Combinatorial Constructions of Low-Density Parity-Check Codes for Iterative Decoding

Combinatorial Constructions of Low-Density Parity-Check Codes for Iterative Decoding

Bane Vasic, *Senior Member, IEEE,* and Olgica Milenkovic, *Member, IEEE*

***Abstract*—This paper introduces several new combinatorial constructions of low-density parity-check (LDPC) codes, in contrast to the prevalent practice of using long, random-like codes. The proposed codes are well structured, and unlike random codes can lend themselves to a very low-complexity implementation. Constructions of regular Gallager codes based on cyclic difference families, cycle-invariant difference sets, and affine 1-configurations are introduced. Several constructions of difference families used for code design are presented, as well as bounds on the minimal distance of the codes based on the concept of a generalized Pasch configuration.**

***Index Terms*—Cyclic difference families, iterative decoding, low-density parity-check (LDPC) codes, Pasch configurations.**

I. Introduction

AFTER the discovery by MacKay and Neal [1] that long Gallager codes [2] can achieve near-optimum performance when used for transmission over white additive Gaussian noise (AWGN) channels, it became a challenge to construct codes that would come as close as possible to the Shannon limit [3]. In the past few years, several low-density parity-check (LDPC) codes were designed with performances very close to this limit [4], [5]. Also, a significant insight into iterative decoding was gained due to its interpretation in terms of message passing and belief propagation in graphical models [6], as described, among others, by Kschischang *et al.* [7], [8], and McEliece *et al.* [9]. The graphical model that provides a natural setting in which to describe message passing was introduced by Tanner [10] and reintroduced by Wiberg *et al.* [11]. Although these requirements related to error performance are important, complexity issues tend to dominate system architecture and design considerations, especially for extremely high-speed applications such as magnetic recording and optical communications. Iterative decoders proposed so far have very high hardware complexity and are incapable of operating at rates above 1 Gb/s, the speed of electronics in the next generation of these channels. The high complexity of the proposed schemes is a direct consequence of the fact that for random codes a large amount of information is necessary to specify positions of the nonzero elements in a parity-check matrix. In this paper, we introduce well-structured LDPC codes, a concept opposed to the prevalent practice of using random-like code constructions, with the exception of the results by Kou, Lin, and Fossorier [12], [13], Tanner, Sridhara, and Fuja [14], Rosenthal and Vontobel [15], and Johnson and Weller [16]. The parity-check matrices of our codes are completely determined by a small set of parameters, and can lend themselves to very low complexity implementations.

Although bipartite graphs are quite useful for visualizing message passing, they are not very convenient for code design. Several constructions presented in this paper are purely combinatorial and based on balanced incomplete block designs (BIBDs) [17], extensively studied in connection with a large number of problems in applied mathematics and communication theory [18]. More specifically, our codes are constructed from the incidence matrix of $2-(v,c,1)$ BIBDs [19]–[21], where v corresponds to the number of parity bits, and c corresponds to the column weight of the regular parity-check matrix. The bipartite graphs of codes based on BIBDs have girth six, which is acceptable in high speed–low complexity applications where one cannot afford more than just a few iterations (up to five); additionally, BIBD-based codes can be designed to have very high rate (≥ 0.8) and relatively short length (less than 5000 bits). High rates are necessary to keep down the equalization loss (an important issue in recording channels), while short block lengths are required to maintain compatibility with existing data formats and to provide for a simpler system architecture [22]–[24].

In this paper we present three general code construction techniques. The first is based on difference families and the additive group of integers Z_v, and it makes use of the Bose [25], [26], Netto [27], and Buratti [28] difference families, leading to BIBDs for which $v \equiv 1 \bmod 6$, where v is a power of a prime. This family of codes gives the best tradeoff between code rate and code length. Unfortunately, there are not very many high-rate, short-length codes in this class, especially for large values of c. We also introduce the notion of m-fold cycle-invariant difference sets over Z_N (with $m>1$), and construct LDPC codes by using circulant permutation matrices determined by such difference sets. Codes constructed by using this new class of difference sets have both minimum distance and girth at least six, and exhibit excellent performance under iterative decoding. The third code construction, based on affine 1-configurations (integer lattices) is conceptually simple and produces a large family of codes, at the expense of the code rate. The number of parity bits of these codes is equal to qc, where q is a prime and the

Manuscript received August 23, 2001; revised January 30, 2004. This work was supported by the National Science Foundation under Grant CCR-0208597 and by Seagate Technology. The material in this paper was presented in part at the IEEE Information Theory Workshop, Cairns, Australia, September 2001.

B. Vasic is with the Department of Electrical and Computer Engineering, University of Arizona, Tucson AZ 85721 USA (e-mail: vasic@ece.arizona.edu).

O. Milenkovic is with the Department of Electrical and Computer Engineering, University of Colorado, Boulder CO 80303 USA.

Communicated by R. Urbanke, Associate Editor for Coding Techniques.

blocks are defined as lines of different slopes connecting points of a $q \times c$ integer lattice and the number of blocks is equal to q^2.

Designs and codes are very closely connected combinatorial entities, since one can be used to construct the other. For example, codewords of fixed weight in many codes, including the Golay code and the class of quadratic residue codes, support designs (see, for example, [29] and [30]). On the other hand, the incidence matrix of a design defines a nonlinear code if the rows of the matrix are viewed as codewords [31]. Designs have also been successfully used for constructing LDPC codes. A construction method for LDPC codes, based on Euclidean and projective geometries (subclasses of BIBDs) was recently proposed by Kou *et al.* [12], [13] (the last section of [13] mentions block designs, but the authors did not developed the idea in detail). Two other approaches for designing structured LDPC codes are a method based on using subgroups of the multiplicative group of a prime field [14] and a method based on Ramanujan graphs [15]. Recently, Johnson and Weller [16], as well as Vasic [32], presented almost identical approaches to the design of LDPC codes using combinatorial designs. The ideas in [32] and [33] were further developed in the series of papers [34]–[37]. MacKay and Davey [38] also used Steiner systems (a subclass of BIBDs) to construct Gallager codes. The approach presented in this paper differs from the aforementioned approaches in the sense that it is both conceptually simpler, and as will be shown, gives much simpler construction algorithms. The encoding complexity of our BIBD codes is extremely low and is basically determined by the size of a cyclic difference family upon which a block design is based, or by the "vertical" dimension of the lattice in the case of lattice construction. We will also give the bounds on the minimum distance of the BIBD codes. Bounds on the minimal distance of regular Gallager codes with column weight three were first discussed by MacKay in [4]. Here, we present tighter bounds for this case, as well as new bounds for the case $c > 3$. Our bounds are derived using simple combinatorial arguments, and are solely a function of the parity-check matrix column weight. Tanner's bounds on the minimum distance based on the eigenvalues of the product of the adjacency matrix of the code graph and its transpose [39] can be shown to be trivial for the codes presented in this paper.

The outline of the paper is as follows. Section II introduces BIBDs and describes their relation to bipartite graphs and parity-check matrices of regular Gallager codes. Section III describes a code-design method involving $2-(v,3,1)$ systems (Steiner triple systems), derived from cyclic difference families, and presents bounds on the minimum distance of the codes, derived by using the concept of a Pasch configuration. Section IV presents three constructions of cyclic difference families for $c = 3$, 4, and 5, and gives a list of known infinite families. Section V contains the description of several new construction techniques for designs that can be used to derive high-rate LDPC codes. Additionally, this section includes the description of a code construction based on circulant permutation matrices and a novel class of combinatorial objects termed cycle-invariant difference sets (CIDSs). The description of another novel BIBD construction based on integer lattices is given in Section VI, while Section VII contains the results of computer simulations. Concluding remarks are presented in Section VIII.

II. Balanced Incomplete Block Designs and Their Equivalence With Bipartite Graphs

In this section, we introduce the definitions and the notation used throughout the paper. A BIBD with parameters (v, c, λ) is an ordered pair (V, B), where V is a v-element set and B is a collection of b c-subsets[1] of V, called blocks, such that every element of V is contained in exactly r blocks and every 2-subset of V is contained in exactly λ blocks. Notice that $cb = rv$, so that the parameter r is uniquely determined by the remaining parameters of the design. A design for which every block contains the same number c of points, and every point is contained in the same number r of blocks is called a *tactical configuration*. Hence, BIBDs are tactical configurations. The notation BIBD (v, c, λ) will henceforth be used to specify a BIBD on v points, with block size c, and index λ. A BIBD with block size $c = 3$ is called a Steiner triple system (henceforth, STS). A BIBD is resolvable if there exists a partition of its block set B into parallel classes, each of which partitions the set V. *R*esolvable STS with index $\lambda = 1$ are called *Kirkman* systems [40]. These combinatorial objects have been intensively studied in the combinatorial literature, and some construction methods for them are described in [19], [20], [41]. Besides resolvable STSs, we will also use λ–configurations for LDPC code design. A λ–*configuration* is an incidence structure of v points and b blocks such that each block contains c points, each point is incident with r blocks, and two different points are contained in *at most* λ blocks.

Definition 2.1: The point-block incidence matrix of a (V, B) design is a $v \times b$ matrix $A_{p,b} = (a_{ij})$, in which $a_{ij} = 1$ if the ith element of V occurs in the jth block of B, and $a_{ij} = 0$ otherwise. The block-point matrix $A_{b,p}$ is the transpose of the point-block incidence matrix.

Example 2.1: The collection $B = \{B_1, B_2, \ldots, B_7\}$ of blocks $B_1 = \{0,1,3\}$, $B_2 = \{1,2,4\}$, $B_3 = \{2,3,5\}$, $B_4 = \{3,4,6\}$, $B_5 = \{0,4,5\}$, $B_6 = \{1,5,6\}$, and $B_7 = \{0,2,6\}$ is a BIBD$(7,3,1)$ system or an STS with $v{=}7$ and $b{=}7$. The point-block incidence matrix is of the form

$$A_{p,b} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}.$$

If $b = v$, and hence $r = c$, the BIBD is *symmetric*, and is for $c \geq 3$ equivalent to a finite projective plane [42].

If one thinks of points as parity-check equations and of blocks as bits of a linear block code, then $A_{p,b}$ defines a parity-check matrix H of an LDPC code [2]. The column weight of H is c,

[1]The standard notation for the subset size is k, but we will use c instead; in this way, we will avoid the subset size being confused with the dimension of the code, typically denoted by k.

the row weight is r, and the code rate is $R = (b - \mathrm{rank}\,(H))/b$, where the rank is evaluated over the field GF(2). Since the rank of H is usually quite hard to determine, one can bound the rate of an LDPC code based on a $2-(v, c, \lambda)$ design as follows:

$$R \geq \frac{\lambda\frac{v(v-1)}{c(c-1)} - v}{\lambda\frac{v(v-1)}{c(c-1)}}. \qquad (1)$$

It should be noticed that the bound given by (1) is generally loose. For example, for the case of codes constructed from projective planes, the bound is trivially equal to zero, while the actual rate of the codes is quite high (see, e.g., [12], [13]). Therefore, for a given designed code rate, many BIBD codes will have a larger dimension than predicted by (1).

Remark 2.1: A more precise characterization of the rank (and "p-rank") of the incidence matrix of 2-designs is given by Hamada [43].

If every 2-element subset of V is contained in exactly λ blocks, the underlying design is known as a 2-*design*. In this paper, we will restrict our attention to 2-designs only; more specifically, to 2-designs with index $\lambda = 1$. We will henceforth refer to them just as BIBDs. The constraint $\lambda = 1$ implies that no more than one block contains the same pair of points, or equivalently, that no pair of columns of the parity-check matrix contains two ones at the same positions.

From the lower bound on the code length b for a given code rate R for codes based on the BIBD $(v, c, 1)$ systems one can see that if the required code rate is higher than $9/10$ (which is typically the rate of interest in recording and optical communications channels), it is impossible to construct codes shorter than approximately 500 bits. Issues regarding the existence and the number of BIBD families for a given set of parameters are addressed in Section V. The construction of maximum-rate BIBD codes with $c = 2$ is trivial and reduces to finding K_v, the complete graph [42].

To visualize the decoding algorithm for LDPC codes, the parity-check matrix is represented as a bipartite graph with two types of vertices [8]–[10]. The first subset of vertices (B) is comprised of code bits, and the second subset of vertices is comprised of parity-check equations (V). An edge between a bit and an equation exists if the bit is involved in the check. Translated to the terminology of block designs, the two sets of vertices correspond to B and V, respectively, and an edge between $u \in V$ and $B_j \in B$ exists if and only if $u \in B_j$. Hence, the parity-check matrix of the LDPC code is the block-point incidence matrix $A_{b,p}$.

Example 2.2: The bipartite graph representation of the Steiner $(7, 3, 1)$ system whose incidence matrix is given in Example 2.1 is shown in Fig. 1.

In order to have good error-control characteristics, it is desirable to have each bit "checked" by as many equations as possible, but due to the iterative character of the decoding algorithm it is also important that the bipartite graph does not contain short cycles. In other words, the *girth* of the graph (i.e., the length of the shortest cycle) must be large. The girth constraint is related

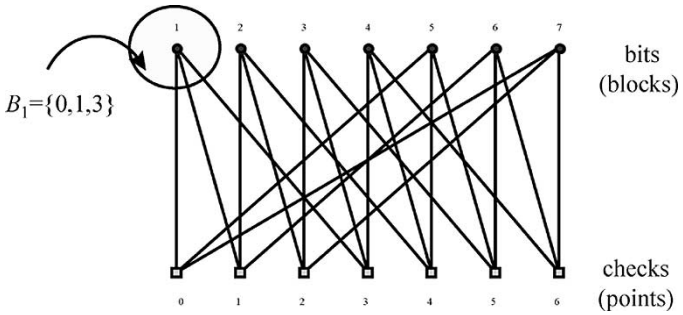

Fig. 1. The bipartite graph representation of the Steiner $(7, 3, 1)$ system.

to the constraint that every 2-element subset of V is contained in as few blocks as possible. On the other hand, check nodes should not have too large a degree in order to allow for efficient iterative decoding. These two requirements are contradictory, and the tradeoff is especially difficult when one is interested in constructing codes with short length and high rate.

Remark 2.2: The constraint $\lambda = 1$ imposed on BIBDs implies that there are no cycles of length four in the bipartite graph of the code.

III. Code Design Based on Difference Families

As pointed out in Section III, BIBDs offer a combinatorial tool for designing codes without short cycles. In this section, we present several simple construction of STSs using difference families of Abelian groups.

Definition 3.1: Let V be a finite additive Abelian group of order v. Then t c-element subsets of V, $B_i = \{b_{i,1}, \ldots, b_{i,c}\}$, $1 \leq i \leq t$, form a (v, c, λ) *difference family* (henceforth, DF) if every nonzero element of V can be represented in exactly λ ways as a difference of two elements within the same member of the family. In other words, every nonzero element of V occurs λ times among the differences $b_{i,m} - b_{i,n}$, $1 \leq i \leq t$, $1 \leq m, n \leq c$. The sets B_i are called *base blocks*. If V is isomorphic to Z_v, the additive group of integers modulo v, then the corresponding (v, c, λ) DF is called a cyclic difference family (henceforth, CDF).

Example 3.1: The block $B_1 = \{0, 1, 3\}$ is a base block of a $(7, 3, 1)$ CDF. To illustrate this, we create an array $\Delta^{(1)} = (\Delta_{i,j})$, of differences $\Delta^{(1)}_{i,j} = b_{1,i} - b_{1,j}$

$$\Delta^{(1)} = \begin{bmatrix} 0 & 6 & 4 \\ 1 & 0 & 5 \\ 3 & 2 & 0 \end{bmatrix}.$$

As can be seen, each nonzero element of Z occurs exactly once in Δ.

Example 3.2: The blocks $B_1 = \{0, 1, 4\}$ and $B_2 = \{0, 2, 7\}$ are the base block of a $(13, 3, 1)$ CDF based on the group $V = Z_{13}$, since the nonzero elements of the difference arrays

$$D^{(1)} = \begin{bmatrix} 0 & 12 & 9 \\ 1 & 0 & 10 \\ 4 & 3 & 0 \end{bmatrix}, \qquad D^{(2)} = \begin{bmatrix} 0 & 11 & 6 \\ 2 & 0 & 8 \\ 7 & 5 & 0 \end{bmatrix}$$

TABLE I
THE ORBITS OF BASE BLOCKS $\{0,1,4\}$ AND $\{0,2,7\}$ IN A $(13,3,1)$ BIBD

B_1 orbits			B_2 orbits		
$b_{11}+g$	$b_{12}+g$	$b_{13}+g$	$b_{21}+g$	$b_{22}-g$	$b_{23}+g$
0	1	4	0	2	7
1	2	5	1	3	8
2	3	6	2	4	9
3	4	7	3	5	10
4	5	8	4	6	11
5	6	9	5	7	12
6	7	10	6	8	0
7	8	11	7	9	1
8	9	12	8	10	2
9	10	0	9	11	3
10	11	1	10	12	4
11	12	2	11	0	5
12	0	3	12	1	6

formed according to ($\Delta_{i,j}^{(1)} = b_{1,i} - b_{1,j}$ and $\Delta_{i,j}^{(2)} = b_{2,i} - b_{2,j}$), are all different.

Definition 3.2: If G is a group that acts on a set X, then the set $O_x = \{gx : g \in G\}$, $x \in X$, is called the *orbit* of x. For the case that G is a cyclic group of order v and $X = B$, where B is the set of all base blocks of a CDF, a BIBD can be defined as the union of the orbits of B. If the number of base blocks is t, the number of blocks in a BIBD is $b = tv$.

Generally, given a (v, c, λ) CDF with base blocks $B_i = \{b_{i,1}, \ldots, b_{i,c}\}$, $1 \leq i \leq t$, the point-block incidence matrix of the BIBD can be written in the form

$$H = [H_1 H_2 \cdots H_t] \tag{2}$$

where each submatrix is of dimension $v \times v$. The orbits of the base block B_j are represented by the positions of nonzero elements in the submatrix H_j.

Example 3.3: The blocks $B_1 = \{0,1,4\}$ and $B_2 = \{0,2,7\}$ are the base block of a $(13,3,1)$ CDF of the group Z_{13}. The orbits of B_1 and B_2 are given in Table I.

The parity-check matrix corresponding to the $(13,3,1)$ BIBD in Table I is given by

$$H = \left[\begin{array}{c:c} 1000000001001 & 1000001000010 \\ 1100000000100 & 0100000100001 \\ 0110000000010 & 1010000010000 \\ 0011000000001 & 0101000001000 \\ 1001100000000 & 0010100000100 \\ 0100110000000 & 0001010000010 \\ 0010011000000 & 0000101000001 \\ 0001001100000 & 1000010100000 \\ 0000100110000 & 0100001010000 \\ 0000010011000 & 0010000101000 \\ 0000001001100 & 0001000010100 \\ 0000000100110 & 0000100001010 \\ 0000000010011 & 0000010000101 \end{array}\right]$$

and contains only columns of weight three. The CDF codes described above have a quasi-cyclic structure similar in form to the self-orthogonal quasi-cyclic structure of Townsend and Weldon's [44] codes and Weldon's difference set codes [45]. Each orbit of a base block in the design corresponds to one circulant submatrix in the quasi-cyclic parity-check matrix of the code. Hence, the self-orthogonal codes of Townsend and Weldon represent a special class of LDPC codes.

The codes based on Z_v are of special interest because they are conceptually very simple and have a structure that can be easily implemented in hardware. Notice also that for a given constraint (v, c, λ) the CDF-based construction maximizes the code rate (see (1)), because for a given v the number of blocks is maximized. The code rate is independent from the choice of the representation of the underlying group as long as the base blocks belong to a CDF. Other choices for the groups may lead to similar or better codes, but they are not considered in this paper.

The rest of this section is devoted to establishing bounds on the minimum distance $(d_{\min})$ of BIBD-based codes constructed from CDFs. The minimum distance $d_{\min}$ determines code performances under maximum-likelihood (ML) decoding at high signal-to-noise ratios (SNRs), the region where most of the practical systems typically operate. Additionally, small minimum distance may result in undesirable high error-floors. Bounds for $d_{\min}$ of Gallager codes with column weight $c = 3$ were first derived in [38]. Another, more general technique for establishing a lower bound for $d_{\min}$ is due to Tanner [39]. It pertains to an arbitrary linear code with parity-check matrix H, represented by a bipartite graph, and is based on combinatorial optimization. The calculation of the bound involves finding the second largest eigenvalue of the matrix HH^T. For the codes based on CDF, these eigenvalues can be found in closed form, but they result in trivial bounds (see Lemma 3.1). Here, we present tight bounds for $c = 3$ and also establish bounds for $c > 3$. These combinatorial bounds are derived using simple counting arguments, and are solely function on the column weight of H.

Lemma 3.1: Let H be a parity-check matrix of the form given by (2). Then the largest eigenvalue of HH^T is equal to $\mu_1 = t(v + \hat{r} - 1)$, while all the remaining eigenvalues are equal to $\mu_2 = t(\hat{r} - 1)$. In the last expression, $\hat{r}$ denotes the number of times each point occurs in different blocks belonging to a given H_i, i.e., $2\hat{r}\,t = v - 1$ for $c = 3$, $\lambda = 1$.

Proof: The proof is given in **Appendix A**.

Based on Lemma 3.1, one can use Tanner's lower bounds [39] to find bounds on the minimum distance of the CDF codes. Tanner's bounds are expressed in terms of the second largest

eigenvalue of HH^T (see also [61]), and for the case of interest are of the form

$$d \geq \frac{vt\,(6-\mu_2)}{3\,\hat{r}\,t-\mu_2}$$
$$d \geq \frac{2\,v\,t\,(4+\hat{r}\,t-\mu_2)}{\hat{r}\,t(3\,\hat{r}\,t-\mu_2)}\,.$$

The first bound is trivial for $\mu_2 \geq 6$, i.e., for $t(\hat{r}-1) \geq 6$. The second bound is trivial for $\mu_2 \geq 4+\hat{r}\,t$, i.e., for $t\,(\hat{r}-1) \geq 4+\hat{r}\,t$.

A general, nontrivial lower bound on $d_{\min}$ for codes based on BIBDs with block size c can be easily obtained by using two different arguments. The first one is based on the idea of majority logic decoding [31]. A code is one-step majority logic decodable if for every bit there exists a set of L parity-check equations that are orthogonal on that bit. In this context, the orthogonality condition imposes the requirement that each of the check equations include the bit under consideration, and that no other bit is checked more than once by any of the equations. If a code is one-step majority decodable, then the minimum distance of the code is at least $L+1$. From the described construction of the LDPC codes based on BIBDs, it follows that $L=c$ and that therefore $d_{\min} \geq c+1$. The same result can be obtained by considering the Tanner graph of the code as follows. Start by selecting a bit node and assume that its value is one. It has c check nodes at distance one and $c(r-1)$ bit nodes at distance two. Since $\lambda=1$, the girth of the code graph is at least six, and therefore all bit nodes at distance two are distinct. However, to satisfy the c parity-check node equations at distance one, at least c of the bit nodes at distance two must have the value one. Therefore, the minimum weight of a codeword has to be at least $c+1$.

In order to improve the previously derived lower bound on $d_{\min}$, we need to define a generalized Pasch configuration.

Definition 3.3: An (m,n) *configuration* in a BIBD is a subset of m blocks of B, whose union is an n-element subset of V. A *Pasch configuration* or a *quadrilateral* is a $(4,6)$ configuration in a STS.

Example 3.4: Consider an STS and a subset of six points, $\{1,2,3,4,5,6\}$. Then the set of blocks

$$\{\{1,2,3\},\{1,4,5\},\{2,4,6\},\{3,5,6\}\}$$

forms a Pasch configuration.

Using the definition of a Pasch configuration for $c=3$, we can now define a generalized Pasch configuration as a $(c, c(c+1)/2)$ configuration of a $(v,c,1)$ BIBD. A generalized Pasch configuration can always be visualized by creating a $(c+1)\times c$ array Λ_c of $c(c+1)/2$ points $\{1,2,\ldots,c(c+1)/2\}$, as shown at the bottom of the page. It is easy to see that no pair of points occurs more than once in the same row. Hence, the rows of Λ_c represent blocks of a $(c+1, c(c+1)/2)$ configuration.

Example 3.5: A generalized Pasch configuration of a BIBD with $c=5$ and $\lambda=1$ can be represented by the rows of the matrix Λ_5 shown as follows:

$$\Lambda_5 = \begin{bmatrix} \ddots & \Omega \\ \Omega^T & \ddots \end{bmatrix} = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 \\ 1 & 6 & 7 & 8 & 9 \\ 2 & 6 & 10 & 11 & 12 \\ 3 & 7 & 10 & 13 & 14 \\ 4 & 8 & 11 & 13 & 15 \\ 5 & 9 & 12 & 14 & 15 \end{bmatrix}.$$

Lemma 3.2: The rows of Λ_c define the *smallest configuration* of a $(v,c,1)$ BIBD in which each point occurs exactly twice.

Proof: The proof is given in **Appendix B**.

Remark 3.2: The existence of a Pasch configuration in an STS is directly related to the minimum distance of the derived LDPC code. If an STS has at least one Pasch configuration, each point occurs exactly twice in the configuration, and therefore there exist four linearly dependent columns in H. This implies that $d_{\min}=4$. If the triple system is Pasch-free, then the minimum distance of the corresponding LDPC code is equal to six. For example, the Bose construction to be described in Section V produces LDPC codes with minimum distance $d_{\min}=6$. The CDFs following from the first construction by Netto result in codes that have $d_{\min}=4$, while the ones obtained from the second construction by Netto result in codes with $d_{\min}=6$ (see Section IV). This result can be extended to arbitrary designs. If a (v,c,λ) CDF and the underlying design contain a generalized Pasch configuration, then $d_{\min}=c+1$, and if the underlying design does not contain a generalized Pasch configuration then $d_{\min} \geq c+2$.

Example 3.6: For the example in Table I, $B_1=\{0,1,4\}$, $B_2=\{0,2,7\}$, and

$$\{B_1+b_{2,m} : 1 \leq m \leq 3\} = \{\{0,1,4\},\{2,3,6\},\{7,8,11\}\}$$
$$\{B_2+b_{1,m} : 1 \leq m \leq 3\} = \{\{0,2,7\},\{1,3,8\},\{4,6,11\}\}$$

and each of the points $\{0,1,2,3,4,6,7,8,11\}$ occurs twice in this group of blocks. Hence, the minimum distance of the code constructed from this CDF is $d_{\min} \leq 6$. Furthermore, since the

$$\Lambda_c = \begin{bmatrix} 1 & 2 & 3 & \ldots & c-1 & c \\ 1 & c+1 & c+2 & \ldots & 2c-2 & 2c-1 \\ 2 & c+1 & 2c & \ldots & 3c-(1+2)-1 & 3c-(1+2) \\ 3 & c+2 & 2c & \ldots & 4c-(1+2+3)-1 & 4c-(1+2+3) \\ \cdot & \cdot & \cdot & \ldots & \cdot & \cdot \\ \cdot & \cdot & \cdot & \ldots & \cdot & \cdot \\ \cdot & \cdot & \cdot & \ldots & \cdot & \cdot \\ c-1 & 2c-2 & 3c-(1+2)-1 & \ldots & (c-1)c-(1+2+\cdots+(c-1))-1 & c^2-(1+2+\cdots+(c-1)) \\ c & 2c-1 & 3c-(1+2) & \ldots & (c-1)c-(1+2+\cdots+(c-2)) & c^2-(1+2+\cdots+(c-1)) \end{bmatrix}.$$

TABLE II
THE ONLY 2-$(15, 3, 1)$ PASCH-FREE DESIGN

{0, 1, 2}	{0, 3, 4}	{0, 5, 6}	{0, 7, 8}	{0, 9,10}
{0,11,12}	{0,13,14}	{1, 3, 5}	{1, 4, 7}	{1, 6, 8}
{1, 9,11}	{1,10,13}	{1,12,14}	{2, 3, 9}	{2, 4, 6}
{2, 5,10}	{2, 7,14}	{2, 8,12}	{2,11,13}	{3, 6,11}
{3, 7,12}	{3, 8,13}	{3,10,14}	{4, 5,13}	{4, 8, 9}
{4,10,12}	{4,11,14}	{5, 7,11}	{5, 8,14}	{5, 9,12}
{6, 7,10}	{6, 9,14}	{6,12,13}	{7, 9,13}	{8,10,11}

TABLE III
SOME SMALL PRIME $(v, 3, 1)$ CYCLIC DIFFERENCE FAMILIES

v	B_1	B_2	B_3	B_4	B_5	B_6	B_7
7	0 1 3						
13	0 1 4	0 2 7					
19	0 1 4	0 2 9	0 5 11				
31	0 1 12	0 2 24	0 3 8	0 4 17	0 6 16		
37	0 1 3	0 4 26	0 5 14	0 6 25	0 7 17	0 8 21	
43	0 1 3	0 4 9	0 6 28	0 7 23	0 8 33	0 11 30	0 12 26

sum of all rows of the parity-check matrix is the all-one codeword, the corresponding LDPC code is even (i.e., all codewords of the code have even weight). Hence, since no two columns are identical, the minimum distance of the code also satisfies $d_{\min} \geq 4$.

Remark 3.3: Pash-free STSs are extremely rare. In other words, most known STSs will result in codes with $d_{\min} = 4$. In [19], Colbourn lists 80 nonisomorphic 2-$(15, 3, 1)$ designs, only one of which is Pasch-free (see Table II) (this particular design is also described in [19]). In their recent work, Ling *et al.* [46] present a construction techniques for anti-Pasch STSs. However, these designs do not necessarily lead to codes with quasi-cyclic structure. The existing connection between Pasch configurations in designs and bounds on the minimum distance $d_{\min}$ calls for investigating the effects of other types of configurations on $d_{\min}$ and the girth of a bipartite graph. For example, Beezer [47] defines the girth of a t-design through the concept of *Erdös configurations*, a generalization of the Pasch configurations.

On the other hand, one can easily establish the following upper bound for the minimum distance of a code constructed from a CDF BIBD.

Theorem 3.1: For a CDF BIBD code with column weight c, one has $d_{\min} \leq 2c$.

Proof: The proof is given in **Appendix C**.

IV. CONSTRUCTION OF CDFS BY NETTO AND BURATTI

It is straightforward to construct a BIBD design once the CDF is known. However, finding the CDF is a much more complex problem and it is solved only for certain values of v, c, and λ. In this section, we will present several constructions for CDFs. One of the first known constructions due to Bose [25], [26] is described next.

The idea for the construction is based on the use of combinatorial objects known as mixed difference sets. Mixed difference sets are formed by taking elements from multiple copies of the additive group (say, Z_N) of the elements of the design, distinguished by different subscripts [50]. One can show that the sets

$$\{0_1, 0_2, 0_3\}, \{1_i, (2u)_i, 0_{i+1}\}, \{2_i, (2u-1)_i, 0_{i+1}\}, \ldots, \{u_i, (u+1)_i, 0_{i+1}\}, \qquad 1 \leq i \leq 3$$

where the elements $u \in Z_N$ are taken modulo $2u+1$, and the subscripts are taken modulo three, form a DF.

Most constructions of CDFs that followed Bose's work are based on the use of finite fields. The following constructions belong to this category.

A. The First Construction by Netto

This construction is applicable for $c = 3$ and v a power of a prime of the form $v \equiv 1 (\mathrm{mod}\ 6)$ [27]. For v a power of a prime, Z_v is to be replaced by GF(v), the Galois field of order v. Let Ψ be the elements of the multiplicative group of the field GF(v). Let ω be a primitive element of the field, and hence a generator of the multiplicative group Ψ[48]. Write v as $v = 6t+1, t \geq 1$, and for $d|v-1$, let Ψ^d be the group of dth powers in GF(v), and let $\omega^i\Psi^d$ be a coset of dth powers of ω^i. Then the set $\{\omega^i\Psi^{2t} \,|\, 1 \leq i \leq t\}$ defines an STS difference family with parameters $(6t+1, 3, 1)$ [27] (see also [49]). The base blocks are typically given in the form $\{0, \omega^i(\omega^{2t}-1), \omega^i(\omega^{4t}-1)\}$ rather than as $\{\omega^i, \omega^{i+2t}, \omega^{i+4t}\}$.

Example 4.1: As an illustration, the difference families for some small primes are given in Table III.

An alternative combinatorial method for constructing a CDF with parameters $(6t+1, 3, 1)$, that can also be extended to the $(6t+3, 3, 1)$ case, was proposed by Rosa in [41]. Furthermore, in [45], Weldon presents a list of constructions for CDFs with the same set of parameters as described above, as well as a list of the resulting codes of rate $1/2$.

B. The Second Construction by Netto

This construction can be used to create CDFs when the number of points v is a power of a prime and $v \equiv 7 (\mathrm{mod}\ 12)$ [27]. As in the first construction, let ω be a generator of the multiplicative group of the field GF(v) and let Ψ^d be the group of dth powers in GF, (v). Then the set $\{\omega^{2i}\Psi^{2t} \,|\, 1 \leq i \leq t\}$ defines base blocks that are also called a Netto triple system. One more design construction technique is known as Netto's construction. It is not directly based on CDFs, but is strongly related to the previously described constructions.

C. The Third Construction by Netto

Let $v = p^n$, where p is a prime of the form $p \equiv 7 (\mathrm{mod}\ 12)$. Let $e1$ and $e2$ be two sixth roots of unity in GF(v). It is straightforward to verify that $e1 + e2 = e1 \cdot e2 = 1$ and that $e1^2 = -e2$ and $e2^2 = -e1$, and that neither $e1$ and $e2$ are perfect squares in GF(v). Define a relation $a \rightarrow b$ as follows: $a \rightarrow b$ holds if and only if $b - a$ is a nonzero square in GF(v). Then exactly one of the relations is true: $a \rightarrow b$ or $b \rightarrow a$. Define a function $f(a, b)$ as $f(a, b) = e1 \cdot a + e2 \cdot b$, for all pairs (a, b) such that $a \rightarrow b$ holds. A Netto system [27] obtained by utilizing the function f is an STS with points and blocks $V =$ GF(v), $B = \{(a, b, c) : a \rightarrow b, c = f(a, b)\}$, repsectively.

Netto systems are of interest because of the following property.

TABLE IV
BASE BLOCKS PARAMETER FOR BURATTI CDF

$c=4$		$C=5$		$C=4$		$c=5$	
v	b	v	b	v	b	v	b
37	-13	461	-8	613	5	1741	-46
61	-5	1021	50	661	-6	1861	19
157	9	1061	-26	853	-18	2621	-327
349	14	1601	268	877	-5	2861	91
373	-4	1621	-209	937	-12	3301	-345
397	18	1721	-268	997	22	3461	19

Theorem 4.1: Netto triple systems for v a power of a prime of the form $v \equiv 19 (\mathrm{mod}\ 24)$ and the third Netto triple system described above are Pasch-free.

Proof: The proofs can be found in [19] (see also [51]).

Consequence: The Netto triple systems described in Theorem 4.1 achieve the upper bound on minimum distance, i.e., $d_{\min} = 6$.

Example 4.2: The base blocks of the Netto triple system difference family for $v = 43$, $\omega = 3$ are: $B_1 = \{0, 2, 14\}$, $B_2 = \{0, 18, 40\}$, $B_3 = \{0, 16, 33\}$, $B_4 = \{0, 15, 39\}$, $B_5 = \{0, 6, 7\}$, $B_6 = \{0, 11, 20\}$, and $B_7 = \{0, 8, 13\}$. The resulting code is quasi-cyclic, has $d_{\min} = 6$, length $b = 301$, and $R \geq 0.857$.

D. The Construction by Buratti ($c = 4$ and $c = 5$)

Buratti's method [28] gives CDFs with v points and block size $c = 4$, provided that v is a prime of the form $v = 12t + 1$. The CDF is a set $\{\omega^{6i} B : 1 \leq i \leq t\}$, where base blocks have the form $B = \{0, 1, b, b^2\}$, and ω denotes a primitive element in GF(v). The numbers $b \in$ GF(v) for several different values of v are given in Table IV. Similarly, for $c = 5$, the CDF is given by $\{\omega^{10i} B : 1 \leq i \leq t\}$, where $B = \{0, 1, b, b^2, b^3\}$, and $b \in$ GF$(20t + 1)$.

If a difference set family with parameters (v, c, t) exists, it is possible to construct difference set families with parameters (lv, c, lt), where $\gcd(l, (c-1)!) = 1$, and $v \not\equiv 0 \bmod c$, by using the following simple augmentation method due to Colbourn and Colbourn [51]: For each block $B_i = \{b_{i,1}, \ldots, b_{i,c}\}$, first subtract the smallest element from all elements of the set to obtain $B_i = \{0, b'_{i,1}, \ldots, b'_{i,c-1}\}$. Then form the difference sets

$$\{0, b'_{i,1} + mv, b'_{i,2} + 2mv, \ldots, b'_{i,c-1} + (c-1)mv\}, \quad 0 \leq m < l$$

with addition performed modulo lv.

V. OTHER RELATED CONSTRUCTION TECHNIQUES

A. Finite Euclidean and Finite Projective Geometries

The existence and construction of short designs ($b < 10^5$) is an active area of research in combinatorial mathematics (the handbook [19] edited by Colbourn and Dinitz is an excellent reference). [19, Table 2.3] gives a summary of known results concerning the existence of short designs. However, very often the construction of these designs is somewhat heuristic or works only for a given block size. In many cases, such constructionsgive a very small set of designs with parameters of practical interests. An important exception is the subclass of BIBDs called *infinite families* [19]. Infinite families of BIBDS include projective geometries, affine geometries, unitals, Denniston designs, as well as certain geometric equivalents of 2-designs [19]. The known infinite families of BIBDs are listed in Table V [19], [51]. For designs with a number of points that is a power of a prime, these families are known as finite Euclidean and finite projective geometries (for more details the reader is referred to [51]).

The first class of finite-geometry codes is comprised of codes with parity-check matrix defined as the point-line incidence matrix of finite geometries, such as the LDPC codes described by Kou, Lin, and Fossorier [13]. The second class of codes is obtained from algebraic curves in a projective plane. In a projective plane, an *algebraic curve* is a collection of points that satisfy a fixed homogeneous algebraic equation of some degree n, i.e., $f(x_0, x_1, x_2) = 0$. An algebraic curve is *irreducible* if $f(x_0, x_1, x_2)$ is an irreducible polynomial over the ground field GF(q). A line meets the curve in at most n points. A *conic* is an algebraic curve of degree two, or more specifically

$$f(x_0, x_1, x_2) = ax_0^2 + bx_1^2 + cx_2^2 + fx_1x_2 + gx_2x_0 + hx_0x_2 = 0.$$

A conic is *irreducible* if $f(x_0, x_1, x_2)$ is irreducible over the ground field GF(q). For $k > m$, an $\{k; m\}$-*arc* in PG$(2, q)$ is a set of k points such that no $(m+1)$ points lie on a line. A c-arc in PG$(2, q)$ is a set of c points such that no three points lie on the same line. A c-arc is *complete* if it is not properly contained in any $(c+1)$-arc. A line of the plane is said to be a *secant*, a *tangent*, or an *exterior line* with respect to the oval, if the number of common points of the line with the oval is 2,1, or 0, respectively. For a given value of q, $(q+1)$-arcs of PG, $(2, q$ odd$)$ are called *ovals*, and $(q+1)$-arcs of PG$(2, q$ even$)$ together with a nucleus point (a point for which every line incident to it is a tangent of the oval) are called *hyperovals*.

In PG $(2, 2^m)$, an *oval design* is the incidence structure with points comprised from the lines exterior to the oval and blocks specified by the points not on the oval. A block contains a point if and only if the corresponding exterior point lies on the exterior line. An oval design is a resolvable $2 - (s(s-1)/2, s/2, 1)$ Steiner 2-system, where $s = 2^m$. In the simulation result section, we will present the performance of the codes on a hyperoval constructed from a nondegenerate conic specified by the equation $x_0x_2 = x_1^2$. Similar results were presented in [55].

Unitals or Hermitian arcs are defined as follows. In PG$(2, q)$, q a square, a *Hermitian arc* is a $\left\{q\sqrt{q}+1; \sqrt{q}+1\right\}$-arc. The arc is constructed from an algebraic curve of order $\sqrt{q}+1$ such that $x_0^{\sqrt{q}+1} + x_1^{\sqrt{q}+1} + x_2^{\sqrt{q}+1} = 0$. The arc intersects any line of the plane at 1 or $\sqrt{q}+1$ points. A unital constructed from an algebraic curve of order $\sqrt{q}+1$ is a $(((\sqrt{q})^3+1, \sqrt{q}+1, 1))$ Steiner system. A code based on this unital is described in terms of the incidence matrix of the corresponding Steiner system. For q a power of 2, the rank of the incidence matrix is $(\sqrt{q})^3$, and for q a power of an odd prime, the rank of the incidence matrix is $((\sqrt{q})^2 - \sqrt{q} + 1)\sqrt{q}$. Such designs are treated in great detail by Assmus and Key and in [56] and in [57].

As it can be seen from Table V, the most known infinite families of designs do not offer sufficient flexibility in choosing the code length and column weight, especially for the high-rate and/or moderate codeword length cases. In this region, the

TABLE V
KNOWN INFINITE FAMILIES OF $2-(v,c,1)$ DESIGNS

c	V	Parameter	Name
q	q^n	dimension $n\geq 2$, q- power of a prime	Affine geometries
$q+1$	$(q^{n+1}-1)/(q-1)$	dimension $n\geq 2$, q- power of a prime	Projective geometries
$q+1$	q^3+1	q- power of a prime	Unitals
$q/2$	$q(q-1)/2$	q- power of two	Ovals
2^m	$2^m(2^s+1)-2^s$	$2\leq m<s$	Denniston designs

column weight must be small and one solution is to use a special class of short, cleverly constructed CDF as in [28]. Notice that in [13] different modifications (including code shortening, for example) of Euclidean and projective geometry codes are given so as to get a larger set of code parameters.

B. The Latin Square Construction by Bose [59]

In order to describe this construction method for STSs due to Bose, we will first define a special class of Latin squares. A Latin square of order m is an $m \times m$ array such that each row and column contains the symbols in $\{1,\ldots,m\}$ exactly once. A Latin square is idempotent if cell (i,i) contains symbol i, $1 \leq i \leq m$, and commutative if cells (i,j) and (j,i) contain the same symbol, $1 \leq i \leq m$. Let *SQ* be an idempotent and commutative Latin square of order $2m+1$, let $V = \{1,2,\ldots,2m+1\}$ and $P = V \times \{1,2,3\}$. Define a collection of triples T, so that T contains

- all triples of the form $\{(i,1),(i,2),(i,3)\}$, where $1 \leq i \leq 2m+1$;
- all triples of the form

$$\{(i,1),(j,1),(i\square j,2)\},\ \{(i,2),(j,2),(i\square j,3)\},\ \{(i,3),(j,3),(i\square j,1)\}$$

  where $1 \leq i < j \leq 2m+1$, and $i\square j$ denotes the entry of the Latin square SQ in row i and column j.

Then (P,T) is an STS of order $6m+3$. If the Latin square SQ is chosen in such a way that $2m+1 \not\equiv 0 \bmod 7$, and that $i\square j = (i+j)/2 \bmod 2m+1$, then the STS resulting from the Bose construction is Pasch-free. This follows from the fact that under the given conditions there are no Latin subsquares of order two in SQ and $x\square(x\square(x\square y)) = y$ cannot hold for distinct symbols x, y. For more details, see [20].

Example 5.1: Consider the following idempotent, commutative Latin square of order 7:

$$\begin{bmatrix} 1 & 6 & 5 & 7 & 2 & 3 & 4 \\ 6 & 2 & 7 & 5 & 4 & 1 & 3 \\ 5 & 7 & 3 & 6 & 1 & 4 & 2 \\ 7 & 5 & 6 & 4 & 3 & 2 & 1 \\ 2 & 4 & 1 & 3 & 5 & 7 & 6 \\ 3 & 1 & 4 & 2 & 7 & 6 & 5 \\ 4 & 3 & 2 & 1 & 6 & 5 & 7 \end{bmatrix}.$$

This Latin square contains a sub-Latin square of order two, specified by the coordinates $\{(2,5),(2,6),(3,5),(3,6)\}$, containing the elements $\{1,4\}$. The four triples

$$\{(2,1),(5,1),(4,2)\},\{(3,1),(6,1),(4,2)\},\ \{(2,1),(6,1),(1,2)\},\{(3,1),(5,1),(1,2)\}$$

derived by fixing the last coordinate of the first two pairs and choosing the first elements of the first two pairs in each block so as to include the points of the sub-Latin square of order two, form a Pasch configuration.

Closely related to the construction due to Bose described earlier is the Skolem construction [59]. This construction is also based on Latin squares, and it produces STSs of order $6m+1$. In this case, the Latin squares of interest are the so-called half-idempotent commutative squares of order $2m$. A square of order $2m$ is half-idempotent if the cells $(i,i),(m+i,m+i)$, $\forall 1 \leq i \leq m$ contain the symbol i. For more details about the Skolem construction, the reader is referred to [59].

C. Circulant LDPC Codes With Permutation Blocks: New Constructions

Based on the description of the Bose Latin square construction, it follows that the parity-check matrix of the corresponding linear code has a block structure, where each block (except for the set of blocks in the first column of blocks) is a permutation matrix. Codes with such a structure are well known and have been used in many applications. For example, in [39], Tanner presented a class of codes of this type named Sparse Difference Codes. The so-called *array codes* introduced by Blaum, Farrell, and Tilborg [60], have parity-check matrices that are also composed of powers of permutation matrices, and can be viewed as LDPC codes [61]. Eleftheriou and Olcer [62] proposed this class of LDPC codes for application in digital subscriber lines. More recently, Kim *et al.* [63] presented a construction of families of LDPC codes based on permutation matrices that have girth at least eight.

We will present next a novel construction for a simple family of LDPC codes using a similar idea. We will define the parity-check matrix H to be a block-circulant (cyclic) matrix with blocks that are permutation matrices. The structure of these codes is extremely simple and allows for finding simple lower bounds for the girth, as well as the minimum distance. The girth of all the constructed codes can be shown to be at least six, while the minimum distance of a certain subclass of these codes is at least six. The rate of these codes can have a large range of values, and the family of codes produced in this way is quite large. It is interesting to observe that for these codes it is also possible to find the eigenvalues of HH^T either in closed form, or asymptotically [64]. Unfortunately, the bounds on minimum distance presented by Tanner in [39] can be shown to be very loose for these codes and are not included in the paper.

Code Description: Let H be an $m \times N, l \times N$ block-circulant matrix with permutation blocks, i.e., block-circulant with blocks

that are different powers of the basic permutation matrix P of order N, i.e.,

$$H = \begin{bmatrix} P^{i_1} & P^{i_2} & P^{i_3} & \dots & P^{i_l} \\ P^{i_l} & P^{i_1} & P^{i_2} & \dots & P^{i_{l-1}} \\ \dots & \dots & \dots & \dots & \dots \\ P^{i_{l-m+2}} & P^{i_{l-m+3}} & P^{i_{l-m+4}} & \dots & P^{i_{l-m+1}} \end{bmatrix} \quad (5.1)$$

where $i_1, \dots, i_l$ are nonnegative integers and

$$P = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & 1 \\ 1 & 0 & 0 & \dots & 0 \end{bmatrix}$$

is an $N \times N$ circular permutation matrix.

The row weight of the code is l, while the column weight is m. The code generated by H of the form given by (5.1) has to be even (i.e., all codewords of the code have to have even Hamming weight), because the sum of all codewords over one row-block is the all-one codeword. We will describe next several construction techniques for the exponent set $\{i_1, i_2, \dots, i_l\}$ that result in a Tanner graph free of cycles of length four.

Lemma 5.1: Let $i_j = j(j-1)/2$, $1 \le j \le l$ and $N \ge l(l+1)/2 + (l-m)(m-2)$, where $l > m \ge 2$. Then the LDPC code specified by H of the form given by (5.1) has girth at least six and for $m = 2$ girth at least eight. Furthermore, no two columns of its parity-check matrix are identical, and no three columns are linearly dependent. Hence, the minimum distance $d_{\min}$ of the code is at least four.

Proof: The proof is given in **Appendix D.**

Example 5.2: Let $N = 10$, $l = 4$, and $m = 2$; then the parity-check matrix H is of the form

$$H = \begin{bmatrix} I & P & P^3 & P^6 \\ P^6 & I & P & P^3 \end{bmatrix}.$$

It can be easily shown that the generator matrix for this code is of the form

$$G = \begin{bmatrix} P^6 & P^4 & P^3 & P^9 \\ P^9 & P^6 & P^4 & P^3 \end{bmatrix}$$

and is also a block-circulant matrix with permutation blocks. This special form of the generator matrix can make the encoding procedure very fast.

Codes described by a parity-check matrix with regular column weight equal to two are also known as *circuit codes* [65]. As pointed out in [66], LDPC codes with column weight two can outperform LDPC codes with larger column weight in magnetic recording applications, despite the fact that the minimum distance of these codes can increase only logarithmically with the block length. This is due to two reasons: first, these codes have girth at least eight; second, in magnetic recoding applications, LDPC codes are inner codes for Reed–Solomon (RS) outer codes, and the block-error statistics of LDPC codes with column weight two is a good match for the RS decoder.

The permutation-matrix codes described above can be easily shown to have minimum distance $d_{\min} = 4$. For our example, it suffices to observe that columns 2, 14, 25, and 39 are linearly dependent. Because of the symmetry of the construction, codes with all possible N and s values will have the same minimum distance.

In order to construct codes with minimum distance higher than four, the exponents of the permutation matrices have to be chosen with more care.

Definition 5.1: Let S be an ordered difference set over Z_N, and let C^i denote the operator that cyclically shifts a sequence i positions to the right. If for $i = 1, \dots, m$ the ordered sets $\Omega_i = C^i S - S \bmod N$ are all different from each other (and from S), and are themselves ordered difference sets, then we say that S is an $(m+1)$-fold cycle invariant difference set over Z_N.

Example 5.3: Consider Z_7 and the ordered difference set $(1,2,4)$. Then

$$\Omega_1 = (4,1,2) - (1,2,4) = (3,6,5) \mod 7$$

which is easily seen to be a difference set. Since

$$\Omega_2 = (1,2,4) = S$$

$(1,2,4)$ is a 2-fold cycle invariant difference set over Z_7. On the other hand, the difference set $(0,1,3,9)$ over Z_{13} gives rise to three different ordered sets, namel,y $\Omega_1 = (9,12,11,7)$, $\Omega_2 = (3,8,10,5), \Omega_3 = (1,2,6,4)$, none of which is a difference set. Hence, $(0,1,3,9)$ is a 1-fold cycle invariant difference set (classical difference set) over Z_{13}.

The requirement for S to be a difference set can be relaxed. It suffices for S to be an *incomplete* difference set, i.e., a set such that the differences of its elements are all different, but do not necessarily cover every possible value. From now on we will refer to incomplete difference sets as difference sets.

One of the first constructions of classical difference sets is due to Bose [25], [26]. For this construction, the parameters of the difference family are $(q^2-1, q, 1)$, where q is an odd prime. Let ω be a primitive element in the field GF(q^2). Define the following set of integers:

$$S = \{i : 0 \le i \le q^2 - 1, \omega^i + \omega \in \text{GF}(q)\}.$$

Clearly, S consists of q elements. It is also straightforward to show that S is a difference set modulo $q^2 - 1$, with $\lambda = 1$. A similar construction method, which predates the Bose construction, is due to Singer [58].

We will present next a construction for the new class of combinatorial object described in Definition 5.1; the construction can be viewed as an extension of the result due to Bose.

Theorem 5.1: Let ω be a primitive element of the finite field GF(q^4) with q an odd prime. Define a set S of integers by

$$S = \{a : 0 \le a < q^4 - 1, \omega^a + \omega \in \text{GF}(q)\}.$$

Then the set S forms a q-fold cycle-invariant difference set $\bmod q^4 - 1$.

Proof: The proof is given in **Appendix E**.

Example 5.4: Let $q = 3$ and $q^2 - 1 = 8$. The set $S = \{0,1,3\}$ constructed by using the approach by Bose is clearly a difference set, since its differences modulo 8 are $\{7,5,1,6,3,2\}$. But the set S is not two-fold cycle invariant, since $(0,1,3) - (3,0,1) = (3,7,6)$, and for the last set $3 - 7 \equiv 7 - 3 \bmod 8$. On the other hand, the set

$$S = \{431, 561, 1201, 1312, 1406, 1579, 1883\}$$

constructed according to Theorem 5.1 with $q = 7$ and primitive polynomial $x^4 + x^2 + 3x + 5$, can be easily checked to be a CIDS of order q.

Lemma 5.4: Let i_j, $1 \leq j \leq l$, be elements of an m-fold cycle invariant difference set. Then, the LDPC code specified by H of the form given by (3) has girth at least six.

Proof: The proof follows along the same lines as the proof of Lemma 5.1.

Theorem 5.2: Let i_j, $1 \leq j \leq l$, be elements of an m-fold cycle-invariant difference set, with $l > m \geq 2$. Then, the LDPC code specified by H of the form given by (5.1) has minimum distance at least six.

Proof: The proof is given in **Appendix F**.

Example 5.5: Consider the two blocks from Example 3.4, $\{0,1,4\}$ and $\{0,2,7\}$, now viewed over Z_{11} rather than Z_{13}. The first difference set is two-fold cycle-invariant in the general setting, since $\Omega_1 = (0,1,4) - (4,0,1) = (4,10,8) \mod 11$, and the differences generated by $(4,10,8) \mod 11$ are $\{5,7,6,2,4,9\}$, so that each element in Z_{11} appears at most once. Hence, an LDPC code specified by the parity-check matrix

$$H = \begin{bmatrix} I & P & P^4 \\ P^4 & I & P \end{bmatrix}$$

where the dimension of P is eleven, has minimum distance at least six and girth at least eight.

Similarly, if we take the two blocks in the cyclic difference family as exponents for the permutation matrices of two different rows of blocks, we obtain

$$H = \begin{bmatrix} I & P & P^4 \\ I & P^2 & P^7 \end{bmatrix}.$$

Notice that H is not any longer block circulant, but the set $(0,1,4) - (0,2,7) = (0,1,3) \mod 11$ generates a set of differences $(10,8,1,9,3,2)$ such that each element of Z_{11} appears at most once. Hence, the underlying code has minimum distance at least six and girth at least eight.

VI. Lattice Construction of LDPC Codes

In this section, we address the problem of constructing LDPC codes of large block lengths. As shown in the previous sections, the Buratti-type CDFs and the projective geometry approach offer a quite limited set of parameters and therefore small families of codes. In this section, we give a novel construction of 1-configurations, in which every 2-tuple is contained in at most $\lambda = 1$ blocks. The blocks of these 1-configurations are lines connecting points of a rectangular integer lattice. The construction problem can be seen as specifying a subset of points and a subset of lines that will result in a set of desired characteristics of the code. In this way, one can trade the code rate and number of blocks for the simplicity of construction and for the flexibility of choosing the design parameters. Also, as will be shown subsequently, 1-configurations greatly simplify the construction of codes with large girth.

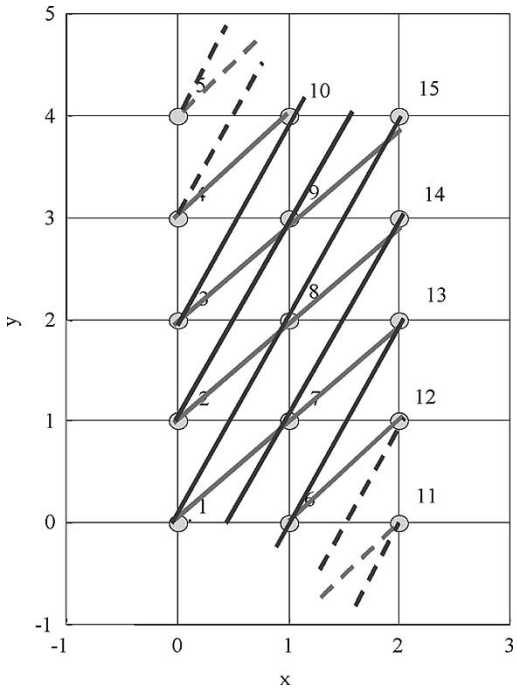

Fig. 2. An example of the rectangular grid for $q = 5$ and $c = 3$.

A. Codes on a Rectangular Subset of an Integer Lattice

Consider a rectangular subset L of the integer lattice defined by

$$L = \{(x,y) : 0 \leq x \leq c-1, 0 \leq y \leq q-1\}$$

where $q \geq c$ is a prime. Let $\mathrm{lab} : L \to V$ be a one-to-one mapping from the set L to the point set V. An example of such a mapping is the simple bi-linear map $\mathrm{lab}(x,y) = q \cdot x + y + 1$. The numbers $\mathrm{lab}(x,y)$ are referred to as point labels.

Example 6.1: Fig. 2 depicts a rectangular subset of the integer lattice with $q = 5$ and $c = 3$.

A set of c points is referred to as a line of slope s, $0 \leq s \leq q-1$, *starting at* the point $(0,a)$, if it contains the points $\{(x, a + sx \mod q) : 0 \leq x \leq c-1\}$, where $0 \leq a < q$. There are q different classes of parallel lines in this geometry.

Example 6.2: In Example 6.1, the lines of slope 1 are the triples $\{1,7,13\}, \{2,8,14\}, \{3,9,15\}$ and so forth. We assume that the point labels are periodic in the vertical dimension, and therefore the line containing the points $\{4,10,11\}$ also has slope 1. Examples of lines with slope two are $\{1,8,15\}$ and $\{2,9,11\}$. Notice that lines of infinite slope are not included in the 1-configuration.

Lemma 6.1: The set of point labels of lines with slopes s, $0 \leq s \leq q-1$ forms the blocks B of a 1-configuration.

Proof: The proof is given in **Appendix G**.

Remark 6.1: In the lattice 1-configuration, the block size is c, the number of points is $q \cdot c$, the number of blocks is $b = q^2$.

Lemma 6.2: A $(q, c = 3)$ lattice 1-configuration is Pasch-free.

Proof: The proof is given in **Appendix H**.

More generally, we have the following result.

Theorem 6.1: A lattice 1-configuration with $(q, c = h)$, for $h > 3$ even or $h = q$, does not contain a generalized Pasch configuration.

Proof: The proof is given in **Appendix I**.

Fig. 3 shows the rate–length characteristics of lattice designs for different line (block) sizes c. The solid lines correspond to the designs with maximal number of blocks. As can bee seen, lattice families have a rate loss compared to theoretically optimal designs. However, this loss becomes negligible for larger c and for longer codes.

The abundance of lattice designs compared to infinite family BIBDs can be easily observed from Fig. 4. White markers denote codes resulting from the known infinite family BIBDs, while dark markers correspond to LDPC codes constructed using lattice designs.

Remark 6.3: The lattice construction can be extended to nonprime vertical dimensions q, provided that the slopes s are co-prime to q.

Fig. 5 shows the growth of the required code length with an upper bound on the minimum distance (of the form $2c$) as parameter.

Example 6.3: Table VI illustrates a 1-configuration based on a lattice shown in Fig. 2.

Notice that there are q parallel classes of blocks (lines), each corresponding to a different slope. Denote the first block (the one incident with the point 1) in the class s as B^s, $B^{(s)} = (b_1^{(s)}, b_2^{(s)}, b_3^{(s)})$. In our example $B^{(0)} = (1, 6, 11)$, $B^{(1)} = (1, 7, 13)$, $B^{(2)} = (1, 8, 15)$, $B^{(3)} = (1, 9, 12)$, and $B^{(4)} = (1, 10, 14)$.

The corresponding parity-check matrix is

$$H = \left[\begin{array}{ccccc}
10000 & 10000 & 10000 & 10000 & 10000 \\
01000 & 01000 & 01000 & 01000 & 01000 \\
00100 & 00100 & 00100 & 00100 & 00100 \\
00010 & 00010 & 00010 & 00010 & 00010 \\
00001 & 00001 & 00001 & 00001 & 00001 \\
\hline
10000 & 00001 & 00010 & 00100 & 01000 \\
01000 & 10000 & 00001 & 00010 & 00100 \\
00100 & 01000 & 10000 & 00001 & 00010 \\
00010 & 00100 & 01000 & 10000 & 00001 \\
00001 & 00010 & 00100 & 01000 & 10000 \\
\hline
10000 & 00010 & 01000 & 00001 & 00100 \\
01000 & 00001 & 00100 & 10000 & 00010 \\
00100 & 10000 & 00010 & 01000 & 00001 \\
00010 & 01000 & 00001 & 00100 & 10000 \\
00001 & 00100 & 10000 & 00010 & 01000
\end{array}\right].$$

Notice that in general, the parity-check matrix of a lattice codes can be written in the form

$$H = \begin{bmatrix}
I & I & I & \dots & I \\
I & P_{2,1} & P_{2,2} & \dots & P_{2,q-1} \\
I & P_{3,1} & P_{3,2} & \dots & P_{3,q-1} \\
\dots & & & & \dots \\
I & P_{c-1,1} & P_{c-1,2} & \dots & P_{c-1,q-1}
\end{bmatrix}$$

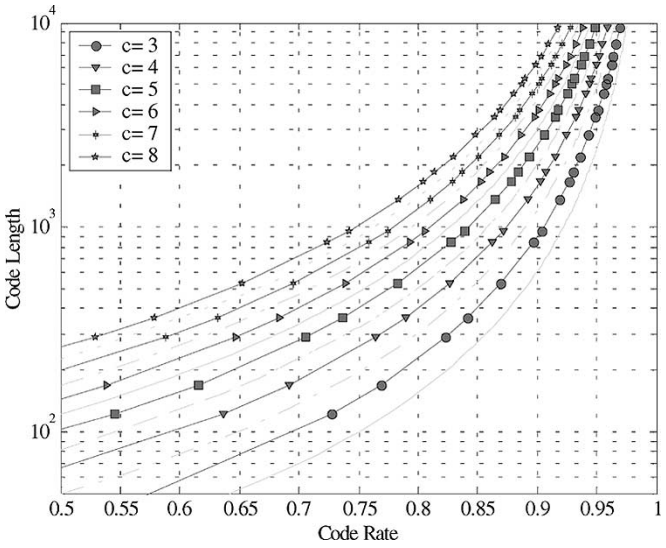

Fig. 3. The rate–length characteristics of lattice designs.

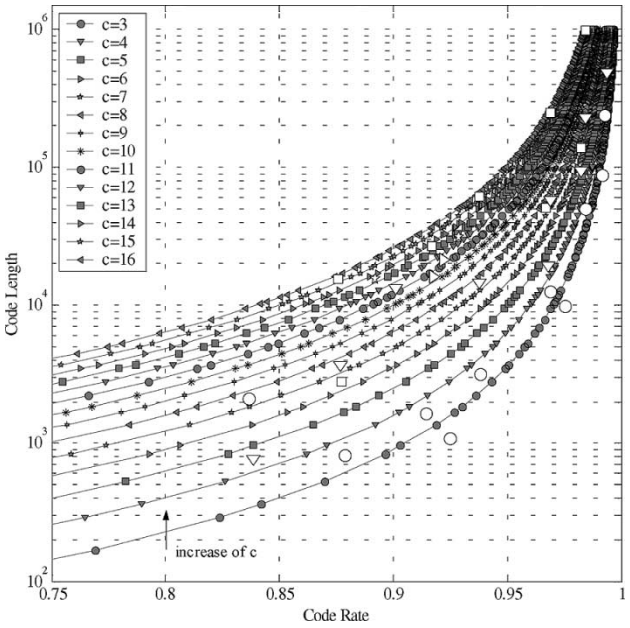

Fig. 4. Comparison of finite geometries and lattice design families.

where each submatrix $P_{i,j}$ is a circulant permutation matrix. The power of P which determines $P_{i,j}$ (i.e., the position of the bit 1 the first column of $P_{i,j}$) can be found by using $B_i^{(j-1)}$, the ith element of the first base block of the class of blocks corresponding to the jth slope, and is equal to $q - (i-1) \cdot j \bmod q$.

Remark 6.4: Notice a similarity of the structure of the above parity-check matrix with that obtained in [63], [67]. The codes denoted by LU $(2, q)$ in [63] have a square parity-check matrix, while our codes have rectangular matrices of parity checks. This is not surprising because it was shown in [63] that LU $(2; 4)$ and LU $(2; 8)$ are equivalent to Euclidean geometry code [13], while a square lattice design (which includes the lines with infinity slope) is equivalent to the Euclidean plane.

TABLE VI
AN EXAMPLE OF A LATTICE 1-CONFIGURATION

s=0			s=1			s=2			s=3			s=4		
1	6	11	1	7	13	1	8	15	1	9	12	1	10	14
2	7	12	2	8	14	2	9	11	2	10	13	2	6	15
3	8	13	3	9	15	3	10	12	3	6	14	3	7	11
4	9	14	4	10	11	4	6	13	4	7	15	4	8	12
5	10	15	5	6	12	5	7	14	5	8	11	5	9	13

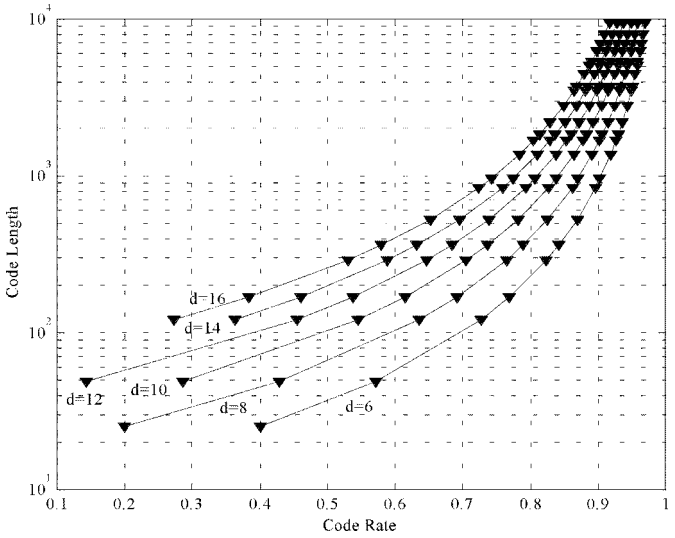

Fig. 5. The rate–length curve for lattice designs with the minimum distance as a parameter.

Remark 6.5: The ensemble of LDPC codes described in terms of parity-check matrices constructed from random permutation matrices has a well-defined asymptotic distance distribution. Litsyn and Shevelev [68] showed that such an ensemble (referred to as "Ensemble A") has superior distance distribution compared to other ensembles they considered in [68], in the context of ML decoding.

B. Integer Lattice Codes With Large Girth

In this subsection, we will show that a Tanner graph with high girth can be obtained by a judicious selection of sets of parallellines included in an integer lattice 1-configuration. The resulting parity-check matrix is also in the form of a block matrix with permutation matrices. We will only discuss codes of girth eight, although a generalization for higher girths is possible.

A lower bound on the minimum distance of LDPC codes with girth g and column weight c is given by the following formula due to Tanner [10]:

$$d_{\min} \geq \begin{cases} 1+\frac{c}{c-2}((c-1)^{\lfloor (g-2)/4 \rfloor}-1), & g/2 \text{ odd} \\ 1+\frac{c}{c-2}((c-1)^{\lfloor (g-2)/4 \rfloor}-1) & \\ \qquad +(c-1)^{\lfloor (g-2)/4 \rfloor}, & g/2 \text{ even.} \end{cases}$$

Although it still remains unclear whether increasing the girth of a bipartite graph is the best way to improve code performance under message-passing decoding, our simulation results indicate that it is a valid approach, especially when the constraint imposed on a number of iterations is not strict.

Recently, Rosenthal and Vontobel [15] constructed some short codes and large girth, using ideas by Margulis [53], including c-regular Caley graphs of the special linear group $\mathrm{SL}_2(\mathrm{GF}(q))$, and the projective general group $\mathrm{PGL}_2(\mathrm{GF}(q))$. Kim *et al.* [63] gave another explicit construction of high-girth LDPC codes using Lazebnik and Ustimenko's [67] method for developing regular graphs. The construction of designs with high girths appears to be a very difficult problem in general [47]. However, the designs based on rectangular integer lattices allow a simple algorithm for finding a girth-eight subdesign. Moreover, for $c = 3$, there is an interesting connection between codes of girth eight and "arithmetically constrained" sequences defined by Odlyzko and Stanley [69].

Denote by $B(s)$ a resolvability class corresponding to the set of lines of slope s, and by $B'(\sum)$ a set of blocks of a subdesign composed of resolvability classes corresponding to the slopes from the set $\sum$, i.e., $B'(\Sigma) = \bigcup_{s\in\Sigma} B(s)$. We are interested in the following problem: find a set of slopes $\sum_m$ with maximum possible cardinality such that $B'(\sum_m)$ specifies a set of blocks that result in a design with girth eight.

Example 6.4: For a subset of an integer lattice shown in Fig. 2 (with $q = 5, c = 3$), it can be shown by inspection that the maximal set of slopes leading to a code of girth eight is of cardinality two (e.g., $\sum_m = \{\{0, 1\}\}$). The resulting parity-check matrix is obtained by deleting columns of a parity-check matrix of the original code corresponding to lines with slopes $\{2, 3, 4\}$, and is of the form

$$H^T = \begin{bmatrix} I & I & I \\ I & P^4 & P^3 \end{bmatrix}$$

where P is of order $q = 5$. Due to the small size of the set $\sum_m$, this matrix is useless for coding purposes and represents only an illustration of the key idea.

Definition 6.1: Let Θ be the set of all sequences of integers that do not contain a 3-term arithmetic progression. We will refer to A as the "earliest" sequence in Θ if A lexicographically precedes all other sequences in Θ. A is of the form $0, 1, 3, 4, 9, 10, 12, \ldots$, and is cataloged in [70] under number M2353.

The sequence A can be generated by the recurrence relation: $a(2n) = 3a(n), a(2n + 1) = a(2n) + 1, a(0) = 0$, and it has the property that it contains all numbers that only have the digits 0 and 1 in their ternary expansion [69].

Theorem 6.2: For an arbitrary integer q and for $c = 3$, the set of slopes $\sum_m$ resulting in codes with girth eight is of the form $\Sigma_m = \{a : a \in A, a \leq q/2\}$, i.e., it is a subsequence of A whose elements are less than or equal to $q/2$.

Proof: The proof is given in **Appendix J**.

VII. SIMULATION RESULTS

In this section, we present the bit-error rate (BER) performance of various regular LDPC codes that were constructed ei-

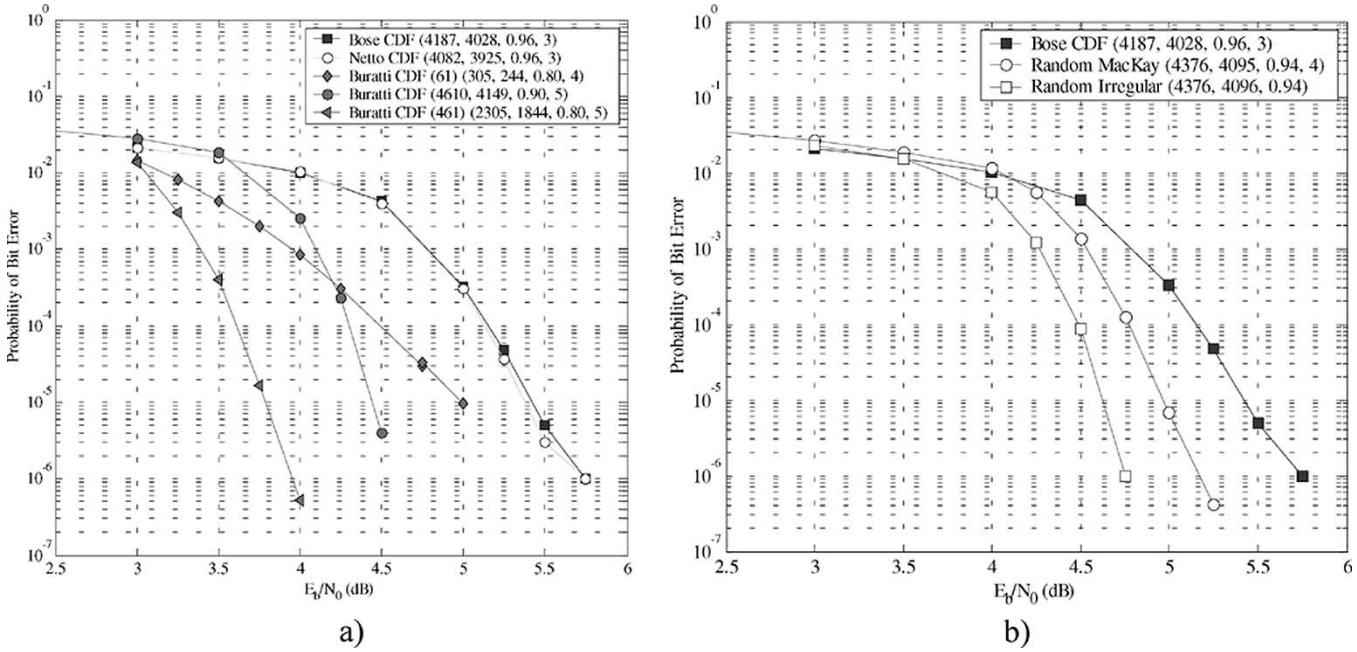

Fig. 6. Performance of LDPC codes on combinatorial designs under message-passing decoding.

ther combinatorially or randomly. The LDPC codes were decoded iteratively by using the standard message-passing algorithm. More details on the message-passing algorithm for bipartite graphs can be found in [6], [7], [72]. The BER performance of an LDPC code was estimated by running Monte Carlo simulations for at least 25 000 codewords and 15 message-passing iterations. Hence, we were able to obtain BER as low as 10^{-7}.

Figs. 6–9 show the BER performance of LDPC codes, constructed using different combinatorial methods presented in this paper, and random regular LDPC codes obtained from Mackay's online resource [73]. The rate of these codes vary from 0.75 to 0.96, and all the BER curves shown in Figs. 6–9 have been adjusted for their respective rate loss (i.e., the SNR is determined from $E_b/(RN_0)$). A legend in each figure gives the following information in the respective order: method used to construct the code, and a quadruple (n, k, R, c).

As can be observed from Fig. 6(a), codes from difference families, due to Bose and Netto, exhibit an error floor at approximately 5.75 dB, which is expected behavior. Fig. 6(b) shows the BER performance comparison between difference family codes and randomly constructed (regular and irregular) codes of similar length and rate. At BER of 10^{-6}, a loss of 0.5 and 0.9 dB with respect to regular/irregular code, respectively, can be seen. In Fig. 7, a relatively short code from a projective plane exhibits a very good BER performance. In Fig. 8, we consider rate-0.88 codes on integer lattices for the following parameters: 1) $m = 23$, $c = 3$; 2) $m = 31$, $c = 4$; 3) $m = 41$, $c = 5$; 4) $m = 59$, $c = 7$; 5) $m = 67$, $c = 8$. In addition, we also consider rate–0.8 CDF codes constructed using Buratti's method for $(p = 61, c = 4)$ and $(p = 461, c = 5)$, and Wilson's method for $(p = 151, c = 5)$. The BER curves of an integer lattice code and a MacKay code [73] of similar parameters is given in Fig. 8(b), demonstrating that short integer lattice codes have performance comparable to random-like MacKay codes. The BER performance of a girth-eight LDPC code with column weight

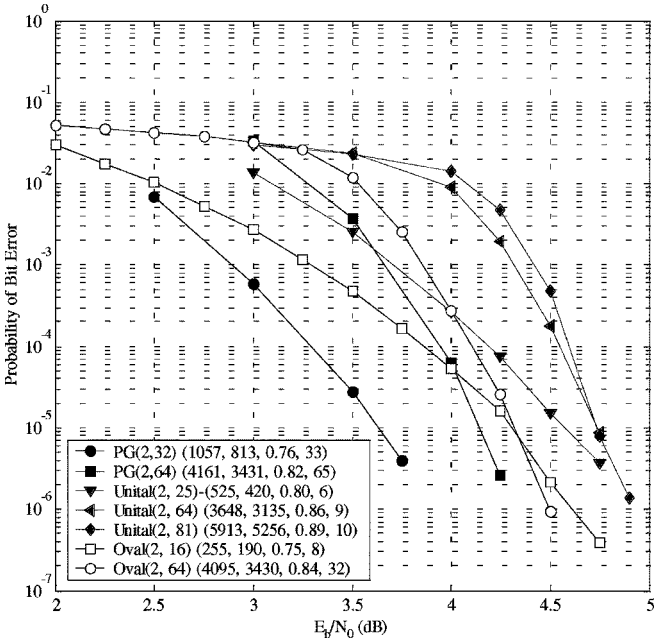

Fig. 7. Performance of LDPC codes on finite geometries.

four is the most impressive among other curves shown in Fig. 9. This code exhibits a sharp fall and its BER reaches 10^{-7} at approximately 3.25 dB. A BER curve of a regular MacKay code in Fig. 9(a) is given as a reference point without any attempt to shorten it to match the length and rate of the given structured code. The comparison of girth-eight and random codes of the same lengths and rates is shown in Fig. 9(b). Random codes are constructed by Neil's method [74], and are free of cycle-four. It can be seen from Fig. 9(a) that girth-eight codes have almost identical performance with randomly constructed codes.

It is important to note that in simulations, that were run to estimate the BER performance of the $(3801, 3260)$ girth-eight

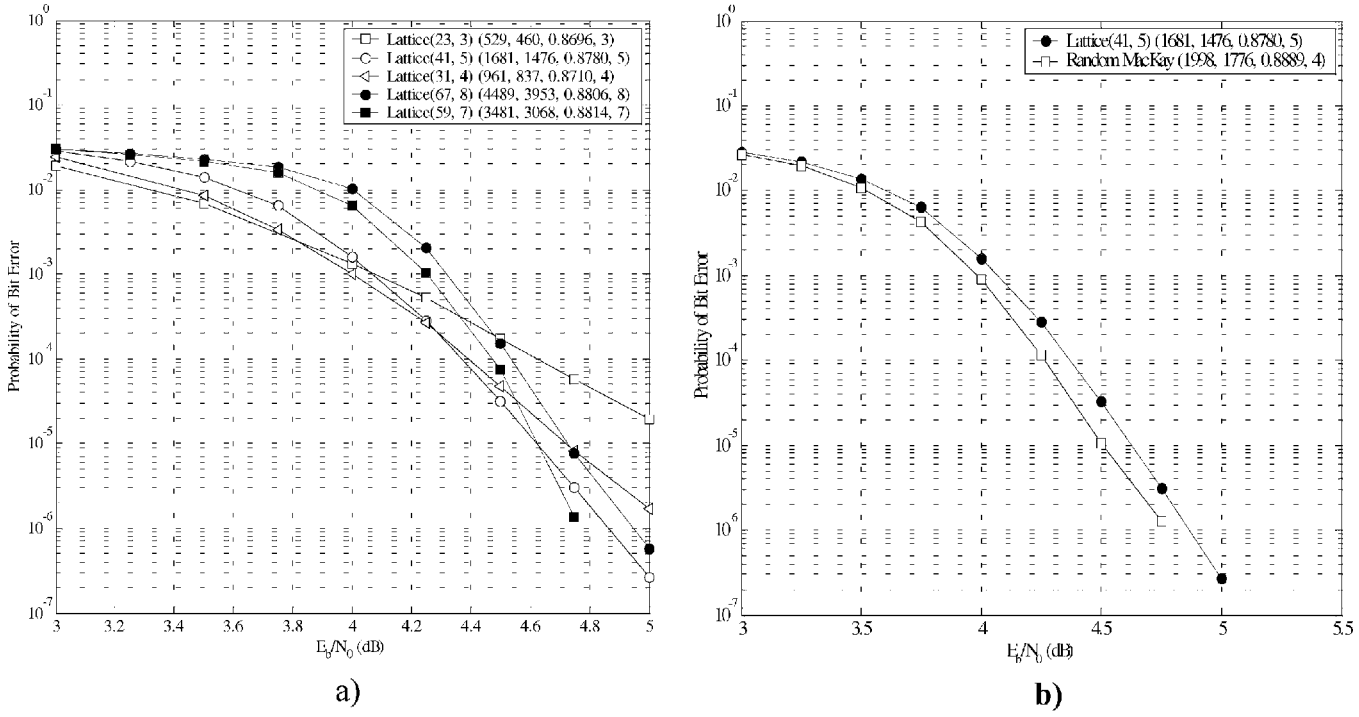

Fig. 8. Performance of integer lattice codes.

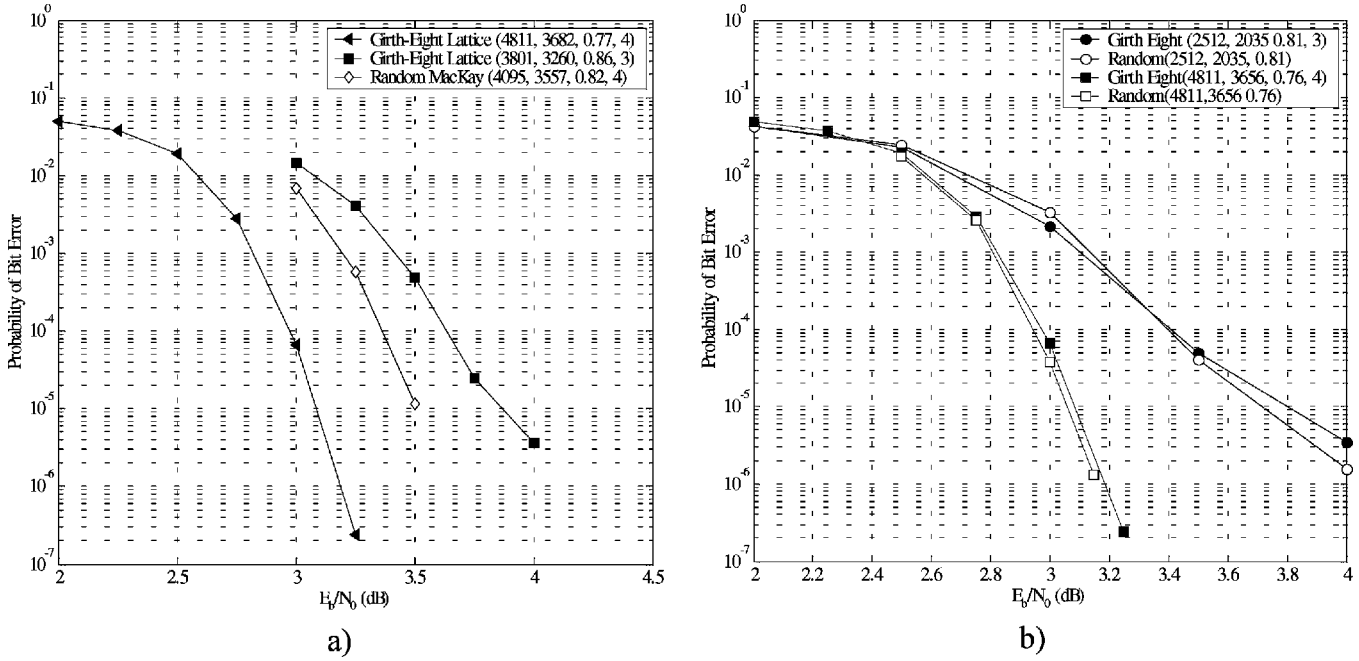

Fig. 9. Performance of girth-eight integer lattice codes.

code, 100 message-passing iterations were performed before making the hard decision on a received word. An error floor was observed if only 15 iterations were performed in the above simulations.

As already described in the Introduction, due to stringent delay constraints some applications do not allow for more than several (5 to 6) iterations of message passing. Fig. 10 shows the BER performance of two girth-eight integer lattice codes after i iterations, $0 \leq i \leq 5$. It can be seen that a significant gain with respect to the uncoded system is achieved only after five iterations, but that with another 10 iterations a performance gain of approximately 1 dB can be achieved.

Finally, the performance of several codes based on the constructions presented in Section V is shown in Figs. 11 and 12.

Fig. 11 shows the performance of the codes described in Lemma 1, while Fig. 12(a) plots the performance of block-circulant codes with exponents taken from the Bose difference set S over GF (q^2), $q = 17$, with $p(x) = x^2 + x + 3$, i.e.,

$$S = \{2, 17, 27, 39, 47, 58, 79, 85, 102, 136, 145, 149, 150, 152, 178, 231, 266\}$$

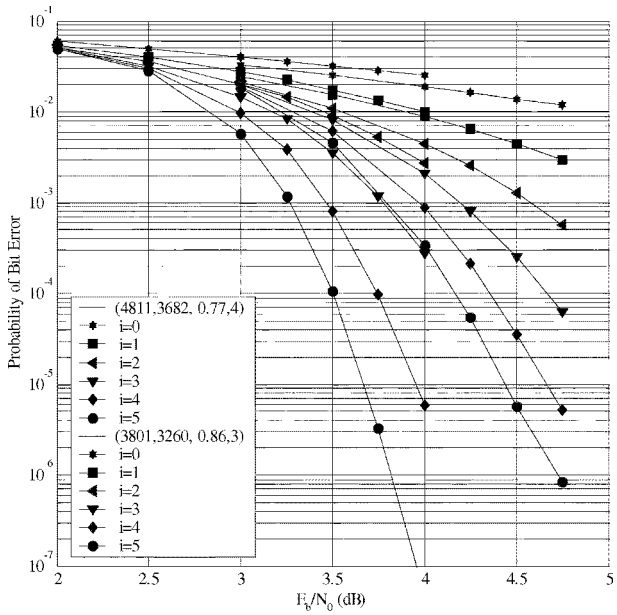

Fig. 10. Performance of girth-eight integer lattice codes with respect to the number of iterations i.

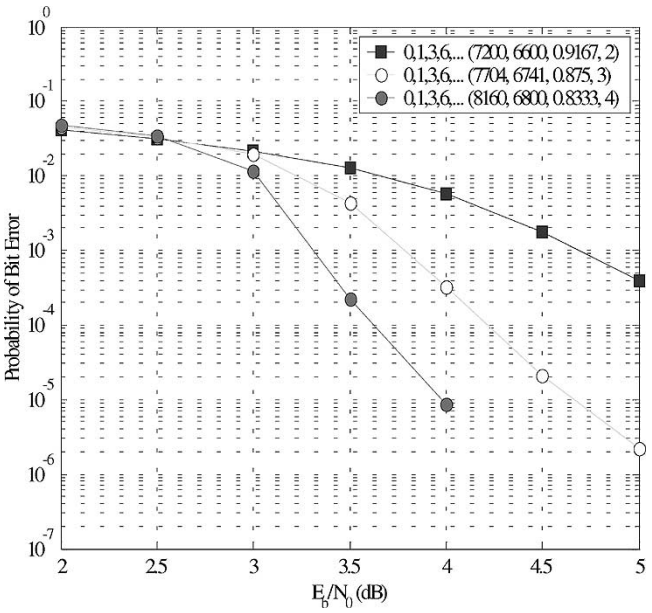

Fig. 11. Performance of codes based on block-circulant parity-check matrices.

Fig. 12(b) shows the performance of CIDS-based codes, with the construction presented in Theorem 5.1 and $q = 7$.

VIII. Conclusion

In this paper, we introduced new combinatorial constructions for a class of high-rate iteratively decodable codes based on $(v, k, 1)$ BIBD and 1-configurations. The resulting codes have girth at least six. We also constructed balanced incomplete block designs using Netto and Buratti cyclic difference families, as well as novel affine geometry lattice configurations. We derived tight bounds on a minimum distance of BIBD codes using the concept of a Pasch configuration.

Appendix A
Proof of Lemma 3.1

Observe first that

$$M = HH^T = [\, H_1 \quad H_2 \quad \dots \quad H_t \,] \begin{bmatrix} H_1^T \\ H_2^T \\ \dots \\ H_t^T \end{bmatrix}$$
$$= H_1H_1^T + H_2H_2^T + \cdots + H_tH_t^T.$$

Based on a well-known result, which states that $H_i\, H_i^T = (\hat{r} - \lambda)\ I + \lambda J$ for any point-block incidence matrix H_i of a design [20], it follows that for the STS of interest

$$M = HH^T = tH_iH_i^T = t(\hat{r} - 1)I + tJ$$

where I and J are the identity matrix and the all-one matrix of order v. The determinant of $M - \mu I$ can be easily found to be of the form

$$t^v(\hat{r} - 1 - \mu/t)^{v-1}(\hat{r} + v - 1 - \mu/t).$$

This proves the claimed result. Q.E.D.

Appendix B
Proof of Lemma 3.2

Let us start from an empty array and create the rows of Λ_c by minimizing the number of used points. The first row can always be filled with points from the set $\{1, 2, \dots, c\}$. Put 1 in the firstposition of the second row (now 1 occurs twice) and start filling out the rest of the second row. Since the first element in the row is 1, we cannot use any other point from the first row. Therefore, we must use points from the set $\{c+1, \dots, 2c-1\}$. These points completely describe the second row.

We can put any number as the first element in the third row. If we choose this entry to be 1, then since 1 already occurs twice, another row containing 1 must be added to keep the number of occurrences of 1 even. Therefore, we chose not to make the first entry 1, but rather a different number from the first row. Let this one be 2. Now 2 is the first entry in the third row and occurs twice in the array. We continue filling the rest of the third row. The points from the first row must be excluded, since 2 is also in the first row. The points from the second row, except for 1, are all allowed so let us take the first available number $c + 1$. The rest of the third row ($c - 2$ positions) must be now filled with (new) numbers that are not in the second row. For these, we choose $\{2c, 2c+1, \dots, 2c+c-3\}$. Suppose we continuine this process until we arrive at the $(j+1)$th row. Every point in the set $\{1, 2, \dots, j\}$ occurs twice in the first $j + 1$ rows of Λ_c, and the point $j + 1$ occurs exactly once (in the first row). Therefore, we have to continue appending rows so as to achieve an even number of occurrences of the point $j + 1$. The above procedure continues until c rows are formed. However, Λ_c must have $(c+1)$ rows because the point c and the points in the last column of Λ_c occur only once in the array. Since, at this stage, there are exactly c points that do not occur twice, we can take the points from the last column of Λ_c and append them as the $(c+1)$th row of Λ_c, which completes the procedure. The number of "new" (nonrepeated) points used to fill the jth row is $c - (j - 1)$. The total number of points used is $c{\cdot}c{-}(1{+}2{+}\cdots{+}(c{-}1))$ which is

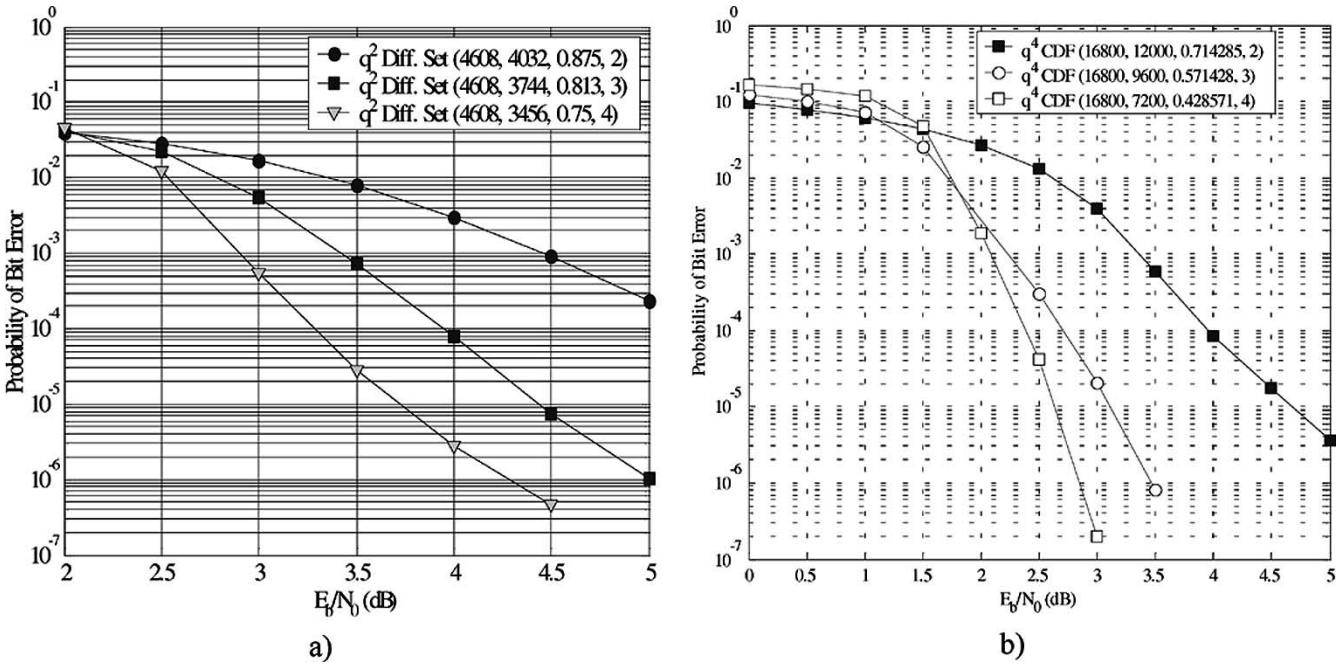

Fig. 12. Performance of CIDS-based codes.

equal to $c(c+1)/2$, the number of points in a generalized Pasch configuration. Q.E.D.

Appendix C
Proof of Theorem 3.1

Take any distinct i and j and consider two base blocks B_i and B_j and the corresponding blocks obtained by adding $g \in Z_v$ to B_i and B_j. Notice that by adding $g \in Z_v$ to elements of B_i, the resulting blocks remain in the orbit of B_i. Create a $c \times c$ array $F = [s_{m,n}]$, where $s_{m,n} = b_{i,m} + b_{j,n}$. (Notice that since B_i and B_j are base blocks, all the elements in F are different.) The columns of F contain blocks $B_i + b_{j,n}$, and the rows of F contain blocks $B_j + b_{i,m}, 1 \leq m, n \leq c$. Each element of F occurs exactly once in $B_i + b_{j,n}$ and once in $B_j + b_{i,m}$, and, therefore, exactly twice in the set $\{B_i + b_{j,n} \cup B_j + b_{i,m}, : 1 \leq m, n \leq c\}$. Q.E.D.

Example C.1: For the example in Table I, the array F is

$$F = \begin{bmatrix} 0 & 2 & 7 \\ 1 & 3 & 8 \\ 4 & 6 & 11 \end{bmatrix}$$

and

$$\{B_i + b_{j,n} \cup B_j + b_{i,m}, : 1 \leq m, n \leq c\}$$

is given in Example 3.3.

Appendix D
Proof of Lemma 5.1

In order to show that the girth of the code is at least six, we have to show that there are no "rectangles" in the matrix H, i.e., four ones that lie at the corners of a rectangle. Without loss of generality, consider two ones in the first row of H. Two ones that belong to the same row must necessarily belong to different permutation matrices, and the same is true for ones that belong to the same column. Hence, the distance between two ones in the same column is of the form

$$(i_l - i_{l+t}) + fN \tag{D1}$$

where i_l, i_{l+t}, $t > 0$, are the exponents of the permutation matrices containing these two ones, respectively, and where the subscripts are taken modulo N.

Next, consider the component-wise difference of the ordered exponents $\{i_1, i_2, \ldots, i_l\}$ of the first block-row of permutation matrices and the ordered exponents $\{i_{l-r+1}, i_{l-r+2}, \ldots, i_{l-r}\}$ of the jth block-row

$$(o_0, o_2, \ldots, o_{l-1}) = (i_1, i_2, \ldots, i_l) - (i_{l-j+1}, i_{l-j+2}, \ldots, i_{l-j}).$$

Then all o_i's are different and negative up to subscript number $r-1$, and all different and positive for subscripts larger than $j-1$. Next, define the following two variables:

$$Max^+ = \max_{i,\, o_i > 0} o_i = \sum_{i=l-j}^{l-1} i = \frac{(j-1)(2l-j)}{2}$$

$$Max^- = \max_{i,\, o_i < 0} |o_i| = \sum_{i=j-1}^{l-1} i.$$

Based on (D1), it follows that the distance between two ones in the same column for the first $j-1$ block-columns is at least

$$M = N - Max^- = l + (l-m)(m-2) + \frac{(j-2)(j-1)}{2}.$$

Since

$$\begin{aligned} M - Max^+ &= l + (l-m)(m-2) + \frac{(j-2)(j-1)}{2} \\ &\quad - \frac{(j-1)(2l-j)}{2} \\ &= l(m-j) + m^2 + 2m + (j-1)^2 > 0 \end{aligned}$$

it follows that no two ones in the same column can be at the same distance as any other pair of ones within a different column.

It can be easily shown that the codes with $m = 2$ have girth at least eight. This follows from the observation that in order to have a cycle of length six in the Tanner graph, there should exist three columns containing pairs of ones described by their corresponding row positions $(r_1, r_2), (r_1, r_3), (r_2, r_3)$. This clearly implies that one pair of ones must belong to the same column of one permutation matrix, which is impossible. The result regarding the minimum distance of the codes is a straightforward consequence of the previous argument. Q.E.D.

Appendix E
Proof of Theorem 5.1

The proof is an extension of the Bose construction [50] for difference sets.

Take an arbitrary ordering of the set S and define $c_i = \omega^{a_i} + \omega$. Consider the following two polynomials:

$$\begin{aligned} p_1(x) &= (x - c_i)(x - c_j) - (x - c_l)(x - c_f), \\ p_2(x) &= (x - c_{i-t})(x - c_{f-t})(x - c_l)(x - c_j) \\ &\quad - (x - c_{l-t})(x - c_{j-t})(x - c_i)(x - c_f) \end{aligned} \tag{E1}$$

where $1 \leq t \leq q-1$ and the indexes in $p_2(x)$ are taken modulo $|S| = q$.

Assume there exist indexes i, j, l, f such that $i \neq l \wedge j \neq f$, $i \neq j \wedge l \neq f$, and $a_i - a_l = a_f - a_j \bmod (q^4 - 1)$. The polynomial $p_1(x)$ has coefficients in GF(q), degree at most one, and ω as its root. Since ω is a primitive element of GF(q^4), no nonzero polynomial over GF(q) of degree smaller than four can have ω as a root. Hence, it follows that $p_1(x) \equiv 0$. Therefore, $\{c_i, c_j\} = \{c_l, c_f\}$ which implies $\{a_i, a_j\} = \{a_l, a_f\}$ and, consequently, $\{i, j\} = \{l, f\}$. This contradicts the starting assumption.

Based on the previous argument, it follows that the elements of S form a difference set modulo $q^4 - 1$.

Assume next that that the indexes i, j, l, f are as described before and

$$\begin{aligned} &(a_{i-t} - a_i) - (a_{j-t} - a_j) \\ &\quad = (a_{l-t} - a_l) - (a_{f-t} - a_f) \bmod (q^4 - 1) \end{aligned} \tag{E2}$$

where the indexes are taken modulo $|S| = q$. If $c_i = \omega^{a_i} + \omega$, then the polynomial $p_2(x)$ has coefficients from GF(q), degree at most three and ω as one of its root. Since ω is a primitive element of GF(q^4) it follows that $p_2(x) \equiv 0$. Hence,

$$\{c_{i-t}, c_{f-t}, c_l, c_j\} = \{c_{l-t}, c_{j-t}, c_i, c_f\}$$

and consequently

$$\{a_{i-t}, a_{f-t}, a_l, a_j\} = \{a_{l-t}, a_{j-t}, a_i, a_f\}$$

or equivalently

$$\{i - t, f - t, l, j\} = \{l - t, j - t, i, f\}.$$

One can distinguish several possible cases for which the two sets above are equivalent.

- $i - t \equiv l - t, j - t \equiv f - t \bmod q$, which implies $i = l, j = f$ contradicting the starting assumption.
- $i - t \equiv j - t, f \equiv l \bmod q$, contradicting the starting assumption.
- $i - t \equiv f, l \equiv j - t \bmod q$ or $i - t \equiv f, l \equiv i \bmod q$. In the first case, one arrives at the conclusion that $i = j, l = f$, and in the second case one arrives at $f = j, l = i$. Both of these results contradict the starting assumption.

Therefore, S is a q-fold cycle invariant difference set. Q.E.D.

Appendix F
Proof of Theorem 5.2

The codes described by (5.1) are even, and it is straightforward to see that no two columns in the parity-check matrix are identical. Hence, it suffices to show that for no two block-rows of permutation matrices one can find four columns that add up to the zero-vector. If the number of permutation blocks per row l is less than or equal to three, the result follows trivially. Hence, let $l > 3$, and assume that, on the contrary, there exist four columns, that sum up to zero. Furthermore, assume that the row indexes of the four ones in the first block-row and within the given columns are r_1, r_2, r_3, r_4.

It is clear from the construction that either

- all four columns belong to different permutation matrices;
- two columns belong to the same permutation matrix and two belong to two different permutation matrices; or
- two columns belong to one, while the other two belong to another permutation matrix.

An illustrative example for the first case of options is shown at the bottom of the page. Consider the first case, and assume without loss of generality that

$$r_1 = r_3, r_2 = r_4. \tag{F1}$$

Based on the cyclic construction of the parity-check matrix, the positions of the ones in the second block-row of permutation matrices and within the same columns are given by $r_1 + d_1, r_2 + d_2, r_3 + d_3, r_4 + d_4 \bmod N$, where

$$\begin{aligned} d_1 &= i_{l_1} - i_{l_2} + t_1 N, \; d_2 = i_{l_3} - i_{l_4} + t_2 N, \\ d_3 &= i_{l_5} - i_{l_6} + t_3 N, \; d_4 = i_{l_7} - i_{l_8} + t_4 N \end{aligned} \tag{F2}$$

for some nonnegative integers t_1, t_2, t_3, t_4 and for different $i_{l_j} \in S, \; j = 1, \ldots, 8$ (i.e., different elements from one of the cyclic shifts of the m-fold cycle invariant difference set). In order for the columns to add up to the zero vector, one has to have

$$r_1 + d_1 = r_2 + d_4, r_2 + d_2 = r_1 + d_3$$

which, based on (F1), implies that

$$d_4 - d_1 = d_2 - d_3 \bmod N.$$

$$\begin{array}{l} first\ \ block - row \\ \\ \\ \\ \\ \\ \\ \end{array} \begin{bmatrix} 0 \\ \cdots \\ \cdots \\ 1 \\ 0 \end{bmatrix} \cdots \begin{bmatrix} 0 \\ \cdots \\ 1 \\ \cdots \\ 0 \end{bmatrix} \cdots \begin{bmatrix} \cdots \\ 0 \\ 1 \\ 0 \\ \cdots \end{bmatrix} \begin{bmatrix} 0 \\ \cdots \\ \cdots \\ 1 \\ 0 \end{bmatrix} \begin{array}{l} \\ \\ \\ r_2, r_4 \\ r_1, r_3 \end{array}$$

$$\mathrm{col}_1 \qquad\qquad \mathrm{col}_2 \qquad\qquad \mathrm{col}_3 \qquad\qquad \mathrm{col}_4.$$

But this is impossible, since the integers d_1, d_2, d_3, d_4 themselves belong to a difference set, and hence no two differences can be the same. For the second case and for the given row positions in the first set of blocks, the row-indexes of ones in the second block-row have to be of the form $r_1+d_1, r_2+d_1, r_3+d_3$, r_4+d_4. Hence, if $r_1 = r_3, r_2 = r_4$ then the following equation has to be satisfied as well:

$$d_2 - d_1 = d_1 - d_2 \bmod N.$$

As in the previous case, this result contradicts the starting assumption that d_1, d_2, d_3, d_4 belong to a difference set. The last case can be treated in a similar manner as the second case, with $r_1 + d_1, r_2 + d_1, r_3 + d_2, r_4 + d_2$, implying $d_1 - d_2 = d_2 - d_1 \mod N$. This again contradicts the assumption that d_1, d_2, d_3, d_4 belong to a difference set.

Notice that since the argument presented above holds for any choice of two rows of blocks, it follows that the minimum distance will be at least six for any number $m > 1$ of row blocks. Q.E.D.

Appendix G
Proof of Lemma 6.1

Since q is a prime, for each lattice point (x, y) there exists exactly one line with slope s that passes through the point (x, y). For each pair of lattice points, there is no more than one line that passes through both points. Therefore, the set B of lines with slopes s is a 1-configuration. Q.E.D.

Appendix H
Proof of Lemma 6.2

Consider a periodically extended lattice. We will show that it is not possible to construct a quadilateral (with no sides with infinite slope allowed), in which each point lies on two lines. Fig. 5 shows one such quadilateral. Without loss of generality, we can assume that the starting point of two of the lines is $(0, 0)$. The slopes of four lines in Fig. 13 are: $s, p, p + a/2$, and $s - a/2$. The points $(0, 0), (0, a), (2, 2s)$, and $(2, a+2p)$ are all different and each of them lies on two lines. The remaining four points will be on two lines if at lest one of the following three conditions holds:

1) $s = p + a/2$ and $s + a/2 = a + p$
2) $s = s + a/2$ and $p + a/2 = p + a$
3) $s = p + a$ and $p + a/2 = s + a/2$

with all additions performed modulo q. Case 1) implies $s - p = a/2$ which means that the points $(2, 2s)$ and $(2, a + 2p)$ are identical, a contradiction. For the remaining two cases, a would have to be 0, which would imply that the two leftmost points are identical, again leading to a contradiction. Q.E.D.

Appendix I
Proof of Theorem 6.1

Let c be an even integer. Assume that there exists a generalized Pasch configuration in the set of lines determined by the set of ordered pairs $[a_1; s_1], \ldots, [a_{c+1}; s_{c+1}]$, where a_i and s_i denote the starting point and the slope of the line, respectively.

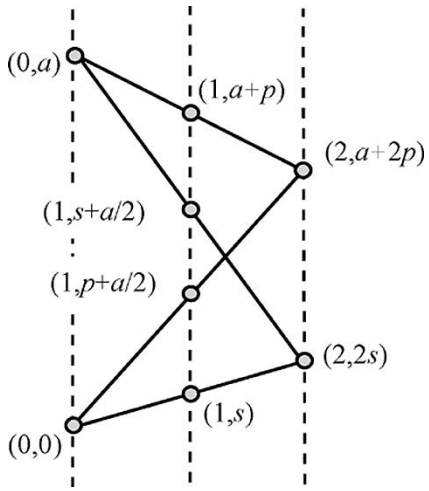

Fig. 13. Quadrilateral in a lattice finite geometry.

In a generalized Pasch configuration, each point has to occur exactly twice. This implies that for $x = 0$, the set of points $(0, a_1), \ldots, (0, a_{c+1})$ has to be such that for each i, $a_i = a_j$ for exactly one $j \neq i$. This is clearly impossible for even values of c. On the other hand, assume that $c = q$ is odd, and that the set of blocks forming a generalized Pasch configuration is again specified by the ordered pairs $[a_1; s_1], \ldots, [a_{c+1}; s_{c+1}]$. By rotating the set of lines in the generalized Pasch configuration appropriately, one of the lines, say $[a_1; s_1]$, can be taken to correspond to the line $y = 0$. Since the point $(0, 0)$ has to belong to exactly two lines, there must exist another line in the configuration specified by $[0; s]$, $s > 0$. Based on the previous discussion regarding the points with $x = 0$, it follows that without loss of generality, the blocks can be divided into $u = (c + 1)/2$ pairs as follows:

$$[0; 0], [0; s]$$
$$[a_2; s_1], [a_2; s_u]$$
$$\cdots$$
$$[a_u; s_{u-1}], [a_u; s_{2u-2}].$$

Here, $a_i \neq a_j \neq 0$, for $2 \leq i, j \leq u, s_l \neq s_{l+u-1}$ for any $l > 0$. We will show next that that all lines of the configuration, except for the one defined by $[0; 0]$, have to intersect the line $y = s$ (note that there are $c = q$ such lines). Since q is prime, for each pair $[a_i; s_i]$ there exists a unique value of x such that $a_i + s_i \cdot x \equiv s \bmod q$. This implies that all c lines differing from the line $y = 0$ intersect the line $y = s$, and all the points on this line have to belong to exactly two lines in the configuration. But this is impossible, since c is by assumption equal to the odd prime q. Q.E.D.

Note that for $c = q$ the parity-check matrix (5.1) is of dimension $q^2 \times q^2$ and hence of no practical importance for code description.

Appendix J
Proof of Theorem 6.2

In order to construct a 1-configuration with girth eight, one has to eliminate those lines from the lattice that lead to the formation of triangles. Before proceeding with the proof of the the-

orem, we need to briefly describe the relationship that exists between the set of slopes and triangles.

Consider a set of three different slopes: s_1, s_2, and s_3. Without loss of generality, the equations for lines with the given slopes are of the form

$$y = y_{0,i} + s_i \cdot x (\mathrm{mod}\ q), i = 1, 2, 3$$

where $0 \leq y_{0,i} \leq c-1$. From this point on, we will assume that all equalities are evaluated modulo q. If a set of three slopes determines a triangle, then the above set of equations has at least one solution and *vice versa*. In other words, a set of three slopes determines a triangle if and only if there exist three ordered pairs (x_{12}, y_{12}), (x_{23}, y_{23}), and (x_{13}, y_{13}) such that

$$y_{12} = y_{0,1} + s_1 x_{12},\ y_{12} = y_{0,2} + s_2 x_{12}$$
$$y_{23} = y_{0,2} + s_2 x_{23},\ y_{23} = y_{0,3} + s_3 x_{23}$$
$$y_{13} = y_{0,1} + s_1 x_{13},\ y_{13} = y_{0,3} + s_3 x_{13}$$

or equivalently

$$(s_1 - s_2)x_{12} + (s_2 - s_3)x_{23} + (s_3 - s_1)x_{13} = 0. \qquad \text{(J1)}$$

The last condition is identical to the condition for three slopes to form a triangle in the Euclidean space, except that for the case of interest, the equality is modulo q. It can also be shown that starting from (J1) one can prove that the slopes s_1, s_2, and s_3 define a triangle. The proof is straightforward and therefore omitted.

Observe that if three lines of slopes s_1, s_2, and s_3 form a triangle, then the same is true of the lines with slopes $s_1 + h$, $s_2 + h$, and $s_3 + h$ $(\mathrm{mod} q)$, for any integer h.

Lemma J.3: A set of three lines with slopes s_1, s_2, and s_3 forms a triangle if and only if there are two nonzero integers α and β, $-c < \alpha, \beta < c$, such that $\alpha + \beta \neq 0 \pmod{c}$ and

$$\alpha(s_2 - s_1) + \beta(s_3 - s_1) = 0.$$

Proof: An equivalent form of the condition given by (J1) is

$$(x_{23} - x_{12})(s_2 - s_1) + (x_{13} - x_{23})(s_3 - s_1) = 0.$$

According to this expression, neither α nor β can be equal to zero; if either α or β were zero, one would have $x_{23} - x_{12} = 0$, $x_{13} - x_{23} = 0$, $x_{13} - x_{12} = 0$, which is impossible by construction. For the same reason, one must have $\alpha + \beta \neq 0 \pmod{c}$. Q.E.D.

Since the x-coordinates of the intersection points must be larger then or equal to zero and smaller than c, α, and β also satisfy the following condition: $|\alpha + \beta| < c$.

We are now ready to prove Theorem 6.1.

Proof: According to the previous discussion, the criterion for including a slope s_3 in Σ_m is that, for all s_1 and s_2 already in Σ_m, it holds that

$$(x_{12} - x_{13})s_1 + (x_{23} - x_{12})s_2 + (x_{13} - x_{23})s_3 \neq 0 \bmod q$$

or equivalently

$$\alpha(s_2 - s_1) + \beta(s_3 - s_1) \neq 0 \bmod q \qquad \text{(J2)}$$

for all nonzero α and β such that $-(c-1) \leq \alpha, \beta \leq c-1$, and $\alpha + \beta \neq 0 \pmod{c}$, $|\alpha + \beta| < c$. According to the previous discussion, for the case $c = 3$, α and β have to be in $\{-2, -1, 1, 2\}$, and they either have different absolute values and opposite signs, or they both are equal to either 1 or -1. Assume that for some nonzero integer a, the set $\{s_1, s_2, s_3\}$ forms a three-term arithmetic progression, say

$$s_2 - s_1 = a, s_3 - s_1 = 2a.$$

Then the left-hand side of (J2) reduces to

$$a \cdot \alpha + 2 \cdot a \cdot \beta. \qquad \text{(J3)}$$

If this value were to be zero, then either both α and β would have to be zero, or α and β would have to be equal to 2 and -1, respectively. On the other hand, if the left-hand side of (J2) is equal to zero, then taking α and β with value 1 or -1 gives $s_2 - s_1 + s_3 - s_1 = 0$, which implies $s_2 - s_1 = s_1 - s_3$ or, equivalently, that s_3, s_1, s_2 form a three-term arithmetic progression.

To complete the proof we will show next that elements in Σ_m cannot be larger than $q/2$. If not stated otherwise, all numbers will be described in terms of their *ternary expansion*.

Case 1: Let the first (i.e., most significant) digit of the expansion of q be 1. Assume that there exists a slope s_3 in Σ_m that exceeds $q/2$. Then the number $d = q - s_3 < q/2$ has zero as its most significant digit. We can now subtract from d a number in Σ_m smaller than $q/2$, say s_1, whose ternary expansion consists only of 0's and 1's. In this way, we obtain a number that has only 2's and 0's in its expansion. This number can be viewed as the product of two and a number that has only 1's and 0's as its digits. This now would imply the existence of a triangle, hence excluding the possibility for s_3 to be in Σ_m (this construction corresponds to $\alpha = 1, \beta = -2$).

Case 2: Let the first (i.e., most significant) digit of q be 2. Assume that there exists a slope s_3 in Σ_m that exceeds $q/2$. Then the number $d = 2s_3 - q$ has zero as its most significant digit. Therefore, d can be written as the sum of two numbers smaller than $q/2$ whose ternary expansion consists only of 1's and 0's. Those two numbers specify s_1 and s_2, thus eliminating s_3 from inclusion in Σ_m (this construction corresponds to $\alpha = 1, \beta = -2$).

This completes the proof of the Theorem. Q.E.D.

ACKNOWLEDGMENT

The authors would like to thank Alexander Kuznjecov and Victor Kolesnik for their comments and suggestions, and Sundararajan Sankaranarayanan for running the simulations. The anonymous referees helped to significantly improve the presentation of the results, and the authors wish to thank them for their thorough reviews and helpful comments.

REFERENCES

[1] D. J. C. MacKay and R. M. Neal, "Good codes based on very sparse matrices," in *Cryptography and Coding, Proc. 5th IMA Conf. (Lecture Notes in Computer Science)*, C. Boyd, Ed. Berlin, Germany: Springer-Verlag, 1995, vol. 1025, pp. 110–111.

[2] R. G. Gallager, *Low-Density Parity-Check Codes*. Cambridge, MA: MIT Press, 1963.

[3] C. E. Shannon, "A mathematical theory of communication," *Bell Syst. Tech. J.*, pp. 372–423, 1948.

[4] D. J. C. MacKay, "Good error-correcting codes based on very sparse matrices," *IEEE Trans. Inform. Theory*, vol. 45, pp. 399–431, Mar. 1999.

[5] T. Richardson, A. Shokrollahi, and R. Urbanke, "Design of capacity-approaching irregular low-density parity-check codes," *IEEE Trans. Inform. Theory*, vol. 47, pp. 619–638, Feb. 2001.

[6] B. J. Frey, *Graphical Models for Machine Learning and Digital Communication*. Cambridge, MA: MIT Press, 1998.

[7] F. R. Kschischang and B. J. Frey, "Iterative decoding of compound codes by probability propagation in graphical models," *IEEE J. Select. Areas Commun.*, vol. 16, pp. 219–230, Feb. 1998.

[8] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. Inform. Theory*, vol. 47, pp. 498–519, Feb. 2001.

[9] R. J. McEliece, D. J. C. MacKay, and J.-F. Cheng, "Turbo decoding as an instance of Pearl's 'Belief propagation' algorithm," *IEEE J. Select. Areas Commun.*, vol. 16, pp. 140–152, Feb. 1998.

[10] R. M. Tanner, "A recursive approach to low complexity codes," *IEEE Trans. Inform. Theory*, vol. IT-27, pp. 533–547, Sept. 1981.

[11] N. Wiberg, H.-A. Loeliger, and R. Kötter, "Codes and iterative decoding on general graphs," *Eurp. Trans. Telecommun.*, vol. 6, pp. 513–525, Sept./Oct. 1995.

[12] Y. Kou, S. Lin, and M. Fossorier, "Construction of low-density parity-check codes: A geometric approach," in *Proc. 2nd Int. Symp. Turbo Codes*, Brest, France, Sept. 4–7, 2000.

[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. Inform. Theory*, vol. 47, pp. 2711–2736, Nov. 2001.

[14] R. M. Tanner, D. Srkdhara, and T. Fuja. A class of group-structured LDPC codes. [Online] Available: http://www.cse.ucsc.edu/~tanner/pubs.html

[15] J. Rosenthal and P. O. Vontobel, "Construction of LDPC codes using Ramanujan graphs and ideas from Margulis," in *Proc. 2001 IEEE Int. Symp. Information Theory*, Washington, DC, June 2001, p. 4.

[16] S. J. Johnson and S. R. Weller, "Regular low-density parity-check codes from combinatorial designs," in *Proc. 2001 IEEE Information Theory Workshop*, Cairns, Australia, Sept. 2–7, 2001, pp. 90–92.

[17] T. Beth, D. Jungnickel, and H. Lenz, *Design Theory*. Cambridge, U.K.: Cambridge Univ. Press, 1986.

[18] C. J. Colbourn, J. H. Dinitz, and D. R. Stinson, "Applications of combinatorial designs to communications, cryptography, and networking," in *Surveys in Combinatorics 1999*. Cambridge, U.K.: Cambridge Univ. Press, pp. 37–100.

[19] C. J. Colbourn and J. H. Dinitz, Eds., *The Handbook of Combinatorial Designs*. Boca Raton, FL: CRC Press, 1996.

[20] C. J. Colbourn and A. Rosa, *Triple Systems*. London, U.K.: Oxford Univ. Press, 1999.

[21] ——, *Steiner Systems*. London, U.K.: Oxford Univ. Press , 1999, Oxford Mathematical Monographs.

[22] B. Vasic, "Low-density parity-check codes: Theory and practice," presented at the National Storage Industry Consortium (NSIC) Quarterly Meeting, Monterey, CA, June 25–28, 2000.

[23] ——, "Structured iteratively decodable codes based on Steiner systems and their application in magnetic recording," in *Proc. GLOBECOM 2001*, vol. 5, San Antonio, TX, Nov. 26–29, 2001, pp. 2954–2960.

[24] B. Vasic, E. Kurtas, and A. Kuznetsov, "Lattice low-density parity-check codes and their application in partial response channels," in *Proc. 2002 IEEE Int. Symp. Information Theory*, Lausanne, Switzerland, June/July 2002, p. 453.

[25] R. C. Bose, "On the construction of balanced incomplete block designs," *Ann. Eugenics*, vol. 9, pp. 353–399, 1939.

[26] ——, "An affine analogue of Singer's theorem," *Ind. Math. Soc.*, vol. 6, pp. 1–5, 1942.

[27] E. Netto, *Zur Theorie der Tripelsysteme, Math. Ann.* , vol. 42, pp. 143–152, 1893.

[28] M. Buratti, "Construction of $(q, k, 1)$ difference families with q a prime power and $k = 4,5$," *Discr. Math.*, vol. 138, pp. 169–175, 1995.

[29] V. K. Bhargava and J. M. Stein, "(v, k, λ) configurations and self-dual codes," *Inform. Contr.*, vol. 28, pp. 352–355, 1975.

[30] E. Spence and V. D. Tonchev, "Extremal self-dual codes from symmetric designs," *Discr. Math.*, vol. 110, pp. 165–268, 1992.

[31] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error-Correcting Codes*: North-Holland, 1977.

[32] B. Vasic, "Combintorial constructions of structured low-density parity-check codes for iterative decoding," in *Proc. 2001 IEEE Information Theory Workshop*, Cairns, Australia, Sept. 2001.

[33] ——, "Combinatorial construction of low-density parity-check codes," in *Proc. IEEE Int. Symp. Information Theory*, Lausanne, Switzerland, June/July 2002, p. 312.

[34] ——, "High-rate low-density parity-check codes based on anti-Pasch affine geometries," in *Proc. IEEE Int. Conf. Communications (ICC 2002)*, vol. 3, New York, Apr./May, pp. 1332–1336.

[35] B. Vasic, E. Kurtas, and A. Kuznetsov, "LDPC codes based on mutually orthogonal Latin rectangles and their application in perpendicular magnetic recording," *IEEE Trans. Magn.*, pt. 1, vol. 38, pp. 2346–2348, Sept. 2002.

[36] B. Vasic and I. Djordjevic, "Low-density parity-check codes for long-haul optical communication systems," *IEEE Photonics Technol. Lett.*, vol. 14, pp. 1208–1210, Aug. 2002.

[37] B. Vasic, E. Kurtas, and A. Kuznetsov, "Kirkman systems and their application in perpendicular magnetic recording," *IEEE Trans. Magn.*, pt. 1, vol. 38, pp. 1705–1710, July 2002.

[38] D. MacKay and M. Davey. Evaluation of Gallager Codes for Short Block Length and High-Rate Applications. [Online]. Available: http://www.cs.toronto.edu/~mackay/CodesRegular.html

[39] R. M. Tanner, "Minimum-distance bounds by graph analysis," *IEEE Trans. Inform. Theory*, vol. 47, pp. 808–821, Feb. 2001.

[40] T. P. Kirkman, "Note on an unanswered prize question," *Cambridge and Dublin Math. J.*, vol. 5, pp. 255–262, 1850.

[41] A. C. H. Ling and C. J. Colbourn, "Rosa triple systems," in *Geometry, Combinatorial Designs and Related Structures*, J. W. P. Hirschfeld, S. S. Magliveras, and M. J. de Resmini, Eds. Cambridge, U.K.: Cambridge Univ. Press, 1997, pp. 149–159.

[42] J. H. van Lint and R. M. Wilson, *A Course in Combinatorics*. Cambridge, U.K.: Cambridge Univ. Press, 1992.

[43] N. Hamada, "On the p-rank of the incidence matrix of a balanced or partially balanced incomplete block design and its applications to error-correcting codes," *Hiroshima Math. J.*, vol. 3, pp. 153–226, 1973.

[44] R. Townsend and E. J. Weldon, "Self-orthogonal quasicyclic codes," *IEEE Trans. Inform. Theory*, vol. IT-13, pp. 183–195, Apr. 1967.

[45] E. J. Weldon Jr., "Difference-set cyclic codes," *Bell Syst. Tech. J.*, vol. 45, pp. 1045–1055, Sept. 1966.

[46] A. C. Ling, C. J. Colbourn, M. J. Grannell, and T. S. Griggs, "Construction techniques for anti-Pasch Steiner triple systems," *J. London Math. Soc.*, vol. 61, no. 3, pp. 641–657, June 2000.

[47] R. A. Beezer, "The girth of a design," *J. Comb. Math. and Comb. Comput.*, to be published.

[48] R. J. McEliece, *Finite Fields for Computer Scientist and Engineers*. Boston, MA: Kluwer Academic, 1987.

[49] R. M. Wilson, "Cyclotomy and difference families in elementary Abelian groups," *J. Number Theory*, vol. 4, pp. 17–47, 1972.

[50] I. Anderson, *Combinatorial Designs and Tournaments*. Oxford, U.K.: Oxford Science/Claredon, 1997.

[51] M. J. Colbourn and C. J. Colbourn, "Recursive construction for cyclic block designs," *J. Statist. Planning Infer.*, vol. 10, pp. 97–103, 1984.

[52] L. M. Batten, *Combinatorics of Finite Geometries*. London, U.K.: Cambridge Univ. Press, 1997.

[53] M. A. Margulis, "Explicit group-theoretic constructions for combinatorial designs with applications to expanders and concentrators," *Probl. Pered. Inform.*, vol. 24, no. 1, pp. 51–60, 1988.

[54] F. Karteszi, *Introduction to Finite Geomteries*. Amsterdam, The Netherlands: North-Holland, 1976.

[55] S. Weller and S. Johnson, "Iterative decoding of codes from oval designs," in *Proc. Defence Applications of Signal Processing, 2001 Workshop*, Adelaide, Australia, Sept. 2001.

[56] E. F. Assmus and J. D. Key, *Designs and their Codes*. London, U.K.: Cambridge Univ. Press, 1994.

[57] W. D. Wallis, *Combinatorial Designs*. New York: Marcel Dekker, 1988.

[58] J. Singer, "A theorem in finite projective geometry and some applications to number theory," *AMS Trans.*, vol. 43, pp. 377–385, 1938.

[59] C. C. Lindner and C. A. Rodger, *Design Theory*. Boca Raton, FL: CRC Press, 1997.

[60] M. Blaum, P. Farrell, and H. van Tilborg, "Array codes," in *Handbook of Coding Theory*, V. Pless and W. Huffman, Eds. Amsterdam, The Netherlands: Elsevier, 1998.

[61] J. L. Fan, "Array codes as low-density parity-check codes," in *Proc. 2nd Int. Symp. Turbo Codes and Related Topics*, Brest, France, Sept. 2000, pp. 543–546.

[62] E. Eleftheriou and S. Olcer, "Low-density parity-check codes for digital subscriber lines," in *Proc. IEEE Int. Conf. Communications (ICC 2002)*, vol. 3, 2002, pp. 1752–1757.

[63] J.-L. Kim, U. Peled, I. Perepelitsa, and V. Pless, "Explicit construction of families of LDPC codes with girth at least six," presented at the 40th Annual Allerton Conference on Communications, Control and Computing, Oct. 2002.

[64] J. Baker, F. Hiergeist, and G. Trapp, "The structure of multi-block circulants," *Kyungpook Math. J.*, vol. 25, no. 1, June 1985.

[65] W. W. Peterson and E. J. Weldon Jr., *Error-Correcting Codes*. Cambridge, MA: MIT Press, 1963.

[66] H. Song, J. Liu, and B. V. Kumar, "Low complexity LDPC codes for partial response channels," unpublished manuscript.

[67] F. Lazebnik and V. A. Ustimenko, "Explicit construction of graphs with arbitrary large girth and of large size," *Discr. Appl. Math.*, vol. 60, pp. 275–284, 1997.

[68] S. Litsyn and V. Shevelev, "On ensembles of low-density parity-check codes: Asymptotic distance distributions," *IEEE Trans. Inform. Theory*, vol. 48, pp. 887–908, Apr. 2002.

[69] M. Odlyzko and R. P. Stanley, "Some Curious Sequences Constructed with the Greedy Algorithm ," Bell Labs Internal Memo., 1978.

[70] N. J. Sloane and S. Plouffe, *The Encyclopedia of Integer Sequences*. San Diego, CA: Academic, 1995.

[71] B. Vasic, M. Ivkovic, K. Pedagani, and A. Cvetkovic, "High-rate girth-eight low-density parity-check codes on rectangular integer lattices," *IEEE Trans. Commun.*, submitted for publication.

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

[73] D. Mackay's Homepage [Online]. Available: www.inference.phy.cam.ac.uk/mackay/CodesFiles.html

[74] R. Niel's Homepage [Online]. Available: ftp://ftp.cs.utoronto.ca/pub/radford/LDPC-2001-11-18/index.html/