The 2013 Workshop on Coding and Information Theory
The University of Hong Kong
December 11-13, 2013

Low-Density Parity-Check Codes on Partial Geometries

Shu Lin
(Co-authors: Qiuju Diao, Ying-yu Tai and Khaled Abdel-Ghaffar)

Department of Electrical and Computer Engineering
University of California, Davis
Davis, CA 95616, U.S.A.

Abstract

Many known algebraic constructions of low-density parity-check (LDPC) codes can be placed in a general framework using the notion of partial geometries. Based on this notion, the structure of such LDPC codes can be analyzed using a geometric approach that illuminates important properties of their parity-check matrices. In this approach, trapping sets are represented by sub-geometries of the geometry used to construct the code. Based on the incidence relations between lines and points in this geometry, the structure of trapping sets is investigated. On the other hand, it is shown that removing a sub-geometry corresponding to a trapping set gives a punctured matrix which can be used as a parity-check matrix of an LDPC code. This relates trapping sets, represented by sub-geometries, and punctured matrices, represented by the residual geometries. The null spaces of these punctured matrices are LDPC codes which inherit many of the good structural properties of the original code. Hence, new LDPC codes, with various lengths and rates, can be obtained by puncturing an LDPC code constructed based on a partial geometry. Furthermore, these punctured matrices and codes can be used in a two-phase decoding scheme to correct combinations of random errors and erasures.

I. Introduction

- Partial geometries generalize both Euclidean and projective geometries which were used to construct the first classes of algebraic LDPC codes ever reported in the literature and which were shown to have excellent performance [1]-[2].
- LDPC Codes constructed based on the more general partial geometries were considered in [3]-[8].
- Diverse classes of algebraic LDPC codes that appear in the literature are actually partial geometry codes although their construction methods do not seem to have any geometrical notion.

- Coverage of this presentation:
    1. Partial geometries and their structural properties;
    2. Code construction;
    3. Trapping set structure;
    4. Punctured codes;
    5. Correction of combinations of random errors and erasures

II. Partial Geometries

- Partial geometries were first introduced by Bose in 1963 [9]. An excellent coverage of partial geometries can be found in Batten [10]-[14].
- Consider a system composed of a set N of n points and a set M of m lines where each line is a set of points. If a line L contains a point $\mathbf{p}$, we say that $\mathbf{p}$ is on L and that L passes through $\mathbf{p}$.
- If two points are on a line, then we say that the two points are adjacent and if two lines pass through the same point, then we say that the two lines intersect, otherwise they are parallel.

- The system composed of the sets N and M is a *partial geometry* if the following conditions are satisfied for some fixed integers $\rho \geq 2$, $\gamma \geq 2$, and $\delta \geq 1$ [9], [10]:
    1. Any two points are on at most one line,
    2. Each point is on γ lines,
    3. Each line passes through ρ points,
    4. If a point $\mathbf{p}$ is not on a line L, then there are exactly δ lines, each passing through $\mathbf{p}$ and a point on L.

- Such a partial geometry will be denoted by PaG(γ, ρ, δ), or PaG for short, and γ, ρ, and δ are called the parameters of the partial geometry.
- A simple counting argument shows that the partial geometry PaG(γ, ρ, δ) has exactly

$$n = \rho((\rho - 1)(\gamma - 1) + \delta)/\delta$$

points and

$$m = \gamma((\gamma - 1)(\rho - 1) + \delta)/\delta$$

lines.

- If $\mathbf{p}$ and $\mathbf{p}'$ are two adjacent points, then there are exactly $\gamma\delta + \rho - \gamma - \delta - 1$ points, such that each of these points is adjacent to both $\mathbf{p}$ and $\mathbf{p}'$.
- On the other hand, if $\mathbf{p}$ and $\mathbf{p}'$ are not adjacent, then there are exactly $\gamma\delta$ points, such that each of these points is adjacent to both $\mathbf{p}$ and $\mathbf{p}'$.
- Well known examples of partial geometries are Euclidean and projective geometries over finite fields [12]-[14].

A Net

- If $\delta = \gamma - 1$, the partial geometry PaG$(\gamma, \rho, \gamma - 1)$ is called a *net* [9] which consists of $n = \rho^2$ points and $m = \gamma\rho$ lines.
- Each point $\mathbf{p}$ not on a line L is on a unique line which is parallel to L.
- The set of $m = \gamma\rho$ lines in PaG$(\gamma, \rho, \gamma - 1)$ can be partitioned into γ classes, each consisting of ρ lines, such that all the lines in each class are parallel, any two lines in two different classes intersect, and each of the $n = \rho^2$ points is on a unique line in each class.
- These classes of lines are called *parallel bundles*.
- A two-dimensional Euclidean geometry (or affine geometry) is a net.

Intersecting Bundles

- For every point $\mathbf{p}$ in PaG(γ, ρ, δ) there are exactly γ lines that intersect at $\mathbf{p}$, i.e., all of them pass through $\mathbf{p}$. These lines are said to form an intersecting bundle at $\mathbf{p}$, denoted by $\Delta(\mathbf{p})$.
- Notice that $\mathbf{p}$ is on every line in $\Delta(\mathbf{p})$, there are exactly $\gamma(\rho - 1)$ points, each is on a unique line in $\Delta(\mathbf{p})$, and all the other $n - \gamma(\rho - 1) - 1$ points in PaG(γ, ρ, δ) are not on any line in $\Delta(\mathbf{p})$.
- If $\delta = \rho$, then every point in PaG(γ, ρ, ρ) is adjacent to $\mathbf{p}$ since every point is on a line in $\Delta(\mathbf{p})$. In this case, any two points in PaG(γ, ρ, ρ) are connected by a line.
- Examples for which $\delta = \rho$ are two-dimensional Euclidean and projective geometries.

Subgeometry

- Let Λ be a set of points in PaG(γ, ρ, δ). Then $\Phi(\Lambda) = \cup_{\mathbf{p}\in\Lambda}\Delta(\mathbf{p})$ is the union of intersecting bundles at points in Λ, i.e., $\Phi(\Lambda)$ is the set of lines in PaG(γ, ρ, δ) such that each line passes through at least one point in Λ.
- For a set $\Lambda \subseteq N$ of points and a line $L \in M$ in PaG(γ, ρ, δ), the restriction of L to Λ is $L \cap \Lambda$ which consists of the points in Λ that are on L.
- The subgeometry induced by Λ in PaG(γ, ρ, δ), denoted by PaG$[\Lambda]$, consists of Λ as the set of its points and the restrictions of the lines in $L \in \Phi(\Lambda)$ as its lines.
- Notice that the subgeometry PaG$[\Lambda]$ has $|\Lambda|$ points and $|\Phi(\Lambda)|$ restricted lines.

- Construct an $m \times n$ matrix, $\mathbf{H}_{PaG}$, based on the partial geometry PaG(γ, ρ, δ) as follows. The rows of $\mathbf{H}_{PaG}$ are labeled by the m lines and the columns are labeled by the n points. The entry at the column labeled by a point $\mathbf{p}$ and the row labeled by a line L is 1 if and only if L passes through $\mathbf{p}$.
- In this case, we say that this row in $\mathbf{H}_{PaG}$ labeled by L is *attached* to that column labeled by $\mathbf{p}$. Since there are γ lines pass the point $\mathbf{p}$, there are γ rows attached to the column labeled by $\mathbf{p}$.

- The matrix $\mathbf{H}_{PaG}$ is called the *incidence matrix* of the partial geometry PaG(γ, ρ, δ) and each row is the *incidence vector* of the line labeling that row.
- Since each line consists of ρ points, the incidence vector of a line in PaG(γ, ρ, δ) has weight ρ.
- It follows that the matrix $\mathbf{H}_{PaG}$ has constant column weight γ and constant row weight ρ.
- Since any two distinct points are connected by at most one line, for any two distinct columns there is at most one row that has ones in the two columns, $\mathbf{H}_{PaG}$ is said to satisfy the Row-Column (RC)-constraint.

- If γ is small compared to m, then $\mathbf{H}_{PaG}$ is sparse. In this case, the null space of $\mathbf{H}_{PaG}$ gives an RC-constrained (γ, ρ)-regular LDPC code, $\mathcal{C}_{PaG}$, of length n. The matrix $\mathbf{H}_{PaG}$ is then a parity-check matrix for $\mathcal{C}_{PaG}$ which is called a PaG-LDPC code.
- It was shown in [13] that the rank of $\mathbf{H}_{PaG}$ is upper bounded by

$$rank(\mathbf{H}_{PaG}) \leq \gamma\rho(\gamma - 1)(\rho - 1)/(\rho(\gamma + \rho - \delta - 1)) + 1.$$

- Furthermore, if $\gamma + \rho + \delta$ is even, then

$$rank(\mathbf{H}_{PaG}) \geq \gamma\rho(\gamma - 1)(\rho - 1)/(\delta(\gamma + \rho - \delta - 1)).$$

- The minimum distance, d_{min}, of the PaG-LDPC code $\mathcal{C}_{PaG}$ is lower bounded by

$$d_{min} \geq \max\{\gamma + 1, \gamma(\rho - \gamma + \delta + 1)/\delta, 2(\rho + \delta - 1)/\delta\}$$

.

- The transpose, $\mathbf{H}_{PaG}^{T}$, of the matrix $\mathbf{H}_{PaG}$ is the incidence matrix of a partial geometry PaG(ρ, γ, δ), called the dual of PaG(γ, ρ, δ) obtained by identifying the points of PaG(γ, ρ, δ) with the lines of PaG(ρ, γ, δ) and vice versa. A point $\mathbf{p}$ is on a line L in PaG(ρ, γ, δ) if and only if the line in PaG(γ, ρ, δ) identified with $\mathbf{p}$ passes through the point in PaG(γ, ρ, δ) identified with L.
- The null space of $\mathbf{H}_{PaG}^{T}$ also gives a PaG-LDPC code, denoted by $\mathcal{C}_{PaG,d}$.

IV. The Tanner Graph of a PaG-LDPC Code

- The *Tanner graph*, G_{PaG}, associated with the matrix $\mathbf{H}_{PaG}$ is a bipartite graph composed of two sets of nodes, the set of variable nodes (VNs) labeled by the points in the partial geometry PaG(γ, ρ, δ) or, equivalently, the columns of $\mathbf{H}_{PaG}$, and the set of check nodes (CNs) labeled by the lines in PaG(γ, ρ, δ) or, equivalently, the rows of $\mathbf{H}_{PaG}$. Edges in G_{PaG} connect only VNs to CNs.
- The VN labeled by a point $\mathbf{p}$ is connected to the CN labeled by a line L by an edge if and only if L passes through $\mathbf{p}$, i.e., if and only if the entry in $\mathbf{H}_{PaG}$ at the corresponding row and column is 1. In this case, we say that this VN and this CN are adjacent.

Girth

- Hence, G_{PaG} is a bipartite graph that has n VNs, m CNs, each VN has degree γ, and each CN has degree ρ. Furthermore, any two distinct VNs are connected to at most one CN as any two points in PaG(γ, ρ, δ) are connected by at most one line. This implies that the girth of G_{PaG}, which is the shortest length of a cycle in the bipartite graph, is at least six.
- G_{PaG} contains $n\gamma(\gamma-1)(\rho-1)(\delta-1)/6$ cycles of length 6.
- As each such cycle contains three VNs, each VN is on $\gamma(\gamma-1)(\rho-1)(\delta-1)/2$ cycles of length six.
- Such a large number of short cycles causes correlation in the messages passed during iterative decoding (after three iterations).

Connectivity

- However, the Tanner graph has a *high-degree* of connectivity as each pair of VNs is connected by *a path of length at most four* in G_{PaG}, as any two points in PaG(γ, ρ, δ) are either adjacent or both adjacent to a common point.
- With iterative message-passing decoding, this high-degree connectivity allows *rapid and large amount of information exchanges* between all the VNs which offsets the effect of short cycles.
- This high-degree of connectivity results in *fast decoding convergence*.

- The major disadvantage of this high-degree connectivity is the decoder complexity, in both hardware and computation, and memory required to store messages for information exchanges between processing units.
- This decoder complexity issue can be overcome for PaG-LDPC codes whose parity-check matrices has *block cyclic structure*.

V. SUBGRAPHS AND PUNCTURED CODES

- The Tanner graph G_{PaG} of a PaG-LDPC code is actually a graphical representation of the partial geometry PaG(γ, ρ, δ) with the VNs and CNs representing the points and lines of PaG(γ, ρ, δ) and the edges connecting the VNs to a CN representing the points labeling the VNs lying on the line labeling the CN.
- Let Λ be a set of points in PaG(γ, ρ, δ) and $\Phi(\Lambda)$ be the set of lines in PaG(γ, ρ, δ) such that each line passes through at least one point in Λ.
- The VNs in G_{PaG} labeled by the points in Λ are adjacent to the CNs labeled by the lines in $\Phi(\Lambda)$. Then, the VNs labeled by the points in Λ and the CNs labelled by the lines in $\Phi(\Lambda)$ form a *subgraph* of G_{PaG}, denoted by $G_{PaG}[\Lambda]$.

- This subgraph $G_{PaG}[\Lambda]$ consists of $|\Lambda|$ VNs and $|\Phi(\Lambda)|$ CNs.
- We say that this subgraph $G_{PaG}[\Lambda]$ is induced by the set Λ of VNs in G_{PaG}.
- $G_{PaG}[\Lambda]$ is the graphical representation of the subgeometry PaG[Λ] of the PaG(γ, ρ, δ) induced by Λ.
- The correspondence $G_{PaG}[\Lambda] \leftrightarrow$ PaG[Λ] is one-to-one.

- Let $\mathbf{H}_{PaG}(\Lambda, \Phi(\Lambda))$ be the incidence matrix of the subgraph $G_{PaG}[\Lambda]$ of the Tanner graph G_{PaG} of the partial geometry PaG(γ, ρ, δ) (or the incidence matrix of the subgeometry PaG$[\Lambda]$ of PaG(γ, ρ, δ)).
- $\mathbf{H}_{PaG}(\Lambda, \Phi(\Lambda))$ is a submatrix of the incidence matrix $\mathbf{H}_{PaG}$ (or a punctured matrix of $\mathbf{H}_{PaG}$ obtained by deleting the columns labeled by the points in Λ^c and the rows labeled by the lines in $\Phi(\Lambda)^c$).
- Then the null space of $\mathbf{H}_{PaG}(\Lambda, \Phi(\Lambda))$ also gives a PaG-LDPC code, denoted by $\mathcal{C}_{PaG}(\Lambda, \Phi(\Lambda))$, which may be considered as a punctured code of PaG-LDPC code $\mathcal{C}_{PaG}$.

- Let PaG($\Lambda^c, \Phi(\Lambda)^c$) denote the residue geometry of PaG(γ, ρ, δ) obtained by deleting the points and lines in PaG(Λ) from PaG(γ, ρ, δ).
- Let $\mathbf{H}_{PaG}(\Lambda^c, \Phi(\Lambda)^c)$ denote the incidence matrix of the residue geometry PaG($\Lambda^c, \Phi(\Lambda)^c$).
- The null space of $\mathbf{H}_{PaG}(\Lambda^c, \Phi(\Lambda)^c)$ also gives a PaG-LDPC code.
- Given a partial geometry PaG(γ, ρ, δ), a family of PaG-LDPC codes can be constructed.
- There are many types of partial geometries. From these types of partial geometries, different families of PaG-LDPC codes can be constructed.

- There are many types of partial geometries that appear in textbooks and can be used to construct LDPC codes. Here, we give present five different types, three classical and two new types.
- The two new types were initially developed without any geometric notion.
- The LDPC codes constructed from these five types of partial geometries are mostly quasi-cyclic (QC) or cyclic codes.

Euclidean Geometries

- The s-dimensional Euclidean geometry, EG(s, q), where q is a prime or a power of a prime, consists of q^s points and $q^{s-1}(q^s - 1)/(q - 1)$ lines [10]-[13].
- Each point is represented by an s-tuple over GF(q). The point represented by the all-zero s-tuple is called the origin.
- A line in EG(s, q) contains q points. A line is either a one-dimensional subspace or its coset of the vector space of all the q^s s-tuples over GF(q).
- A point is on $(q^s - 1)/(q - 1)$ lines. Any two distinct points in EG(s, q) are connected by one and only one line.

- Hence, EG(s, q) is a partial geometry with parameters $\gamma = (q^s - 1)/(q - 1)$ and $\rho = \delta = q$, i.e., EG$(s, q) =$ PaG$((q^s - 1)/(q - 1), q, q)$.
- GF(q^s) as an extension field of GF(q) is a realization of EG(s, q) and hence, the points of EG(s, q) can be represented by the q^s elements of GF(q^s).
- Based on EG(s, q), a large class of Euclidean geometry (EG) LDPC codes can be constructed [1]-[5], including cyclic and QC-LDPC codes as subclasses.

Projective Geometries

- The s-dimensional projective geometry, PG(s,q), where q is a prime or a power of a prime, has $n = (q^{s+1} - 1)/(q-1)$ points and $m = (q^s - 1)(q^{s+1} - 1)/(q-1)(q^2 - 1)$ lines. Each line passes through $q+1$ points and each point is on $(q^s - 1)/(q-1)$ lines [12] - [14]. Any two distinct points are on a unique line.
- Hence, PG(s,q) is a partial geometry with parameters $\gamma = (q^s - 1)/(q-1)$ and $\rho = \delta = q+1$.
- Based on lines and points of PG(s,q), families of cyclic and quasi cyclic PG-LDPC codes can be constructed.

Balanced Incomplete Block Designs with $\lambda = 1$

- A balanced incomplete block design (BIBD) consists of a set of n points and distinct subsets, called blocks, each consisting of ρ points, such that each point is in exactly γ blocks and each pair of distinct points is in exactly λ blocks. By viewing the blocks as lines, a BIBD with $\lambda = 1$ is a partial geometry PaG(γ, ρ, ρ).
- Numerous constructions of BIBDs appear in [15] and the references therein.
- Constructions of LDPC codes based on BIBDs with $\lambda = 1$ can be found in [16] - [18]. These codes are called BIBD-LDPC codes and they perform well with iterative decoding.

Partial Geometries from RC-Constrained Matrices

- Let $\mathbf{H}$ be an RC-constrained matrix of size $m \times n$ with row weight ρ and column weight γ, where $n = (\rho - 1)\gamma + 1$.
- Then, it can be shown that $\mathbf{H}$ is the incidence matrix of a partial geometry PaG(γ, ρ, ρ). The partial geometry has n points corresponding to the columns of $\mathbf{H}$ and m lines corresponding to the rows of $\mathbf{H}$.
- The RC-constraint implies that any two points are on at most one line. Furthermore, since each row has weight ρ and each column has weight γ, each line passes through ρ points and each point is on γ lines.

- Next, we will argue that if a point $\mathbf{p}$ is not on a line L then there are exactly ρ lines, each passing through $\mathbf{p}$ and a point on L in case $n = (\rho - 1)\gamma + 1$.
- Since every row has ρ ones, then by adding all the γ rows attached to the column corresponding to the point $\mathbf{p}$, where the sum is over the integers rather than over GF(2), we obtain a vector, $\mathbf{z}$, of length n whose components as integers add up to $\gamma\rho$.
- Notice that the entry in the column corresponding to the point $\mathbf{p}$ in $\mathbf{z}$ is γ. Hence, all other $(\rho - 1)\gamma$ components in $\mathbf{z}$ add up to $(\rho - 1)\gamma$.

- Because of the RC-constraint, all these components are at most equal to 1 and, hence, all of them equal 1. Therefore, every column other than the one corresponding to $\mathbf{p}$ is attached to a unique row corresponding to a line passing through $\mathbf{p}$.
- Since L is a line not passing through the point $\mathbf{p}$ that passes through exactly ρ points, each one of these points is on a line passing through $\mathbf{p}$. This completes the proof that $\mathbf{H}$ is the incidence matrix of a partial geometry PaG(γ, ρ, ρ).
- Notice that the projective geometry, PG(s, q), which is a partial geometry PaG($(q^s - 1)/(q - 1), q + 1, q + 1$), is a special case of this construction.

Partial Geometries from an RC-constrained Arrays of Circulant Permutation Matrices

- Let $\mathbf{H}$ be an $m \times n$ RC-constrained matrix which is a $\gamma \times \rho$ array of $\gamma \times \gamma$ circulant permutation matrices (CPMs), where $m = \gamma^2$ and $n = \gamma\rho$.
- Then $\mathbf{H}$ is the incidence matrix of a partial geometry PaG($\gamma, \rho, \rho - 1$). The partial geometry has $n = \gamma\rho$ points corresponding to the columns of $\mathbf{H}$ and $m = \gamma^2$ lines corresponding to the rows of $\mathbf{H}$. Each point is on γ lines and each line passes through ρ points. The RC-constraint implies that any two points are on at most one line.

- The code constructed based on this partial geometry, i.e., whose parity-check matrix is $\mathbf{H}$, is quasi cyclic.
- There are many constructions of a matrix $\mathbf{H}$ which is an $m \times n$ RC-constrained matrix in the form of a $\gamma \times \rho$ array of $\gamma \times \gamma$ CPMs based on finite fields and Latin squares, see e.g., [19] - [25].
- If $\gamma = \rho$, then the partial geometry PaG$(\gamma, \rho, \rho - 1)$ constructed in this way is actually a net where each parallel bundle of lines in this net corresponds to the rows comprising a row of CPMs in $\mathbf{H}$.

- The above two cases shows that many algebraic constructions of LDPC codes can be unified under the framework of partial geometries.
- Consequently, the structure of these finite field LDPC codes can be studied based on a geometrical approach, especially the trapping set structure and connectivity of the VNs in the Tanner graph of such a code.

- LDPC codes perform well with iterative decoding based on belief propagation, such as the *sum-product algorithm* (SPA) or the *min-sum algorithm* (MSA) [20], [26].
- However, with iterative decoding, most LDPC codes have a common severe weakness, known as the *error-floor*. The error-floor of an LDPC code is characterized by the phenomenon that as the SNR continues to increase, the error probability suddenly drops at a rate much slower than that in the region of low to moderate SNR.
- The error-floor may preclude LDPC codes from applications where very low error rates are required, such as high-speed satellite communications, optical communications, hard-disk drives and flash memories.

- High error-floors most commonly occur for unstructured random or pseudo-random LDPC codes constructed using computer-based methods or algorithms. Structured LDPC codes constructed based on finite geometries, finite field and combinatorial designs [2], [19]-[25], [27], in general, have much lower error-floors.
- Ever since the phenomenon of the error-floors of LDPC codes with iterative decoding became known [28], a great deal of research effort has been expended in finding its causes and methods to resolve or mitigate the error-floor problem [20], [24], [28]-[54].
- For the AWGN channel, the error-floor of an LDPC code is mostly caused by an undesirable structure, known as a *trapping set* [28], in the Tanner graph of the code based on which the decoding is carried out.

- Extensive studies and simulation results show that most trapping sets that cause high error-floors of LDPC codes are the trapping sets of small size.
- In a very recent paper [24], we investigated trapping set structure of RC-constrained regular LDPC codes and showed that, for an RC-constrained (γ, ρ)-regular LDPC code, its Tanner graph contains no trapping set of size at most equal to γ with the number τ of odd-degree CNs smaller than γ.
- The second part of this presentation is on trapping set structure of the PaG-LDPC codes.

Definitions and Basic Concepts

- For the AWGN channel, we adopt from literature definitions of trapping sets and related structures as combinatorial objects that capture the failing mechanisms of iterative decoding algorithms in general and which are independent of the particular decoder used.
- After we briefly review these definitions and concepts of trapping sets of an LDPC code, we give bounds on the sizes of these trapping sets for PaG-LDPC codes.
- First, we define trapping sets and some subclasses of trapping sets and follow this with a motivation of these definitions.

Definition 1. Let G be the Tanner graph of a binary LDPC code, $\mathcal{C}$, of length n given by the null space of an $m \times n$ matrix $\mathbf{H}$ over GF(2). For $1 \leq \kappa \leq n$ and $0 \leq \tau \leq m$, we have the following definitions [28], [29]:

1. A (κ, τ) trapping set is a set, Λ, of κ VNs in G which induces a subgraph, $G[\Lambda]$, of G with exactly τ odd-degree CNs and an arbitrary number of even-degree CNs.
2. A (κ, τ) trapping set is elementary if all the CNs in the induced subgraph $G[\Lambda]$ have degree one or degree two, and there are exactly τ degree-one CNs.

3. A (κ, τ) trapping set is small if $\kappa \leq \sqrt{n}$ and $\tau/\kappa \leq 4$.
4. A $(\kappa, \tau\ )$ trapping set is absorbing if every VN in the trapping set is connected in $G[\Lambda]$ to fewer CNs of odd degree than CNs of even degree. If in addition, every VN not in the trapping set is connected to fewer CNs of odd degree in $G[\Lambda]$ than other CNs, i.e., CNs not in $G[\Lambda]$ or in $G[\Lambda]$ but of even degree, then the trapping set is fully absorbing [48]

- In each decoding iteration, we call a CN a satisfied CN if it satisfies its corresponding check-sum constraint (i.e., its corresponding check-sum is equal to zero), otherwise we call it an unsatisfied CN.
- During the decoding process, the decoder undergoes state transitions from one state to another until all the CNs satisfy their corresponding check-sum constraints or a predetermined maximum number of iterations is reached. The i-th state of an iterative decoder is represented by the hard-decision decoded sequence obtained at the end of the i-th iteration.
- In the process of a decoding iteration, the messages from the satisfied CNs try to reinforce the current decoder state, while the messages from the unsatisfied CNs try to change some of the bit decisions to satisfy their check-sum constraints.

- If errors affect the κ code bits (or the κ VNs) of a $(\kappa, \tau\ )$ trapping set Λ, the τ odd-degree CNs, each connected to an odd number of VNs in Λ, will not be satisfied while all other CNs will be satisfied.
- The decoder will succeed in correcting the errors in Λ if the messages coming from the τ unsatisfied CNs connected to the VNs in Λ are strong enough to overcome the messages coming from the satisfied CNs. However, this may not be the case if τ is small. As a result, the decoder may not converge to a valid codeword even if more decoding iterations are performed and this non-convergence of decoding results in an error-floor.
- In this case, the decoder is said to be trapped.

- For the AWGN channel, error patterns with small number of errors are more probable to occur than error patterns with larger number of errors. Consequently, in message-passing decoding algorithms, the most harmful (κ, τ) trapping sets are usually those with small values of κ and τ.
- Extensive studies and simulation results show that the trapping sets that result in high decoding failure rates and contribute significantly to high error-floors are those with small values κ and small ratios τ/κ.
- These conclusions are captured by the notions of elementary trapping sets and small trapping sets, see Definition 1, parts 2 and 3.

- The notion of absorbing sets is motivated by the fact that for the binary symmetric channel (BSC), if the channel causes errors in the VNs of an absorbing set, then a Gallager type-B decoder (or a one-step majority-logic) decoder will fail.
- With soft-decision iterative decoding, such as the SPA or the MSA, if most of the soft messages become saturated, i.e., their magnitudes are clipped to some finite values to avoid numerical overflow [77] (which is usually true in the error-floor region), then the decoder will behave like a Gallager type-B decoder and will fail.
- Absorbing sets characterize the non-codeword states to which the decoder converges when it fails.

- As all check-sums of a codeword in the code are satisfied, the VNs corresponding to the nonzero bits in a codeword forms a $(\kappa, 0)$ trapping set, where κ is the weight of the codeword. If an error pattern determined by these positions occurs, the decoder converges to an incorrect codeword and commits an undetected error. In this case, the decoder is permanently trapped.
- If there are no harmful trapping sets of sizes smaller than the minimum distance of an LDPC code, then the error-floor of the code decoded with iterative decoding is primarily dominated by the minimum distance.
- An LDPC code with relative large minimum distance whose Tanner graph does not contain harmful trapping set with size smaller than its minimum distance is said to have a *good trapping set structure*.

- Consider the PaG-LDPC code $\mathcal{C}_{PaG}$ constructed based on the partial geometry PaG(γ, ρ, δ).
- A (κ, τ) trapping set in the Tanner graph G_{PaG} of $\mathcal{C}_{PaG}$ is defined by the subgraph $G_{PaG}[\Lambda]$ induced by the VNs labeled by the points in a set Λ of size κ in the partial geometry PaG(γ, ρ, δ) such that $G_{PaG}[\Lambda]$ has exactly τ odd-degree CNs. The CNs adjacent to the κ VNs in the induced subgraph are labeled by the lines in PaG(γ, ρ, δ), each passing through at least one of the κ points labeling the VNs, i.e., the lines in $\Phi(\Lambda)$.

- Recall that the subgraph $G_{PaG}[\Lambda]$ of G_{PaG} is the graphical representation of the subgeometry PaG$[\Lambda]$ of PaG(γ, ρ, δ) induced by Λ which consists of the points in Λ and the restricted lines in $\Phi(\Lambda)$.
- Since the correspondence $G_{PaG}[\Lambda] \leftrightarrow$ PaG$[\Lambda]$ is one-to-one. The subgraph $G_{PaG}[\Lambda]$ has exactly τ CNs of odd degree if and only if there are exactly τ lines in $\Phi(\Lambda)$ that pass through an odd number of points in Λ.
- The above says that a trapping set in the Tanner graph G_{PaG} can be represented by a subgeometry in PaG(γ, ρ, δ).

Enumeration of CNs of Odd-Degree in a Trapping Set

- Based on the geometrical representation of trapping sets given above, we can analyze the trapping set structure of a PaG-LDPC code.
- Let m_i be the number of lines in $\Phi(\Lambda)$, each passing through exactly i points in Λ, where $1 \leq i \leq \kappa$.
- Then, τ is the sum of m_i over all odd integers i such that $1 \leq i \leq \kappa$. Since $2\lfloor(\kappa+1)/2\rfloor - 1$ is the largest odd integer not exceeding κ, we have

$$\tau = m_1 + m_3 + m_5 + \cdots + m_{2\lfloor(\kappa+1)/2\rfloor-1}. \tag{1}$$

- Let the subgeometry PaG$[\Lambda]$ represented by $(\Lambda, \Phi(\Lambda))$
- Let $\mathbf{p}$ be a point in Λ and L be a line in $\Phi(\Lambda)$ passing through $\mathbf{p}$. The pair $(\mathbf{p}, L)$ is called a point-line pair in the subgeometry $(\Lambda, \Phi(\Lambda))$. Such a point-line pair in $(\Lambda, \Phi(\Lambda))$ represents a pair of adjacent VN and CN in a (κ, τ) trapping set.
- There are two ways of counting the total number of such point-line pairs.

- Since each line in $\Phi(\Lambda)$ containing i points in Λ gives i point-line pairs in $(\Lambda, \Phi(\Lambda))$, the total number of point-line pairs in $(\Lambda, \Phi(\Lambda))$ is
$$m_1 + 2m_2 + \cdots + \kappa m_\kappa. \tag{2}$$
- Since each of the κ points in Λ is on γ lines, the total number of such pairs in $(\Lambda, \Phi(\Lambda))$ is also equal to $\kappa\gamma$. Consequently, we have the following equality:
$$m_1 + 2m_2 + \quad + \kappa m_\kappa = \kappa\gamma. \tag{3}$$

- Next, we count, also in two different ways, the number of pairs of adjacent points in Λ. (Throughout this paper, by a pair of points we mean an unordered pair of distinct points.)
- Since Λ consists of κ points, there are at most $\binom{\kappa}{2}$ such pairs.
- Alternatively, since every pair of adjacent points in Λ is on a unique line in $\Phi(\Lambda)$ and a line passing through i points in Λ connects $\binom{i}{2}$ pairs of points, the total number of pairs of adjacent points in Λ is

$$\binom{2}{2}m_2 + \binom{3}{2}m_3 + \ldots + \binom{\kappa}{2}m_\kappa.$$

- Hence, we have the following inequality:

$$\binom{2}{2}m_2 + \binom{3}{2}m_3 + \dots + \binom{\kappa}{2}m_\kappa \leq \binom{\kappa}{2}. \tag{4}$$

- Multiplying both sides in (4) by 2 and subtracting them from the corresponding sides in (3), we have the following inequality:

$$m_1 - \sum_{i=3}^{\kappa} i(i-2)m_i \geq \gamma\kappa - \kappa(\kappa - 1) \tag{5}$$

- From (5) with some algebraic manipulations, we obtain the following lower bound on for the number τ of lines in the subgeometry PaG[Λ]$= (\Lambda, \Phi(\Lambda))$, each containing an odd number of points in Λ:

$$\tau \geq \sum_{i=1,3,5\ldots} m_i = (\gamma+1-\kappa)\kappa + \sum_{i=3,5,\ldots} (i-1)^2 m_i + \sum_{4,6,\ldots} i(i-2)m_i. \tag{6}$$

- Equality in the above lower bound on τ holds if $\delta = \rho$, i.e., every pair of points in the partial geometry PaG(γ, ρ, δ) are adjacent. This is the case for the first 4 types of partial geometries mentioned earlier.

- For this case, if we know the distribution of points in Λ over the lines in $\Phi(\Lambda)$, we can enumerate τ exactly. In fact, we can even determine the configuration the trapping set corresponding to the subgeometry $\text{PaG}[\Lambda] = (\Lambda, \Phi(\Lambda))$. By configuration, we mean the degree distributions of the VNs and CNs of the trapping set.
- Since the two sums in the right side of (6) are non-negative, we have the following lower bound on τ:

$$\tau \geq (\gamma + 1 - \kappa)\kappa. \tag{7}$$

- For $\kappa < \gamma$, τ can be many times larger than κ. It follows from Definition 1-3 that the Tanner graph G_{PaG} of the PaG-LDPC code $\mathcal{C}_{PaG}$ contains no small trapping set with size $\kappa < \gamma - 3$. For $\kappa < \gamma - 3$, τ is at least 5 time larger than κ, i.e., $\tau/\kappa \geq 5$.

- There are two special cases for which the equality of (6) holds.
- The first case is that each line in $\Phi(\Lambda)$ passes through at most two points in Λ is that equality (4) holds and no three points in Λ are collinear.
- In this case, $m_3 = m_4 = ... = m_\kappa = 0$ and the subgeometry PaG(Λ) of PaG(γ, ρ, δ) induced by the set Λ of points represents a $(\kappa, (\gamma + 1 - \kappa)\kappa)$ elementary trapping set with $(\gamma + 1 - \kappa)\kappa$ CNs of degree-1 and $\kappa(\kappa - 1)/2$ CNs of degree-2.
- It can be shown that for $\kappa < \lfloor (2\gamma + 3)/3 \rfloor$, the number of CNs of degree-1 is greater than the number of CNs of degree-2.

- As another special case is that all the points in Λ are collinear. In this case, $m_2 = ... = m_{\kappa-1} = 0$ and $m_\kappa = 1$. Then, the equalities of (4) and (6) hold.
- It follows from (6) that: if κ is even, PaG(Λ) represents a $(\kappa, (\gamma-1)\kappa)$ trapping set with $(\gamma-1)\kappa$ CNs of degree-1 and one CN of degree-κ; and (2) if κ is odd, PaG(Λ) represents a $(\kappa, (\gamma-1)\kappa+1)$ trapping set with $(\gamma-1)\kappa$ CNs of degree-1 and one CN of degree-κ (all CNs have odd degree).

- Based on the intersecting structure of lines in a partial geometry PaG(γ, ρ, δ), it can be easily prove that the Tanner graph G_{PaG} of the PaG-LDPC code $\mathcal{C}_{PaG}$ does not have any absorbing set of size $\kappa \leq \lfloor \gamma/2 \rfloor + 1$.
- The smallest size of an absorbing set is $\lfloor \gamma/2 \rfloor + 2$.

Improved bound on Trapping Sets for Net-LDPC Codes

- Recall that the partial geometry PaG(γ, ρ, δ) is a net if $\delta = \gamma - 1$ in which case the lines can be partitioned into γ parallel bundles, each consisting of ρ parallel lines, and each point is on a unique line in each parallel bundle.
- Examples of nets are two-dimensional Euclidean geometries and partial geometries corresponding certain arrays of CPMs constructed based on finite fields and Latin squares.
- In case of a net, we can improve upon the bound in (6) by considering the distribution of points labeling the VNs in a trapping set over the lines in a parallel bundle.
- Recall that each parallel bundle of ρ lines contains all the points in PaG$(\gamma, \rho, \gamma - 1)$ and, in particular, all the points in Λ.

- Let P be a parallel bundle of lines and $L_1, L_2, ..., L_\rho$ be the lines in P.
- For $1 \leq l \leq \rho$, let Λ_l be the (possibly empty) set of points in Λ that are on the line L_l and let κ_l be the number of such points.
- Since the lines $L_1, L_2, ..., L_\rho$ are parallel, each point in Λ is on one and only one of these lines. Hence, $\Lambda_1, \Lambda_2, ..., \Lambda_\rho$ are disjoint sets whose union is Λ and $\kappa_1 + \kappa_2 + \; + \kappa_\rho = \kappa$.

- Then, the number τ of odd-degree CNs is a (κ, τ) trapping set of a net-LDPC code is lower bounded as below:

$$\tau \geq (\gamma - 1)\kappa - \kappa^2 + \sum_{l=1}^{\rho} \kappa_l^2 + |l : 1 \leq l \leq b, \kappa_l \text{ is odd}|. \qquad (8)$$

- This bound agrees with (7) whenever $\kappa_l \leq 2$ for all $1 \leq l \leq \rho$ and improves upon it in all the other cases.

- The bound on τ given in (8) can be applied easily once the distribution of the set of points Λ corresponding to the VNs of the trapping set over the lines in a parallel bundle is given without the need to explicitly determine $\Phi(\Lambda)$.
- The bound depends on the numbers $\kappa_1, \kappa_2, ..., \kappa_\rho$, which in turn depend on the set of points Λ as well as on the choice of the parallel bundle P.
- For example, if the net is the two-dimensional Euclidean geometry EG$(2, q)$, where q is a prime or a power of a prime, then each point can be represented by a two-tuple (a_0, a_1) over GF(q) and $\{(a_0, a_1) : a_0 \in \text{GF}(q)\}$ for some $a_1 \in$ GF(q) is a line associated with this value of a_1. The q lines associated with the q values of $a_1 \in$ GF(q) form a parallel bundle.

- This parallel bundle can be viewed as the set of the q horizontal lines in a two-dimensional plane where each point in the Euclidean geometry is represented by its cartesian coordinates.
- The number of points in Λ on the line associated with a_1 is the number of points $(a_0, a_1) \in \Lambda$. This gives the numbers $\kappa_1, \kappa_2, ..., \kappa_\rho$ which can be used in (8) to obtain a lower bound on τ in a (κ, τ) trapping set.

REFERENCES

[1] Y. Kou, S. Lin, and M. P. C. Fossorier, "Low density parity check codes based on finite geometries: a rediscovery," *Proc. IEEE 2000 Int. Symp. Inform. Theory (ISIT)*, June 25-30, 2000, Sorrento, Italy, p. 200.

[2] Y. Kou, S. Lin, and M. P. C. Fossorier, "Low-density parity-check codes based on finite geometries: A rediscovery and new results," *IEEE Trans. Inf. Theory*, vol. 47, no. 7, pp. 2711-2736, Nov. 2001.

[3] S. J. Johnson and S. R. Weller, "Codes for iterative decoding from partial geometries," *IEEE Trans. Commun.*, vol. 52, no. 2, pp. 236-243, Feb. 2004.

[4] H. Tang, J. Xu, S. Lin, and K. A. S. Abdel-Ghaffar, "Codes on finite geometries," *IEEE Trans. Inf. Theory*, vol. 51, no. 2, pp. 572-596, Feb. 2005.

[5] J. Xu, L. Chen, I. Djurdjevic, S. Lin, and K. Abdel-Ghaffar, "Construction of regular and irregular LDPC codes: Geometry decomposition and masking," *IEEE Trans. Inf. Theory*, vol. 53, no. 1, pp. 121-134, Jan. 2007.

[6] X. Li, C. Zhang, and J. Shen, "Regular LDPC codes from semipartial geometries," *Acta Appl. Math.*, vol. 102, pp. 25-35, 2008.

[7] J. R. H. Hutton, “LDPC Codes from Semipartial Geometries”. *Ph.D. Dissertation*: University of Sussex, UK, 2011.
[8] P. Vandendriessche, “LDPC codes arising from partial and semipartial geometries,” in *Int. Workshop Coding and Cryptography*, Paris, France, Apr. 11-15, 2011.
[9] R. C. Bose, “Strongly regular graphs, partial geometries and partially balanced designs, *Pacif. J. Math.*, vol. 13, pp. 389-419, 1963.
[10] L. M. Batten, Combinatorics of Finite Geometries, 2nd ed. Cambridge, UK: Cambridge Univ. Press, 1997.
[11] P. J. Cameron and J. H. van Lint, Designs, Graphs, Codes, and their Links. Cambridge, UK: Cambridge Univ. Press, 1991.
[12] H. Mann, Analysis and Design of Experiments. New York, NY: Dover, 1949.

REFERENCES

[13] R. D. Carmichael, Introduction to the Theory of Groups of Finite Orders. New York, NY: Dover, 1956.

[14] A. P. Street and D. J. Street, Combinatorics of Experimental Designs. New York, NY: Oxford, Clarendon Press, 1987.

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

[16] L. Lan, Y. Y. Tai, S. Lin, B. Memari, and B. Honary, "New constructions of quasi-cyclic ldpc codes based on special classes of BIBDs for the awgn and the binary erasure channels," *IEEE Trans. Commun.*, vol. 56, no. 1, pp. 39-48, Jan. 2008.

[17] B. Ammar, B. Honary, Y. Kou, J. Xu, and S. Lin, "Construction of low density parity check codes based on balanced incomplete block designs," *IEEE Trans. Inf. Theory.*, vol. 50, no. 6, pp. 1257-1268, Jun. 2004.

[18] S. J. Johnson and S. R. Weller, "Regular low-density parity-check codes from combinatorial designs," in *Proc. IEEE Inf. Theory Workshop*, Cairns, Australia, Sep. 2001, pp. 90-92.

REFERENCES

[19] L. Lan, L. Zeng, Y. Y. Tai, L. Chen, S. Lin, and K. Abdel-Ghaffar, "Construction of quasi-cyclic LDPC codes for AWGN and binary erasure channels: A finite field approach," *IEEE Trans. Inf. Theory*, vol. 53, no. 7, pp. 2429-2458, Jul. 2007.
[20] W. E. Ryan and S. Lin, Channel Codes: Classical and Modern. New York, NY: Cambridge Univ. Press, 2009.
[21] J. Kang, Q. Huang, L. Zhang, B. Zhou, and S. Lin, "Quasi-cyclic LDPC codes: An algebraic construction," *IEEE Trans. Commun.*, vol. 58, no. 5, pp. 1383-1396, May 2010.
[22] L. Zhang, Q. Huang, S. Lin, and K. Abdel-Ghaffar, "Quasi-cyclic LDPC codes: An algebraic construction, rank analysis, and codes on latin squares," *IEEE Trans. Commun.*, vol. 58, no. 11, pp. 3126-3139, Nov. 2010.
[23] L. Zhang, S. Lin, K. Abdel-Ghaffar, Z. Ding, and B. Zhou, "Quasi-cyclic LDPC codes on cyclic subgroups of finite fields," *IEEE Trans. Commun.*, vol. 59, no. 9, pp. 2330-2336, Sep. 2011.
[24] Q. Huang, Q. Diao, S. Lin, and K. Abdel-Ghaffar, "Cyclic and quasi-cyclic LDPC codes on constrained parity-check matrices and their trapping sets," *IEEE Trans. Inf. Theory*, vol. 58, no. 5, pp. 2648-2671, May 2012.

REFERENCES

[25] Q. Diao, Q. Huang, S. Lin, and K. Abdel-Ghaffar, "A matrix-theoretic approach for analyzing quasi-cyclic low-density parity-check codes," *IEEE Trans. Inf. Theory*, vol. 58, no. 6, pp. 4030-4048, Jun. 2012.

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

[27] N. Kamiya, "High-rate quasi-cyclic low-density parity-check codes derived from finite affine planes," *IEEE Trans. Inf. Theory*, vol. 53, no. 4, pp. 1444-1459, Apr. 2007.

[28] T. Richardson, "Error floors of LDPC codes," in *Proc. 41st Annual Allerton Conf. Commun.*, Control and Comp., Monticello, IL, Oct. 2003, pp. 1426-1435.

[29] S. Laendner and O. Milenkovic, "Algorithmic and combinatorial analysis of trapping sets in structured LDPC codes," in *Proc. Int. Conf. Wireless Networks, Communications and Mobile Computing*, Maui, HI, Jun. 2005, pp. 630-635.

[30] S. Sankaranarayanan, S. K. Chilappagari, R. Radhakrishnan, and B. Vasic, "Failures of the Gallager B decoder: Analysis and applications," in *Proc. Inf. Theory and Appl. Workshop* (ITA), San Diego, CA, Feb. 6-10, 2006.

REFERENCES

[31] S. K. Chilappagari, S. Sankaranarayanan, and B. Vasic, "Error floors of LDPC codes on binary symmetric channel," in *Proc. IEEE Int. Conf. Commun.*, vol. 3, Istanbul, Turkey, Jun. 11-15, 2006, pp. 1089-1094.

[32] M. Stepanov and M. Chertkov, "Instanton analysis of low-density parity-check codes in error-floor regime," in *Proc. IEEE Int. Symp. Inf. Theory*, Seattle, WA, Jul. 9-14, 2006, pp. 552-556.

[33] Z. Zhang, L. Dolecek, B. Nikolic, V. Anantharam, and M. Wainwright, "Investigation of error floors of structured low density parity-check codes by hardware emulation," in *Proc. IEEE Globecom*, San Francisco, CA, Nov. 27-Dec. 1, 2006.

[34] O. Milenkovic, E. Soljanin, and P. Whiting, "Asymptotic spectra of trapping sets in regular and irregular LDPC code ensembles," *IEEE Trans. Inf. Theory*, vol. 53, no. 1, pp. 39-55, Jan. 2007.

[35] S. Laendner and O. Milenkovic, "LDPC codes based on latin squares: Cycle structure, stopping set, and trapping set analysis," *IEEE Trans. Commun.*, vol. 55, no. 2, pp. 303-312, Feb. 2007.

[36] L. Dolecek, Z. Zhang, V. Anantharam, M. Wainwright, and B. Nikolic, "Analysis of absorbing sets for array-based LDPC codes," in *Int. Conf. Commun.*, Glasgow, Scotland, Jun. 24-28, 2007, pp. 6261-6268.

REFERENCES

[37] M. Ivkovic, S. K. Chilappagari, and B. Vasic, "Designing LDPC codes without small trapping sets by using Tanner graph covers," in *Proc. IEEE Int. Symp. Inf. Theory*, Nice, France, Jun. 24-29, 2007, pp. 2266-2270.

[38] S. Abu-Surra, W. E. Ryan, and D. Divsalar, "Ensemble trapping set enumerators for protograph-based LDPC codes," in *Proc., 45th Annual Conf. Commun., Control, and Computing*, Monticello, IL, Sep. 2007.

[39] A. McGregor and O. Milenkovic, "On the hardness of approximating stopping and trapping sets in LDPC codes," in *Proc. IEEE Inf. Theory Workshop*, Lake Tahoe, CA, Sep. 2-6, 2007, pp. 248-253.

[40] S. Abu-Surra, W. E. Ryan, and D. Divsalar, "Asymptotic ensemble enumerators for protograph-based generalized LDPC codes: Computational complexity," in *Proc. Inf. Theory and Appl. Workshop (ITA)*, San Diego, CA, Jan. 27-Feb. 1, 2008, pp. 63-65.

[41] M. Ivkovic, S. K. Chilappagari, and B. Vasic, "Eliminating trapping sets in low-density parity check codes by using Tanner graph covers," *IEEE Trans. Inf. Theory*, vol. 54, no. 8, pp. 3763-3768, Aug. 2008.

[42] Z. Zhang, L. Dolecek, B. Nikolic, V. Anantharam, and M. J. Wainwright, "Lowering LDPC error floors by post processing," in *Proc. IEEE GlobeCom.* Conf., New Orleans, LA, Nov. 30-Dec. 4, 2008.

REFERENCES

[43] M. Hirotomo, Y. Konishi, and M. Morii, "Approximate examination of trapping sets of LDPC codes using the probabilistic algorithm," in *Proc. Int. Symp. Inf. Theory and Its Applications*, Auckland, New Zealand, Dec.7-8, 2008, pp. 1-6.

[44] D. G. M. Michell, A. E. Pusane, and D. J. Costello, Jr., "Asymptotic trapping set analysis of regular protograph-based LDPC convolutional code ensembles," in *Proc. Inf. Theory and Appl. Workshop* (ITA), San Diego, CA, Feb. 8-13, 2009, pp. 264-271.

[45] E. Pusane, D. J. Costello, Jr., and D. G. M. Mitchell, "Trapping set analysis of protograph-based LDPC convolutional codes," in *Proc. IEEE Int. Symp. Inf. Theory*, Seoul, South Korea, Jun. 28-Jul. 3, 2009, pp. 561-565.

[46] E. Cavus, C. L. Haymes, and B. Daneshrad, "Low BER performance estimation of LDPC codes via application of importance sampling to trapping sets," *IEEE Trans. Commun.*, vol. 57, no. 7, pp. 1886-1888, Jul. 2009.

[47] Z. Zhang, L. Dolecek, B. Nikolic, V. Anantharam, and M. J. Wainwright, "Design of LDPC decoders for improved low error rate performance: Quantization and algorithm choices," *IEEE Trans. Commun.*, vol. 57, no. 11, pp. 3258-3268, Nov. 2009.

[48] L. Dolecek, Z. Zhang, V. Anantharam, M. J. Wainwright, and B. Nikolic, "Analysis of absorbing sets and fully absorbing sets of array-based LDPC codes," *IEEE Trans. Inf. Theory*, vol. 56, no. 1, pp. 181-201, Jan. 2010.

REFERENCES

[49] S. Abu-Surra, D. Declercq, D. Divsalar, and W. E. Ryan, "Trapping set enumerators for specific LDPC codes," in *Proc. Inf. Theory and Appl. Workshop (ITA)*, San Diego, CA, Jan. 31-Feb. 5, 2010.
[50] H. Yang and W. E. Ryan, "Low-floor detection/decoding of LDPC-coded partial response channels," *IEEE J. Selec. Areas Commun.*, vol. 28, no. 2, pp. 252-260, Feb. 2010.
[51] S. K. Chilappagari, D. V. Nguyen, B. Vasic, and M. W. Marcellin, "On trapping sets and guaranteed error correction capability of LDPC codes and GLDPC codes," *IEEE Trans. Inf. Theory*, vol. 56, no. 4, pp. 1600-1611, Apr. 2010.
[52] J. Kang, Q. Huang, S. Lin, and K. Abdel-Ghaffar, "An iterative decoding algorithm with backtracking to lower the error floors of LDPC codes," *IEEE Trans. Commun.*, vol. 59, no. 1, pp. 64-73, Jan. 2011.
[53] M. Karimi and A. Banihashemi, "An efficient algorithm for finding dominant trapping sets of irregular LDPC codes," in *Proc. IEEE Int. Symp. Inf. Theory*, Saint Petersburg, Russia, Jul. 31-Aug. 5, 2011.
[54] H. Liu, Y. Li, L. Ma, and J. Chen, "On the smallest absorbing sets of LDPC codes from finite planes, *IEEE Trans. Inf. Theory*, vol. 58, no. 6, pp. 4014-4020, Jun. 2012.

REFERENCES

[55] S. Lin and D. J. Costello, Jr., Error Control Coding: Fundamentals and Applications, 2nd ed. Upper Saddle River, NJ: Prentice Hall, 2004.

[56] T. Kasami, S. Lin, and W. W. Peterson, "Polynomial codes," *IEEE Trans. Inf. Theory*, vol. 14, no. 6, pp. 807-814, Nov. 1968.

[57] S. Lin, "Number of information symbols in polynomial codes, *IEEE Trans. Inf. Theory*, vol. 18, no. 6, pp. 785-794, Nov. 1972.

[58] T. Kasami and S. Lin, "On majority-logic decoding for duals of primitive polynomial codes," *IEEE Trans. Inform. Theory*, vol. 17, no. 3, pp. 322-331, May 1973.

[59] Q. Diao, Y. Y. Tai, S. Lin and K. Ghaffar, "LDPC codes on partial geometries: trapping sets structure, puncturing, and correction of combinations of errors and erasures," submitted to *IEEE Trans. Inf. Theory* 2012 and revised Mar. 2013.