On Pivot Orbits of Boolean Functions

Constanza Riera * Lars Eirik Danielsen† Matthew G. Parker‡

April 21, 2006

Abstract

We derive a spectral interpretation of the pivot operation on a graph and generalise this operation to hypergraphs. We establish lower bounds on the number of flat spectra of a Boolean function, depending on internal structures, with respect to the $\{I, H\}^n$ and $\{I, H, N\}^n$ sets of transforms. We also construct a family of Boolean functions of degree higher than two with a large number of flat spectra with respect to $\{I, H\}^n$, and compute a lower bound on this number. The relationship between pivot orbits and equivalence classes of error-correcting codes is then highlighted. Finally, an enumeration of pivot orbits of various types of graphs is given, and it is shown that the same technique can be used to classify codes.

1 Introduction

The *pivot* operation on a graph G was used by Arratia, Bollobás and Sorkin [1, 2] to define the *interlace polynomial* $q(G, z)$, as a variant of the Tutte and Tutte-Martin polynomials [4]. It was also described by Van den Nest [20], under the name of *edge-local complementation*. In [17], we related the interlace polynomials of a graph to the spectra of a quadratic Boolean function with respect to a strategic subset of local unitary transforms. Our main motivation in doing this was to establish links between graph theory, cryptography, coding theory, and quantum entanglement.

*C. Riera, L. E. Danielsen and M. G. Parker are with the Selmer Center, Dept. of Informatics, University of Bergen, PB 7800, N-5020 Bergen, Norway. C. Riera is supported by a Norwegian Government Scholarship. E-mail: {riera,larsed,matthew}@ii.uib.no. Web: http://www.ii.uib.no/~{larsed,matthew}

Let the graph $G = (V, E)$, with vertex set, V, and edge set, E, of order n be represented by its $n \times n$ adjacency matrix, Γ. Identify G with a quadratic Boolean function $p(x_0, x_1, \ldots, x_{n-1})$, where $p(\boldsymbol{x}) = \sum_{i<j} \Gamma_{ij} x_i x_j$ [15], i.e., the term $x_i x_j$ occurs in $p(\boldsymbol{x})$ if and only if $ij \in E$. This identification allows us to interpret $q(G, 1)$ as the number of *flat spectra* of $p(\boldsymbol{x})$ with respect to (w.r.t.) the set of transforms $\{I, H\}^n$. In this paper we characterise the pivot operation using *algebraic normal form* (ANF). We also generalise pivot to hypergraphs, and state the (necessary and sufficient) condition that a function of degree higher than two must fulfil in order to allow such an operation. Then we show how the pivot operation on a (hyper)graph can be written as a transform from $\{I, H\}^n$ on the bipolar vector of the function associated to it. We then prove that all (not necessarily all) flat spectra of a quadratic (general) Boolean function, p, w.r.t. $\{I, H\}^n$, can be realised via a series of pivot operations on the graph (hypergraph) associated to p, respectively. We then construct a family of Boolean functions that have a large number of flat spectra w.r.t. $\{I, H\}^n$, and compute this number. We also study the pivot orbit of structures that include a clique and develop lower bounds on the number of flat spectra of a graph w.r.t. $\{I, H\}^n$ and $\{I, H, N\}^n$. It is shown that orbits of bipartite graphs under the pivot operation correspond to equivalence classes of binary linear codes, and that all information sets of a code can be found by pivoting on its associated graph. We also give an enumeration of pivot orbits of all graphs on up to 12 vertices, and of all bipartite graphs on up to 13 vertices.

To the best of our knowledge, the results mentioned above have not appeared in the literature before.

2 Definitions and Notation

Let $H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$ be the Walsh-Hadamard kernel, $N = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & i \\ 1 & -i \end{pmatrix}$, where $i^2 = -1$, be the Negahadamard kernel, and let I the 2×2 identity matrix. Let $s = (s_0, s_1, ..., s_{2^n-1})^t = (-1)^{p(\boldsymbol{x})}$, where $s_{\boldsymbol{i}} = (-1)^{p(\boldsymbol{i})}$ and $p(\boldsymbol{x}) : \mathrm{GF}(2)^n \to \mathrm{GF}(2)$ is a Boolean function. With this notation, p is *bent* [19] if $P = 2^{-\frac{n}{2}} (\bigotimes_{i=0}^{n-1} H)(-1)^{p(\boldsymbol{x})}$ has a *flat* spectrum, or, in other words, if $P = (P_{\boldsymbol{k}}) \in \mathbb{C}^{2^n}$ is such that $|P_{\boldsymbol{k}}| = 1$, $\forall \boldsymbol{k} \in \mathrm{GF}(2)^n$, where '$\otimes$' denotes the tensor product of matrices, also known as the Kronecker product. If the function is quadratic, we associate to it a simple

undirected graph, and in this case a flat spectrum is obtained if and only if Γ, the adjacency matrix of the graph, has maximum rank as a binary matrix. In [15], we generalised this concept, considering not only the Walsh-Hadamard transform $\bigotimes_{i=0}^{n-1} H$, but the complete set of 3^n unitary transforms $\{I, H, N\}^n$, comprising all transforms U of the form $U = \prod_{j\in \boldsymbol{R_I}} I_j \prod_{j\in \boldsymbol{R_H}} H_j \prod_{j\in \boldsymbol{R_N}} N_j$, where the sets $\boldsymbol{R_I}, \boldsymbol{R_H}$ and $\boldsymbol{R_N}$ partition the set of vertices $\{0, \ldots, n-1\}$, and H_j, say, is short for $I \otimes I \otimes \ldots \otimes I \otimes H \otimes I \otimes \ldots \otimes I$, with H in the j^{th} position. For instance, if $n = 4$, $\boldsymbol{R_I} = \{1\}$, $\boldsymbol{R_H} = \{0, 3\}$, and $\boldsymbol{R_N} = \{2\}$, then $U = H \otimes I \otimes N \otimes H$, where U is a 16×16 unitary matrix. The *orbit* of a Boolean function p w.r.t. a set of transforms T comprises all Boolean functions, p', where $s'_{\mathbf{i}} = (-1)^{p'(\mathbf{i})}$, and where s' can be obtained by the application of any $t \in T$ to $s = (-1)^{p(\boldsymbol{x})}$.

In [15, 18] we studied the number of flat spectra of a function w.r.t. $\{I, H, N\}^n$, or in other words the number of unitary transforms $U \in \{I, H, N\}^n$ such that $P_U = (P_{U,\boldsymbol{k}}) \in \mathbb{C}^{2^n}$ has $|P_{U,\boldsymbol{k}}| = 1$, $\forall \boldsymbol{k} \in \mathrm{GF}(2)^n$, where $(P_{U,\boldsymbol{k}}) = U(-1)^{p(\boldsymbol{x})}$. We also considered the number of flat spectra w.r.t. some subsets of $\{I, H, N\}^n$, namely $\{H, N\}^n$ (when $\boldsymbol{R_I} = \emptyset$) and $\{I, H\}^n$ (when $\boldsymbol{R_N} = \emptyset$). We also proved that a quadratic Boolean function will have a flat spectrum w.r.t. a transform $U \in \{I, H, N\}^n$ if and only if a certain modification of its adjacency matrix has maximal binary rank.

As will be explained in the next section, the *pivot orbit* of a (hyper)graph G consists of all graphs obtained by the application of any sequence of *pivot* operations to G. Similarly, the *LC orbit* comprises all graphs obtained by applying *local complementations* to G. In this paper, we will study the pivot orbits of (hyper)graphs, and the subsets of their LC orbits that are associated to the pivot transform.

There are two names for the pivot operation on graphs that are currently in use in the literature, namely *pivot* and *edge-local complementation* (ELC). The name "edge-local complementation" comes from Bouchet's original definition of "local complementation on the edge" in [3] and the name "edge-local complementation" has been used recently by Van den Nest in [20]. The name "pivot" has a long history with respect to Gaussian elimination and, in the context of graphs, would be the operation of ELC on a bipartite graph. A few authors [1, 2, 13, 16] have, since Bouchet, extended the use of "pivot" to apply to all graphs, not just bipartite. We call the ELC operation, "pivot", in this paper, although we acknowledge that "edge-local complementation" is equally valid. Note, however, that in this paper we further

generalise to hypergraphs the applicability of pivot.

3 Pivot

We recall the definition of two graph operations, *local complementation* (LC), also known as *vertex neighbourhood complementation* (VNC), and *pivot*, also known as *edge-local complementation* (ELC).

Definition 1 ([3, 4, 8, 10, 11])**.** Let $G = (V, E)$ be a graph and $i \in V$ be some vertex. $\mathcal{N}(i)$ denotes the neighbourhood of i, i.e., the set of vertices adjacent to i. The action of *local complementation* at vertex i, denoted $\mathrm{LC}(i)$, is defined as the graph transformation obtained by replacing $G[\mathcal{N}(i)]$, i.e., the subgraph induced on the neighbourhood of i, by its complement.

Definition 2 ([1, 2, 3])**.** Given a graph $G = (V, E)$ and an edge $ij \in E$, the action of *pivot* on ij is given by $\mathrm{LC}(i)\mathrm{LC}(j)\mathrm{LC}(i)$, the action of LC at vertex i, then vertex j, then vertex i again.[1] Note that the operations $\mathrm{LC}(j)\mathrm{LC}(i)\mathrm{LC}(j)$ would give exactly the same result.

Definition 3 ([1, 2, 3])**.** Pivoting on $ij \in E$ of $G = (V, E)$ can equivalently be defined as follows. Decompose $V \setminus \{i, j\}$ into four disjoint sets, as visualised in Fig. 1,

- $\mathcal{N}(i) \setminus \mathcal{N}(j)$, the set of vertices adjacent to i, but not to j,
- $\mathcal{N}(j) \setminus \mathcal{N}(i)$, the set of vertices adjacent to j, but not to i,
- $\mathcal{N}(i) \cap \mathcal{N}(j)$, the set of vertices adjacent to both i and j,
- and the set of vertices adjacent to neither i nor j.

For any pair of vertices $\{x, y\}$, where x belongs to one of the first three classes listed above, and y also belongs to one of the first three classes, but a different class than x, 'toggle' the pair $\{x, y\}$, i.e., if $xy \in E$, delete the edge, and if $xy \notin E$, add the edge xy to E. Finally, swap the labels of vertices i and j, or, equivalently, swap the two sets $\mathcal{N}(i)$ and $\mathcal{N}(j)$.[2]

[1] Bouchet's original definition of pivot, called *complementation along an edge* [3], includes a final swapping of the vertices u and v.

[2] In Bouchet's original definition of pivot, this swapping does not occur.

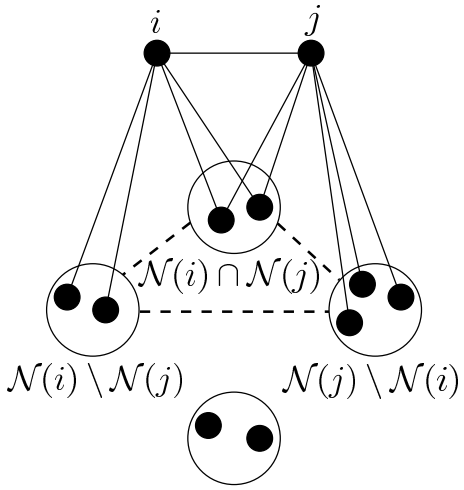

Fig. 1: Visualisation of the Pivot Operation

Let the vertex i of the graph $G = (V, E)$ correspond to the variable x_i in $p(\boldsymbol{x})$, the quadratic Boolean function associated to G. As defined above, $\mathcal{N}(i)$ is the set of vertices that are adjacent to i. We identify $\mathcal{N}(i)$ with the linear Boolean function $\mathcal{N}_i = \sum_{k \in \mathcal{N}(i)} x_k$. Thus $x_i \mathcal{N}_i$ is the quadratic Boolean function corresponding to all edges incident on i. We can now redefine the pivot operation in terms of Boolean functions.

Lemma 1. *Let $p = x_i x_j + x_i \mathcal{N}_i + x_j \mathcal{N}_j + R$ be a quadratic Boolean function, where $\mathcal{N}_i$, $\mathcal{N}_j$, and R are not functions of x_i or x_j. p corresponds to the graph $G = (V, E)$, the linear function $\mathcal{N}_i$ corresponds to the neighbourhood of $i \in V$, $\mathcal{N}_j$ to the neighbourhood of $j \in V$, and the quadratic function R to all edges in E that are incident on neither i nor j. The Boolean function corresponding to the graph obtained by pivoting on the edge $ij \in E$ is*

$$\begin{aligned} p_{iji} &= x_i x_j + x_i \mathcal{N}_j + x_j \mathcal{N}_i + \mathcal{N}_i \mathcal{N}_j + R \\ &= p + (x_i + x_j)(\mathcal{N}_i + \mathcal{N}_j) + \mathcal{N}_i \mathcal{N}_j. \end{aligned} \tag{1}$$

Note that both p and p_{iji} can contain linear terms which may be ignored. We consider p and p_{iji} to be equivalent, since the corresponding graphs are equivalent up to pivot operations.

3.1 A Generalisation to Hypergraphs

Let p be a function of degree at least two. Let $\mathcal{N}_i$ now be the Boolean function comprising all terms which multiply x_i in p, such that $\mathcal{N}_i$ is independent of x_i. Note

that there is no longer a restriction on the degree of $\mathcal{N}_i$.

Definition 4. For Boolean functions, f and g, we say that $g \in f$ or $g \notin f$, when f does or does not depend on g, respectively.

Definition 5. For Boolean functions, f and g, we say that $g \in_t f$ or $g \notin_t f$, when g is or is not a term in the algebraic normal form of f, respectively.

Definition 6. For Boolean functions, f and g, we say that $g \in_m f$ and $g \notin_m f$ when g is or is not a *multiplying term* in f, respectively, where g is a multiplying term in f iff $\exists r$ such that $gr \in_t f$.

Definition 7. Let $p = x_i x_j + q(x_0, \ldots, x_{n-1})$ be a function of degree at least two such that $x_i x_j \notin_m q$. The function p corresponds to the *hypergraph* $G = (V, E)$, and $x_i x_j$ corresponds to the edge $ij \in E$ of degree two. The Boolean function corresponding to the graph obtained by pivoting on $ij \in E$ is defined as

$$\begin{aligned} p_{iji} &= x_i x_j + x_i \mathcal{N}_j + x_j \mathcal{N}_i + \mathcal{N}_i \mathcal{N}_j + R \\ &= p + (x_i + x_j)(\mathcal{N}_i + \mathcal{N}_j) + \mathcal{N}_i \mathcal{N}_j, \end{aligned} \tag{2}$$

where $p = x_i x_j + x_i \mathcal{N}_i + x_j \mathcal{N}_j + R$ as before.

As a visualisation of pivot on hypergraphs, consider Fig. 1, where hyperedges can be added anywhere, with the exception that no edge of degree higher than two can be incident on both i and j. Due to (and equivalently to) the condition on p in Definition 7, $\mathcal{N}_i$ and $\mathcal{N}_j$ are independent of both x_i and x_j, and so the formula is well-defined. If we did not have this condition, the definition would have been ambiguous. When p is quadratic, and the vertices i and j of the corresponding graph are connected, the condition is always fulfilled and the definition is consistent.

Lemma 2. *Let $G = (V, E)$ be a bipartite (hyper)graph. This means that $V = X \cup Y$ such that none of the induced subgraphs $G[X]$ and $G[Y]$ contain any edges. If we interpret $\boldsymbol{x}$ and $\boldsymbol{y}$ as vectors of variables, representing the sets X and Y, then G corresponds to a Boolean function $p = h(\boldsymbol{x}) \cdot g(\boldsymbol{y})$, where $h(\boldsymbol{x})$ and $g(\boldsymbol{y})$ are vectors of Boolean functions of any degree. After pivoting on any permissible edge of G, the resulting (hyper)graph always remains bipartite. Moreover, the sizes of the two partitions will not change under pivot operations.*

Proof. It follows from the definition of a bipartite (hyper)graph that for any edge $ij \in E$, i and j have no common neighbours, and the subgraphs of G induced on $\mathcal{N}(i)$ and $\mathcal{N}(j)$ contain no edges. It follows from lemma 1 for graphs and definition 7 for hypergraphs that the (hyper)graph obtained by pivoting is also bipartite. □

3.2 Pivot in Spectral Terms

In [15], we proved that local complementations on a graph can be realised via the application of successive negahadamard (N) transforms on the bipolar vector, $s = (-1)^p$, of the associated function p. We here show that pivot operations on a (hyper)graph also correspond to certain transformations from the set $\{I, H, N\}^n$.

Let $m : \mathrm{GF}(2)^n \to \mathrm{GF}(2)$. In the following, we shall embed the output of m in the complex numbers by the operation $[m] \in \mathbb{C}$, where $[0] = 0$, and $[1] = 1$.[3]

Let $s = [m(\mathbf{x})](-1)^{p(\mathbf{x})}$ be dependent on all binary variables $x_i,\ 0 \leq i \leq n-1$, where $m = \prod_{k=0}^{u-1} h_k$ and the h_k are Boolean functions in n variables [4] , and p is a Boolean function of degree less or equal than two. In the sequel, expressions of the form $s = c[m](-1)^p$, with $c \in \mathbb{C}$, shall always be written as $s = [m](-1)^p$, i.e. we ignore normalisation coefficients. For an index i, we write $m = rv$, where all the terms in $v = \prod_{k \in V} h_k$, for some $V \subseteq \{0, \ldots, u-1\}$, depend on x_i, and r does not depend on x_i. We denote $p_a = p|_{x_i=a},\ m_a = m|_{x_i=a},\ v_a = v|_{x_i=a}$, for $a \in \mathrm{GF}(2)$. From the conditions above, and by results of [14], we get the following theorems.

Theorem 1. *Let $s = [m](-1)^p$. Then*

$$H_i s = [r(v_0 + v_1)](-1)^{p_0 + v_1(p_0 + p_1 + x_i)} + 2[r v_0 v_1 (p_0 + p_1 + x_i + 1)](-1)^{p_0}. \quad (3)$$

Proof. $s = [m](-1)^p = [(1 + x_i)m_0](-1)^{p_0} + [x_i m_1](-1)^{p_1}$. Applying H_i gives,

$$\begin{aligned} s' = \ & [1 + x_i]([m_0](-1)^{p_0} + [m_1](-1)^{p_1}) + [x_i]([m_0](-1)^{p_0} - [m_1](-1)^{p_1}) \\ = \ & [1 + x_i]([m_0(p_0 + 1)] + [m_1(p_1 + 1)] - [m_0 p_0] - [m_1 p_1]) \\ & + [x_i]([m_0(p_0 + 1)] - [m_1(p_1 + 1)] - [m_0 p_0] + [m_1 p_1]) \end{aligned} \quad (4)$$

By applying the following identity to (4), for Boolean functions A_0, A_1, B_0, B_1,

$$\begin{aligned} [A_0] + [A_1] + [B_0] + [B_1] = \ & [A_0 + A_1 + B_0 + B_1](-1)^{A_0 A_1 + B_0 B_1 + B_0 + B_1} \\ & + 2[(A_0 + A_1 + B_0 + B_1)(A_0 A_1 + B_0 B_1)](-1)^{A_0 + 1}, \end{aligned}$$

[3] Note that $[1 + 1] = [0] = 0$, while $[1] + [1] = 1 + 1 = 2$

[4] Such a factorisation of m is not necessarily unique.

we obtain, after a bit more manipulation, the theorem. [5] □

Theorem 2 (theorem 18 of [14]). *Let $s = [m](-1)^p$. If $x_i \notin m$, then*

$$H_i s = [m \cdot (p_0 + p_1 + x_i + 1)](-1)^{p_0}. \tag{5}$$

Theorem 3 (theorem 20 of [14]). *Let $s = [m](-1)^p$. If $x_i \in m$ and if there exists a factorisation of v such that all $h_k \in_m v$ are linearly dependent on x_i, then*

$$H_i s = [r \cdot (v_0 + v_1)](-1)^{p_0 + h_{z,1}(p_0 + p_1 + x_i)}, \tag{6}$$

where $h_{z,1} = h_z|_{x_i=1}$ and $v_0 + v_1 = \prod_{k \neq j}(h_j + h_k + 1)$, with h_z and h_j chosen arbitrarily among the divisors of v.

Remark. Typically we will choose $z = j$.

Theorem 4. *Let p be a Boolean function that fulfils the condition of definition 7. Then any (hyper)graph obtained by pivoting on the (hyper)graph associated to p corresponds to some member of the set of $\{I, H\}^n$ transforms of p. Concretely, if p_{iji} is the function obtained by pivoting on the edge ij of the (hyper)graph associated with p, then $(-1)^{p_{iji}} = (H_i \cdot H_j)(-1)^p$.*

Proof. Let $p = x_i x_j + x_i \mathcal{N}_i + x_j \mathcal{N}_j + R$, and let $s = (-1)^p$. Then, by theorem 2,

$$s' = H_i s = [x_j + \mathcal{N}_i + x_i + 1](-1)^{x_j \mathcal{N}_j + R}. \tag{7}$$

Now, applying theorem 3, we get

$$s'' = H_j s' = 1 \cdot (-1)^{R + (\mathcal{N}_i + x_i)(\mathcal{N}_j + x_j)} = (-1)^{x_i x_j + x_i \mathcal{N}_j + x_j \mathcal{N}_i + \mathcal{N}_i \mathcal{N}_j + R}, \tag{8}$$

which is what we wanted. By the condition on p, $\mathcal{N}_i$ does not depend on x_j, which ensures that the conditions on m necessary to apply theorem 3 are fulfilled. □

Corollary 1. *Let p be a Boolean function of any degree that satisfies the conditions of definition 7. Then p has a flat spectrum with respect to the transform $U = H_i \cdot H_j$.*

Theorem 5. *Each of the flat spectra of p with respect to the set of transforms $\{H_i \cdot H_j \mid i, j \in \mathcal{Z}_n, i \neq j\}$, identifies an edge ij in the hypergraph associated with p, and therefore can be obtained by pivoting on the hypergraph associated with p at the edge ij.*

[5] Theorem 1 and its proof relate to theorem 17 of [14]. However, we have included a new proof as the proof of theorem 17 was incorrect in [14]. We have also simplified the statement of the theorem.

Proof. We will show that, given some arbitrary spectrum, $H_i(-1)^p$, the only way one can obtain a flat spectrum, $H_j \cdot H_i(-1)^p$, $i \neq j$, is when

$$x_i x_j \in_t p, \quad x_i x_j \notin_m p - x_i x_j.$$

In such a case, theorem 4 states that $H_j \cdot H_i(-1)^p$ always corresponds to a pivot operation on the hypergraph associated to p at the edge ij.

From theorem 2, for arbitrary i,

$$H_i(-1)^p = [p_0 + p_1 + x_i + 1](-1)^{p_0} = [\tilde{m}](-1)^{p_0},$$

for some $\tilde{m}$. In order that $H_j \cdot H_i(-1)^p = (-1)^{p'}$, for some p', we must transform the factor, $[p_0 + p_1 + x_i + 1]$, back to 1. This is trivially possible if $j = i$, but the theorem excludes the case where $i = j$. Let

$$H_j \cdot H_i(-1)^p = [m'](-1)^{p'},$$

for some m' and p', where i and j are arbitrary, $i \neq j$. We wish to choose j such that $m' = 1$. There are three possible scenarios:

- $x_j \notin \tilde{m}$: In this case, from theorem 2, $(p_0 + p_1 + x_i + 1) \in_m m'$ so $m' \neq 1$.
- $x_j \in \tilde{m}$: There are three subcases. Let

  $$p_{00} = p_0|_{x_j=0}, \quad p_{10} = p_1|_{x_j=0}, \quad p_{01} = p_0|_{x_j=1}, \quad p_{11} = p_1|_{x_j=1}.$$

  Considering theorem 1 acting on $[\tilde{m}](-1)^{p_0}$, then m' can be 1 iff one or more of the following three conditions are met:

  $$v_0 + v_1 = 1, \quad v_0 v_1 (p_{00} + p_{01} + x_j + 1) = 0 \tag{9}$$

  $$v_0 + v_1 = 0, \quad v_0 v_1 (p_{00} + p_{01} + x_j + 1) = 1 \tag{10}$$

  $$v_0 + v_1 = 1, \quad v_0 v_1 (p_{00} + p_{01} + x_j + 1) = 1 \tag{11}$$

  As, in this case, $v = p_0 + p_1 + x_i + 1$, $v_0 = v|_{x_j=0}$, $v_1 = v|_{x_j=1}$, then we observe that

  $$v_0 + v_1 = p_{00} + p_{10} + p_{01} + p_{11},$$
  $$v_0 v_1 = (p_{00} + p_{10} + x_i + 1)(p_{01} + p_{11} + x_i + 1).$$

  The three subcases for $x_j \in \tilde{m}$ are:

- $x_i x_j \notin_m p$: In this case $v_0 + v_1 = 0$ so (9) and (11) are not satisfied. Furthermore, $\deg(v_0 v_1(p_{00}+p_{01}+x_j+1)) > 0$ as $v_0 v_1 = p_{00}+p_{10}+x_i+1 = p_{01}+p_{11}+x_i+1$, so $x_i \in v_0 v_1, x_j \notin v_0 v_1$, and $x_i \notin p_{00}+p_{01}+x_j+1$, $x_j \in p_{00}+p_{01}+x_j+1$, so (10) is not satisfied.
- $x_i x_j \in_t p$, $x_i x_j \notin_m p - x_i x_j$: In this case $v_0+v_1 = 1$. Moreover, $p_{00}+p_{10} = p_{01}+p_{11}+1$, so $v_0 v_1 = 0$. Therefore (9) is satisfied.
- $x_i x_j \in_t p$, $x_i x_j \in_m p - x_i x_j$: In this case $\deg(v_0 + v_1) > 0$ so none of (9),(10), or (11) are satisfied.

From the above analysis, $m' = 1$ iff $x_i x_j \in_t p$ and $x_i x_j \notin_m p - x_i x_j$. This is precisely the condition required to ensure pivot at the edge ij on the hypergraph associated to p, as stated by definition 7. □

Theorem 6. *Let p be a quadratic Boolean function over n variables. Then all flat spectra of p with respect to transforms from the set $\{I, H\}^n$, other than the identity, can be obtained via a sequence of pivot operations on the graph associated to p.*

Proof. Consider the following hypothesis:

Let $\mathbf{X}$ be a fixed subset of $\{0, 1, \ldots, n-1\}$, where $|\mathbf{X}| > 2$. Let $U = \prod_{i \in \mathbf{X}} H_i$. Then it is possible for $U(-1)^p$ to be flat, and for $U'(-1)^p$ not to be flat $\forall\, U'$ satisfying $U' = \prod_{i \in \mathbf{Z}} H_i$, where $\mathbf{Z} \subset \mathbf{X}$ and $\mathbf{Z} \neq \emptyset$.

The theorem is proved if the hypothesis is proven false, as $H(-1)^p$ is never flat. If the hypothesis is true for some $\mathbf{X}$, then $\nexists\ i, j \in \mathbf{X}$ such that $H_i \cdot H_j(-1)^p$ is flat. We know, from theorem 4 that, therefore, the set of vertices, $\mathbf{X}$, forms an independent set[6] in the graph, G, associated to p. But $U(-1)^p$ cannot be flat if $\mathbf{X}$ is an independent set in G as, applying H to $(-1)^p$ at all index positions in $\mathbf{X}$ requires $|\mathbf{X}|$ invocations of theorem 2, each of which contributes a new linear factor to m. Therefore the final m cannot be 1 and the hypothesis is false. But, for $|\mathbf{X}| = 2$, we know from theorem 5 that all flat spectra are obtained via pivot operations. It is trivial to show that $U(-1)^p$ is never flat if $|\mathbf{X}| = 1$. □

Lemma 3. *Let p be a Boolean function of any degree over n variables. Then there may exist flat spectra of p with respect to transforms from the set $\{I, H\}^n$, other*

[6] If $\mathbf{X}$ is not an independent set, then there is an edge ij between vertices of $\mathbf{X}$, and thereby we can pivot on it, and $H_i \cdot H_j(-1)^p$ is flat.

than the identity, that cannot be obtained via a sequence of pivot operations on the hypergraph associated to p.

Proof. By example, the Boolean function,

$$\begin{aligned} p(\mathbf{x}) = \; & x_0x_1x_2 + x_0x_1x_3 + x_0x_1x_5 + x_0x_2x_4 + x_0x_2x_5 + x_0x_3x_4 + x_0x_3x_5 + x_0x_4x_5 \\ & + x_1x_2x_3 + x_1x_2x_4 + x_1x_2x_5 + x_1x_3x_4 + x_1x_4x_5 + x_2x_3x_4 + x_2x_3x_5 + x_3x_4x_5, \end{aligned}$$

has two flat spectra w.r.t. the set $\{I, H\}^6$. Apart from the identity transform, $(-1)^p$ is also flat w.r.t. $H \otimes H \otimes H \otimes H \otimes H \otimes H$. Such a flat spectrum cannot be obtained via a series of pivot operations as p does not contain any quadratic terms. □

Remark. The example used in the proof of lemma 3 was taken from an interesting catalogue of *homogeneous bent functions*, as provided by [5].

Let $p : \mathrm{GF}(2)^n \to \mathbb{Z}_4$, $m : \mathrm{GF}(2)^n \to \mathrm{GF}(2)$, and let $s \in \mathbb{C}^{2^n}$ be such that $s = (s_0, s_1, \ldots, s_{2^n-1})^t = [m(\boldsymbol{x})]i^{p(\boldsymbol{x})}$, where $s_{\boldsymbol{j}} = [m(\boldsymbol{j})]i^{p(\boldsymbol{j})}$. Sometimes, for brevity, we write the above as $s = [m]i^p$, when it is clear from the context what we mean. Let $m_a : \mathrm{GF}(2)^n \to \mathrm{GF}(2)$ represent $m_a = m|_{x_j=a}$. Similarly, let $p_a : \mathrm{GF}(2)^n \to \mathbb{Z}_4$ represent $p_a = p|_{x_j=a}$.

Theorem 7. *Let $m : \mathrm{GF}(2)^n \to \mathrm{GF}(2)$ and $p : \mathrm{GF}(2)^n \to \mathbb{Z}_4$. Then,*

$$N_j[m]i^p = \frac{1}{\sqrt{2}}([m_0]i^{p_0} + [m_1]i^{p_1+2x_j+1}). \tag{12}$$

Proof. Without loss of generality, we set $j = n-1$. Then, we can write the complex vector $[m]i^p$ (seen as a $2^n \times 1$ matrix) as

$$[m]i^p = \begin{pmatrix} [m_0]i^{p_0} \\ [m_1]i^{p_1} \end{pmatrix},$$

where $[m_0]i^{p_0}$ and $[m_1]i^{p_1}$ are $2^{n-1} \times 1$ complex matrices. Then,

$$
\begin{aligned}
N_{n-1}[m]i^p &= \begin{pmatrix} N & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & N & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & N \end{pmatrix} \begin{pmatrix} [m_0]i^{p_0} \\ [m_1]i^{p_1} \end{pmatrix} \\
&= \begin{pmatrix} [m(0,\dots,0)]i^{p(0,\dots,0)} + i[m(0,\dots,1)]i^{p(0,\dots,1)} \\ [m(0,\dots,0)]i^{p(0,\dots,0)} - i[m(0,\dots,1)]i^{p(0,\dots,1)} \\ [m(0,\dots,1,0)]i^{p(0,\dots,1,0)} + i[m(0,\dots,1,1)]i^{p(0,\dots,1,1)} \\ [m(0,\dots,1,0)]i^{p(0,\dots,1,0)} - i[m(0,\dots,1,1)]i^{p(0,\dots,1,1)} \\ \vdots \\ [m(1,\dots,1,0)]i^{p(1,\dots,1,0)} + i[m(1,\dots,1)]i^{p(1,\dots,1,1)} \\ [m(1,\dots,1,0)]i^{p(1,\dots,1,0)} - i[m(1,\dots,1,1)]i^{p(1,\dots,1,1)} \end{pmatrix} \\
&= [m_0]i^{p_0} + [m_1]i^{p_1+2x_{n-1}+1}.
\end{aligned}
$$

□

In [15], we proved that local complementation can be realised via the application of successive Ns on the bipolar vector of the function, $s = (-1)^p$. Let D be the set of (unitary) diagonal or anti-diagonal 2×2 matrices. Define $\delta, \gamma \in \{D\}^n$ as $\delta = \frac{\sqrt{2}}{1+i} \prod_{k=l,j} \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix}_k$ and $\gamma = -\prod_{k=l,j} \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}_k$. Then,

Theorem 8. *Let p be a function that fulfils the condition of definition 7. Then the local complementation of its associated (hyper)graph, seen as a weighted (hyper)graph, lies in the orbit of p w.r.t. $\{I, H, N\}^n$ to within a post-multiplication by a tensor product of members of D. Concretely, if p_l, p_{jl}, and p_{ljl} are the functions obtained by local complementations on the vertices l, j, then l again, of the (hyper)graph associated with p, then*

$$
\begin{aligned}
i^{p_l} &= \delta N_l (-1)^p, \\
i^{p_{jl}} &= \delta N_j \delta N_l (-1)^p, \\
(-1)^{p_{ljl}} &= \gamma \delta N_l \delta N_j \delta N_l (-1)^p.
\end{aligned} \tag{13}
$$

Remark. We do not distinguish between l and j, so one can obtain the hypergraphs associated to the functions, p_l, p_j, p_{jl}, p_{lj}, and p_{ljl}, via local complementation. Note that $p_{jlj} = p_{ljl}$.

Proof. Let $p = x_l x_j + x_l \mathcal{N}_l + x_j \mathcal{N}_j + R$, and $s = (-1)^p$. Let $\mathcal{N}_l = \sum_{r=0}^{\rho} u_r$, and $\mathcal{N}_j = \sum_{t=0}^{\tau} v_t$, (note that they are not necessarily linear). Then, applying theorem 7[7] (or by the results on [15]), $N_l s = \frac{1+i}{\sqrt{2}} i^{p'}$, where $p' : \mathrm{GF}(2)^n \to \mathbb{Z}_4$, with explicit formula[8]

$$p' = 2\left(p(x) + x_j \sum_{r=0}^{\rho} u_r + \sum_{r \neq s} u_r u_s\right) + 3\left(x_l + x_j + \sum_{r=0}^{\rho} u_r\right). \tag{14}$$

Applying δ to $N_l s$, we get $s' = \delta N_l s = i^{p_l}$, where

$$p_l = 2\left(p(x) + x_j \sum_{r=0}^{\rho} u_r + \sum_{r \neq s} u_r u_s\right) + 3\sum_{r=0}^{\rho} u_r. \tag{15}$$

This is the result of the action of LC(l). Now we apply LC(j); that is, we first apply N_j to s'. By theorem 7, the result is $N_j s' = \frac{1+i}{\sqrt{2}} i^{p''}$, where $p'' : \mathrm{GF}(2)^n \to \mathbb{Z}_4$, with explicit formula

$$\begin{aligned} p'' = 2\Bigg(& x_l x_j + x_l \sum_{t=0}^{\tau} v_t + x_j \left(\sum_{r=0}^{\rho} u_r + \sum_{t=0}^{\tau} v_t\right) \\ &+ \sum_{t \neq u} v_t v_u + \sum_{r,t} u_r v_t + \sum_{r=0}^{\rho} u_r + R\Bigg) \\ &+ 3(x_l + x_j + \sum_{t=0}^{\tau} v_t) \end{aligned} \tag{16}$$

Then we apply δ to $N_j s'$ to get $s'' = \delta N_j s' = i^{p_{lj}}$, where

$$\begin{aligned} p_{lj} = 2\Bigg(& x_l x_j + x_l \sum_{t=0}^{\tau} v_t + x_j \left(\sum_{r=0}^{\rho} u_r + \sum_{t=0}^{\tau} v_t\right) \\ &+ \sum_{t \neq u} v_t v_u + \sum_{r,t} u_r v_t + \sum_{r=0}^{\rho} u_r + R\Bigg) + 3\sum_{t=0}^{\tau} v_t \end{aligned} \tag{17}$$

Now we apply LC(l) again; that is, we first apply N_l to s''. By theorem 7, the result is $N_l s'' = \frac{1+i}{\sqrt{2}} i^{p'''}$, where $p''' : \mathrm{GF}(2)^n \to \mathbb{Z}_4$, with explicit formula

$$\begin{aligned} p''' = 2\Bigg(& x_l x_j + x_l \sum_{t=0}^{\tau} v_t + x_j \sum_{r=0}^{\rho} u_r \\ &+ \sum_{r,t} u_r v_t + \sum_{r=0}^{\rho} u_r + \sum_{t=0}^{\tau} v_t + R\Bigg) + 3(x_l + x_j) \end{aligned} \tag{18}$$

[7]One can lift the Boolean function p to a function $q : \mathrm{GF}(2)^n \to \mathbb{Z}_4$, with $q(\boldsymbol{x}) = 2p(\boldsymbol{x})$.

[8]We denote as $\lambda_0\phi_0 + \lambda_1\phi_1$ or, more generally, as $\sum \lambda_i \phi_i$, with $\lambda_i \in \mathbb{Z}_4$ and ϕ_i Boolean functions, the result of embedding the output of the ϕ_i's into $\mathbb{Z}_4$, multiplying them by a scalar $\lambda_i \in \mathbb{Z}_4$, and then adding the result mod 4. For instance, for $x_0 = x_1 = 1$, we have $2[x_0 + x_1] + 3x_1 + 2 = 1$.

Then we apply δ to $N_l s''$ to get $s''' = \delta N_l s'' = (-1)^{p'_{ljl}}$, where

$$p'_{ljl} = x_l x_j + x_l \sum_{t=0}^{\tau} v_t + x_j \sum_{r=0}^{\rho} u_r + \sum_{r,t} u_r v_t + \sum_{r=0}^{\rho} u_r + \sum_{t=0}^{\tau} v_t + R \tag{19}$$

If we now apply γ to s''', we get

$$p_{ljl} = x_l x_j + x_l \sum_{t=0}^{\tau} v_t + x_j \sum_{r=0}^{\rho} u_r + \sum_{r,t} u_r v_t + R, \tag{20}$$

which is, by definition 7, the formula for pivot on the hypergraph associated to p. Note that this gives as well an alternative proof of theorem 4: Let $d = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix}$, and let $d' = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}$. We see that we have applied:

- In position l: $d'dNddN = d' \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} H = H$
- In position j: $\frac{-1}{e^{3\pi i/4}} d'ddNd = (-1)d' \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} H = H$
- Remaining positions: I

□

4 Number of Flat Spectra w.r.t. $\{I, H\}^n$

In this section, we first study the behaviour of a graph that contains a certain inner structure, namely a *clique*. Then we give bounds on the number of flat spectra of graphs based on their subgraphs, specifically in the case where some of the subgraphs are cliques.

In order to provide some context for the results, we first state the results of some computer experiments. Table 1 shows the expected number of flat spectra w.r.t. $\{I, H\}^n$ for a random Boolean function, and for a random Boolean function of degree ≤ 2. Table 1 demonstrates, empirically [9] that, for n large enough, the expected number of flat spectra w.r.t. $\{I, H\}^n$ for a random Boolean function, and for a random Boolean function of degree ≤ 2, respectively, is 1.0 and approximately

[9] Exhaustive search for random, $n \leq 4$, and random quad., $n \leq 7$, otherwise 100000 samples were taken.

Table 1: Average # flat spectra w.r.t. $\{I, H\}^n$

	n							
	2	3	4	5	6	7	8	9
random	1.500	1.750	1.390	1.039	1.000	1.000	1.000	1.000
random quad	1.500	2.500	4.438	8.188	15.486	29.726	57.918	113.227

2^{n-2}, respectively. The structures and constructions considered in this section will be seen to produce (hyper)graphs with relatively high numbers of flat spectra w.r.t. $\{I, H\}^n$, in comparison to the average.

4.1 Cliques

The *complete graph* or *clique* on n vertices corresponds to the Boolean function $p = \sum_{0 \leq i < j \leq n-1} x_i x_j$.

Lemma 4 ([18]). *The Boolean function corresponding to the complete graph on n vertices has 2^{n-1} flat spectra w.r.t. $\{I, H\}^n$, and maximises over the set of Boolean functions of n variables the number of flat spectra w.r.t. $\{I, H\}^n$.*

We now study the behavior of a graph that contains a *clique*, i.e., a complete subgraph. We consider three cases, depending on the positions of the vertices a and b, where we pivot on the edge ab. Let C_r be the clique on r vertices contained in the graph. We denote by $\mathcal{N}(a)$ and $\mathcal{N}(b)$ the neighbourhoods of a and b.

- $a, b \in C_r$: The clique remains invariant.
- $a \in C_r$, $b \notin C_r$: Let m be the number of variables of C_r that are in $\mathcal{N}(a) \cap \mathcal{N}(b)$. Then, C_r splits and we get the cliques C_{r-m}, C_{m+2}, connected just by b. Moreover $a \notin C_{r-m}$, $b \in C_{r-m}$, and $a, b \in C_{m+2}$.
  - Particular case: Two connected cliques: $a \in C_{r_a}$, $b \notin C_{r_a}$, and $b \in C_{r_b}$. Let m_a be the number of vertices of the clique C_{r_a} that are in $\mathcal{N}(a) \cap \mathcal{N}(b)$, and m_b the number of vertices of the clique C_{r_b} that are in $\mathcal{N}(a) \cap \mathcal{N}(b)$. Then, both cliques split and we get the cliques $C_{r_a - m_a}$, C_{m_a+2}, $C_{r_b - m_b}$, and C_{m_b+2}.

- $a, b \notin C_r$: In this case, C_r remains invariant, independently of whether a or b are connected to it or not.

4.2 Bounds on the Number of Flat Spectra

We give lower bounds on the number of flat spectra w.r.t. $\{I, H\}^n$ and $\{I, H, N\}^n$ depending on internal structures:

Lemma 5. *Consider an unconnected graph G, composed of two connected components, G_1 and G_2. The number of flat spectra of G w.r.t. $\{I, H\}^n$, K_{IH}, has as lower bound: $K_{IH}(G) \geq K_{IH}(G_1) \cdot K_{IH}(G_2)$*

Corollary 2. *If we decompose an unconnected graph into connected its components $G_1, \ldots, G_t$, then $K_{IH}(G) \geq \prod_{i=1}^{t} K_{IH}(G_i)$. For instance, if we can decompose the graph into cliques $C_{r_1}, \ldots, C_{r_t}$, of respective sizes $r_1, \ldots, r_t$, then $K_{IH}(G) \geq \prod_{i=1}^{t} 2^{n_i - 1}$.*

Lemma 6. *Consider the number of flat spectra w.r.t. $\{I, H, N\}^n$. If we decompose an unconnected graph into connected components $G_1, \ldots, G_t$, then we have that $K_{IHN}(G) \geq \prod_{i=1}^{t} K_{IHN}(G_i)$.*

Corollary 3. *The maximum clique size, n_x, of any member of the pivot orbit of G is upper-bounded by $n_x \leq \lfloor \log_2(K_{IH}) \rfloor$.*

5 A Construction of Boolean Functions with High Number of Flat Spectra

We now design a family of Boolean functions in n variables of degree less than or equal to $\max\{t, 2\}$, where $0 \leq t \leq n-1$, whose members have a large number of flat spectra w.r.t. $\{I, H\}^n$. Let

$$f^{n,t} = \sum_{i=0}^{t-1} \sum_{j=t}^{n-1} x_i x_j + \sum_{i=t}^{n-2} \sum_{j=i+1}^{n-1} x_i x_j + a(x_0, x_1, \ldots, x_{n-1}), \tag{21}$$

where $\deg(a) \leq 1$. We then define the family $\mathcal{F}^{n,t}$,

$$\mathcal{F}^{n,t} = \{f^{n,t} + h(x_0, x_1, \ldots, x_{t-1})\}, \tag{22}$$

where h is an arbitrary Boolean function of t variables.

Conjecture 1. Let $f \in \mathcal{F}^{n,t}$. Then the pivot orbit of f is contained in $\bigcup_{k=0}^{n-1} \mathcal{F}^{n,k}$.

Theorem 9. *Let $f \in \mathcal{F}^{n,t}$. Then the number of flat spectra of f w.r.t. $\{I, H\}^n$ is at least $(t+1)2^{n-t-1}$, where the bound is tight if f has degree t.*

Proof. Let $f \in \mathcal{F}^{n,t}$. Then it fulfils the condition of definition 7 for every edge ij such that $t \leq i, j \leq n$. We showed in Section 4.1 that pivoting on any of these edges leaves the clique invariant. This means that the number of flat spectra of f will be at least the number of times we can pivot on the clique on the last $n-t$ variables times the number of times we can pivot on the complete bipartite graph $\sum_{i=0}^{t-1}\sum_{j=t}^{n-1} x_i x_j$ (not counting repetitions), plus the identity transform. The number of times we can pivot on the clique of the hypergraph is the same as the number of times we can pivot on a clique of size $n-t$. By lemma 4, this number is 2^{n-t-1}. We can pivot on each edge of the complete bipartite graph, but note that the pivoting now changes the graph, so a new pivot may not be possible (depending on $h(x_0, \ldots, x_{t-1})$). Avoiding repetitions, that makes one pivot for every vertex on the first t variables, plus the identity transform. In total we get the lower bound $(t+1)2^{n-t-1}$.

Let $f \in \mathcal{F}^{n,t}$ such that its degree is t. Take $h(x_0, x_1, \ldots, x_{t-1}) = x_0 x_1 \cdots x_{t-1}$. Then, it is easy to see that after doing pivot on any edge mentioned above, the obtained function does not fulfil the condition of definition 7. □

Lemma 7. *Let $f \in \mathcal{F}^{n,t}$. Then the number of flat spectra of f w.r.t. $\{I, H, N\}^n$ is at least $(n+1)(t+1)2^{n-t-1}$.*

Proof. Let $f \in \mathcal{F}^{n,t}$. By theorem 9, its number of flat spectra w.r.t. $\{I, H\}^n$ is at least $(t+1)2^{n-t-1}$; furthermore, we can see that all the flat spectra correspond to graph operations, so the resulting state is associated to a graph. It can be shown [15] that local complementation at vertex j is realised by the application of N_j to the bipolar vector of the function, followed by a diagonal transform, which implies that the result of applying N_j to the bipolar vector of a function associated to a (simple, undirected) graph is always flat (this also follows as a special case of theorem 7). On the other hand, the result of applying the identity transform to the bipolar vector of a function associated to a graph is always flat. Therefore, the number of flat spectra of f w.r.t. $\{I, H, N\}^n$ is at least $n+1$ times its number of flat spectra w.r.t. $\{I, H, N\}^n$; i.e. $(n+1)(t+1)2^{n-t-1}$. □

6 Pivot Orbits and Codes

A binary linear $[n, k]$ code $\mathcal{C}$ is a linear subspace of $\mathrm{GF}(2)^n$ of dimension k. The 2^k elements of $\mathcal{C}$ are called *codewords*. We define the *dual* of the code $\mathcal{C}$ with respect to the standard inner product, $\mathcal{C}^\perp = \{\boldsymbol{u} \in \mathrm{GF}(2)^n \mid \boldsymbol{u} \cdot \boldsymbol{c} = 0, \forall \boldsymbol{c} \in \mathcal{C}\}$. The code $\mathcal{C}$ can be defined by a $k \times n$ *generator matrix*, C, whose rows span $\mathcal{C}$. Two codes, $\mathcal{C}$ and $\mathcal{C}'$, are considered to be *equivalent* if one can be obtained from the other by some permutation of the coordinates, or equivalently, a permutation of the columns of the generator matrix. $\mathcal{C}$ is called *self-dual* if $\mathcal{C} = \mathcal{C}^\perp$, and *isodual* if $\mathcal{C}$ is equivalent to $\mathcal{C}^\perp$. Self-dual and isodual codes must be *even*, i.e., all codewords must have even weight. A set of k independent columns of C is called an *information set* of $\mathcal{C}$. The remaining $n-k$ columns is called a *redundancy set*. We can permute the columns of C such that an information set makes up the first k columns. This matrix can now be transformed, by elementary row operations, into a matrix of the form $C' = (I \mid P)$, where I is a $k \times k$ identity matrix, and P is some $k \times (n-k)$ matrix. The matrix C' generates a code equivalent to $\mathcal{C}$ and is said to be of *standard form*. It follows that every code is equivalent to a code with generator matrix of standard form. The matrix $H = (P^{\mathrm{T}} \mid I)$, where I is an $(n-k) \times (n-k)$ identity matrix is called the *parity check matrix* of $\mathcal{C}$. Observe that $GH^{\mathrm{T}} = \mathbf{0}$, where $\mathbf{0}$ is the all-zero vector. It follows that H must be the generator matrix of $\mathcal{C}^\perp$. A code is *decomposable* if it can be written as the *direct sum* of two smaller codes. For example, let $\mathcal{C}$ be an $[n, k]$ code and $\mathcal{C}'$ an $[n', k']$ code. The direct sum, $\mathcal{C} \oplus \mathcal{C}' = \{u||v \mid u \in \mathcal{C}, v \in \mathcal{C}'\}$, where $||$ means concatenation, is an $[n+n', k+k']$ code.

It has previously been discovered that the LC orbits of simple undirected graphs corresponds to the equivalence classes of *self-dual additive codes over* $\mathrm{GF}(4)$ [3, 7, 10, 21]. We now show that pivot orbits of bipartite graphs correspond the equivalence classes of binary linear codes.

Definition 8. Let $\mathcal{C}$ be a binary linear $[n, k]$ code. Let $C = (I \mid P)$ be a generator matrix of standard form that generates a code equivalent to $\mathcal{C}$. Then the code $\mathcal{C}$ corresponds to the $(k, n-k)$-bipartite graph on n vertices with adjacency matrix

$$\Gamma = \begin{pmatrix} \mathbf{0}_{k\times k} & P \\ P^{\mathrm{T}} & \mathbf{0}_{(n-k)\times(n-k)} \end{pmatrix},$$

where $\mathbf{0}$ denote all-zero matrices of the specified dimensions. Note that the graph

corresponding to a code, like the generator matrix, is not uniquely defined.

An alternative description of the relationship between bipartite graphs and codes was given by Parker and Rijmen [14]. We have previously shown how a graph corresponds to a Boolean function. Applying the Hadamard transform, H, to all variables corresponding to vertices in one partition of the graph (and I to the other variables) produces, to within normalisation, the binary *indicator vector* of the corresponding code $\mathcal{C}$, i.e., a vector $(s_{\boldsymbol{c}})$, $\boldsymbol{c} \in \mathbb{Z}_2^n$, where $s_{\boldsymbol{c}} = 1$ if $\boldsymbol{c} \in \mathcal{C}$, and $s_{\boldsymbol{c}} = 0$ otherwise. More explicitly, for $s = (-1)^p$, and p a quadratic Boolean function representing the bipartite graph of the code $\mathcal{C}$, we have $(s_{\boldsymbol{c}}) = \mu(I \otimes \cdots \otimes I \otimes \cdots \otimes H \otimes \cdots \otimes H)s$, with μ some normalisation constant. Similarly, applying the H transform to the vertices of the other partition will give the indicator vector of $\mathcal{C}^\perp$

Lemma 8. *Let $G = (V, E)$ be a $(k, n-k)$-bipartite graph derived from the standard form generator matrix C of the $[n, k]$ code $\mathcal{C}$. Let G' be the graph obtained by pivoting on the edge $uv \in E$, followed by a swapping of vertices u and v. Both G and G' have generator matrices of the form given in definition 8, with submatrices P and P', respectively. Let the rows of P be labelled $1, 2, \ldots, k$, and let the columns of P be labelled $k+1, k+2, \ldots, n$. Assuming, without loss of generality, that $u \leq k$ and $v > k$, P can be transformed into P' by the following steps.*

1. *Store the current value of column v for later.*
2. *Add row u to all rows in $\mathcal{N}(v) \setminus \{u\}$, i.e., all rows that have 1 in coordinate v, except row u. (Observe that column v is now the basis vector e_u, i.e., it has 0 in all coordinates except coordinate u.)*
3. *Reset column v to the value that was stored initially.*

Proof. According to lemma 2, G' will remain $(k, n-k)$-bipartite. The transformation of P' follows from definition 3. Pivoting on the edge ij of the bipartite graph G is done by 'toggling' all pairs of vertices $\{x, y\}$, where $x \in \mathcal{N}(u) \setminus \{v\}$ and $y \in \mathcal{N}(v) \setminus \{u\}$. This is obtained by step 2 above, since row u of P defines $\mathcal{N}(u)$, and column v defines $\mathcal{N}(v)$. But in step 2 we have also 'toggled' the pairs $\{u, y\}$, where $y \in \mathcal{N}(v) \setminus \{u\}$, and we need steps 1 and 3 to correct this. □

Theorem 10. *Let $G = (V, E)$ be the bipartite graph derived from the standard form generator matrix $C = (I \mid P)$ of the code $\mathcal{C}$. The graph G' obtained by pivoting on*

the edge $uv \in E$ and then swapping vertices u and v corresponds to the standard form generator matrix $C' = (I \mid P')$ of the code $\mathcal{C}'$. The code $\mathcal{C}'$ is equivalent to $\mathcal{C}$, and can be obtained by interchanging coordinates u and v of $\mathcal{C}$.

Proof. Assume that $u \leq k$ and $v > k$. The effect of pivoting on the submatrix P was described in lemma 8. Now consider the following operations on $C = (I \mid P)$, where rows are labelled $1, 2, \ldots, k$, and columns are labelled $1, 2, \ldots, n$.

1. Observe that column u is the basis vector e_u.
2. Add row u to all rows in $\mathcal{N}(v) \setminus \{u\}$.
3. Column v is now the basis vector e_u, and column u has the value that column v had initially.
4. Swap columns u and v.

Comparing this with the algorithm for pivoting on P, it is easy to see that the resulting matrix is $C' = (I \mid P')$, i.e., the generator matrix corresponding to G'. The operations we have performed on C preserve the equivalence of linear codes, namely row additions and the swapping of columns u and v. □

Corollary 4. *Applying any sequence of pivot operations to the graph G corresponding to the code $\mathcal{C}$ will produce a graph corresponding to a code equivalent to $\mathcal{C}$.*

Consider a code $\mathcal{C}$. It is possible to go from a generator matrix of standard form, $C = (I \mid P)$, to a generator matrix of standard form, C', of any code equivalent to $\mathcal{C}$ by one of the $n!$ possible permutations of the columns of C. More precisely, we can get from C to C' via a combination of the following operations.

1. Permuting the columns of P.
2. Permuting the columns of I, followed by the same permutation on the rows of P, to restore standard form.
3. Swapping columns from I with columns from P, such that the first k columns of the generator matrix is an information set, followed by some row additions to restore standard form.

Theorem 11. *Let $\mathcal{C}$ and $\mathcal{C}'$ be equivalent codes. Let C and C' be any matrices of standard form generating $\mathcal{C}$ and $\mathcal{C}'$. Let G and G' be the bipartite graphs corresponding to C and C'. G' must be isomorphic to a graph obtained by performing some sequence of pivot operations on G.*

Proof. $\mathcal{C}$ and $\mathcal{C}'$ must be related by a combination of the operations 1, 2, and 3 listed above. It is easy to see that operations 1 and 2 applied to G produce a graph isomorphic to G. It remains to prove that operation 3 always correspond to some sequence of pivot operations. We know from theorem 10 that swapping columns u and v of C, where u is part of I and v is part of P, corresponds to pivoting on the edge uv of G and then swapping vertices u and v. When uv is not an edge of G, we can not swap columns u and v of C via pivoting. In this case, coordinate v of column u is 0, and column u is the basis vector e_u. Swapping these columns would result in a generator matrix where the first k columns have 0 at coordinate u. These columns can not correspond to an information set. It follows that if uv is not an edge of G, swapping columns u and v is not a valid operation of type 3 in the above list. Thus graph pivoting covers all possible permutations that map standard form generator matrices of equivalent codes to each other. □

Let us now consider the labelled graphs in the pivot orbit of $G = (V, E)$, i.e., graph isomorphism is not considered when the pivot orbit is generated. G is the bipartite graph representing the code $\mathcal{C}$. When we pivot on the edge $uv \in E$, without swapping vertices u and v afterwards, the resulting adjacency matrix will not be of the type we saw in definition 8. We can think of G as a graph corresponding to the information set $\{1, 2, \ldots, k\}$ of $\mathcal{C}$. Pivoting on the edge $uv \in E$, where $u \leq k$ and $v > k$, produces a graph representing another information set of $\mathcal{C}$, namely $\{1, 2, \ldots, k\} \setminus \{u\} \cup \{v\}$. With this interpretation, the next corollary follows from theorem 11.

Corollary 5. *Let G be the bipartite graph representing the code $\mathcal{C}$. Each labelled graph in the pivot orbit of G corresponds to an information set of $\mathcal{C}$. Moreover, the number of information sets of $\mathcal{C}$ equals the number of labelled graphs in the pivot orbit of G.*

Table 2: Numbers of LC Orbits of Graphs on n Vertices

	n											
	1	2	3	4	5	6	7	8	9	10	11	12
i_n^{LC}	1	1	1	2	4	11	26	101	440	3,132	40,457	1,274,068
t_n^{LC}	1	2	3	6	11	26	59	182	675	3,990	45,144	1,323,363

7 Enumeration of Pivot Orbits

We have previously classified all self-dual additive codes over GF(4) of length up to 12 [6, 7], by classifying orbits of simple undirected graphs with respect to local complementation and graph isomorphism. In Table 2, the sequence (i_n^{LC}) gives the number of LC orbits of connected graphs on n vertices, while (t_n^{LC}) gives the total number of LC orbits of graphs on n vertices. A database containing one representative from each LC orbit is available at `http://www.ii.uib.no/~larsed/vncorbits/`.

By recursively applying pivot operations to all edges of a graph, whilst checking for graph isomorphism using the program *nauty* [12], we can quickly find all members of the pivot orbit. Let $\boldsymbol{G}_n$ be the set of all unlabelled simple undirected connected graphs on n vertices. Let the set of all distinct pivot orbits of connected graphs on n vertices is a partitioning of $\boldsymbol{G}_n$ into i_n^P disjoint sets. Our previous classification of the LC orbits of all graphs of up to 12 vertices helps us to classify pivot orbits, since it follows from definition 2 that each LC orbit can be partitioned into some set of disjoint pivot orbits. We have used this fact to classify all pivot orbits of graphs on up to 12 vertices. In Table 3, the sequence (i_n^P) gives the number of pivot orbits of connected graphs on n vertices, while (t_n^P) gives the total number of pivot orbits of graphs on n vertices. A database containing one representative from each pivot orbit can be found at `http://www.ii.uib.no/~larsed/pivot/`.

We are particularly interested in bipartite graphs, because of their connection to binary linear codes. For the classification of the orbits of bipartite graphs with respect to pivot and graph isomorphism, the following technique is helpful. If G is an (a, b)-bipartite graph, it has 2^a+2^b-2 possible *extensions*. Each extension is formed by adding a new vertex and joining it to all possible combinations of at least one of the old vertices. Let $\boldsymbol{P}_n$ be a set containing one representative from each pivot orbit

Table 3: Numbers of Pivot Orbits of Graphs on n Vertices

n	i_n^P	t_n^P	$i_n^{P,B}$	$t_n^{P,B}$
1	1	1	1	1
2	1	2	1	2
3	2	4	1	3
4	4	9	2	6
5	10	21	3	10
6	35	64	8	22
7	134	218	15	43
8	777	1,068	43	104
9	6,702	8,038	110	250
10	104,825	114,188	370	720
11	3,370,317	3,493,965	1,260	2,229
12	231,557,290	235,176,097	5,366	8,361
13			25,684	36,441

of all connected bipartite graphs on n vertices. The set $\boldsymbol{E}_n$ be formed by making all possible extensions of all graphs in $\boldsymbol{P}_{n-1}$. It can then be shown that $\boldsymbol{P}_n \subset \boldsymbol{E}_n$, i.e., that the set $\boldsymbol{E}_n$ will contain at least one representative from each pivot orbit of connected bipartite graphs on n vertices. The set $\boldsymbol{E}_n$ will be much smaller than $\boldsymbol{G}_n$, so it will be more efficient to search for a set of pivot orbit representatives within $\boldsymbol{E}_n$.

In Table 3, the sequence $(i_n^{P,B})$ gives the number of pivot orbits of connected bipartite graphs on n vertices, and $(t_n^{P,B})$ gives the total number of pivot orbits of bipartite graphs on n vertices. A database containing one representative from each of these orbits can be found at `http://www.ii.uib.no/~larsed/pivot/`.

It follows from theorem 11 that the orbits of simple undirected graphs with respect to pivot and graph isomorphism correspond to equivalence classes of binary linear codes. Note that the codes $\mathcal{C}$ and $\mathcal{C}^\perp$ correspond to isomorphic graphs. This means that the pivot orbit of an $[n, k]$ code is simultaneously the pivot orbit of a non-equivalent $[n, n-k]$ code, with the exception of isodual codes, which are equivalent to their duals.

Table 4: Numbers of Pivot Orbits and Binary Linear Codes

n	$i_n^{P,B}$	i_n^C	i_n^{Ciso}
1	1	1	-
2	1	1	1
3	1	2	-
4	2	3	1
5	3	6	-
6	8	13	3
7	15	30	-
8	43	76	10
9	110	220	-
10	370	700	40
11	1,260	2,520	-
12	5,366	10,503	229
13	25,684	51,368	-

Theorem 12. *Let $k \neq \frac{n}{2}$. Then the number of inequivalent binary linear $[n,k]$ codes, which is also the number of inequivalent $[n,n-k]$ codes, is equal to twice the number of pivot orbits of $(n-k,k)$-bipartite graphs.*

When n is even and $k = \frac{n}{2}$, the number of inequivalent binary linear $[n,k]$ codes is equal to twice the number of pivot orbits of (k,k)-bipartite graphs minus the number of isodual codes of length n.

Note that if we only consider connected graphs on n vertices, we get the number of indecomposable codes of length n, i_n^C. The total number of codes can easily be derived from the values of (i_n^C). Table 4 gives the number of pivot orbits of connected bipartite graphs on n vertices $(i_n^{P,B})$, the number of indecomposable binary linear codes of length n (i_n^C), and the number of indecomposable isodual codes of length n (i_n^{Ciso}).

The number of linear codes of high length can be calculated by using computer algebra tools [9], and a complete classification has been carried out for codes of length up to 14 [22] by using a different graph based approach. We hope, however, that our method will be more efficient than existing algorithms for classifying special

Table 5: Numbers of Pivot Orbits of Labelled Graphs on n Vertices

n	$i_n^{P,L}$	$t_n^{P,L}$	$i_n^{P,B,L}$	$t_n^{P,B,L}$
1	1	1	1	1
2	1	2	1	2
3	2	6	1	5
4	11	29	4	18
5	119	240	26	92
6	2,303	3,623	251	693
7	80,923	105,564	3,412	7,613

types of codes.

Finally, we have also enumerated the orbits of labelled graphs with respect to the pivot operation only, i.e., not considering graph isomorphism. In Table 5, the sequence $(i_n^{P,L})$ gives the number of pivot orbits of connected labelled graphs on n vertices, while $(t_n^{P,L})$ gives the total number of pivot orbits of labelled graphs on n vertices. Similarly $(i_n^{P,B,L})$ and $(t_n^{P,B,L})$ give the numbers for connected and unconnected bipartite labelled graphs.

References

[1] Arratia, R., Bollobás, B., and Sorkin, G. B.: "The interlace polynomial: a new graph polynomial". In *Proceedings of the Eleventh Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 237–245, ACM, New York, 2000.

[2] Arratia, R., Bollobás, B., and Sorkin, G. B.: "The interlace polynomial of a graph". *J. Combin. Theory Ser. B*, **92**(2), pp. 199–233, 2004. `arXiv:math.CO/0209045`.

[3] Bouchet, A.: "Graphic presentations of isotropic systems". *J. Combin. Theory Ser. B*, **45**(1), pp. 58–76, 1988.

[4] Bouchet, A.: "Tutte-Martin polynomials and orienting vectors of isotropic systems". *Graphs Combin.*, **7**(3), pp. 235–252, 1991.

[5] Charnes, C., Rötteler, M., and Beth, T.: "Homogeneous bent functions, invariants, and designs". *Designs, Codes and Cryptography*, **26**(1–3), pp. 139–154, 2002. `http://www.iqc.ca/~mroetteler/bent.html`.

[6] Danielsen, L. E.: *On Self-Dual Quantum Codes, Graphs, and Boolean Functions*. Master's thesis, Department of Informatics, University of Bergen, Norway, March 2005. arXiv:quant-ph/0503236.

[7] Danielsen, L. E. and Parker, M. G.: "On the classification of all self-dual additive codes over GF(4) of length up to 12", 2005. To appear in J. Comb. Theory Ser. A. arXiv:math.CO/0504522.

[8] Fon-der Flaas, D. G.: "On local complementations of graphs". In *Combinatorics (Eger, 1987)*, volume 52 of *Colloq. Math. Soc. János Bolyai*, pp. 257–266, North-Holland, Amsterdam, 1988.

[9] Fripertinger, H. and Kerber, A.: "Isometry classes of indecomposable linear codes". In *Applied algebra, algebraic algorithms and error-correcting codes*, volume 948 of *Lecture Notes in Comput. Sci.*, pp. 194–204, Springer-Verlag, Berlin, 1995.

[10] Glynn, D. G.: "On self-dual quantum codes and graphs", 2002. Submitted to Electron. J. Combin. http://homepage.mac.com/dglynn/.Public/SD-G3.pdf.

[11] Hein, M., Eisert, J., and Briegel, H. J.: "Multi-party entanglement in graph states". *Phys. Rev. A*, **69**(6), p. 062 311, 2004. arXiv:quant-ph/0307130.

[12] McKay, B. D.: *nauty User's Guide*. 2003. http://cs.anu.edu.au/~bdm/nauty/.

[13] Monaghan, J. and Sarmiento, I.: "Properties of the interlace polynomial via isotropic systems". http://academics.smcvt.edu/jellis-monaghan/#Papers.

[14] Parker, M. G. and Rijmen, V.: "The quantum entanglement of binary and bipolar sequences". In *Sequences and Their Applications – SETA'01*, Discrete Math. Theor. Comput. Sci., pp. 296–309, Springer-Verlag, London, 2002. arXiv:quant-ph/0107106.

[15] Riera, C. and Parker, M. G.: "Generalised bent criteria for Boolean functions (I)", 2004. Accepted for IEEE Trans. Inform. Theory. arXiv:cs.IT/0502049.

[16] Riera, C. and Parker, M. G.: "On pivot orbits of boolean functions". In *Optimal Codes and Related Topics, Pamporovo, Bulgaria*, 2005. http://www.ii.uib.no/~matthew/octalk4.pdf.

[17] Riera, C. and Parker, M. G.: "Spectral interpretations of the interlace polynomial", 2005. Proceedings of the Workshop on Coding and Cryptography (WCC). http://www.ii.uib.no/~matthew/WCC7.pdf.

[18] Riera, C., Petrides, G., and Parker, M. G.: "Generalised bent criteria for Boolean functions (II)", 2004. arXiv:cs.IT/0502050.

[19] Rothaus, O. S.: "On "bent" functions". *J. Comb. Theory Ser. A*, **20**(3), pp. 300–305, 1976.

[20] Van den Nest, M. and De Moor, B.: "Edge-local equivalence of graphs".

arXiv:math.CO/0510246.

[21] Van den Nest, M., Dehaene, J., and De Moor, B.: "Graphical description of the action of local Clifford transformations on graph states". *Phys. Rev. A*, **69**(2), p. 022 316, 2004. arXiv:quant-ph/0308151.

[22] Östergård, P. R. J.: "Classifying subspaces of Hamming spaces". *Des. Codes Cryptogr.*, **27**(3), pp. 297–305, 2002.