# Transactions Letters

## Resolvable 2-Designs for Regular Low-Density Parity-Check Codes

Sarah J. Johnson, *Student Member, IEEE,* and Steven R. Weller, *Member, IEEE*

***Abstract*—This paper extends the class of low-density parity-check (LDPC) codes that can be algebraically constructed. We present regular LDPC codes based on resolvable Steiner 2-designs which have Tanner graphs free of four-cycles. The resulting codes are $(3, \rho)$-regular or $(4, \rho)$-regular for any value of $\rho$ and for a flexible choice of code lengths.**

***Index Terms*—Combinatorial designs, iterative decoding, low-density parity-check (LDPC) codes.**

### I. Introduction

LOW-DENSITY parity-check (LDPC) codes were first presented by Gallager [13] in 1962 and created much interest when rediscovered and shown to perform remarkably close to the Shannon limit [6], [25]. Gallager proposed a decoding algorithm which utilizes the sparsity of the parity-check matrix to decode iteratively with complexity linear in the code length. It has since been realized that this algorithm is a special case of the sum-product decoding algorithm which is also used by turbo codes [20], [27].

It is known that the sum-product decoding algorithm converges to the optimal solution provided that the Tanner graph of the code satisfies a structural constraint, namely, that it is free of cycles [11], [27]. The existence of cycles in the Tanner graph prevents an exact error-probability analysis of iterative decoding procedures, and the shorter the cycles in the graph, the sooner the analysis breaks down [32]. Various improvements have been made to Gallager's original random construction method to avoid cycles and obtain the desired column and row weights [17], [23], [25], [31], [32]. As well, algebraic constructions of LDPC matrices have been proposed to provide regular codes with Tanner graphs free of four-cycles [10], [12], [18], [19], [21], [22], [24], [34], [36], [38].

In the special case of the binary-erasure channel, code performance with sum-product decoding can be determined explicitly, and is based solely on the stopping set distribution of the codes [9]. A stopping set is a set of codeword bit positions with the property that every parity-check equation, including a bit in the stopping set, is connected to at least two such bits. Recently, it was shown in [29] that if cycles of size four are avoided in the codes, then the size of the smallest stopping set is at least $\gamma + 1$, where $\gamma$ is the column weight of $H$.

The random constructions for LDPC codes produce parity-check matrices with small column weights, allow flexible choice of code parameters, and can be adapted to remove small cycles from the Tanner graph of the code. However, the resulting codes are irregular and the removal of all small cycles is difficult for higher rate codes. The algebraically constructed codes, on the other hand, offer guaranteed code properties including regularity, minimum distance, girth, and rate, although choice of code parameters is limited in existing constructions. Ideally, we seek codes with the flexible choice of code parameters of the random constructions, but with the benefits of algebraic construction.

A key idea in this paper is that resolvable 2-designs provide an algebraic construction method for a large class of regular LDPC codes with Tanner graphs free of four-cycles. The link between combinatorial designs and codes is not a new one (see, e.g., [2] and [26]) and many well-known codes are associated with designs, however, the methods for relating a design to a code are varied. The blocks of an orthogonal array, for example, can be used as the codewords of a nonbinary maximum distance separable code [26, p. 329], while the rows of the incidence matrix of a design can be used as the codewords of a nonlinear binary code [26, p. 63]. More recently, erasure-resilient codes for disk arrays have been constructed using the incidence matrix of a design as a portion of the parity-check matrix of the code [5].

Codes designed for majority logic decoding [37] using the incidence matrix of certain finite geometry designs as the parity-check matrix have recently shown excellent performances with sum-product decoding [22]. Like the codes for majority logic decoding, the codes for sum-product decoding are designed via their parity-check matrix, and the properties required for both are similar. The design of codes for majority logic decoding focused on the designs which give a larger column weight in $H$ because the minimum distance of the codes was a priority and increased column weight gives better minimum distance properties in the resulting code. However, when sum-product decoding is considered, the low density of the parity-check matrix is an important property of the codes and the designs we consider have an advantage in this area.

The codes presented in this work are $(3, \rho)$-regular (or $(4, \rho)$-regular); that is, all column weights of $H$ are equal to

Paper approved by M. Fossorier, the Editor for Coding and Communication Theory of the IEEE Communications Society. Manuscript received December 21, 2001; revised June 12, 2002 and March 22, 2003. This work was supported in part by a CSIRO Telecommunications and Industrial Physics postgraduate scholarship, and in part by Bell Laboratories Australia, Lucent Technologies, with the Australian Research Council under Linkage Project Grant LP0211210. This paper was presented in part at the IEEE Global Communications Conference (Globecom), San Antonio, TX, November 2001.

The authors are with the School of Electrical Engineering and Computer Science, University of Newcastle, Callaghan, NSW 2308, Australia (e-mail: sarah@ee.newcastle.edu.au; steve@ee.newcastle.edu.au).

Digital Object Identifier 10.1109/TCOMM.2003.816946

three (or four), and all row weights are $\rho$. An advantage of these codes is that the number of nonzero entries in the parity-check matrix increases only linearly with the code length. Further, the length of the code can be chosen independently of the rate, and existence is proved for an infinite class of code lengths for each allowable code rate.

As our construction is based on combinatorial design theory, we present in Section II some background material on designs before describing the construction of resolvable designs. Section III describes the LDPC codes from resolvable designs and outlines their properties. The performance of the codes when sum-product decoding is used is given in Section IV, and Section V concludes the paper.

## II. Resolvable 2-Designs

A combinatorial design is an arrangement of a set of $v$ points into $b$ subsets, called blocks. A design is regular if the number of points in each block, and the number of blocks which contain each point, designated $k$ and $r$, respectively, are the same for every point and block in the design. The number of blocks that two points $x$ and $y$ appear in together is denoted $\lambda_{x,y}$. A regular design is called a $t$ design if every $t$ points of the design are in a constant number $\lambda$ of blocks together. 2-designs are also called balanced incomplete block designs (BIBDs), and are denoted as 2-$(v,b,r,k,\lambda)$, or simply 2-$(v,k,\lambda)$ designs.

Every design can be described by a $b \times v$ incidence matrix $N$ where each row in $N$ represents a block $B_i$ of the design and each column a point $P_j$

$$N_{i,j} = \begin{cases} 1, & \text{if } P_j \in B_i \\ 0, & \text{otherwise.} \end{cases} \tag{1}$$

For a regular design, the number of ones in $N$ is $vr = bk$.

The incidence matrix of a combinatorial design is a binary matrix which can be used as the parity-check matrix $H$ of a binary LDPC code, with the proportion of nonzero entries in $H$ given by $k/v$. The parameters of an error-correcting code are denoted $[n, K, d]$, where $n$ is the code length, $K$ the code dimension, and $d$ the minimum distance of the code. When considering designs for LDPC codes, choosing a 2-design with $\lambda = 1$, called a Steiner 2-design, is beneficial as it guarantees the absence of four-cycles in the Tanner graph of the resulting code.

One class of Steiner 2-designs that have been proposed for generating LDPC codes are Steiner triple systems (STS), or $(v,b,r,3,1)$ designs [24]. These designs exist for all $v \equiv 1,3 (\mathrm{mod}\ 6)$ with $b = v(v-1)/6$ blocks. For STS designs $v < b$, and so the transpose of the STS incidence matrix is used as the parity-check matrix of an STS-LDPC code; this convention is retained in the remainder of the letter. The codes produced from STS designs are regular with $v$ parity checks, $b$ code bits, column weight of $H$ equal to three, and row weight of $H$ equal to $(v-1)/2$. The dimension of the code is $n - \mathrm{rank}_2(H)$ where $\mathrm{rank}_2(H)$ is the rank of $H$ over GF(2), also called the 2-rank of $H$. Since the dimension of the code is at least $b - v$, and since the code length $b = v(v-1)/6$ increases with the square of the number of parity checks, the codes quickly become high rate. It would be useful to have codes with the properties of STS codes but with a wider range of available rates for each codeword length.

A simplistic approach to obtaining a lower rate code is to choose a design with $b$ greater than the required code length and remove some columns of $N^T$. For each pair of points $x$, $y$ in the omitted column, the corresponding $\lambda_{x,y}$ is zero, and thus, the matrix so formed no longer represents the incidence matrix of a 2-design. Importantly, the incidence between the points and remaining blocks is retained and four-cycles will still be avoided in the Tanner graph of the resulting code. Randomly removing columns from $N^T$, however, results in a parity-check matrix with variable row weights, and can lead to rows with all entries zero. Ideally, we would like to be able to remove a group of columns of $N^T$ in such a way that we reduce by one the weight of every row in the matrix. For this, we need the design described by $N$ to be resolvable. A design is resolvable if the blocks of the design can be arranged into $r$ groups, called resolution classes, such that the $v/k$ blocks of each resolution class are disjoint, and each class contains every point precisely once. STS designs which are resolvable are called Kirkman triple systems (KTS). Unlike the STS designs which exist for all $v \equiv 1,3 (\mathrm{mod}\ 6)$, KTS designs on $v$ points, denoted KTS$(v)$, exist for all $v \equiv 1,3 (\mathrm{mod}\ 6)$ [7, p. 89, Th. 6.7].

### A. Resolvable Designs From Difference Systems

To generate the resolvable designs the concept of mixed difference systems is required. We first present background material on difference sets, systems, and mixed systems before presenting in Constructions 1 and 2 the KTS designs. Our treatment of difference systems follows Anderson [1], while the material on KTS designs is essentially Ray–Chaudri and Wilson's original presentation [30], using the terminology of Anderson [1].

Consider an arbitrary Abelian group $\mathcal{G}$ of order $v$. A $(v,k,\lambda)$ difference set is a $k$ subset of $\mathcal{G}$, $D = \{d_1, \ldots, d_k\}$, such that each nonzero element $g \in \mathcal{G}$ occurs exactly $\lambda$ times in the set of differences $\{d_i - d_j : d_i, d_j \in D\}$ [4, p. 330]. The translates of $D$ are the sets $D + g := \{d + g : d \in D\}$, for all elements $g \in \mathcal{G}$. The translates of a difference set in an Abelian group are the blocks of a symmetric Steiner 2-design with point set the elements of the group [1, p. 51]. The projective geometry designs used in [37] can be constructed from difference sets in this way. For example, the subset {1,2,4} of $Z_7$ is a difference set with differences

$$\begin{aligned} 1-2 &\equiv 6 (\mathrm{mod}\ 7), \quad 1-4 \equiv 4 (\mathrm{mod}\ 7), \\ 2-4 &\equiv 5 (\mathrm{mod}\ 7), \quad 2-1 \equiv 1 (\mathrm{mod}\ 7), \\ 4-1 &\equiv 3 (\mathrm{mod}\ 7), \quad 4-2 \equiv 2 (\mathrm{mod}\ 7) \end{aligned}$$

and the translates of {1,2,4} give the blocks of the projective geometry (7,7,3,3,1) design [1].

*Definition 1:* For an Abelian group $\mathcal{G}$ of order $v$, the $t$ $k$ element subsets $D_i = \{d_{i,1}, d_{i,2}, \ldots, d_{i,k}\}$ of $\mathcal{G}$ form a $(v,k,\lambda)$ difference system if the differences $d_{i,x} - d_{i,y}, (i = 1, \ldots, t; x, y = 1, \ldots, k)$ give each nonzero element of $\mathcal{G}$ exactly $\lambda$ times. The translates of the sets of a $(v,k,\lambda)$ difference system in $\mathcal{G}$ make up the blocks of a $(v, vt, kt, k, \lambda)$ design with points the elements of $\mathcal{G}$ [1, Th. 2.2.2]. For example, the subsets {1,2,5} and {1,3,9} of $Z_{13}$ form a difference system, and the translates of these sets are the blocks of a (13,26,6,3,1) design with point set $P = \{0, 1, \ldots, 12\}$.

An extension to the method of difference systems allows several copies of each element of an Abelian group to be used. For $\mathcal{G}$, an Abelian group of order $v$, let $\mathcal{H} = \mathcal{G} \times Z_t$. Then $\mathcal{H}$ consists of $tv$ elements, $t$ copies of each element of $\mathcal{G}$. An element $(a, i) \in \mathcal{H}$ represents the $i$th copy of the element $a$ in $\mathcal{G}$.

*Definition 2:* For $\mathcal{H} = \mathcal{G} \times Z_t$, the $k$ subsets $D_1, \ldots, D_s \in \mathcal{H}$ form a mixed difference system if there exists an integer $\lambda$ such that for every $i, j \in \{1, 2, \ldots, t\}$, every element $g \in \mathcal{G}$ occurs $\lambda$ times as the difference $(x, i) - (y, i)$, where $g = x - y$ and $(x, i), (y, i) \in D_1, \ldots, D_s$.

The translates of the set $D_l$ are the sets $D_l + g := \{(x+g, i) : (x, i) \in D_l\}$ for all $g \in \mathcal{G}$. The translates of the sets of a mixed difference system form the blocks of a $(tv, sv, sk/t, k, \lambda)$ design with point set the elements of $\mathcal{H}$ [1, Th. 2.4.1]. KTS designs can be constructed in this way, and we now present constructions for the mixed difference systems required for the KTS designs with $v = 3q$ and $v = 2q + 1$, $q$ a prime power.

*Construction 1:* [1, Th. 2.2.4] Let $q = 6m + 1$ be a prime power, $m$ an integer and take $\theta$, a primitive element of $\mathrm{GF}(q)$, so that $\theta^{6m} = 1$, $\theta^{3m} = -1$, and $\theta^{2m} + 1 = \theta^m$. The point set is $\mathcal{H} = \mathrm{GF}(q) \times Z_3$ and the mixed difference system consists of the sets

$$\begin{aligned} A &= \{0_1, 0_2, 0_3\} \\ B_{i,j} &= \left\{\theta_j^i, \theta_j^{i+2m}, \theta_j^{i+4m}\right\}, \quad 1 \le i \le m \\ C_{i,j} &= \left\{\theta_j^{i+m}, \theta_{j+1}^{i+3m}, \theta_{j+2}^{i+5m}\right\}, \quad 1 \le i \le m \\ D_{i,j} &= \left\{\theta_j^{i}, \theta_{j+1}^{i+2m}, \theta_{j+2}^{i+4m}\right\}, \quad 1 \le i \le m \end{aligned}$$

for $1 \le j \le 3$ (mod 3), where $\theta_j^i = (\theta^i, j) \in \mathcal{H}$. The sets $A$, $B_{i,j}$, and $C_{i,j}$ of the mixed difference system make up the blocks of one resolution class of a design, and each translate of these sets gives a further resolution class. Next, each set $D_{i,j}$ with its translates gives a resolution class; so we obtain a total of $9m + 1$ resolution classes and we have a KTS $(3q)$.

For example, take $\mathcal{H} = \mathrm{GF}(7) \times Z_3$, $m = 1$, $q = 7$, and $v = 21$. Choose $\theta = 3$ and the mixed difference system is

$$\begin{aligned} A &= \{0_1, 0_2, 0_3\} \\ B &= \{3_1, 6_1, 5_1\}, \quad \{3_2, 6_2, 5_2\}, \quad \{3_3, 6_3, 5_3\} \\ C &= \{2_1, 4_2, 1_3\}, \quad \{2_2, 4_3, 1_1\}, \quad \{2_3, 4_1, 1_2\} \\ D &= \{3_1, 3_2, 3_3\}, \quad \{6_2, 6_3, 6_1\}, \quad \{5_3, 5_1, 5_2\}. \end{aligned} \tag{2}$$

The sets $A$, $B$, and $C$ make up the blocks of the first resolution class of the design and the six translates of these sets make up the blocks of the next six resolution classes. The blocks in the second resolution class (the translate of $A$, $B$, and $C$ with $g = 1$) are

$$\begin{aligned} &\{1_1, 1_2, 1_3\}, \quad \{4_1, 0_1, 6_1\}, \quad \{4_2, 0_2, 6_2\}, \\ &\{4_3, 0_3, 6_3\}, \quad \{3_1, 5_2, 2_3\}, \quad \{3_2, 5_3, 2_1\}, \quad \{3_3, 5_1, 2_2\}. \end{aligned}$$

Next, the translates of each block in $D$ make up a resolution class; for the first block, $D_{1,1}$, this class is

$$\begin{aligned} &\{3_1, 3_2, 3_3\}, \quad \{4_1, 4_2, 4_3\}, \quad \{5_1, 5_2, 5_3\}, \\ &\{6_1, 6_2, 6_3\}, \quad \{0_1, 0_2, 0_3\}, \quad \{1_1, 1_2, 1_3\}, \quad \{2_1, 2_2, 2_3\}. \end{aligned}$$

Altogether, there are ten resolution classes, each with seven blocks, to give the KTS (21,70,3,10,1) design. Each block defines a row of a binary incidence matrix, as in (1), the transpose of which is a parity-check matrix for a [70,49,4] LDPC code.

*Construction 2:* [1, Th. 9.1.5] Let $q = 6m + 1$ be a prime power, $m$ an integer and take $\mathcal{H} = \mathrm{GF}(q) \times Z_t \cup \infty$. Choose $\theta$, a primitive element of $\mathrm{GF}(q)$, so that $\theta^{6m} = 1$ and $\theta^{3m} = -1$, and choose an integer $u$, so that $\theta^m + 1 = 2\theta^u$. Then the sets

$$\begin{aligned} A &= \{0_1, 0_2, \infty\} \\ B_i &= \left\{\theta_2^{i+u+m}, \theta_2^{i+u+3m}, \theta_2^{i+u+5m}\right\}, \quad 0 \le i \le m-1 \\ C_i &= \left\{\theta_1^{i}, \theta_1^{i+m}, \theta_2^{i+u}\right\}, \quad 0 \le i \le m-1 \\ D_i &= \left\{\theta_2^{i+2m+u}, \theta_1^{i+2m}, \theta_1^{i+3m}\right\}, \quad 0 \le i \le m-1 \\ E_i &= \left\{\theta_2^{i+4m+u}, \theta_1^{i+4m}, \theta_1^{i+5m}\right\}, \quad 0 \le i \le m-1 \end{aligned}$$

form a mixed difference system in $\mathcal{H}$. The sets of the mixed difference system partition the $2q + 1$ elements of $\mathcal{H}$ and make up the first resolution class. Each translate of the sets give a further resolution class, and we have a KTS$(2q + 1)$ design. Note that when forming the translates $g + \infty = \infty$.

For example, take $\mathcal{H} = \mathrm{GF}(7) \times Z_2 \cup \infty$ and $m = 1$, $q = 7$, and $v = 15$. Choose $\theta = 3$ and $u = 2$ and the required mixed difference system is

$$\begin{aligned} &A = \{0_1, 0_2, \infty\}, \quad B = \{3_2, 5_2, 6_2\}, \\ &C = \{1_1, 3_1, 2_2\}, \quad D = \{2_1, 6_1, 4_2\}, \quad E = \{4_1, 5_1, 1_2\}. \end{aligned} \tag{3}$$

These sets make up the six blocks of the first resolution class of the KTS(15), and each successive resolution class is obtained by forming translates of these sets. The first translate is

$$\begin{aligned} &\{1_1, 1_2, \infty\}, \quad \{4_2, 6_2, 0_2\}, \\ &\{2_1, 4_1, 3_2\}, \quad \{3_1, 0_1, 5_2\}, \quad \{5_1, 6_1, 2_2\}. \end{aligned}$$

If we take the ordering of the points to be $\{0_1, \ldots, 6_1, 0_2, \ldots, 6_2, \infty\}$, the first 20 columns of $N^T$ for this KTS (15,35,3,7,1) design are shown in Fig. 1. The entire incidence matrix of the KTS design provides the parity-check matrix for a [35,21,4] LDPC code, while if just the first four resolution classes (shown in Fig. 1) are used for $H$, the code has parameters [20,6,6].

Alternative constructions for STS designs exist which, while not producing resolvable designs, do produce designs which are 3-resolvable. That is, the blocks of the STS design on $v$ points can be grouped into classes of $v$ blocks with each point incident in exactly three of the blocks in the class. These 3-resolvable STS designs are cyclic, that is, whenever $\{a, b, c\}$ is a triple of the design, so is $\{a + 1, b + 1, c + 1\}$. The cyclic STS designs exist for all $v \equiv 1, 3$ (mod 6) except $v = 9$, with blocks the translates of a difference system in $Z_v$ [1, Sec. 8.3]. Cyclic STS designs were used in [34] and [35] to produce LDPC codes for magnetic recording channels. The codes presented in [34] and [35] cannot be derived from KTS designs (see, e.g., [7, p. 89, Th. 6.7]).

## III. LDPC Codes From Resolvable Steiner 2-Designs

KTS-LDPC codes can be constructed for any number of parity checks $m \equiv 3$ (mod 6). The number of resolution classes $\rho \in \{4, 5, \ldots, (v-1)/2\}$ determines the row weight of $H$, which is $\rho$, the code length $n = (\rho v)/3$, and the rate, $R \approx (\rho - 3)/\rho$. Conversely, for a given code rate $R \approx (\rho - 3)/\rho$, for $\rho$ any integer, KTS-LDPC codes exist for any block length $n \equiv 3/(1 - R) \bmod (6/(1 - R))$. The number

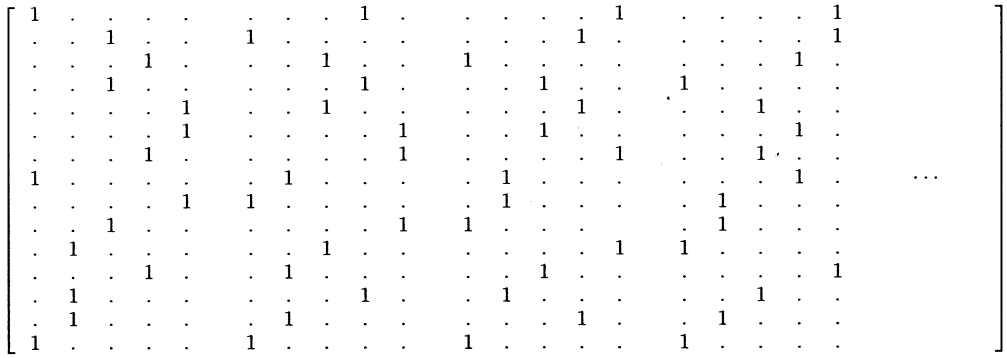

Fig. 1. First four resolution classes of a KTS(15) design.

of nonzero entries in the parity-check matrix of a KTS code is $3n$ and so increases only linearly with $n$. This will result in a decoding complexity advantage over those algebraically constructed codes which have code length as a function of column weight. For example, the LDPC codes from projective geometry designs have $n = q^2 + q + 1$, column weight $q + 1$, and $(q + 1)(q^2 + q + 1)$ nonzero entries in $H$.

For any given set of code parameters, there will be multiple choices of resolution classes. At present, the best method seems to be to choose a combination of resolution classes which gives good minimum distance results using Tanner's parity-check bound [33].

### A. Rank

If the 2-rank of the parity-check matrix of a code is known, we can determine the rate of the code exactly ($R = (n - \mathrm{rank}_2(H))/n$). For the STS designs, full-rank incidence matrices are guaranteed for all choices of $v \equiv 1, 9 \pmod{12}$, while for $v \equiv 3, 7 \pmod{12}$ the 2-rank of $N$ depends on the structure of the design (and hence, its construction) and is bounded as [2]

$$\mathrm{rank}_2(N) \geq v - \log_2(v + 1). \tag{4}$$

As the KTS designs are a special case of STS designs, the results above hold for them also. For the KTS designs presented in this letter, we observe that Construction 1 produces designs with full-rank incidence matrices, while Construction 2 produces designs with one linearly dependent row in their incidence matrix, corresponding to the point at $\infty$ which will remain linearly dependent in the KTS codes regardless of the selection of resolution classes. For all $v \leq 500$ corresponding to prime $q$, full rank (Construction 1) and rank $v - 1$ (Construction 2) codes have easily been constructed for any number of resolution classes greater than three. In fact, the majority of the possible selections of resolution classes produce codes of the maximum rank, but we leave as an open problem formally determining existence results.

### B. Cycles

As no two points in a Steiner 2-design can be incident in more than one block together, four-cycles are avoided in the Tanner graph of all LDPC codes obtained in this way. However, the requirement that every pair of points occur in exactly one block together guarantees the existence of six-cycles in the Tanner graph of the code. The exact number of six-cycles in the Tanner graph of a code from a Steiner 2-design on $v$ points with $k$ points in each block can be counted

$$N_6(v) = \binom{k}{2} \frac{v(v-1)(v-k)}{3k(k-1)}. \tag{5}$$

The number of six-cycles in the codes taking a subset of the blocks of a Steiner 2-design will, of course, be upper bounded by (5), as removing columns from a matrix cannot add cycles.

### C. Minimum Distance

The minimum distance of a code is equal to the minimum nonzero number of columns in the parity-check matrix for which a nontrivial linear combination sums to zero [39, p. 84]. The properties of Steiner 2-designs ensure that all columns in the parity-check matrix have weight $k$, and that no two columns share more than one point. Therefore, at least $k + 1$ columns are needed to sum to zero and $d_{\min} \geq k + 1$ for the codes presented in this letter.

For KTS designs to obtain codes with minimum distance of at least six, we need to establish the existence of KTS designs that lack collections of four blocks employing just six points, as this configuration of blocks will lead to a minimum distance of four. The particular configuration consisting of just four blocks and six points, with each block containing three points, and each point incident with precisely two blocks is called a Pasch configuration, or quadrilateral. The term anti-Pasch is used to describe a design that lacks a Pasch configuration. It has recently been proven that anti-Pasch STS designs exist for all $v$ for which STS designs exist [15]. Recent constructions have been found for anti-Pasch KTS with $v \equiv 9 \pmod{18}$ [5] and codes constructed from the resolution classes of these designs will have $d_{\min} \geq 6$. When using KTS designs which are not anti-Pasch, an option is to discard those resolution classes that involve a Pasch configuration when selecting resolution classes of the design to construct a KTS-LDPC code. For example, a rate-1/2 anti-Pasch LDPC code can be constructed from the KTS(21) in (2), by selecting the first, second, third, fourth, eighth, and ninth resolution classes.

The Pasch configuration is also the only possible configuration of bits and checks which results in a stopping set of size four in a KTS code. So by choosing an anti-Pasch KTS design, the minimum stopping set size is increased to five. To further increase this bound to six, mitre configurations [8] (which consist of just five blocks and seven points, with each block containing

three points, one point incident with three blocks and every other point incident with precisely two blocks) need to be avoided. Although some STS designs without both mitre and Pasch configurations, called five-sparse designs, have been found, a general existence result is lacking [8].

Resolvable Steiner 2-designs exist for $k \geq 4$, and these designs produce codes with larger column weights, and hence, larger minimum distances. However, as for randomly constructed codes, simulation results suggest that, in general, when the column weights of these codes are increased from three, a performance degradation results. In [10], it was shown that at very high rates array codes with column weight four performed well with sum-product decoding. In the following section, we will show that this is also true of resolvable Steiner 2-designs with $k = 4$. These resolvable $2-(v,4,1)$ designs can be constructed in a similar manner to KTS designs and exist for all $v \equiv 4 \pmod{12}$ [1], [16]. The codes from these designs have minimum distance $\geq 5$ and avoid stopping sets of size smaller than five.

### D. Implementation

One benefit of a deterministic construction is that the storage requirements necessary to completely describe the code are reduced. For the codes from KTS and resolvable $2-(v,4,1)$ designs, only the sets of the difference system are required; the translates can be constructed online, whereas for a random code, the entire parity-check matrix must be stored. For a code from Construction 1, this requires storage of $(v+3)/6$ sets of size three, while codes from Construction 2 require storage of $v/3$ sets of size three. The codes from resolvable $2-(v,4,1)$ designs require $v/4$ sets of size four to be stored. If storage is a significant issue, it is possible to specify only the required value of $m$ and the primitive element $\theta$ and the entire code can be constructed online with some expenditure in terms of computational complexity.

Alternatively, where hardwiring of the Tanner graph is employed [3], the regularity of the codes from KTS and resolvable $2-(v,4,1)$ designs translate directly into regularity in the very large scale integration (VLSI) layout. Further, if cyclically resolvable cyclic KTS designs [14] are employed, encoding can be achieved in linear time with shift register circuits in much the same way as for quasi-cyclic codes.

The resolution classes of the codes from resolvable Steiner 2-designs also offer a significant degree of flexibility when it comes to selecting code lengths and rates online. Once the sets of the difference family are stored, longer higher rate codes can be achieved simply by adding another translate to the code which adds to the number of message bits without changing the number of parity bits. The only information that needs to be communicated to completely specify the code in use is the resolution classes employed.

## IV. Simulation Results Using Iterative Decoding

We employed sum-product decoding, also known as belief propagation decoding, as presented in [23]. In the simulation results that follow, codes from KTS and resolvable $2-(v,4,1)$ designs are compared with randomly constructed codes and Euclidean geometry (EG) codes. The EG codes used are those from [21]. The random LDPC codes were constructed using the method from [25] and [28], and we have chosen parity-check matrices which lead to codes with Tanner graphs free of four-cycles. For all the codes presented in this section, the dimension of the code has been calculated exactly.

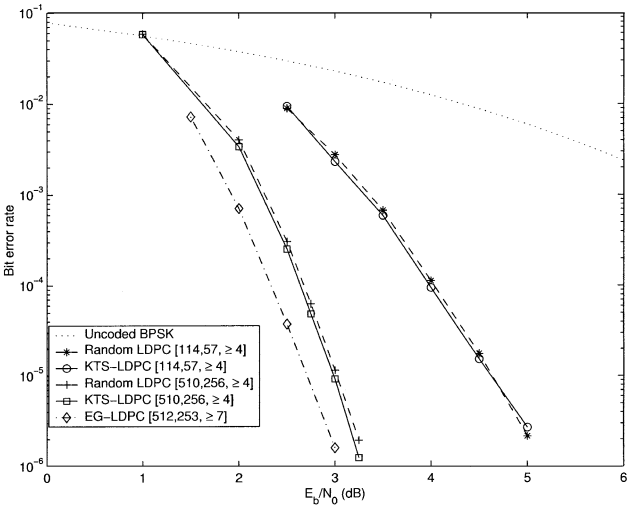


Fig. 2. Bit-error rate (BER) versus $E_b/N_0$ for rate 1/2 LDPC codes, maximum iterations $= 500$.

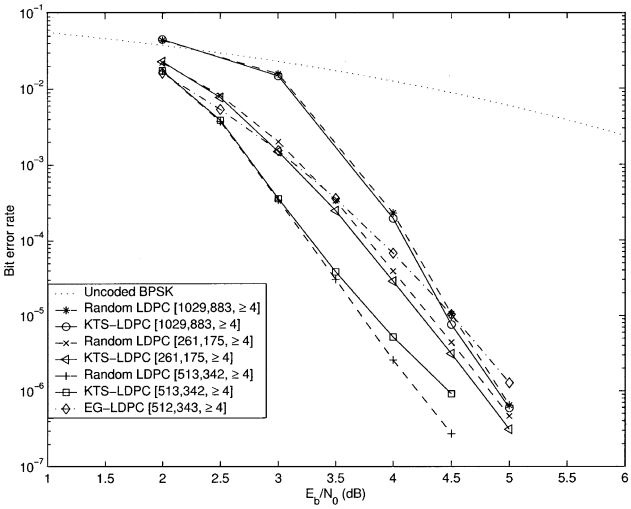


Fig. 3. BER versus $E_b/N_0$ for LDPC codes, maximum iterations $= 50$.

Fig. 2 shows the performance of rate-1/2 KTS, EG, and randomly generated LDPC codes. The KTS codes are from KTS(57) and KTS(255) designs. The performance of the KTS codes is similar to the performance of the randomly generated codes which have the same rate, codeword length, and an equal number of nonzero entries in $H$. All three length $n \simeq 510$ codes have similar block lengths and rates, however, they are not equally sparse. The EG code is (6,8)-regular while the KTS code is (3,6)-regular. While all codes have the same number of columns in their parity-check matrix, the EG code has more rows, some of which are linearly dependent. Consequently, the EG code has twice as many nonzero entries in its parity-check matrix, resulting in a significant increase in computational complexity for the same number of decoding iterations.

Fig. 3 shows the performance of rate-2/3 LDPC codes and two rate-0.86 LDPC codes. The KTS codes are from KTS(87),

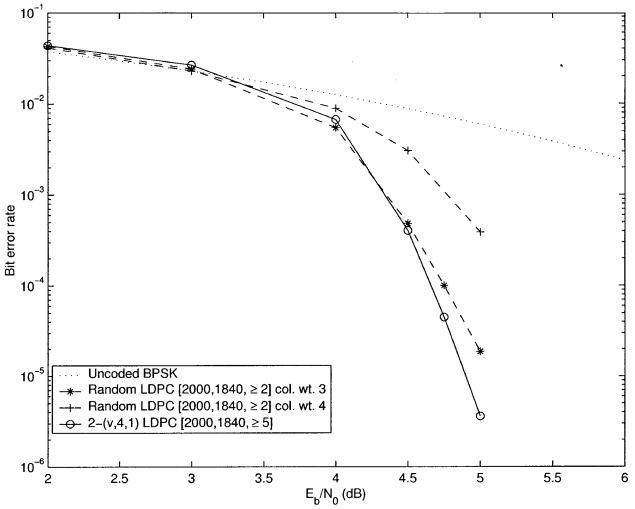


Fig. 4. BER versus $E_b/N_0$ for LDPC codes, maximum iterations $= 50$.

KTS(147), and KTS(171) designs. All three length $\simeq$513 codes have parity-check matrices with column weight three and so require equal computational complexity for decoding, as do both length 1029 codes. The KTS codes perform as well as randomly constructed codes.

Fig. 4 shows the performance of high-rate length 2000 codes. The column weight four code from the resolvable 2-(160,4,1) design significantly outperforms both the column weight three and column weight four random LDPC codes. In neither case were we able to construct random codes at this high rate which were completely free of four-cycles.

## V. Conclusion

We have presented a construction method for LDPC codes based on the resolution classes of resolvable Steiner 2-designs. The method produces very sparse parity-check matrices having constant column and row weight, girth equal to six and with a flexible choice of code parameters. We have shown by considered application of design theory that it is not necessary for LDPC codes to be constructed randomly to achieve good decoding performances at moderate lengths and for a wide range of code parameters.

## Acknowledgment

The authors wish to thank the reviewers and editor whose constructive comments and suggestions greatly improved this letter, and Prof. R. M. Neal for his online repository of LDPC-related software.

## References

[1] I. Anderson, "Combinatorial designs: construction methods," in *Mathematics and Its Applications*. Chichester, U. K.: Ellis Horwood, 1990.

[2] E. F. Assmus, Jr. and J. D. Key, "Designs and their codes," in *Cambridge Tracts in Mathematics*. Cambridge, U. K.: Cambridge Univ. Press, 1993, vol. 103.

[3] A. J. Blanksby and C. J. Howland, "A 690-mW 1-Gb/s 1024-b, rate-1/2 low-density parity-check code decoder," *IEEE J. Solid-State Circuits*, vol. 37, pp. 404–412, Mar. 2002.

[4] P. J. Cameron and J. H. van Lint, "Graphs, codes and designs," in *London Mathematical Society Lecture Note Series*. Cambridge, U. K.: Cambridge Univ. Press, 1980, vol. 43.

[5] Y. M. Chee, C. J. Colbourn, and A. C. H. Ling, "Asymptotically optimal erasure-resilient codes for large disk arrays," *Discrete Appl. Math.*, vol. 102, no. 1–2, pp. 3–36, May 2000.

[6] S.-Y. Chung, G. D. Forney, Jr., and T. J. Richardson, "On the design of low-density parity-check codes within 0.0045 dB of the Shannon limit," *IEEE Commun. Lett.*, vol. 45, pp. 58–60, Feb. 2001.

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

[8] C. J. Colbourn, E. Mendelsohn, A. Rosa, and J. Siran, "Anti-mitre Steiner triple systems," *Graphs Combin.*, vol. 10, pp. 215–224, 1994.

[9] C. Di, D. Proietti, I. E. Telatar, T. J. Richardson, and R. L. Urbanke, "Finite-length analysis of low-density parity-check codes on the binary erasure channel," *IEEE Trans. Inform. Theory*, vol. 48, pp. 1570–1579, June 2002.

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

[11] T. Etzion, A. Trachtenberg, and A. Vardy, "Which codes have cycle-free Tanner graphs?," *IEEE Trans. Inform. Theory*, vol. 5, pp. 2173–2181, Sept. 1999.

[12] J. L. Fan, "Array codes as low-density parity-check codes," in *Proc. 2nd Int. Symp. Turbo Codes*, Brest, France, Sept. 4–7, 2000, pp. 543–546.

[13] R. G. Gallager, "Low-density parity-check codes," *IRE Trans. Inform. Theory*, vol. IT-8, pp. 21–28, Jan. 1962.

[14] M. Genma, M. Mishima, and M. Jimbo, "Cyclic resolvability of cyclic Steiner 2-designs," *J. Combin. Des.*, vol. 5, no. 3, pp. 177–187, May 1997.

[15] M. J. Grannell, T. S. Griggs, and C. A. Whitehead, "The resolution of the anti-Pasch conjecture," *J. Combin. Des.*, vol. 8, no. 4, pp. 300–309, July 2000.

[16] H. Hanani, D. K. Ray-Chaudri, and R. M. Wilson, "On resolvable designs," *Discrete Math.*, vol. 3, pp. 343–357, 1972.

[17] D. Hösli, E. Svensson, and D. Arnold, "High-rate low-density parity-check codes: construction and application," in *Proc. 2nd Int. Symp. Turbo Codes*, Brest, France, Sept. 4–7, 2000, pp. 447–450.

[18] S. J. Johnson and S. R. Weller, "Codes for iterative decoding from partial geometries," in *Proc. Int. Symp. Information Theory*, Lausanne, Switzerland, June 30–July 5, 2002, p. 310.

[19] 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.

[20] 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.

[21] S. Lin, H. Tang, Y. Kou, J. Xu, and K. Abdel-Ghaffar, "Codes on finite geometries," in *Proc. IEEE Information Theory Workshop*, Cairns, Australia, Sept. 2001, pp. 14–16.

[22] R. Lucas, M. P. C. Fossorier, Y. Kou, and S. Lin, "Iterative decoding of one-step majority logic decodable codes based on belief propagation," *IEEE Trans. Commun.*, vol. 48, pp. 931–937, June 2000.

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

[24] D. J. C. MacKay and M. C. Davey, "Evaluation of Gallager codes for short block length and high rate applications," in *Codes, Systems and Graphical Models; Volume 123 of IMA Volumes in Mathematics and Its Applications*, B. Marcus and J. Rosenthal, Eds. New York: Springer-Verlag, 2000, vol. 123, pp. 113–130.

[25] D. J. C. MacKay and R. M. Neal, "Near-Shannon limit performance of low density parity check codes," *Electron. Lett.*, vol. 32, no. 18, pp. 1645–1646, Mar. 1996.

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

[27] 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.

[28] , R. M. Neal. [Online]. Available: www.cs.toronto.edu/radford/homepage.html

[29] A. Orlitsky, R. L. Urbanke, K. Viswanathan, and J. Zhang, "Stopping sets and the girth of Tanner graphs," in *Proc. Int. Symp. Information Theory*, Lausanne, Switzerland, June 30–July 5, 2002, p. 2.

[30] D. K. Ray-Chaudhuri and R. M. Wilson, "Solution of Kirkmans schoolgirl problem," in *Proc. Symp. Mathematics*, vol. 19, 1971, pp. 187–203.
[31] M. Sipser and D. A. Spielman, "Expander codes," *IEEE Trans. Inform. Theory*, vol. 42, pp. 1710–1722, Nov. 1996.
[32] V. Sorokine, F. R. Kschischang, and S. Pasupathy, "Gallager codes for CDMA applications—part I: Generalizations, constructions, and performance bounds," *IEEE Trans. Commun.*, vol. 48, pp. 1660–1668, Oct. 2000.
[33] R. M. Tanner, "Minimum-distance bounds by graph analysis," *IEEE Trans. Inform. Theory*, vol. 47, pp. 808–821, Feb. 2001.
[34] B. Vasic, "Structured iteratively decodable codes based on Steiner systems and their application in magnetic recording," in *Proc. IEEE Globecom Conf.*, San Antonio, TX, Nov. 2001, pp. 2954–2960.
[35] B. Vasic, E. M. Kurtas, and A. V. Kuznetsov, "Kirkman systems and their application in perpendicular magnetic recording," *IEEE Trans. Magn.*, vol. 38, pp. 1705–1710, July 2002.
[36] P. O. Vontobel and R. M. Tanner, "Construction of codes based on finite generalized quadrangles for iterative decoding," in *Proc. Int. Symp. Information Theory*, Washington, DC, June 24–29, 2001, p. 223.
[37] E. J. Weldon, "Difference-set cyclic codes," *Bell Syst. Tech. J.*, vol. 7, pp. 1045–1055, Sept. 1966.
[38] S. R. Weller and S. J. Johnson, "Regular low-density parity-check codes from oval designs," *Eur. Trans. Telecommun.*, to be published.
[39] S. B. Wicker, *Error Control Systems for Digital Communication and Storage*. Upper Saddle River, NJ: Prentice-Hall, 1995.