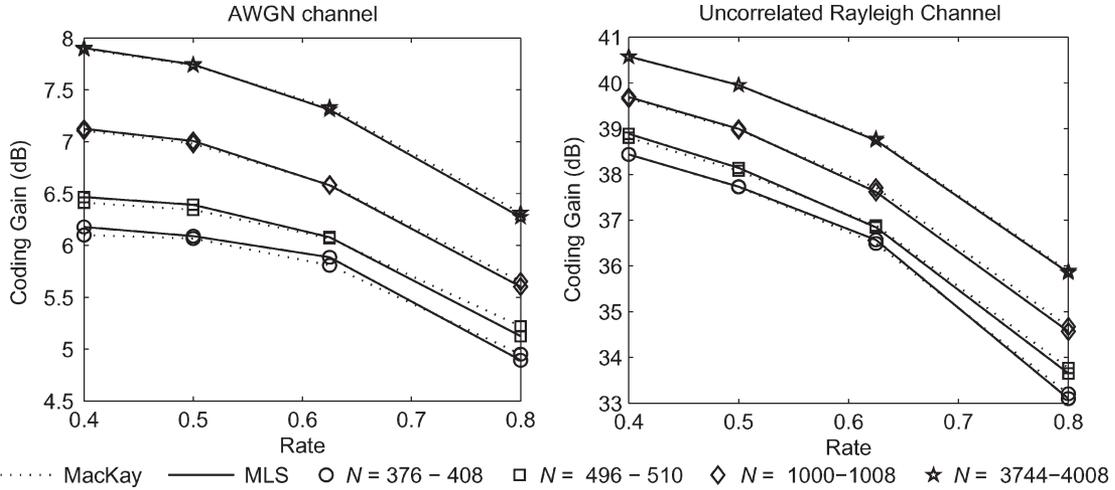


Fig. 2. Coding gain achieved at a BER of 10^{-5} by MacKay's [39] and Class-I MLS LDPC codes when communicating over the AWGN and UR channels using BPSK modulation using different block lengths at various coding rates.

the unfruitful inspection of isomorphic graphs and, thus, only search a representative from each isotopy class and four of its conjugates. Given that there are a total of 22 isotopy classes⁶ for an order-six Latin square, the number of Latin squares that has to be examined is reduced to only 88 Latin squares.

We also note that Class-I MLS codes are effectively a subclass of Class-II MLS codes. It is easy to demonstrate that by permuting the rows, columns, or symbols and/or by permuting the roles of the rows, columns, and symbols of the Latin square, one can obtain the adjacency matrix of a Class-II MLS code from that of a Class-I code. The more constrained Class-I MLS codes are characterized by a simpler description for their adjacency matrix \mathbf{P}_J , since it is only required to store the position of each constituent matrix \mathbf{Q}_j in the first level of \mathbf{P}_J , while the respective positions of each \mathbf{Q}_j in the other levels can effectively be generated by means of a shift register.

IX. RESULTS AND DISCUSSION

The results presented in this section were obtained using binary phase-shift keying (BPSK) modulation, when transmitting over the AWGN and UR channels and using a maximum of $I = 100$ decoding iterations of the SPA. We note that, in all the test cases, both \mathbf{H}_b and \mathbf{Q}_j , $j = 0, \dots, J - 1$, which are selected to construct a J -level MLS code having an $(M \times N)$ -element PCM, will contain $(M/J \times N/J)$ elements. The base matrix was pseudorandomly constructed, without any effort for optimization. Subsequently, the constituent matrices were constructed according to the heuristic search described in Section VIII and satisfying the aforementioned constraints. Our forthcoming discourse is organized in two sections: Section IX-A describes the results obtained for MLS LDPC codes satisfying the necessary constraints of Section IV, while Section IX-B details the results obtained for MLS LDPC codes satisfying both the necessary constraints and the additional constraints of Section VII.

⁶A list of the isotopy classes for Latin squares of small orders is given by McKay in [38].

TABLE I
NUMBER OF LEVELS J FOR THE MLS CODES WHOSE PERFORMANCE IS ILLUSTRATED IN FIG. 2

Rate	Block length N			
	376 - 408	496 - 510	1000 - 1008	3744 - 4008
0.4	5	5	5	5
0.5	5	6	6	6
0.625	3	3	6	6
0.8	3	3	3	5

A. MLS Codes Satisfying Only the Necessary Constraints

The BER/BLER performance of MLS codes is very much dependent on the number of levels employed, where, generally, the BER/BLER performance improves upon decreasing the number of levels J , which is at the expense of a higher code's description complexity. We particularly emphasize that the MLS performance results reported in this section represent the worst-case scenario in terms of the associated BER/BLER performance of the MLS code, because a code having a lower number of levels J will exhibit better performance than the codes characterized here.⁷ However, this is actually the best-case scenario in terms of the achievable code description complexity reduction. In this section, we provide simulation results for codes having $\gamma = 3$, a block length of N ranging from 376 to 4008 bits, and code rates R spanning from 0.4 to 0.8.⁸ Fig. 2 depicts the coding gain achieved by the proposed MLS LDPC codes and by MacKay's pseudorandom code [39] at a BER of 10^{-5} . The number of levels that was actually used is summarized in Table I for each code rate and block length range. It can be observed from Fig. 2 that, despite the reduction in the code's description complexity, the performance

⁷For example, the coding gain achieved at a BER of 10^{-5} by a half-rate Class-I MLS codes having a block length of $N = 504$ when communicating over the AWGN channel and constructed using $J = 2, 3, 4, 5$, and 6 levels was equal to 6.450, 6.440, 6.427, 6.412, and 6.393 dB, respectively. The corresponding MacKay benchmark code exhibited a coding gain of 6.344 dB.

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

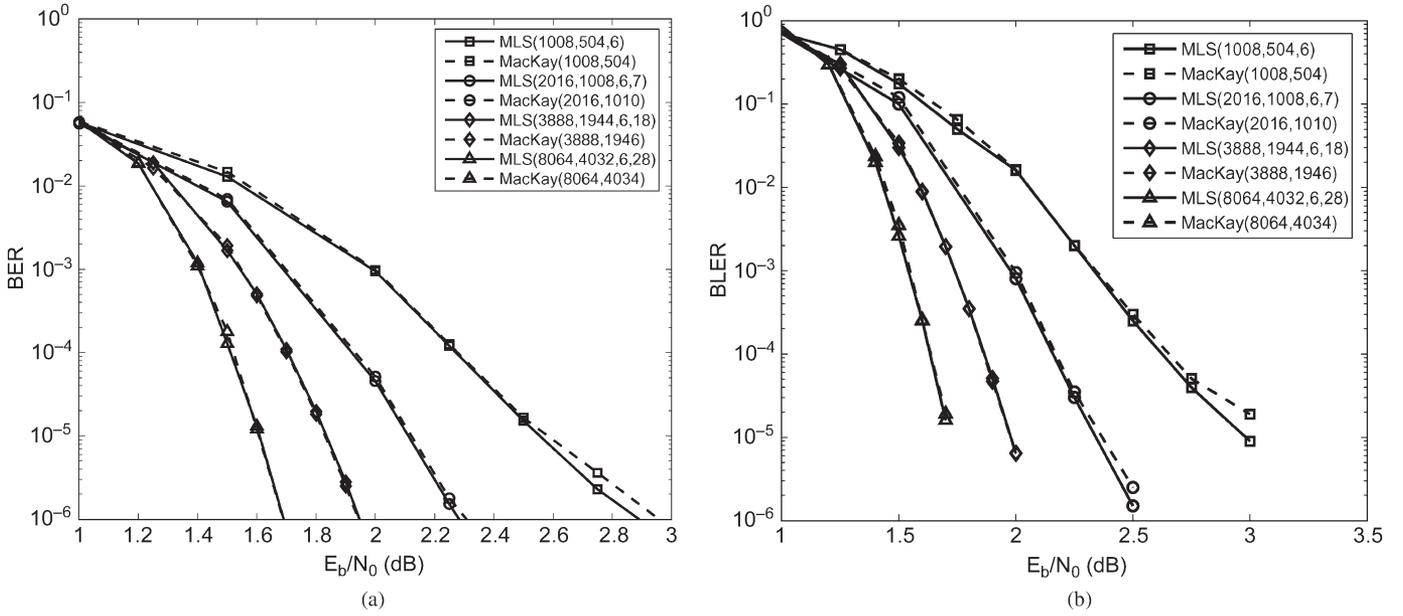


Fig. 3. BER and BLER performance comparison of $R = 0.5$ Class-I MLS and MacKay's [39] LDPC codes with $N = 1008 - 8064$ and a maximum of $I = 100$ decoder iterations when transmitting over the AWGN channel using BPSK modulation. All the codes shown have a PCM associated with a column weight of $\gamma = 3$ and a row weight of $\rho = 6$. (a) BER performance. (b) BLER performance.

of the proposed MLS codes is still comparable with that of the corresponding pseudorandom benchmarker codes.

It can also be observed from Table I that it was always possible to design MLS codes that achieve the maximum parallelization factor (equal to N/ρ) for the case of low to medium code rates. However, it becomes quite difficult to increase the number of levels up to the PCM row weight for the case of high-rate codes without suffering any BER or BLER performance loss compared with the pseudorandom benchmarker codes. Notice that, as the code rate is increasing, the row weight is also increasing while the number of rows in the base matrix is decreasing. At short block lengths, the latter makes it quite difficult to find a base matrix having a girth of $g > 4$, and consequently, the first necessary constraint described in Section IV may not be satisfied.

B. MLS Codes Satisfying Both the Necessary and the Additional Constraints

This section details our BER/BLER performance results for MLS LDPC codes that also satisfy the additional constraints. We remark that no BER/BLER performance degradation was observed for the MLS LDPC codes satisfying one or both of the additional constraints, when compared with the corresponding MLS codes satisfying only the necessary constraints. On the contrary, the average girth of the associated Tanner graphs was slightly improved⁹ after imposing the first additional constraint (see constraint 3 in Section VII). We will appropriately distinguish between MacKay's pseudorandom codes, the proposed MLS codes satisfying the first three constraints, and QC MLS

⁹It can easily be demonstrated that it is more beneficial (in terms of improving the girth of the associated Tanner graph) to uniformly distribute the non-zero entries of the base matrix \mathbf{H}_b across the J constituent matrices, instead of using any other random distribution of the logical one values.

codes satisfying all the previously mentioned constraints using the notation (N, K) , (N, K, J) , and (N, K, J, q) , respectively.

Fig. 3(a) illustrates the comparison of the achievable BER performance for transmission over the AWGN channel employing half-rate six-level Class-I MLS codes, as well as the corresponding MacKay codes having block lengths of 1008, 2016, 3888, and 8064. The achievable BLER performance is then portrayed in Fig. 3(b). It was ensured that at least 100 block errors were collected at each point on the simulation curve. The MLS(1008,504,6) code was constructed using an (84×168) -element base matrix and six constituent matrices. Both the QC MLS(2016,1008,6,7) and the QC MLS(8064,4032,6,28) codes were constructed using the same (24×48) -element base matrix, but the former was expanded using circulant matrices of size seven, while the latter used circulant matrices of size 28. The QC MLS(3888,1944,6,18) code was then constructed using a base matrix having dimensions of (18×36) elements, decomposed over six constituent matrices, and then expanded by circulant matrices of size 18. The adjacency matrix for these four MLS codes is based on a six-point HCC, while the row and column weight of their PCM is equal to three and six, respectively. It can be observed that despite their constrained PCM, the MLS codes exhibit no BER and BLER performance loss with compared with their pseudorandom counterparts, although the MLS codes exhibit substantial implementational benefits. Similar BER and BLER performance trends are exhibited over the UR channel, as illustrated in Fig. 4(a) and (b).

Table II summarizes the distance between the Shannon limit of the codes' exhibited BER performance for both the AWGN and UR channels, which is measured at a BER of 10^{-6} . We also compared the complexities of the codes' description for the MLS and the corresponding MacKay benchmarker codes by quantifying the effective number of edges ϵ that must be stored or, equivalently, the number of LUT entries that

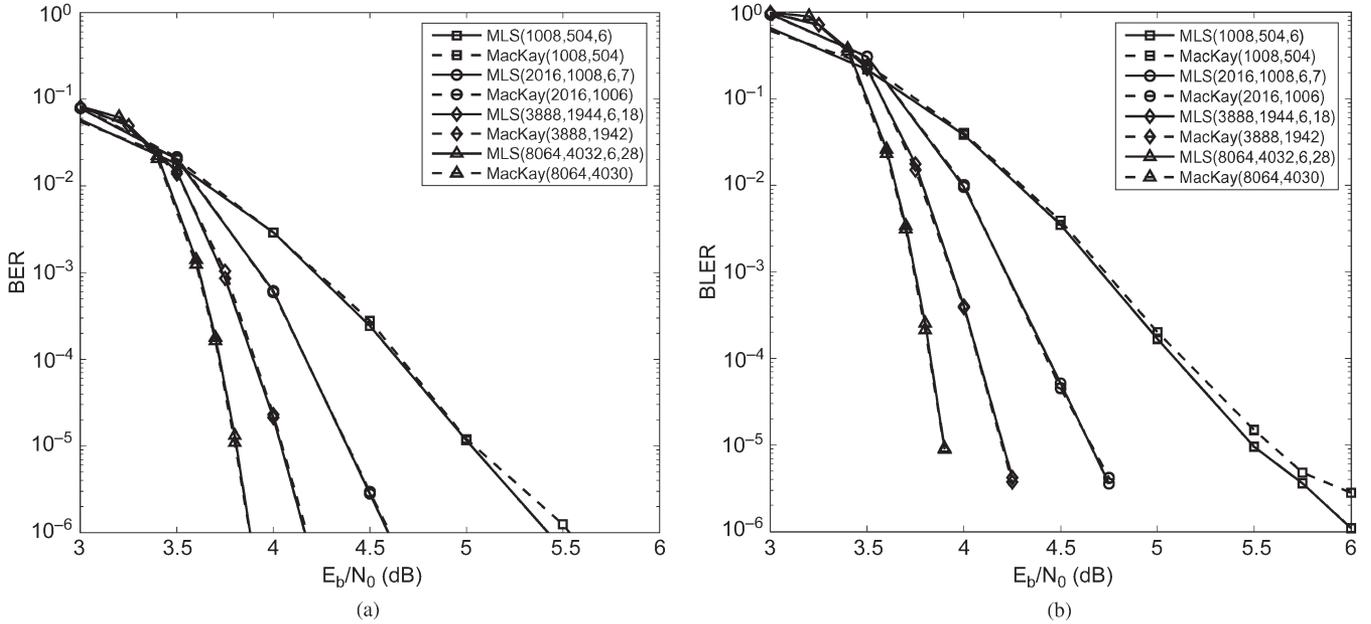


Fig. 4. BER and BLER performance comparison of $R = 0.5$ Class-I MLS and MacKay's [39] LDPC codes with $N = 1008 - 8064$ and a maximum of $I = 100$ decoder iterations when transmitting over the UR channel using BPSK modulation. All the codes shown have a PCM associated with a column weight of $\gamma = 3$ and a row weight of $\rho = 6$. (a) BER performance. (b) BLER performance.

TABLE II
PERFORMANCE COMPARISON BETWEEN THE CLASS-I MLS AND MACKEY CODES

Code	ϵ^\dagger	Shannon Gap* (dB) (AWGN)	Shannon Gap* (dB) (UR)
MLS(1008,504,6)	504	2.704	3.590
MacKay(1008,504)	3024	2.771	3.702
MLS(2016,1008,6,7)	144	2.095	2.762
MacKay(2016,1006)	6048	2.120	2.771
MLS(3888,1944,6,18)	108	1.759	2.330
MacKay(3888,1942)	11664	1.759	2.339
MLS(8064,4032,6,28)	144	1.501	2.046
MacKay(8064,4030)	24192	1.503	2.046

\dagger The effective number of edges that must be stored, or equivalently, the number of entries in the memory LUT storing the code description.

* The distance between the Shannon limit and the exhibited code's performance at a BER of 10^{-6} . The Shannon limit for the AWGN and the UR channel was assumed to be 0.188 dB and 1.834 dB, respectively.

are needed to store the code's description. It is evident from Table II that the proposed MLS codes benefit from considerable gains in terms of the required storage memory. For example, the QC MLS(8064,4032,6,28) is uniquely and unambiguously described by as few as 144 edges, while the corresponding MacKay(8064,4030) code requires the enumeration of a significantly higher number of 24 192 edges.

Our BER performance comparison between the half-rate Class-I and Class-II MLS codes and the MacKay benchmark codes is provided in Fig. 5 for transmission over the AWGN channel. It can be observed that the Class-II MLS(1008,504,6) code exhibits a BER versus E_b/N_0 performance that is approximately 0.15 dB better than that of the aforementioned Class-I MLS and 0.21 dB better than that of the corresponding MacKay-style benchmark code. Furthermore, a modest but measurable gain of approximately 0.07 and 0.10 dB was attained by the Class-II QC MLS(2016,1008,6,7) over the respective Class-I QC MLS and the MacKay code. Class-II MLS codes attain superior BER/BLER performance in comparison

with Class-I codes since the former have to satisfy a lower number of constraints and, thus, attain a higher average girth. Moreover, Class-II MLS codes have the advantage that, given the same J number of constituent matrices, we can represent ζ_J as the number of different PCMs, where ζ_J depends on the order J of the corresponding Latin square. This attractive property of Class-II MLS codes was exploited in [40] for creating the concept of channel code-division multiple access, where the different users are uniquely identified by their user-specific channel codes, namely, by MLS codes having the same constituent matrices but employing user-specific adjacency matrices.

Fig. 5 also depicts the BER performance for transmission over the AWGN channel for the Class-II QC MLS(8064,4032,8,4) having a PCM associated with a column weight of $\gamma = 4$ and a row weight of $\rho = 8$. This code achieves a BER of 10^{-6} at a signal-to-noise ratio of approximately 1.64 dB and is, thus, only 1.45 dB away from the Shannon limit. This code achieves similar performance to that of a corresponding

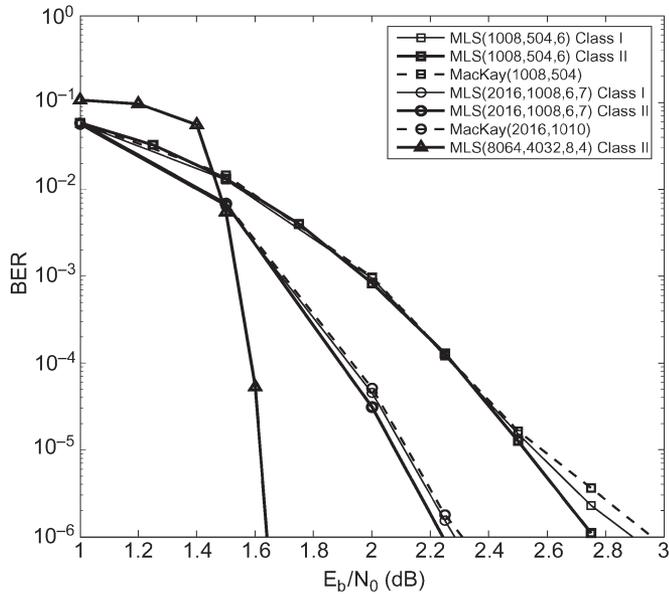


Fig. 5. BER performance comparison of $R = 0.5$ Class-I and Class-II MLS and MacKay’s [39] LDPC codes with $N = 1008$ and 2016 bits, when transmitting over the AWGN channel using BPSK modulation. The maximum number of allowable decoder iterations I was set to 100 . All the codes shown have a PCM associated with a column weight of $\gamma = 3$ and a row weight of $\rho = 6$, except for the Class-II QC MLS(8064,4032,8,4), which has a PCM associated with a column weight of $\gamma = 4$ and a row weight of $\rho = 8$.

QC half-rate code based on the Euclidean subgeometry $EG^*(3, 2^3)$ (see [21, Tab. I]) and having block length $N = 8176$, which was designed by decomposing circulant matrices constructed from finite geometries. Moreover, all our MLS codes benefit from a readily parallelizable protograph decoder structure [31], which is not the case for the $EG^*(3, 2^3)$ code in [21]. Furthermore, these LDPC codes constructed from finite and Euclidean geometries such as those presented in [13] and [21] tend to have higher row and column weights than other LDPC codes (see, for example, [13, Tab. I–III]). Thus, their attractive BER/BLER performance is somewhat achieved at the expense of a higher decoding complexity imposed by their higher logic depth. On the other hand, we were still able to attain excellent BER/BLER performance with codes having only $\gamma = 3$.

X. SUMMARY AND CONCLUSION

In this paper, we have proposed the construction of MLS LDPC codes, which benefit from having a low-complexity description due to the structured row–column connections based on protographs, while also having low-complexity encoding and decoding implementations due to their semiparallel architectures. We have investigated their BER and BLER performance for transmission over both AWGN and UR channels for various code rates, as well as block lengths, and compared their performance with both pseudorandom and structured benchmarks. Explicitly, our experimental results have demonstrated that these considerable implementational benefits provided by the proposed MLS LDPC codes accrue without any BER/BLER performance loss. Indeed, no performance loss has been observed, at least down to a BER/BLER of 10^{-6} .

REFERENCES

- [1] R. G. Gallager, “Low-density parity-check codes,” *IRE Trans. Inf. Theory*, vol. IT-8, no. 1, pp. 21–28, Jan. 1962.
- [2] D. MacKay, “Good error-correcting codes based on very sparse matrices,” *IEEE Trans. Inf. Theory*, vol. 45, no. 2, pp. 399–431, Mar. 1999.
- [3] T. J. 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.
- [4] T. J. Richardson 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.
- [5] 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.
- [6] M. G. Luby, M. Mitzenmacher, M. A. Shokrollahi, and D. A. Spielman, “Improved low-density parity-check codes using irregular graphs,” *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 585–598, Feb. 2001.
- [7] 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.
- [8] 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.
- [9] T. Tian, C. Jones, J. D. Villaseñor, and R. D. Wesel, “Construction of irregular LDPC codes with low error floors,” in *Proc. IEEE ICC*, May 11–15, 2003, vol. 5, pp. 3125–3129.
- [10] T. Tian, C. R. Jones, J. D. Villaseñor, and R. D. Wesel, “Selective avoidance of cycles in irregular LDPC code construction,” *IEEE Trans. Commun.*, vol. 52, no. 8, pp. 1242–1247, Aug. 2004.
- [11] 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, Dec. 2004.
- [12] N. Bonello, S. Chen, and L. Hanzo, “LDPC codes and their rateless relatives,” *Commun. Surveys Tuts.*, to be published.
- [13] Y. Kou, S. Lin, and M. P. C. Fossorier, “Low-density parity-check codes based on finite geometries: A rediscovery and new results,” *IEEE Trans. Inf. Theory*, vol. 47, no. 7, pp. 2711–2736, Nov. 2001.
- [14] C. J. Colbourn and H. J. Dinitz, *The CRC Handbook of Combinatorial Designs*. Boca Raton, FL: CRC, 1996.
- [15] I. F. Blake and R. Mullin, *The Mathematical Theory of Coding*. New York: Academic, 1975.
- [16] S. J. Johnson and S. R. Weller, “Regular low-density parity-check codes from combinatorial designs,” in *Proc. IEEE Int. Telecommun. Workshop*, Sep. 2–7, 2001, pp. 90–92.
- [17] B. Vasic, “Structured iteratively decodable codes based on Steiner systems and their application in magnetic recording,” in *Proc. IEEE Global Telecommun. Conf.*, San Antonio, TX, Nov. 25–29, 2001, vol. 5, pp. 2954–2960.
- [18] B. Ammar, B. Honary, Y. Kou, J. Xu, and S. Lin, “Construction of low-density parity-check codes based on balanced incomplete block designs,” *IEEE Trans. Inf. Theory*, vol. 50, no. 6, pp. 1257–1269, Jun. 2004.
- [19] B. Vasic, “High-rate low-density parity check codes based on anti-Pasch affine geometries,” in *Proc. Int. Conf. Commun.*, Apr. 28–May 2, 2002, vol. 3, pp. 1332–1336.
- [20] E. Eleftheriou and S. Ölçer, “Low density parity-check codes for digital subscriber lines,” in *Proc. IEEE Int. Conf. Commun.*, 2002, vol. 3, pp. 1752–1757.
- [21] 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.
- [22] 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.
- [23] F. Zhang, X. Mao, W. Zhou, and H. Pfister, “Girth-10 LDPC codes based on 3-D cyclic lattices,” *IEEE Trans. Veh. Technol.*, vol. 57, no. 2, pp. 1049–1060, Mar. 2008.
- [24] 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. 11, pp. 71–81, Jan. 2006.
- [25] R. Tanner, “A recursive approach to low complexity codes,” *IEEE Trans. Inf. Theory*, vol. IT-27, no. 5, pp. 533–547, Sep. 1981.
- [26] H. Imai and S. Hirawaki, “A new multilevel coding method using error-correcting codes,” *IEEE Trans. Inf. Theory*, vol. IT-23, no. 3, pp. 371–377, May 1977.
- [27] F. Kschischang, B. 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.

- [28] J. Thorpe, "Low-density parity-check LDPC codes constructed from protographs," Jet Propulsion Lab., Pasadena, CA, IPN Progress Rep. 42-154, Aug. 2003.
- [29] N. Bonello, S. Chen, and L. Hanzo, "Construction of regular quasi-cyclic protograph LDPC codes based on Vandermonde matrices," *IEEE Trans. Veh. Technol.*, vol. 57, no. 4, pp. 2583–2588, Jul. 2008.
- [30] K. Andrews, S. Dolinar, D. Divalsar, and J. Thorpe, "Design of low-density parity-check codes LDPC codes for deep-space applications," Jet Propulsion Lab., Pasadena, CA, IPN Progress Rep. 42-159, Nov. 2004.
- [31] 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.
- [32] R. A. Bailey, *Association Schemes, Designed Experiments, Algebra and Combinatorics*. Cambridge, U.K.: Cambridge Univ. Press, 2004.
- [33] C. F. Laywine and G. L. Mullen, *Discrete Mathematics Using Latin Squares*. New York: Wiley-Interscience, 1998.
- [34] N. L. Biggs, *Discrete Mathematics*, 2nd ed. Oxford, U.K.: Oxford Univ. Press, Dec. 2002.
- [35] Y. Mao and A. H. Banihashemi, "A heuristic search for good LDPC codes at short block lengths," in *Proc. Int. Conf. Commun.*, Helsinki, Finland, Nov. 2001, vol. 1, pp. 41–44.
- [36] R. A. Bailey and P. J. Cameron, *Latin Squares: Equivalents and Equivalence*. [Online]. Available: <http://designtheory.org/>
- [37] B. D. McKay and E. Rogoyski, "Latin squares of order 10," *Electron. J. Combinatorics*, vol. 2, no. 3, pp. 1–4, 1995.
- [38] B. D. McKay, *Latin Squares*. [Online]. Available: <http://cs.anu.edu.au/~bdm/data/latin.html>
- [39] D. MacKay, *Online Database of Low-Density Parity-Check Codes*. [Online]. Available: wol.ra.phy.cam.ac.uk/mackay/codes/data.html
- [40] N. Bonello, R. Zhang, S. Chen, and L. Hanzo, "Channel code-division multiple access and its multilevel-structured LDPC-based instantiation," *IEEE Trans. Veh. Technol.*, vol. 58, no. 5, pp. 2549–2553, Jun. 2009.



Nicholas Bonello received the B.Eng. degree (Hons.) in electrical engineering from the University of Malta, Tal-Qroqq, Malta, in 2004 and the M.Sc. degree in radio-frequency communications systems and the Ph.D. degree in wireless communications from University of Southampton, Southampton, U.K., in 2006 and 2009, respectively.

He is currently with the School of Electronics and Computer Science, University of Southampton. His primary research interests include fixed-rate and rateless error-correction techniques, signal process-

ing, and statistics.



Sheng Chen (M'90–SM'97–F'08) received the B.Eng. degree in control engineering from the East China Petroleum Institute, Dongying, China, in 1982, the Ph.D. degree in control engineering from the City University, London, U.K., in 1986, and the D.Sc. degree from the University of Southampton, Southampton, U.K., in 2005.

Since 1999, he has been with the School of Electronics and Computer Science, University of Southampton. He previously held research and academic appointments with the University of Sheffield, Sheffield, U.K.; the University of Edinburgh, Edinburgh, U.K.; and the University of Portsmouth, Portsmouth, U.K. He is the author of more than 280 research papers. In the database of the world's most highly cited researchers in various disciplines, compiled by Institute for Scientific Information (ISI) of the U.S., he is on the list of the highly cited researchers in the engineering category; see <http://www.ISIHighlyCited.com>. His recent research includes adaptive signal processing, wireless communications, modeling and identification of nonlinear systems, neural network and machine learning, finite-precision digital controller design, evolutionary computation methods, and optimization.



Lajos Hanzo (F'04) received the Dipl.Ing. (Master's) and Ph.D. degrees from the Technical University of Budapest, Budapest, Hungary, in 1976 and 1983, respectively, and the D.Sc. degree from the University of Southampton, Southampton, U.K., in 2004.

During his career in telecommunications, he has held various research and academic posts in Hungary, Germany, and the U.K. Since 1986, he has been with the School of Electronics and Computer Science, University of Southampton, where he holds the Chair in telecommunications. He is an enthusiastic supporter of industrial and academic liaison, and he offers a range of industrial courses. Currently, he heads an academic research team, working on a range of research projects in the field of wireless multimedia communications sponsored by industry, the Engineering and Physical Sciences Research Council, U.K., the European IST Program, and the Mobile Virtual Centre of Excellence, U.K. He is the author or a coauthor of 17 books, totaling 10 000 pages on mobile radio communications, and about 900 research papers.

Dr. Hanzo is a Fellow of the Royal Academy of Engineering and the Institute of Engineering of Technology (IET). He is an IEEE Distinguished Lecturer. He is the Editor-in-Chief of the IEEE Press and an editorial board member of the PROCEEDINGS OF THE IEEE, as well as a Governor of both the IEEE Vehicular Technology and IEEE Communications Societies. He has acted as the Technical Program Committee Chair of numerous major IET and IEEE conferences, has presented various keynote lectures, and has received a number of distinctions.