Construction of low-density parity-check codes from Kirkman triple systems

Sarah J. Johnson and Steven R. Weller
sarah@ee.newcastle.edu.au, steve@ee.newcastle.edu.au
Department of Electrical and Computer Engineering
University of Newcastle, Callaghan, NSW 2308, Australia

Abstract— **Gallager introduced LDPC codes in 1962, presenting a construction method to randomly allocate bits in a sparse parity-check matrix subject to constraints on the row and column weights. Since then improvements have been made to Gallager's construction method and some analytic constructions for LDPC codes have recently been presented. However, analytically constructed LDPC codes comprise only a very small subset of possible codes and as a result LDPC codes are still, for the most part, constructed randomly. This paper extends the class of LDPC codes that can be systematically generated by presenting a construction method for regular LDPC codes based on combinatorial designs known as Kirkman triple systems. We construct $(3, \rho)$-regular codes whose Tanner graph is free of 4-cycles for any value of ρ divisible by 3.**

I. Introduction

Low-density parity-check (LDPC) codes were discovered by Gallager [1] in 1962 and have recently been rediscovered [2], [3]. LDPC codes are designed by specifying a parity-check matrix H so that the relationship between code bits and parity-check sums can be adjusted to optimize the flow of information in the decoding process. In particular, H is chosen to be sparse so that the calculation of each check sum depends on few code word bits and the evaluation of code bit validity on few check sums. Using this property of LDPC codes Gallager presented iterative decoding algorithms whose complexity remains linear in the block length [4]. Recently it has been shown that the encoding complexity of LDPC codes can also be linear in the block length [5]. When iteratively decoded using belief propagation, also known as sum-product decoding, [6], [7], LDPC codes can perform remarkably close to the Shannon limit [2].

A Tanner graph displays the relationship between codeword bits and parity checks and is a useful way to describe LDPC codes [3]. Each of the n code bits, and m parity checks in H are represented by a vertex in the graph. A graph edge joins a code bit vertex to the vertices of the parity checks that include it. As no parity check is connected to another by an edge, and no two code bits are connected, the graph is *bipartite*. The number of edges connected to a code bit vertex is the degree of that code bit, which is simply the number of parity check equations that include it. Likewise, the degree of a parity check vertex is the number of bits in the parity check equation. An LDPC code is said to be *regular* if the degrees of all code bit vertices are equal and the degrees of all parity check vertices are equal. For a regular LDPC code, the weight γ of each column is the degree of each parity check vertex, while the weight ρ of each row is the degree of each code bit vertex. Such a code is said to be (γ, ρ)-regular.

It is known that the iterative sum-product decoding algorithm converges to the optimal solution provided the Tanner graph of the code satisfies a structural constraint—namely, that it be free of *cycles* [8], [9]. A cycle in a Tanner graph is a sequence of connected code bits and check sums which start and end at the same vertex in the graph, and which contain other vertices no more than once. The length of the cycle is simply the number edges it contains and the *girth* of a Tanner graph is the size of its smallest cycle.

The existence of short cycles in the Tanner graph prevents an exact error-probability analysis of iterative decoding procedures, and the shorter are the cycles in the graph, the sooner the analysis breaks down. To date, randomly constructed LDPC codes have largely relied on the sparsity of the parity-check matrix to avoid short cycles in the Tanner graph.

A key idea in this paper is that the presence of cycles of length 4 in the Tanner graph associated with an LDPC code can be systematically avoided by taking as parity-check matrices the incidence matrices of suitably chosen combinatorial designs.

Gallager's original construction of LDPC codes involved random assignments of ones to positions in H subject to the constraints that each column of H have a small uniform weight, that the weight per row also be uniform, and that there are no 4-cycles in the Tanner graph of H. Various improvements have been made to Gallager's original construction method to avoid cycles and obtain the desired column and row weights [2], [6], [10]. When the block lengths are small, however, good LDPC codes become more difficult to find using these random construction methods [11]. So for small block lengths in particular, an analytic construction method that guarantees

1. small, uniform row and column weights; and
2. the absence of 4-cycles in the Tanner graph

is expected to be particularly useful.

MacKay [6] found that iteratively decoded LDPC codes perform best with column weight 3 and that their performance degrades as the column weight is increased. Our construction provides $(3, \rho)$-regular codes for any value of ρ divisible by 3. Selecting $(3, 6)$-regular LDPC codes in particular can provide an advantage in terms of encoding complexity. Richardson and Urbanke [5] have recently shown that the actual number of operations required to encode $(3, 6)$-regular codes is no more than $0.017^2 n^2$.

In this paper we present a construction for a family of parity-check matrices having column weight 3, and that satisfy both items 1 and 2 above. As our construction is based on combinatorial design theory, we present in Section II of this paper some background material on designs before describing the constructions and some of their properties in Section III. Section IV details the performance of our LDPC codes when decoded using the sum-product algorithm. Section V concludes the body of

Work supported by a CSIRO Telecommunications & Industrial Physics postgraduate scholarship, and by the Centre for Integrated Dynamics and Control.

the paper, and explicit constructions of Kirkman triple systems are presented in the Appendix.

II. Combinatorial designs

A combinatorial design is an arrangement of a set $\mathcal{P}$ of m *points* into n subsets, called *blocks*, which satisfy certain regularity conditions. A design is *regular* if the number of points γ in each block B, and the number of blocks ρ which contain each point P are the same for every point and block in the design. The covalency λ_{xy} of points x and y is the number of blocks that contain them both. A design is *balanced* if λ_{xy} is a constant for all x and y, the covalency of the design is then λ. A regular balanced design is often denoted as a $(m, n, \rho, \gamma, \lambda)$-design.

Every design can be described by an $m \times n$ incidence matrix I where each column in I represents a block B_j of the design and each row a point P_i:

$$I_{i,j} = \begin{cases} 1 & \text{if } P_i \in B_j, \\ 0 & \text{otherwise.} \end{cases}$$

For a regular design the number of ones in I is

$$m \times \rho = n \times \gamma. \tag{1}$$

The incidence matrix of a combinatorial design can be used as the parity-check matrix of a binary LDPC code to give favorable properties to the code. A $(m, n, \rho, \gamma, \lambda)$-design will give an $m \times n$ parity-check matrix H having m parity-check equations, n codeword bits, row weight ρ and column weight γ. Choosing a design with $\lambda = 0$ or 1 guarantees the absence of 4-cycles in the code. As is the case for random constructions of parity-check matrices, the H constructed in this way are not necessarily full rank in which case the number of message bits in the code is $n - \text{rank}(H)$.

One class of combinatorial designs that have been proposed for generating LDPC codes are *Steiner triple systems* on m points, or $(m, n, \rho, 3, 1)$-designs [11], denoted simply as STS(m). These designs exist for all $m \equiv 1, 3 \pmod 6$, are regular with column weight 3, row weight $(m-1)/2$, and are free of 4-cycles. The resulting codes (STS-LDPC codes) have codeword length $n = m(m-1)/6$, producing high rate codes even for small codeword lengths. If the restriction that $\lambda = 1$ is relaxed to allow $\lambda = 1$ or 0 this difficulty can be avoided. A simplistic approach is to remove some columns of H; for each pair of points in the omitted column the corresponding λ_{xy} will be 0. However, this results in a parity-check matrix with variable row weights, in many cases as low as 1 or 0, leading to performance penalties when iteratively decoded.

The key idea presented in this paper is to use a class of designs called *Kirkman triple systems* (KTS) to derive regular $(m, n, \rho, \gamma, \{1, 0\})$-designs. By definition, Kirkman triple systems are *resolvable* Steiner triple systems. That is, the blocks of a Kirkman triple system, with row weight ρ, can be arranged into ρ groups such that the n/ρ blocks of each group are disjoint, each group contains every point precisely once, and thus the weight of each row in a group is one. Consequently, if all blocks in a group are removed from H what remains is a parity-check matrix H_0 with row weight $\rho - 1$ containing $\frac{(\rho-1)n}{\rho}$ blocks. If we retain the blocks from just two groups, for example, the result is a $(3, 6)$-regular code without 4-cycles. (As there were no 4-cycles in the original H removing columns cannot add any).

In general, we can take any KTS and use one or more of its groups to make up our parity-check matrix. The resulting H has the same number of parity-checks as the original, still has column weight 3 and no 4-cycles, but can have any desired row weight $\rho \in \{1, 2, \ldots, (m-1)/2\}$. The choice of ρ determines the number of code bits n in accordance with equation (1). Kirkman triple systems exist for all $m \equiv 3 \pmod 6$ [12]. Construction methods for $m = 3q$ and $m = 2q + 1$, q a prime power, are given in [13], and for ease of reference are presented in the Appendix.

III. LDPC codes from combinatorial designs

In this section we apply combinatorial designs to construct LDPC codes. For a given selection of code parameters there may exist multiple, non-isomorphic, designs. Two designs are *isomorphic* if there exists a one-to-one mapping between their incidence matrices, that is, the incidence matrix of one can be obtained via column and row permutations of the other. There are 80 non-isomorphic STS $(15, 35, 7, 3, 1)$-designs, and while it is not known exactly how many non-isomorphic designs exist for larger Steiner triple systems, the number rapidly increases with the size of H. For example, there are at least 2 million non-isomorphic STS on 19 points, and it is conjectured that the true figure may be as high as 12 billion [14]. The vast majority of the non-isomorphic Steiner triple systems for any given design are full rank [14]. Thus when considering a particular code we have assumed a full rank design has been chosen and therefore the rate of the code is $\frac{n-m}{n}$.

In addition to Steiner and Kirkman triple systems we examine:

- Codes from binary Euclidean geometries (EG-LDPC), which can be described as $(q^2-1, q^2-1, q, q, \{0, 1\})$-designs, and codes from binary projective geometries (PG-LDPC), which can be described as $(q^2+q+1, q^2+q+1, q+1, q+1, \{0, 1\})$-designs, q any prime, and have been investigated in [7], [15], [16]. While these designs produce square parity-check matrices not all rows are linearly independent and code rate depends on rank(H). An important feature of this construction method is that it produces cyclic LDPC codes resulting in a simplified encoding process.
- A construction for LDPC codes using Ramanujan graphs is presented in [17]. The advantage of this method of construction is that it produces $(3, 6)$-regular codes. However these codes can only be constructed for block lengths $n = 2(q^3 - q)$ where q is prime.

As can be seen in Fig. 1, one advantage of using Kirkman triple systems to construct $(3, \rho)$-regular LDPC codes is the wealth of code rates and block lengths that are available.

A. Girth

A simple lower bound on the block length of an LDPC code avoiding cycles of size c can be found by considering the associated Tanner graph. Our line of reasoning is similar to, though extends, Lemma 1 presented in [17] lower bounding the girth of LDPC codes for any girth $\equiv 2 \pmod 4$. Consider an arbitrary bit vertex n_1 which is connected to γ parity-check vertices. Each of these is in turn connected to $\rho - 1$ bit vertices other than n_1. If any of these $\gamma(\rho - 1)$ bit vertices are the same a 4-cycle results. (Say checks p_1 and p_2 on n_1 both contain bit n_2 then the edges

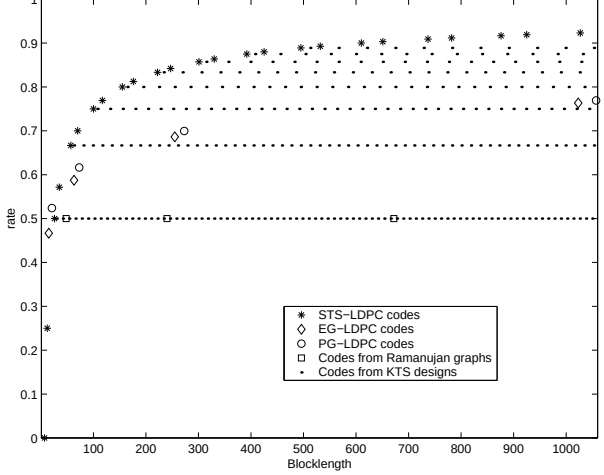

Fig. 1. Available block lengths and rates of analytically constructed LDPC codes

from n_1 to p_1 to p_2 to n_2 and back to n_1 form a cycle containing 4 edges). Thus to avoid 4-cycles there must be at least

$$n \geq \gamma(\rho - 1) + 1$$

code word bits.

Now consider the $\gamma(\rho - 1)$ bit vertices above; each is connected to a further $\gamma - 1$ parity-check vertices. To avoid both 4- and 6-cycles these $\gamma(\rho - 1)(\gamma - 1)$ vertices and the γ other parity-check vertices already connected to n_1 must be distinct. Thus to avoid 6-cycles

$$m \geq \gamma(\rho - 1)(\gamma - 1) + \gamma.$$

Similarly, we can start with an arbitrary parity-check vertex p_1, and follow the reasoning above to get the following restrictions. To avoid 4-cycles,

$$m \geq \rho(\gamma - 1) + 1,$$

and to avoid 6-cycles,

$$n \geq \rho(\gamma - 1)(\rho - 1) + \rho.$$

This reasoning can be extended to any cycle size c to obtain the following relationship between the block length n needed to avoid a cycle size c, and the parity-check and codeword bit degrees, γ and ρ, respectively:

$$\begin{aligned} n &\geq 1 + \gamma(\rho - 1) + \cdots + \gamma(\rho - 1)\alpha^{\frac{c}{4}-1}, \\ &\qquad c \equiv 0 \pmod 4 \qquad (2) \\ n &\geq \rho + \rho\alpha + \cdots + \rho\alpha^{\frac{-2c}{4}}, \\ &\qquad c \equiv 2 \pmod 4, \qquad (3) \end{aligned}$$

where $\alpha = (\gamma - 1)(\rho - 1)$.

The inequalities in (2)–(3), and in particular the restriction on n to avoid 6-cycles, can be used to prove the following.

Lemma 1: The girth of any STS-LDPC, EG-LDPC, or PG-LDPC code is 6.

Proof: In each case, we use the appropriate design parameters and substitute into equations (2)–(3) for $c = 6$.

Using the properties of STS designs, $\gamma = 3$, $\rho = (m-1)/2$, and $n = m(m-1)/6$, we obtain an inequality in m:

$$\frac{m(m-1)}{6} \geq 2\frac{(m-1)}{2}\left[\frac{m-1}{2} - 1\right] + \frac{m-1}{2}$$

or, equivalently, $m \leq 3$. So for any $m > 3$, 6-cycles must exist in STS-LDPC codes. Further, for $m = 3$, the blocklength is 1, which is not a valid code.

Using the properties of projective geometries, $\rho = \gamma = q + 1$, and $n = q^2 + q + 1$ we get an inequality in q:

$$q^2 + q + 1 \geq (q+1)q^2 + (q+1)$$

so that $q \leq 0$, which cannot be met for any prime q, implying the existence of 6-cycles in all PG-LDPC codes.

Finally, using the properties of Euclidean geometries, $\rho = \gamma = q$, and $n = q^2 - 1$ we get an inequality in q:

$$q^2 - 1 \geq q(q-1)(q-1) + q,$$

so that

$$q(q^2 + 2) + 1 \leq 3q^2,$$

which cannot be met for any prime q, hence 6-cycles must exist in all EG-LDPC codes. Further, the existence of cycles shorter than 6 are excluded by the restriction that $\lambda = 0$ or 1 and the result follows. ■

No such bounds can be placed on the regular codes derived from Kirkman triple systems, or the regular codes derived from Ramanujan graphs. These codes have constant row and column weight as n increases and so their density decreases, allowing the girth to grow with n.

B. Minimum distance

For regular LDPC codes whose parity-check matrix is the incidence matrix of a Steiner triple system, MacKay and Davey [18, Theorem 1] showed that the minimum distance is at most 10. While this upper bound on minimum distance is so small as to preclude the use of Steiner triple system LDPCs for all but the shortest block lengths, it is possible to systematically construct STS LDPCs having minimum distance at least 6 for all m for which STS(m) exist except for $m = 7$ or 13, where the minimum distance is 4.

To see this, recall that 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 [19, p. 84]. The definition of Steiner triple systems ensures that all columns in the incidence (i.e. parity-check) matrix have weight 3, and that no two columns share more than one point. Therefore at least 4 columns are needed to sum to zero.

To show that the minimum distance is at least 6, we need to establish the existence of Steiner triple systems that lack collections of 4 blocks employing just 6 points. In the combinatorial design literature, collections of blocks in incidence structures are referred to as *configurations*. The particular configuration consisting of just 4 blocks and 6 points, with each block containing 3 points, and each point incident with precisely 2 blocks is called a *Pasch configuration*, or *quadrilateral*. The term *anti-Pasch* is used to describe a Steiner triple systems that lacks a Pasch configuration.

It is well known that Steiner triple systems of order m exist if and only if $m \equiv 1, 3 \pmod 6$, and it had been a long standing conjecture that anti-Pasch Steiner triple systems also exist for all these values except for $m = 7$ or 13, for which they are known not to exist. For the case $m \equiv 3 \pmod 6$, an anti-Pasch STS(m) is known to exist [20], [21]. The case in which $m \equiv 1 \pmod 6$ proved to be much more problematical, but after some twenty years has recently been solved [22], [23].

We therefore have the following result, and refer the reader to the papers [20], [24], [21], [25], [22], [23] for explicit constructions of the associated anti-Pasch Steiner triple systems.

Lemma 2: For all $m \equiv 1, 3 \pmod 6$ except for $m = 7, 13$, there exist regular LDPC codes of length $m(m-1)/6$, having parity check matrices with uniform column weight 3, uniform row weight $(m-1)/2$ and minimum distance at least 6.

IV. Simulation results using iterative decoding

We employed belief propagation decoding, also known as sum-product decoding, as presented in [7]. This is similar to Gallager's original algorithm, with the difference that log-likelihood metrics are employed in place of probabilities, reducing the influence of numerical problems on the decoding process.

In the simulation results that follow, a number of randomly generated LDPC codes have been used. Where possible we have used codes already published [6], [26]. Where there are none available we have used the best random construction we could generate using the following construction method [2], [27]:

- γ ones are placed in each column of H with an attempt made to make the number of ones in each row approximately the same;
- extra ones are randomly added so that the weight of each row is greater that one;
- if the column weights of H are all even, further ones are added to positions selected randomly from the entire matrix (as it is undesirable for the sum of the rows to be weight one or less); and
- to remove 4-cycles, ones are moved randomly within columns involved in the cycle.

Fig. 2 shows the performance of rate-$1/2$ KTS and randomly generated LDPC codes. For the codeword length $n \approx 500$ we have found a randomly constructed (nearly regular) LDPC code which slightly outperform the regular code constructed from Kirkman triple systems. However for a smaller codeword length $n \approx 100$ the KTS derived system performs better.

Fig. 3 shows the performance of rate $2/3$ KTS, EG and randomly generated LDPC codes. The LDPC code generated from Kirkman triple systems is a $(3, 9)$-regular code, the EG code is (16,16)-regular, and the randomly generated LDPC code has row weights between 7 and 12, and constant column weight 3. While all three codes have similar block lengths and rates the EG code has four times as many non zero entries in its parity-check matrix, resulting in a significant increase in computational complexity for the same number of decoding iterations.

Fig. 4 shows the performance of high rate KTS, STS and randomly generated LDPC codes. The two length $n = 247$ codes have the same rate and density of H, while the two $n = 378$ codes have the same rate and density of H. Using the random construction method we were unable to eliminate 4-cycles from the high rate $n = 247$ code. This is perhaps the primary advantage of our analytically constructed codes over random constructions and the reason we attribute to their improved performance.

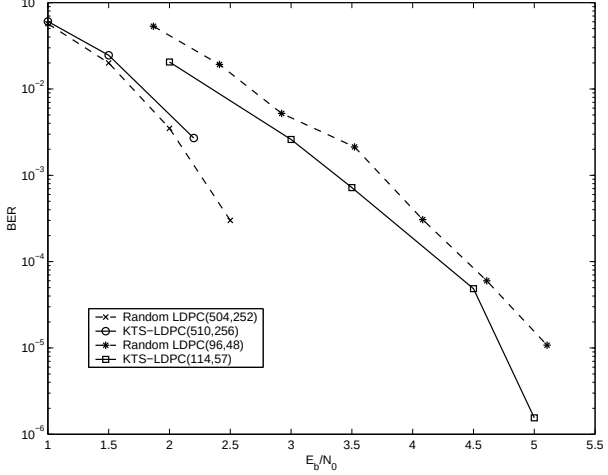

Fig. 2. BER vs. E_b/N_0 for rate $1/2$ LDPC codes, max. iterations = 500

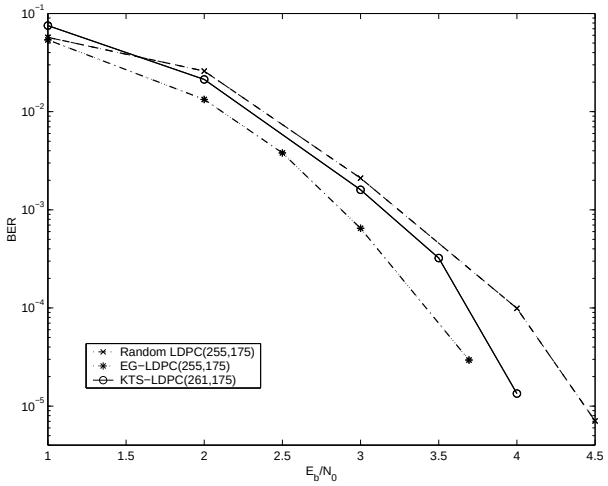

Fig. 3. BER vs. E_b/N_0 for rate $2/3$ LDPC codes, max. iterations = 50

MacKay and Davey rejected Steiner triple systems as LDPC codes due to their poor minimum distance properties [11]. While they do have poor minimum distances our results suggest that their good girth properties for small n compensate for this when belief propagation decoding is used. Even more promising is that the $(3, \rho)$-regular codes derived from Kirkman triple systems do not have the minimum distance constraints of the STS codes and have the additional advantage that they improve upon their good girth properties.

V. Conclusion

We have presented a construction method for LDPC codes that produces parity-check matrices with constant column and row weight, and girth at least 6. These $(3, \rho)$-regular codes can be constructed for any number of parity-check sums $m \equiv 3 \pmod 6$, and for all row weights $\rho \in \{1, 2, \ldots, (m-1)/2\}$. The construction is particularly useful for small block lengths, and for high rate codes for which random construction methods

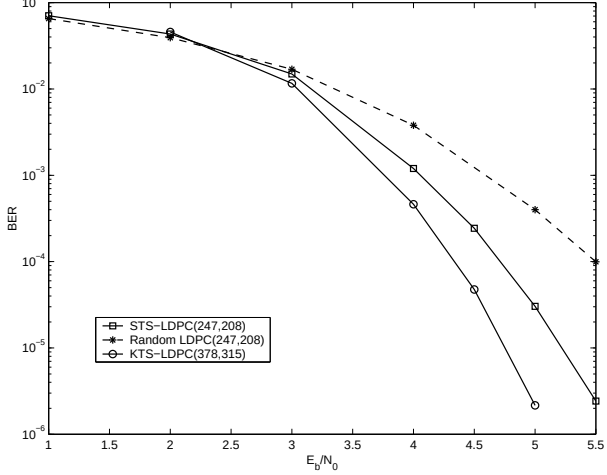

Fig. 4. BER vs. E_b/N_0 for rate 0.84 LDPC codes, max. iterations = 200

have difficulty removing 4-cycles.

APPENDIX

CONSTRUCTING KIRKMAN TRIPLE SYSTEMS [13]

Let $q = 6m + 1$ be a prime power, m any integer, then a Kirkman triple system with $n = 3q$ exists.

Construction: take θ, a primitive element of GF(q), so that $\theta^{6m} = 1$, $\theta^{3m} = -1$, and $\theta^{2m} + 1 = \theta^m$. Then use three copies of each element of GF(q) to construct the sets

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

The sets A, B_{ij}, C_{ij}, $(1 \le i \le m, 1 \le j \le 3)$ form one resolution class, and the translates of this class give a further $6m$ classes. Next, each D_{ij} with its translates give a further resolution class; so we obtain a total of $9m+1$ resolution classes.

Let $q = 6m + 1$ be a prime power, m any integer, then a Kirkman triple system with $n = 2q + 1$ exists.

Construction: Take θ, a primitive element of GF(q), so that $\theta^{6m} = 1$, $\theta^{3m} = -1$, and choose u so that $\theta^m + 1 = 2\theta^u$. Then use two copies of each element of GF(q) and, with an extra element ∞, construct the sets

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

The sets A, B_i, C_i, D_i, E_i, $(0 \le i \le m-1)$ form one resolution class, and the translates of this class give a further $q-1$ classes.

REFERENCES

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

[2] D. J. C. MacKay and R. M. Neal, "Near Shannon limit performance of low density parity check codes", *Electronics Letters*, vol. 32, no. 18, pp. 1645–1646, March 1997.

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

[4] R. G. Gallager, *Low Density Parity Check Codes*, no. 21 in Research Monograph Series. Cambridge, MA: MIT Press, 1963.

[5] T. Richardson and R. Urbanke, "Efficient encoding of low-density parity-check codes", *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 638–656, February 2001.

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

[7] 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, no. 6, pp. 931–937, June 2000.

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

[9] T. Etzion, A. Trachtenberg, and A. Vardy, "Which codes have cycle-free Tanner graphs?", *IEEE Trans. Inform. Theory*, vol. 45, no. 6, pp. 2173–2181, September 1999.

[10] 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, no. 10, pp. 1660–1668, October 2000.

[11] D. J. MacKay and M. C. Davey, "Evaluation of Gallager codes for short block length and high rate applications", *Proceedings of the IMA Workshop on Codes, Systems and Graphical Models*, 1999.

[12] D. K. Ray-Chaudhuri and R. M. Wilson, "Solution of Kirkmans schoolgirl problem", *Proc. Symp. Math.*, vol. 19, pp. 187–203, 1971.

[13] I. Anderson, *Combinatorial Designs: Construction Methods*, Mathematics and its Applications. Ellis Horwood, Chichester, 1990.

[14] E. F. Assmus, Jr., "On 2-ranks of Steiner triple systems", *Electron. J. Combinatorics*, vol. 2, pp. Research paper #R9, 1995.

[15] Y. Kou, S. Lin, and M. Fossorier, "Low density parity check codes: Construction based on finite geometries", in *Proc. IEEE Globecom Conf.*, San Francisco, CA, November 2000, pp. 825–829.

[16] Y. Kou, S. Lin, and M.P.C. Fossorier, "Low density parity check codes based on finite geometries: A rediscovery and more", *submitted to IEEE Trans. Inform. Theory*, 1999.

[17] J. Rosenthal and P. O. Vontobel, "Constructions of LDPC codes using Ramanujan graphs and ideas from Margulis", *Proceedings of 38th Allerton Conference on Communications, Control and Computing.*, October 2000.

[18] 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., pp. 113–130. Springer-Verlag, New York, 2000, available from ⟨http://wol.ra.phy.cam.ac.uk/mackay/CodesRegular.html⟩.

[19] S. B. Wicker, *Error Control Systems for Digital Communication and Storage*, Prentice-Hall, Upper Saddle River, NJ 07458, 1995.

[20] A. E. Brouwer, "Steiner triple systems without forbidden subconfigurations", Tech. Rep. ZW 104/77, Mathematisch Centrum Amsterdam, 1977.

[21] T. S. Griggs, J. Murphy, and J. S. Phelan, "Anti-Pasch Steiner triple systems", *J. Combin. Inform. Systems Sci.*, vol. 15, pp. 79–84, 1990.

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

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

[24] M. J. Grannell, T. S. Griggs, and J. S. Phelan, "A new look at an old construction for Steiner triple systems", *Ars Combinatoria*, vol. 25A, pp. 55–60, 1988.

[25] D. R. Stinson and Y. J. Wei, "Some results on quadrilaterals in Steiner triple systems", *Discrete Math.*, vol. 105, pp. 207–219, 1992.

[26] D. J. MacKay, ", http://wol.ra.phy.cam.ac.uk/mackay/.

[27] R. M. Neal, ", http://www.cs.toronto.edu/ radford/homepage.html.