# Cliques, Codes, and Association Schemes

Andrew Nagarajah
Supervised by: Prof. Mike Newman

7 April 2021

**Abstract**

This report is an exploration of some of the topics within *Delsarte theory*, which uses ideas from graph theory, algebra, and optimization to address questions in coding theory in particular, and combinatorics more broadly. The centre of study is the *association scheme*, which provides a setting in which to view various objects, especially *distance-regular graphs*. This perspective enables the computation of various parameters of interest, including the eigenvalues of graphs and upper bounds on codes, cliques, and independent sets.

This report aims to be mostly self-contained, though background knowledge of basic linear algebra and group theory is required. The important aspects of this theory is reviewed in the appendix.

# Contents

# 1. Introduction

## 1.1 Coding Theory

In most communication, a transmitter in one location must send a message to a receiver in another location, possibly by means of a faulty (or *noisy*) channel which may introduce errors into the message.

For example, a spacecraft might send scientific data back to earth from another planet by means of a radio signal. Along the way, other sources of electromagnetic radiation might interfere with the signal, such that when it arrives at earth, the signal received is slightly different from the one sent. The scientific data, if successfully received, might needed to be saved in some storage medium so that it may be accessed in the future. Over time however, the storage medium may degrade, so that when the data is retrieved, it may be slightly different from when it was saved.

In human speech, if two friends are speaking in a loud environment and one says "I love you", if the ambient noise is loud enough, the friend might hear "I lave you" instead. Not all possible sounds (or combinations of letters) are valid messages in English though, so the friend could reasonably guess at what was meant. In this sense, English contains *redundancy*. However, if the environment is so loud that the friend hears "I late you", then they might not know what the message what intended to mean, or incorrectly guess that the true message was "I hate you". On the other hand, if the friend hears "l shgbot gtin", then they will know for sure that the message was corrupted; they will know not to misinterpret the message, and may ask that it be re-transmitted.

These are the two fundamental goals of coding theory: to encode data in such a way that it can be recovered if some errors are introduced; or if it cannot be recovered, then at least to detect that the message was corrupted. As the above example illustrates, and encoding scheme may be able to correct a limited number of errors, or detect some number of errors. For this report, we will focus on the former goal.

In general, a coding scheme will consist of a set of signals $V$, among which a subset $Y$ (called the CODE) might consist of valid messages. Then, the receiver will need a function

$V \times V \to \mathbb{N}$ to tell how many errors would be required to change one signal into another. We assume that if $s$ errors convert signal $u$ into signal $v$, then $s$ "opposite" errors can convert $v$ into $u$. Furthermore, if $v$ can be converted into $w$ with $t$ errors, then $u$ can be converted into $w$ with no more that $s+t$ errors (this is called the TRIANGLE INEQUALITY). Finally, $u$ will require 0 errors to be converted into itself; conversely for any pair of distinct signals $u$ and $v$, a nonzero number of errors should be required to convert one into the other. Such a set of signals, paired with such an error-counting function, is called a (DISCRETE) METRIC SPACE, and its function is typically called a DISTANCE. When a signal $u$ arrives at the receiver, they can try and find a signal $v$ in the code $Y$ which minimizes the distance between $u$ and $v$. If there is a unique such minimizer $v$, then we decode $u$ into $v$.

For example, if we wish to encode the messages 0 (or 'no') and 1 (or 'yes'), the set of signals might consist of all binary strings of length three, with 000 and 111 as the codewords. If errors consist of individual bit flips, then the message 101 can be decoded into 111, as only one error is required to convert the latter into the former. However if two errors occur and 100 is received, then this will be decoded into 000, since fewer errors are required to convert between the two. Therefore, this code (called a *repetition code*) can correct at most 1 error.

More generally, if $d$ is the MINIMUM DISTANCE between any two codewords, and $w$ is a signal at distance no more than $\left\lfloor \frac{d-1}{2} \right\rfloor$ from a codeword $u$, then its distance to any other codeword $v$ must be greater. Otherwise, $u$ could be converted to $v$ with at most $d-1$ errors – a contradiction!

On the other hand, in the example repetition code given above, there were $2^1 = 2$ codewords, and $2^3 = 8$ possible signals. So, for every message we wish to send, three times as many bits need to be transmitted. This is called the RATE of the code.

These two parameters, the minimum distance of the code, and the proportion of signals which are codewords, are the two primary measures of the effectiveness and efficiency of a code. A large minimum distance in a code will mean that many errors can be corrected, while a large proportion will mean that messages are being transmitted efficiently. The challenge, then, is to design codes which have simultaneously a large minimum distance and a large proportion of codewords. One of the aims of this report is to put bounds on how efficient a code can be given a target effectiveness; plainly, if we want a certain minimum distance code in a set of signals, at most how many codewords can there be? The approach to this question is ultimately due to Delsarte [1], and has be expanded upon by Schrijver [5].

## 1.2 Linear Codes and Hamming Graphs

Let $V$ be denote a vector space of dimension $d$ over a finite field of order $q$. Then $Y \subseteq V$ is a LINEAR CODE if $Y$ is a linear subspace of $V$. One such vector space is the set of length $d$ bitstrings: this is a vector space of dimension $d$ over the finite field of two elements. (Here, the addition of vectors corresponds to the bitwise exclusive-or of bitstrings.) A code in such a vector space would be immediately applicable to digital data. This motivates the study of linear codes. One way in which binary data can be corrupted is through bit flips. Analogously, if $v = v_1e_1 + \cdots + v_de_d$ is a vector written in the fixed basis $e_1, \ldots, e_d$, then an error may occur by picking an index $i$, and changing the value of $v_i$. In order to count such errors, we define the following distance function on $V$.

**Definition 1.2.1**
*If $V$ is the vector space of $d$-tuples over a finite field, then the* (HAMMING) WEIGHT *of a vector $v = (v_1, \ldots, v_d)$ is the number of nonzero components $v_i$; this is denoted* $\mathrm{wt}\,(v)$. *Then the* (HAMMING) DISTANCE *between vectors $u, v$ is defined*

$$\mathrm{dist}\,(u, v) := \mathrm{wt}\,(u - v) \ .$$

Since a linear subspace must be closed under subtraction, the minimum distance of a linear code is the same as its minimum (nonzero) weight. Also, the distance between two vectors is the number of coordinates on which they differ. Meanwhile, the rate of the code is the ratio of the dimension of the code to the dimension of its ambient space.

While much of coding theory is done in this setting, this report will look at codes through a combinatorial lens. A DIRECTED GRAPH on a vertex set $V$ is (for the purposes of this report) a binary relation $R \subseteq V \times V$. If the relation is symmetric, then the graph is said to be UNDIRECTED. For each vector space as describe above, we can define a corresponding graph, and investigate codes in this graph instead.

**Definition 1.2.2**
*The* HAMMING GRAPH $H(d, q)$ *is defined so that its vertex set $V$ is the set of all $d$-tuples, each entry of which belongs to a fixed set of $q$ elements. Two vertices are said to be adjacent if they have hamming distance* 1.

*The graph $H_i(d, q)$ is defined on the same vertex set; however, vertices in this graph are adjacent if they have hamming distance $i$. With this definition, $H(d, q) = H_1(d, q)$.*

Note that in the case of linear codes, if $F_q$ is a finite field of order $q$, then $q$ must be a prime power. Also, all vector spaces of dimension $d$ are isomorphic to $F_q^d$. However, in the setting of Hamming graphs $H(d, q)$, the integer $q$ need not be a prime power. Thus, the

vertex set can be modelled as $\mathbb{Z}_q^d$. If $q$ is a prime power, then $\mathbb{Z}_q$ still may not be isomorphic to the additive group of $F_q$ (they are isomorphic if and only if $q$ is prime). However, the Hamming graph with $Z_q^d$ as its vertex set will be isomorphic to the Hamming graph with $F_q^d$ as its vertex set, as adjacency is solely determined by the number of components of two tuples which as the same. The algebraic structure plays no role.

By generalizing finite vector spaces to Hamming graphs in this way, the explicit algebraic structure of the vector space is lost. However, the Hamming graphs have many combinatorial properties which will prove useful in this report. (It will also turn out that the Hamming graphs have a number of elegant algebraic properties as well, but this is not obvious *a priori*.)

## 1.3 Distance-Regular Graphs

Define $M$-cliques.

I'm not sure if I should maybe merge this section with $P$-polynomial section?

- Definition
- Basic parameters
- Examples?

# 2. Association Schemes

## 2.1 Association Schemes

- Definition(s)
- Examples?
- Basic parameters

**Definition 2.1.1** (Commutative Association Scheme – Combinatorial [1, Section 2.1])
*Let $D = \{0, 1, \ldots, d\}$ for some $d \geq 1$. A* COMMUTATIVE ASSOCIATION SCHEME *$\mathcal{A}$ is a set $X$, called the* VERTEX SET*, together with a set of relations $\{R_i\}_{i=0}^{d}$ satisfying the following axioms:*

1. *The set of relations $\{R_i\}_{i=0}^{d}$ partitions $X \times X$;*
2. *$R_0$ is the diagonal relation $\{(x, x) \mid x \in X\}$;*
3. *For each $i \in D$, there is an $i' \in D$ such that $R_{i'}$ is the opposite relation $\{(y, x) \mid (x, y) \in R_i\}$ of $R_i$;*
4. *For every triple $i, j, k \in D$, there exists a constant $p_{i,j}^k$ such that for all $(x, y) \in R_k$, there are exactly $p_{i,j}^k$ vertices $z$ such that $(x, z) \in R_i$ and $(z, y) \in R_j$; furthermore, $p_{i,j}^k = p_{j,i}^k$.*

*As used above, the elements of $X$ are called* VERTICES*, and vertices $(x, y) \in R_i$ are called* $i^{\text{TH}}$ ASSOCIATES.

*Each relation $R_i$ is called a* CLASS *of $\mathcal{A}$, which has* DIAMETER *$d$ (this will be explained in connection with distance-regular graphs in the next section (2.2)). Sometimes, $\mathcal{A}$ is said to be a $d$-class association scheme (as the diagonal relation is discounted).*

To every relation $R \subseteq X \times X$ there exists a $X \times X$ 01 matrix $A$, where

$$A_{xy} = \begin{cases} 1 & \text{if} \quad (x, y) \in R \\ 0 & \text{if} \quad (x, y) \notin R \end{cases}; \tag{2.1}$$

$A$ is then called the ADJACENCY MATRIX of $R$. This is a bijective correspondence between relations on $X \times X$, and $X \times X$ matrices with each entry either 0 or 1. (Note that binary relations are precisely directed graphs without parallel edges, and this usage of the term *adjacency matrix* agrees with its usage in graph theory.)

By exchanging the relations $R_i$ for their adjacency matrices $A_i$, the combinatorial definition of an association scheme can be reformulated as follows:

**Definition 2.1.2** (Commutative Association Scheme – Algebraic [3, Chapter 12])
*Let $\circ$ denote the* SCHUR *product of matrices of the same shape:*

$$(A \circ B)_{xy} = A_{xy} B_{xy} \ . \tag{2.2}$$

*(This is also called the Hadamard, or entrywise product.)*

*A* COMMUTATIVE ASSOCIATION SCHEME *is a vertex set $X$ along with $X \times X$ matrices $A_0, A_1, \ldots A_d$ such that*

1. $\sum_{i=0}^{d} A_i = J$, *the all-ones matrix, and* $A_i \circ A_j = \delta_{i,j} A_i$;
2. $A_0 = I$, *the identity matrix;*
3. *For every $i \in D$ there is an $i' \in D$ such that $A_i^T = A_{i'}$;*
4. *For every $i, j$,*
$$A_i A_j = A_j A_i = \sum_{k=0}^{d} p_{i,j}^k A_k \ .$$

It is clear from the first axiom that the $A_i$ are 01 matrices.

The first three equivalences are straightforward translations. To see the last, observe that

$$(A_i A_j)_{xy} = \sum_{z \in X} (A_i)_{xz} (A_j)_{zy}$$

which counts the number of vertices $z \in X$ such that

$$\begin{cases} (A_i)xz = 1 & \iff (x,z) \in R_i \\ (A_j)zy = 1 & \iff (z,y) \in R_j \end{cases} .$$

Then, $(A_i A_j)_{xy} = p_{ij}^k$ exactly when $(A_k)_{xy} = 1 \iff (x,y) \in R_k$.

The requirement in (Combinatorial 4) that $p_{i,j}^k = p_{j,i}^k$ corresponds to the requirement that $A_i A_j = A_j A_i$ (Algebraic 4), which is why such association schemes are called *commutative*.

If the requirement of *commutativity* is dropped, then every finite group $G$ is an association scheme in the following way. Cayley's theorem says that every group is isomorphic to the group of permutations on $G$ given by left multiplication (the same construction will work if right multiplication is used everywhere instead of left). By identifying each element of $G$ with the $G \times G$ *permutation matrix*, we obtain an association scheme with $G$ as the vertex set, and permutation matrices as the classes. Since the identity element of $G$ will map to the identity matrix, the product of two such permutations is another such permutation, and the transpose of a permutation matrix is its inverse, axioms (2, 3, 4) are satisfied. Since for every $g, h \in G$, only the element $hg^{-1}$ maps $g$ to $h$, exactly one of the permutation matrices will have a 1 in the $(g, h)$-entry; all the others will be 0. This association scheme will be commutative if and only if the group $G$ is.

Not every association scheme (commutative or not) arises is this way, but many schemes of interest are closely related to a particular group. Nevertheless, much of the utility of association schemes, and the elegance of their theory, derives from the connection between the combinatorial view of schemes as relations or (di)graphs (2.1.1), and the algebraic view of schemes as matrices (2.1.2).

A particularly important class of association scheme, called SYMMETRIC, is one in which every relation is symmetric (i.e. $i = i'$ in Combinatorial 3), or equivalently, each adjacency matrix is symmetric ($A_i^T = A_i$ in Algebraic 3). In this case, the relations $R_i$ form *undirected* graphs $\Gamma_i$ with vertex set $X$, and edge set given by

$$x \sim y \iff (x, y) \in R_i \iff (y, x) \in R_i \ .$$

The most important class of symmetric association scheme (for the purposes of this report) arise as the distance graphs of a distance-regular graph (2.2). From this setting, we can generalize the following definition. (TODO define for DRGs)

**Definition 2.1.3** (Cliques and Cocliques [1, Section 3.3])
*Let $Y \subseteq X$, and $C \subseteq D$, where $0 \in C$. Then $Y$ is called a $C$-*CLIQUE *if*

$$R_i \cap Y^2 = \emptyset \quad \forall i \in D \setminus C \ .$$

*Equivalently, $Y$ is a $C$-clique if for all $x, y \in Y$,*

$$(x, y) \in R_i \implies i \in C \ .$$

*Let $C^* := C \setminus \{0\}$, and $\overline{C} = D \setminus C^*$. Then $Y$ is a $C$-clique if and only if it is a $\overline{C}$-*COCLIQUE.

*(TODO Move below note to DRG defn.) (Note that such a set $Y$ is called a $C$-code, or $\overline{C}$-anticode in [3].)*

### 2.1.1 The Bose-Mesner Algebra

For any association scheme $\mathcal{A}$, there are the adjacency matrices $A_0, A_1, \ldots, A_d$ (2.1.2). Since any product of these matrices belongs to their span (4), their span

$$\mathbb{A} := \operatorname{span}\{A_0, A_1, \ldots, A_d\} \tag{2.3}$$

is closed under matrix multiplication. This structure is called the Bose-Mesner algebra of $\mathcal{A}$.

Note also that from the first axiom of an association scheme (1), $\mathbb{A}$ is also closed under the *Schur product*, so that $\mathbb{A}$ is actually an algebra with respect to two *different* products. This *duality* will form an important aspect of the theory of association schemes, and will be discussed in detail in this section and the next (2.1.2).

Since the adjoint $A_i^* = A_i^T$ belongs to $A_0, A_1, \ldots, A_d$ for every $i$, and these matrices all commute, each $A_i$ is a normal operator (TODO reference). From the spectral theorem, we can decompose each matrix

$$A_i = \sum_j \theta_{ij} \widetilde{F_{ij}}$$

into a linear combination of orthogonal idempotents $\widetilde{F_{ij}}$, where

$$I = \sum_j \widetilde{F_{ij}} \ .$$

Since each $\widetilde{F_{ij}}$ is a polynomial in $A_i$, and the $A_i$ all commute, so too do the $\widetilde{F_{ij}}$. Therefore, the products $\prod_i \widetilde{F_{ik_i}}$ (for any choices of $k_i$) are also orthogonal idempotents, though some may be zero, so that the nonzero products are linearly independent. Furthermore, their sum is the identity matrix

$$I = I^{d+1} = \prod_i \left( \sum_j \widetilde{F_{ij}} \right) = \sum \left( \prod_i \widetilde{F_{ik_i}} \right)$$

where the latter sum is taken over all choices of $k_i$. Therefore, we can express each $A_i$ as

linear combinations of these products, as

$$\begin{aligned} A_i \prod_{i'} \widetilde{F_{ik_i}} &= A_i \widetilde{F_{ik_i}} \prod_{i' \neq i} \widetilde{F_{ik_i}} \\ &= \theta_{ik_i} \widetilde{F_{ik_i}} \prod_{i' \neq i} \widetilde{F_{ik_i}} \\ &= \theta_{ik_i} \prod_{i'} \widetilde{F_{ik_i}} \\ \implies A_i = A_i I &= \sum \theta_{ik_i} \prod_{i'} \widetilde{F_{ik_i}} \ . \end{aligned}$$

Therefore, the non-zero products $\prod_i \widetilde{F_{ik_i}}$ span $\mathbb{A}$ and are linearly independent, so there are precisely $d+1 = \dim \mathbb{A}$ of them: $F_0, F_1, \ldots, F_d$. They are orthogonal idempotents, and each is self-adjoint since the $\widetilde{F_{ij}}$ are self-adjoint and commute. These matrices are called the PRINCIPAL IDEMPOTENTS of $\mathcal{A}$, and form an alternative basis to $A_0, A_1, \ldots, A_d$. (Because the $A_i$ are orthogonal idempotents under the Schur product, they are sometimes called SCHUR IDEMPOTENTS of $\mathcal{A}$.) Then, one can write

$$A_i = \sum_{j=0}^{d} P_i(j) F_j$$

and form the $(d+1) \times (d+1)$ matrix $P$ by $P_{ji} = P_i(j)$ (note the reversed indices). This is called the MATRIX OF EIGENVALUES of $\mathcal{A}$. [3, Theorem 12.2.1]

In the case that $\mathcal{A}$ is symmetric, each $A_i$ is a real, symmetric operator, so its eigenvalues $\theta_{ij}$ are real, as are its idempotents $\widetilde{F_{ij}}$. Therefore, the idempotents $F_j$ are real, as is the matrix of eigenvalues $P$.

### 2.1.2 Duality and Characterization

Already a comparison between the two bases $A_0, A_1, \ldots, A_d$ and $F_0, F_1, \ldots, F_d$ of $\mathbb{A}$ suggests a duality. For example, the $A_i$ are orthogonal idempotents with respect to the Schur product, while the $F_j$ are orthogonal idempotents with respect to the usual product; $\sum_i A_i = J$, the identity of the Schur product, while $\sum_j F_j = I$, the identity of the usual product. In this section, additional "dual" properties will be discovered by examining relations with respect to one basis, and searching for analogous ones in the opposite basis.

Many of these properties will involve two matrices to be introduced, which describe how each basis is transformed into the other. Some will be vital to the later analysis of cliques in association schemes. Others will be used to show that either of these matrices, taken in isolation, completely characterize the scheme!

First, the axioms of an association scheme (2.1.2) require that $A_0 = I$, the identity of the usual product. As already illustrated, $J$ the all-ones matrix acts as the "dual" of $I$, so let us examine:

$$JF_j = \left(\sum_i A_i\right) F_j = \left(\sum_i P_i(j)\right) F_j \ .$$

Letting $\gamma_j := \sum_i P_i(j)$, $JF_j = \gamma_j F_j$ demonstrates that $\gamma_j$ is an eigenvalue of $J$, with eigenvectors in the column space of $F_j$. Since $J$ is orthogonally diagonalizable, with eigenvalues $n = |X|$ of multiplicity 1, and 0 of multiplicity $n-1$, and the $F_j$ are orthogonal idempotents whose columns are eigenvectors of $J$, there is exactly one $j$ with $\gamma_j = n$. Since $JF_j = nF_j$, each row of $F_j$ is equal; but $F_j$ is self-adjoint the columns are all equal as well. Therefore, $F_j = \alpha J$, and $F_j$ is idempotent, so that

$$F_j = \alpha J = \alpha^2 J^2 = \alpha^2 nJ \implies \alpha = \frac{1}{n} \ .$$

Since the ordering $F_0, F_1, \ldots, F_d$ is arbitrary, by convention an order is taken such that $F_0 = \frac{1}{n}J$.

Since the $F_j$ are orthogonal, $\gamma_j = 0$ for all $j > 0$, so that the row sums of $P$ are known:

$$\sum_{i=0}^{d} P_i(j) = \begin{cases} n & \text{if} \quad j = 0 \\ 0 & \text{if} \quad j > 0 \end{cases} \ . \tag{2.4}$$

Moreover, since $F_0 = \frac{1}{n}J$, the all-ones vector $\mathbf{1}$ is an eigenvector of each $A_i$: let $\nu_i$ be the eigenvalue, so that $A_i\mathbf{1} = \nu_i\mathbf{1}$. Note that $\nu_i$ is the (constant) row sum of $A_i$. Since $A_i^T = A_{i'}$, the column sums of $A_i$ are constant as well, and if the scheme is symmetric, $A_i^T = A_i$, so the column sums are equal to $\nu_i$ as well. The quantity $\nu_i$ is called the VALENCY of $A_i$. (This usage agrees with its usage for regular graphs, where it equals the (constant) degree of vertices. This is also equal to the row sum of the adjacency matrix.)

Next, since $F_0, F_1, \ldots, F_d$ is a basis for $\mathbb{A}$, we can write

$$A_i = \sum_j P_i(j) F_j \tag{2.5}$$

uniquely, and since the $F_j$ are orthogonal idempotents, $A_i F_j = P_i(j) F_j$. Dually, $A_0, A_1, \ldots, A_d$ is a basis for $\mathbb{A}$, so we can write

$$F_j = \frac{1}{n} \sum_i Q_j(i) A_i \tag{2.6}$$

uniquely, and since the $A_i$ are Schur orthogonal idempotents, $F_j \circ A_i = \frac{1}{n} Q_i(j) A_i$. These

coefficients form the MATRIX OF DUAL EIGENVALUES $Q$, defined such that $Q_{ij} = Q_j(i)$.

By substituting (2.6) into (2.5), we obtain

$$nA_i = \sum_j P_i(j) \sum_{i'} Q_j(i') A_{i'} = \sum_{i'} \left( \sum_j P_i(j) Q_j(i') \right) A_{i'} \ .$$

Because the $A_i$ form a basis, $nA_i$ is written uniquely, so that

$$\sum_j P_i(j) Q_j(i') = n\delta_{ii'} \implies PQ = nI \ .$$

Since the first row sum of $P$ is $n$ (2.4), if we are given $P$ and *no other information*, then by computing its inverse we can derive $Q$.

If we were given only $Q$ instead, we could calculate $P$ by taking the inverse and multiplying by $n$. All that remains, then, is to derive $n$ from $Q$. In the other direction, this was done by examining the row sums of $P$, so we do the same with $Q$. The rows of $Q$ correspond to the $A_i$, the columns to the $F_j$, and $F_j \circ A_i = \frac{1}{n} Q_j(i) A_i$, so we fix a row and sum over the columns:

$$\left( \sum_j F_j \right) \circ A_i = \frac{1}{n} \left( \sum_j Q_j(i) \right) A_i \ .$$

But $\sum_j F_j = I = A_0$ and $A_0 A_i = \delta_{0i} A_i$ so

$$\sum_{j=0}^{d} Q_j(i) = \begin{cases} n & \text{if} \quad i = 0 \\ 0 & \text{if} \quad i > 0 \end{cases} \ . \tag{2.7}$$

Therefore, given either $P$ or $Q$, and no other information, the other can be derived. However, this process involves matrix inversion, which is both expensive to compute and awkward to contemplate. In fact, if a little extra information is known, then there exists an extraordinarily simple formula relating the two matrices. This extra information consists of the valencies and their dual concept, the multiplicities, of the scheme.

We have already encountered the valency $\nu_i$ of $A_i$. Since by definition $\nu_i = P_i(0)$, consider the first row of $Q$ and denote $Q_j(0)$ by $\mu_j$. From the definition of $Q$ we have $F_j \circ A_0 = \frac{1}{n} \mu_j A_0$, but $A_0$ is the identity matrix, so that the diagonal entries of $F_j$ are identically $\mu_j / n$. Moreover, $F_j$ is an $n \times n$ matrix, so $\operatorname{trace} F_j = \mu_j$, and since $F_j$ is idempotent, $\operatorname{rank} F_j = \operatorname{trace} F_j$ (TODO reference). If we fix $i$ and suppose momentarily that the $P_i(j)$ are all distinct, then the column space of $F_j$ is the $P_i(j)$-eigenspace of $A_i$, in which case $\mu_j = \dim \operatorname{col} F_j$, which is the algebraic multiplicity. For this reason (even if the

$P_i(j)$ are not distinct), $\mu_j$ will be called the MULTIPLICITY of $F_j$. (Is it possible that the $P_i(j)$ are not all distinct for any $i$?)

Consider the different products between matrices in one basis and matrices in the other: $A_iF_j$ and $A_i \circ F_j$. Since $F_j$ is self-adjoint, and commutes with $A_i$, $A_iF_j = F_j^* A_i$. Writing the product in this way, we might notice that the trace of the regular product is actually the (standard) inner product of $A_i$ and $F_j$, as is the sum of all entries in the Schur product.

$$\langle A_i \ , \ F_j \rangle = \operatorname{trace} F_j^* A_i = \operatorname{sum} A_i \circ F_j$$

On one hand,

$$\operatorname{trace} A_iF_j = P_i(j) \operatorname{trace} F_j = P_i(j)\mu_j \ ,$$

while on the other hand, the $n$ rows of $A_i$ all sum to $\nu_i$, so

$$\operatorname{sum} A_i \circ F_j = \frac{1}{n} Q_j(i) \operatorname{sum} A_i = Q_j(i)\nu_i \ .$$

Of course these two expressions must be equal, so letting $\nu$ denote the vector of valencies, and $\mu$ the vector of multiplicities, we see that

$$P^T \operatorname{diag}(\mu) = \operatorname{diag}(\nu) Q \ . \tag{2.8}$$

Therefore, given the valencies and multiplicities, $Q$ can be computed inexpensively from $P$ and vice versa. Furthermore, this explicit expression will be useful in later chapters when contemplating expressions involving $Q$ and vice versa.

Briefly, note that in the definition of an association scheme, it was required that $A_0 = I$, but it is only by a conventional order that $F_0 = \frac{1}{n}J$. If instead we only required that $A_i = I$ for some $i$, then $\nu$ can still be identified within the permuted matrix of eigenvalues as the only row with nonzero sum. Similarly, $\mu$ can be identified within $Q$ as the only row with nonzero sum, even if we do not adhere to our convention.



For the remainder of this report, the above properties of the Bose-Mesner algebra and its duality will suffice. However, with just a little extra work, we show that $\mathbb{A}$ is actually characterized by its matrix of eigenvalues (or its matrix of dual eigenvalues).

The Bose-Mesner algebra has three kinds of structures on it: it is a vector space over $\mathbb{C}$, it is a ring with respect to the usual matrix product, and it is a ring with respect to the Schur product. In order to characterize $\mathbb{A}$, we will need to define all three structures from $P$ alone.

First, $P$ is a $(d+1) \times (d+1)$ matrix when $\dim \mathbb{A} = d+1$. Since vector spaces are characterized complete by their field and their dimension, the shape of $P$ suffices here.

Next, since we have a basis $F_0, F_1, \ldots, F_d$ in which the usual matrix product is "known" (that is, they are orthogonal idempotents), this structure also will be characterized (vacuously) by $P$.

Finally, we must characterize the Schur product in $\mathbb{A}$. We also have a basis of orthogonal idempotents with respect to this product – namely $A_0, A_1, \ldots, A_d$ – but since this is a different basis from the one used previously, for the two products to be characterized simultaneously, we must be able to perform a change of basis. However, $A_i = \sum_j P_i(j) F_j$ and $F_j = \sum_i Q_j(i) A_i$, where $Q$ can be derived from $P$.

In other words, all the information needed to compute the isomorphism between $\mathbb{A}$ and $\mathbb{C}^{d+1}$ is encapsulated in $P$. By picking a basis $f_0, f_1, \ldots, f_d$ in $\mathbb{C}^{d+1}$, the regular product can be defined by $f_i f_j := \delta_{ij} f_i$, and extended bilinearly. Then, the Schur product can be defined by rewriting any element of $\mathbb{C}^{d+1}$ in the basis $a_i = \sum_j P_i(j)$ computing $a_i a_j = \delta_{ij} a_i$ and extending bilinearly.

Because $P$ can be derived from $Q$, the Bose-Mesner algebra of $\mathcal{A}$ is also characterized by $Q$. Moreover, if $\mathcal{A}$ is symmetric, then $\mathbb{C}$ can be replaced by $\mathbb{R}$ throughout the previous discussion.

However, more can be said about the characterization of the two products on $\mathbb{A}$. In particular, the usual product can be defined in terms of the basis of Schur idempotents where

$$A_i A_j = \sum_{k=0}^{d} p_{ij}^k A_k \ .$$

These $p_{ij}^k$ are called the INTERSECTION NUMBERS of $\mathcal{A}$. Similarly, we can define numbers $q_{ij}^k$, called the KREIN PARAMETERS, such that

$$F_i \circ F_j = \frac{1}{n} \sum_{k=0}^{d} q_{ij}^k F_k \ .$$

These dual families of parameters can also be derived from $P$ or $Q$ in the following manner.

Since the $A_i$ are Schur orthogonal idempotents, we can pick out the $p_{ij}^k$ by multiplying $(A_i A_j) \circ A_k = p_{ij}^k A_k$. Summing over all entries of this product, $\operatorname{sum} p_{ij}^k A_k = p_{ij}^k n \nu_k$. However, since the sum of entries in a Schur product is the inner product of the factors,

$$\operatorname{trace} A_k^* A_i A_j = \operatorname{trace} \left( \sum_{r=0}^{d} P_i(r) P_j(r) \overline{P_k(r)} F_r \right) = \sum_{r=0}^{d} P_i(r) P_j(r) \overline{P_k(r)} \mu_r \ ,$$

where $\overline{P_k(r)}$ is the complex conjugate of $P_k(r)$. (This is done by rewriting each of $A_i, A_j, A_k$

in terms of the $F_r$ and expanding the product.) In the case that $\mathcal{A}$ is symmetric $P$ is real, so that $\overline{P_k(r)} = P_k(r)$. Then, it is clear that $p_{ij}^k$ remains unchanged when the indices $i, j, k$ are permuted.

Similarly, $(F_i \circ F_j)F_k = q_{ij}^k F_k$, and $F_k$ is self-adjoint, so taking the trace $\operatorname{trace} q_{ij}^k F_k = q_{ij}^k \mu_k$. However, since the trace of a product is the inner product of the factors,

$$\operatorname{sum} F_k \circ F_i \circ F_j = \operatorname{sum}\left(\frac{1}{n^3}\sum_{r=0}^{d} Q_i(r)Q_j(r)Q_k(r)A_r\right) = \frac{1}{n^2}\sum_{r=0}^{d} Q_i(r)Q_j(r)Q_k(r)\nu_r \ .$$

As with the intersection numbers, $q_{ij}^k$ remains unchanged when the indices $i, j, k$ are permuted.

(TODO? Derive $P$ from the $p_{ij}^k$?)



Thus far, almost purely algebraic techniques have been used to derive results. There two last dual pairs of results: the first is purely algebraic, while the second comes from graph theory.

**Proposition 2.1.4**
*The first columns of $P$ and $Q$ are the all-ones vector, $\mathbf{1}$.*

*Proof.* The first column of $P$ holds the eigenvalues of $A_0$, which by definition is equal to $I$. Therefore, $A_0 F_j = P_0(j)F_j = F_j$, so $P_0(j) = 1$.

The first column of $Q$ corresponds to $F_0 = \frac{1}{n}J$. Therefore, $F_0 \circ A_i = \frac{1}{n}Q_0(i)A_i$, so $Q_0(j) = 1$. □

Note that because $P^T \operatorname{diag}(\mu) = \operatorname{diag}(\nu)\, Q$, the each of the two results above could be derived from the other.

**Proposition 2.1.5**
*For all $i, j \in D$, we have $|P_i(j)| \leq \nu_i$.*

*Proof.* We know that for each $i$, we have $A_i\mathbf{1} = \nu_i\mathbf{1}$. Let $x$ be a $\theta$-eigenvector of $A_i$, scaled so that its largest component (say, $x_u$) has modulus 1; let $\theta$ have largest modulus among the eigenvalues of $A_i$. Then $|(A_i x)_u| = |(\theta x)_u| = |\theta|$, and

$$|\theta| = |(A_i x)_u| \leq \sum_v (A_i)_{vu}\,|x_v| \leq \sum_v (A_i)_{vu}\,|x_u| = (A_i\mathbf{1})_u = \nu_i \ .$$

The proposition follows, as each $P_i(j)$ is an eigenvalue of $A_i$. □

Note that $A_i\mathbf{1}$ is the vector of row sums of $A_i$; the sum of each row is precisely the out-degrees of the relation (directed graph) of $A_i$. Therefore, this proposition shows that the eigenvalues of a directed graph all have modulus less than the maximum out-degree. In the case of a regular graph, this is simply the valency.

**Corollary 2.1.6**
*For all $i, j \in D$, we have $|Q_j(i)| \leq \mu_j$.*

*Proof.* This follows from the previous proposition, as $Q_j(i) = \frac{\mu_j}{\nu_i} P_i(j)$, and both the valencies and multiplicities are all positive integers. □

This completes the list of standard properties of the Bose-Mesner algebra and its duality. The results are summarized in the table on the next page.

Table 2.1: Dual Properties of the Bose-Mesner Algebra

| | | |
|---|---|---|
| 2.1.1 | $I = A_0$ | $\frac{1}{n}J = F_0$ |
| 2.1.2 | $J = \sum_{i=0}^{d} A_i$ | $I = \sum_{j=0}^{d} F_j$ |
| 2.1.3 | $A_i \circ A_j = \delta_{ij} A_i$ | $F_i F_j = \delta_{ij} F_j$ |
| 2.1.4 | $A_i = \sum_{j=0}^{d} P_i(j) F_j$ | $F_j = \frac{1}{n} \sum_{i=0}^{d} Q_j(i) A_i$ |
| 2.1.5 | $A_i F_j = P_i(j) F_j$ | $F_j \circ A_i = \frac{1}{n} Q_j(i) A_i$ |
| 2.1.6 | $P(0) = \nu^T$ | $Q(0) = \mu^T$ |
| 2.1.7 | $P_0 = \mathbf{1}$ | $Q_0 = \mathbf{1}$ |
| 2.1.8 | $\sum_{i=0}^{d} P_i(j) = n\delta_{0j}$ | $\sum_{j=0}^{d} Q_j(i) = n\delta_{0i}$ |
| 2.1.9 | $\lvert P_i(j) \rvert \leq \nu_i$ | $\lvert Q_j(i) \rvert \leq \mu_j$ |
| 2.1.10 | $P \operatorname{diag}(\nu)^{-1} P^T = n \operatorname{diag}(\mu)^{-1}$ | $Q \operatorname{diag}(\mu)^{-1} Q^T = n \operatorname{diag}(\nu)^{-1}$ |
| 2.1.11 | $PQ = nI$ | |
| 2.1.12 | $P^T \operatorname{diag}(\mu) = \operatorname{diag}(\nu) Q$ | |
| 2.1.13 | $A_i A_j = \sum_{k=0}^{d} p_{ij}^k A_k$ | $F_i \circ F_j = \frac{1}{n} \sum_{k=0}^{d} q_{ij}^k F_k$ |
| 2.1.14 | $p_{ij}^k = \frac{1}{n\nu_k} \sum_{r=0}^{d} P_i(r) P_j(r) \overline{P_k(r)} \mu_r$ | $q_{ij}^k = \frac{1}{n^2 \mu_k} \sum_{r=0}^{d} Q_i(r) Q_j(r) Q_k(r) \nu_r$ |

## 2.2 $P$-Polynomial Schemes

- Definitions
- "Equivalence" to DRGs

Our motivation in this report for the definition of an association scheme was the class of distance-regular graphs. However, the class of association schemes – even symmetric ones – is much more general. In this section, we will investigate a restricted class of symmetric association schemes: the $P$-polynomial schemes.

Note that in this section, and for the remainder of this report, all association schemes will be symmetric unless otherwise specified.

**Definition 2.2.1**
*An association scheme of adjacency matrices $A_0, A_1, \ldots, A_d$ is* $P$-POLYNOMIAL *if they can be ordered such that $A_i$ is a polynomial in $A_1$ of degree $i$.*

It happens that every distance-regular graph forms a $P$-polynomial scheme, and that every $P$-polynomial scheme arises in this way.

To demonstrate that distance-regular graphs generate $P$-polynomial schemes, we show that each distance matrix $A_i$ is a degree-$i$ polynomial in $A_1$ by induction. In order to do this, we need to be able to write $A_{i+1}$ in terms of the matrices at distance $i' \leq i$.

**Lemma 2.2.2**
*Let $\Gamma$ be a distance-regular graph with diameter $d$, vertex set $V$, and adjacency matrices $A_0, A_1, \ldots, A_d$. Then for all $i, j$,*

$$A_i A_j = \sum_{k=0}^{d} p_{ij}^k A_k$$

*where $p_{ij}^k = |\{w \in V \mid d(u,w) = i \quad \text{and} \quad d(w,v) = j\}|$ for fixed vertices $u$ and $v$ such that $d(u,v) = k$.*

*Proof.* Let $\Gamma$ be a distance-regular graph with vertex set $V$, diameter $d$, and distance graphs $\Gamma_0, \Gamma_1, \ldots, \Gamma_d$ with adjacency matrices $A_0, A_1, \ldots, A_d$. Then for vertices $u, v$ at distance $k$,

$$\begin{aligned} p_{ij}^k &= |N_i(u) \cap N_j(v)| \\ &= \{w \in V \mid d(u,w) = i \quad \text{and} \quad d(w,v) = j\} \\ &= \{w \in V \mid (u,w) \in \Gamma_i \quad \text{and} \quad (w,v) \in \Gamma_j\} \ . \end{aligned}$$

Then $A_i A_j = \sum_k p_{ij}^k A_k$ as shown in the equivalence between the two definitions of an association scheme (2.1.2, 2.1.1). □

**Lemma 2.2.3**
*Let $\Gamma$ be a distance-regular graph with diameter $d$ and adjacency matrices $A_0, A_1, \ldots, A_d$. Then for all $i$,*

$$A_i A_1 = \sum_{k=0}^{d} p_{i1}^k A_k$$

*where $p_{i1}^{i+1} \neq 0$, and $p_{i1}^k = 0$ for all $k > i+1$.*

*Proof.* If $\Gamma$ has diameter $d$, then there is a shortest path of length $d$. Note that every subpath of a shortest path is also a shortest path, so that there exists a shortest path of every length $k \leq d$.

Let $u, v$ be vertices at distance $i+1$, and consider a shortest path between them. On this path from $u$ to $v$, consider the first vertex $w$ after $u$. Then $d(u,w) = 1$, and since the remaining portion of the path, from $w$ to $v$ is also a shortest path, $d(w,v) = i$. Therefore,

$$|\{w \in V \mid d(u,w) = 1 \quad \text{and} \quad d(w,v) = i\}| = p_{1i}^{i+1} = p_{i1}^{i+1} > 0 \ .$$

Now consider $u, v$ at distance any $k$, and $w \in N_1(u) \cap N_i(v)$. By the triangle inequality,

$$k = d(u,v) \leq d(u,w) + d(w,v) = 1 + i \ ,$$

so that if $k > i+1$, then $p_{i1}^k = 0$. □

**Theorem 2.2.4**
*Every distance-regular graph $\Gamma$ forms a $P$-polynomial association scheme $\mathcal{A}$, with the distance graphs of $\Gamma$ for classes. Moreover, the diameter of $\Gamma$ is the diameter of $\mathcal{A}$.*

*Proof.* Let $\Gamma$ be a distance-regular graph with diameter $d$, distance graphs $\Gamma_0, \Gamma_1, \ldots, \Gamma_d$, and distance matrices $A_0, A_1, \ldots, A_d$. From lemma (2.2), the product of distance matrices of a distance-regular graph belongs to the span of distance matrices. Since the distance matrices are symmetric, the rest of the axioms follow. Therefore, $\Gamma$ is an association scheme.

Note that $A_0 = I$ is a polynomial of degree 0 in $A_1$, and $A_1$ is a polynomial of degree 1 in itself. Then, supposing $A_i$ is a polynomial of degree $i$ in $A_1$, we have that $A_i A_1$ is a polynomial of degree $i+1$. Then by lemma (2.2.3),

$$A_i A_1 = \sum_{k=0}^{i+1} p_{i1}^k A_k \implies A_{i+1} = \frac{1}{p_{i1}^{i+1}} \left( A_i A_1 - \sum_{k=0}^{i} p_{i1}^k A_k \right) \ ,$$

which demonstrates that $A_{i+1}$ is a polynomial of degree $i+1$ in $A_1$. □

### 2.2.1 $Q$-Polynomial Schemes

## 2.3 Automorphisms and Cayley Graphs

This section may be merged with the following section.

- Action of a regular group of automorphisms
- Cayley graphs
- Eigenspaces from characters

The material in this section follows mostly from [3, Chapter 9].

This section and the next describes a special class of symmetric graphs (respectively, association schemes). As with many other mathematical structures, the *symmetry* of graphs (or schemes) is made precise by examining its automorphisms – those transformations of the object in question which leave its structure unchanged. Graphs (or schemes) with certain automorphisms may be classified in this way.

More importantly for the purposes of this report, the structure revealed by the automorphisms of graphs (or schemes) allows one to compute their eigenvalues significantly more efficiently than otherwise would be the case, as outlined in the previous sections.

**Definition 2.3.1** (Automorphism)
*For graphs* $\Gamma, \Gamma'$*, a map* $\varphi : V(\Gamma) \to V(\Gamma')$ *is a* HOMOMORPHISM *if*

$$\forall u, v \in V(\Gamma)\ \ u \sim_\Gamma v \implies \varphi(u) \sim_{\Gamma'} \varphi(v)\ .$$

*An* ISOMORPHISM *is an invertible homomorphism whose inverse is also a homomorphism; an* AUTOMORPHISM *is an isomorphism from a graph to itself.* $\operatorname{Aut}\Gamma$ *denotes the set of all automorphisms on* $\Gamma$*; it is the subgroup of* $\operatorname{Sym} V(\Gamma)$*, consisting of those permutations which preserve the (edge) structure of* $\Gamma$*.*

*An automorphism of an association scheme* $\mathcal{A}$ *on vertex set* $X$ *is a map* $X \to X$ *which is simultaneously an automorphism of every class in the scheme. In other words,* $\operatorname{Aut}\mathcal{A}$ *is the intersection of the automorphism groups of each class.*

**Definition 2.3.2** (Cayley Graphs)
*Given any group* $G$ *and a subset* $C \subseteq G$*, then* $C$ *is* INVERSE-CLOSED *if for each* $g \in C$*,* $g^{-1} \in C$ *as well.*

*If* $C \subseteq G$ *is an inverse-closed subset of a group* $G$*, then the* CAYLEY GRAPH *of* $G$ *with respect to* $C$ *is denoted* $\operatorname{Cay}(G, C)$*, and defined as follows:*

- *Its vertex set is* $G$

- $g \sim h$ *in* $\mathrm{Cay}(G, C)$ *if and only if* $gh^{-1} \in C$

*Since* $C$ *is inverse closed,* $gh^{-1} \in C \iff hg^{-1} \in C$ *so that* $\mathrm{Cay}(G, C)$ *is undirected.*

*Furthermore, if* $1_G \notin C$*, then* $g \not\sim g$ *so that the graph is loopless. (By definition it already lacks parallel edges.)*

Because Cayley graphs are defined from groups using only the group structure, it is intuitive that these graphs should be highly symmetric. For example, every Cayley graph is VERTEX TRANSITIVE: for every pair $u, v$ in the vertex set, there is a automorphism taking $u \mapsto v$. To see this, note that $G$ acts (B.1.1) on $\mathrm{Cay}(G, C)$ through the group operation, since the vertex set is also the group. To verify that this is a homomorphism, if $u \sim v$ in $\mathrm{Cay}(G, C)$, then

$$uv^{-1} \in C \implies (ug)(vg)^{-1} = ugg^{-1}v^{-1} = u1v^{-1} = uv^{-1} \in C$$

so that $ug \sim vg$. Then, $\mathrm{Cay}(G, C)$ is clearly vertex transitive if $G$ acts transitively, and for any vertices $u, v$, the group element $u^{-1}v$ maps $u$ to $v$. Moreover, this action is free, since if $ug = u$, then the group cancellation law implies that $g$ is trivial. Together, this implies that the action of $G$ is regular, which suggests the following lemma which provides a characterization of Cayley graphs.

(This action is also faithful since for $g \neq h \in G$, the vertex 1 gets mapped to $g$ and $h$ respectively, which are unequal. However this observation irrelevant for this lemma, since it restricts to the action of an automorphism group, which is automatically faithful.)

**Lemma 2.3.3**
*For a graph* $\Gamma$*, there exists a subgroup* $G \leq \mathrm{Aut}\,\Gamma$ *which acts regularly on* $\Gamma$ *if and only if* $\Gamma \cong \mathrm{Cay}(G, C)$ *for some inverse-closed* $C \subseteq G$*.*

Since the above argument demonstrates the reverse implication, only the forward direction will be shown here.

Before beginning the proof, it will be worthwhile to note the neighbours of $1_G$ in $\mathrm{Cay}(G, C)$: $g \sim 1_G$ precisely when $g1^{-1} = g \in C$.

*Proof.* Choose a vertex $v \in V(\Gamma)$ to identify with $1_G$. (This choice will not matter in the end, as Cayley graphs are vertex transitive.) Since the action is regular, for each $u \in V(\Gamma)$ there exists a unique $g_u \in G$ such that $vg_u = u$ (B.1.4).

Then define

$$C := \{g \in G \mid vg \sim v\}$$

and observe that for $u, w \in V(\Gamma)$, $ug_u^{-1} = v$, and $w = vg_w \implies wg_u^{-1} = vg_wg_u^{-1}$. So, since

$g_u^{-1}$ is an automorphism of $\Gamma$,

$$u \sim w \iff ug_u^{-1} \sim wg_u^{-1} \iff v \sim vg_wg_u^{-1} \iff g_wg_u^{-1} \in C \ .$$

Therefore, the map $u \mapsto g_u$ is the desired isomorphism $\Gamma \to \mathrm{Cay}(G, C)$. □

As promised at the beginning of the section, the next lemma demonstrates (for graphs) how automorphisms may be used to derive eigenvalues, and moreover, their eigenvectors. Naively, computing the eigenvalues of a matrix $A$ involves solving its characteristic polynomial, which is generically difficult. Then for an eigenvalue $\theta$, finding a $\theta$-eigenvector involves computing the kernel of $A - \theta I$, which can be computed in polynomial time (though not in linear time), and fast numeric algorithms are typically inexact. (TODO citation)

However, given the right information about a group, the following result finds the eigenvectors and eigenvalues almost instantaneously.

**Lemma 2.3.4**
*Let $G$ be a finite abelian group, let $C \subseteq G \setminus \{1\}$ be inverse-closed, and define $\Gamma := \mathrm{Cay}(G, C)$. Then the rows of the character table of $G$ provide a complete set of eigenvectors for the adjacency matrix $A$ of $\Gamma$. Specifically, if $\psi$ is a character of $G$ (equivalently, a row of its character table), then $\psi(C)$ is the eigenvalue of $\psi$.*

*Proof.* Note first that the neighbours $h \sim g$ of a vertex $g \in G$ consist of precisely the set $\{cg \mid c \in C\} = Cg$ since $h \sim g \iff hg^{-1} \in C$, and multiplication by $g$ is invertible.

As in (B.1), characters are identified with row vectors such that $\psi(g) \rightsquigarrow \psi_g$.

Then

$$(A\psi)_g = \sum_{h \in G} A_{g,h}\psi(h) = \sum_{h \sim g} \psi(h) = \sum_{c \in C} \psi(cg) = \psi(g) \sum_{c \in C} \psi(c) = \psi_g \psi(C) \ .$$

Furthermore, since the rows of the character table are orthogonal, the eigenvectors $\psi$ are linearly independent, and since $G \cong G^*$ (B.2.2) implies that the character table is square, the rows form a basis of eigenvectors for $A$. □

TODO How to get real eigenvectors out of this?

## 2.4 Partitions and Translation Schemes

- Equitable partitions of matrices
- Group partitions yielding association schemes
- Dual schemes? (Interesting, but not particularly necessary for the rest of this report)

In a sense, this section generalizes the characterization of Cayley graphs from the previous section to the setting of association schemes. Throughout this section, a transitive, abelian group of automorphisms will replace the regular automorphism group which corresponds to a Cayley graph. As per (B.1.3) the transitive, abelian group will act regularly, so that (2.3.3) still applies. This motivates the following definition.

**Definition 2.4.1** (Translation Schemes)
*A* TRANSLATION SCHEME *is an association scheme whose automorphism group contains a transitive, abelian subgroup.*

**Lemma 2.4.2**
*If $\mathcal{A}$ is a translation scheme, and $G$ is a transitive, abelian automorphism group, then there is a partition into inverse-closed sets $C_0, C_1, \ldots, C_d$ of $G$ where $C_0 = \{1\}$, and each graph $\Gamma_i$ in $\mathcal{A}$ is isomorphic to* $\mathrm{Cay}(G, C_i)$.

*Proof.* Since $G$ is abelian and is a transitive subgroup of $\mathrm{Aut}\,\Gamma_i$ for each $i = 0, 1, \ldots, d$, $G$ acts regularly on $\Gamma_i$. Therefore, by (2.3.3), there exists an inverse-closed set $C_i \subseteq G$ such that $\Gamma_i \cong \mathrm{Cay}(G, C_i)$.

In particular, since the edges of $\Gamma_0$ are the diagonal relation, $C_0 = \{1\}$ generates the graph.

Otherwise, it suffices to show that $C_0, C_1, \ldots, C_d$ partition $G$. Recall from the proof of (2.3.3) that any vertex may be chosen to identify with $1_G$, so that the same vertex (say, $v$) may be chosen for each graph $\Gamma_i$ without loss of generality, in which case $C_i$ consists of the neighbours of $v$. By the definition of an association scheme, for each vertex $u$ there is exactly one graph $\Gamma_i$ in which $u \sim v$, so that for each vertex, there is exactly one $C_i$ containing it. □

In order to characterize the translation schemes in a similar manner to the Cayley graphs, an examination of partitions of matrices and groups will be required. This will lead to a simple criterion that distinguishes those partitions which generate a translation scheme translation scheme from those which do not. [3, Section 12.10]

**Definition 2.4.3** (Partition Matrix)
*If $\sigma$ is a partition of a set $X$, then the* PARTITION MATRIX *of $\sigma$ is the* 01 *matrix whose rows are indexed by the elements of $X$, and whose columns are indexed by the parts of $\sigma$, in which each row – corresponding to $x \in X$ – has exactly one* 1*, in the column corresponding to the part that contains $x$.*

Any partition matrix may be obtained from an $X \times X$ identity matrix by merging the columns which correspond to elements in the same part. Note that this implies that the columns are linearly independent. (The rows will **not** be linearly independent unless the

partition is induced by the diagonal relation.)

**Definition 2.4.4** (Induced Row Partition)
*Given a matrix $H$, if $\sigma$ is a partition of the columns with partition matrix $\chi(\sigma)$ then the* INDUCED ROW PARTITION *$\sigma^*$ is the partition of the rows of $H$ such that two rows are in the same part if and only if the corresponding rows in $H\chi(\sigma)$ are equal.*

In other words, if $f$ is the function which maps each row index $i$ of $H$ to the row vector $(H\chi(\sigma))_i$, then $\sigma^*$ is the partition given by the fibres of $f$. [3, Section 12.7]

**Theorem 2.4.5** (Bridges and Mena [3, Theorem 12.10.1])
*Let $G$ be a finite abelian group, let $\sigma = \{C_0, C_1, \ldots, C_d\}$ be a partition of $G$ into inverse-closed parts where $C_0 = \{1\}$, and let $\sigma^*$ be the induced row partition of the character table $H$ of $G$.*

*Then $|\sigma^*| \geq |\sigma|$, and the graphs $\Gamma_i := \mathrm{Cay}(G, C_i)$ form the classes of an association scheme if and only if $|\sigma^*| = |\sigma|$.*

*Proof.* Let $A_i$ be the adjacency matrix of $\Gamma_i$, and observe that the set $\{A_0, A_1, \ldots, A_d\}$ is linearly independent. This is because the sets $C_i$ partition $G$, and in each $\Gamma_i$ the set $C_i$ consists of the neighbours of 1. The fact that the $C_i$ partition $G$ also implies that $\sum_i A_i = J$, and since $C_0 = \{1\}$, $A_0 = I$.

By (2.3.4), each character $\psi$ of $G$ (i.e. row of $H$) is a common eigenvector of $A_0, A_1, \ldots, A_d$, with eigenvalue $\psi(C_i)$ at $A_i$. Define $\mathbb{A} := \mathrm{span}\{A_0, A_1, \ldots, A_d\}$. Let $\chi_{C_i}$ be the characteristic vector of $C_i$ in $G$, and let

$$\chi(\sigma) = \begin{bmatrix} | & | & & | \\ \chi_{C_0} & \chi_{C_1} & \cdots & \chi_{C_d} \\ | & | & & | \end{bmatrix} \tag{2.9}$$

be the partition matrix of $\sigma$.

Let $D_0, D_1, \ldots, D_e$ be the parts of $\sigma^*$; then $i, k$ (or, their characters $\psi^i, \psi^k$) belong to the same part $D_j$ precisely when the rows $\psi^i\chi(\sigma), \psi^k\chi(\sigma)$ in

$$H\chi(\sigma) = \begin{bmatrix} — & \psi^1 & — \\ & \vdots & \\ — & \psi^n & — \end{bmatrix} \begin{bmatrix} | & | & & | \\ \chi_{C_0} & \chi_{C_1} & \cdots & \chi_{C_d} \\ | & | & & | \end{bmatrix} \tag{2.10}$$

are equal. Together, the characters of each $D_j$ span a common eigenspace of the $A_i$: let $F_j$ be the orthogonal projection matrix onto this subspace.

Define $\mathbb{F} := \mathrm{span}\{F_0, F_1, \ldots, F_e\}$. Since the $\mathrm{col}\, F_j$ are spanned by disjoint sets of characters, the subspaces are orthogonal and the $F_j$ are linearly independent; since together the

characters span $\mathbb{C}^n$ (where $n$ is the order of $G$), the (direct) sum of the subspaces is $\mathbb{C}^n$ as well. Therefore,

$$I = F_0 + F_1 + \cdots + F_e \ .$$

Furthermore, since $\operatorname{col} F_j$ is a common eigenspace for the $A_i$, for $i = 0, 1, \ldots, d$ and $j = 0, 1, \ldots, e$ there exist constants $P_i(j)$ such that

$$A_i F_j = P_i(j) F_j \implies A_i = \sum_{j=0}^{e} P_i(j) F_j \implies \mathbb{A} \leq \mathbb{F} \ . \tag{2.11}$$

This implies that

$$|\sigma| = d = \dim \mathbb{A} \leq \dim \mathbb{F} = e = |\sigma^*| \ .$$

Note that the $F_0, F_1, \ldots, F_e$ are orthogonal idempotents, so they are closed under the regular matrix product. This implies that the algebra they generate is simply $\mathbb{F}$. On the other hand, while $A_0, A_1, \ldots, A_d$ are orthogonal idempotents with respect to the *Schur product*, they may generate an algebra with the usual product that is strictly larger than $\mathbb{A}$ – it must, however, be contained in $\mathbb{F}$. We will show that these two algebras are actually equal. In this case, $e = d$ if and only if $\mathbb{A}$ is closed under regular matrix multiplication; given the results above, this will then be true if and only if $A_0, A_1, \ldots, A_d$ forms an association scheme.

From (TODO reference) and (2.11), if $g(x)$ is any polynomial, then $g(A_i) = \sum_j g(P_i(j)) F_j$. In particular, if $x_0^{s_0} x_1^{s_1} \cdots x_d^{s_d}$ is any monomial, $A_i^{s_i} = \sum_j P_i(j)^{s_i} F_j$ as above, so that evaluating the monomial at $(A_0, A_1, \ldots, A_d)$ yields

$$A_0^{s_0} A_1^{s_1} \cdots A_d^{s_d} = \prod_i \sum_j P_i(j)^{s_i} F_j = \sum_j \left( \prod_i P_i(j)^{s_i} \right) F_j$$

since the $F_j$ are orthogonal idempotents. Since any polynomial $g$ in $d+1$ variables is a linear combination of such monomials, it follows that

$$\begin{aligned} g(A_0, A_1, \ldots, A_d) &= \sum_j g(P_0(j), P_1(j), \ldots, P_d(j)) F_j \\ \implies g(A_0, A_1, \ldots, A_d) F_j &= g(P_0(j), P_1(j), \ldots, P_d(j)) F_j \end{aligned}$$

for all $j = 0, 1, \ldots, e$.

Now let $P$ be the $(e+1) \times (d+1)$ matrix such that $P_{ji} = P_i(j)$. Note that the rows of $P$ are precisely the distinct rows of $H\chi(\sigma)$, so that for any two rows $j \neq j'$ of $P$, there exists a column $i(j, j')$ such that $P_{i(j,j')}(j) \neq P_{i(j,j')}(j')$. This allows for the definition of

the polynomials

$$g_j(x_0, x_1, \ldots, x_d) := \prod_{j' \neq j} \left(x_{i(j,j')} - P_{i(j,j')}(j')\right)$$

so that when applied at $A_0, A_1, \ldots, A_d$,

$$\begin{aligned} g_j(A_0, A_1, \ldots, A_d)F_{j''} &:= \prod_{j' \neq j} \left(P_{i(j,j')}(j'') - P_{i(j,j')}(j')\right) F_{j''} \\ &= g_j\left(P_0(j''), P_1(j''), \ldots, P_d(j'')\right) F_{j''} \ . \end{aligned}$$

By construction, if $j'' = j$, then $f_j := g_j\left(P_0(j''), P_1(j''), \ldots, P_d(j'')\right)$ will be non-zero, but if $j'' \neq j$, then there will be some $j' = j''$ at which $P_{i(j,j')}(j'') - P_{i(j,j')}(j') = 0$ so that $g_j\left(P_0(j''), P_1(j''), \ldots, P_d(j'')\right) = 0$.

This construction demonstrates that for each $j$, there exists a polynomial $g_j$ such that

$$g_j(A_0, A_1, \ldots, A_d) = \sum_{j'} f_j \delta_{jj'} F_{j'} = f_j F_j$$

where $f_j \neq 0$, so that $F_j$ can be written as a polynomial in $A_0, A_1, \ldots, A_d$. This proves that each $F_j$ is contained in the algebra generated by $\mathbb{A}$, and so all of $\mathbb{F}$ is contained in this algebra. Since the reverse inclusion was already shown, this proves that the two algebras are equal, as desired. □

It is interesting to note that, while the character table $H$ may in general be complex, each of the orthogonal projection matrices $F_j$ is real. To see this, note that the $A_i$ are real, symmetric matrices, so that their eigenvalues $P_i(j)$ are real as well. Then, each of the polynomials $g_j$ (used to express $F_j$ in the algebra generated by the $A_i$) must also be real, since they were defined in terms of the $P_i(j)$. Therefore, not only are the $\mathbb{C}$-algebras of $\mathbb{A}$ and $\mathbb{F}$ equal, but so are their $\mathbb{R}$-algebras.

With this result, translation schemes are characterized by a finite abelian group $G$ and partition $\sigma$ satisfying the condition given. Moreover, a finite abelian group $G$ is isomorphic to its groups of characters, $G^*$ (B.2.2), and the group of characters is completely described by the character table $H$. Likewise, the group partition $\sigma$ is completely described by its partition matrix $\chi(\sigma)$. Since the condition in (2.4.5) depends only on these two matrices (both with respect to the same ordering on $G$), if it is satisfied, then the matrices completely describe the translation scheme they generate.

## 2.5 The Eigenvalues of the Hamming Scheme

The theory just developed in the previous section can be applied immediately to the Hamming scheme. This scheme is of great utility in the setting of coding theory, in part because it is a $P$-polynomial scheme, generated by the distance-regular Hamming graph. Moreover, it is also a translation scheme, with respect to a particularly nice group, and a simple partition, which will allow us to deduce a formula for the eigenvalues of the scheme. In the particular case of the Hamming graph, an explicit expression for its eigenvalues can be given.

To this end, let $\mathbb{Z}_q$ denote the cyclic group of order $q$ (written additively), and consider the direct product $\mathbb{Z}_q^d$ with subsets

$$C_i := \left\{ x \in \mathbb{Z}_q^d \;\middle|\; \text{there are exactly } i \text{ 0's in } x \right\} .$$

In particular, $C_0 = \{0\}$ and

$$C_1 = \mathbb{Z}_q \setminus \{0\} \times \{0\}^{d-1} \bigsqcup \{0\} \times \mathbb{Z}_q \setminus \{0\} \times \{0\}^{d-2} \bigsqcup \cdots \bigsqcup \{0\}^{d-1} \times \mathbb{Z}_q \setminus \{0\} .$$

Then $x, y$ are $i^{\text{th}}$ associates if and only if $x - y \in C_i$; that is, the Hamming distance between $x$ and $y$ is $i$, so that this partition yields the Hamming scheme. In particular, $H(d, q) \cong \mathrm{Cay}(\mathbb{Z}_q^d, C_1)$.

For a character of the group $\psi \in (\mathbb{Z}_q^d)^*$ (B.2.1) and $x = (x_1, \ldots, x_d) = \sum_{i=1}^d x_i e_i$ in the group (here $e_i$ is the tuple of all zeroes, and a 1 in the $i^{\text{th}}$ spot), then since $\psi$ is a homomorphism,

$$\psi(x) = \prod_{i=1}^d \psi(x_i e_i) = \prod_{i=1}^d \psi\left( \sum_{j=1}^{x_i} e_i \right) = \prod_{i=1}^d \prod_{j=1}^{x_i} \psi(e_i) = \prod_{i=1}^d \psi(e_i)^{x_i}$$

so that $\psi$ is completely determined by its values on $e_1, \ldots, e_d$.

Let $\omega$ be a primitive $q^{\text{th}}$ root of unity, so that $1 = \omega^0, \omega^1, \ldots, \omega^{q-1}$ are distinct. Then every choice in $\left\{\omega^0, \omega^1, \ldots, \omega^{q-1}\right\}^d$ will yield a distinct character by assigning the $i^{\text{th}}$ entry to $\psi(e_i)$ (identifying $\psi$ with the tuple as in (B.1)), and defining the value of $\psi$ at all other $x = (x_1, \ldots, x_d)$ by

$$\psi(x) := \prod_{i=1}^d \psi(e_i)^{x_i} .$$

Note that this is a homomorphism since

$$\psi(x+y) = \prod_{i=1}^d \psi(e_i)^{x_i + y_i} = \prod_{i=1}^d \psi(e_i)^{x_i} \psi(e_i)^{y_i} = \prod_{i=1}^d \psi(e_i)^{x_i} \prod_{i=1}^d \psi(e_i)^{y_i} = \psi(x)\psi(y) .$$

Therefore, $(\mathbb{Z}_q^d)^* \cong \left\{\omega^0, \ldots, \omega^{q-1}\right\}^n$ (taking entrywise multiplication as the group product on the right), so that the characters $\psi$ will be identified with row vectors

$$\psi \rightsquigarrow \begin{bmatrix} \omega^{\psi_1} & \cdots & \omega^{\psi_d} \end{bmatrix}$$

where $\psi_i \in \{0, 1, \ldots, q-1\}$.

(Note that this notation deviates from (B.1), but will be more convenient for this purpose. In fact, this shows that $(\mathbb{Z}_q^d)^* \cong \mathbb{Z}_q^d$ directly, confirming (B.2.2).)

Then, in the $i^{\text{th}}$-distance Hamming graph, $\psi$ is a $\psi(C_i)$-eigenvalue (2.3.4), and for the Hamming graph, it can be computed directly.

$$\begin{aligned}
\psi(C_1) &= \sum_{c \in C_1} \psi(c) \\
&= \sum_{i=1}^{d} \sum_{j=1}^{q-1} \psi(je_i) \qquad \text{here, the outer sum picks which entry of } c \text{ will be non-zero} \\
&\qquad\qquad\qquad\qquad\qquad\qquad \text{and the inner sum picks the value} \\
&= \sum_{i=1}^{d} \sum_{j=1}^{q-1} \psi(e_i)^j = \sum_{i=1}^{d} \sum_{j=1}^{q-1} \left(\omega^{\psi_i}\right)^j \\
&= \sum_{i=1}^{d} \left( \frac{1 - \left(\omega^{\psi_i}\right)^q}{1 - \omega^{\psi_i}} - 1 \right) \qquad \text{using the usual formula for geometric sums} \\
&= \sum_{i=1}^{d} (q-1)\delta_{0,\psi_i} - d \\
&= (q-1)\,(\text{the number of } i \text{ with } \psi_i = 0) - d
\end{aligned}$$

Since the number $k$ of indices $i$ for which $\psi_i = 0$ can vary from $0, 1, \ldots, d$, and there are $\binom{d}{k}$ places $i$ at which $\psi_i = 0$ and $(q-1)^{d-k}$ choices for the other $\psi_j$, the eigenvalues of the Hamming graph are given

$$\begin{cases} qk - d & k = 0, 1, \ldots, d \\ \binom{d}{k}(q-1)^{d-k} & \text{(multiplicity)} \end{cases}. \tag{2.12}$$

For the other graphs in the Hamming scheme, their eigenvalues can be computed as $\psi(C_i)$, although there may not be a particularly nice expression for this sum.

# 3. Delsarte's Linear Programming Bound

## 3.1 Linear Programming

- Basics of Linear Programming – done
- Duality – done
- Algorithms?

The terminology and results from this section, except for the adjective *principal* for constraints, follows from [4].

A LINEAR PROGRAMMING PROBLEM (or LINEAR PROGRAM) is an optimization problem in which one seeks to maximize or minimize a linear function of one or more variables, subject to linear constraints. That is, fixing a vector $c$, one tries to maximize or minimize the linear combinations of the components of $c$:

$$c_1x_1 + \cdots + c_nx_n = c^Tx$$

for some $x$. Note that maximizing $c^Tx$ is equivalent to minimizing $(-c)^Tx$, so that for the theory of linear programming, it suffices to consider maximization problems without loss of generality. As in other optimization problems, the function to be maximized ($c^Tx$ in this case) is called the OBJECTIVE (FUNCTION).

In most cases, there will be constraints on the inputs to the objective function, and for the purposes of linear programming these will also have to be linear. That is, there will be a matrix $A$ and vector $b$ such that only inputs $x$ satisfying $Ax \leq b$ will be allowed. (Note that for *vectors* $a$ and $b$, $a \leq b$ will mean that each component $a_i$ is less than or equal to the corresponding component $b_i$.) These are called the (PRINCIPAL) CONSTRAINTS, and vectors $x$ which satisfy the constraints will be called FEASIBLE (SOLUTIONS). (Note that in [1], the term *program* is used to refer to a feasible solution.)

If there are no constraints on the problem (and even in some cases where there are) through appropriate choices of feasible solution $x$, the objective $c^Tx$ may be made arbitrarily

large, and such problems are called UNBOUNDED. Conversely, if no feasible solutions exist, then the problem is called INFEASIBLE.

Finally, in most applications of linear programming – in particular to the cliques of association schemes – the feasible solutions will be further constrained to those with all non-negative components (i.e. $x \geq 0$). These are called the NON-NEGATIVITY CONSTRAINTS, in contrast with the *principal constraints*. The non-negativity constraints will be required throughout the remainder of this report.

Therefore, for an objective $c^T x$ and constraints $Ax \leq b$, the associated linear program will be written in STANDARD FORM:

$$\max\left\{c^T x \;\middle|\; Ax \leq b,\ x \geq 0\right\} \ .$$

### 3.1.1 Duality

The most important observation about linear programs (for the purposes of this report, at least) is that they come in dual pairs.

Given a linear program $\mathcal{P}$ written in standard form

$$\max\left\{c^T x \;\middle|\; Ax \leq b,\ x \geq 0\right\}$$

its DUAL program is $\mathcal{P}^*$:

$$\min\left\{b^T y \;\middle|\; A^T y \geq c,\ y \geq 0\right\} \ .$$

Re-writing it in standard form,

$$\max\left\{(-b)^T y \;\middle|\; -A^T y \leq -c,\ y \geq 0\right\}$$

taking the dual

$$\min\left\{-c^T x \;\middle|\; -Ax \geq -b,\ x \geq 0\right\}$$

and re-writing in standard form

$$\max\left\{c^T x \;\middle|\; Ax \leq b,\ x \geq 0\right\}$$

the original (called PRIMAL) linear program is recovered.

This demonstrates that $(\mathcal{P}^*)^* = \mathcal{P}$, so that linear programs come in dual pairs.

**Theorem 3.1.1** (Weak Duality)
*If $x$ is a feasible solution to a linear program*

$$\max\left\{c^T x \;\middle|\; Ax \leq b,\ x \geq 0\right\},$$

*and $y$ is a feasible solution to its dual program,*

$$\min\left\{b^T y \;\middle|\; A^T y \geq c,\ y \geq 0\right\},$$

*then $c^T x \leq b^T y$.*

*Proof.* Let $u, v, w$ be vectors with $u \geq 0$, and $v \leq w$. Then for all components $i$, $u_i \geq 0$ and $v_i \leq w_i$ implies that $u_i v_i \leq u_i w_i$ so that

$$u^T v = \sum_i u_i v_i \leq \sum_i u_i w_i = u^T w \ .$$

In particular, since $y$ is a feasible solution to the dual program, and $x \geq 0$,

$$c \leq A^T y \implies x^T c \leq x^T A^T y = y^T A x \ .$$

(Here one may take the transpose of the whole expression, since the result is a scalar.) Similarly, since $x$ is a feasible solution to the primal program, and $y \geq 0$,

$$b \geq Ax \implies y^T b \geq y^T A x \ .$$

By combining the two inequalities,

$$b^T y = y^T b \geq y^T A x \geq x^T c = c^T x$$

which is the desired result. □

As a result of the weak duality of linear programs, every feasible solution to the dual program provides an upper bound on the maximum of the primal, and every feasible solution to the primal program provides a lower bound on the minimum of the dual.



In fact the extremal values of dual programs (the maximum of the primal, and the minimum of the dual) coincide, although this will not be needed for the purposes of this report. This is referred to as *Strong Duality* of linear programs.

## 3.2 The LP Bound

**Definition 3.2.1**

*The* INNER DISTRIBUTION *is TODO. I might also put this in the section on association schemes; I'm not sure if it belongs better there or here.*

**Theorem 3.2.2** (Delsarte Thm 3.3 [1])
*For any inner distribution $y$,*

$$Q^T y \geq 0$$

*where $Q$ is the matrix of dual eigenvalues. (Here, $x \geq 0$ means that each component of the vector $x$ is not less than 0.)*

*Proof.* TODO. Note that this will require a number of lemmas which I've omitted here for brevity, but will include in the final product. □

This theorem provides the key inequality that will allow the application of linear programming to cliques in association schemes. However, because the constraint vector in a primal linear program becomes the objective in the dual program, this inequality will require some transformation to make it suitable for use in linear programming.

Let $Y$ be an $M$-clique with inner distribution $y$. Then $y_i = 0$ for all $i \notin M$, so $Q^T y \geq 0 \iff Q^T \operatorname{diag}(\chi_M)\, y \geq 0$ since the action of $\operatorname{diag}(\chi_M)$ acting on the left is to zero out the *rows* of $y$ with index not in $M$. Similarly,

$$Q^T \operatorname{diag}(\chi_M)\, y = Q(0)^T y_0 + Q^T \operatorname{diag}(\chi_{M^*})\, y = \mu + Q^T \operatorname{diag}(\chi_{M^*})\, y$$

since the action of $\operatorname{diag}(\chi_{M^*})$ on the right is to zero out the *columns* of $Q^T$ with index not in $M^*$, $y_0 = 1$, and

$$Q^T = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ \mu_1 & & & \\ \vdots & & * & \\ \mu_d & & & \end{bmatrix}.$$

Finally, since $y \geq 0$, $Q_0^T y \geq 0$ adds no new constraint, so that under the non-negativity constraint $Q^T y \geq 0 \iff \operatorname{diag}(\chi_{N^*})\, Q^T y \geq 0$.

Putting all this together, $Q^T y \geq 0 \iff \operatorname{diag}(\chi_{N^*})\, Q^T \operatorname{diag}(\chi_{M^*})\, y \geq -\mu$ so that Delsarte's LP can be written in standard form:

$$\max\left\{\mathbf{1}^T \operatorname{diag}(\chi_M)\, y \;\middle|\; Q^T \operatorname{diag}(\chi_M)\, y \geq 0,\ y \geq 0,\ y_0 = 1\right\} \tag{3.1}$$

$$= \max\left\{\chi_{M^*}^T y \;\middle|\; -\operatorname{diag}(\chi_{N^*})\, Q^T \operatorname{diag}(\chi_{M^*})\, y \leq \operatorname{diag}(\chi_{N^*})\, \mu,\ y \geq 0\right\} + 1\ . \tag{3.2}$$

Taking the dual yields

$$\min\left\{\mu^T \operatorname{diag}(\chi_{N^*})\, z \;\middle|\; -\operatorname{diag}(\chi_{M^*})\, Q \operatorname{diag}(\chi_{N^*})\, z \geq \chi_{M^*},\ z \geq 0\right\} + 1 \tag{3.3}$$

$$= \min\left\{\mu^T z \;\middle|\; -\operatorname{diag}(\chi_{M^*})\, Q \operatorname{diag}(\chi_{N^*})\, z \geq \chi_{M^*},\ z \geq 0,\ z_0 = 1\right\}\ . \tag{3.4}$$

Therefore, if $z_0 = 1$ is required, recalling that $Q_0 = \mathbf{1}$ and $\operatorname{diag}(\chi_{M^*})\,\mathbf{1} = \chi_{M^*}$, then

$$\begin{aligned}
&\operatorname{diag}(\chi_{M^*})\, Q \operatorname{diag}(\chi_N)\, z \\
=&\operatorname{diag}(\chi_{M^*})\,(Q_0 z_0 + Q \operatorname{diag}(\chi_{N^*})\, z) \\
=&\chi_{M^*} + \operatorname{diag}(\chi_{M^*})\, Q \operatorname{diag}(\chi_{N^*})\, z \\
\leq&0\ .
\end{aligned}$$

This equivalence recovers Delsarte's formulation of the dual linear program:

$$\min\left\{\mu^T z \;\middle|\; \operatorname{diag}(\chi_{M^*})\, Qz \leq 0,\ z \geq 0,\ z_0 = 1\right\}\ . \tag{3.5}$$

## 3.3 The Ratio Bound

We will use (3.5) frequently.

## 3.4 The Clique-Coclique Bound

# 4. Schrijver's SDP Bound

## 4.1 The Terwilliger Algebra of the Hamming Scheme

## 4.2 Semi-Definite Programming

# 5. Computation

I wasn't sure if I ought to mention anything about the code I've written for this project (or even if there's anything worth saying that won't be covered elsewhere in the report).

Also, if there are some specific results that would be interesting to show, but do not fit naturally into other sections of the report, then perhaps they could go here as well.

- Computing the character table of an abelian group

# 6. Discussion

I'm not sure if "Discussion" is the right name for a chapter of this sort, if it is even worth including. If it is, I would try and keep this part brief.

## 6.1 Conclusion

I know papers typically have some sort of conclusion or summary towards the end, but I wasn't sure if it would be valuable to include something like that in a report of this kind.

## 6.2 Other Applications

Perhaps it might be worth mentioning number of other applications of association schemes, for example to design theory, or statistics?

# A. Linear Algebra

## A.1 The Spectral Theorem

### A.1.1 Orthogonal Projection

- The trace of an idempotent matrix is its rank.

### A.1.2 Spectral Decomposition

## A.2 Adjacency Matrices

Basic results about the spectra of adjacency matrices, which may be used elsewhere in the report. E.g. the sum of eigenvalues with multiplicity, and consequences.

## A.3 Positive Semi-Definite Matrices

Depending on which proof of the clique-coclique bound I use, and how much detail I go into Schrijver's SDP bound, I could make some comments about PSD matrices.

# B. Group Theory

## B.1 Group Actions

The material of this section comes primarily from [2, Section 1.7, Chapter 4].

**Definition B.1.1** (Group Action)
*Given a group $G$ and a set $X$,* GROUP ACTION *is a homomorphism $G \to \operatorname{Sym} X$, where $\operatorname{Sym} X$ is the symmetric group on $X$.*

A group action $\varphi : G \to \operatorname{Sym} X$ induces a product $X \times G \to X$ by mapping $(x, g) \mapsto \varphi(g)(x)$. When the action is clear from context, this will be denoted $x \cdot g$, or simply $xg$. This is called a RIGHT ACTION, as *$g$ acts on the right of $x$* (the corresponding notion of a LEFT ACTION can also be defined.)

Conversely, given a product $X \times G \to X$, the same expression defines a map $G \to \operatorname{Sym} X$. If such a product satisfies

$$\forall x \in X \; x1_G = x$$
$$\text{and } \forall x \in X \; \forall g, h \in G \; (xg)h = x(gh)$$

then the induced map $G \to \operatorname{Sym} X$ will be a homomorphism, so that these definitions are equivalent.

(In [2] this is taken as the definition of a group action, and the homomorphism $G \to \operatorname{Sym} X$ is called its PERMUTATION REPRESENTATION. It will be occasionally convenient to adopt each perspective.)

**Definition B.1.2** (Types of Group Actions)
*If if a homomorphism $G \to \operatorname{Sym} X$ is injective, then the action is called* FAITHFUL. *Note that a group homomorphism is injective if and only if it has a trivial kernel.*

*Given a group action $G \to \operatorname{Sym} X$, $g \in G$ is called* FIXED POINT-FREE *if $\forall x \in X \; xg \neq x$. The group action itself is called* FIXED POINT-FREE *(or just* FREE*) if all its nontrivial elements are fixed point-free.*

*A group action $G \to \operatorname{Sym} X$ is called* TRANSITIVE *if $\forall x, y \in X$ there exists some $g \in G$ such that $xg = y$.*

*A group action is called* REGULAR *if it is simultaneously transitive and free. (This terminology follows [3].)*

Note that if $X$ is a structure with automorphisms (such as a graph or group), $G$ is a subgroup of $\operatorname{Aut} X$, and $G$ acts in the natural way on $X$ (i.e. $xg = g(x)$), then this action is faithful. That is, $\operatorname{Aut} X \leq \operatorname{Sym} X$, so that this action is induced by the identity $G \hookrightarrow \operatorname{Sym} X$, which is clearly injective.

**Lemma B.1.3**

*If an abelian group $G$ acts faithfully and transitively on a set $X$, then the action is free, and thus also regular. [2, Section 4.1, Exercise 3]*

*Proof.* Let $g \in G$ be nontrivial, and $x \in X$. The goal is to prove that $xg \neq x$.

Since $g$ is not the identity, there exists some $y \in X$ such that $z := yg \neq y$. Furthermore, since $G$ acts transitively on $X$, there exists some $h \in G$ such that $yh = x \iff y = xh^{-1}$. Then,

$$\begin{aligned} xg &= (yh)g \\ &= y(hg) \\ &= y(gh) \quad \text{since } G \text{ is abelian} \\ &= (yg)h \\ &= zh\ . \end{aligned}$$

If $zh = x$ then, $z = xh^{-1} = y$, but by definition, $z = yg \neq y$, so $xg = zh \neq x$. □

An alternate characterization of regular actions will be useful in this report. To see this, note that for a pair $x, y \in X$, there exists a $g \in G$ such that $xg = y$ by transitivity; for any $g' \in G$ satisfying $xg' = y$,

$$xg = xg' \implies x = xg'g^{-1}$$

so that $g'g^{-1} = 1$ since the action is free, and so $g' = g$. Conversely, if for each $x, y \in X$ there existed a unique $g \in G$ satisfying $xg = y$, then the action would clearly be transitive; since $x1 = x$, 1 is the unique group element fixing any point, so the action must be free.

**Lemma B.1.4**

*A group action $G$ on $X$ is regular if and only if for all $x, y \in X$, there exists a unique $g \in G$ such that $xg = y$.*

## B.2 Character Theory

**Definition B.2.1** (Characters)
*Given a group $G$, a* CHARACTER *of the group $G$ is a homomorphism $G \to \mathbb{C}^\times$, the group of non-zero complex numbers under multiplication. Then $G^*$ will denote the set of characters of $G$. [3, Chapter 8]*

*(For abelian groups, this corresponds to irreducible degree 1 characters over $\mathbb{C}$ in [2, Section 18.3].)*

TODO Maybe use $\circ$ for this product to mimic the usage for the Schur product?

On $G^*$ a product of characters can be defined by setting

$$\varphi\psi : g \mapsto \varphi(g)\psi(g)$$

under which the character taking each $g \in G$ identically to 1 acts as identity.

Furthermore, for any character $\psi \in G^*$, the map $g \mapsto \psi\left(g^{-1}\right)$ is a homomorphism since

$$gh \longmapsto \psi\left((gh)^{-1}\right) = \psi\left(h^{-1}g^{-1}\right) = \psi\left(h^{-1}\right)\psi\left(g^{-1}\right) = \psi\left(g^{-1}\right)\psi\left(h^{-1}\right)$$

and for any $g \in G$,

$$\psi(g)\psi\left(g^{-1}\right) = \psi\left(g^{-1}\right)\psi(g) = \psi(1) = 1$$

so that this homomorphism is an inverse for $\psi$.

Therefore, $G^*$ forms a group under the above product of characters.

For the remainder of this section (and the rest of this report), discussion of characters will be restricted to the case of finite abelian groups. Throughout this section, $G$ will denote a finite abelian group of order $n$.

In this case, by Lagrange's theorem, $g^n = 1_G$ for every $g \in G$, and so for any character $\psi \in G^*$

$$1 = \psi(1) = \psi\left(g^n\right) = \psi(g)^n$$

– that is, the image of each character is contained in the set of $n^{\text{th}}$ roots of unity.

Since the inverse of a complex number with modulus 1 is also its complex conjugate, looking at the inversion in $G^*$,

$$\psi\left(g^{-1}\right) = \psi(g)^{-1} = \overline{\psi(g)}$$

so that the inverse of $\psi \in G^*$ is $\overline{\psi} : g \mapsto \overline{\psi(g)}$.

**Theorem B.2.2**

*For all finite abelian groups $G$,*

$$G \cong G^* \ .$$

*Proof.* TODO □

While by transitivity this shows that $G^{**} \cong G$, this can be seen more directly via the isomorphism

$$g \mapsto (\psi \mapsto \psi(g)) \ .$$

Given an ordering $G = \{g_1, \ldots, g_n\}$, the character $\psi \in G^*$ can be identified with the row vector

$$\psi \rightsquigarrow \begin{bmatrix} \psi(g_1) & \cdots & \psi(g_n) \end{bmatrix} \ . \tag{B.1}$$

With this identification, the product of characters becomes the entrywise product of vectors (the *Schur product* of $n \times 1$ matrices), and the inverse of a character in $G^*$ is the entrywise inversion of the vector.

Furthermore, given an ordering $G^* = \{\psi^1, \ldots, \psi^n\}$, the matrix whose rows consist of the characters of $G^*$ is called the CHARACTER TABLE of $G$:

$$H = \begin{bmatrix} — & \psi^1 & — \\ & \vdots & \\ — & \psi^n & — \end{bmatrix} \ . \tag{B.2}$$

Remarkably, this matrix turns out to be (almost) unitary.

For any subset $C \subseteq G$, define

$$\psi(C) := \sum_{g \in C} \psi(g) = \psi \chi_C$$

where $\chi_C$ is the characteristic vector of $C$ in $G$ (with the same ordering). So, given characters $\psi, \varphi$, their inner product can be written

$$\psi \varphi^* = \sum_{g \in G} \psi(g) \overline{\varphi}(g) = \sum_{g \in G} (\psi \overline{\varphi})(g) = (\psi \overline{\varphi}) \chi_G = (\psi \overline{\varphi})(G) \ .$$

(Note here that $\varphi^*$ denotes the conjugate transpose of $\varphi$, and $\chi_G$ is also the all-ones column vector $\mathbf{1}$.)

**Lemma B.2.3**

*For any $\psi \in G^*$*

$$\psi(G) = \begin{cases} |G| & \text{if } \psi \text{ is the identity of } G^* \\ 0 & \text{else.} \end{cases}$$

*Proof.* For any $h \in G$, since $g \mapsto hg$ is an automorphism of $G$, $hG = G$, so that

$$\psi(G) = \sum_{g \in G} \psi(g) = \sum_{g \in G} \psi(hg) = \sum_{g \in G} \psi(h)\psi(g) = \psi(h) \sum_{g \in G} \psi(g) = \psi(h)\psi(G)$$

which implies that either $\psi(h) = 1$ or $\psi(G) = 0$.

But this holds for arbitrary $h \in G$, so that either

$$\forall h \in G \ \psi(h) = 1 \implies \psi = 1_{G^*} \quad \text{and} \quad \psi(G) = \sum_{g \in G} 1 = |G|$$

or else

$$\exists h \in G \ \psi(h) \neq 1 \implies \psi(G) = 0 \ .$$

□

**Corollary B.2.4**
*If $H$ is the character table of a finite abelian group $G$ of order $n$, then $HH^* = nI$, where $H^*$ is the conjugate transpose of $H$.*

*Proof.* If $\psi, \varphi$ are characters of $G$, and $\psi \neq \varphi$, then letting $\theta = \psi\overline{\varphi}$, the $(\psi, \varphi)$-entry of $HH^*$ is given by $\psi\varphi^* = \theta(G) = 0$, since $\theta$ is not the identity of $G^*$.

However, for the diagonal, $(\psi, \psi)$-entries of $HH^*$, $\psi\psi^* = 1_{G^*}(G) = n$, which proves the claim. □

## B.3 The Structure of Finite Abelian Groups

TODO I need a citation for this. I'm also not sure if this section shouldn't go before the section on Character Theory, as it will be used in (B.2.2).

**Theorem B.3.1** (Structure Theorem)
*If $G$ is a finitely generated abelian group, then $G$ is isomorphic to a direct product of cyclic groups. Specifically,*

$$G \cong \mathbb{Z}_{q_1}^{d_1} \times \cdots \times \mathbb{Z}_{q_t}^{d_t} \times \mathbb{Z}^f$$

*where the $q_i$ are distinct prime powers. Moreover, this decomposition is unique up to the ordering of its factors.*

This is a well-known and well-used result. It comes as a direct result of the Struc-

ture Theorem for Finitely Generated Modules over PIDs, using the language of rings and modules, but this is out of the scope of this report.

# C. Notation

I've included these sections mostly as an excuse to add the citations to the bibliography, though it may be somewhat useful to have a sort of guide to the similarities and differences in notation in this report, as well as in the references. (At least, I would find such a thing useful.)

## C.1 Godsil

See [3] Ch. 10.

## C.2 Delsarte

See [1] Ch. 3.

## C.3 Schrijver

See [5].

# Bibliography


[1] P. Delsarte. "An algebraic approach to the association schemes of coding theory". PhD thesis. 1973.

[2] David Steven. Dummit. *Abstract algebra*. eng. 3rd ed. New York: Wiley, 2004. ISBN: 0471433349.

[3] C. Godsil. *Algebraic Combinatorics*. Chapman Hall/CRC Mathematics Series. Taylor & Francis, 1993. ISBN: 9780412041310.

[4] Jiří Matoušek. *Understanding and using linear programming*. eng. Universitext. Berlin ; Springer, 2007.

[5] Alexander Schrijver. "New Code Upper Bounds From the Terwilliger Algebra and Semidefinite Programming". In: *Information Theory, IEEE Transactions on* 51 (Sept. 2005), pp. 2859–2866. DOI: 10.1109/TIT.2005.851748.