8-2007

PERMUTATION DECODING OF CODES FROM GRAPHS AND DESIGNS

Padmapani Seneviratne
Clemson University, psenevi@clemson.edu

PERMUTATION DECODING OF CODES FROM GRAPHS AND DESIGNS

A Dissertation
Presented to
the Graduate School of
Clemson University

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy
Mathematical Sciences

---

by
Padmapani Seneviratne
August 2007

Accepted by:
Dr. Jennifer D. Key, Committee Chair
Dr. Shuhong Gao
Dr. John Komo
Dr. Gretchen Matthews

ABSTRACT

Permutation decoding is a technique, developed by Jessie McWilliams in 1960's. It involves finding a set of automorphisms of the code, called a PD-set. If such a set exists and if the generator matrix of the code is in standard form then a simple algorithm using this set can be followed to correct the maximum number of errors of which the code is capable. Primarily this method was used originally on cyclic codes and Golay codes.

In this dissertation we study binary codes formed from an adjacency matrix of some classes of graphs and apply the permutation decoding method to these codes. First we do a literature survey on the permutation decoding method and list the known results.

We find a full error correcting PD-set for the binary codes from rectangular lattice graphs and we use partial permutation decoding for the codes from line graphs of multipartite graphs. Next we derive codes from hypercubic graphs and show that these codes are self-dual and find 3-PD sets for these codes.

First-order Reed-Muller codes are the simplest examples of the class of geometrical codes. We use the translation group as a partial permutation decoding set and find 4-PD sets for these codes.

Finally we study the complexity of the permutation decoding algorithm and restate earlier results for the lattice graphs and rectangular lattice graphs.

DEDICATION

Dedicated to my advisor Dr. Jennifer D. Key on her retirement and to my parents.

ACKNOWLEDGMENTS

I appreciate the advice and encouragement given to me by my advisor Dr. Jennifer D. Key. I am indebted to her for the continuous support and patience throughout my graduate career.

I am thankful to Dr. S. Gao, J. Komo and G. Matthews for serving in my dissertation committee and for their useful comments and suggestions. Also I would like to thank the Department of Mathematical Sciences for the continuous support.

Finally I would like to thank my parents for their support and guidance and my wife Upeksha for her love and encouragement.

TABLE OF CONTENTS

Table of Contents (Continued)

CHAPTER 1

INTRODUCTION

The subject of error correcting codes or coding theory began in the late 1940's due to Claude Shannon's paper "A Mathematical Theory of Communication", who showed that good codes exist. Codes were invented to correct errors in noisy communication channels. The early work of Golay, Hamming and Shannon made this a new discipline in electrical engineering as well as in mathematics. Mathematical techniques in algebra and combinatorics have proven useful for error correcting codes. During the development of coding theory, it turned out that several results from design theory and finite geometry could be used to construct 'good' codes. Later, results from coding theory contributed to the development of design theory.

Codes generated by incidence matrices of combinatorial designs have been studied rather extensively [1]; codes generated by the adjacency matrix of a graphs have had less attention. In particular for strongly regular graphs there is a strong analogy with designs and therefore similar results may be expected. In this dissertation we study binary codes from an adjacency matrix of some classes of graphs and apply the permutation decoding method to these codes.

Permutation decoding was first developed by MacWilliams [30] and involves finding a set of automorphisms of a code called a PD-set. The method is described fully in MacWilliams and Sloane [31, Chapter 16, p. 513] and Huffman [13, Section 8]. It is most useful when the code has a fairly large group of automorphisms. Codes from strongly regular graphs (including triangular graphs), lattice graphs and graphs from triples, were good candidates for permutation decoding as shown in [19], [21], [18].

The main purpose of this dissertation is to apply the permutation decoding method to codes from other types of graphs and certain combinatorial structures.

This dissertation is organized as follows: In this chapter we introduce the notation and background in Chapter 2. We provide the basic concepts from designs, graphs and codes necessary for later chapters. Also we introduce the permutation decoding method and describe the decoding algorithm.

In Chapter 3, we give a comprehensive survey of results in permutation decoding. We list all the known results of this method. MacWilliams paper "Permutation decoding of systematic codes" [30] in 1964, was the first article on permutation decoding. After some sporadic results, Gordon [10] and Wolfmann [35] used this method to find minimal permutation sets for decoding the Golay code. In 1978 Goodman and Green [11] proposed the implementation of both hard and soft-decision permutation decoding on an Intel 8080 microprocessor. Results were added by Key, et al. and they showed codes from graphs and designs were good candidates for permutation decoding.

We define binary codes from rectangular lattice graphs in Chapter 4. The codes are formed by the row space over $\mathbb{F}_2$ of an adjacency matrix for the rectangular lattice graph $L_2(m, n)$. When $m = n$, these are the square lattice graphs and had been studied in [21]. We use a point ordering of the vertices to find an information set for the codes and show that $S_m \times S_n$ acts as a PD set for full error correction.

The square lattice graph $L_2(n)$ is the line graph of the complete bi-partite graph. We generalize this concept to multi-partite graphs in Chapter 5. In Section 5.2 we define the binary codes from the line graph of a multipartite graph and find the parameters of the code. The code parameters depend on the size of each partite set n and the number of partite sets m. For some values of m and n we find explicit PD sets for the full error correction capability of the code.

In Chapter 6, we construct binary self-dual codes from hypercubes. The hypercube Q_n or the n-cube is the graph with vertices the 2^n vectors of $\mathbb{F}_2^n$ and two vertices adjacent if their coordinates differ in precisely one place. General properties of the graph

Q_n, the symmetric design obtained from it and its binary code are discussed in section 6.2. Then we use the notion of partial permutation decoding to find 3-PD sets for these codes.

Reed-Muller codes are examples of the class of geometrical codes which also include affine and projective geometry codes. Key, McDonough and Mavron [16] constructed information sets for the generalized Reed-Muller codes. We use these information sets in Chapter 7 to construct s-PD sets for partial permutation decoding of the first order Reed-Muller codes. We show that the translation group will provide 4-PD sets for these codes.

The worst case time complexity of the permutation decoding algorithm can be expressed in terms of the length n and dimension k of the code C and the size m of the PD set and is of order $\mathcal{O}(knm)$. By arranging the PD-set elements in a certain manner we could reduce the complexity of the algorithm. In Chapter 8, we consider codes from lattice graphs and rectangular lattice graphs and use nested PD-sets to reduce the complexity.

We summarize the results obtained from this study in Chapter 9 and propose further study.

CHAPTER 2

BACKGROUND AND TERMINOLOGY

This chapter introduces the notation and terminology used in this dissertation. We provide a brief background on designs, graphs and codes. Most of the details can be found in [1]. General notation for graph theory is standard and can be found in [36].

2.1 Designs

An **incidence structure** $\mathcal{S} = (\mathcal{P}, \mathcal{B}, \mathcal{I})$ consists of two disjoint sets, $\mathcal{P}$ and $\mathcal{B}$, and a subset $\mathcal{I}$ of $\mathcal{P} \times \mathcal{B}$. The members of $\mathcal{P}$ are called **points** and the members of $\mathcal{B}$ are called **blocks**. For $p \in \mathcal{P}, B \in \mathcal{B}$ if the ordered pair (p, B) is in $\mathcal{I}$ we say that p is **incident** with B, or that B contains the point p, or that p is on B.

Definition 2.1. *Let $\mathcal{S} = (\mathcal{P}, \mathcal{B}, \mathcal{I})$ with $|\mathcal{P}| = v$ and $|\mathcal{B}| = b$. Let the points be labeled $\{p_1, p_2, \ldots, p_v\}$ and let the blocks be labeled $\{B_1, B_2, \ldots, B_b\}$. An* incidence matrix *for $\mathcal{S}$ is a $b \times v$ matrix $A = (a_{ij})$ of 0's and 1's such that*

$$a_{ij} = \begin{cases} 1 & if \quad (p_j, B_i) \in \mathcal{I} \\ 0 & if \quad (p_j, B_i) \notin \mathcal{I}. \end{cases}$$

Definition 2.2. *An incidence structure $\mathcal{D} = (\mathcal{P}, \mathcal{B}, \mathcal{I})$ is a t-(v, k, λ) design, i.e. a t-design, where t,v,k and λ are non-negative integers, if*

1. *$|\mathcal{P}| = v$;*
2. *every block $B \in \mathcal{B}$ is incident with precisely k points; and*
3. *every t distinct points are together incident with precisely λ blocks.*

A 2-(v, k, λ) design is called a **symmetric design** if the number of blocks is the same as the number of points. If $k = 2$, a t-(v, k, λ) design is a **graph**, and points are called vertices and blocks are called edges.

Definition 2.3. *Let $\mathcal{S} = (\mathcal{P}, \mathcal{B}, \mathcal{I})$ and $\mathcal{T} = (\mathcal{Q}, \mathcal{C}, \mathcal{J})$ be incidence structures, and let ϕ be a bijection from $\mathcal{P} \cup \mathcal{B}$ to $\mathcal{Q} \cup \mathcal{C}$. If $\phi(\mathcal{P}) = \mathcal{Q}$ and $\phi(\mathcal{B}) = \mathcal{C}$ with $p \in P$ incident with $B \in \mathcal{B}$ if and only if $\phi(p) \in \mathcal{Q}$ is incident with $\phi(\mathcal{B}) \in C$, then ϕ is an isomorphism from $\mathcal{S}$ to $\mathcal{T}$ and we write $\mathcal{S} \approx \mathcal{T}$. If $\mathcal{S} = \mathcal{T}$, then ϕ is an automorphism. The set of all automorphisms forms the automorphism group of $\mathcal{S}$ and will be denoted by $Aut(\mathcal{S})$.*

2.2 Graphs

A graph, $\Gamma = (V, E)$ consists of a finite, nonempty set of vertices V together with a set E of edges, where an edge is subset of V, of cardinality 2. Equivalently a graph consists of a finite set of vertices V and a set of edges E, with an incidence relation between vertices and edges, having the property that any edge is incident with exactly two vertices, and any two vertices are incident with atmost one edge. Our graphs will be undirected and without loops.

If x is a vertex of a graph Γ, the **valency** of x is the number of edges containing x. If all vertices have the same valency, the graph is called **regular**, and the common valency is the valency of the graph. Two graphs are **isomorphic** if there exists a bijection between their point sets that preserves adjacency. An **automorphism** of a graph is any permutation of the vertices preserving adjacency. The set of all automorphisms forms the automorphism group of the graph. An **independent** set in a graph is a set of pairwise nonadjacent vertices.

A **complete graph** is a graph all of whose vertices are pairwise adjacent. The complete graph with n vertices is denoted by K_n. A **complete bipartite graph** is a bipartite graph such that two vertices are adjacent if and only if they are in different partite sets. When the sets have sizes r and s the complete bipartite graph is denoted by $K_{r,s}$. The **line graph** of a graph Γ is the graph having as vertices the edges of Γ, two edges being adjacent if they have a common vertex.

Definition 2.4. *A graph Γ on v vertices is said to be* **strongly regular** *with parameters (v, k, λ, μ), if*

1. *Γ is regular of degree k;*
2. *any two adjacent vertices are simultaneously adjacent to λ other vertices;*
3. *any two non-adjacent vertices are simultaneously adjacent to μ other vertices.*

Example 2.5. *The line graph of $K_{n,n}$ is the* **lattice graph** *$L_2(n)$. Then the lattice graph $L_2(n)$ $(n \geq 2)$ has as vertices the ordered pairs (i, j), $1 \leq i, j \leq n$, where two pairs are adjacent if they have a common coordinate. $L_2(n)$ is strongly regular of type $(n^2,\ 2(n-1),\ n-2,\ 2)$.*

Example 2.6. *The* **triangular graph** *$T(n)$ $(n \geq 3)$ is the line graph of K_n. Then the triangular graph $T(n)$ has as vertices the 2-subsets of a given n-set, say $\{1, 2, \ldots, n\}$, where two vertices are adjacent if the corresponding 2-subsets have a common element. $T(n)$ is strongly regular of type $(\frac{n(n-1)}{2}, 2(n-2), n-2, 4)$.*

Let Γ be the graph with vertex set $V = \{v_1, \ldots, v_n\}$. The **adjacency matrix** of Γ, is the $n \times n$ matrix $A = [a_{i,j}]$, in which entry $a_{i,j}$ is the number of edges in Γ with endpoints $\{v_i, v_j\}$. The p-rank of the adjacency matrix A, denoted by $rank_p(A)$, is the dimension of the row space of A over the finite field $\mathbb{F}_p$.

2.3 Groups

Let G be the permutation group on a set X. Then we define the **orbit** of an element x in X to be the set $\{xg | g \in G\}$, where xg denotes the image of x under g. The group G is said to be **transitive** if it only has one orbit, i.e. for every $x, y \in X$, there exists an element (permutation) in G that maps x into y. More generally, the group G is k-transitive if, for every pair of k-tuples of distinct elements in X, say $(x_1, \ldots, x_k)$ and $(y_1, \ldots, y_k)$, there is an element of G that maps x_i to y_i for all $1 \leq i \leq k$. If G is transitive then a block of G is a non-empty subset $Y \subseteq X$ such that, for any $g \in G$, either $Yg = Y$

or $Yg \cap Y = \emptyset$. The group G is said to be **primitive** if the only blocks in X are X itself and the singleton subsets of X; otherwise, it is said to be **imprimitive**.

Definition 2.7. *Let H and K be groups and suppose that we have an action of H on K of which respects the group structure on K, i.e for each $x \in H$ the mapping $u \mapsto u^x$, $u \in K$, is an automorphism of K. Let*

$$G := \{(u, x) | u \in K, x \in H\}$$

and define

$$(u, x)(v, y) = (uv^{x^{-1}}, xy) \; for \; each \; (u, x), (v, y) \in G$$

We call G the **semi-direct product** *of K by H and write*

$$G := K \rtimes H$$

Further $|G| = |H||K|$.

If Γ and Δ are non empty sets, let $Fun(\Gamma, \Delta)$ denote the set of all functions from Γ into Δ. If K is a group, then $Fun(\Gamma, K)$ is a group through the binary operation defined by

$$(fg)(\gamma) = f(\gamma)g(\gamma) \; for \; all \; f, g \in Fun(\Gamma, K).$$

In the case that Γ is finite of size m, say $\Gamma = \{\gamma_1, \gamma_2, \ldots, \gamma_m\}$, then the group $Fun(\Gamma, K)$ is isomorphic to K^m.

Definition 2.8. *Let K and H be groups and suppose H acts on the non-empty set Γ. Then the* **wreath product** *of K by H with respect to this action is defined to be the semi-direct product $Fun(\Gamma, K) \rtimes H$, where H acts on the group $Fun(\Gamma, K)$ via*

$$f^x(\gamma) := f(\gamma^{x^{-1}}) \; for \; all f \in Fun(\Gamma, K), \gamma \in \Gamma \; and \; x \in H.$$

We denote this group by $K \wr H$.

2.4 Error correcting codes

The codes that we will associate with designs are linear codes. The alphabet will be a finite field $F = \mathbb{F}_q$ of order q and the codewords will be vectors in $V = F^n = F \times F \times \ldots \times F$. The background on codes can be found in MacWilliams and Sloane [31], Hill [9], Van Lint [28], Assmus and Key [1].

Definition 2.9. *A* **linear code** *C over $F = \mathbb{F}_q$ of length n is a subspace of $V = F^n$.*

Definition 2.10. *Let $v = (v_1, v_2, \ldots, v_n)$ and $w = (w_1, w_2, \ldots, w_n)$ be two vectors in F^n. The Hamming distance $d(v, w)$, between v and w is the number of coordinate places in which they differ:*

$$d(v, w) = |\{i \mid v_i \neq w_i\}|.$$

Definition 2.11. *The minimum distance $d(C)$ of a code C is the smallest of the distances between distinct codewords:*

$$d(C) = min\{d(v, w) | v, w \in C, v \neq w\}.$$

If $dim(C) = k$ and $d(C) = d$, then we write $[n, k, d]_q$ to denote the parameters of the q-ary code C.

Let C be a code of minimum distance d. If $d \geq s + 1 > 1$, then C can be used to detect up to s errors in any codeword. If $d \geq 2t + 1$, then C can be used to correct up to t errors in any codeword.

Definition 2.12. *Let $V = F^n$. For any vector $v = (v_1, v_2, \ldots, v_n) \in V$, set $S = \{i \mid v_i \neq 0\}$. Then S is called the support of v and the weight of v is $|S|$. The minimum weight of a code C is the minimum of the weights of the non-zero codewords.*

Let C be an $[n, k, d]$ code. Then the minimum distance $d = d(C)$ is the minimum weight of C.

Definition 2.13. *Two linear codes in F^n are isomorphic if and only if each can be obtained from the other by permuting the coordinate positions of F^n.*

To define the orthogonal code we need an inner product defined on the vector space. We take the standard inner product: for $v, w \in F^n$, $v = (v_1, v_2, \ldots, v_n), w = (w_1, w_2, \ldots, w_n)$, we write the **inner product** of v and w as (v, w) where

$$(v, w) = \sum_{i=1}^{n} v_i w_i.$$

Definition 2.14. *Let C be a q-ary $[n, k, d]$ code. The* **orthogonal code** *is denoted by $C^\perp$ and is given by*

$$C^\perp = \{v \in F^n | (v, c) = 0 \; for \; all \; c \in C\}.$$

We call C **self-orthogonal** if $C \subseteq C^\perp$ and **self dual** if $C = C^\perp$.

A **generator matrix** for the code is a $k \times n$ matrix made up of a basis for C. A **check matrix** H or a **parity-check matrix** for C is a generator matrix for $C^\perp$; the **syndrome** of a vector $y \in F^n$ is Hy^T, where H is a check matrix for C. Any code is isomorphic to a code with generator matrix in so-called **standard form**, i.e. the form $[I_k|A]$ where A is $k \times n - k$; a check matrix is then given by $[-A^T|I_{n-k}]$. An **automorphism** of a code C is any permutation of the coordinate positions that maps codewords to codewords. The set of all the automorphisms form the automorphism group of the code, denoted by $Aut(C)$.

Definition 2.15. *For any field F and any set Ω, denote by F^Ω the vector space over F of functions from Ω to F, with pointwise addition and scalar multiplication. For any subset Y of Ω, the characteristic function on Y is the function (vector) $v^Y \in F^\Omega$ defined by*

$$v^Y(w) = \begin{cases} 1 & if \quad w \in Y \\ 0 & if \quad w \notin Y \end{cases}$$

The standard basis for F^{Ω} is $\{v^{\{w\}}|w \in \Omega\}$, and we write v^w instead of $v^{\{w\}}$.

Definition 2.16. *The code of $\mathcal{S} = (\mathcal{P}, \mathcal{B}, \mathcal{I})$ over the field F is the subspace $C_F(\mathcal{S})$ of $F^{\mathcal{P}}$ spanned by the vectors corresponding to the characteristic functions of the blocks of $\mathcal{S}$. Thus*

$$C_F(\mathcal{S}) =< v^B|B \in \mathcal{B} > .$$

It is clear that $Aut(\mathcal{D}) \leq Aut(C_F(\mathcal{S}))$.

The binary code from a graph, is the code formed from the span of the adjacency matrix of that graph. This code is also the code of the design obtained by taking the rows of the adjacency matrix as the incidence vectors of the blocks.

2.5 Permutation decoding

In this section we will discuss the method of permutation decoding. Permutation decoding can be used when a code has sufficiently many automorphisms to ensure the existence of a set of automorphisms that satisfies certain conditions. The method is described fully in MacWilliams and Sloane [31] and Huffman [13].

Definition 2.17. *A PD-set for a code C is a set $\mathcal{S}$ of automorphisms of C which is such that, if C can correct t errors, then every possible error vector of weight t or less can be moved by some member of $\mathcal{S}$ out of the information positions. That is if $\{k+1, \ldots, n\}$ are the check positions, then every t-tuple from $\{1, \ldots, n\}$ can be moved by some automorphism of C into $\{k+1, \ldots, n\}$.*

Such a set will fully use the error-correction potential of the code. The property of having a PD-set for a code may not be invariant under isomorphisms of codes. It depends on the choice of the information set that we select. Sometimes it is observed that permutation decoding cannot be used to correct the full error capacity of the code [14]. The notion of partial permutation decoding was introduced to correct smaller number of errors.

Definition 2.18. *If C is a t-error-correcting code with information set $\mathcal{I}$ and a check set $\mathcal{C}$, then, for $s \leq t$, an* **s-PD-set** *is a set $\mathcal{S}$ of automorphisms of C which is such that every s-set of coordinate positions is moved by atleast one member of $\mathcal{S}$ into the check positions.*

The following result is stated and proved in Huffman [13].

Result 2.19. *Let C be an $[n, k, d]_q$ t-error correcting code. Suppose H is a check matrix for C in standard form, i.e such that I_{n-k} is in the redundancy positions. Let $y = c + e$ be a vector, where $c \in C$ and e has weight $\leq t$. Then the information symbols in y are correct if and only if the weight of the syndrome Hy^t of y is $\leq t$.*

Proof: Suppose C has generator matrix G in standard form, i.e. $G = [I_k|A]$ and that the encoding is done using G, i.e. the data set $x = (x_1, \ldots, x_k)$ is encoded as xG. The information symbols are then the first k symbols, and the check matrix H is $H = [-A^T|I_{n-k}]$. Suppose the information symbols of y are correct. Then $Hy^T = He^T = e^T$, and thus $wt(Hy^T) \leq t$.

Conversely, suppose that not all the information symbols are correct. Then if $e = e_1 \ldots e_n$, let $e^{'} = e_1 \ldots e_k$, $e^{''} = e_{k+1} \ldots e_n$, and we assume that $e^{'}$ is not the zero vector. Now use the fact that for any two vectors $wt(x+y) \geq wt(x) - wt(y)$. Then $wt(Hy^T) = wt(He^T) = wt(-A^T e^{'T} + e^{''T}) \geq wt(-A^T e^{'T}) - wt(e^{''T}) = wt(e^{'}A) - wt(e^{''}) = wt(e^{'}A) + wt(e^{'}) - wt(e^{'}) - wt(e^{''}) = wt(e^{'}G) - wt(e) \geq d - t \geq t + 1$. ■

Algorithm for permutation decoding

Let C be a t-error correcting $[n, k, d]_q$ code with a check matrix H in standard form and let

$$\mathcal{S} = \{\sigma_1, \sigma_2, \ldots, \sigma_s\}$$

be a PD-set for C. Suppose a codeword x is sent and y is received and at most t errors occur. Then we can state the permutation decoding algorithm as follows:

1. Compute the syndromes $H(y\sigma_i)^T$ for $i \in \{1, 2, \ldots, s\}$ until an i is found such that $wt(H(y\sigma_i)^T) \leq t$;
2. Look at the information positions in $y\sigma_i$ and form the codeword c that has these information symbols;
3. Decode y as $c\sigma_i^{-1}$.

This algorithm is valid since for any $\sigma \in S_n$ if $y = x + e$ where $x \in C$, then $y\sigma = x\sigma + e\sigma$, and if $\sigma \in Aut(C)$ then $x\sigma \in C$.

We use the following observation which we state as a more general lemma:

Result 2.20. *Suppose C is a $[n, k, d]_q$ t-error-correcting code, and let $r = n - k$. Let $\mathcal{T}$ denote the set of t-tuples of the elements of $\{1, \ldots, n\}$ and $\mathcal{E}$ the set of t-tuples of the elements of the check positions $\{k+1, \ldots, n\}$. Then a set $\mathcal{S} = \{g_1, \ldots, g_s\}$ of automorphisms will be a PD-set for C if*

$$\bigcup_{g \in \mathcal{S}} \mathcal{E}^{g^{-1}} = \mathcal{T}.$$

Furthermore, for any $g \in Aut(C)$, the set $g\mathcal{S} = \{gg_1, \ldots, gg_s\}$ will also be a PD-set.

Proof: The first part is clear. The second statement can be proved as follows: we need to show that any t-tuple $\beta \in \mathcal{T}$ satisfies $\beta = \alpha^{e^{-1}}$ for some $\alpha \in \mathcal{E}$ and $e \in g\mathcal{S}$. If $\beta^g = \gamma = \alpha^{h^{-1}}$ for some $\alpha \in \mathcal{E}$ and $h \in \mathcal{S}$, then $\beta = \alpha^{h^{-1}g^{-1}} = \alpha^{(gh)^{-1}}$, as required. ■

Furthermore, there is a bound on the minimum size that the set $\mathcal{S}$ may have, due to Gordon [10] and Schönheim [32] and quoted and proved (using counting arguments) in [13]:

Result 2.21. *If $\mathcal{S}$ is a PD-set for a t-error-correcting $[n, k, d]_q$ code C, and $r = n - k$, then*

$$|\mathcal{S}| \geq \left\lceil \frac{n}{r} \left\lceil \frac{n-1}{r-1} \left\lceil \cdots \left\lceil \frac{n-t+1}{r-t+1} \right\rceil \cdots \right\rceil \right\rceil \right\rceil.$$

CHAPTER 3

A SURVEY OF THE PERMUTATION DECODING METHOD

In this chapter we will do a complete literature survey on permutation decoding method.

Permutation decoding of systematic codes

Jessie MacWilliams in her paper [30] developed permutation decoding method and applied permutation decoding to cyclic codes. An $[n, k, d]_q$ code C is cyclic provided that for all codewords c in C, the vector obtained from c by the cyclic shift of coordinates is also in C. Any k consecutive coordinates form an information set. To construct PD-sets for the code C, we look for a set of automorphisms of C in which at least one map will send each error vector of weight $\leq t$ to a vector with atleast k consecutive zeros (a vector with gap size atleast k).

Consider the cyclic permutations.

$$\begin{aligned} \sigma &: i \mapsto i+1 \pmod n, \\ \sigma^2 &: i \mapsto i+2 \pmod n, \\ &\vdots \\ \sigma^n &: i \mapsto i+n \pmod n. \end{aligned}$$

The permutation $\sigma = (1, 2, \ldots, n)$ is in Aut(C). So if there is a gap of length $\geq k$ in an error vector, the successive cyclic shifts will move non-zero entries in the error vector out of the first k positions. In particular, the set of maps

$$1, \sigma, \sigma^2, \ldots, \sigma^{n-1}$$

will always correct all single error vector. This set cannot correct an error vector in which there is no gap of length $\geq k$. In general, if $k < \frac{n}{t}$, then any error vector of weight t or less will automatically have a gap of size k or more.

Result 3.1. *Let C be a t-error correcting $[n, k, d]_q$ code. If $k < \frac{n}{t}$, then $\{\sigma^i | 0 \leq i \leq n-1\}$ is a PD-set for C, where $\sigma = (1, 2, \ldots, n)$.*

Minimal permutation sets for decoding binary Golay codes

The automorphism groups of the binary Golay codes $\mathcal{G}_{23}$ and $\mathcal{G}_{24}$ are $\mathcal{M}_{23}$ and $\mathcal{M}_{24}$ respectively. The Gordon bound for $\mathcal{G}_{23}$ is 15 and for $\mathcal{G}_{24}$ is 14. By using a computer program Gordon computationally obtained minimal PD-sets of size 15 and 14 respectively that satisfy his bound.

This was done by collecting a generating set of permutations from each automorphism group, which moved as few elements as possible between the check and information symbols. The computer program had taken each r-tuple and gone through the permutations, applying the ones which took the most digits of the r-tuple into the check symbols. This had been continued until all digits were in the check places. The permutations used were composed forming the permutation corresponding to the r-tuple.

Wolfmann [35] found a minimal set consisting of 14 permutations to decode the $[24, 12, 8]$ Golay code using permutation decoding.

Result 3.2. *The $[24, 12, 8]$ Golay code is equivalent to the binary linear code with generator matrix: $G = [I_{12} : M]$, where I_{12} is the identity matrix of order 12 and*

$$\begin{pmatrix} I_3 & A & A^2 & A^4 \\ A & I_3 & A^4 & A^2 \\ A^2 & A^4 & I_3 & A \\ A^4 & A^2 & A & I_3 \end{pmatrix}$$

where I_3 is the identity matrix of order 3 and

$$A = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 0 & 0 \cdot \\ 1 & 0 & 1 \end{pmatrix}$$

Result 3.3. *If* $s = (4,7,16,10,22,19,13)(5,8,17,11,23,20,14)(6,9,18,12,24,21,15)$ *and* $t = (1,13)(2,14)(3,15)(4,16)(5,17)(6,18)(7,19)(8,20)(9,21)(10,22)(11,23)(12,24)$ *in* S_{24}, *then* $\mathcal{S} = \{t^i s^j \mid 0 \leq i \leq 1,\ 0 \leq j \leq 6\}$ *is a PD-set of size* 14 *for* $\mathcal{G}_{24}$.

Binary cyclic double-error correcting codes of certain length

Shiva, Fung and Tan in their paper [33] gave an analysis of acyclic codes of specified lengths from the the point of view of permutation decoding. They give a different interpretation of the permutation decoding method for binary cyclic codes using generator polynomials. Suppose C is an $(n,k,2)$ binary cyclic code generated by

$$g(X) = \frac{1+X^n}{h(X)}$$

where n is odd and $h(X)$ has degree k. Let

$$R(X) = V(X) + E(X)$$

where $V(X)$ belongs to C and $E(X)$ has weight 2 or less. Suppose also that

$$R_{i\beta}X = (X^\beta R^{2^i}(X) \ \ modulo \ \ 1+X^n) \quad modulo\, g(X)$$

where $\beta = 0,1,2,\ldots,n-1$ and $i = 0,1,2,\ldots,q$, q is such that $nc = 2^q - 1$.
It is known that under approriate restrictions on the rate k/n, the consideration of $R_{0\beta}(X), R_{1\beta}(X)\ldots$ will yield $E(X)$. Use R_i to denots $R_{i\beta}(X)$. Since every $(n,k,2)$ code with $\frac{k}{n} < \frac{1}{2}$ can be decoded with just R_0 the paper discusses the case when $\frac{k}{n} \geq \frac{1}{2}$ $[17,k,2]$ codes:

1. If $k \leq 11$, then R_0 and R_1 will give $E(X)$.

2. If $k \leq 13$, then R_0, R_1 and R_2 will suffice.

3. If $k > 13$, permutation decoding is not possible.

4. The relevant BCH codes are [17, 9, 2] and can be decoded with R_0 and R_1.

$[21, k, 2]$ codes:

1. If $k \leq 13$ R_0 and R_1 will suffice.

2. If $k > 13$, permutation decoding is not possible.

3. Relevant BCH codes are $[21, 12, 2]$ and can be decoded with R_0 and R_1.

$[23, k, 2]$ codes:

1. When $k \leq 21$ the authors give relevant R_i's .

2. When $k > 21$ permutation decoding is not possible.

$[31, k, 2]$ codes:

1. If $k \leq 25$ the R_i's are listed.

2. permutation decoding is not possible otherwise.

$[45, k, 2]$ codes:

1. For $k \leq 29,\ R_0$ and R_1 will suffice.

2. For $k > 29$ permutation decoding is not applicable.

$[47, k, 2]$ codes:

1. If $k \leq 46,\ R_i$'s are listed.

2. For $k > 46$ permutation decoding does not work.

$[51, k, 2]$ codes:

1. For $k \leq 33,\ R_0$ and R_1 will suffice.

2. For $k > 33$ permutation decoding is not possible.

3. The relevant BCH code $[51, 35, 2]$ is not permutation decodable.

$[63, k, 2]$ codes:

1. For $k \leq 41,\ R_0$ and R_1 will suffice.

2. For $k > 41$ not possible.

3. The relevant BCH code $[63, 51, 2]$ is not permutation decodable.

$[65, k, 2]$ codes:

1. If $k \leq 51,\ R_i$'s are listed.
2. For $k > 51$ permutation decoding does not work.

$[69, k, 2]$ codes:

1. For $k \leq 45,\ R_0$ and R_1 will suffice.
2. For $k > 45$ permutation decoding is not possible.
3. The relevant BCH code is $[69, 36, 2)]$ and require R_0 and R_1 .

$[73, k, 2]$ codes:

1. If $k \leq 61,\ R_i$'s are listed.
2. For $k > 61$ permutation decoding does not work.

$[127, k, 2]$ codes:

1. If $k \leq 105,\ R_i$'s are listed.
2. For $k > 105$ permutation decoding does not work.

$[255, k, 2]$ codes:

1. For $k \leq 169,\ R_0$ and R_1 will suffice.
2. For $k > 169$ permutation decoding is not possible.

Certain triple error correcting binary codes

Shiva and Fung in this paper [34] continued to analize certain triple error correcting binary codes as in the previous section. They gave corresponding results for triple error correcting binary codes of certain lengths. They use the same notation given in [33]. The authors analyzed $[n, k, 3]_2$ codes, where $n = 15, 17, 21, 23, 31, 45, 47, 51, 63, 65, 69, 73$ and 127.

Group codes

Group codes are generated as follows. Consider a group G of $N \times N$ orthogonal matrices which forms a faithful representation of an abstract group $\mathcal{G}$ with M elements, and an "initial vector" $x \in \mathbb{R}^N$, where $\mathbb{R}^N$ is the N-dimensional Euclidean space. A group code χ is the orbit x under $\mathcal{G}$. i.e. the set of vectors Gx. By assuming that the only solution of the equation $gx = x,\ g \in G$, is $g = 1$, the identity. The code χ has M elements. Thus we denote by x_g the code vector associated with $g \in \mathcal{G}$. Biglieri [2] applied the permutation decoding method to group codes and constructed the following algorithm.

With the vectors of χ transmitted over the additive white Gaussian noise channel, the optimum decoder, upon reception of the noisy vector $r = x_g + n$, chooses as the most likely transmitted vector the one that yields

$$min\ ||r - x_g||^2.$$

If $\mathcal{G}$ is not endowed with any special structure, decoding is obtained by exhaustive search among all the candidates $g \in \mathcal{G}$. This requires a number of calculations $v_c = NM$ and a storage of $v_s = NM$. In addition to this, the minimum has to be found, which requires v_M operations.

The permutation signal set (PSS) is a set of vectors that are obtained by applying a group $\mathcal{G}$ of permutations to an initial vector x. If the vectors have n components, application of the symmetric group S_n of all the permutations of n letters to an initial n-vector gives a class of codes known as "permutation modulation".

If the PSS is generated by a subgroup $\mathcal{G}$ of S_n.

- First decode r as if $\mathcal{G} = S_n$, obtaining as a result a permutation π of n letters. This may not belong to $\mathcal{G}$.
- Next algebraically decode π into an element of $\mathcal{G}$.

It can be proved that

- Every group code can be represented in the form of a permutation signal set acting on an initial vector x with n components.
- The minimum value of n is obtained as follows: If $|\mathcal{H}^{'}|$ denotes the largest non-normal subgroup of $\mathcal{G}$ that does not include normal subgroups of $\mathcal{G}$ other than the identity , then n is given by the ratio

$$n = \frac{|\mathcal{G}|}{|\mathcal{H}^{'}|}$$

Permutation decoding of abelian codes

Chabanne [4] introduced permutation decoding procedure for abelian codes using the Groebner basis theory [5]. The method is valid for decoding all the binary abelian codes. The paper explains how to calculate syndromes via Groebner basis theory, then generalizes the permutation decoding procedure due to MacWilliams.

Definition 3.4. *Let K be the finite field $GF(2)$, we denote by R_{N_1,N_2} the quotient algebra $R_{N_1,N_2} = K[x,y]/(x^{N_1}+1, y^{N_2}+1)$, where N_1, N_2 are odd integers. A binary 2D cyclic code of area N_1N_2 is an ideal C of the semisimple algebra R_{N_1,N_2}. Each codeword $\bar{a} = (a_{i,j})$ is represented as a bivariate polynomial (modulo $(x^{N_1}+1, y^{N_2}+1)$)*

$$a(x,y) = \sum_{i=0}^{N_1-1} \sum_{j=0}^{N_2-1} a_{i,j} x^i y^j.$$

If $a(x,y) \in R_{N_1N_2}$ is a codeword of C then $xa(x,y), ya(x,y)$ and $a(x^2,y^2)$ are also codewords of C.

Let α and β be primitive N_1th and N_2th roots of 1 and denote by $Z_{N_1N_2}$ the set of all N_1N_2 pairs (α^i, β^j), $i \in [0, N_1]$ and $j \in [0, N_2]$. As $R_{N_1N_2}$ is semisimple, all these ideals are principal. An ideal C of $R_{N_1N_2}$ can be uniquely determined by a set $Z_C = \{(\alpha^i, \beta^j) : (i,j) \in T\}$ such that $a(x,y)$ is a codeword of C if and only if $a(\alpha^i, \beta^j) = 0$ for all $(i,j) \in T$. Since for all $a(x,y) \in R_{N_1N_2}, a(\alpha^i, \beta^j) = 0 \Leftrightarrow a(\alpha^{2i}, \beta^{2j}) = 0$, T is invariant under the multiplication by 2 modulo (N_1N_2). We shall denote by $\hat{Z}_C$ the subset of Z_C that contain

only one element of each orbit under $(\alpha_i, \beta^j) \mapsto (\alpha^{2i}, \beta^{2j})$. The dimension of the $K-$ vector space C is $\dim(C) = N_1N_2$ - card(Z_C).

Conversely given a set Z of pairs (α^i, β^j) invariant by $(\alpha^i, \beta^j) \mapsto (\alpha^{2i}, \beta^{2j})$, define the ideal C_Z of $R_{N_1N_2}$ by

$$C_Z = \{a \in R_{N_1N_2} : a(\alpha^i, \beta^j) = 0,\ \forall (\alpha^i, \beta^j) \in Z.$$

Moreover, via the inverse Fourier transform $\mathcal{F}^{-1}$, we can give a generator g of $C = C_Z$. In fact,

$$g = \mathcal{F}^{-1}\left(\sum_{(\alpha^i,\beta^j)\in Z_{N_1N_2}\setminus Z_C} x^i y^j\right)$$

where $\mathcal{F}^{-1}$ is defined by,

$$\mathcal{F}^{-1}(b) = \frac{1}{N_1N_2}\sum_{i=0}^{N_1-1}\sum_{j=0}^{N_2-1} b(\alpha^{-i}, \beta^{-j})x^i y^j.$$

Let C be an ideal in $R_{N_1N_2}$ generated by $g(x, y)$ and consider the ideal $I_C = (g, x^{N_1} + 1, y^{N_2} + 1)$ of $K[x, y]$. Let $B = \{GB_1, GB_2, \ldots, GB_s\}$ be a fixed Groebner basis [5] for I_C. Let $R(m, I_C)$ be the remainder of m with respect to B.

Lemma 3.5. *If $a \in K[x, y]$ then $a + R(a, I_C) \in I_C$.*

Result 3.6. *Let C be a $[N_1N_2, k, d]$ linear code which is an ideal of $R_{N_1N_2}$. If $m = c + e$, where $c \in C$ and $e \in P(Rec)\backslash C$ with $wt(e) \leq \lfloor (d-1)/2 \rfloor$. Moreover $k = Card(St(I_C) \cap P(Rec))$*

The following decoding algorithm generalizes the permutation decoding method for Abelian codes.

Input $m = c + e$ with $c \in C$ and e as in the theorem,

$\{GB_1, GB_2, \ldots, GB_s\}$, a minimal Groebner basis for I_C,

Π, a subset of the automorphism group of C.

Begin Repeat

Take $\pi \in \Pi$

$\Pi = \Pi \backslash \{\pi\}$

$S = R(\pi(m), I_C)$

Until wt(S)$\leq t$ or $\Pi = \emptyset$

If wt(S)$\leq t$ then $c = \pi^{-1}(\pi(m) + S)$

End

Microprocessor controlled permutation decoding of block error correcting codes

Goodman and Green [11] proposed a new soft-decision permutation decoding algorithm for cyclic block codes. The implementation of both hard and soft-decision permutation decoding on an Intel 8080A microprocessor system is discussed.

Permutation decoding for the binary codes from triangular graphs

For any n the triangular graph $T(n)$ is defined to be the line graph of the complete graph K_n. Key, Moori and Rodrigues [19] found explicit PD-sets for the binary codes obtained from an adjacency matrix of the triangular graph $T(n)$ for any $n \geq 5$.

Result 3.7. *Let $\mathcal{I}$ denote the subset*

$$P_1 = \{1, n\}, P_2 = \{2, n\}, \ldots, P_{n-1} = \{n-1, n\}$$

of vertices of the triangular graph $T(n)$ where $n \geq 5$, and let C denote a binary code of $T(n)$ with $\mathcal{I}$ in the first $n-1$ positions. Then

1. *C is a $[\binom{n}{2}, n-1, n-1]_2$ code for n odd and, with $\mathcal{I}$ as the information positions,*

$$\mathcal{S} = \{1_G\} \cup \{(i, n) | 1 \leq i \leq n-1\}$$

is a PD- set for C of n elements in S_n.

2. *C is a $[\binom{n}{2}, n-2, 2(n-1)]_2$ code for n even, and with $\mathcal{I}$ excluding P_{n-1} as the information positions,*

$$\mathcal{S} = \{1_G\} \cup \{(i,n) | 1 \leq i \leq n-1\} \cup \{[(i,n-1)(j,n)]^{\pm 1} | 1 \leq i,j \leq n-2\}$$

is a PD-set for C of $n^2 - 2n + 2$ elements in S_n.

Binary codes from lattice graphs

The lattice graph $L_2(n)$ $(n \geq 2)$ has as vertices the ordered pairs $(i,j),\ 1 \leq i,j \leq n$, where two pairs are adjacent if they have a common coordinate. The graph $L_2(n)$ is strongly regular of type $(n^2,\ 2(n-1),\ n-2,\ 2)$.

Result 3.8. *For $n \geq 5$, the automorphism group of the lattice graph $L_2(n)$ is $S_n \wr S_2$, the wreath product of S_n with S_2. The binary code formed by the row space over F_2 of an adjacency matrix for $L_2(n)$ is a $[n^2, 2(n-1), 2(n-1)]_2$ code with $S_n \wr S_2$ acting as an automorphism group.*

Key and Seneviratne [21] and Seneviratne [27] found explicit PD-sets for the full error correction of these codes.

Result 3.9. *Let C be the $[n^2, 2(n-1), 2(n-1)]$ binary code formed from the row span of an adjacency matrix for the lattice graph $L_2(n)$. Then by using the set of points $\{(i,n) | 2 \leq i \leq n-1\} \cup \{(n,i) | 1 \leq i \leq n\}$ as information symbols, the set of automorphisms,*

$$\mathcal{S} = \{((i,n),(j,n)) | 1 \leq i,j \leq n\}$$

forms a PD-set of size n^2 for C.

Graphs on triples and permutation decoding

Key, Moori and Rodrigues [18] showed permutation decoding can be used to find explicit PD-sets for binary codes obtained from adjacency matrices of the graphs on $\binom{n}{3}$ vertices for $n \geq 7$.

Result 3.10. *Let Ω be a set of size n, where $n \geq 7$ and n is odd. Let $\mathcal{P} = \Omega^{\{3\}}$, the set of subsets of Ω of size 3, be the vertex set of the graph $A_2(n)$ with adjacency defined by two vertices(as 3-sets) being adjacent if the 3-sets meet in two elements. Let $C_2(n)$ denote the code formed from the row span over $\mathbb{F}_2$ of an adjacency matrix for $A_2(n)$.*
The dual $C_2(n)^\perp$ is a $[\binom{n}{3}, \binom{n-1}{3}, n-2]$ code with,

$$\mathcal{I} = \{\{i,j,n\} | 1 \leq i < j < n\} \cup \{\{n-3, n-2, n-1\}\} \backslash \{\{n-2, n-1, n\}\}$$

as information set. Then $C_2(n)^\perp$ has a PD-set in S_n given by the following elements of S_n in their natural action on triples of elements of $\Omega = \{1, 2, \ldots, n\}$:

$$\mathcal{S} = \{(n,i)(n-1,j)(n-2,k) | 1 \leq i \leq n, 1 \leq j \leq n-1, 1 \leq k \leq n-2\},$$

where (i,i) denotes the identity element of S_n.

Partial permutation decoding for codes from Paley graphs

Key and Limbupasiriporn [15] examined codes from Paley graphs and observed that after a certain length, PD sets to correct errors upto the code's error capability will not exist. They used partial permutation decoding for correcting two errors.

Result 3.11. *Let $C = [n,k,d]_q$ be a cyclic code of prime length n over the field F_q of order q, where $n \equiv 1 (mod 8)$, $(n,q) = 1$ and $d \geq 5$. Label the coordinate positions $0, 1, \ldots, n-1$ and suppose that $0, 1, \ldots, k-1$ form the information symbols. Let $\tau_{a,b} : i \mapsto ai + b$ for $a, b \in \mathbb{F}_n$ and a, a nonzero-square and suppose that $\tau_{a,b} \in Aut(C)$ for all such $a, b \in \mathbb{F}_n$. Then*

1. *if $k = \frac{n-1}{2}$ the set*

$$\{\tau_{1,b} | b \in \{0, k\}\} \cup \{\tau_{k,b} | b \in \{k, 2k, \frac{3k}{2}, \frac{k}{2} - 1\}\}$$

is a 2-PD set of size 6 for C;

2. if $k = \frac{n+1}{2}$ *the set*

$$\{\tau_{1,b} | b \in \{0, 1, k, k-1, n-1\}\} \cup \{\tau_{k,b} | b \in \{0, k, k-1, \frac{k-1}{2}, \frac{3k-1}{2}\}\}$$

is a 2-PD set of size 10 *for* C.

Partial permutation decoding for codes from finite planes

Key, McDonough and Mavron [17] defined the notion of s-PD sets to correct s errors. Also they discuss to what extent permutation decoding could be used for the codes from the desarguesian projective and affine planes. We briefly state some results from [17].

Result 3.12. *Let* $\Pi = PG_2(\mathbb{F}_q)$, *where* $q = p^e$ *and* p *is a prime,* $C = [q^2, (p(p+1)/2)^e + 1, q+1]_p$, *its* p*-ary code, and* G *its automorphism group. Then if* $q \geq 7$, *a* 3*-PD set can be found in* G *for* C *using any information set; similarly for* $q \geq 5$ *for the dual code* $C^\perp = [q^2+q+1, q^2+q-(p(p+1)/2)^e, d^\perp]_p$, *where* $q + p \leq d^\perp \leq 2q$.

If $q \geq 8$, *information sets exists for* C *such that* 4*-PD sets can be found in* G; *similarly for* $C^\perp$ *for* $q \geq 5$.

Result 3.13. *Let* $\pi = AG_2(\mathbb{F}_q)$, *where* $q = p^e$ *and* p *is a prime,* $C = [q^2+q+1, (p(p+1)/2)^e, q]_p$, *its* p*-ary code, and* G *its automorphism group. Then if* $q \geq 7$, *a* 3*-PD set can be found in* G *for* C. *Similarly, for* $q \geq 5$, *a* 3*-PD set can be found in* G *for the dual code* $C^\perp = [q^2, q^2-(p(p+1)/2)^e, d^\perp]_p$, *where* $p + q \leq d^\perp \leq 2q$.

For $q = p$, using the Moorhouse basis, they obtained a similar result for prime-order desarguesian affine planes for 4-PD sets. Further they constructed explicit 2-PD sets for desarguesian planes of prime order. We refer to [17] for a complete discussion.

Codes from finite geometries

Key, McDonough and Mavron [16] determined information sets for the generalized Reed-Muller codes and used these to apply partial permutation decoding to codes from finite geometries over prime fields.

For any finite field $\mathbb{F}_q$ of order q, the set of points and r-dimensional subspaces (respectively flats) of an m-dimensional projective (respectively affine) geometry forms a 2-design which is denoted by $PG_{m,r}(\mathbb{F}_q)$ (respectively $AG_{m,r}(\mathbb{F}_q)$). The automorphism groups, $P\Gamma L_{m+1}(\mathbb{F}_q)$ or $A\Gamma L_m(\mathbb{F}_q)$, respectively, of these designs (and codes) are the full projective or affine semi-linear groups, and always 2-transitive on points. If $q = p^e$, where p is a prime, the codes of these designs are over $\mathbb{F}_p$ and are subfield subcodes of the generalized Reed-Muller codes. The dimension and the minimum weight is known in each case.

The generalized Reed-Muller codes are defined as follows:

Definition 3.14. *Let $V = \mathbb{F}_q^m$ be the vector space of m-tuples, for $m \geq 1$, over $\mathbb{F}_q$, where $q = p^t$ and p is a prime. For any ρ such that $0 \leq \rho \leq m(q-1)$, the ρ^{th} order generalized Reed-Muller code $\mathcal{R}_{\mathbb{F}_q}(\rho, m)$ is the subspace of $\mathbb{F}_q^V$ (with basis the characteristic functions of vectors in V) of all m-variable polynomial functions (reduced modulo ${x^r}_i - x_i$) of degree at most ρ. Thus*

$$\mathcal{R}_{\mathbb{F}_q}(\rho, m) = < {x_1}^{i_1} {x_2}^{i_2} \dots {x_m}^{i_m} | 0 \leq i_k \leq q-1, \ for \ 1 \leq k \leq m, \sum_{k=1}^{m} i_k \leq \rho > .$$

The authors obtained a general lemma that finds a number s such that a code C with an automorphism group G will have G as an s-PD set.

Result 3.15. *Let C be a code with minimum distance d, $\mathcal{I}$ an information set, $\mathcal{C}$ the corresponding check set and $\mathcal{P} = \mathcal{I} \cup \mathcal{C}$. Let G be an automorphism group of C, and n maximum of $|\mathcal{O} \cap \mathcal{I}|/|\mathcal{O}|$, where $\mathcal{O}$ is a $G-$ orbit. If $s = min(\lceil \frac{1}{n} \rceil - 1, \lfloor \frac{d-1}{2} \rfloor)$, then G is an s-PD set for C.*

The authors give several s-PD sets for generalized Reed-Muller codes. We refer to [16] for a full description.

Anti-blocking systems and PD-sets

Kroll and Vincenti [25] introduced the notion of anti-blocking system to check whether the size of a PD-set $\mathcal{S}$ satisfies the Gordon bound. They presented four examples where the Gordon bound is not sharp.

PD-sets for the codes related to some classical varieties

Kroll and Vincenti [24] generalized the notion of a PD-set of a code to that of a t-PD-set of an arbitrary permutation set. They found PD-sets for miquelian Benz planes of small order and for the ruled rational normal surface of order 3 in PG(4, 3) and in PG(4, 4).

PD-sets for Grassmannian codes of dimension $k \leq 6$

An $[n,k]_q-$projective system X of the $(k-1)$-dimensional projective geometry $P_{k-1} = PG(k-1,q)$ over $F = GF(q)$ is a collection of n not necessarily distinct points. An $[n,k]_q-t-$error correcting linear code C is a $k-$dimensional vector subspace of F^n. There exists a natural 11 correspondence between the equivalence classes of $[n,k]-$projective systems X and those of linear $[n,k]-$codes

Kroll and Vincenti [26] focused their attention on codes related to classical varieties, essentially on some Grassmannian codes. They found PD-sets for the binary first-order Reed-Muller codes $\bar{\mathcal{R}}(1,4)$ and $\bar{\mathcal{R}}(1,5)$ and for the codes related to the Klein quadric KQ of $PG(5,2)$ and to the Schubert subvariety of KQ. In each case, to get a PD-set they constructed a collection of bases of P_{k-1} contained in the projective system related to the code.

Binary codes and Permutation decoding sets from the class of odd graphs

W. Fish [7] is working on codes related to odd graphs and permutation decoding.

Conjectural permutation decoding of some algebraic geometric codes

In Joyner [6] discussed permutation decoding of certain algebraic geometric codes. The main results are the conjectures regarding complexity of the permutation decoding of these codes.

Conjecture 3.16. *For one point AG codes C in standard form associated to $y^2 = x^p - x$ over $GF(p)$ of length $n = p$, permutation decoding always applies. The complexity in codeword operations is at worst the size of the permutation group of C, which is $O(p^2) = O(g^2) = O(n^2)$.*

Conjecture 3.17. *Assume $p \equiv 3(mod\ 4), p > 3$. For one-point AG codes $C = C(m.(1 : 0 : 0), O_2)$ of length $n = 2p(p-1)$ in standard form associated to the hyperelliptic curve X over $GF(p^2)$ defined by $y^2 = x^p - x$, permutation decoding always applies. If the points in $X(f)$ are arranged suitably then the image of $Stab_G((1 : 0 : 0)) \subset G = Aut_F(X)$ in the permutation group of C may be used as a PD-set. The complexity in codeword operations is at worst $O(p^2) = O(g^2) = O(n)$.*

CHAPTER 4

BINARY CODES FROM RECTANGULAR LATTICE GRAPHS

4.1 Introduction

In this chapter we define binary codes from rectangular lattice graphs and get PD-sets for full error correction. The rectangular lattice graph $L_2(m,n)$ is defined to be the line graph of the complete bipartite graph $K_{m,n}$, where $m,n \in \mathbb{Z}$. It is a regular graph of valency $m+n-2$ on $v=mn$ vertices, i.e. on the ordered pairs $< i,j >$ where $1 \leq i \leq m$ and $1 \leq j \leq n$, with adjacency defined by $< i,j >$ and $< k,l >$ being adjacent if $i=k$ and $j \neq l$ or $j=l$ and $i \neq k$. If $m=n$ then this is the strongly regular square lattice graph, $L_2(n)$. In Section 3.9 we applied permutation decoding to the square latttice graphs and obtained PD-sets of size n^2 for full error correction and so we exclude the case $n=m$.

4.2 The binary codes

Let $2 \leq m < n$ be integers and let $L_2(m,n)$ denote the rectangular lattice graph with the vertex set $\mathcal{P}$ the mn ordered pairs $< i,j >, 1 \leq i \leq m, 1 \leq j \leq n$. The 1-design $\mathcal{D}=(\mathcal{P},\mathcal{B})$ will have point set $\mathcal{P}$ and for each point $< i,j > \in \mathcal{P}, 1 \leq i \leq m, 1 \leq j \leq n$, a block, which we denote by $\overline{< i,j >}$, is defined in the following way:

$$\overline{< i,j >} = \{< i,k > | k \neq j\} \cup \{< k,j > | k \neq i\}.$$

Thus the block size is $m+n-2$ and $\mathcal{D}$ is symmetric 1-design with the block set

$$\mathcal{B} = \{\overline{< i,j >} | 1 \leq i \leq m, 1 \leq j \leq n\}.$$

The incidence vector of the block $\overline{< i,j >}$ is

$$v^{\overline{<i,j>}} = \sum_{k \neq j} v^{<i,k>} + \sum_{k \neq i} v^{<k,j>} = \sum_{k=1}^{n} v^{<i,k>} + \sum_{k=1}^{m} v^{<k,j>} \tag{4.1}$$

where, as usual with the notation from [1], the incidence vector of the subset $\mathcal{X} \subseteq \mathcal{P}$ is denoted by v^X, but writing $v^{<i,j>}$ instead of $v^{\{(i,j)\}}$.

The group $S_m \times S_n$ acts naturally on $\mathcal{D}$ and thus on C in the following way: if $\sigma = (\sigma_1, \sigma_2)$ where $\sigma_1 \in S_m$ and $\sigma_2 \in S_n$, then for $< i, j > \in \mathcal{P}, < i, j >^{\sigma} = < i^{\sigma_1}, j^{\sigma_2} >$.

Proposition 4.1. *Let C be the binary code of $L_2(m, n)$ where $1 \leq m < n$. Then*

$$dim(C) = \begin{cases} m + n - 1 & \text{for } m + n \text{ odd} \\ m + n - 2 & \text{for } m + n \text{ even.} \end{cases} \tag{4.2}$$

Proof: Let M be a vertex-edge incidence matrix for $K_{m,n}$, where the two parts of the graph are $\Lambda_1 = \{1, 2, \ldots, m\}$ and $\Lambda_2 = \{1, 2, \ldots, n\}$, ordering the rows of M by taking the points of Λ_1 followed by the points of Λ_2, and ordering the edged by taking all the edges throught the first point, followed by all the edges through the second point, and so on. Then $M^T M = A$ is an adjacency matrix for $L_2(m, n)$. If C_A denotes the row span of A over $\mathbb{F}_2$ and C_M that of M, then $C_A \subseteq C_M$. Clearly $dim(C_M) = m + n - 1$.

If V denotes the row span of M^T then $\tau : V \to C_A$ by $\tau : v \mapsto vM$ has $V\tau = C_A$, so $dim(C_A) = m + n - 1$ or $m + n - 2$, the latter if and only if $\boldsymbol{\jmath} = (1, 1, \ldots, 1) \in \mathbb{F}_2^{m+n}$ is in V. Considering the form M^T that we have chosen, it is easy to see that if both m and n are odd, then $\boldsymbol{\jmath} \in V$. Similarly if both are even, it follows that $\boldsymbol{\jmath} \in V$.

The only case that needs further consideration is when one is odd and other even, and in this case we show that $\boldsymbol{\jmath} \notin V$. The rows of M^T are arranged in m sections of n rows each; the columns are in two sections, the first of m columns , the second of n. If $\boldsymbol{\jmath} \in V$ then as a sum of the rows of M^T, $\boldsymbol{\jmath} = (x_1, x_2, \ldots, x_m, y_1, y_2, \ldots, y_n)$, where x_i is the number of rows in the sum from the i^{th} section of rows, for $i = 1, 2, \ldots, m$. Thus x_i is odd for $i = 1, 2, \ldots, m$. For the entries in the columns starting at the $(m + 1)^{th}$, (i.e the vertices through the Λ_2 points), suppose the i^{th} point, for $i = 1, 2, \ldots, n$, in Λ_2 contributes $n_{i,j}$ to the vector from the j^{th} section of rows of M^T, for $j = 1, 2, \ldots, m$, where $n_{i,j} = 0$ or 1, from the form of M^T.

Thus $y_i = \sum_{j=1}^{m} n_{i,j}$ for $i = 1, 2, \ldots, n$, and $x_i = \sum_{i=1}^{n} n_{j,i}$ for $i = 1, 2, \ldots, m$. Summing by rows and by columns gives $s = \sum_{i,j} n_{i,j} = \sum_{i=1}^{m} x_i = \sum_{j=1}^{n} y_j$. If $m + n$ is odd, this contradicts all the x_i and y_j being odd, and thus $\boldsymbol{\jmath} \notin V$ in this case, which completes the proof. ■

Proposition 4.2. *Let C be the binary code of $L_2(m, n)$ where $1 \leq m < n$. Then C has minimum weight m if $m + n$ is odd, and $2m$ if $m + n$ is even.*

Proof: If $m + n$ is odd then $C = C_M$ from the previous proposition, so clearly there are words of weight m. That there cannot be words of smaller weight in C_M is clear from the form of M. In general we have from $v^{\overline{<i,j>}} = \sum_{k=1}^{n} v^{<i,k>} + \sum_{k=1}^{m} v^{<k,j>}$ that

$$\sum_{j=1}^{n} v^{\overline{<i,j>}} = n \sum_{k=1}^{n} v^{<i,k>} + \jmath,$$

$$\sum_{i=1}^{m} v^{\overline{<i,j>}} = m \sum_{k=1}^{m} v^{<k,j>} + \jmath,$$

and

$$v^{\overline{<i,j>}} + v^{\overline{<i,k>}} = \sum_{l=1}^{m} v^{<l,j>} + \sum_{l=1}^{m} v^{<l,k>}.$$

If $m+n$ is even and $C \neq C_M$, then again the form of M shows that $2m$ is the next possible weight, and C does have such words, as shown by the last equation above. ■

Proposition 4.3. *Let C be the binary code of $L_2(m, n)$ where $2 \leq m < n$. Then for i_1, i_2 distinct elements in $\{1, 2, \ldots, m\}$ and j_1, j_2 distinct elements in $\{1, 2, \ldots, n\}$, the vector*

$$u(i_1, i_2; j_1, j_2) = v^{<i_1,j_1>} + v^{<i_1,j_2>} + v^{<i_2,j_1>} + v^{<i_2,j_2>}$$

is a weight-4 vector in $C^{\perp}$.

If $s_i = \{i, i + 1\}$ for $1 \leq i \leq m - 1$ and $t_i = \{i, i + 1\}$ for $1 \leq i \leq n - 1$, the set of vectors$\{u(s_i; t_j) | 1 \leq i \leq m - 1, 1 \leq j \leq n - 1\}$ form a linearly independent set of $mn - m - n + 1$ vectors that form a basis for $C^{\perp}$ for $m + n$ odd, and together with $\boldsymbol{\jmath}$ when

$m+n$ is even. Furthermore, the points

$$<1,1>,\ldots,<1,n-1>,<2,1>,\ldots,<2,n-1>,\ldots,<m-1,1>,\ldots,<m-1,n-1>$$

form an information set for $C^{\perp}$ for $m+n$ odd, and together with $<1,n>$ when $m+n$ is even.

Proof: It is easy to verify that the vectors $u(i_1,i_2;j_1,j_2)$ are in $C^{\perp}$. If the coordinate positions are then arranged as shown in the statement and the vectors $u(s_i;t_j)$ as rows in the order

$$(s_1;t_1),(s_1;t_2),\ldots,(s_1;t_{n-1}),(s_2;t_1),\ldots,(s_2;t_{n-1}),\ldots,(s_{m-1};t_1),\ldots,(s_{m-1};t_{n-1})$$

then the resulting matrix is already in row echelon form. In the case $m+n$ even, the vector $\boldsymbol{\jmath}$ can be added to obtain a further basis element. ■

4.3 PD-sets

Theorem 4.4. *If C is the binary code formed by the row space over $\mathbb{F}_2$ of an adjacency matrix for the rectangular lattice graph $L_2(m,n)$ for $2 \leq m < n$. Let*

$$S_e = \{((i,m),(j,n))|1 \leq i \leq m, 1 \leq j \leq m\} \cup \{id\},$$

$$S_o = \{((i,m),(i,n))|1 \leq i \leq m\} \cup \{id\}$$

be sets of permutations in $S_m \times S_n$. Then for $3 \leq m < n$, S_e is a PD-set m^2+1 elements for C for $m+n$ even, and S_o is a PD-set of $m+1$ elements for C for $m+n$ odd.

Note that we use (r,r) to denote the identity element of S_r. We also take $m \geq 3$ since we only need PD-sets for t-error correction where $t \geq 2$.

Proof: From Proposition 4.3, an information set for C is

$$\mathcal{I} = \{<i,n>|1 \leq i \leq m\} \cup \{<m,i>|1 \leq i \leq n-1\}$$

for $m+n$ odd and $\mathcal{I}\backslash\{< 1, n >\}$ for $m+n$ even. Suppose C can correct t errors and let

$$\mathcal{T} = \{(a_1, b_1), (a_2, b_2), \ldots, (a_s, b_s)\}$$

be a set of $s \leq t$ points. We need an element from our set of involutions that maps $\mathcal{T}$ into the check positions $\mathcal{C}$. Let $\Omega_1 = \{a_i | 1 \leq i \leq s\}$ and $\Omega_2 = \{b_i | 1 \leq i \leq s\}$.

Take first $m+n$ even, so $t = m-1$. Then $|\Omega_i| \leq m-1 < n-1$. Thus we can find l such that $1 \leq l \leq m$ and $l \notin \Omega_2$. If there exists k such that $1 \leq k \leq m-1$ and $k \notin \Omega_1$, then the element $\sigma = ((k, m), (l, n)) \in S_e$ will move all the elements of $\mathcal{T}$ into C. If $\Omega_1 = \{1, \ldots, m-1\}$, then $\sigma = ((m, m), (l, n))$ will map $\mathcal{T}$ to $\mathcal{C}$, where (m, m) denotes the identity permutation. Including the identity automorphism covers the case where $\mathcal{T} \subset \mathcal{C}$.

If $m+n$ is odd, then the minimum weight is m so $t = \lfloor \frac{m-1}{2} \rfloor$. Thus $s \leq \frac{m-1}{2}$ and there exist k such that $1 \leq k \leq m$ and $k \notin \Omega_1$ and $k \notin \Omega_2$. The element $\sigma = ((k, m), (k, n)) \in S_o$ will map $\mathcal{T}$ to $\mathcal{C}$. Again including the identity automorphism covers the case where $\mathcal{T} \subset \mathcal{C}$. This completes the proof of the theorem. ■

CHAPTER 5

BINARY CODES FROM THE LINE GRAPH OF COMPLETE MULTIPARTITE GRAPHS

5.1 Introduction

In this chapter we consider the line graphs $L_m(n)$ of the complete multipartite graphs $K_{n_1,\ldots,n_m}$ where $n_i = n$ for $1 \leq i \leq m$ and $n \geq 2$, as candidates for binary codes to which permutation decoding can be applied, i.e. PD-sets (for full error-correction) or s-PD-sets (for correcting s errors, see Section 5.2) can be found. Since the automorphism group of $L_m(n)$ and of its binary code is $G = S_n \wr S_m$ (where S_r denotes the symmetric group of degree r) it can be expected that information sets can be found for permutation decoding. We have found PD-sets for some classes and s-PD-sets for all the classes. Our main results are summarized in the following theorem, where the notation for the points of the information set is defined in Equation 5.1 of Section 5.2 and where we take $m \geq 3$, since $m = 2$ has been considered earlier in Chapter 3.9 and Chapter 4.

Theorem 5.1. *If C is the binary code of the line graph $L_m(n)$ of the complete multipartite graph $K_{n,\ldots,n}$ of nm vertices, where $n \geq 2$, $m \geq 3$, then*

- *C is a $[\frac{1}{2}m(m-1)n^2, mn-2, 2n(m-1)-2]_2$ code for mn even;*
- *C is a $[\frac{1}{2}m(m-1)n^2, mn-1, n(m-1)]_2$ code for mn odd.*

Let

$$\mathcal{I} = \{(1,1:i,j) \mid 2 \leq i \leq m, 1 \leq j \leq n\} \cup \{(1,i:2,1) \mid 2 \leq i \leq n\} \setminus \{(1,1:m,n)\}$$

and $\mathcal{I}^* = \mathcal{I} \cup \{(1,2:m,n)\}$. Then $\mathcal{I}$ is an information set for C if mn is even, and $\mathcal{I}^*$ is an information set for mn odd. Using these information sets

1. *if $n = 2$ and $m \geq 3$, C has a PD-set of size $16m^2$;*
2. *if $n = 3$ and $m \geq 3$ is odd, C has a PD-set of size $27m$;*

3. if $m = 3$ and $n \geq 3$ is odd, C has a PD-set of size $2n^3$.

Furthermore, s-PD-sets of size N exist as follows: $s < m/2$, $N = m$; $s < m$, $N = mn^2$; $s < 3m/2$, $N = mn^3$; $s < 2m$, mn even, $N = 4m^2n^2$; $s < n/2$, $N = n$ for mn even, $N = 2n$ for mn odd; $s < n$, $N = n^3$ for mn even, $N = 2n^3$ for mn odd.

The parts of this theorem are proved, and the explicit PD-sets or s-PD-sets are given, in the following sections as Propositions 5.2, 5.4, 5.7, 5.11, 5.13, 5.15 and Corollaries 5.12 and 5.14. Note that we do not deal with the case $m = 2$ since this was done previously in Chapter 3 and Chapter 4. Also note that these sizes are not necessarily the best, and in most explicit cases, smaller ones can be found with Magma [29], for example. It is also assumed that one only considers using s-PD-sets if $s \leq t$, where t is the full error-correction cabability of the code.

5.2 The binary codes

Let $K_{n_1,\ldots,n_m}$ denote the complete multipartite graph on m components. If $n_i = n \geq 2$, for $1 \leq i \leq m$, where $m \geq 3$, then denote the graph by K_n^m. The vertices of K_n^m correspond to the ordered pairs (i, j) for $1 \leq i \leq m$ and $1 \leq j \leq n$, which is the j^{th} point on the i^{th} component, Λ_i. The **line graph** $L_m(n)$ of K_n^m has for vertices the edges of K_n^m and two vertices $L_m(n)$ are adjacent if as edges of K_n^m they had a vertex in common.

We will use the following compact notation for the vertices (points) of $L_m(n)$:

$$\{(i, j), (k, r)\} = (i, j : k, r) = (k, r : i, j). \tag{5.1}$$

Let $\mathcal{P}$ denote the set of all vertices of $L_m(n)$. The symmetric 1-design $\mathcal{D} = (\mathcal{P}, \mathcal{B})$ defined from $L_m(n)$ by taking the vertices as point set $\mathcal{P}$ and defining a block $B = \overline{P}$ for each point P as consisting of the vertices adjacent to that point, i.e.

$$\overline{(a, b : c, d)} = \{(a, b : e, f) \mid (e, f) \neq (c, d)\} \cup \{(c, d : e, f) \mid (e, f) \neq (a, b)\},$$

is a

$$1 - (\frac{1}{2}m(m-1)n^2, 2n(m-1) - 2, 2n(m-1) - 2)$$

design. The binary code of $\mathcal{D}$ is $C_{\mathbb{F}_2}(\mathcal{D}) = \langle v^B | B \in \mathcal{B} \rangle$ where the incidence vector of $B = \overline{(a, b : c, d)}$ is given by

$$v^{\overline{(a,b:c,d)}} = \sum_{(e,f),\, e \neq a} v^{(a,b:e,f)} + \sum_{(e,f),\, e \neq c} v^{(c,d:e,f)}$$

of weight $2n(m-1) - 2$.

Now we obtain the basic results about the binary codes of the graphs $L_m(n)$, starting with the dimension of the codes:

Proposition 5.2. *Let C be the binary code of $L_m(n)$ where $n \geq 2$ and $m \geq 3$. Then C has dimension $mn - 1$ if mn is odd, and dimension $mn - 2$ if mn is even.*

Proof: Let M be a vertex-edge incidence matrix for K_n^m. Then $M^T M = A$ is an adjacency matrix for $L_m(n)$. Thus C is the row span of A over $\mathbb{F}_2$, and clearly C is a subcode of C_M, the row span of M. M has mn rows and each column has two entries. Thus the sum of all the rows is the zero vector, and the rank of C_M is at most $mn - 1$.

Order the vertices (rows) and edges (columns) of K_n^m as follows: for the vertices, take the m components in turn, that is, calling these Λ_i for $i = 1$ to m, take $\Lambda_1, \Lambda_2, \ldots, \Lambda_m$. For the edges, take all the edges through the first point in Λ_1, then all the edges through the second point on Λ_1, and so on. Then take the remaining edges through the first point on Λ_2, then the second point on Λ_2, and so on. (See the illustration of this for $m = n = 3$ at the end of the proof.

From this form of M, it is evident that it has rank $mn - 1$ over $\mathbb{F}_2$. (This also follows from Proposition 5.4.) Thus $\det C \leq mn - 1$.

Let V be the row span of M^T over $\mathbb{F}_2$. Then $\dim V = mn - 1$. The map $\tau : V \to C$ is defined by $\tau : v = (v_1, \ldots, v_{mn}) \mapsto (v_1, \ldots, v_{mn})M$, so that $V\tau = C$ and

$\dim C + \dim\ \ker(\tau) = \dim V = mn - 1$. A vector v is in the kernel if and only if $v \in V$ and $vM = \mathbf{0}$, and since $\jmath M = \mathbf{0}$, we need determine when $\jmath \in V$.

Clearly V is spanned by vectors of weight 2, so V is an even weight code. Thus if mn is odd then $\jmath \notin V$, and $\dim C = mn - 1$. From the form of M^T as described above, each vertex of K_n^m is adjacent to $n(m-1)$ vertices, so the number of entries in each column is $n(m-1)$. Adding all the rows will give $\jmath$ if $n(m-1)$ is odd, i.e. if n is odd and m is even, so in this case $\dim C = mn - 2$.

If n is even then if P is a point in the i^{th} component Λ_i, adding all the rows (edges) corresponding to edges through P and the points of Λ_j will give the incidence vector v^{Λ_j} of weight n. Thus $\jmath = \sum_{i=1}^{m} v^{\Lambda_i} \in V$, and $\dim C = mn - 2$. ■

We give an illustration for M for $m = n = 3$, where

$\Lambda_1 = \{(1,1),(1,2),(1,3)\}$, $\Lambda_2 = \{(2,1),(2,2),(2,3)\}$, $\Lambda_3 = \{(3,1),(3,2),(3,3)\}$:

$$\left[\begin{array}{ccc|ccc|ccc|ccc|ccc|ccc|ccc}
1&1&1&1&1&1&&&&&&&&&&&&&&&&&\\
&&&&&&1&1&1&1&1&1&&&&&&&&&&&\\
&&&&&&&&&&&&1&1&1&1&1&1&&&&&&\\
\hline
1&&&&&&1&&&&&&1&&&&&&1&1&1&&&\\
&1&&&&&&1&&&&&&1&&&&&&1&1&1&&\\
&&1&&&&&&1&&&&&&1&&&&&&1&1&1\\
\hline
&&&1&&&&&&1&&&1&&&1&&&1&&&1&&\\
&&&&1&&&&&&1&&&1&&&1&&&1&&&1&\\
&&&&&1&&&&&&1&&&1&&&1&&&1&&&1
\end{array}\right]$$

We need the following lemma in order to establish information sets for the codes:

Lemma 5.3. *Let $A_\ell = I_\ell + J_\ell$ with entries in $\mathbb{F}_2$, where $\ell \geq 1$, I_ℓ and J_ℓ are the $\ell \times \ell$ identity and all-one matrices, respectively. For any positive integers j and k let*

$$M_{\ell,k} = \left[\begin{array}{cccc|ccc} & & & & 1 & \dots & 1 \\ & A_\ell & & & 0 & \dots & 0 \\ & & & & \vdots & \vdots & \vdots \\ & & & & 0 & \dots & 0 \\ \hline 1 & 0 & \dots & 0 & & & \\ 1 & 0 & \dots & 0 & & A_k & \\ \vdots & \vdots & \vdots & \vdots & & & \\ 1 & 0 & \dots & 0 & & & \end{array}\right] \quad \text{and } M^*_{\ell,k} = \left[\begin{array}{cccc|ccc} & & & & 1 & \dots & 1 & 0 \\ & A_\ell & & & 0 & \dots & 0 & 0 \\ & & & & \vdots & \vdots & \vdots & \vdots \\ & & & & 0 & \dots & 0 & 0 \\ \hline 1 & 0 & \dots & 0 & & & & 1 \\ 1 & 0 & \dots & 0 & & A_k & & 0 \\ \vdots & \vdots & \vdots & \vdots & & & & \vdots \\ 1 & 0 & \dots & 0 & & & & 0 \\ \hline 0 & 0 & \dots & 0 & 1 & 0 & \dots & 0 \end{array}\right].$$

If ℓ and k are both even or both odd then $\det M_{\ell,k} = 1$. *Furthermore,* $\det M^*_{\ell,k} = \det M_{\ell,k-1}$.

Proof: Notice that $\det A_\ell = 1$ for ℓ even and that for ℓ and k both even $\det M_{\ell,k} = \det A_\ell \det A_k = 1$. Simple row and column manipulation lead to the remaining results. ■

We use this lemma to establish the information sets for the codes.

Proposition 5.4. *Let C be the binary code of the graph $L_m(n)$ where $n \geq 2$ and $m \geq 3$. Let $\mathcal{I}$ be the set*

$$\{(1,1:i,j) \mid 2 \leq i \leq m, 1 \leq j \leq n\} \cup \{(1,i:2,1) \mid 2 \leq i \leq n\} \setminus \{(1,1:m,n)\}$$

and $\mathcal{I}^* = \mathcal{I} \cup \{(1,2:m,n)\}$. Then $\mathcal{I}$ is an information set for C if mn is even, and $\mathcal{I}^*$ is an information set for mn odd.

Proof: Arrange the vertices (points) of an adjacency matrix for $L_m(n)$ in the following order:

$$(1,1:2,1),(1,1:2,2),\ldots,(1,1:2,n),(1,1:3,1),\ldots,(1,1:m,n-1)$$

followed by

$$(1,2:2,1),(1,3:2,1),\ldots,(1,n:2,1),(1,2:m,n),$$

and any ordering for the remaining points. Writing $\ell = mn-n-1$ and $k = n-1$, the upper left $\ell+k+1 = mn-1$ part of the adjacency matrix has the form $M^*_{\ell,k}$ of the lemma. We need to show that if mn is even, then $\det M_{\ell,k} = 1$ and if mn is odd then $\det M^*_{\ell,k} = 1$.

If mn is even, then if n is even, $\ell = mn-n-1$ is odd, and $k = n-1$ is odd, so by the lemma, $\det M_{\ell,k} = 1$. If mn is even and n is odd, then $\ell = mn-n-1$ is even and $k = n-1$ is even and again we have $\det M_{\ell,k} = 1$.

If mn is odd then both n and m are odd, and ℓ is odd. Then $\det M^*_{\ell,k} = \det M_{\ell,k-1} = 1$ since ℓ and $k-1$ are odd. This proves that $\mathcal{I}$ or $\mathcal{I}^*$ are information sets. ■

As an example, if $m = n = 3$ then $\ell = 5$ and $k = 2$ and ordering the vertices $(1,1:2,1),(1,1:2,2),(1,1:2,3),(1,2:3,1),(1,1:3,2),(1,2:2,1),(1,3:2,1),(1,2:3,3)$, the top 8×8 part of the adjacency matrix is

$$M^*_{5,2} = \left[\begin{array}{ccccc|cc|c} 0&1&1&1&1&1&1&0\\ 1&0&1&1&1&0&0&0\\ 1&1&0&1&1&0&0&0\\ 1&1&1&0&1&0&0&0\\ 1&1&1&1&0&0&0&0\\ \hline 1&0&0&0&0&0&1&1\\ 1&0&0&0&0&1&0&0\\ \hline 0&0&0&0&0&1&0&0 \end{array}\right].$$

We now find the minimum weight of the codes, and look first at the dual code.

Proposition 5.5. *Let C be the binary code of $L_m(n)$ for $n \geq 2$, $m \geq 3$. Then $C^\perp$ contains all vectors of the form $v^{\{P,Q,R\}}$ where $P = (a, b : c, d)$, $Q = (c, d : e, f)$ and $R = (a, b : e, f)$ and has minimum weight 3.*

Proof: Let $w = v^{\{(a,b:c,d),(c,d:e,f),(a,b:e,f)\}} = w(a, b; c, d; e, f)$. We need to show that $\langle w, v^B \rangle = 0$ for all blocks B. Suppose $B = \overline{(a, b : c, d)}$. Then it is clear that v^B meets w twice. Similarly for the other points of w. If $B = \overline{(a, b : x, y)}$ where $(x, y) \neq (c, d), (e, f)$, then again it meets twice. Any block $\overline{(x, y : r, t)}$ where (x, y) and (r, t) are none of $(a, b), (c, d), (e, f)$, will not meet w at all. This completes all the cases. That there can be no vectors of smaller weight follows from an easy argument. ■

Note: The words of weight 3 in the dual code of C are precisely as described in the proposition, except for the case $n = 2$ and $m = 3$, when extra words of weight 3 are present, e.g. $v^{\{(1,1:2,2),(1,2:3,2),(2,1:3,1)\}}$.

We will use the notation

$$w(a, b; c, d; e, f) = v^{\{(a,b:c,d),(c,d:e,f),(a,b:e,f)\}} = v^{(a,b:c,d)} + v^{(c,d:e,f)} + v^{(a,b:e,f)}. \tag{5.2}$$

Lemma 5.6. *Any vector of $\mathbb{F}_2^v$, where $v = \frac{1}{2}m(m-1)n^2$, that is orthogonal to every vector $w(a, b; c, d; e, f)$ has weight at least $n(m - 1)$.*

Proof: Let w be a non-zero word of $\mathbb{F}_2^v$ and let $P = (1, 1 : 2, 1)$ be in the support of w. Every $w(1, 1; 2, 1; x, y)$ for $x \geq 3$ must meet w again, and thus w has at least another $n(m - 2)$ points in its support. Either $(1, 1 : 3, 1)$ or $(2, 1 : 3, 1)$ must be in the support, and thus by the same argument $w(1, 1; 3, 1; 2, a)$ or $w(2, 1; 3, 1; 1, a)$, $a \neq 1$, must meet w again, yielding a further $n - 1$ points of the form $(1, a : 3, 1)$ or $(2, a : 3, 1)$ and $(1, 1 : 2, a)$ or $(2, 1; 1, 1)$ $a \neq 1$, that are not already counted. Thus we get at least $1 + n(m - 2) + n - 1 = n(m - 1)$ for the weight of w. ■

Proposition 5.7. *If C is the binary code of $L_m(n)$, $n \geq 2$, $m \geq 3$, then C has minimum weight $n(m-1)$ if mn is odd, and $2n(m-1)-2$ if mn is even.*

Proof: If mn is odd then $C = C_M$, where M is the vertex/edge matrix of the graph K_n^m as described in Proposition 5.2. Every row of M has weight $n(m-1)$, the valency of K_n^m, yielding thus minimum words of C.

For mn even, C has codimension 2 in C_M and is spanned by the the sums of rows of M corresponding to adjacent vertices. The vector w described in Lemma 5.6 has support at positions $(1,1:x,y)$ or $(2,1:r,s)$, where $x \neq 1, r \neq 2$, and since $w + v^{\overline{(1,1:2,1)}} \in C_M$, $w = \sum_{(x,y),x\neq 1} v^{(1,1:x,y)}$ or $w = \sum_{(x,y),x\neq 2} v^{(2,1:x,y)}$, and thus be the row of M corresponding to the vertex $(1,1)$ or $(2,1)$ of K_n^m. This shows that in the even case, since $C \subset C_M$, the minimum weight is the block size, i.e. the valency of $L_m(n)$, viz. $2n(m-1)-2$. ■

Note: In the mn even case the code is spanned by minimum weight vectors that are incidence vectors, v^B, of blocks of the design.

5.3 PD-sets

The automorphism group of $L_m(n)$ is $G = S_n \wr S_m$ where S_r denotes the symmetric group of degree r. Thus $G = H \rtimes K$ where $H = S_n \times \ldots \times S_n$, m terms in the product, and $K = S_m$. Thus G is also an automorphism group of the binary code of $L_m(n)$. We denote the identity element of G by ι, and that of S_n by ι_n, and also use (k,k) (for any integer k) to mean the identity permutation. The action of G on $L_m(n)$ is as follows: for $\tau \in S_m$ and $\sigma = (\sigma_1, \ldots, \sigma_m) \in H$, for $P = (h, i : j, k) \in \mathcal{P}$,

$$P^\tau = (h^{\tau^{-1}}, i : j^{\tau^{-1}}, k),$$

$$P^\sigma = (h, i^{\sigma_h} : j, k^{\sigma_j}),$$

so that $\tau^{-1}\sigma\tau = \sigma^*$, where

$$P^{\sigma^*} = (h, i^{\sigma_{h^\tau}} : j, k^{\sigma_{j^\tau}}).$$

Now we define some special elements of G for our PD-sets:

Definition 5.8. *Let $K = S_m$ and $H = S_n \times \ldots \times S_n$, and $G = HK$, $m \geq 3$, $n \geq 2$.*

- $\tau_a^i = (a, i) \in K$ *where* $i, a \in \{1 \ldots m\}$;
- $K^i = \{\tau_a^i \mid 1 \leq a \leq m\} \subseteq K$ *for* $1 \leq i \leq m$;
- $\sigma_b^j \in H$, *where* $\sigma_b^j[j] = (1, b) \in S_n$ *and* $\sigma_b^j[i] = \iota_n$ *for* $i \neq j$, *where* $1 \leq j \leq m$ *and* $1 \leq b \leq n$;
- $H^j = \{\sigma_b^j \mid 1 \leq b \leq n\} \subseteq H$ *for* $1 \leq j \leq m$;
- $\delta_b^j \in H$, *where* $\delta_b^j[j] = (b, n) \in S_n$ *and* $\delta_b^j[i] = \iota_n$ *for* $i \neq j$, *where* $1 \leq j \leq m$ *and* $1 \leq b \leq n$;
- $D^j = \{\delta_b^j \mid 1 \leq b \leq n\} \subseteq H$ *for* $1 \leq j \leq m$;
- $\sigma_a = \prod_{i=1}^m \sigma_a^i = ((1, a), \ldots, (1, a)) \in H$, *for* $1 \leq a \leq n$;
- $H_n = \{\sigma_a \mid 1 \leq a \leq n\} \subseteq H$.

Each of the sets K^i, H^j, D^j, H_n contain the identity $\imath$ of G, since we use the notation (a, a) to denote $\imath$. For all the following propositions and lemmas we will use the information sets $\mathcal{I}$ and $\mathcal{I}^*$, for mn even or odd, respectively, as found in proposition 5.4

Definition 5.9. *Let $\mathcal{T} = \{P_i = (a_i, b_i : c_i, d_i) \mid 1 \leq i \leq s\}$ be a set of s points in $\mathcal{P}$, where $a_i \neq c_i$. Let $A = \{a_i, c_i \mid 1 \leq i \leq s\}$ and $B = \{b_i, d_i \mid 1 \leq i \leq s\}$. For $a \in \{1 \ldots m\}$, let $\alpha(a) = |\{P_j \mid a = a_j \text{ or } a = c_j\}|$. Similarly, for $1 \leq j \leq n$, let $\beta(j)$ be the number of times j appears in the b_i and d_i positions in $\mathcal{T}$.*

Lemma 5.10. *With notation as in Definition 5.9 and $\mathcal{T}$ a set of s points of $\mathcal{P}$, if $2s \leq km - 1$, (for some integer k), then there is an i such that $\alpha(i) \leq k - 1$. If $2s \leq kn - 1$, (for some integer k), then there is a point j such that $\beta(j) \leq k - 1$.*

Proof: Note that $\sum_{i=1}^m \alpha(i) = 2s$ and if $\alpha(i) \geq k$ (for some k) for all $i \in \{1 \ldots m\}$, then $2s \geq km$. Thus if $2s \leq km - 1$, then there is an i such that $\alpha(i) \leq k - 1$. Similarly $\sum_{j=1}^n \beta(j) = 2s$. If $\beta(j) \geq k$, where $k \geq 1$, for all j, then $2s \geq kn$. If $2s \leq kn - 1$, then there is a point j such that $\beta(j) \leq k - 1$.

We use this argument in the following propositions.

Proposition 5.11. *Let C be the $[2m(m-1), 2m-2, 4m-6]_2$ binary code of $L_m(2)$ for $m \geq 3$, correcting $t = 2m-4$ errors. Taking $\mathcal{I}$ as information set, a PD-set for C is the set*

$$S = K^1 H^1 \{K^2 \cup K^m\} H^m \{\iota, \tau_3^2\}$$

of size less than $16m^2$.

Proof: Notice first that the points of the information set in this case are

$$\{(1, 1 : a, 1), (1, 1 : a, 2) \mid 2 \leq a \leq m\} \cup \{(1, 2 : 2, 1)\} \setminus \{(1, 1 : m, 2).$$

We denote the check setcheck set] by $\mathcal{C}$. From Lemma 5.10 we see that for $t = 2m-4$ errors, i.e. $\mathcal{T}$ having t points, there is an i for which $\alpha(i) \leq 3$.

Note that is $\mathcal{T} \subseteq \mathcal{C}$ then ι_G will take $\mathcal{T}$ to $\mathcal{C}$, and $\iota_G \in S$. Every point$P = (a, b : c, d) \in \mathcal{I}$ has $a = 1$ or $c = 1$.

Suppose first there is an i with $\alpha(i) = 0$. If $i = 1$ then $\mathcal{T} \subseteq \mathcal{C}$; if $i \neq 1$ then $\tau_i^1 \in K^1$ will map $\mathcal{T}$ into $\mathcal{C}$, so K^1 will suffice for these t-sets.

Suppose that $\alpha(i) \geq 1$ for all i and that there is an i for which $\alpha(i) = 1$, and thus we take $\alpha(1) = 1$ by using K^1. Then let $P = (1, b; c, d)$ be the only point in $\mathcal{I} \cap \mathcal{T}$. We need to map $\mathcal{T}$ into $\mathcal{C}$. Looking at the cases, if $P = (1, 1 : c, d)$ then σ_2^1 will map P to $P' = (1, 2 : c, d)$ and fix all the other points of $\mathcal{T}$. If $P' = (1, 2 : 2, 1)$ then τ_3^2 will map P' into $\mathcal{C}$ and not move the other points out of $\mathcal{C}$. Similarly if $P = (1, 2 : 2, 1)$ then τ_3^2 will suffice.

Next suppose that $\alpha(i) \geq 2$ for all i and there exists an i with $\alpha(i) = 2$. Then $2s = 4m - 8 \geq 2m$ gives that $m \geq 4$. Again we suppose $\alpha(1) = 2$, using K^1. Let the two points containing 1 be $P = (1, a : b, c)$ and $Q = (1, u : v, w)$. If $a = u = 1$ then $\sigma_2^1 \in H^1$ will map these to $(1, 2 : b, c)$ and $(1, 2 : v, w)$, and so there are only two main cases to consider.

The other one being $P = (1, 1 : a, b)$ and $Q = (1, 2 : u, v)$. Suppose $P = (1, 2 : 2, 1)$ and $Q = (1, 2 : v, w)$, so $Q \in \mathcal{C}$. Then take $a \neq 1, 2, v$, $a \in \{1 \dots m\}$ (possible because $m \geq 4$), $\tau_a^2 \in K^2$ maps P to $(1, 2 : a, 1)$ and either fixes Q or maps it to $(1, 2 : a, w)$.

If $P = (1, 1 : a, b)$ and $Q = (1, 2 : u, v)$, then we need to map P to $(1, 1 : m, 2)$. Either $\tau_a^m \in K^m$ followed by $\sigma_b^m \in H^m$, or one of these, or ι_g will achieve this; Q can only map to another point of the form $(1, 2 : c, d)$ under these, and should the point be $(1, 2 : 2, 1)$, then τ_3^2 can be used to map it into $\mathcal{C}$, and not move $(1, 1 : m, 2)$ since $m \geq 4$. Thus S will deal with all these cases.

Finally suppose $\alpha(i) \geq 3$ for all i and there exists an i with $\alpha(i) = 3$. Then $2t = 4m - 8 \geq 3m$ gives that $m \geq 8$. Again with K^1 we can ensure that $\alpha(1) = 3$, and using the same argument as in the case of two points, we see that $K^1 H^1$ will reduce the problem to the two cases:

(i) $P = (1, 2 : a, b)$, $Q = (1, 2 : c, d)$, $R = (1, 2 : e, f)$

(ii) $P = (1, 2 : a, b)$, $Q = (1, 2 : c, d)$, $R = (1, 1 : e, f)$

Suppose (i), and suppose $P = (1, 2 : 2, 1)$. Then if $k \neq 1, 2, c, e$ and $k \in \{1 \dots m\}$ (possible because $m \geq 8$), then $\tau_k^2 \in K^2$ will map P to $(1, 2 : k, 1)$ and Q and R will remain in $\mathcal{C}$.

Suppose (ii). Then we need to map R to $(1, 1 : m, 2)$ and this can be achieved as in the case of $\alpha(1) = 2$, with τ_e^m and σ_f^m. This will map P and Q into points of the form $(1, 2 : x, y)$. If one should result in $(1, 2 : 2, 1)$, then τ_3^2 will move this point to $\mathcal{C}$ and not take the other two points out of $\mathcal{C}$.

Thus is all cases $S = K^1 H^1 (K^2 \cup K^m) H^m \{\iota, \tau_3^2\}$ will be a PD-set for the code to correct all $t = 2m - 4$ errors. The size of this set is at most $8m(2m - 1) = 16m^2 - 8m$, since $|K^2 \cup K^m| = 2m - 1$, due to ι being in both of the sets. ■

Corollary 5.12. *For $m, n \geq 3$ and mn even, if C is the binary code of $L_m(n)$, then $K^1H^1\{K^2 \cup K^m\}H^m\{\iota, \tau_3^2\}$ is an s-PD-set for C of size $4m^2n^2$ for $s \leq 2m-1$ using the information set $\mathcal{I}$.*

Proof: This follows from the proposition and is restricted to the even case since our arguments excluded $(1, 2 : m, n)$ being an information point. The rest goes through, since $\alpha(i) \leq 3$ for some i. ■

Proposition 5.13. *Let C be the binary code of the graph $L_m(n)$ for $m, n \geq 3$. Then taking for information set $\mathcal{I}$ (mn even) or $\mathcal{I}^*$ (mn odd), the set $S = K^1H^1H^2D^m$ is an s-PD-set for C, for $s < 3m/2$, of size mn^3.*

In particular, if $n = 3$ and m is odd, then S is a PD-set of size $27m$ for C, a $[\frac{9}{2}m(m-1), 3m-1, 3m-3]_2$ code.

Proof: From Lemma 5.10 we see that for $s < 3m/2$ errors, i.e. $\mathcal{T}$ having s points, there is an i for which $\alpha(i) \leq 2$. Again we use K^1 to ensure that $\alpha(1) \leq 2$, and as in Proposition 5.11, consider the possibilities for $\alpha(1) = 0, 1, 2$.

If $\alpha(1) = 0$ then ι will suffice. Suppose $\alpha(1) = 1$ and let $P = (1, i : a, b) \in \mathcal{T} \cap \mathcal{I}^*$. Then if $i = 1, (a, b) \neq (2, 1)$, use $\sigma_n^1 \in H^1$; if $i \neq 1, (a, b) = (2, 1)$, use $\sigma_n^2 \in H^2$; if $P = (1, 1 : 2, 1)$, use $\sigma_n^1\sigma_n^2 \in H^1H^2$; if $P = (1, 2 : m, n)$, use $\sigma_2^1 \in H^1$. Thus maps in $K^1H^1H^2$ will suffice to map $\mathcal{T}$ into $\mathcal{C}$.

Suppose $\alpha(1) = 2$ and let $P = (1, i : a, b)$ and $Q = (1, j : c, d)$ be in $\mathcal{T}$. First suppose $i = j$; if $i = j = 1$ then $\sigma_n^1 \in H^1$ will map P and Q to points $(1, n : a, b)$ and $(1, n : c, d)$ which are either both in $\mathcal{C}$, or $(a, b) = (2, 1)$, in which case the map $\sigma_e^2 \in H^2$, where $e \neq 1, d$, will map them both into $\mathcal{C}$. Thus $K^1H^1H^2$ suffices so far.

If $i = j \neq 1, 2$, then H^2 will work, as above. If $i = j = 2$, then $P = (1, 2 : a, b)$ and $Q = (1, 2 : c, d)$ and at least one is assumed to be in $\mathcal{I}$. If $P = (1, 2 : 2, 1)$ and $Q \in \mathcal{C}$, then $\sigma_e^2 \in H^2$, where $e \neq 1, d$ will map all the points to $\mathcal{C}$.

If $P = (1, 2 : 2, 1)$ and $Q = (1, 2 : m, n)$ (in the mn odd case), then $\sigma_2^2\delta_1^m \in H^2D^m$ will map all into $\mathcal{C}$. If $P = (1, 2 : m, n)$ and $Q = (1, 2 : m, d)$, then use $\delta_e^m \in D^m$, where $e \neq n, d$. Thus $K^1H^1H^2D^m$ suffices.

If $i \neq j$, then if $i = 1$, $\sigma_k^1 \in H^1$, where $k \neq j$, will map the points to $P = (1, k : a, b)$ and $Q = (1, j : c, d)$, where $1 < k < j \leq n$, say. If $(a, b) = (2, 1)$ then σ_e^2, where $e \neq 1, d$, will map the points to $\mathcal{C}$ provided that $Q \neq (1, 2 : m, n)$. If $Q = (1, 2 : m, n)$ then the map δ_e^m, where $e \neq n, b$ can be used.

This covers all cases, i.e. $K^1H^1H^2D^m$ acts as an s-PD-set.

Note that when $n = 3$ and m is odd, we have a $[9m(m-1)/2, 3m-1, 3m-3]_2$ code that can correct up to $t = (3m-5)/2$ errors. Thus $2t < 3m$ so that the partial PD-set is a full PD-set. ■

Following from the proof of this proposition, we get

Corollary 5.14. *For $m, n \geq 3$ and C the binary code of $L_m(n)$, then using the usual information set $\mathcal{I}$ or $\mathcal{I}^*$,*

- *K^1 is an s-PD-set of size m for C for $s \leq \lceil m/2 \rceil - 1$;*
- *$K^1H^1H^2$ is an s-PD-set of size mn^2 for C for $s \leq m - 1$.*

Proof: The proof is immediate from the proof of the previous proposition, noting that from the earlier discussion, if $2s \leq m - 1$ then $\alpha(i) = 0$ for some i for any set of s points, and if $s \leq m - 1$ then $\alpha(i) \leq 1$ for some i. ■

Now a condition involving the size of n.

Proposition 5.15. *Let C be the binary code of $L_m(n)$ where $m, n \geq 3$. Using the usual information set $\mathcal{I}$ or $\mathcal{I}^*$,*

1. *if $s \leq \lceil n/2 \rceil - 1$, then H_n is an s-PD-set of size n for mn even, and $H_n\{\iota, \delta_1^m\}$ is an s-PD-set of size $2n$ for mn odd;*
2. *if $s \leq n-1$ then $H_nH^1H^2$ is an s-PD-set of size n^3 for mn even, and $H_nH^1H^2\{\iota, \delta_1^m\}$ is an s-PD-set of size $2n^3$ for mn odd;*

3. *if $m = 3$ and $n \geq 3$ is odd, then $H_n H^1 H^2 \{\iota, \delta_1^m\}$ is a PD-set of size $2n^3$ for C, a $[3n^2, 3n-1, 2n]_2$ code.*

Proof: Considering now $\beta(j)$ and Lemma 5.10, for the first condition, $k = 1$, i.e. $s \leq (n-1)/2$. There is a j such that $\beta(j) = 0$ and we can use H_n to map $\mathcal{T}$ to a set of s points for which $\beta(1) = 0$. First take mn even. In this case, since $1 \notin B$, it follows that the set in now in $\mathcal{C}$. Thus H_n will suffice as an s-PD-set.

Now take mn odd, so that $P = (1, 2 : m, n) \in \mathcal{I}$. Using H_n to ensure that $\beta(1) = 0$, if P is in the new s-set, then the map δ_1^m will move all the points into $\mathcal{C}$. This proves the first part of the proposition.

Taking now $k = 2$, if $s \leq n-1$ then we can assume $\beta(1) \leq 1$ using H_n again. If $\beta(1) = 0$, use the same argument as above. Thus now suppose $\beta(1) = 1$. Assuming the set $\mathcal{T}$ is not in $\mathcal{C}$, there is one point of the form $(1, 1 : j, k)$, where $k \neq 1$, or $(1, k : 2, 1)$, where $k \neq 1$, and possibly the point $(1, 2 : m, n)$.

If $(1, 1 : j, k) \in \mathcal{T}$, ($(j, k) \neq (m, n)$ since we assume that the point is in $\mathcal{I}$), then we can find an e, $1 \leq e \leq n$, such that $(1, e : j, k) \notin \mathcal{T}$, since there are n such elements and our set has at most $n-1$ elements. The map $\sigma_e^1 \in H^1$ will map $\mathcal{T}$ to a set with $\beta(1) = 0$, and then δ_1^m can be used if necessary.

If $(1, k : 2, 1) \in \mathcal{T}$, then there is an e such that $(1, k : 2, e) \notin \mathcal{T}$, so the map $\sigma_e^2 \in H^2$ can be used, followed by δ_1^m if necessary.

Finally, if $m = 3$ and n is odd, then C corrects at most $t = n - 1$ errors, so we have a PD-set. ■

Based on computations and similar arguments to those in the propositions, we believe that the codes for all the graphs $L_m(n)$ will have PD-sets for full error correction, although we have only found explicit sets for certain classes.

CHAPTER 6

BINARY SELF-DUAL CODES FROM THE GRAPH Q_n

6.1 Introduction

For $n \geq 2$, the graph with vertices the 2^n vectors of $\mathbb{F}_2^n$ and two vertices adjacent if their coordinates differ in precisely one place, is called the n-cube, denoted by Q_n. In this chapter we examine the binary codes obtained from the row span over $\mathbb{F}_2$ and show that when n is even it is self dual and can be used for permutation decoding. Our main result obtaining 3-PD sets is stated and proved in section (6.3). General properties of the graph Q_n, the symmetric design obtained from it, and its binary codes, are in Section (6.2).

6.2 Binary codes of cubic graphs

For $n \geq 2$ let Q_n denote the n-cube and $\mathcal{D}_n$ the symmetric 1-design obtained by defining the 2^n vertices (i.e.vectors in $\mathbb{F}_2^n$) to be the points $\mathcal{P}$, and a block $\overline{v}$ for every point (vector) v by

$$\overline{v} = \{w | w \in \mathcal{P} \text{ and } w \text{ adjacent to } v \text{ in } Q_n\}.$$

Then $\mathcal{D}_n$ is a $1-(2^n, n, n)$ symmetric design with the property that two distinct blocks meet in zero or two points and similarly any two distinct points are together on zero or two blocks.

We will use the following notation: for $r \in \mathbb{Z}$ and $0 \leq r \leq 2^n - 1$, if $r = \sum_{i=1}^{n} r_i 2^{i-1}$ is the binary representation of r, let $\mathbf{r} = (r_1, \ldots, r_n)$ be the corresponding vector in $\mathbb{F}_2^n$, i.e. point in $\mathcal{P}$.

The complement of $v \in \mathcal{P}$ will be denoted by v_c. Thus $v_c(i) = 1 + v(i)$ for $1 \leq i \leq n$, where $v(i)$ denotes the i^{th} coordinate entry of v. Similarly, for $\alpha \in \mathbb{F}_2$, $\alpha_c = \alpha + 1$. Clearly $v_c = v + \mathbf{2^n - 1}$.

The binary code C_n of the design $\mathcal{D}_n$ is the same as the row span over $\mathbb{F}_2$ of an adjacency matrix for Q_n, and for n even and $n \geq 4$, it is a $[2^n, 2^{n-1}, n]_2$ self-dual code. Before showing this, we show why the case for n odd is not of interest.

Proposition 6.1. *For n odd, the binary code C_n of $\mathcal{D}_n$ is the full space $\mathbb{F}^{2^n}$.*

Proof: For n odd, it can be verified directly that

$$v^{(x_1,\ldots,x_n)} = v^{\overline{(x_1,\ldots,(x_n)_c)}} + \sum_{i=1}^{n-1} v^{\overline{(x_1,\ldots,(x_i)_c,\ldots,x_{n-1},x_n)}}$$

for all choices of $x = (x_1, \ldots, x_n)$. Thus C_n contains all the vectors of weight 1 and is the full space. ■

The automorphism group of the design and of the code contains (properly, for $n \geq 4$) the automorphism group $TS_n = T \rtimes S_n$ of the graph (see [12]), where T is the translation group of order 2^n and S_n is the symmetric group acting on the n coordinate positions of the points $v \in \mathcal{P}$. We will write, for each $w \in \mathcal{P}$, $T(w)$ for the automorphism of C_n defined by the translation on $\mathbb{F}_2^n$ given by $T(w) : v \mapsto v + w$ for each $v \in \mathbb{F}_2^n$. The identity map will be denoted by $\iota = T(0)$. Then $T = \{T(w) \mid w \in \mathcal{P}\}$.

Lemma 6.2. *The group TS_n acts imprimitively on the points of the design $\mathcal{D}_n$ for $n \geq 4$ with $\{v, v_c\}$, for each $v \in \mathbb{F}_2^n$, a block of imprimitivity.*

Proof: We need only show that for $g \in TS_n$, and any $v \in \mathbb{F}_2^n$, $v_c g = (vg)_c$, which will make the set $\{v, v_c\}$ a block of imprimitivity. Clearly TS_n is transitive on points. For $g \in S_n$ the assertion is clear. If g is the translation $T(u)$, where $T(u) : v \mapsto v + u$, then $v_c g = v_c T(u) = v + \mathbf{2^n - 1} + u = vT(u) + \mathbf{2^n - 1} = (vg)_c$. Thus for any $g \in TS_n$ and any $v \in \mathbb{F}_2^n$, $v_c g = (vg)_c$. ■

For each i such that $1 \leq i < n$ let $t_i = (i, n) \in S_n$, i.e. the automorphism of C_n defined by the transposition of the coordinate positions. For $n \geq 4$ let

$$P_n = \{t_i \mid 1 \leq i \leq n-1\} \cup \{\iota\} \tag{6.1}$$

$$T_n = TP_n. \tag{6.2}$$

Since the translation group T is normalized by S_n, elements of the form $T(w)t_iT(u)$ are all in T_n, i.e. $\sigma^{-1}T(u)\sigma = T(u\sigma^{-1})$, so that for transpositions t, $tT(u) = T(ut)t$.

Proposition 6.3. *For n even, $n \geq 4$, C_n is a $[2^n, 2^{n-1}, n]_2$ self-dual code with*

$$\mathcal{I} = [\mathbf{0}, \mathbf{1}, \ldots, \mathbf{2^{n-1} - 3}, \mathbf{2^n - 2}, \mathbf{2^n - 1}]$$

as an information set.

Proof: Using the natural ordering for the points and blocks, the incidence matrix for Q_n has the form

$$B_n = \begin{pmatrix} B_{n-2} & I_{2^{n-2}} & I_{2^{n-2}} & 0 \\ I_{2^{n-2}} & B_{n-2} & 0 & I_{2^{n-2}} \\ I_{2^{n-2}} & 0 & B_{n-2} & I_{2^{n-2}} \\ 0 & I_{2^{n-2}} & I_{2^{n-2}} & B_{n-2} \end{pmatrix} \tag{6.3}$$

where B_{n-2} is the incidence matrix of the graph Q_{n-2}. It is easy to prove that the matrix has rank 2^{n-1} and it can be shown by induction that the minimum weight is n. That the code is self-dual follows from the earlier observation that blocks meet in 0, 2 or n points.

To show that $\mathcal{I}$ is an information set, let B_n^* be the first 2^{n-1} rows of B_n. Clearly B_n^* has rank 2^{n-1} and generates the same code as B_n. We want to switch the column indexed by $\mathbf{2^{n-1} - 2}$ with that indexed by $\mathbf{2^n - 2}$, and the column indexed by $\mathbf{2^{n-1} - 1}$ with that indexed by $\mathbf{2^n - 1}$. Notice that $\mathbf{2^{n-1} - 2} \in \overline{\mathbf{2^{n-1} - 1}}$, so the 2×2 submatrix of B_n^* from the $(2^{n-1} - 2)^{th}$ and $(2^{n-1} - 1)^{th}$ rows and columns has the form $\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$, while

the corresponding 2×2 submatrix from the same rows but the last two columns are just I_2. Thus the column interchanges described will give the information set $\mathcal{I}$. ■

If $\mathcal{I}$ is as in the proposition, the corresponding check set is $\mathcal{C}$. We will write

$$\mathcal{I}_1 = [\mathbf{0}, \mathbf{1}, \ldots, \mathbf{2^{n-1} - 3}] \tag{6.4}$$

$$\mathcal{C}_1 = [\mathbf{2^{n-1}}, \mathbf{2^{n-1} + 1}, \ldots, \mathbf{2^n - 3}] \tag{6.5}$$

$$\mathcal{I}_2 = [\mathbf{2^n - 2}, \mathbf{2^n - 1}] \tag{6.6}$$

$$\mathcal{C}_2 = [\mathbf{2^{n-1} - 2}, \mathbf{2^{n-1} - 1}] \tag{6.7}$$

and

$$a = \mathbf{2^n - 2} = (0, 1, \ldots, 1, 1) \quad , \quad b = \mathbf{2^n - 1} = (1, 1, \ldots, 1, 1) \tag{6.8}$$

$$A = \mathbf{2^{n-1} - 2} = (0, 1, \ldots, 1, 0) \quad , \quad B = \mathbf{2^{n-1} - 1} = (1, 1, \ldots, 1, 0). \tag{6.9}$$

Notice that the points a and b are placed in $\mathcal{I}$ in order to have points and their complements in $\mathcal{I}$ since under any automorphism $g \in TS_n$ of the design, if $vg = w$ then $v_c g = w_c$, by Lemma 6.2. Thus we have $a_c = \mathbf{1}$ and $b_c = \mathbf{0}$, $A_c = \mathbf{1} + \mathbf{2^{n-1}}$, $B_c = \mathbf{2^{n-1}}$, and $v + v_c = b$ for any vector $v \in \mathcal{P}$.

6.3 3-PD-sets

In this section we prove the main result obtaining 3-PD-sets. Since the minimum weight is n, the code cannot correct three errors if $n < 8$. However the proof of the theorem holds for $n = 4, 6$ as well.

Theorem 6.4. *For n even and $n \geq 8$, let*

$$T_n = \{T(w)t_i \mid w \in \mathbb{F}_2^n, 1 \leq i \leq n\},$$

where $T(w)$ is the translation by $w \in \mathbb{F}_2^n$, $t_i = (i, n)$ for $i < n$ is a transposition in the symmetric group S_n, and t_n is the identity map. Then T_n is a 3-PD-set of size $n2^n$ for

the self-dual $[2^n, 2^{n-1}, n]_2$ *code* C_n *from an adjacency matrix for the* n*-cube* Q_n*, with the information set*

$$\mathcal{I} = [\mathbf{0}, \mathbf{1}, \ldots, \mathbf{2^{n-1} - 3}, \mathbf{2^n - 2}, \mathbf{2^n - 1}].$$

Proof: Let $\mathcal{T} = \{x, y, z\}$ be a set of three points in $\mathcal{P}$. We need to show that there is an element in T_n that maps $\mathcal{T}$ into $\mathcal{C}$. We consider the various possibilities for the points in $\mathcal{T}$. If $\mathcal{T} \subseteq \mathcal{C}$ then use ι. Thus suppose at least one of the points is in $\mathcal{I}$ and, by using a translation, suppose that one of the points, say z, is $\mathbf{0}$. If $\mathcal{T} \subseteq \mathcal{I}$, then $T(\mathbf{2^{n-1}})$ will work. Now we consider the other cases.

1. $x \in \mathcal{I}_1$, $y \in \mathcal{C}_1$
   Then there are i_x, i_y such that $2 \leq i_x, i_y \leq n-1$ such that $x(i_x) = y(i_y) = 0$. If $i_x = i_y = i$, then $\mathcal{T}t_i \subseteq \mathcal{I}$, unless $yt_i \in \{A, B\}$, so $t_i T(\mathbf{2^{n-1}})$ will work unless $yt_i \in \{A, B\}$. If $yt_i = A$, then $y(1) = y(i) = 0$, $y(j) = 1$ otherwise. If $x(1) = 0$, then $t_1 T(\mathbf{2^{n-1}})$ will work. If $x(1) = 1$, then take any $j \neq 1, i, n$, and use $T(\mathbf{2^{j-1}})t_i T(\mathbf{2^{n-1}})$. If $yt_i = B$, then $y(i) = 0$ and $y(j) = 1$ otherwise. Here we can take any $j \neq 1, i, n$, and use $T(\mathbf{2^{j-1}})t_i T(\mathbf{2^{n-1}})$.
   If x and y have no common zero, then if $y = x_c$, so that $x + y = b$, we can use $T(x)T(\mathbf{2^{n-1}})$. If $x(i) = y(i) = 1$, where $1 \leq i \leq n-1$, then $t_i T(\mathbf{2^{n-1} - 1})$ can be used.

2. $x \in \mathcal{C}_1$, $y \in \mathcal{C}_\in$
   Since $x \in \mathcal{I}_1$, $x(i) = 0$ for some i such that $2 \leq i \leq n-1$. If there is a j such that $j \neq i$ and $2 \leq j \leq n-1$ with $x(j) = 0$, then $T(\mathbf{2^{i-1} + 2^{n-1}})$ can be used.
   If there is no such j, then either $x(1) = x(i) = x(n) = 0$ and $x(j) = 1$ for $j \notin \{1, i, n\}$, or $x(i) = x(n) = 0$ and $x(j) = 1$ for $j \notin \{i, n\}$. In either case, take $j \neq i$, $2 \leq j \leq n-1$. Then the map $T(\mathbf{2^{j-1} + 2^{n-1}})$ can be used.

3. $x \in \mathcal{I}_2$, $y \in \mathcal{C}_1$

   (a) $x = a$: since $y \in \mathcal{C}_\infty$, there is a j such that $2 \leq j \leq n-1$ with $y(j) = 0$. If $y(i) = 1$ for $i \neq j$ and $1 \leq i \leq n$, or if $y(1) = 0$ and $y(i) = 1$ for $i \neq j$ and $2 \leq i \leq n$, then $T(A)$ will work. If there is an $i \neq j$ such that $y(i) = y(j) = 0$ where $2 \leq i, j \leq n-1$, then $t_j T(\mathbf{2^{n-1}})$ can be used.

   (b) $x = b$: this follows exactly as in the $x = a$ case except that in the first two cases for y use $T(B)$ instead of $T(A)$.

4. $x \in \mathcal{I}_2$, $y \in \mathcal{C}_2$

   (a) $x = a$, $y = A$: use $T(a)t_2 T(\mathbf{2^{n-1}})$.

   (b) $x = a$, $y = B$: use $t_{n-1} T(B)$.

(c) $x = b$, $y = A$: use $t_{n-1}T(B)$.

(d) $x = b$, $y = B$: use $t_1 T(\mathbf{1} + \mathbf{2^{n-1}})$.

5. $x, y \in \mathcal{C}$

(a) $x, y \in \mathcal{C}_1$: if $x + y = B$ then $T(B)$ will work. Otherwise $x(i) = y(i)$ for some i such that $1 \leq i \leq n-1$. Again $T(B)$ will work unless x or y are $(0, \ldots, 0, 1)$ or $(1, 0, \ldots, 0, 1)$. If $x = (0, \ldots, 0, 1)$ then $y(i) = 0$ for some i such that $2 \leq i \leq n-1$. Then $t_i T(\mathbf{2^{n-1}})$ can be used unless $y(j) = 1$ for all $j \neq i$, or $y(1) = y(i) = 0$ and $y(j) = 1$ for $j \neq 1, i$; in these cases $t_i T(\mathbf{2^{i-1}} + \mathbf{2^{n-1}})$ can be used. The same arguments hold if $x = (1, 0, \ldots, 0, 1)$.

(b) $x \in \mathcal{C}_1$, $y \in \mathcal{C}_2$: since $x \in \mathcal{C}_1$, there is a j such that $2 \leq j \leq n-1$ with $x(j) = 0$. Then $t_j T(\mathbf{2^{j-1}} + \mathbf{2^{n-1}})$ can be used.

(c) $x, y \in \mathcal{C}_2$: $T(\mathbf{2^{n-2}} + \mathbf{2^{n-1}})$ will work.

This completes all the cases and proves the theorem. ■

Note that this result also shows that the set T_n is a 2-PD-set for C_n for $n = 6$. However, this set T_n with this information set $\mathcal{I}$ will not give a 4-PD-set, since it is quite easy to verify that the set of four points $\{\mathbf{0}, \mathbf{2}, \mathbf{2^n - 2}, \mathbf{2^{n-1} - 1}\}$ cannot be moved by any element of T_n into the check positions.

The automorphism group of the symmetric 1-design is much larger than that of the graph. In particular, it will contain any invertible $n \times n$ matrix over $\mathbb{F}_2$ with the property that the sum of any two of its rows has weight 2. There are also other, non-linear, automorphisms, of the design, as is indicated by computations with Magma [29].

It is possible to arrange more interchanges so that more instances of a point and its complement in the information set occur. Thus s-PD-sets for $s > 3$ seem possible in general.

CHAPTER 7

FIRST-ORDER REED-MULLER CODES

Reed-Muller codes are one of the oldest and most well-known families of codes. They were introduced in 1954 by D. E. Muller and I. S. Reed. The major advantage of Reed-Muller codes is that they are easy to decode using majority logic decoding algorithm. These codes also can be defined recursively. The major disadvantage of Reed-Muller codes is that they become weak as their length increases. Reed-Muller codes are the simplest examples of the class of geometrical codes, which also includes affine and projective geometry codes.

In this chapter we will define and construct the Reed-Muller codes. Then we use a result due to Key, McDonough and Mavron [16] for determining information positions of generalized Reed-Muller codes and find 2-PD sets and 3-PD sets of first-order Reed-Muller codes. We use the construction and notation given in [1, Chapter 5].

7.1 Construction of RM codes

In this chapter we use F to denote $\mathbb{F}_2$. Let V be a vector space of dimension m over F. All functions from V to F form a vector space over F and we denote this by V^F. We denote a typical vector in V as $\mathbf{v} = (v_1, \ldots, v_m)$. Any function $f(v) = f(v_1, \ldots, v_m)$ which takes on the values 0 and 1 is called a Boolean function. Such a function can be specified by a truth table, which gives the value of f at all of its 2^m arguments. The standard logical operations can be applied to Boolean functions.

$$f + g \equiv f \text{ or } g \text{ but not both}$$

$$fg \equiv f \text{ and } g$$

$$f + g + fg \equiv f \text{ or } g$$

$$1 \ + \ f \ \equiv \ not \ f.$$

Note that $f^2 = f = 1f$ for any Boolean function, where 1 is the identical function. If we choose characteristic functions of V as a basis for F^V, then the dimension of F^V is 2^m. The vector space $V = F^m$ also has the standard basis $\mathbf{e_1}, \mathbf{e_2}, \ldots, \mathbf{e_m}$, where $\mathbf{e_i} = (\underbrace{0, \ldots, 1}_{i}, \ldots, 0)$.

Definition 7.1. *Let V be the vector space of dimension m over $F = \mathbb{F}_2$ and let r satisfy $0 \leq r \leq m$. The r^{th}-order* **Reed-Muller** *code, denoted by $\mathcal{R}(r, m)$ is the subspace of F^V (with basis the characteristic functions on the vectors of V) that consists of all polynomial functions in the x_i of degree at most r, i.e.*

$$\mathcal{R}(r, m) = \left\langle \prod_{i \in I} x_i | I \subseteq \{1, 2, \ldots, m\}, \ 0 \leq |I| \leq r \right\rangle$$

By the linear independence of the functions in $\mathcal{M}$ and the definition we have

$$dim(\mathcal{R}(r, m)) \ = \ \binom{m}{0} \ + \ \binom{m}{1} \ + \ \binom{m}{2} \ + \ \ldots \ + \ \binom{m}{r}$$

Therefore in particular for the first order Reed-Muller code we have

$$dim(\mathcal{R}(1, m)) \ = \ 1 \ + \ m$$

Theorem 7.2. *Let $G(r, m)$ be the generator matrix for $\mathcal{R}(r, m)$. Then*

$$G(r+1, m+1) = \begin{pmatrix} G(r+1, m) & G(r+1, m) \\ 0 & G(r, m) \end{pmatrix}$$

Proof: Refer to [31].

Theorem 7.3. *The Reed-Muller code $\mathcal{R}(r, m)$ has minimum distance 2^{m-r}.*

Proof: By inducton on m.

Hence we could summarize these results for the first-order Reed-Muller codes.

Result 7.4. *The first-order Reed-Muller code $\mathcal{R}(1,m)$ is a $[2^m, 1+m, 2^{m-1}]$ code.*

7.2 PD-sets

In order to find PD-sets for $\mathcal{R}(1,m)$ codes, we need to order the point set in a suitable manner so that the generator matrix is in standard form. We use the following two results due to Key, McDonough and Mavron [16] for generalized Reed-Muller codes.

Theorem 7.5. *Let $V = \mathbb{F}_q^m$ be the vector space of $m-$tuples, for $m \geq 1$, over the finite field $\mathbb{F}_q$ of order q, where $q = p^t$ and p is a prime. Let $\alpha_0, \alpha_1, \ldots, \alpha_{q-1}$ be the elements of $\mathbb{F}_q$ and let*

$$\mathcal{S} = \{[i_1, i_2, \ldots, i_m] \mid i_k \in \mathbb{Z},\ 0 \leq i_q \leq q-1,\ 1 \leq k \leq m\}$$

Let $\leq$ denote the partial order defined on $\mathcal{S}$ by $[i_1, i_2, \ldots, i_m] \leq [j_1, j_2, \ldots, j_m]$ if and only if $i_k \leq j_k$ for all k such that $1 \leq k \leq m$.
let $\chi \subseteq \mathcal{S}$ have the property that $y \in \chi$ if $y \in \mathcal{S}$ and $y \leq x$ for some $x \in \chi$, and let $C = \langle x_1^{i_1} x_2^{i_2} \ldots x_m^{i_m} | [i_1, i_2, \ldots, i_m] \in \chi \rangle$. Then the set of vectors

$$\mathcal{I} = \{(\alpha_{i_1}, \ldots, \alpha_{i_m}) \mid [i_1, i_2, \ldots, i_m] \in \chi\}$$

is an information set for C. In particular, if $\chi = \{[i_1, i_2, \ldots, i_m] \in \mathcal{S} \mid \sum_{k=1}^m i_k \leq \nu\}$, then $\mathcal{I}$ is an information set for the $\nu-$th order generalized Reed-Muller code $\mathcal{R}_{\mathbb{F}_q}(\nu, m)$.

Corollary 7.6. *If p is a prime, the code $\mathcal{R}_{\mathbb{F}_p}(\nu, m)$ has information set*

$$\mathcal{I} = \{(i_1, \ldots, i_m) \mid i_k \in \mathbb{F}_p,\ 1 \leq k \leq m,\ \sum_{k=1}^m i_k \leq \nu\}.$$

So we have the following result for the binary first order Reed-Muller codes $\mathcal{R}(1,m)$.

Result 7.7. *The first-order Reed-Muller code $\mathcal{R}(1,m)$ has the information set $\{\mathbf{e_0}, \mathbf{e_1}, \ldots, \mathbf{e_m}\}$, where $\mathbf{e_i} = (\underbrace{0, \ldots, i}_{i}, \ldots, 0)$ and let $\mathbf{e} = \sum_{i=1}^m \mathbf{e_i}$.*

We will use the following notation. Let $u = (u_1, u_2, \ldots, u_m) \in V$. Then we define

$$|u| = \sum_{i=1}^{m} u_i$$

$$C_i = \{u \in V \mid |u| = i\}$$

Then $V = C_0 \cup C_1 \cup \ldots \cup C_m$ and the information set $\mathcal{I} = C_0 \cup C_1$.

The translation of a vector a by the vector u is defined

$$T_u : V \longrightarrow V \tag{7.1}$$

$$T_u : a \mapsto a + u \tag{7.2}$$

i.e. $aT_u = a + u$. Define the translation group of order 2^n to be $T = \{T_u : u \in V\}$.

Theorem 7.8. *The set $\mathcal{S} = \{T_u \mid u \in C_0 \cup C_m \cup C_{m-1} \cup \{w\}\}$, where $w \in C_{m-2}$ is a fixed element and $m > 4$, is a 2-PD set of size $m + 3$ for $\mathcal{R}(1, m)$.*

Proof: Let $\{a, b\}$ be a set of two coordinates. If both a, b are in the check positions or both of them are in the information positions we could use the two translations $T_{\mathbf{e_0}}$ and $T_{\mathbf{e}}$ respectively to take $\{a, b\}$ into $\mathcal{E}$.

Suppose $a \in C_0 \cup C_1$ and $b \in \mathcal{E}$.

Case 1: If $b \in C_i$, where $2 \leq i \leq m-2$. Then $|a| \leq 1$ and $2 \leq |b| \leq m-2$. Then we could use the translation $T_{\mathbf{e}}$. Since $|aT_{\mathbf{e}}| = |a + e| \geq m - 1$ and $|bT_{\mathbf{e}}| = |b + e| \in \{2, \ldots, m - 2\}$ implies that $aT_{\mathbf{e}},\ bT_{\mathbf{e}} \in \mathcal{E}$.

Case 2: If $b \in C_{m-1}$. Then pick a vector $u \in C_{m-1}$ such that $u \neq b$. Then $|bT_u| = |b + u| = 2$ and $|aT_u| \in \{m - 2, m - 1, m\}$ implies that $aT_u,\ bT_u \in \mathcal{E}$.

Case 3: If $b \in C_m$. Then we use the vector w. Then $|aT_w| \in \{m - 3, m - 2, m - 1\}$ and $|bT_w| = 2$ implies that $aT_w,\ bT_w \in \mathcal{E}$. ■

Theorem 7.9. *The set $\mathcal{S} = \{T_u | u \in C_{m-2} \cup C_{m-1} \cup C_m \cup C_0\}$ for $m > 4$, is a $3-PD$ set of size $\frac{m(m+1)}{2} + 2$, for $\mathcal{R}(1, m)$.*

Proof: Let $\{a, b, c\}$ be a set of three coordinate positions. If all a, b, c are in the check positions or all of them are in the information positions we could use the two translations $T_{\mathbf{e_0}}$ and $T_{\mathbf{e}}$ respectively to take $\{a, b, c\}$ into $\mathcal{E}$.

Case 1: If $a, b \in C_0 \cup C_1$ and $c \in C_k$ for $k \neq 0, 1$

Subcase A: Suppose $k \notin \{m-1, m\}$. Then as $k \neq 0, 1$, we have $2 \leq k \leq m-2$. i.e. $2 \leq |c| \leq m-2$. Take $u = \mathbf{e}$ as the translation. Then

$$|aT_u|,\ |bT_u| \geq m-1 \quad and \quad 2 \leq |cT_u| \leq m-2$$

So $aT_u,\ bT_u,\ cT_u\ \in \mathcal{E}$.

Subcase B: Suppose $k = m-1$. Then $|c| = m-1$. Pick $u \in C_{m-1}$ such that $u \neq c$. Then $|cT_u| = 2$ implies that $cT_u \in \mathcal{E}$. But $|aT_u|,\ |bT_u| \in \{m-2, m-1, m\}$ and so $aT_u,\ bT_u \in \mathcal{E}$.

Subcase C: If $k = m$ then $|c| = m$. Suppose $a = \mathbf{e_i}$ and $b = \mathbf{e_j}$, pick $u \in C_{m-2}$ such that

$$u_i = 0,\ u_j = 0\ and\ u_k = 1\ \forall\ k \neq i, j$$

Then $|aT_u| = m-1,\ |bT_u| = m-1$ and clearly $|cT_u| = 2$. Hence $aT_u,\ bT_u,\ cT_u \in \mathcal{E}$ (Suppose $a = \mathbf{e_0}$ and $b = \mathbf{e_i}$, select $u_i = 0$).

Case 2: If $a \in C_0 \cup C_1$ and $b \in C_k$ and $c \in C_l$, where $k, l \neq 0, 1$

Subcase A: Suppose $k, l \notin \{m-1, m\}$. Then the translation $T_{\mathbf{e}}$ will work.

Subcase B: Suppose $k = m-1$ and $l = m-1$. Then pick $u \in C_{m-1}$ so that $u \neq b$ and $u \neq c$. This is similar to case $3B$

Subcase C: Suppose $k = m-1$ and $l = m$. i.e. $b \in C_{m-1}$ and $c = \mathbf{e}$. Let $b = (b_1, b_2, \ldots, b_m)$ such that $b_j = 0$ and let $a = \mathbf{e_i}$.

If $i \neq j$: Pick $u = (u_1, u_2, \ldots, u_m) \in C_{m-2}$ such that $u_i = 0$ and $u_j = 1$. Then

$$|aT_u| = m-1,\ \ |bT_u| = 3,\ \ |cT_u| = 2\ \ \Rightarrow$$

$$aT_u,\ \ bT_u,\ \ cT_u \in \mathcal{E}$$

If $i = j$: Pick $u = (u_1, u_2, \ldots, u_m) \in C_{m-2}$ such that $u_i = 1$. Then

$$|aT_u| = m-3, \;\; |bT_u| = 3, \;\; |cT_u| = 2 \;\; \Rightarrow \;\; aT_u, \;\; bT_u, \;\; cT_u \in \mathcal{E}.$$

(If $a = \mathbf{e_0}$ then pick $u \in C_{m-2}$ such that $u_j = 1$). ■

Theorem 7.10. *The translation group, T is a 4-PD set of size 2^m for $\mathcal{R}(1,m)$, for $m > 4$.*

Proof: Let $\{a, b, c, d\}$ be a set of four coordinate positions. If all $a, b.c, d$ are information positions or all are check positions then we use the translations $T_{\mathbf{e_0}}$ and $T_{\mathbf{e}}$ respectively.

Case 1: Suppose $a, b, c \in C_0 \cup C_1$ and $d \in C_k$, where $k \neq 0, 1$.

Subcase A: If $k \notin \{m-1, m\}$. Then we use the translation $T_{\mathbf{e}}$ as $|aT_{\mathbf{e}}|, |bT_{\mathbf{e}}|, |cT_{\mathbf{e}}| \geq m-1$ and $2 \leq |dT_{\mathbf{e}}| \leq m-2$, we have $aT_{\mathbf{e}}, bT_{\mathbf{e}}, cT_{\mathbf{e}}, dT_{\mathbf{e}} \in \mathcal{E}$.

Subcase B: If $k = m-1$. Pick $u \in C_{m-1}$ such that $u \neq d$. Then $|dT_u| = 2$, and $|aT_u|, |bT_u|, |cT_u| \in \{m-2, m-1, m\}$. So aT_u, bT_u, cT_u and $dT_u \in \mathcal{E}$.

Subcase C: If $k = m$. Pick $u \in C_{m-2}$. Then $|dT_u| = 2$ and $|aT_u|, |bT_u|, |cT_u| \in \{m-3, m-2, m-1\}$. Hence aT_u, bT_u, cT_u and $dT_u \in \mathcal{E}$.

Case 2: Suppose $a, b \in C_0 \cup C_1$ and $c \in C_k, d \in C_l$, where $k, l \notin \{0, 1\}$.

Subcase A: If $k, l \notin \{m-1, m\}$. Then the translation T_e will work.

Subcase B: If $k = m-1$ and $l = m-1$. Pick $u \in C_{m-1}$ such that $u \neq c, d$. Then $|cT_u|, |dT_u| = 2$ and $|aT_u|, |bT_u| \in \{m-2, m-1, m\}$. So aT_u, bT_u, cT_u and $dT_u \in \mathcal{E}$.

Subcase C: If $k = m-1$ and $l = m$. i.e. $c \in C_{m-1}$ and $d \in C_m$ respectively. Then pick $u \in C_2$ such that $supp(u) \cap supp(a) = \emptyset$ and $supp(u) \cap supp(b) = \emptyset$. Then $|aT_u|, |bT_u| \in \{2, 3\}$ and $|cT_u| \in \{m-3, m-1\}, |dT_u| = m-2$. This implies that aT_u, bT_u, cT_u and $dT_u \in \mathcal{E}$.

Case 3: Suppose $a \in C_0 \cup C_1$ and $b \in C_k, c \in C_l, d \in C_p$, where $k, l, p \notin 0, 1$.

Subcase A: If $2 \leq k, l, p \leq m-2$. Then T_e can be used as the translation.

Subcase B: If all $k, l, p = m-1$. Pick $u \in C_{m-1}$ such that $u \neq b, c, d$. Then $|bT_u|, |cT_u|, |dT_u| = 2$ and $|aT_u| \in \{m-2, m-1, m\}$. Therefore aT_u, bT_u, cT_u and $dT_u \in \mathcal{E}$.

Subcase C: If $d \in C_m, c \in C_{m-1}$ and $b \in C_2$. Pick $u \in C_2$ such that $supp(a) \cap supp(u) = \emptyset$. Then $|aT_u| \in \{2,3\}, |bT_u| \in \{2,4\}, |dT_u| = m-2$ and $|cT_u| \in \{m-3, m-1\}$. So aT_u, bT_u, cT_u and $dT_u \in \mathcal{E}$.

Subcase D: If $d \in C_m, c \in C_{m-1}$ and $b \in C_{m-2}$. If $a \neq e_0$, pick $u \in C_1$ such that $u \neq a$. Then $|aT_u| = 2, |bT_u| \in \{m-3, m-1\}, |cT_u| \in \{m-2, m\}$ and $|dT_u| = m-1$. If $a = e_0$ then pick $u \in C_2$ such that $supp(u) \cap supp(b) \neq 2$. Then $|aT_u| = 2, |bT_u| \in \{m-2, m\}, |cT_u| \in \{m-3, m-1\}$ and $|dT_u| = m-2$. So in both cases aT_u, bT_u, cT_u and $dT_u \in \mathcal{E}$. ■

CHAPTER 8

COMPLEXITY OF PERMUTATION DECODING METHOD

8.1 Introduction

In this chapter we will study the complexity of the permutation decoding algorithm. First we state a result given by Limbupasiriporn [14] on the worst-case time complexity of the algorithm. Then we try to reduce the complexity by ordering elements of the PD-set in a certain manner. In particular we look at PD-sets for the codes from lattice graphs (Chapter 3.9, p. 24) and rectangular lattice graphs (Chapter 4, p. 31) and use nested PD-sets. The permutation decoding algorithm is given in (Chapter 2.5, p. 12).

Result 8.1. *The* **worst-case time complexity** *of the permutation decoding algorithm can be expressed in terms of the size m of the s-PD set $\mathcal{S}$, the dimension k and the length n of the code C. The permutation decoding algorithm requires at most $\mathcal{O}(knm)$ operations in a worst-case situation.*

8.2 Codes from lattice graphs

In Chapter 3, p. 24 we found a PD-set of size n^2 for the binary codes of the lattice graphs. By using nested sets, we could reduce the size of the PD-set to s^2, where $0 \leq s \leq t$, if s errors are to be corrected.

Proposition 8.2. *Let C be the $[n^2, 2(n-1), 2(n-1)]$ binary code from the row span of an adjacency matrix for the lattice graph $L_2(n)$. Then by using the set of points $\{(i,n)|2 \leq i \leq n-1\} \cup \{(n,i)|1 \leq i \leq n\}$ as information symbols, the set of automorphisms*

$$S_s = \{((i,n),(j,n))|n-s \leq i,j \leq n\}$$

is an s-error correcting PD-set for any $0 \leq s \leq t$ errors, where $S_0 = \{id\}$.

Proof: When $s = 0$, we use $S_0 = \{id\}$. So assume $s \neq 0$.

Let $\{(a_1, b_1), (a_2, b_2), \ldots (a_2, b_s)\}$ be a set of $1 \leq s \leq t = n-2$ points.

Let $\Omega_1 = \{a_1, a_2, \ldots, a_s\}$ and $\Omega_2 = \{b_1, b_2, \ldots, b_s\}$.

Since $|\Omega_1| \leq s \leq n-2$, there exists atleast $n-s$ elements of $\{1, 2, \ldots, n\}$ not in Ω_1. Similarly there exists $n-s$ elements of {1,2,. . . , n} not in Ω_2.

case(1): Suppose $n \notin \Omega_1$ and $n \notin \Omega_2$.

Then all the errors are in check positions. Then we use the identity element in this case.

case(2): Suppose $n \in \Omega_1$ and $n \notin \Omega_2$.

Let $\Omega_1^* = \Omega_1 \backslash \{n\}$. Then $|\Omega_1^*| \leq s-1$. Since $|\{n-s, \ldots, n-1\}| = s$, there exists $l \in \{n-s, \ldots, n-1\}$ such that $l \notin \Omega_1$. Therefore we could use the map $\sigma := ((l, n), (n, n)) \in S_s$ to move any error.

case(3): Suppose $n \notin \Omega_1$ and $n \in \Omega_2$.

Let $\Omega_2^* = \Omega_2 \backslash \{n\}$. Then $|\Omega_2^*| \leq s-1$. Since $|\{n-s, \ldots, n-1\}| = s$, there exists $k \in \{n-s, \ldots, n-1\}$ such that $k \notin \Omega_2$. Therefore we could use the map $\sigma := ((n, n), (k, n)) \in S_s$ to move any error.

case(4): Suppose $n \in \Omega_1$ and $n \in \Omega_2$.

Since $|\Omega_1^*| \leq s-1$ and $|\Omega_2^*| \leq s-1$, and $|\{n-s, \ldots, n-1\}| = s$, there exists $l, k \in \{n-s, \ldots, n-1\}$ such that $l \notin \Omega_1$ and $k \notin \Omega_2$. Therefore we could use the map $\sigma := ((l, n), (k, n)) \in S_s$ to move any error. ■

8.3 Codes from rectangular lattice graphs

In [Chapter 4, p. 31] we found PD-sets for the rectangular lattice graphs $L_2(m, n)$. When $m+n$ is odd, we found a PD-set of size $m+1$ and when $m+n$ is even the size of the PD-set was m^2+1. The next proposition shows we could reduce the size of these PD-sets, and hence the complexity, by ordering the elements of the PD set.

Proposition 8.3. *If C is the binary code formed by the row space over $\mathbb{F}_2$ of an adjacency matrix for the rectangular lattice graph $L_2(m, n)$ for $2 \leq m < n$, then C is*

- $[mn, m+n-2, 2m]_2$ *for* $m+n$ *even;*
- $[mn, m+n-1, m]_2$ *for* $m+n$ *odd.*

The set $\mathcal{I} = \{(i,n)|1 \leq i \leq m\} \cup \{(m,i)|1 \leq i \leq n-1\}$ *is an information set for* $m+n$ *odd, and* $\mathcal{I}\backslash\{(1,n)\}$ *is an information set for* $m+n$ *even. The sets of automorphisms*

- $S_s = \{((i,m),(i,n))|1 \leq i \leq 2s\} \cup \{id\}$ *for* $m+n$ *odd;*
- $S_s = \{((i,m),(j,n))|1 \leq i \leq m, 1 \leq j \leq s\} \cup \{id\}$ *for* $m+n$ *even*

are s*-error correcting PD-sets for any* s *such that* $0 \leq s \leq t$ *errors.*

Proof: When $m+n$ is odd: For $s=0$ use the $\{id\}$. Assume $1 \leq s \leq t$, so $2s \leq 2t \leq m-1$. Suppose $\mathcal{T}$ be the set of s points

$$\mathcal{T} = \{(a_i, b_i)|1 \leq i \leq s,\ 1 \leq a_i \leq m,\ 1 \leq b_i \leq n\}.$$

Let $\Omega_1 = \{a_i|i = 1 \ldots s\}$ and $\Omega_2 = \{b_i|i = 1 \ldots s\}$, then $|\Omega_1 \cup \Omega_2| \leq 2s$. If $m \notin \Omega_1$ and $n \notin \Omega_2$ then $\mathcal{T} \subseteq \mathcal{E}$. So we could use the $\{id\}$. Thus suppose not both $m \notin \Omega_1, n \notin \Omega_2$, i.e. $\mathcal{T} \cap \mathcal{E} \neq \emptyset$. So $|(\Omega_1 \cup \Omega_2) - \{n, m\}| \leq 2s - 1 \leq m - 2$. Thus there exists an i such that $1 \leq i \leq 2s$ and $i \notin \Omega_1 \cup \Omega_2$ and $\tau = ((i,m),(i,n))$ will satisfy

$$\tau : (a,b) \longmapsto (a,b) \quad \text{if } a \neq m,\ b \neq n$$

$$\tau : (m,b) \longmapsto (i,b) \quad \text{if } b \neq n$$

$$\tau : (a,n) \longmapsto (a,i) \quad \text{if } a \neq m$$

$$\tau : (m,n) \longmapsto (i,i)$$

When $m+n$ is even: If $s=0$ use $\{id\}$ as a Pd-set. So assume $s \neq 0$. Let

$$\mathcal{T} = \{(a_1,b_1),(a_2,b_2),\ldots,(a_s,b_s)\}$$

be a set of $s \leq t = m-1$ points. Define $\Omega_1 = \{a_1,\ldots,a_s\}, \Omega_2 = \{b_1,\ldots,b_s\}$.

Case(1): If $m \notin \Omega_1$ and $n \notin \Omega_2$. Then use the $\sigma = (id)$.

Case(2): If $m \in \Omega_1$ and $n \notin \Omega_2$. Since $|\Omega_1| \leq s \leq t = m-1$, there exists $k \in \{1, \ldots, m\}, k \neq m$ such that $k \notin \Omega_1$. Then use the map $\sigma := ((k,m),(n,n)) \in S_s$.

Case(3): If $m \notin \Omega_1$ and $n \in \Omega_2$. let $\Omega_2^* = \Omega_2 \setminus \{n\}$. Therefore $|\Omega_2^*| \leq s-1$. This implies that, there exists $l \in \{1, \ldots, s\}$ such that $l \notin \Omega_2^*$. Then the map $\sigma := ((m,m),(l,n))$ would work.

Case(4): If $m \notin \Omega_1$ and $n \notin \Omega_2$. Let $\Omega_1^* = \Omega_1 \setminus \{m\}$ and $\Omega_2^* = \Omega_2 \setminus \{n\}$. Then $|\Omega_1^*| \leq s-1$, $|\Omega_2^*| \leq s-1$. Then there exists $l \in \{1, \ldots, s\}, l \notin \Omega_1$ and $k \in \{1, \ldots, s\}, k \notin \Omega_2$. Then we can use the map $\sigma := ((l,m),(k,n))$. ■

CHAPTER 9

CONCLUSION

We studied binary codes from graphs and designs and used these codes for permutation decoding and partial permutation decoding. In this chapter we will give a brief summary of the results we obtained and explorer further extensions.

We used the permutation decoding method on codes from rectangular graphs $L_2(m, n)$ in Chapter 3 and found full error correcting PD-sets for these codes. The size of these PD-sets are larger than that of the Gordon bound. By taking different information sets, one might be able to find minimal PD-sets and reduce the complexity.

In Chapter 4, we introduced binary codes from line graphs of multipartite graphs $L_m(n)$. We found full error correcting PD-sets for certain values of m and n. But we only found partial PD-sets for some m and n. A further study is to find full error correcting PD-sets for all values of m and n.

Binary self dual codes were obtained from the n-cube in Chapter 5. We applied the permutation decoding method and obtained 3-PD sets for these codes. The automorphism group of the graph is known, but an intersting problem is to determine the automorphism group of the code. Also by using different blocks of imprimitivity we think its possible to find PD-sets that correct more errors.

In Chapter 6, we found 4-PD-sets for the first order Reed-Muller codes. An interesting problem is to use permutation decoding method on generalized Reed-Muller codes.

The complexity of the permutation decoding method depends on the size of the PD-set. Arranging the elements of these sets in a particular order reduces the complexity of the algorithm. We used nested sets in Chapter 7 and ordered the PD-set elements that were obtained from the lattice graph and the rectangular lattice graphs.

In this dissertation we did not obtain any minimal PD-sets that satisfy the Gordon bound. In general permutation decoding is useful when the code has a large automorphism group.

INDEX

BIBLIOGRAPHY

1. E. F. Assmus, Jr and J. D. Key. *Designs and their Codes.* Cambridge: Cambridge University Press, 1992. Cambridge Tracts in Mathematics, Vol. 103 (Second printing with corrections, 1993).

2. Ezio Biglieri. *Permutation decoding of group codes.* IEEE Trans. Inform. Theory.

3. A. E. Brouwer and C. J. van Eijl. *On the p-rank of the adjacency matrices of strongly regular graphs.* J. Algebraic Combinatorics, 1:329 - 346, 1992.

4. Herve Chabanne. *Permutation decoding of abelian codes.* IEEE Trans. Inform. Theory, Vol.38, No 6, November 1992.

5. David A. Cox, John B. Little and Don O'Shea. *Ideals, Varieties and Algorithms.* Springer Verlag, second edition, 1996.

6. David Joyner. *Conjectural permutation decoding of some AG codes.* ACM SIGSAM Bulletin, volume 39, No 1(March 2005).

7. W. Fish. *Codes from Uniform Subset graphs and Cycle products.* Ph.D Thesis, University of the Western Cape, 2007.

8. William H. Haemers, Rene Peeters and Jeroen M. van Rijckevorsel. *Binary codes of strongly regular graphs.* Designs, Codes and Crypto., 17: 187-209, 1999.

9. Raymond Hill. *A first course in coding theory.* Oxford University Press, 1986.

10. D. M. Gordon. *Minimal permutation sets for decoding the binary Golay codes.* IEEE Trans. Inform. Theory, 28:541–543, 1982.

11. R. M. F. Goodman and A. D. Green. *Microprocessor controlled permutation decoding of block error correcting codes.* Proceedings of the IEEE international conference on microprocessors in automation and communications, No 41 Sep 1978, pp 365-376.

12. Gordon Royle. *Coloring the cube.* Preprint.

13. W. Cary Huffman. *Codes and groups.* In V. S. Pless and W. C. Huffman, editors, *Handbook of Coding Theory*, pages 1345–1440. Amsterdam: Elsevier, 1998. Volume 2, Part 2, Chapter 17.

14. J. Limbupasiriporn. *Partial permutation decoding for codes from designs and finite geometries.* Ph.D Thesis. Clemson University, 2005.

15. J. D. Key and J. Limbupasiriporn. *Permutation decoding of codes from Paley graphs.* Congr. Numer., 170:143–155, 2004.

16. J. D. Key, T. P. McDonough, and V. C. Mavron. *Information sets and partial permutation decoding of codes from finite geometries.* Finite Fields Appl., To appear.

17. J. D. Key, T. P. McDonough, and V. C. Mavron. *Partial permutation decoding of codes from finite planes.* European J. Combin., 26:665–682, 2005.

18. J. D. Key, J. Moori, and B. G. Rodrigues. *Permutation decoding of binary codes from graphs on triples.* Ars Combin., To appear.

19. J. D. Key, J. Moori, and B. G. Rodrigues. *Permutation decoding for binary codes from triangular graphs.* European J. Combin., 25:113–123, 2004.

20. J. D. Key and P. Seneviratne. *Binary codes from rectangular lattice graphs and permutation decoding.* European J. Combin., 28 (2007), 121–126.

21. J. D. Key and P. Seneviratne. *Permutation decoding of binary codes from lattice graphs.* Discrete Math. (Special issue dedicated to J. Seberry), To appear.

22. J. D. Key and P. Seneviratne. *Permutation decoding for binary codes from the line graphs of complete multipartite graphs.* Discrete Math. To appear.

23. J. D. Key and P. Seneviratne. *Permutation decoding for binary self-dual codes from the graph Q_n where n is even.* To appear.

24. H. J. Kroll and R. Vincenti. *PD-sets for the codes related to some classical varieties.* Discrete Math., 301: 89-105 (September 2005).

25. H. J. Kroll and R. Vincenti. *Antiblocking systems and PD-sets.* Preprint.

26. H. J. Kroll and R. Vincenti. *PD-sets for Grassmannian codes of dimension k6.* Preprint.

27. P. Seneviratne *Permutation decoding and binary codes from lattice graphs.* Masters Project, Clemson University, 2003.

28. J. H Van Lint. *Coding Theory.* Springer, New York, 1980.

29. Wieb Bosma and John Cannon. *Handbook of Magma Functions.* Department of Mathematics, University of Sydney, November 1994. http://magma.maths.usyd.edu.au/magma/.

30. F. J. MacWilliams. *Permutation decoding of systematic codes.* Bell System Tech. J., 43:485–505, 1964.

31. F. J. MacWilliams and N. J. A. Sloane. *The Theory of Error-Correcting Codes.* Amsterdam: North-Holland, 1983.

32. J. Schönheim. *On coverings.* Pacific J. Math., 14:1405–1411, 1964.

33. S. G. S. Shiva, K. C. Fung and H. S. Y. Tan. *On permutation decoding of binary cyclic double error correcting codes of certain lengths.* IEEE Trans. Inform. Theory, 641–643, September 1970.

34. S. G. S. Shiva and K. C. Fung. *Permutation decoding of certain triple error correcting binary codes.* IEEE Trans. Inform. Theory, 444–446, May 1972.

35. J. Wolfmann. *A permutation decoding of the* $(24, 12, 8)$ *Golay code.* IEEE Trans. Inform. Theory, 29 no 5:748–750, 1983.

36. Douglas. B. West *Introduction to graph theory.* Prentice Hall, 2001, second edition.