Masterarbeit

Semidefinite Bounds for Unequal Error Protection Codes

Daniel Brosch

17. Oktober 2018

Mathematisches Institut
Universität zu Köln

Erstprüfer: Prof. Dr. Frank Vallentin
Zweitprüfer: Prof. Dr. Alexander Lytchak

Contents

Chapter 1

Error correcting codes

Whenever data is transmitted or stored, errors can happen randomly, with usually little to no control over it. It happens when reading data from CDs, using the internet, satellite transmissions and even when sending data from space probes. Often, especially when it is slow or even impossible to request the same data again, one uses error correcting codes to be able to correct most errors.

The most basic code just repeats the same bits multiple times: If there is a zero in the data, we could instead send five zeroes right after another. Even if two bits of every message, forty percent of the received data, are incorrect, we can still be quite sure what the original message was. While this would be very easy to implement, it was shown early on it is a very inefficient way to do error correction. Instead one uses an *error correcting code* C, a subset of $\{0, \ldots, q-1\}^n$ such that every element, called *word*, has a minimum *Hamming distance* of $d \in \mathbb{N}$ to every other word:

$$\mathrm{d}(x, y) = |\{i \in \{1, \ldots, n\} \mid x_i \neq y_i\}| \geq d$$

If we always decode received words as the nearest word of the code, we can correct up to $\lfloor \frac{d-1}{2} \rfloor$ errors.

The goal is then to maximize the amount of words at a certain minimum distance and block length n, or to maximize the *rate* $\frac{\log_q |C|}{n}$ for a fixed fractional minimum distance $\frac{\min_{x \neq y} \mathrm{d}(x,y)}{n}$. Codes of this type were discussed by plenty of authors before, and new developments are still being made.

This thesis will take a look at a closely related problem, where we have two sets of data: one of them small but important, the other larger but less important. Examples include control signals compared to the payload data for internet/phone/satellite connections, as well as TCP headers and their data sections. Another example are multiple resolution source codes, which send data (for example video or audio streams) at multiple different levels of quality at once, such that the receiver can always decode it at a level according to his connection speed and quality. Here it makes sense to protect the lower resolution more, since the amount of data is much lower anyway, allowing everyone to at least reconstruct a crude version of the data.

1.1 Unequal error protection

In the papers [1] and [2] the authors introduce two variants of unequal error protection (UEP) codes, and prove multiple bounds for asymptotic rates of these codes. The first variant is called *bit-wise* UEP, in which every message is split into two parts, a "special", important part and a less important, usually larger part. We denote an element of this code as a pair $(x, y) \in \{0, \ldots, q-1\}^n$, were x is the decoded first part of the message, and y the second. In general it does not imply the words of the code need to have this split structure, but we do have a bijection

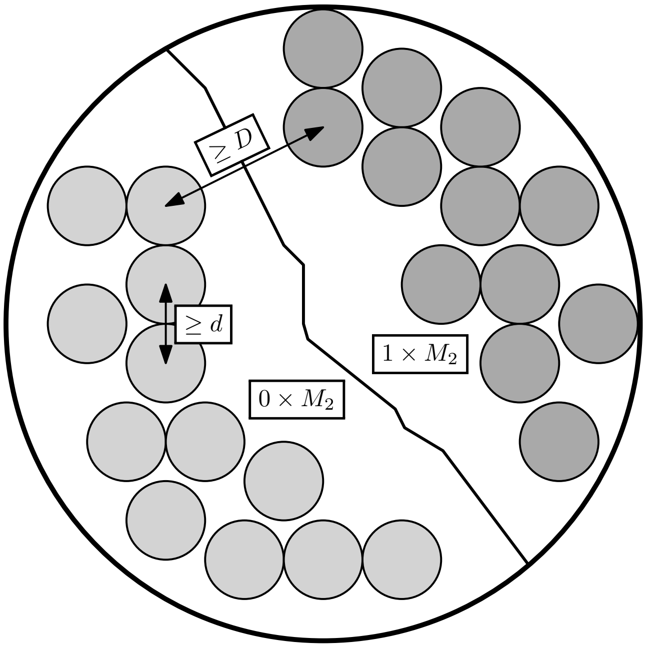
$$M = M_1 \times M_2 \to C$$

from the messages M we want to encode to the code C itself.

To realize the different strengths we want any two words to have a large distance D if their important parts are different:

$$\mathrm{d}\big((c_1, c_2), (c_1', c_2')\big) \geq D \quad \forall c_1, c_2, c_1', c_2' \text{ with } c_1 \neq c_1'$$

and at least a (smaller) distance d if their less important parts are different. For example in the case $M_1 = \{0, 1\}$ we have a single special bit, which means that we want to find two equal sized sets of words with distance D to each other, and distance d between the words of the same set.

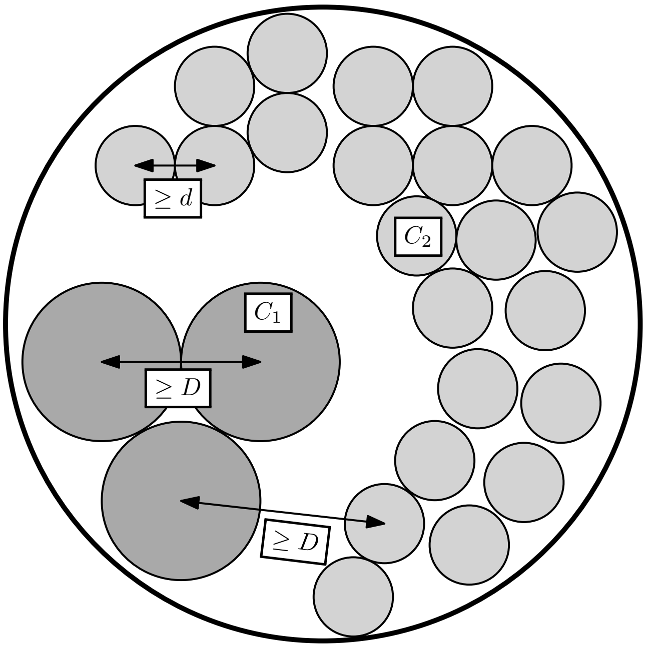


This variant of UEP was explored before by multiple authors (see [1] for a detailed list) for the linear case, since it offers itself well for TCP and multiple resolution codes. For the general one-bit case an upper bound for $\frac{D}{n}$ (which is strictly greater $\frac{d}{n}$ in most cases) was proven in [2] such that the maximum rate of an ordinary block code of fractional distance $\frac{d}{n}$ can still be reached asymptotically.

The second variant, *message-wise* UEP, which is the focus of this thesis, was discussed less often, but does have potential. Here we want to protect certain code words more than others, instead of packing two messages into each word, meaning we split the Code into two sets $C = C_1 \cup C_2$. Here we want all words in C_1 to have minimum distance D to all other words, and the same for the words in C_2 for a smaller distance d:

$$\begin{aligned} \mathrm{d}(c, c') &\geq D \quad \forall c \in C_1, c' \in C \setminus \{c\} \\ \mathrm{d}(c, c') &\geq d \quad \forall c \in C_2, c' \in C \setminus \{c\} \end{aligned}$$

While this might look similar to a multi-size sphere packing at first, notice here that the distance between elements in C_1 and C_2 has to be the larger distance D as between different elements of C_1, instead of the sum of the two radii as in the sphere packing case.



This actually makes it harder to decide when such a code is "good". We cannot just try to fill as much space as possible (by using different weights, as for sphere packings), since that would make it either prefer only small or only large words. In other literature the goal is usually to find a series of codes satisfying two chosen rates at the same time.

Instead we can fix the cardinality of C_1 to be 1, the case of a single special word, and try to maximize the amount of words in C_2 (or maximize D for fixed $|C_2|$). This case was was studied in [2] too, in connection with asymptotic rates. They have proven we can always asymptotically reach the optimal rate of a code with fractional distance $\frac{d}{n}$, even with $D = n$, if we can increase the size q of the alphabet of the code. Furthermore, and more interestingly, they have shown that if we only consider binary alphabets, then we can still reach the optimal rate asymptotically for at least $D = \frac{n}{2}$. This bound is sharp for $d = 0$ and they expect that it is sharp for any $\frac{d}{n} \in [0, \frac{1}{2}]$.

The goal of this thesis will be to study this case further, and determine upper bounds for $|C_2|$ for small values of n, giving us some more insight in how fast we can approach the optimal rate.

1.2 A graph problem

For this thesis we decided to focus on the case message-wise UEP, with a single special word, and a binary alphabet. Because of the symmetry of this problem, we can always set the special word to be the all zero word 0^n. Calling the non-special word set C, we get the following problem:

$$\begin{aligned} A(n, D, d) \coloneqq \max \ & |C| \\ & \mathrm{d}(c_1, c_2) \geq d \quad \forall c_1, c_2 \in C \\ & \mathrm{d}(0^n, c) \geq D \quad \forall c \in C \\ & C \subseteq \{0,1\}^n \end{aligned}$$

For integers $D \geq d$ and dimension n. We can lower the amount of variables by replacing the second condition with a restriction on the set the words are chosen from:

$$\begin{aligned} A(n, D, d) = \max \ & |C| \\ & \mathrm{d}(c_1, c_2) \geq d \quad \forall c_1, c_2 \in C \\ & C \subseteq \{0,1\}^n_{\geq D} \coloneqq \{x \in \{0,1\}^n \ \mid \ |x| \geq D\} \end{aligned}$$

We now want to reformulate this problem as a graph problem, since relaxations for these are well explored. The independence number α of a graph is the largest number of vertices you can choose, such that no two chosen vertices are adjacent. We can easily construct a graph $G_{n,D,d} = (V, E)$ for our problem by setting

$$\begin{aligned} V &= \{0,1\}^n_{\geq D} \\ E &= \{(x, y) \in V \times V \ \mid \ \mathrm{d}(x, y) < d\} \end{aligned}$$

Two vertices of this graph are adjacent, if and only if their Hamming distance is less than d. Hence independent sets in this graph are exactly the feasible codes of our problem:

$$A(n, D, d) = \alpha(G_{n,D,d})$$

Calculating the independence number is NP-complete, and our graph has exponential size, which makes this problem too hard to solve exactly. Instead we will calculate an upper bound for it, the Lovász-Theta number, which is the semidefinite relaxation of the independence number.

1.3 The Lovász-Theta number

The independence number can be reformulated with the matrices $X = \frac{1}{x^T x} x x^T$, where $x \in \{0,1\}^V$ is the characteristic vector of the vertices in C. These matrices are positive semidefinite as they are outer products of vectors with themselves.

Lemma 1.3.1.

$$\begin{aligned} \alpha(G) = \max\ & \langle J, X\rangle \\ & \mathrm{tr}(X) = 1 \\ & X_{ij} = 0 \quad \forall (i,j) \in E \\ & \mathrm{rank}(X) = 1 \\ & X \succcurlyeq 0 \end{aligned}$$

Proof. To verify the upper bound it is enough to set $X = \frac{1}{x^T x} x x^T$, as was mentioned before, which is feasible for this program.

Let X now be a feasible solution of this program. The rank of X is one, so there is a vector a with $X = aa^T$. We have $X_{ij} = a_i a_j = 0$ for every edge of the graph, thereby the support of a is an independent set of the graph. The objective function $\langle J, X\rangle = (e^T a)^2$ is maximized if and only if a is parallel to its own support, so we can now scale the vector a to have only entries in $\{0, 1\}$, let this vector be x. Because of $\mathrm{tr}(X) = a^T a = 1$ we have $a = \frac{x}{\sqrt{x^T x}}$, which lets us reformulate the objective function:

$$\langle J, X\rangle = (e^T a)^2 = \left(\frac{e^T x}{\sqrt{x^T x}}\right)^2 = \frac{1}{x^T x}(e^T x)^2 = e^T x$$

Which is the size of the independent set given by the vector's support □

We can now relax this to a semidefinite program by removing the condition to the rank of the matrix:

Theorem 1.3.2. *Let $G = (V, E)$ be a graph. Then:*

$$\begin{aligned} \alpha(G) \le \vartheta'(G) = \max\ & \langle J, X\rangle & \le \vartheta(G) = \max\ & \langle J, X\rangle \\ & \mathrm{tr}(X) = 1 & & \mathrm{tr}(X) = 1 \\ & X_{ij} = 0 \quad \forall ij \in E & & X_{ij} = 0 \quad \forall ij \in E \\ & X \in S^V_{\succcurlyeq 0} & & X \in S^V_{\succcurlyeq 0} \\ & X \ge 0 & & \end{aligned}$$

The function ϑ is called Lovász theta function, and was first introduced in [8] by László Lovász. The prime variant is a strengthening of the bound towards the independence number, but it does add a condition for every entry of the matrix.

Semidefinite programs can be (approximately) solved in polynomial time, but the size of our graph is still exponential in n. In this case we can simplify the program further by taking into account the symmetry of the problem.

Chapter 2

General symmetry reductions

In this chapter we explain the general approach to take advantage of symmetry of semidefinite programs, following closely Vallentin's approach in [14] and [13]. The section about representation theory additionally has some parts similar to chapter one of Sagan's book [10].

We start with a general semidefinite program of the form

$$\begin{aligned} p = \sup\ & \langle C, X\rangle \\ & \langle A_i, X\rangle = b_i \quad \forall i = 1, \dots, n \\ & X \succcurlyeq 0 \end{aligned}$$

where the matrices are indexed by a set V. For this thesis it is enough to only consider real matrices, but everything can be generalized to complex numbers with slight changes. Let Γ be a group that acts on V by permutations, which we can extend to an operation on matrices by $\pi(X)_{ij} = X_{\pi^{-1}(i)\pi^{-1}(j)}$, which is exactly the matrix $P_\pi^T X P_\pi$ if P_π is the permutation matrix corresponding to π (So $(P_\pi)_{ij} = 1 \Leftrightarrow \pi(i) = j$). We call the program Γ-*invariant*, if $\pi(X)$ is feasible for every feasible X, and $\langle C, X\rangle = \langle C, \pi(X)\rangle$. For our problem we will later see that Γ is exactly the group of graph-automorphisms of our graph.

Because of the convexity of the program we can symmetrize an optimal solution X^* of an invariant program by forming the group average $\frac{1}{|\Gamma|}\sum_{\pi\in\Gamma}\pi(X^*)$ to get another optimal feasible solution. That means that we can restrict our program to only have invariant feasible solutions, that is to say matrices with $\pi(X) = X$ for all $\pi \in \Gamma$. Lets call this subspace of invariant matrices $\mathcal{B}$.

$$\begin{aligned} p = \sup\ & \langle C, X\rangle \\ & \langle A_i, X\rangle = b_i \quad \forall i = 1, \dots, n \\ & X \succcurlyeq 0 \\ & X \in \mathcal{B} \end{aligned}$$

We can define a first basis of $\mathcal{B}$, called the *canonical basis*, by decomposing

$V \times V$ into its Γ-orbits, and defining the basis elements as their characteristic functions:

$$\left(B_{[x,y]}\right)_{i,j} := \begin{cases} 1 \text{ if } (i,j) \in [x,y] \\ 0 \text{ else} \end{cases}$$

where $[x,y] \in \mathbb{V} := {}^{(V \times V)}\!/_{\Gamma}$ is the Γ-orbit of $(x,y) \in V \times V$. It is clear this is a basis for $\mathcal{B}$, and we have $B_{[x,y]}^T = B_{[y,x]}$ by definition. We only consider symmetric matrices in our program, hence we can combine their optimization variables in the following way:

$$\begin{aligned} p = \sup \quad & \sum_{r \in \mathbb{V}} c_r z_r \\ & \sum_{r \in \mathbb{V}} a_{ir} z_r = b_i \quad \forall i = 1, \ldots, n \\ & \sum_{r \in \mathbb{V}} z_r B_r \succcurlyeq 0 \\ & z_r \in \mathbb{R} \quad \forall r \in \mathbb{V} \\ & z_{[x,y]} = z_{[y,x]} \quad \forall [x,y] \in \mathbb{V} \end{aligned}$$

The new coefficients are defined as $c_r := \langle C, B_r \rangle$ and $a_{ir} := \langle A_i, B_r \rangle$, which results in the same objective functions and conditions as before.

The space $\mathcal{B}$ is closed under matrix multiplication, since for invariant matrices A and B we have $\pi(AB) = P_\pi^T A B P_\pi = P_\pi^T A P_\pi P_\pi^T B P_\pi = AB$, so AB is again an invariant matrix. thus this vector space is an algebra, and we will see next section that there is an algebra isomorphism of the form

$$\varphi \colon \mathcal{B} \to \bigoplus_{k=1}^{d} \mathbb{R}^{m_k \times m_k}$$

Applying it to our program is called *block diagonalization*, which results in:

$$\begin{aligned} p = \sup \quad & \sum_{r \in \mathbb{V}} c_r z_r \\ & \sum_{r \in \mathbb{V}} a_{ir} z_r = b_i \quad \forall i = 1, \ldots, n \\ & \sum_{r \in \mathbb{V}} z_r \varphi(B_r) \succcurlyeq 0 \\ & z_r \in \mathbb{R} \quad \forall r \in \mathbb{V} \\ & z_{[x,y]} = z_{[y,x]} \quad \forall [x,y] \in \mathbb{V} \end{aligned} \tag{2.0.1}$$

Often the sum of the m_k is much smaller than the dimension of the original program, significantly reducing the time needed to solve it. In our case we will be able to go from exponential to quadratic size in the length of the code. Furthermore a lot of solvers are able to make use of the block structure of the program, reducing the time needed further.

2.1 Representation theory

If we want to determine this isomorphism, we first need some representation theory. We call a vector space W a Γ-*module*, if there is a group homomorphism

$$\rho\colon \Gamma \to \mathrm{GL}(W)$$

So Γ operates on W with linear transformations. We have seen an example earlier, where we assigned each $\pi \in \Gamma$ a permutation matrix P_π, which operates on $\mathbb{R}^V$.

We call a Γ-module W *irreducible*, if each Γ-submodule of W (Subspaces of W, which are closed under the operations of Γ) is either $\{0\}$ or W itself.

The goal is now to decompose a reducible module into irreducible ones. If we have an inner product $\langle\cdot,\cdot\rangle$ on W and a subspace $U \subset W$, we can form the orthogonal complement of U:

$$U^\perp := \{w \in W \mid \langle u, w\rangle = 0 \quad \forall u \in U\}$$

We have $W = U \oplus U^\perp = U \perp U^\perp$, but we do not know if $U^\perp$ is a submodule, if U is one. If we additionally require the inner product to be Γ-invariant, that is

$$\langle \pi(v), \pi(w)\rangle = \langle v, w\rangle \quad \forall \pi \in \Gamma, v, w \in W$$

we get following proposition:

Proposition 2.1.1. *Let W be a Γ-module, $U \subset W$ a submodule and $\langle\cdot,\cdot\rangle$ an Γ-invariant inner product on W. Then $U^\perp$ is also a submodule.*

Proof. If $\pi \in \Gamma$ and $v \in U^\perp$, then $\pi(v)$ is also in $U^\perp$:

$$\langle \pi(u), v\rangle = \langle \pi^{-1}(\pi(u)), \pi^{-1}(v)\rangle = \langle u, \pi^{-1}(v)\rangle = 0$$

since the inner product is invariant, and $\pi^{-1}(v) \in U$. That means $U^\perp$ is a subspace closed under Γ, and with that a submodule of W. □

Applying this recursively to a reducible module lets us decompose it into irreducible subspaces. We can always find an Γ-invariant inner product (if Γ is finite) by taking the group average $\langle v, w\rangle' = \sum_{\pi\in\Gamma}\langle \pi(v), \pi(w)\rangle$ of any fixed inner product $\langle\cdot,\cdot\rangle$. Alternatively we can fix an invariant inner product beforehand, as we will do later, to get an orthogonal decomposition.

Theorem 2.1.2 (Maschke)**.** *If Γ is a finite group and W a (nonzero) Γ-module, then we can find irreducible submodules W_i of W with*

$$W = W_1 \oplus \ldots \oplus W_k$$

A reducible module can have multiple "copies" of the same irreducible submodule. If we have two Γ-modules H and H' we call them *equivalent*, if there is an Γ-*isometry* $\phi\colon H \to H'$ between them. An isometry is a linear isomorphism, which preserves the action of Γ, $\phi(\pi(v)) = \pi(\phi(v))$, and the inner products, $\langle v, w\rangle = \langle \phi(v), \phi(w)\rangle$.

Let us now fix the module to be $\mathbb{R}^V$, where V is the set of vertices of the graph earlier. The group of symmetries Γ acts on $\mathbb{R}^V$ by $\pi(v)_i = v_{\pi^{-1}(i)}$ (i.e. $\pi(v) = P_\pi v$), and we get an invariant inner product $(v, w) = \frac{1}{|V|}\sum_{i \in V} v_i w_i$ (because Γ just permutes the indices). Applying Maschke's Theorem to this module for this inner product, and sorting the irreducible submodules afterwards gives us following decomposition:

$$\mathbb{R}^V = (H_{1,1} \perp \ldots \perp H_{1,m_1}) \perp \ldots \perp (H_{d,1} \perp \ldots \perp H_{d,m_d}) \tag{2.1.2}$$

Where all the $H_{k,i}$ are irreducible Γ-modules, which are equivalent if and only if their first indices are identical (This decomposition is a special case of the Peter-Weyl Theorem).

We can now take a look at the algebra $\mathcal{A} \subseteq \mathbb{R}^{V \times V}$ generated by the P_π for $\pi \in \Gamma$. The $H_{k,i}$ are closed and irreducible under Γ, so we get $\mathcal{A}|_{H_{k,i}} \cong \mathbb{R}^{h_k \times h_k}$ by fixing a basis, where h_k is the dimension of $H_{k,j}$ for all $j = 1, \ldots, m_k$. We also know that, for a fixed k, the $H_{k,i}$ are all equivalent, so there are isometries between $H_{k,1}$ and $H_{k,i}$ for all i. And thus the elements of $\mathcal{A}$ operate the same on all of them, meaning there is an isomorphism such that the blocks corresponding to the different $H_{k,i}$ are exact copies. Taking into account the full decomposition of $\mathbb{R}^V$ we get a decomposition of $\mathcal{A}$:

$$\mathcal{A} \cong \bigoplus_{k=1}^{d} \mathbb{R}^{h_k \times h_k} \otimes I_{m_k}$$

Why have we done this? Earlier we defined $\mathcal{B}$ as the set of Γ-invariant matrices, that is matrices with $P_\pi^T A P_\pi = A$ for all $\pi \in \Gamma$. That means it is exactly the *commutant* of $\mathcal{A}$:

$$\mathcal{B} = \mathrm{Comm}(\mathcal{A}) = \{X \in \mathbb{R}^{V \times V} \mid YX = XY \quad \forall Y \in \mathcal{A}\}$$

Which means that

$$B \cong \bigoplus_{k=1}^{d} I_{h_k} \otimes \mathbb{R}^{m_k \times m_k}$$

since $\mathrm{Comm}(I_k) = \mathbb{R}^{k \times k}$, $\mathrm{Comm}(\mathbb{R}^{k \times k}) = I_k$ and $(A \otimes B)(C \otimes D) = (AC) \otimes (BD)$ if the products AC and BD are defined. If we then remove the copies of the blocks, we get the existence of the before mentioned isomorphism

$$\varphi \colon \mathcal{B} \to \bigoplus_{k=1}^{d} \mathbb{R}^{m_k \times m_k}$$

for which we now want to find a more explicit formula.

2.2 The Isomorphism in the general case

To determine the isomorphism we want to find a different basis of $\mathcal{B}$, to which we can assign the entries of the new blocks directly. By the decomposition (2.1.2) of $\mathbb{R}^V$, we can fix orthonormal bases $\{e_{k,1,1}, \ldots, e_{k,1,h_k}\}$ of the $H_{k,1}$.

The isometries $\phi_{k,i}\colon H_{k,1} \to H_{k,i}$ preserve the inner products, so applying them to the chosen bases constructs orthonormal bases $e_{k,i,l}$ for the remaining $H_{k,i}$. Such a set of orthonormal bases is called *orthonormal system*, with which we can define a second basis $E_{k,i,j} \in \mathbb{R}^{V\times V}$ of $\mathcal{B}$ by

$$(E_{k,i,j})_{x,y} = \frac{1}{|V|}\sum_{l=1}^{h_k} e_{k,i,l}(x)e_{k,j,l}(y)$$

where $k = 1, \dots, d$ and $i, j = 1, \dots, m_k$.

Theorem 2.2.1. *The $E_{k,i,j}$ form a basis of $\mathcal{B}$ and*

$$E_{k,i,j}E_{k',i',j'} = \delta_{k,k'}\delta_{j,i'}E_{k,i,j'}$$

Proof. We have

$$\begin{aligned}(E_{k,i,j}E_{k',i',j'})_{x,y} &= \frac{1}{|V|^2}\sum_{r=1}^{|V|}\left(\sum_{l=1}^{h_k} e_{k,i,l}(x)e_{k,j,l}(r)\right)\left(\sum_{l'=1}^{h_{k'}} e_{k',i',l'}(r)e_{k',j',l'}(y)\right)\\ &= \frac{1}{|V|^2}\sum_{l=1}^{h_k}\sum_{l'=1}^{h_{k'}} e_{k,i,l}(x)e_{k',j',l'}(y)e_{k,j,l}^T e_{k',i',l'}\\ &= \delta_{k,k'}\delta_{j,i'}(E_{k,i,j'})_{x,y}\end{aligned}$$

since the vectors are orthonormal for $(\cdot,\cdot)$. So the matrices have to be linearly independent and $\dim(\mathcal{B}) = \sum_{k=1}^{d} m_k^2$, as we have seen in the decomposition of $\mathcal{B}$, therefore they form a basis of $\mathcal{B}$. □

Hence we get an algebra isomorphism by

$$\varphi(E_{k,i,j}) = 0 \oplus \ldots \oplus 0 \oplus \underbrace{E_{i,j}}_{k\text{-th block}} \oplus 0 \oplus \ldots \oplus 0$$

where $E_{i,j} \in \mathbb{R}^{m_k,m_k}$ is the matrix with only a one in position (i, j), because

$$\varphi(E_{k,i,j}E_{k',i',j'}) = \delta_{k,k'}\delta_{j,i'}\varphi(E_{k,i,j'}) = \varphi(E_{k,i,j})\varphi(E_{k',i',j'})$$

If we now expand the canonical basis in this basis, we can calculate the matrices $\varphi(B_r)$ needed for the block diagonalization:

$$\begin{aligned}\varphi(B_r) &= \varphi(\sum_{k=1}^{d}\sum_{i,j=1}^{m_k} p_r(k,i,j)E_{k,i,j})\\ &= \sum_{k=1}^{d}\sum_{i,j=1}^{m_k} p_r(k,i,j)\varphi(E_{k,i,j})\end{aligned}$$

It is generally easier easier to express the new basis in the canonical basis, considering we just need to know the value of $E_{k,i,j}$ in one coordinate of each orbit in $\mathbb{V}$:

$$E_{k,i,j} = \sum_{r\in\mathbb{V}} q_{k,i,j}(r)B_r$$

But it turns out that we can easily calculate the coefficients $p_r(k, i, j)$ if we know the $q_{k,i,j}(r)$. For this we first need another orthogonality relation:

Lemma 2.2.2.

$$\sum_{r\in\mathbb{V}} |r| q_{k,i,j}(r) q_{k',i',j'}(r) = \delta_{k,k'}\delta_{j,j'}\delta_{i,i'} h_k$$

Proof. We have

$$\sum_{r\in\mathbb{V}} |r| q_{k,i,j}(r) q_{k',i',j'}(r) = \sum_{x,y\in V} (E_{k,i,j})_{x,y} (E_{k',i',j'})_{x,y}$$

Since the B_r have exactly $|r|$ one-entries, of which none are in two different basis elements. By definition we have $(E_{k',i',j'})_{x,y} = (E_{k',j',i'})_{y,x}$, which gives us

$$= \sum_{x,y\in V} (E_{k,i,j})_{x,y} (E_{k',j',i'})_{y,x} = \sum_{x\in V} (E_{k,i,j} E_{k',j',i'})_{x,x}$$

If we now apply theorem (2.2.1) we can simplify it to:

$$\begin{aligned}
&= \sum_{x\in V} \delta_{k,k'}\delta_{j,j'} (E_{k,i,i'})_{x,x} = \delta_{k,k'}\delta_{j,j'} \operatorname{trace}(E_{k,i,i'}) \\
&= \delta_{k,k'}\delta_{j,j'} \sum_{x\in V} \frac{1}{|V|} \sum_{l=1}^{h_k} e_{k,i,l}(x) e_{k,i',l}(x) = \delta_{k,k'}\delta_{j,j'} \sum_{l=1}^{h_k} (e_{k,i,l}, e_{k,i',l}) \\
&= \delta_{k,k'}\delta_{j,j'}\delta_{i,i'} h_k
\end{aligned}$$

□

This allows us to calculate one set of indices from the other:

Proposition 2.2.3.

$$p_r(k,i,j) = \frac{|r|}{h_k} q_{k,i,j}(r)$$

Proof. By definition of the indices we have:

$$\begin{aligned}
E_{k,i,j} &= \sum_{r\in\mathbb{V}} q_{k,i,j}(r) B_r \\
&= \sum_{r\in\mathbb{V}} q_{k,i,j}(r) \sum_{k'=1}^{d} \sum_{i',j'=1}^{m_{k'}} p_r(k',i',j') E_{k',i',j'}
\end{aligned}$$

Since the $E_{k,i,j}$ form a basis we get

$$\sum_{r\in\mathbb{V}} q_{k,i,j}(r) p_r(k',i',j') = \delta_{k,k'}\delta_{i,i'}\delta_{j,j'}$$

Which is the same orthogonality relation (up to a factor) as in Lemma (2.2.2), hence:

$$p_r(k,i,j) = \frac{|r|}{h_k} q_{k,i,j}(r)$$

□

We now have all the tools for general symmetry reductions we need for this thesis. To simplify our program, the Lovász-Theta number of the graph defined in the first section, we will first have to find the symmetry of our program. Afterwards we will decompose the symmetric matrices into irreducible components, which we will use to determine the basis $E_{k,i,j}$, and more importantly the $p_r(k,i,j)$ explicitly.

Chapter 3

The decomposition

We want to calculate the Lovász theta (prime) number of the graph $G_{n,D,d} = (V, E)$ which was defined as:

$$V = \{0,1\}^n_{\geq D}$$
$$E = \{(x,y) \in V \times V \mid \mathrm{d}(x,y) < d\}$$

The first thing we have to do is to figure out what the group Γ is exactly for our problem, which acts on V by permutations. In the definition of

$$\begin{aligned} \vartheta'(G) = \max \ & \langle J, X \rangle \\ & \mathrm{tr}(X) = 1 \\ & X_{ij} = 0 \quad \forall (i,j) \in E \\ & X \in S^V_{\succcurlyeq 0} \\ & X \geq 0 \end{aligned}$$

the elements of V appear only in one condition: $X_{ij} = 0$, if there is an edge between i and j. Since the program does not differentiate between the edges, we know that Γ is exactly the group of permutations, which sends pairs of vertices to an edge if and only if they were connected before. This group is the set of *graph automorphisms* of $G_{n,D,d}$.

We can find one set of permutations in Γ directly: Permutations that act on the coordinates of elements in $V \subseteq \{0,1\}^n$. These do not change the Hamming distance between vertices, hence they are graph automorphisms since the edge set was defined with it only. Furthermore they do not change the Hamming weight of vertices either, which we will make use of for the decomposition first.

These are in general not all of the automorphisms of the graphs encountered here. For example if $n - D + 1 < d$, the graphs are fully connected. In that case the automorphism group is the permutation group on the vertices (not just on the coordinates of the vertices), but we can also just solve it directly. Since all vertices are connected, every off-diagonal entry of feasible solutions is zero, and the trace is fixed to one, resulting in the same objective value. It does make sense, as we can only fit in one codeword in V, because its diameter (for the Hamming distance) is less than d.

In the case $D = 0$ (which we will allow, even if we set $D \geq d$ earlier, since this case describes ordinary binary codes) we can additionally bit-flip every coordinate of the vertices, since we do not have a special corner that needs to be fixed any more, and it does not change the distance between vertices either.

In the general case we might have more automorphisms in some cases, but the algebra we get by coordinate permutation symmetry was explored well before (e.g. [14], [12]), and will be enough for this thesis.

So let us fix $\Gamma = S_n$ for this thesis, which acts by permutation on the coordinates of elements in $\{0,1\}^n$. For now let $\mathcal{B}$ be the algebra of Γ-invariant $\mathbb{R}^{\{0,1\}^n \times \{0,1\}^n}$ matrices. While this is a larger algebra than the invariant matrices in $\mathbb{R}^{\{0,1\}^n_{\geq D} \times \{0,1\}^n_{\geq D}}$, we will be able to use a diagonalization of $\mathcal{B}$ to get all diagonalizations for the reductions we want later. This algebra $\mathcal{B}$ is called the *Terwilliger algebra of the binary Hamming scheme*.

3.1 Irreducible S_n-modules

The first thing we have to do is to decompose $\mathcal{B}$ into irreducible S_n-modules. To do this, we will introduce explicit constructions for every irreducible S_n-module, which are called *Specht modules*, similarly to Sagan in [10].

To do this we first determine the *conjugacy classes* of S_n. These are defined for $h \in S_n$ as the set $\{ghg^{-1} \mid g \in S_n\}$. We can describe h using (disjunct) cycles as

$$h = (i_1, \ldots, i_l) \ldots (i_m, \ldots, i_n)$$

where the first cycle symbolizes $h(i_1) = i_2, \ldots, h(i_{l-1}) = i_l, h(i_l) = i_1$. Since the cycles are disjunct, we can order them in any way we want, so let us just order them by size, larger cycles first. If we now conjugate h with a $g \in S_n$ we have $(ghg^{-1})(g(i)) = g(h(i))$, that means

$$ghg^{-1} = (g(i_1), \ldots, g(i_l)) \ldots (g(i_m), \ldots, g(i_n))$$

in cycle notation. Since we conjugate h with every element of S_n to form the conjugacy class, it contains every single permutation with the same cycle sizes, that means with the same *cycle type*. Neither the exact arrangement of the elements in the cycles nor the order of the cycles themselves matter, so we can assign every conjugacy class one to one a *partition* $\lambda = (\lambda_1, \ldots, \lambda_t)$ of n, where $\lambda_i \in \mathbb{N}$, $\lambda_i \geq \lambda_{i+1}$ and $\sum_{i=1}^{t} \lambda_i = n$.

Why did we determine the conjugacy classes? The number of different irreducible modules of a group is exactly the number of conjugacy classes. We will not prove it here, since we will not need it for the decomposition later, but it does explain why we start the construction of irreducible modules with a partition λ.

For the actual construction we first need to define the *Young tableaux* of shape λ. They are arrays filled with every integer from 1 to n exactly once, which have l rows of λ_i entries each, aligned on the left. For example we have for $\lambda = (3, 2, 1)$, $n = 6$ the tableau:

$$t = \begin{array}{|c|c|c|}\hline 1 & 3 & 5 \\ \hline 2 & 4 \\ \cline{1-2} 6 \\ \cline{1-1}\end{array}$$

We call two tableaux t_1, t_2 of the same shape *row equivalent*, written $t_1 \sim t_2$, if their corresponding rows contain the same elements. The equivalence classes of Young tableaux are called *Young tabloids*, for example:

$$t = \begin{array}{|c|c|}\hline 1 & 3 \\ \hline 2 & 4 \\ \hline\end{array}, \quad [t] = \left\{ \begin{array}{|c|c|}\hline 1 & 3 \\ \hline 2 & 4 \\ \hline\end{array}, \begin{array}{|c|c|}\hline 1 & 3 \\ \hline 4 & 2 \\ \hline\end{array}, \begin{array}{|c|c|}\hline 3 & 1 \\ \hline 2 & 4 \\ \hline\end{array}, \begin{array}{|c|c|}\hline 3 & 1 \\ \hline 4 & 2 \\ \hline\end{array} \right\} = \begin{array}{cc}\hline 1 & 3 \\ \hline 2 & 4 \\ \hline\end{array}$$

Which we write as arrays with horizontal lines only.

We can define an action of S_n on the tableaux by applying the permutation element wise on all their entries. We can extend this to tabloids as well by $\pi[t] = [\pi(t)]$, which is independent from the choice of t, since the order of elements in each row does not matter after forming the equivalence class again afterwards. This action gives us a first S_n-module (which is generally not irreducible):

$$M^\lambda := \mathbb{R}^T$$

where $T = \{[t_1], \ldots, [t_k]\}$ is the set of all λ-tabloids, so M^λ consists of vectors indexed by the tabloids, and S_n acts on these by permuting the coordinates.

We now want to find an irreducible submodule in M^λ. If we have a tableau t, with columns $C_1, \ldots, C_k$, then we call $C_t := S_{C_1} \times \ldots \times S_{C_k} \subseteq S_n$ the *column-stabilizer* of t. If we have a subset H of S_n, we can construct a vector in $\mathbb{R}[S_n]$, the group ring of S_n over $\mathbb{R}$. It can be seen as $\mathbb{R}^{S_n}$ with multiplication $\left(\sum_{\pi \in S_n} a_\pi \pi\right)\left(\sum_{\sigma \in S_n} b_\sigma \sigma\right) = \sum_{\pi\sigma = \mu} a_\pi b_\sigma \mu$.

$$H^- := \sum_{\pi \in H} \operatorname{sign}(\pi)\pi \in \mathbb{R}[S_n]$$

Since we have an operation of S_n on M^λ, we can extend this to an operation of $\mathbb{R}[S_n]$ on M^λ by

$$\left(\sum_{\pi \in S_n} a_\pi \pi\right)\left(\sum_{\lambda\text{-tabloids } [t]} b_{[t]}[t]\right) = \sum_{\substack{\pi \in S_n \\ \lambda\text{-tabloids } [t]}} a_\pi b_{[t]}[\pi(t)]$$

This allows us to define the *polytabloid* of a tableau t:

$$e_t := C_t^-[t] \in M^\lambda$$

For example if we start with the tableau

$$t = \begin{array}{|c|c|c|}\hline 1 & 2 & 3 \\ \hline 4 & 5 \\ \cline{1-2}\end{array}$$

we get the column stabilizer

$$\begin{aligned}
C_t &= S_{\{1,4\}} \times S_{\{2,5\}} \times S_{\{3\}} \\
C_t^- &= \{\mathrm{id}, (1,4)\}^- \{\mathrm{id}, (2,5)\}^- \\
&= (\mathrm{id} - (1,4))(\mathrm{id} - (2,5)) \\
&= \mathrm{id} - (1,4) - (2,5) + (1,4)(2,5)
\end{aligned}$$

Which means the polytabloid of t is the vector

$$e_t = C_t^-[t] = \begin{array}{|ccc}\hline 1 & 2 & 3 \\ \hline 4 & 5 & \\ \hline\end{array} - \begin{array}{|ccc}\hline 4 & 2 & 3 \\ \hline 1 & 5 & \\ \hline\end{array} - \begin{array}{|ccc}\hline 1 & 5 & 3 \\ \hline 4 & 2 & \\ \hline\end{array} + \begin{array}{|ccc}\hline 4 & 5 & 3 \\ \hline 1 & 2 & \\ \hline\end{array}$$

We defined these polytabloids because they have the useful property to turn into other polytabloids after applying a permutation $\pi \in S_n$ to them:

Lemma 3.1.1.

$$\pi e_t = e_{\pi t}$$

Proof. Let $C_1, \ldots, C_k$ again be the columns of t. We have $C_{\pi t} = \pi C_t \pi^{-1}$, since if $\sigma \in C_t$, then

$$\sigma C_i = C_i \Leftrightarrow \sigma \pi^{-1} \pi C_i = C_i \Leftrightarrow (\pi \sigma \pi^{-1}) \pi C_i = \pi C_i$$

Because the sign of a permutation only depends on its cycle type, we get

$$C_{\pi t}^- = \sum_{\sigma \in C_t} \mathrm{sign}(\sigma) \pi \sigma \pi^{-1} = \pi C_t^- \pi^{-1}$$

Using this proves the lemma:

$$e_{\pi t} = C_{\pi t}^-[\pi t] = \pi C_t^- \pi^{-1}[\pi t] = \pi C_t^-[t] = \pi e_t$$

$\square$

So the polytabloids span a submodule of M^λ, the *Specht module* S^λ. Furthermore it is *cyclic*, generated by any polytabloid, since we can always find a permutation π for a tableau t' of the same shape with $\pi t = t'$, implying S^λ is the smallest submodule of M^λ which contains a polytabloid. We will now show that these modules are irreducible.

Let $\langle \cdot, \cdot \rangle$ be the standard inner product on M^λ, that is the unique inner product with $\langle [s], [t] \rangle = \delta_{[s],[t]}$. First we need some properties of this product and the $(\cdot)^-$ operation from earlier.

Lemma 3.1.2 (Sign Lemma)**.** *If $H \subseteq S_n$ is a subgroup, then the following properties apply:*

(i) $\forall \pi \in H\colon \quad \pi H^- = H^- \pi = \mathrm{sign}(\pi) H^-$

(ii) $\forall u, v \in M^\lambda\colon \quad \langle H^- u, v \rangle = \langle u, H^- v \rangle$

(iii) $(b,c) \in H \Rightarrow \exists k \in \mathbb{R}[S_n]: \quad H^- = k(\mathrm{id} - (b,c))$

(iv) If $(b,c) \in H$ *and* $(b,c)[t] = [t]$*, then* $H^-[t] = 0$*.*

Proof. We use $\mathrm{sign}(\pi^{-1})\mathrm{sign}(\pi) = \mathrm{sign}(\pi\pi^{-1}) = 1$ throughout this proof.

(i) Let $\pi \in H$. We then have by definition:

$$\pi H^- = \sum_{\sigma \in H} \mathrm{sign}(\sigma)\pi\sigma = \sum_{\pi^{-1}\sigma \in H} \mathrm{sign}(\pi^{-1}\sigma)\sigma = \mathrm{sign}(\pi)H^-$$

The case $H^-\pi$ works analogously.

(ii) The inner product is S_n invariant, since $\langle \pi[s], \pi[t] \rangle = \delta_{\pi[s],\pi[t]} = \delta_{[s],[t]} = \langle [s],[t] \rangle$, so we have:

$$\langle H^- u, v \rangle = \sum_{\pi \in H} \mathrm{sign}(\pi)\langle \pi u, v \rangle = \sum_{\pi \in H} \mathrm{sign}(\pi)\langle u, \pi^{-1} v \rangle = \langle u, H^- v \rangle$$

(iii) We have a subgroup $K = \{\mathrm{id}, (b,c)\} \subseteq H$. This subgroup can be used to decompose H into disjoint K-orbits, of which we can choose a set of representatives $k_1, \ldots, k_l$, called a *transversal*:

$$H = \dot{\bigcup} k_i K$$

Which gives us the factor we want:

$$H^- = \sum_{i=1}^{l} k_i^- K^- = \left(\sum_{i=1}^{l} k_i^- \right) (\mathrm{id} - (b,c))$$

Here we applied $(\cdot)^-$ to a single element, meaning to the set of just that element.

(iv) This property follows directly from (iii):

$$H^-[t] = k(\mathrm{id} - (b,c))[t] = k([t] - [t]) = 0$$

□

This lemma is enough to prove the submodule theorem, which gives us the irreducibility of the S^λ:

Theorem 3.1.3 (Submodule theorem of James)**.** *If* $U \subseteq M^\lambda$ *is a submodule, then we have* $U \supseteq S^\lambda$ *or* $U \subseteq S^{\lambda\perp}$*.*

Proof. Let s and t be any two tableaux of the same shape λ with $C_t^-[s] \neq 0$. If two elements b and c are in the same row of s (so $(b,c)[s] = [s]$), then they cannot be in the same column of t, since that would imply (b,c) is an element of the column stabilizer C_t of t. Thus would follow with (iv) of the sign lemma that $C_t^-[s] = 0$.

Since we do not have to "break" any columns of t to get s, we can always find an element $\pi \in C_t$ with $[s] = \pi[t]$. This allows us to calculate $C_t^-[s]$ with the first part of the lemma above:

$$C_t^-[s] = C_t^- \pi[t] = \operatorname{sign}(\pi) C_t^-[t] = \pm e_t$$

Together with the assumption at the beginning of the proof we have $C_t^-[s] \in \{0, e_t, -e_t\}$ for any tableaux s and t.

We can now extend this to elements $u \in U$. Since we can write $u = \sum_{i \in I} c_i[s_i]$ for tabloids $[s_i]$ it follows for any tableau t:

$$C_t^- u = \sum_{i \in I} c_i C_t^-[s_i] = \left(\sum_{i \in I} d_i c_i \right) e_t \quad \text{with } d_i \in 0, 1, -1$$

Which means that $C_t^- u = c e_t$ is a multiple of e_t.

We now need to consider two cases. First let us assume that there is an $u \in U$ and a tableau t with $C_t^- u = c e_t \neq 0$. Since U is a submodule the vector $c e_t$, and with that e_t, has to be an element in U. We have seen earlier that every polytabloid generates S^λ, thus $S^\lambda \subseteq U$.

If we always have $C_t^- u = 0$, then follows for any $u \in U$ and tableau t by (ii) of the lemma:

$$\langle u, e_t \rangle = \langle u, C_t^-[t] \rangle = \langle C_t^- u, [t] \rangle = \langle 0, [t] \rangle = 0$$

Which implies $U \subseteq {S^\lambda}^\perp$ because the polytabloids span S^λ. □

Since $S^\lambda \cap {S^\lambda}^\perp = \emptyset$ over the field $\mathbb{R}$, the S^λ cannot have a non trivial submodule, so they are irreducible. For the decomposition we still need to know if two submodules are equivalent:

Proposition 3.1.4. *The Specht modules are pairwise inequivalent.*

Proof. If there is an isometry $S^\lambda \to S^\mu$, then there is also a non trivial homomorphism $\theta \colon S^\lambda \to M^\mu \supseteq S^\mu$. The polytabloids form a basis of S^λ, so there has to be at least one with $\theta(e_t) \neq 0$. Decomposing $M^\lambda = S^\lambda \oplus {S^\lambda}^\perp$ allows us to extend θ to an homomorphism $M^\lambda \to M^\mu$ by setting $\theta({S^\lambda}^\perp) = 0$. Hence we have

$$0 \neq \theta(e_t) = \theta(C_t^-[t]) = C_t^- \theta([t]) = C_t^- \left(\sum_i c_i[s_i] \right)$$

for μ-tabloids $[s_i]$, implying that there has to be at least one μ-tabloid $[s]$ with $C_t^-[s] \neq 0$.

With the same argument as in the proof of 3.1.3 we can show that if two elements b, c are in the same row of s, then they cannot be in the same column of t. In consequence we can permute within columns of t to shift every element of the first row of s into the first row of t, meaning $\lambda_1 \geq \mu_1$. Since this does not change that rows of s and columns of t have at most one common element, we can repeat it for the second row of s, shifting them as much up as possible.

Doing this we will need at most two rows of t, so $\lambda_1 + \lambda_2 \geq \mu_1 + \mu_2$. Repeating it for the remaining rows of s in the same way gives us:

$$\forall i: \quad \lambda_1 + \ldots \lambda_i \geq \mu_1 + \ldots + \mu_i$$

This property is called the *dominance lemma for partitions*. We can do the same with the isometry in the other direction $S^\mu \to S^\lambda$, resulting in

$$\forall i: \quad \lambda_1 + \ldots \lambda_i = \mu_1 + \ldots + \mu_i$$

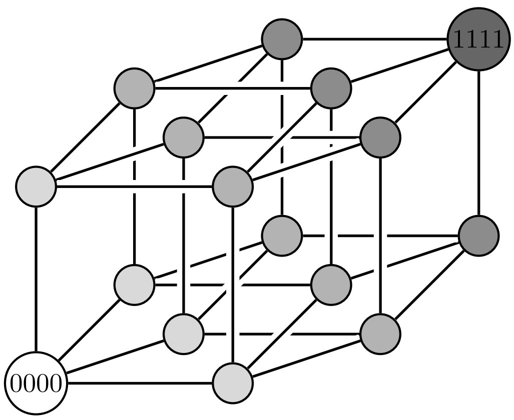
which implies $\lambda = \mu$. □

If we want to determine the isomorphism explicitly later, we are going to need the dimensions of the Specht modules. This is a lot of work for the general case, but it will be quick for the specific modules we encounter later, so we can leave it out here.

3.2 The decomposition in four dimensions

We now want to block diagonalize the Terwilliger algebra, which we do by first decomposing $\mathbb{R}^{\{0,1\}^n}$ into irreducible Γ-modules. The general method next section is more technical, since we will derive some properties needed to find the $E_{k,i,j}$ basis at the same time, as well as start by defining a special operator between different parts of the algebra. To see where this comes from, and show this a bit more visually, it makes sense to take a look at a specific smaller example first.

Figure 3.2.1: $\{0,1\}^4$ coloured by Hamming weights. Vertices are connected if the Hamming distance between them is 1.



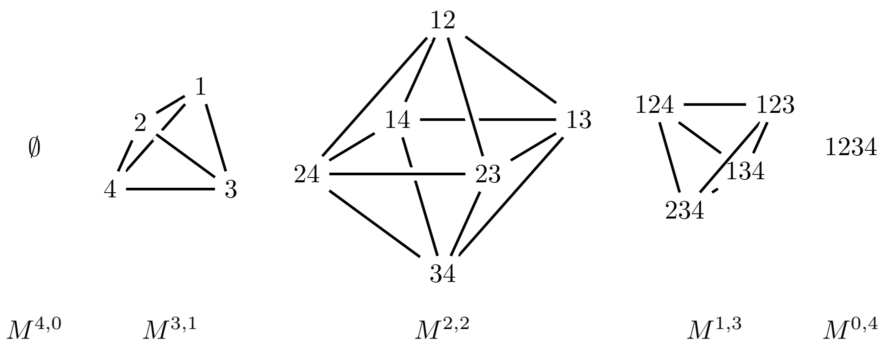
Let us take a look at the four dimensional Terwilliger algebra B. We defined this to be the set of matrices in $\mathbb{R}^{\{0,1\}^4 \times \{0,1\}^4}$ which are invariant under coordinate permutations, that is elements of $\Gamma = S_4$. The first thing we notice here, is that permuting coordinates of a binary word does not change its Hamming weight. Hence we get an easy first decomposition of $\mathbb{R}^{\{0,1\}^4}$ into 5 submodules, one for each weight from 0 to 4. We define these subsets of coordinates as $\Omega_m := \{x \in \{0,1\}^4 \mid |x| = m\}$ and the corresponding submodules $M^{4-m,m} := \mathbb{R}^{\Omega_m}$.

$$\mathbb{R}^{\{0,1\}^4} = M^{4,0} \perp M^{3,1} \perp M^{2,2} \perp M^{1,3} \perp M^{0,4}$$

Note here $M^{n-m,m}$ and $M^{m,n-m}$ are equivalent since inverting all bits of a word is an isometry between these modules, therefore it is enough to take a closer look at the first three modules here.

For the next step of the decomposition we want to interpret the binary coordinates in $\{0,1\}^4$ as their supports, that is subsets of $\{1,2,3,4\}$:

Figure 3.2.2: The first decomposition of $\mathbb{R}^{\{0,1\}^4}$ embedded in $\mathbb{R}^3$, such that S_n acts with the same elements of $O(3)$ on each module. Vertices are connected if the Hamming distance between them is 2.



Because the sets are unordered, we can interpret them as one row of a tabloid. The idea is now to uniquely complete them to full tabloids by adding a second row above with the remaining elements, which is longer since the sets we consider have at most $\lfloor \frac{n}{2} \rfloor$ elements.

This should look familiar: Our modules are indexed by all tabloids of a certain shape, and S_4 operates on them by permuting their elements. They are exactly the (reducible) modules M^λ for the shape $\lambda = (n-m, m)$, which we worked with last section. We already know of the irreducible submodule $S^\lambda \subseteq M^\lambda$, but nearly nothing about the complement $S^{\lambda^\perp}$ of them.

Let us take a look at $S^{(2,2)}$ specifically: It is the subspace spanned by the polytabloids of tableaux of shape $(2,2)$. We will see later it is spanned by the polytabloids of *standard* tableaux, which are the tableaux with increasing elements in both rows and columns. Here these are:

$$t_1 = \begin{array}{|c|c|}\hline 1 & 2 \\ \hline 3 & 4 \\ \hline\end{array} \quad e_{t_1} = \begin{array}{cc}\hline 1 & 2 \\ \hline 3 & 4 \\ \hline\end{array} - \begin{array}{cc}\hline 2 & 3 \\ \hline 1 & 4 \\ \hline\end{array} - \begin{array}{cc}\hline 1 & 4 \\ \hline 2 & 3 \\ \hline\end{array} + \begin{array}{cc}\hline 3 & 4 \\ \hline 1 & 2 \\ \hline\end{array}$$

$$t_2 = \begin{array}{|c|c|}\hline 1 & 3 \\ \hline 2 & 4 \\ \hline\end{array} \quad e_{t_2} = \begin{array}{cc}\hline 1 & 3 \\ \hline 2 & 4 \\ \hline\end{array} - \begin{array}{cc}\hline 2 & 3 \\ \hline 1 & 4 \\ \hline\end{array} - \begin{array}{cc}\hline 1 & 4 \\ \hline 2 & 3 \\ \hline\end{array} + \begin{array}{cc}\hline 2 & 4 \\ \hline 1 & 3 \\ \hline\end{array}$$

Since we only need to consider tableaux with two rows, we can notice a pattern here: If the bottom rows of two tabloids differ by only one element, then their signs are opposite. This is caused by the column size being two of the columns with a bottom row, so $\{\mathrm{id}, (i,j)\}$ can be factored out of the column stabilizer, where $\{i,j\}$ is any column. This implies for any vector in S^λ, that the sum of the entries indexed by all tabloids with $m-1$ common entries in the bottom row is zero. For example if we have $v \in S^{(2,2)}$, then $(dv)_1 := v_{12} + v_{13} + v_{14} = 0$. Taking a look at figure 3.2.2 above, we notice it is exactly a face of the octahedron. If we now apply a permutation to this face, we get another face of the octahedron, and the new face has again $m-1$ common elements. In our example we get four faces of this type, giving us a second description of $S^{(2,2)}$, since $\dim(S^{(2,2)}) = 2 = 6 - 4$:

$$v \in S^{(2,2)} \Leftrightarrow dv = \begin{pmatrix} v_{12} + v_{13} + v_{14} \\ v_{12} + v_{23} + v_{24} \\ v_{13} + v_{23} + v_{34} \\ v_{14} + v_{24} + v_{34} \end{pmatrix} = 0$$

Meaning $S^{(2,2)}$ is the kernel of d, which we will prove more formally later. But what does d do to elements in ${S^{(2,2)}}^\perp$? It is easy to check that d preserves the action of S_4, if we see it as a function from $M^{2,2}$ to $M^{3,1}$. Since the image of ${S^{(2,2)}}^\perp$ has dimension 4, it even is an isometry between ${S^{(2,2)}}^\perp$ and $M^{3,1}$:

$$M^{2,2} \cong S^{(2,2)} \perp {S^{(2,2)}}^\perp \cong S^{(2,2)} \perp M^{3,1}$$

This can be repeated recursively, fully decomposing $\mathbb{R}^{\{0,1\}^4}$:

$$\begin{aligned} \mathbb{R}^{\{0,1\}^4} &= M^{4,0} \perp M^{3,1} \perp M^{2,2} \perp M^{1,3} \perp M^{0,4} \\ &= S^{(4,0)} \\ &\perp S^{(4,0)} \perp S^{(3,1)} \\ &\perp S^{(4,0)} \perp S^{(3,1)} \perp S^{(2,2)} \\ &\perp S^{(4,0)} \perp S^{(3,1)} \\ &\perp S^{(4,0)} \end{aligned}$$

As shown earlier, the Specht modules are all different, which means this already gives us the amount and sizes of blocks of the block diagonalization of the Terwilliger algebra (for $n = 4$) already:

$$\mathcal{B} \cong \mathbb{R}^{5\times 5} \oplus \mathbb{R}^{3\times 3} \oplus \mathbb{R}$$

Which reduces the amount of entries from $16^2 = 256$ to $5^2+3^2+1 = 35$. Since the first decomposition ordered the elements by weight, we get the decompositions of the subspaces $\mathbb{R}^{\{0,1\}^4_{\geq D}}$ by removing the rows and columns belonging to the lower weights.

For the next step we would now have to find orthonormal bases and isometries of these submodules explicitly, which will be done next chapter for the general case.

3.3 Distance regular graphs

In this section a more general approach to find the decomposition is explained, because in most cases one cannot "guess" the decomposition as easily as in the example above. Both this general section and the application of it to our problem next section are done similarly to [4], chapters 5 and 6.

In this section and the following ones we will interpret elements $x \in \{0,1\}^n$ as subsets $A \subseteq \{1,\dots,n\}$, as we have done last section to find the tabloids. We defined Ω_m as $\{x \in \{0,1\}^n \mid |x| = m\}$, which means from now on it will instead be seen as

$$\Omega_m \coloneqq \{A \subseteq \{1,\dots,n\} \mid |A| = m\}$$

We now want to determine the general decomposition of $M^{n-m,m}$ into irreducible submodules and calculate the $E_{k,i,j}$ basis in the process. But first we need a few more general results. The decision to draw graphs in figure (3.2.2) was not random, since S_4 acts on all of them as graph automorphisms. We will see that all of these graphs are *distance regular*, but let us define the general graph first:

$$G^{n-m,m} \coloneqq \big(\Omega_m, \delta^{-1}(1)\big)$$

where δ is the *Johnson distance*

$$\delta\colon \Omega_m \times \Omega_m \to \mathbb{N}, (A,B) \mapsto m - |A \cap B| = |A \setminus B| = |B \setminus A|$$

Which coincides with $\frac{1}{2}d$, half of the Hamming distance, if we go back to the binary interpretation (But only within the same Ω_m). So this distance gives us exactly the length of shortest paths between nodes of the graphs.

Now let us define what a distance regular graph is:

Definition 3.3.1. *A graph $G = (V,E)$ is called distance regular, if there exist constants $b_0,\dots,b_N$ and $c_0,\dots,c_N$, where*

$$N = \operatorname{diam}(G) \coloneqq \max_{x,y\in V} \delta(x,y)$$

such that if x and y have distance i, then exactly b_i neighbours of x have distance $i+1$ from y, and exactly c_i neighbours of x have distance $i-1$ from y.

Proposition 3.3.2. *$G^{n-m,m}$ is distance regular with diameter* $\min\{n-m,m\}$ *and*

$$b_i = (n-m-i)(m-i)$$
$$c_i = i^2$$

Proof. Two disjunct sets in Ω_m can be found if and only if $n \geq 2m$, in that case we have $\operatorname{diam}(G^{n-m,m}) = m$. Otherwise they have at least $m-(n-m) = 2m-n$ elements in common, so $m - \min_{A,B\in\Omega_m} |A \cap B| = m - (2m-n) = n-m$. Since in this case $n-m \leq m$ we have

$$\operatorname{diam}(G^{n-m,m}) = \min\{m, n-m\}$$

Suppose now that $\delta(A, B) = i$. The neighbours of B are exactly the elements given by $(B \setminus \{b\}) \cup \{a\}$, where $b \in B$ and $a \neq B$. If we want to increase the distance to A we need to choose $b \in A$ and $a \neq A$ at the same time. Since $|A \cap B| = m - i$ and $|\overline{A \cup B}| = n - m - i$ we have $(n - m - i)(m - i)$ ways to choose a and b.

If we want to lower the distance to A instead, we select $b \neq A$ and $a \in A$. As $|B \setminus A| = \delta(A, B) = i$ and $|A \setminus B| = \delta(A, B) = i$ we get $c_i = i^2$. □

What can we use this property for? For any graph $G = (V, E)$ we can define a family of linear operators on $\mathbb{R}^V$ for $j = 0, 1, \ldots$:

$$\Delta_j \colon \mathbb{R}^V \to \mathbb{R}^V, \ \ (\Delta_j v)(x) \coloneqq \sum_{\substack{y \in V \\ \delta(x,y)=j}} v(y)$$

With $\Delta_j = 0$ for $J > N$. If we now have an eigenvector w of Δ_j for the eigenvalue λ and a graph automorphism π of G, then

$$\begin{aligned}(\Delta_N \pi(w))(x) &= \sum_{\substack{y \in V \\ \delta(x,y)=N}} w(\pi^{-1}(y)) = \sum_{\substack{y \in V \\ \delta(\pi^{-1}(x),y)=N}} w(y) \\ &= (\pi(\Delta_N w))(x) = \lambda(\pi(w))(x)\end{aligned}$$

So $\pi(w)$ is in the same eigenspace of Δ_N as w. This means that the eigenspaces of all the Δ_N are closed under action of $\mathrm{Aut}(G)$, implying they are $\mathrm{Aut}(G)$-modules! If the graph additionally is distance regular, then we can even show it is enough to only consider the eigenspaces of the *adjacency operator* $\Delta \coloneqq \Delta_1$:

Proposition 3.3.3. *For $j = 0, \ldots, N$:*

(i) $\Delta_j \Delta_1 = b_{j-1}\Delta_{j-i} + (b_0 - b_j - c_j)\Delta_j + c_{j+1}\Delta_{j+1}$

(ii) There exists a real polynomial p_j of degree j such that $\Delta_j = p_j(\Delta_1)$.

(iii) $p(\Delta_1)$ is a linear combination of $\Delta_0, \ldots, \Delta_N$ for all polynomials p, and the $\Delta_0, \ldots, \Delta_N$ are linearly independent.

Proof. Let $v \in \mathbb{R}^V$ and $x \in V$:

$$\begin{aligned}(\Delta_j \Delta_1 v)(x) &= \sum_{\substack{z \in V \\ \delta(z,x)=j}} \sum_{\substack{y \in V \\ \delta(y,z)=1}} v(y) \\ &= \sum_{\substack{z \in V \\ \delta(z,x)=j}} \left(\sum_{\substack{y \in V \\ \delta(y,z)=1 \\ \delta(y,x)=j-1}} v(y) + \sum_{\substack{y \in V \\ \delta(y,z)=1 \\ \delta(y,x)=j}} v(y) + \sum_{\substack{y \in V \\ \delta(y,z)=1 \\ \delta(y,x)=j+1}} v(y) \right)\end{aligned}$$

By switching the two sums we can calculate the first sum:

$$\sum_{\substack{z\in V\\ \delta(z,x)=j}} \sum_{\substack{y\in V\\ \delta(y,z)=1\\ \delta(y,x)=j-1}} v(y) = \sum_{\substack{y\in V\\ \delta(y,x)=j-1}} \sum_{\substack{z\in V\\ \delta(z,x)=j\\ \delta(z,y)=1}} v(y) = b_{j-1}(\Delta_{j-1}v)(y)$$

And analogously $c_{j+1}(\Delta_{j+1}v)(y)$ for the third sum. The second is a bit different:

$$\begin{aligned}\sum_{\substack{y\in V\\ \delta(y,x)=j}} \sum_{\substack{z\in V\\ \delta(z,x)=j\\ \delta(z,y)=1}} v(y) &= \sum_{\substack{y\in V\\ \delta(y,x)=j}} \left(\sum_{\substack{z\in V\\ \delta(z,y)=1}} 1 - \sum_{\substack{z\in V\\ \delta(z,x)=j+1\\ \delta(z,y)=1}} 1 - \sum_{\substack{z\in V\\ \delta(z,x)=j-1\\ \delta(z,y)=1}} 1 \right) v(y)\\ &= (b_0 - b_j - c_j)(\Delta_j v)(x)\end{aligned}$$

To prove (ii) we first use (i) to get the formula

$$\Delta_1^2 = b_0\Delta_0 + (b_0 - b_1 - c_1)\Delta_1 + c_2\Delta_2$$

Note here that $c_2, \dots, c_N > 0$ by definition of N. Because of $\Delta_0 = \mathrm{id}$ this formula gives us p_2:

$$\Delta_2 = \frac{1}{c_2}\Delta_1^2 - \frac{b_0 - b_1 - c_1}{c_2}\Delta_1 - \frac{b_0}{c_2}\mathrm{id}$$

The general case follows inductively by using (i) in each step.

The first part of (iii) is shown by repeatedly applying (i) to reduce the degree of p by adding terms in other Δ_j. We have seen it for Δ_1^2, and for higher degrees we simply split $\Delta_1^k = \Delta_1\Delta_1^{k-1}$, apply the case $k-1$ and use (i) on all resulting terms.

If δ_y is the characteristic vector of $y \in V$, then

$$\left(\sum_{j=0}^{N} \alpha_j \Delta_j \delta_y \right)(x) = \alpha_{\delta(x,y)}$$

So all α_j have to be zero for the linear combination to be zero as well, and the Δ_j are linearly independent. □

This proposition implies that the eigenspaces of all the Δ_j are eigenspaces of $\Delta = \Delta_1$ too. Furthermore Δ is selfadjoint:

$$\langle \Delta v, w\rangle = \sum_{x\in V} \sum_{\substack{y\in V\\ \delta(x,y)=1}} v(y)w(x) = \sum_{y\in V} \sum_{\substack{x\in V\\ \delta(x,y)=1}} v(y)w(x) = \langle v, \Delta w\rangle$$

So the different eigenspaces are orthogonal to each other:

$$\lambda_v\langle v, w\rangle = \langle \Delta v, w\rangle = \langle v, \Delta w\rangle = \lambda_w\langle v, w\rangle$$

We can even say how many eigenspaces Δ has exactly, leading to:

Theorem 3.3.4. *There is an orthogonal decomposition*

$$\mathbb{R}^V = \bigoplus_{j=0}^{N} V_i$$

into distinct eigenspaces V_i of Δ, which are closed under actions of $\mathrm{Aut}(G)$*.*

Proof. We only have to show that the amount of distinct eigenspaces is $N + 1$.

The *Bose-Mesner algebra* $\mathcal{A}_G$ of the graph is the span of $\Delta_0, \ldots, \Delta_N$, which by last proposition is exactly

$$\mathcal{A}_G = \{p(\Delta) \mid p \text{ polynomial}\}$$

Since the Δ_j are independent they form a basis of $\mathcal{A}_G$, and the dimension of the algebra is $N + 1$.

Let $\lambda_0 > \ldots > \lambda_M$ now be the distinct eigenvalues of Δ corresponding to the V_j. Because of $\Delta_j = p_j(\Delta)$ we know that that the V_i are eigenspaces of Δ_j for the eigenvalue $p_j(\lambda_i)$, giving us a decomposition

$$\Delta_j = \sum_{i=0}^{M} p_j(\lambda_i) E_i$$

where E_i is the orthogonal projection onto V_i. These projections are orthogonal (as the V_i are orthogonal), and are elements of $\mathcal{A}_G$:

$$E_i = \prod_{j \neq i} \frac{\Delta - \lambda_j \mathrm{id}}{\lambda_i - \lambda_j} = p'_i(\Delta)$$

Because E_i is linear and for $v \in V_k$:

$$p'_i(\Delta)(v) = \left(\prod_{j \neq i} \frac{\lambda_k - \lambda_j}{\lambda_i - \lambda_j} \right) v = \begin{cases} v \text{ if } k = i \\ 0 \text{ else} \end{cases}$$

So the E_i form a second basis of $\mathcal{A}_G$, of which the dimension is $N + 1$, proving $M = N$. □

3.4 The decomposition of $M^{n-m,m}$

We now have everything we need for the decomposition of $M^{n-m,m}$ (and with that of the whole module), which we start by defining the operator d, as seen in the example, formally as

$$\begin{aligned} d\colon M^{n-m,m} &\to M^{n-m+1,m-1}, \\ (dv)_A &= \sum_{A \subset B \in \Omega_m} v_B \qquad \text{for } v \in M^{n-m,m}, A \in \Omega_{m-1} \end{aligned}$$

Since this operator is linear, we can define its adjoint operator

$$\begin{aligned} d^*\colon M^{n-m+1,m-1} &\to M^{n-m,m}, \\ (d^*w)_B = \sum_{B \supset A \in \Omega_{m-1}} w_A & \qquad \text{for } w \in M^{n-m+1,m-1}, B \in \Omega_m \end{aligned}$$

which we can check by

$$\begin{aligned} \langle dv, w \rangle &= \sum_{A \in \Omega_{m-1}} (dv)_A w_A = \sum_{\Omega_{m-1} \ni A \subset B \in \Omega_m} v_B w_A \\ &= \sum_{B \in \Omega_m} v_B (d^*w)_B = \langle v, d^*w \rangle \end{aligned}$$

To prove properties of these operators it is helpful to take a look at Dirac functions $\delta_B \in M^{n-m,m}$, that is a unit vector with a one at coordinate B. These form a basis of $M^{n-m,m}$, and it is easy to see how d and d^* act on them:

$$(d\delta_B)_A = \sum_{A \subset C \in \Omega_m} (\delta_B)_C = \begin{cases} 1 & \text{if } A \subset B \\ 0 & \text{otherwise} \end{cases}$$

Which means that

$$d\delta_B = \sum_{j \in B} \delta_{B \setminus \{j\}}$$

Similarly we have

$$d^*\delta_A = \sum_{i \notin A} \delta_{A \cup \{i\}}$$

We now technically have n different d and d^* operators, for each $m = 0, \dots, n-1$. To make things easier from now on we combine them to linear operators on the whole space from before the first decomposition:

$$d\colon \bigoplus_{m=0}^{n} M^{n-m,m} \to \bigoplus_{m=0}^{n} M^{n-m,m}$$

defined for $v = \sum_{m=1}^{n} v^m$, $v^m \in M^{n-m,m}$ by

$$dv = \sum_{m=1}^{n} dv^m$$

Notice here that $dv^0 = 0$. In the same way we can generalize d^* to

$$d^*\colon \bigoplus_{m=0}^{n} M^{n-m,m} \to \bigoplus_{m=0}^{n} M^{n-m,m}, \quad d^*v = \sum_{m=0}^{n-1} d^*v^m$$

This time we have $d^*v^n = 0$.

Last section we have seen the importance of the adjacency operator Δ. For our graph applying it to an $v \in M^{n-m,m}$ gives us

$$(\Delta v)_A = \sum_{B\in\Omega_m,\delta(A,B)=1} v_B$$

Hence its image of a Dirac function is

$$\Delta\delta_B = \sum_{A\in\Omega_m,\delta(A,B)=1} \delta_A$$

The Operators d, d^* and Δ are closely connected, as can be seen in the following lemma:

Lemma 3.4.1. *For $v \in M^{n-m,m}$ we have:*

(i) $dd^*v = \Delta v + (n-m)v$

(ii) $d^*dv = \Delta v + mv$

Proof. Since the Dirac functions form a basis of $M^{n-m,m}$ and all operators are linear, it is enough to check the lemma for these. Let A be an element of Ω_m:

$$\begin{aligned} dd^*\delta_A &= d\sum_{j\notin A}\delta_{A\cup\{j\}} = \sum_{j\notin A, i\in A\cup\{j\}} \delta_{(A\cup\{j\})\setminus\{i\}} \\ &= \sum_{j\notin A}\delta_A + \sum_{j\notin A, i\in A} \delta_{(A\cup\{j\})\setminus\{i\}} = (n-m)\delta_A + \Delta v \end{aligned}$$

Since $\delta(A,B) = 1 \Leftrightarrow B = (A\cup\{j\})\setminus\{i\}$ for $j \notin A, i \in A$. Similarly we have

$$d^*d\delta_A = d^*\sum_{j\in A}\delta_{A\setminus\{j\}} = \sum_{j\in A}\delta_A + \sum_{j\in A, i\notin A}\delta_{(A\setminus\{j\})\cup\{i\}} = m\delta_A + \Delta v$$

□

Starting with the next lemma the *Pochhammer symbol* will appear in most formulas. It is defined for $a \in \mathbb{R}$, $i \in \mathbb{N}$ as

$$(a)_i \coloneqq \prod_{l=0}^{i-1}(a+l) = a(a+1)\dots(a+i-1)$$

It is straightforward to check a few helpful properties:

- $a(a+1)_{i-1} = (a)_i$
- $(1)_n = n!$
- $\binom{n}{k} = \frac{(n-k+1)_k}{k!} = \frac{(-1)^k(-n)_k}{k!}$

We can now take a look at what multiple applications of d and $d*$ do to a vector:

Lemma 3.4.2. *For $v \in M^{n-m,m}$ and $1 \leq p \leq q \leq n-m$ we have:*

(i) $d(d^*)^q v = (d^*)^q dv + q(n-2m-q+1)(d^*)^{q-1} v$

(ii) $dv = 0 \quad \Rightarrow \quad d^p(d^*)^q v = (q-p+1)_p (n-2m-q+1)_p (d^*)^{q-p} v$

Proof. From last lemma it follows that

$$dd^*v - d^*dv = (n-2m)v$$

which is the case $q = 1$ of the first formula. The general case follows inductively:

$$d(d^*)^q v = dd^* \left((d^*)^{q-1} v\right)$$

Here we apply the case $q = 1$ with $(d^*)^{q-1} v \in M^{n-m-(q-1),m-(q-1)}$:

$$= d^* d(d^*)^{q-1} v + (n - 2(m-q+1))(d^*)^{q-1} v$$

And the case $q - 1$:

$$\begin{aligned} &= d^* \left((d^*)^{q-1} dv + (q-1)(n-2m-(q-1)+1)(d^*)^{q-2} v\right) \\ &\quad + (n-2m+2q-2)(d^*)^{q-1} v \\ &= (d^*)^q dv + q(n-2m-q+1)(d^*)^{q-1} v \end{aligned}$$

The second formula is shown similarly. If $p = 1$ then it matches the first formula since $dv = 0$, and the general case is again proven inductively by applying first the case $p = 1$ and then $p - 1$:

$$\begin{aligned} d^p(d^*)^q v &= d^{p-1} q(n-2m-q+1)(d^*)^{q-1} v \\ &= q(n-2m-q+1)(q-p+1)_{p-1}(n-2m-(q-1)+1)_{p-1}(d^*)^{q-p} v \\ &= (q-p+1)_p (n-2m-q+1)_p (d^*)^{q-p} v \end{aligned}$$

□

We now a get a second description of (some) Specht modules:

Definition 3.4.3.

$$\hat{S}^{n-m,m} := M^{n-m,m} \cap \operatorname{Ker}(d)$$

For now we will not assume that they are the same modules as defined before, since we do not need any of their properties here. We will prove that they are in fact the same after next theorem.

We can now decompose $M^{n-m,m}$ into the Δ-eigenspaces from theorem 3.3.4:

Theorem 3.4.4. *We have:*

(i) For $0 < m \leq \frac{n}{2}$*:*

$$\dim\ \hat{S}^{n-m,m} = \binom{n}{m} - \binom{n}{m-1}$$

(ii) *For* $0 \leq m \leq n$, $0 \leq k \leq \min\{n-m, m\}$, $v \in \hat{S}^{n-k,k}$*:*

$$\|(d^*)^{m-k} v\|^2 = (m-k)!(n-k-m+1)_{m-k}\|v\|^2$$

which implies that $(d^*)^{m-k}$ *is injective from* $\hat{S}^{n-k,k}$ *to* $M^{n-m,m}$*.*

(iii) *For* $0 \leq m \leq n$*:*

$$M^{n-m,m} = \bigoplus_{k=0}^{\min\{m,n-m\}} (d^*)^{m-k}\hat{S}^{n-k,k}$$

is the decomposition of $M^{n-m,m}$ *into distinct* Δ*-eigenspaces.*

(iv) *The eigenvalue corresponding to* $(d^*)^{m-k}\hat{S}^{n-k,k}$ *is*

$$m(n-m) - k(n-k+1)$$

Proof. The length change in (ii) is a direct consequence of lemma 3.4.2 (ii), in fact we even get more general case for $v \in \hat{S}^{n-h,h}$, $w \in M^{n-k,k}$ with $0 < h \leq k \leq m \leq n$:

$$\begin{aligned} \langle (d^*)^{m-h} v, (d^*)^{m-k} w\rangle &= \langle d^{m-k}(d^*)^{m-h} v, w\rangle \\ &= (k-h+1)_{m-k}(n-h-m+1)_{m-k}\langle (d^*)^{k-h} v, w\rangle \end{aligned}$$

If we set $h = k$ and $v = w$ then we get

$$\begin{aligned} \|(d^*)^{m-k} v\|^2 &= (1)_{m-k}(n-k-m+1)_{m-k}\|v\|^2 \\ &= (m-k)!(n-k-m+1)_{m-k}\|v\|^2 \end{aligned}$$

To prove (iv) we take an element in $\hat{S}^{n-k,k} = M^{n-k,k} \cap \mathrm{Ker}(d)$ and apply first lemma 3.4.1 (i) and then lemma 3.4.2 (i):

$$\begin{aligned} \Delta(d^*)^{m-k} v &= dd^*(d^*)^{m-k} v - (n-m)(d^*)^{m-k} v \\ &= (m-k+1)(n-k-m)(d^*)^{m-k} v - (n-m)(d^*)^{m-k} v \\ &= (m(n-m) - k(n-k+1))\,(d^*)^{m-k} v \end{aligned}$$

It is enough to show (iii) for $0 \leq m \leq \frac{n}{2}$, since the complement (respectively bit wise inversion) is an isometry from $M^{n-m,m}$ to $M^{m,n-m}$. Since we have already shown that they are (distinct) eigenspaces in (iv), and their number is $\operatorname{diam}(G^{n-m,m}) + 1 = \min\{n-m, m\} + 1$, we only need to show that this decomposition includes all of $M^{n-m,m}$ because of theorem 3.3.4. We do not know the dimension of the Specht modules yet, so we do this inductively like in the example earlier. First of all, we can decompose

$$M^{n-1,1} = \mathbb{R}^{\{\{0\},\ldots,\{n\}\}} = S^{n-1,1} \oplus d^* S^{n,0}$$

Since the elements $v \in \hat{S}^{n-1,1} = \mathrm{Ker}(d) \cap \mathbb{R}^{\{\{0\},\ldots,\{n\}\}}$ are the elements with mean zero, because $d(v) = \sum_i v_i = 0$, and $d^*\hat{S}^{n,0} = d^*\mathbb{R}^{\{\emptyset\}}$ are the constant elements. Furthermore we have

$$M^{n-m,m} = \hat{S}^{n-m,m} \oplus d^* M^{n-m+1,m-1}$$

in the general case. To show this let $v \in \hat{S}^{n-m,m}$ and $w \in M^{n-m+1,m-1}$, then

$$0 = \langle 0, w\rangle = \langle dv, w\rangle = \langle v, d^* w\rangle$$

so $\hat{S}^{n-m,m} \subseteq \left(d^* M^{n-m+1,m-1}\right)^{\perp}$. The other direction follows because

$$0 = \langle v, d^* w\rangle = \langle dv, w\rangle = 0$$

for all w implies $dv = 0$.

Note that d^* preserves orthogonality after the more general formula of (*ii*) above:

$$\langle (d^*)^{m-h} v, (d^*)^{m-k} w\rangle = c\langle (d^*)^{k-h} v, w\rangle \quad \text{with } c \neq 0$$

for $v \in \hat{S}^{n-h,h}$, $w \in M^{n-k,k}$ and $h \leq k$, so can now use it inductively to get the orthogonal decomposition (*iii*):

$$M^{n-m,m} = \bigoplus_{k=0}^{\min\{m,n-m\}} (d^*)^{m-k} \hat{S}^{n-k,k}$$

This decomposition gives us the dimension formula (*i*) since

$$\begin{aligned} \dim\ \hat{S}^{n-m,m} &= \dim\ M^{n-m,m} - \dim\ d^* M^{n-m+1,m-1} \\ &= \dim\ M^{n-m,m} - \dim\ M^{n-m+1,m-1} \\ &= \binom{n}{m} - \binom{n}{m-1} \end{aligned}$$

where we used that d^* is injective for $0 \leq m \leq \frac{n}{2}$, which follows from (*ii*). □

We can now prove the equivalence of the two Specht module definitions:

Lemma 3.4.5.

$$\hat{S}^{n-m,m} = S^{n-m,m}$$

Proof. Let t be a $(n-m, m)$-tableau, with $m \leq \frac{n}{2}$ since $S^{n-m,m} \cong S^{m,n-m}$, and $k = n - m$:

$$t = \begin{array}{|c|c|c|c|c|c|} \hline a_1 & a_2 & \dots & a_m & \dots & a_k \\ \hline b_1 & b_2 & \dots & b_m \\ \cline{1-4} \end{array}$$

The column stabilizer of this tableau is

$$C_t = S_{\{a_1,b_1\}} \times S_{\{a_2,b_2\}} \times \ldots \times S_{\{a_m,b_m\}}$$

so C_t^- factorizes as

$$C_t^- = \prod_{i=1}^{m} (\mathrm{id} - (a_i, b_i))$$

Let B now be an element of Ω_{m-1}. We showed that

$$(dv)_B = \sum_{j \notin B} v_{B \cup \{j\}}$$

Since the tableau contains every number from 1 to n once, applying d to a polytabloid $e_t = C_t^-[t]$ at coordinate B results either a sum of zeroes if B is never contained in the bottom row of the tabloids appearing in e_t, or there are exactly two permutations with B in the bottom row:

$$(de_t)_B = \sum_{\sigma \in \pi S_{\{a_i,b_i\}}} \mathrm{sign}(\sigma) = \mathrm{sign}(\pi) + \mathrm{sign}(\pi(a_i,b_i)) = 0$$

where π is the unique permutation in $\prod_{j=1,j\neq i}^m S_{\{a_j,b_j\}}$ such that B is contained in the bottom row of $\pi[t]$. This means that all polytabloids, which span $S^{n-m,m}$, are elements of $\hat{S}^{n-m,m}$.

We now want to show that dim $S^{n-m,m} \geq$ dim $\hat{S}^{n-m,m} = \binom{n}{m} - \binom{n}{m-1}$, from which follows the lemma. For this we define *standard tableaux* as those tableaux of which the entries increase from left to right and from top to bottom in all rows and columns. Applying a permutation of the column stabilizer to a standard tableau t is never another standard tableau, since at least one column is not ordered right afterwards. In tabloids the order of the row elements does not matter, but even then it is not possible to get the tabloid of a different standard tableau t' by only permuting within columns, because then at least two columns would have to be in the wrong order in t, or $t = t'$.

So the value at coordinate $[t]$ of a polytabloid of a standard tableau t is not zero only in the case that the polytabloid is e_t, and by this the polytabloids of standard tableaux are linearly independent. We can show that there are $\binom{n}{m} - \binom{n}{m-1}$ standard tablaux inductively over m and n:

If $m = 0$ then the tableau has only one row, which is sorted, so there is only $1 = 1 - 0 = \binom{n}{0} - \binom{n}{-1}$ standard tableaux of shape $(n, 0)$. Assume now that the formula is correct for the shapes $((n-1)-m, m)$ and $((n-1)-(m-1), m-1)$. Since the new highest entry n can only be at the end of either the first or the second row, removing it gives us a standard tableau of one of the above mentioned shapes. Hence the amount of standard tableaux of shape (n, m) is the sum of the amount of standard tableaux of those two shapes:

$$\left(\binom{n-1}{m} - \binom{n-1}{m-1}\right) + \left(\binom{n-1}{m-1} - \binom{n-1}{m-2}\right) = \binom{n}{m} - \binom{n}{m-1}$$

So $S^{n-m,m} = \hat{S}^{n-m,m}$, and we even found a basis of these modules. □

Theorem 3.4.6. *The decomposition of $M^{n-m,m}$ into irreducible S_n-modules is:*

$$M^{n-m,m} \cong \bigoplus_{k=0}^{\min\{m,n-m\}} S^{n-k,k}$$

Proof. Again it is only needed to prove the theorem for $m \leq \frac{n}{2}$ because of symmetry. We just need to show that there is an isometry between $S^{n-k,k}$ and its image, since we have already proven a similar decomposition in theorem 3.4.4. In part (ii) of the same theorem we have shown that the operator $(d^*)^{m-k}$ is injective, and by that a (linear) isomorphism to its image.

We still have to show that $(d^*)^{m-k}$ preserves the action of S_n and the inner products. For this let π be an element of S_n and $B \in \Omega_m$:

$$d^*\pi(\delta_B) = d^*\delta_{\pi(B)} = \sum_{j \notin \pi(B)} \delta_{\pi(B)\cup\{j\}} = \sum_{j \notin B} \delta_{\pi(B\cup\{j\})} = \pi(d^*\delta_B)$$

Since the δ_B form a basis of $M^{n-m,m}$ the whole operator preserves the permutations, and using this repeatedly gives us the same for $(d^*)^{m-k}$. The inner products on $S^{n-k,k}$ are (nearly) compatible with $(d^*)^{m-k}$ because for $v, w \in S^{n-k,k} \subseteq M^{n-k,m}$ we either have $m = k$, or

$$\begin{aligned}\langle (d^*)^{m-k}v, (d^*)^{m-k}w\rangle &= \langle d^{m-k}(d^*)^{m-k}v, w\rangle \\ &= \langle (n-2k-(m-k)+1)_{m-k}v, w\rangle\end{aligned}$$

by lemma 3.4.2. That means by scaling we get an isometry between $S^{n-k,k}$ and $(d^*)^{m-k}S^{n-k,k}$:

$$((n-k-m+1)_{m-k})^{-\frac{1}{2}}\,(d^*)^{m-k} = \sqrt{\frac{(n-k-m)!}{(n-2k)!}}(d^*)^{m-k}$$

The factor is positive, since $k < m \leq \frac{n}{2}$. □

This does not give us the block decomposition of our program yet, but we do know how large the programs will be afterwards: The Terwilliger algebra block diagonalizes to $\lfloor \frac{n}{2} \rfloor + 1$ blocks of sizes

$$n+1, n-1, n-3, \ldots, \begin{cases} 1 \text{ if } n \text{ is even} \\ 2 \text{ if } n \text{ is odd}\end{cases}$$

Even with the empty space outside of blocks the program now only uses matrices of dimension $\mathcal{O}(n^2)$, where we had matrices of dimension $\mathcal{O}(2^n)$ before! For cases with $D > 0$ the matrix sizes reduce further, since we remove rows and columns corresponding to the $M^{n-m,m}$ with $m < D$. We can now fix $H_{k,i} \cong S^{n-k,k}$ and $h_k = \dim(H_{k,i}) = \binom{n}{k} - \binom{n}{m-1}$ for the rest of the thesis.

Chapter 4

Determining the isomorphism

In this chapter we are going to determine the isomorphism for the block diagonalization explicitly. We again follow Vallentin's paper [14], while the calculation of the needed invariant functions is a mix of [4], chapter 6, and the two papers [5], [6] of Dunkl. As seen in section 2.2, the calculation of the isomorphism comes down to calculating the matrices $E_{k,i,j}$ explicitly:

4.1 An approach to find $E_{k,i,j}$

We now want to find an approach to determine the basis $E_{k,i,j}$ of the Terwilliger algebra $\mathcal{B}$, which we defined using orthonormal bases of all modules of the decomposition. As can be seen in the example decomposition in dimension 4, the basis of polytabloids of standard tableaux is not orthogonal.

Instead we use a different approach, with which we can avoid determining the orthonormal bases explicitly: We construct special functions for each basis element by:

$$z_{k,i,j} \colon S_n \to \mathbb{R}, \quad \pi \mapsto E_{k,i,j}(\pi(1^i 0^{n-i}), 1^j 0^{n-j})$$

Since the algebra $\mathcal{B}$ is S_n invariant, this function is left-invariant for elements σ in $H = S_j \times S_{n-j}$, where S_j acts on the first j and S_{n-j} on the remaining $n-j$ coordinates:

$$\begin{aligned} z_{k,i,j}(\sigma\pi) &= E_{k,i,j}\left(\sigma(\pi(1^i 0^{n-i})), 1^j 0^{n-j}\right) \\ &= E_{k,i,j}\left(\sigma^{-1}(\sigma(\pi(1^i 0^{n-i}))), \sigma^{-1}(1^j 0^{n-j})\right) \\ &= E_{k,i,j}\left(\pi(1^i 0^{n-i}), 1^j 0^{n-j}\right) \\ &= z_{k,i,j}(\pi) \end{aligned}$$

And it is $S_i \times S_{n-i} = K$-right-invariant:

$$\begin{aligned} z_{k,i,j}(\pi\sigma) &= E_{k,i,j}\left(\pi(\sigma(1^i 0^{n-i})), 1^j 0^{n-j}\right) \\ &= E_{k,i,j}\left(\pi(1^i 0^{n-i}), 1^j 0^{n-j}\right) \\ &= z_{k,i,j}(\pi) \end{aligned}$$

Furthermore it lies in the subspace spanned by the matrix entries of an irreducible representation corresponding to $S^{n-k,k} \cong H_{k,i}$. To show this let $e_{k,i,1}, \ldots, e_{k,i,h_k}$ be an orthonormal basis of this module, which allows us to construct an orthogonal, irreducible representation $T_{k,i} \colon S_n \to O(\mathbb{R}^{h_k})$ of it by:

$$\pi(e_{k,i,l}) = \sum_{l'=1}^{h_k} [T_{k,i}(\pi)]_{l',l}\, e_{k,i,l'}$$

Since we have $T_{k,i}(\sigma\pi) = T_{k,i}(\sigma)T_{k,i}(\pi)$:

$$\begin{aligned} \sigma(\pi(e_{k,i,l})) &= \sum_{l'=1}^{h_k} [T_{k,i}(\pi)]_{l',l}\, \sigma(e_{k,i,l'}) \\ &= \sum_{l'=1}^{h_k} [T_{k,i}(\pi)]_{l',l} \sum_{l''=1}^{h_k} [T_{k,i}(\sigma)]_{l'',l'}\, e_{k,i,l''} \\ &= \sum_{l''=1}^{h_k} [T_{k,i}(\sigma)T_{k,i}(\pi)]_{l'',l}\, e_{k,i,l''} \end{aligned}$$

This representation is irreducible because the $H_{k,i}$ are. We define $V_k \subseteq \mathbb{R}[S_n]$ as the span of the functions $\pi \mapsto [T_{k,i}(\pi)]_{r,s}$ for $1 \leq r, s \leq h_k$, which is independent of the choice of i, as there is an isometry between $H_{k,i}$ and $H_{k,i'}$ for all i'.

The functions $z_{k,i,j}$, which give us the entries of the $E_{k,i,j}$ basis, lie in V_k:

$$\begin{aligned} z_{k,i,j}(\pi) &= E_{k,i,j}(\pi(1^i 0^{n-i}), 1^j 0^{n-j}) \\ &= \frac{1}{|V|} \sum_{l=1}^{h_k} e_{k,i,l}(\pi(1^i 0^{n-i})) e_{k,j,l}(1^j 0^{n-j}) \\ &= \frac{1}{|V|} \sum_{l=1}^{h_k} (\pi(e_{k,i,l}))\,(1^i 0^{n-i}) e_{k,j,l}(1^j 0^{n-j}) \\ &= \frac{1}{|V|} \sum_{l=1}^{h_k} \sum_{l'=1}^{h_k} [T_{k,i}(\pi)]_{l',l}\, e_{k,i,l'}(1^i 0^{n-i}) e_{k,j,l}(1^j 0^{n-j}) \end{aligned}$$

We call functions in V_k which are H-left-invariant and K-right-invariant *H-K-invariant functions* or *intertwining functions*. We will see that these functions form a one dimensional subspace of V_k.

Let us now fix an element $b \coloneqq 1^j 0^{n-j} \in \Omega_j$, called the *base point*. Notice here that the stabilizer of b is $H = S_j \times S_{n-j}$, so there is an isomorphism

$$S_n/H \to S_n b = \Omega_j, \quad \pi \mapsto \pi(b)$$

Since the H-(left)-invariant elements of $\mathbb{R}[S_n] \supseteq V_k$ can be seen as elements of $\mathbb{R}\left[{S_n}/{H}\right]$, we can now search for K-invariant elements in $\mathbb{R}^{\Omega_j} = M^{n-j,j}$, which turn K-right-invariant after applying the isomorphism.

If the vector v additionally lies in the submodule of $M^{n-j,j}$ equivalent to $S^{n-k,k}$, then we can use the base point to construct a vector in V_k:

$$\begin{aligned} z(\pi) &:= v(\pi(b)) \\ &= \sum_{l=1}^{h_k} c_l e_{k,i',l}(\pi(b)) \\ &= \sum_{l=1}^{h_k} c_l \sum_{l'=1}^{h_k} \left[T_{k,i'}(\pi)\right]_{l',l} e_{k,i',l'}(b) \\ &= \sum_{l,l'=1}^{h_k} \left[T_{k,i'}(\pi)\right]_{l',l} c_l e_{k,i',l'}(b) \in V_k \end{aligned}$$

since V_k is independent of i'.

Therefore we now want to search for $S_i \times S_{n-i}$-invariant elements in the submodule of $M^{n-j,j}$ equivalent to $S^{n-k,k}$.

4.2 Invariant functions

We start by determining the $K_i := S_i \times S_{n-i}$-invariant vectors in $M^{n-k,k}$. To do this we first define a multiplication on the Dirac functions for $A, B \subseteq \{1, \ldots, n\}$:

$$\delta_A \delta_B := \delta_{A \cup B}$$

This gives us a convenient way to use certain symmetric polynomials to construct invariant vectors:

$$\prod_{t=1}^{i} (\delta_{\{t\}}\alpha + 1) \prod_{t=i+1}^{n} (\delta_{\{t\}}\beta + 1)$$

This polynomial is invariant for permutations in K_i, since S_i acts on the first product, and S_{n-i} on the second. As a result the coefficients of monomials of degree k are K_i-invariant elements of $M^{n-k,k}$:

$$F^i_{kl} := \quad \text{coefficient of } \alpha^l \beta^{k-l} \text{ in above polynomial}$$

For example for $n = 4, i = 3$ we get the vectors:

$$\begin{aligned} &F^3_{00} = \delta_\emptyset \\ &F^3_{10} = \delta_{\{4\}}, \quad F^3_{11} = \delta_{\{1\}} + \delta_{\{2\}} + \delta_{\{3\}} \\ &F^3_{21} = \delta_{\{14\}} + \delta_{\{24\}} + \delta_{\{34\}}, \quad F^3_{22} = \delta_{\{12\}} + \delta_{\{23\}} + \delta_{\{13\}} \\ &F^3_{32} = \delta_{\{124\}} + \delta_{\{234\}} + \delta_{\{134\}}, \quad F^3_{3,3} = \delta_{\{123\}} \\ &F^3_{43} = \delta_{\{1234\}} \end{aligned}$$

For other parameters F^4_{kl} is zero. In general they are not trivial if $\max\{0, k+i-n\} \leq l \leq \min\{k, i\}$, and they are all orthogonal to each other since every Dirac function appears in exactly one of them for fixed n and i. In fact, they span the subspace of $\mathbb{R}^{\mathcal{P}(\{1,\dots,n\})}$ of K_i-invariant vectors:

$$L_{K_i} := \operatorname{span}\{F^i_{kl} \mid k = 0, \dots, n, \max\{0, k+i-n\} \leq l \leq \min\{k, i\}\}$$

as K_i has the same number of orbits as there are non zero F^i_{kl}, which is easy to check: Because K_i is $S_i \times S_{n-i}$ the orbit of a coordinate A depends only on the number of elements of $\{1, \dots, i\}$ and of $\{i+1, \dots, n\}$ in A, which means we can assign each orbit a pair $(k, k-l)$ with the same range as for the F^i_{kl}.

We now want to cut L_{K_i} with $\ker(d)$ to get vectors in the Specht modules. For that, let us first check what applying d to an F^i_{kl} does:

Lemma 4.2.1.

$$dF^i_{kl} = (i-l+1)F^i_{k-1,l-1} + (n-i-k+l+1)F^i_{k-1,l}$$

Proof. We can split the F^i_{kl} into a product corresponding to the two large products of the polynomial:

$$F^i_{kl} = \sigma_l(\{1, \dots, i\})\sigma_{k-l}(\{i+1, \dots, n\})$$

where we define

$$\sigma_l(A) := \sum_{B \subseteq A, |B|=l} \delta_B$$

Applying d to these parts on their own results in:

$$\begin{aligned} d\sigma_l(A) &= \sum_{B \subseteq A, |B|=l} d\delta_B \\ &= \sum_{B \subseteq A, |B|=l} \sum_{j \in B} \delta_{B \setminus \{j\}} \\ &= \sum_{B \subseteq A, |B|=l-1} \sum_{j \in A \setminus B} \delta_B \\ &= (|A| - l + 1)\sigma_{l-1}(A) \end{aligned}$$

Furthermore we can show a product formula for d (it can be seen as a differential operator, if we interpret coordinates as monomials):

$$\begin{aligned} d(\delta_A \delta_B) = d\delta_{A \cup B} &= \sum_{j \in A \cup B} \delta_{(A \cup B) \setminus \{j\}} \\ &= \sum_{j \in A} \delta_{A \setminus \{j\}} \delta_B + \sum_{j \in B} \delta_A \delta_{B \setminus \{j\}} = d\delta_A \delta_B + \delta_A d\delta_B \end{aligned}$$

From which follows the general formula $d(vw) = dvw + vdw$ since d is linear. Together we get

$$\begin{aligned} dF^i_{kl} &= d\left(\sigma_l(\{1, \dots, i\})\sigma_{k-l}(\{i+1, \dots, n\})\right) \\ &= (i-l+1)\sigma_{l-1}(\{1, \dots, i\})\sigma_{k-l}(\{i+1, \dots, n\}) \\ &\quad + \sigma_l(\{1, \dots, i\})(n-i-k+l+1)\sigma_{k-l-1}(\{i+1, \dots, n\}) \\ &= (i-l+1)F^i_{k-1,l-1} + (n-i+k+l+1)F^i_{k-1,l} \end{aligned}$$

□

We can now calculate the K_i-invariant elements in $S^{n-k,k} = M^{n-k,k} \cap \ker(d)$, which are also called *spherical functions*:

Theorem 4.2.2. *For $0 \leq i \leq n$ and $0 \leq k \leq \frac{n}{2}$:*

(i) If $i < k$ or $i > n-k$ then

$$L_{K_i} \cap S^{n-k,k} = \{0\}$$

(ii) If $k \leq i \leq n-k$ then $L_{K_i} \cap S^{n-k,k}$ is spanned by

$$\psi_{ki} := \sum_{j=0}^{k} (-1)^j \frac{(n-i-k+1)_j}{(i-j+1)_j} F^i_{k,j}$$

Proof. We have determined a base of L_{K_i}, of which the elements F^i_{kl} with parameters $a = \max\{0, k+i-n\} \leq l \leq \min\{k, i\} = b$ span $L_{K_i} \cap M^{n-k,k}$, so we want to find linear combinations of them which lie in $\ker(d)$:

$$\begin{aligned}
0 &= d \sum_{l=a}^{b} \alpha_l F^i_{kl} \\
&= \sum_{l=a}^{b} \alpha_l \left((i-l+1) F^i_{k-1,l-1} + (n-i-k+l+1) F^i_{k-1,l} \right) \\
&= \sum_{l=a}^{b-1} \left(\alpha_{l+1}(i-l) + \alpha_l (n-i-k+l+1) \right) F^i_{k-1,l} \\
&\quad + \alpha_b (n-i-k+b+1) F^i_{k-1,b} + \alpha_a (i-a+1) F^i_{k-1,a-1}
\end{aligned}$$

Since the F^i_{kl} are orthogonal to each other this gives us a system of linear equations.

If $i < k$ then $F^i_{k-1,b} = F^i_{k-1,i} \neq 0$, hence α_b has to be zero. Substituting $\alpha_b = 0$ into the other equations gives us $\alpha_{b-1} = 0$, and repeating it shows that all α_l have to be zero, so there are no invariant elements in $S^{n-k,k}$ for these parameters apart from 0. Analogously we can show that if $i > n-k$ then $F^i_{k-1,a-1} = F^i_{k-1,k+i-n-1} \neq 0$ and again all α_l have to be zero.

In the case $k \leq i \leq n-k$ we have $F^i_{k-1,b} = F^i_{k-1,a-1} = 0$, while the other $F^i_{k-1,l}$ appearing in the equations are all not zero. Solving

$$\alpha_{l+1}(i-l) + \alpha_l(n-i-k+l+1) = 0 \quad \text{for } 0 \leq l \leq k-1$$

results in the one dimensional space

$$\alpha_t = \alpha_0 \prod_{l=0}^{t-1} - \frac{n-i-k+l+1}{i-l} = \alpha_0 (-1)^t \frac{(n-i-k+1)_t}{(i-t+1)_t}$$

$\square$

We have now found the K-invariant elements in $S^{n-k,k}$, but we really wanted the K-invariant elements in the equivalent submodule in $M^{n-j,j}$. We assumed that $k \leq \frac{n}{2}$, so we know from the decomposition earlier that if $j < k$, then there is no such submodule, and with that no intertwining function. Otherwise we know that

$$S^{n-k,k} \cong (d^*)^{j-k} S^{n-k,k} \subseteq M^{n-j,j}$$

Hence

$$\psi_{kij} := (d^*)^{j-k} \psi_{ki}$$

is the function we now need to calculate.

Lemma 4.2.3. *For $k \leq j$ and $k + i - n \leq l \leq \min\{k, i\}$:*

$$(d^*)^{j-k} F^i_{kl} = (j-k)! \sum_{t=\max\{l,j-n+i\}}^{\min\{i,j-k+l\}} \binom{t}{l} \binom{j-t}{k-l} F^i_{jt}$$

Proof. We first check what multiple applications of d^* to δ_A with $A \in \Omega_k$ do. We have seen that

$$d^* \delta_A = \sum_{t \notin A} \delta_{A \cup \{t\}} = 1! \sum_{B \supseteq A, |B|=k+1} \delta_B$$

Inductively one finds the formula

$$(d^*)^{s+1} \delta_A = d^* (d^*)^s \delta_A = d^* s! \sum_{B \supseteq A, |B|=k+s} \delta_B = (s+1)! \sum_{B \supseteq A, |B|=k+s+1} \delta_B$$

so we have

$$(d^*)^{j-k} \delta_A = (j-k)! \sum_{B \supseteq A, |B|=j} \delta_B \quad \text{for } A \in \Omega_k$$

To express the F^i_{kl} in the Dirac basis we use the formula seen in the proof of lemma 4.2.1, where $I = \{1, \ldots, i\}$:

$$F^i_{kl} = \sigma_l(I) \sigma_{k-l}(I^c) = \left(\sum_{B \subseteq I, |B|=l} \delta_B \right) \left(\sum_{B \subseteq I^c, |B|=k-l} \delta_B \right) = \sum_{|B|=k, |B \cap I|=l} \delta_B$$

The operator d^* is linear, meaning we can combine the formulas to get

$$\begin{aligned}
(d^*)^{j-k} F^i_{kl} &= \sum_{\substack{|B|=k \\ |B \cap I|=l}} (d^*)^{j-k} \delta_B \\
&= \sum_{\substack{|B|=k \\ |B \cap I|=l}} (j-k)! \sum_{\substack{A \supseteq B \\ |A|=j}} \delta_A
\end{aligned}$$

The second sum can be split further by how much A intersects I. The minimum intersection is l if all new entries fit in I^c, otherwise it is $j-(n-i)$, while the maximum intersection is either all of I or $j-(k-l)$:

$$= (j-k)! \sum_{\substack{|B|=k \\ |B\cap I|=l}} \sum_{t=\max\{l,j-n+i\}}^{\min\{i,j-k+l\}} \sum_{\substack{A\supseteq B \\ |A|=j \\ |A\cap I|=t}} \delta_A$$

$$= (j-k)! \sum_{t=\max\{l,j-n+i\}}^{\min\{i,j-k+l\}} \sum_{\substack{|A|=j \\ |A\cap I|=t}} \sum_{\substack{|B|=k \\ |B\cap I|=l \\ B\subseteq A}} \delta_A$$

In the last sum we have to choose l elements of B in $A\cap I$ and the remaining $k-l$ elements in $A\setminus I$:

$$= (j-k)! \sum_{t=\max\{l,j-n+i\}}^{\min\{i,j-k+l\}} \sum_{\substack{|A|=j \\ |A\cap I|=t}} \binom{t}{l}\binom{j-t}{k-l}\delta_A$$

$$= (j-k)! \sum_{t=\max\{l,j-n+i\}}^{\min\{i,j-k+l\}} \binom{t}{l}\binom{j-t}{k-l} F^i_{jt}$$

$\square$

We can now calculate $(d^*)^{j-k}\psi_{ki}$, which we will use to determine the intertwining functions:

Theorem 4.2.4. *The $S_i \times S_{n-i}$ invariant elements in the submodule equivalent to $S^{n-k,k}$ in $M^{n-j,j}$ are spanned by:*

$$\psi_{kij} = (d^*)^{j-k}\psi_{ki} = (j-k)! \sum_{t=0}^{i} \left(\sum_{l=0}^{k} (-1)^l \frac{(n-i-k+1)_l}{(i-l+1)_l} \binom{t}{l}\binom{j-t}{k-l} \right) F^i_{jt}$$

Proof.

$$(d^*)^{j-k}\psi_{ki} = \sum_{l=0}^{k} (-1)^l \frac{(n-i-k+1)_l}{(i-l+1)_l} (d^*)^{j-k} F^i_{k,l}$$

$$= \sum_{l=0}^{k} (-1)^l \frac{(n-i-k+1)_l}{(i-l+1)_l} (j-k)! \sum_{t=\max\{l,j-n+i\}}^{\min\{i,j-k+l\}} \binom{t}{l}\binom{j-t}{k-l} F^i_{jt}$$

We can switch the two sums if we switch the sides of $t \geq l$ and $t \leq j-k+l$:

$$= (j-k)! \sum_{t=\max\{0,j-n+i\}}^{i}$$

$$\cdot \left(\sum_{l=\max\{0,t-j+k\}}^{\min\{t,k\}} (-1)^l \frac{(n-i-k+1)_l}{(i-l+1)_l} \binom{t}{l}\binom{j-t}{k-l} \right) F^i_{jt}$$

The ranges of the sums can be simplified: If $t < j-n+i$ then $F^i_{jt} = 0$, if $l < t-j+k$ then $\binom{j-t}{k-l} = 0$, and if $l > t$ then $\binom{t}{l} = 0$:

$$= (j-k)! \sum_{t=0}^{i} \left(\sum_{l=0}^{k} (-1)^l \frac{(n-i-k+1)_l}{(i-l+1)_l} \binom{t}{l} \binom{j-t}{k-l} \right) F^i_{jt}$$

$\square$

We can rewrite this formula with Hahn polynomials, which are a family of orthogonal polynomials for certain weights (which we will encounter later):

$$Q_k(t; -a-1, -b-1, j) \coloneqq \frac{1}{\binom{j}{k}} \sum_{l=0}^{k} (-1)^l \frac{\binom{b-k+l}{l}}{\binom{a}{l}} \binom{j-t}{k-l} \binom{t}{l}$$

Using the formula $\binom{n}{k} = \frac{(n-k+1)_k}{k!}$ from earlier we can see that this formula is, apart from a factor, the same as the one we found:

$$\psi_{kij} = (j-k)! \binom{j}{k} \sum_{t=0}^{i} Q_k(t; -i-1, -(n-i)-1, j) F^i_{jt}$$

Since we are only interested in the subspace spanned by ψ_{kij} we can ignore the factors before the sum.

If we now apply the (inverse of the) isomorphism ${S_n}/{H} \to S_n b = \Omega_j, \quad \pi \mapsto \pi(b)$ (with $b = 1^j 0^{n-j}$) to this function we find the span of intertwining, or H-K-invariant functions in V_k. To do this we first want to define

$$v'(\pi) \coloneqq t \in \mathbb{N} \quad \text{with } \left(F^i_{jt}\right)_{\pi(b)} \neq 0$$

which is well defined since the F^i_{jt} form an orthonormal basis of 0/1 vectors of the K invariant elements of $\mathbb{R}^{\Omega_j}$. So we can assign each permutation a number depending on in which K-orbit $\pi(b)$ lies. We have seen that F^i_{jt} is one at exactly the coordinates with t elements in $\{1, \dots, i\}$, hence

$$v'(\pi) = |\pi(\{1, \dots, j\}) \cap \{1, \dots, i\}|$$

Theorem 4.2.5. *Let $H = S_j \times S_{n-j}$, $K = S_i \times S_{n-i}$ and $V_k \subseteq \mathbb{R}^{S_n}$ be the span of the matrix entries of a representation corresponding to the irreducible module $S^{n-k,k}$. The H-K invariant vectors in V_k are spanned by*

$$\psi'_{k,H-K}(\pi) \coloneqq Q_k(v'(\pi); -i-1, -(n-i)-1, j)$$

Proof. By definition of $v'(\pi)$ we have

$$(\psi_{kij})_{\pi(b)} = (j-k)! \binom{j}{k} Q_k(v'(\pi); -i-1, -(n-i)-1, j)$$

The two factors can be left out because they are independent of π. $\square$

To make things a bit more consistent with other sources ([4], [6], [14]) and get the additional property $\psi_{kij}(\mathrm{id}) = 1$ we apply another tranformation to the Hahn polynomial:

Theorem 4.2.6. *The H-K invariant vectors in V_k are spanned by*

$$\psi_{k,H-K}(\pi) \coloneqq Q_k(v(\pi); -(n-i)-1, -i-1, j)$$

with

$$v(\pi) \coloneqq j - |\pi(\{1, \dots, j\}) \cap \{1, \dots, i\}| = j - v'(\pi)$$

Proof.

$$\begin{aligned}\psi'_{k,H-K}(\pi) &= Q_k(v'(\pi); -i-1, -(n-i)-1, j)\\ &= \frac{1}{\binom{j}{k}} \sum_{l=0}^{k} (-1)^l \frac{(n-i-k+l)!(i-l)!}{(n-i-k)!i!} \binom{v'(\pi)}{l} \binom{j-v'(\pi)}{k-l}\end{aligned}$$

Substituting $k-l$ for l gives us:

$$\begin{aligned}&= \frac{1}{\binom{j}{k}} (-1)^k \sum_{l=0}^{k} (-1)^l \frac{(n-i-l)!(i-k+l)!}{(n-i-k)!i!} \binom{v(\pi)}{l} \binom{j-v(\pi)}{k-l}\\ &= (-1)^k \frac{(i-k)!(n-i)!}{(n-i-k)!i!} \frac{1}{\binom{j}{k}}\\ &\quad \cdot \sum_{l=0}^{k} (-1)^l \frac{(n-i-l)!(i-k+l)!}{(i-k)!(n-i)!} \binom{v(\pi)}{l} \binom{j-v(\pi)}{k-l}\\ &= (-1)^k \frac{(i-k)!(n-i)!}{(n-i-k)!i!} Q_k(v(\pi); -(n-i)-1, -i-1, j)\end{aligned}$$

The factor before the Hahn polynomial is independent of π, so we can ignore it. □

4.3 The Isomorphism

We have now determined

$$z_{k,i,j} \colon S_n \to \mathbb{R}, \quad \pi \mapsto E_{k,i,j}(\pi(1^i 0^{n-i}), 1^j 0^{n-j})$$

up to a factor. To determine it, we now just need to calculate the norms of z_{kij} and $\psi_{k,H-K}$:

Lemma 4.3.1.

$$(z_{kij}, z_{kij}) = \binom{n}{i}^{-1} \binom{n}{j}^{-1} h_k$$

Proof.

$$
\begin{aligned}
(z_{kij}, z_{kij}) &= \frac{1}{|S_n|} \sum_{\pi \in S_n} z_{kij}(\pi)^2 \\
&= \frac{1}{n!} \sum_{\pi \in S_n} E_{k,i,j}(\pi(1^i 0^{n-i}), 1^j 0^{n-j})^2 \\
&= \frac{1}{n!} \sum_{\pi \in S_n} \left(\frac{1}{|V|} \sum_{l=1}^{h_k} e_{k,i,l}(\pi(1^i 0^{n-i})) e_{k,j,l}(1^j 0^{n-j}) \right)^2 \\
&= \frac{1}{n!|V|^2} \sum_{l,l'=1}^{h_k} \Bigg(e_{k,j,l}(1^j 0^{n-j}) e_{k,j,l'}(1^j 0^{n-j}) \\
&\qquad \cdot \sum_{\pi \in S_n} e_{k,i,l}(\pi(1^i 0^{n-i})) e_{k,i,l'}(\pi(1^i 0^{n-i})) \Bigg)
\end{aligned}
$$

Since $H_{k,i}$ has non-zero entries only on the coordinates with weight i, the right sum is exactly a multiple of $\langle e_{k,i,l}, e_{k,i,l'} \rangle$. The base is orthonormal, so this sum is zero for $l \neq l'$, and it sums over every coordinate of Ω_i as many times as there are elements in the stabilizer $|(S_n)_{1^i 0^{n-i}}| = |S_i \times S_{n-i}|$:

$$
\begin{aligned}
&= \frac{i!(n-i)!}{n!|V|} \sum_{l=1}^{h_k} e_{k,j,l}(1^j 0^{n-j})^2 \\
&= \binom{n}{i}^{-1} E_{k,j,j}(1^j 0^{n-j}, 1^j 0^{n-j})
\end{aligned}
$$

Since $E_{k,j,j}$ is S_n-invariant, every entry on the diagonal for coordinates of weight j is the same, and the others are zero because of the decomposition. So the entry above is exactly an $|\Omega_j| = \binom{n}{j}$'th of the trace of $E_{k,j,j}$:

$$
\operatorname{trace}(E_{k,j,j}) = \sum_{x \in \Omega_j} \frac{1}{|V|} \sum_{l=1}^{h_k} e_{k,j,l}(x)^2 = \sum_{l=1}^{h_k} (e_{k,j,l}, e_{k,j,l}) = h_k
$$

□

Lemma 4.3.2. *For $k \leq j \leq i \leq n-k$ we have*

$$
(\psi_{k,H-K}, \psi_{k,H-K}) = \frac{1}{h_k} \frac{(-i)_k (j-n)_k}{(-j)_k (i-n)_k}
$$

Proof. Here we require $j \leq i$ to make the calculations easier, but this is not a problem since $E_{kij}^T = E_{kji}$. First, let us calculate how often $v(\pi) = j - |\pi(\{1,\ldots,j\}) \cap \{1,\ldots,i\}| = j - |\pi(J) \cap I|$ assumes a certain value $0 \leq x \leq j$. It has to send the elements of J to $j-x$ positions in I, and x positions in I^c. At the same time we can permute all elements in J and in J^c freely, so we have:

$$
|\{\pi \in S_n \mid v(\pi) = x\}| = j!(n-j)! \binom{i}{j-x} \binom{n-i}{x}
$$

These are, with another $n!$ factor, exactly the before (not explicitly) mentioned weights the Hahn polynomials are orthogonal for, which also means that the norm for these weights is well known:

$$\begin{aligned}&\sum_{x=0}^{j} \frac{\binom{a}{x}\binom{b}{j-x}}{\binom{a+b}{j}} Q_k(x; -a-1, -b-1, j)^2 \\ = &\left[\binom{a+b}{k} - \binom{a+b}{k-1}\right]^{-1} \frac{(j-k)!(a-k)!(a+b-j)!b!}{j!a!(a+b-j-k)!(b-k)!}\end{aligned}$$

We are not going to prove this formula here, as its proof is quite long and technical (see [7] for an outline).

$$\begin{aligned}(\psi_{k,H-K}, \psi_{k,H-K}) &= \frac{1}{n!} \sum_{\pi \in S_n} Q_k(v(\pi); -(n-i)-1, -i-1, j)^2 \\ &= \sum_{x=0}^{j} \frac{\binom{n-i}{x}\binom{i}{j-x}}{\binom{n}{j}} Q_k(x; -(n-i)-1, -i-1, j)^2 \\ &= \frac{1}{h_k} \frac{(j-k)!(n-i-k)!(n-j)!i!}{j!(n-i)!(n-j-k)!(i-k)!} \\ &= \frac{1}{h_k} \frac{(-i)_k (j-n)_k}{(-j)_k (i-n)_k}\end{aligned}$$

Where we used $\frac{n!}{(n-k)!} = (n-k+1)_k = (-1)^k(-n)_k$ in the last step. □

We have now everything we need to determine $E_{k,i,j}$ explicitly:

Proposition 4.3.3. *For $k \leq j \leq i \leq n-k$ we have:*

$$E_{k,i,j}(x,y) = h_k \left(\binom{n}{i}\binom{n}{j} \frac{(-i)_k (j-n)_k}{(-j)_k (i-n)_k}\right)^{-\frac{1}{2}} Q_k(v(x,y); -(n-i)-1, -i-1, j)$$

if $x \in \Omega_i$ and $y \in \Omega_j$, otherwise $E_{k,i,j}(x,y) = 0$. For $i \leq j$ we have $E_{k,i,j} = E^T_{k,j,i}$. The function v is defined for $x \in \Omega_i$ and $y \in \Omega_j$ by:

$$v(x,y) = |\{l \mid x_l = 0, y_l = 1\}| = j - |x \cap y| = |y \setminus x|$$

Where we interpret $\Omega_l = \{0,1\}^n_{=l}$ in the first and $\Omega_l = \{A \subseteq \{1,\dots,n\} \mid |A| = l\}$ in the second and third formula.

Proof. By the definition of $E_{k,i,j}$ and the decomposition into the $H_{k,i}$ we know this matrix has to be zero for coordinates (x,y) with $x \notin \Omega_i$ or $y \notin \Omega_j$, and it is easy to see that $E^T_{k,i,j} = E_{k,j,i}$.

For the remaining coordinates we first determine z_{kij} with the correct scale:

$$\begin{aligned}|z_{kij}(\pi)| &= E_{k,i,j}(\pi(1^i 0^{n-i}), 1^j 0^{n-j}) \\ &= \frac{\sqrt{(z_{kij}, z_{kij})}}{\sqrt{(\psi_{k,H-K}, \psi_{k,H-K})}} \psi_{k,H-K} \\ &= h_k \left(\binom{n}{i}\binom{n}{j} \frac{(-i)_k (j-n)_k}{(-j)_k (i-n)_k}\right)^{-\frac{1}{2}} Q_k(v(\pi); -(n-i)-1, -i-1, j)\end{aligned}$$

The matrix $E_{k,i,j}$ is S_n invariant, hence the entry of the matrix at position (x, y) for $x \in \Omega_i$ and $y \in \Omega_j$ depends only on how much x and y overlap. Because we have $v(x, y) = v(\pi)$ for the coordinates above, we have determined the functions up to sign.

We know by definition that the diagonal entries of $E_{k,i,i}$ are non-negative, which are given by $z_{kii}(\mathrm{id})$. Because of $v(\mathrm{id}) = 0$ and

$$Q_k(0; -a-1, -b-1, j) = \frac{\binom{j}{k}}{\binom{j}{k}} = 1$$

the sign of z_{kii} has to be positive.

For the remaining signs positivity should follow by theorem 2.2.1, but I could not find a full argument. ☐

With this we can finally calculate the matrices $\varphi(B_r)$ for elements B_r of the canonical basis of $\mathcal{B}$, which are needed for the block diagonalization. Remember here that $\varphi(E_{k,i,j})$ was defined as the matrix with only a 1 in position i, j of the block corresponding to k.

Theorem 4.3.4. *For* $0 \leq s \leq r \leq n$*:*

$$\varphi(B_{(r,s,d)}) = \sum_{k=0}^{\min\{n-r,s\}} \frac{v_{r,s,d}}{h_k} E_{k,r,s}(x, y)\varphi(E_{k,r,s})$$

For any $(x, y) \in (r, s, d)$*, which is defined below. If* $s \geq r$*, then* $\varphi(B_{(r,s,d)}) = \varphi(B_{(s,r,r-s+d)})^T$*.*

Proof. Earlier we saw for S_n-orbits r of $V \times V$ that

$$\varphi(B_r) = \sum_{k=0}^{\lfloor \frac{n}{2} \rfloor} \sum_{i,j=k}^{n-k} p_r(k, i, j)\varphi(E_{k,i,j})$$

where $B_r = \sum p_r(k, i, j)E_{k,i,j}$. Shortly after we found a formula in proposition 2.2.3 to instead use use coefficients $q_{k,i,j}(r)$ of the "reverse" linear combination $E_{k,i,j} = \sum q_{k,i,j}(r)B_r$:

$$p_r(k, i, j) = \frac{|r|}{h_k} q_{k,i,j}(r)$$

Let us now first explicitly determine the S_n-orbits, giving us the elements B_r of the canonical basis of $\mathcal{B}$. If we have an element $(x, y) \in V \times V = \{0, 1\}^n \times \{0, 1\}^n$, then S_n operates on the pair by permuting the coordinates of the indices of x and y. This does neither change the Hamming distance between x and y (and so $v(\pi(x), \pi(y)) = v(x, y)$), nor does it change the weight of x or y. If we have a second pair (x', y') with the same distance and weights, then it is clear we can always find a $\pi \in S_n$ with $(\pi(x), \pi(y)) = (x', y')$, meaning the orbits can be indexed by triples:

$$(r, s, d) := \{(x, y) \in V \times V \mid |x| = r, |y| = s, v(x, y) = d\}$$

The size of the orbit is

$$v_{r,s,d} \coloneqq |(r,s,d)| = \underbrace{\binom{n}{d}}_{\substack{x_i=0\\ y_i=1}} \underbrace{\binom{n-d}{s-d}}_{\substack{x_i=1\\ y_i=1}} \underbrace{\binom{n-s}{r-s+d}}_{\substack{x_i=1\\ y_i=0}}$$

It is now simple to express the $E_{k,i,j}$ with elements of $B_{(r,s,d)}$: The entry $E_{k,i,j}(x,y)$ is zero if $x \notin \Omega_i$ or $y \notin \Omega_j$, and we have determined the value of the matrix depending on $v(x,y)$, so we have:

$$q_{k,i,j}(r,s,d) = \begin{cases} E_{k,i,j}(x,y) & \text{if } i=r, j=s \text{ for } (x,y) \in (r,s,d) \\ 0 & \text{otherwise} \end{cases}$$

Together we get for $(x,y) \in (r,s,d)$:

$$\begin{aligned} \varphi(B_{(r,s,d)}) &= \sum_{k=0}^{\lfloor \frac{n}{2} \rfloor} \sum_{i,j=k}^{n-k} p_{(r,s,d)}(k,i,j)\varphi(E_{k,i,j}) \\ &= \sum_{k=0}^{\lfloor \frac{n}{2} \rfloor} \sum_{i,j=k}^{n-k} \frac{v_{r,s,d}}{h_k} q_{k,i,j}(r,s,d)\varphi(E_{k,i,j}) \\ &= \sum_{k=0}^{\min\{n-r,s\}} \frac{v_{r,s,d}}{h_k} E_{k,r,s}(x,y)\varphi(E_{k,r,s}) \end{aligned}$$

The range of k was changed in the last step, because $E_{k,r,s}$ is indexed by $k \leq i,j \leq n-k$, hence we sum over zeroes for the k's outside of $k \leq s \leq r \leq n-k$.

Finally, to see that $\varphi(B_{(r,s,d)}) = \varphi(B_{(s,r,r-s+d)})^T$ for $s \geq r$, we use the same symmetry of the $E_{k,i,j}$ and $v(y,x) = |x| - |y| + v(x,y)$. $\square$

We could use this formula as-is to compute all bounds for up to $n = 17$, but afterwards we run into issues with double-precision solvers. Instead we can use a small transformation, which allows us to use integer valued matrices, increasing the range of usable n to 26. We do this with a very similar factor to the one used in [12] by Schrijver:

Theorem 4.3.5. *For all coefficients $\alpha_{r,s,d} \in \mathbb{R}$ with $\alpha_{r,s,d} = \alpha_{s,r,r-s+d}$:*

$$\sum_{r,s,d} \alpha_{r,s,d}\varphi(B_{(r,s,d)}) \succcurlyeq 0 \quad \Leftrightarrow \quad \sum_{r,s,d} \alpha_{r,s,d}\varphi'(B_{(r,s,d)}) \succcurlyeq 0$$

Where φ' is defined for $0 \leq s \leq r \leq n$ by:

$$\varphi'(B_{(r,s,d)}) = \sum_{k=0}^{\min\{n-r,s\}} v_{r,s,d}\binom{n-r}{k}\binom{s}{k} Q_k(d; -(n-r)-1, -r-1, s)\varphi(E_{k,r,s})$$

Proof. We require $\alpha_{r,s,d} = \alpha_{s,r,r-s+d}$ because we only consider symmetric matrices, and we had the same condition in the original convex program earlier in this thesis.

A symmetric block matrix is positive semidefinite if and only if all of its blocks are positive semidefinite, so we can modify each block separately, as long as their positivity is unchanged. A matrix $X \in S^N_{\succcurlyeq 0}$ is positive semidefinite if and only if it is the Gram matrix of vectors $x_1, \dots, x_N \in \mathbb{R}^N$:

$$X_{ij} = \langle x_i, x_j \rangle$$

Thus for any function $f\colon \{1, \dots, N\} \to \mathbb{R} \setminus \{0\}$ the following holds true:

$$X \succcurlyeq 0 \quad \Leftrightarrow \quad X' \succcurlyeq 0 \text{ with } X'_{ij} = f(i)f(j)X_{ij} = \langle f(i)x_i, f(j)x_j \rangle$$

The factor before the Hahn polynomial in our formula is not entirely symmetric in r and s, but it turns out we can remove most of it anyway. For this we set for $k = 0, \dots, \lfloor \frac{n}{2} \rfloor$:

$$f_k(i) = \frac{1}{k!}\sqrt{\frac{n!}{(n-k-i)!(i-k)!}}$$

This function is independent of d, which means that we can apply it to blocks we get by summing over different d's. Applying this function to the coefficients $p_{r,s,d}(k, r, s)$ results in the simplified formula:

$$\begin{aligned}
&f_k(r)f_k(s)p_{r,s,d}(k,r,s)\\
&\qquad = v_{r,s,d}\frac{n!}{k!k!}\Bigg((n-k-r)!(r-k)!(n-k-s)!(s-k)!\\
&\qquad\qquad \cdot \binom{n}{r}\binom{n}{s}\frac{(-r)_k(s-n)_k}{(-s)_k(r-n)_k}\Bigg)^{-\frac{1}{2}} Q_k(\dots)\\
&\qquad = v_{r,s,d}\frac{n!}{k!k!}\Bigg(\frac{1}{(n-k-r)!(r-k)!(n-k-s)!(s-k)}\\
&\qquad\qquad \cdot \frac{(n-r)!r!(n-s)!s!(r-k)!(n-s-k)!(n-r)!s!}{n!n!(s-k)!(n-r-k)!(n-s)!r!}\Bigg)^{\frac{1}{2}} Q_k(\dots)\\
&\qquad = v_{r,s,d}\frac{(n-r)!s!}{k!k!(n-r-k)!(s-k)!}Q_k(\dots)\\
&\qquad = v_{r,s,d}\binom{n-r}{k}\binom{s}{k}Q_k(\dots)
\end{aligned}$$

□

Chapter 5

Applying the symmetry reduction

At this point we have determined the isomorphism φ, and have to apply it to the Lovász-Theta function introduced in section 1.3. Before we do this for the general case, we will first block diagonalize the case $D = 0$, which describes ordinary binary block codes, and interpret the result.

5.1 The case $D = 0$

Here we simply have $V = \{0,1\}^n$, and the set of S_n-invariant matrices is exactly the Terwilliger algebra $\mathcal{B} \subseteq \mathbb{R}^{V \times V}$. The Lovász-Theta number for our graph is, as we have defined earlier, the value of the following program:

$$\max \ \{\langle J, X\rangle \ \mid \ \mathrm{tr}(X) = 1, \ \ X_{ij} = 0 \text{ if } \mathrm{d}(i,j) < d \text{ and } i \neq j, \ \ X \geq 0, \ \ X \in S^V_{\succcurlyeq 0}\}$$

To determine the block diagonalization we have to calculate the values $c_r := \langle J, B_r\rangle$ and $a_{ir} := \langle A_i, B_r\rangle$ for every orbit $r = (r, s, d')$ and constraint $\langle A_i, X\rangle = b_i$. The elements B_r of the canonical basis were defined to be one at coordinates $(x, y) \in r$, and zero otherwise, hence $c_{(r,s,d')} = |(r, s, d')| = v_{r,s,d'}$. All constraints except the trace one are, apart from an auxiliary variable to turn the inequalities into equalities, of the form $\langle E_{ij}, X\rangle = b_i$. Because of

$$\langle E_{ij}, B_{r,s,d'}\rangle = \begin{cases} 1 & \text{if } r = |i|, s = |j|, d' = v(i,j) \\ 0 & \text{otherwise} \end{cases}$$

a lot of constraints are one and the same after the block diagonalization. We only need one for each orbit for $X \geq 0$ and another one for every orbit with $2d' + r - s < d$. We get this formula by calculating the Hamming distance from the earlier defined v, which was used for the orbits:

$$\mathrm{d}(x, y) = |(x \setminus y) \cup (y \setminus x)| = |x| - |x \cap y| + v(x, y) = |x| - |y| + 2v(x, y)$$

Furthermore the constraint $X_{ij} \geq 0$ can be removed for orbits with $2d'+r-s < d$ as it is weaker than $X_{ij} = 0$. Finally we have $\operatorname{tr}(X) = \langle \mathbb{1}, X\rangle = 1$, which is only affects orbits with $r = s$ and $d = 2d' + r - s = 2d' = 0$ with factor $v_{r,r,0} = \binom{n}{r}$.

Theorem 5.1.1. *The Lovász-Theta (prime) number of the graph $G_{n,0,d}$ is the value of the following block-diagonalized program:*

$$\begin{aligned}
p(n,0,d) \coloneqq \max \quad & \sum_{(r,s,d')} v_{r,s,d'} z_{r,s,d'} \\
& \sum_{(r,r,0)} v_{r,r,0} z_{r,r,0} = 1 \\
& z_{r,s,d'} = 0 \quad \textit{if} \quad 2d' + r - s < d \ \textit{and} \ (r,s,d') \neq (r,r,0) \\
& z_{r,s,d'} \geq 0 \quad \textit{if} \quad 2d' + r - s \geq d \ \textit{or} \ (r,s,d') = (r,r,0) \\
& z_{r,s,d'} = z_{s,r,r-s+d'} \\
& \sum_{(r,s,d')} z_{r,s,d'} \varphi'(B_{(r,s,d')}) \succcurlyeq 0 \\
& z_{r,s,d'} \in \mathbb{R} \ \textit{for all } S_n\textit{-Orbits } (r,s,d') \ \textit{of } V \times V
\end{aligned}$$

Proof. We applied the block diagonalization (2.0.1) with the values calculated above. Afterwards we removed factors from two sets of constraints in the middle, since these do not influence the result. □

The program can be simplified a bit further if one wants to implement it: The variables which are always zero can be removed fully without changing the outcome, and substituting

$$z'_{r,s,d'} = \frac{z_{r,s,d'}}{v_{r,s,d'}}$$

removes all $v_{r,s,d'}$ factors, including from within $\varphi'(B_{(r,s,d')})$. Note that it does not conflict with $z_{r,s,d'} = z_{s,r,r-s+d'}$ because $v_{r,s,d'} = v_{s,r,r-s+d'}$.

Here one can interpret the variables nicely: They are a similar to the *distance distribution* (see [9]) of a code:

$$a_i = \frac{1}{|C|} |\{(x,y) \mid \mathrm{d}(x,y) = i\}|$$

which are the variables of Delsarte's linear programming bound. It is clear that

$$|C| = \sum_i a_i \qquad\qquad a_0 = 1 \qquad\qquad a_1, \ldots, a_{d-1} = 0$$

for any code C with minimum distance d.

Similarly, for our program we saw that for a code C the matrix $X = \frac{xx^T}{x^Tx} = \frac{xx^T}{|C|}$ is feasible after forming the group average, where x is the characteristic vector of C. Hence we get a feasible set of variables for the block diagonalized form by

$$z'_{r,s,d'} = \frac{1}{|C|} |\{(x,y) \mid |x| = r, |y| = s, v(x,y) = d'\}|$$

because of the definition of $B_{r,s,d'}$ as characteristic function of the Γ-orbits. If we set

$$a_i = \sum_{\substack{r,s,d' \\ 2d'+r-s=i}} z'_{r,s,d'}$$

then it is easy to see that the three properties of the a_i translate directly to the conditions and objective function of our program:

$$|C| = \sum_{r,s,d'} z'_{r,s,d'} \qquad \sum_{r} z'_{r,r,0} = 1 \qquad z'_{r,s,d'} = 0 \text{ if } 2d' + r - s < d$$

It is even possible to calculate a feasible set of $z'_{r,s,d'}$ starting with a distance distribution a_i. To do this one uses that the Delsarte bound is the block diagonalized Lovász-Theta number of the graph $G_{n,0,d}$, if one additionally includes bit-switches in Γ (see [11]). Hence if we require the $z'_{r,s,d'}$ set to be invariant under the bit-switches, we do get enough conditions to uniquely determine them if we know the a_i.

To actually solve this program we used Brian Borchers' CSDP ([3]) by generating SDPA files first, for which we need to transform the program into a primal SDP of the form

$$\max \ \{\langle C, X\rangle \ | \ \langle A_i, X\rangle = b_i \ \text{ for } i = 1, \ldots, m, X \succcurlyeq 0\}$$

To do this we remove one of the variables $z_{r,s,d'}$, $z_{s,r,r-s+d'}$ for each such pair, allowing us to replace the matrices $\varphi'(B_{(r,s,d')})$ with the symmetric matrices $\varphi'(B_{(r,s,d')}) + \varphi'(B_{(s,r,r-s+d')})$. Afterwards the program is in the negative dual form of above program:

$$-\min \ \{y^T b \ | \ y_1 A_1 + \ldots + y_m A_m - C \succcurlyeq 0\}$$

The actual transformation is then straightforward.

5.2 The general case

We can easily find the diagonalized programs for the case $D \neq 0$ since this just removes the vertices of the graph with weight smaller than D, which corresponds to a subset of the orbits. That means we simply set $z_{r,s,d'} = 0$ if $r < D$ or $s < D$ (or remove the variables in the first variant) to get the program for the general case:

Theorem 5.2.1. *The Lovász-Theta (prime) number of the graph $G_{n,D,d}$ is the* *value of the following block-diagonalized program:*

$$\begin{aligned} p(n, D, d) \coloneqq \max \quad & \sum_{(r,s,d') \in R} v_{r,s,d'} z_{r,s,d'} \\ & \sum_{(r,r,0) \in R} v_{r,r,0} z_{r,r,0} = 1 \\ & z_{r,s,d'} \geq 0 \quad \forall (r,s,d') \in R \\ & z_{r,s,d'} = z_{s,r,r-s+d'} \quad \forall (r,s,d') \in R \\ & \sum_{(r,s,d') \in R} z_{r,s,d'} \varphi'(B_{(r,s,d')}) \succcurlyeq 0 \\ & z_{r,s,d'} \in \mathbb{R} \quad \forall (r,s,d') \in R \end{aligned}$$

Where R is the set of relevant orbits:

$$R \coloneqq \Big\{ (r,s,d') \;\mid\; r \geq D, s \geq D, \Big(2d' + r - s \geq d \;\; \textit{or} \;\; (r,s,d') = (r,r,0) \Big) \Big\}$$

Note here that we can additionally size down the matrices $\varphi'(B_r)$ a bit, as $\varphi'(B_{(r,s,d')})$ only has entries at coordinates (r,s) in each block, implying these rows and columns are zero for indices smaller than D.

A small remark about solving the SDPs in practice: To further avoid rounding errors as many factors as possible were cancelled out, including from within the Hahn polynomials. The solver needs the most time in cases with small d and small D, since then only a few/no orbits do not correspond to variables. Solving the programs with $D = d = 0$ takes about 0.8 seconds at $n = 10$, 3.2 seconds at $n = 15$, 6.6 seconds at $n = 20$ and 19.3 seconds at $n = 25$. It is significantly faster at larger d and D.

Chapter 6

Results

First we take a look at the case $D = 0$: The values of these programs match the known bound of Delsarte for binary error-correcting codes of length n and distance d, which was expected since it is exactly the Lovász-Theta prime number of the graphs $G_{n,0,d}$ (which was proven in [11]).

Before we take a look at the data for the general case, we should consider if there is a similar property to $A(n, d) = A(n+1, d+1)$ for odd d for binary codes. But it turns out we only get a partial analogue for unequal error protection codes:

Proposition 6.0.1.

$$A(n, D, d) = A(n + 1, D + 1, d + 1) \quad \textit{if } D \textit{ and } d \textit{ are odd}$$

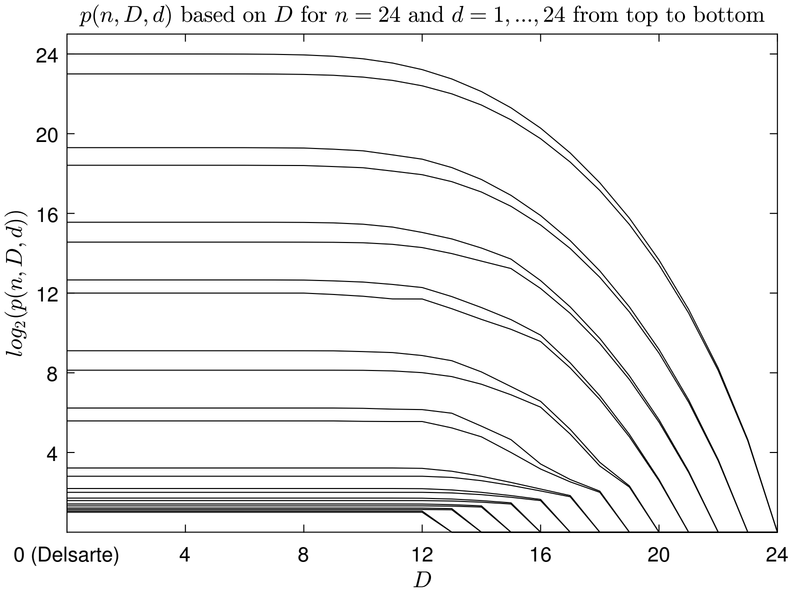
Proof. We can extend an $A(n, D, d)$ code with a parity bit (add a bit to each word, such that it has an even number of ones), which increases the minimum weight to an even number. The minimum distance increases to an even number as well, since two words with an even number of ones can only have an even distance, and the additional bit cannot decrease the distance between words.

Conversely if we have an $A(n+1, D+1, d+1)$ code removing the last bit (or any other bit as long as its the same for each word) decreases n by one, while also decreasing D and d by at most one. □

We do not have a similar property if D is even and d is odd, so we would need to include both cases $A(n, D, d)$ and $A(n+1, D+1, d+1)$ here. One could argue that one is only interested in codes with both an odd D and odd d (as even parameters do not correct more errors), but for completeness sake we will include the data for all parameters here, including $D < d$.

Behaviour based on D

In the general case we can observe that, if we fix d, the upper bound for the amount of words we can fit into $\{0,1\}^n_{\geq D}$ falls off with increasing D at about the same rate as $|\{0,1\}^n_{\geq D}|$ decreases:



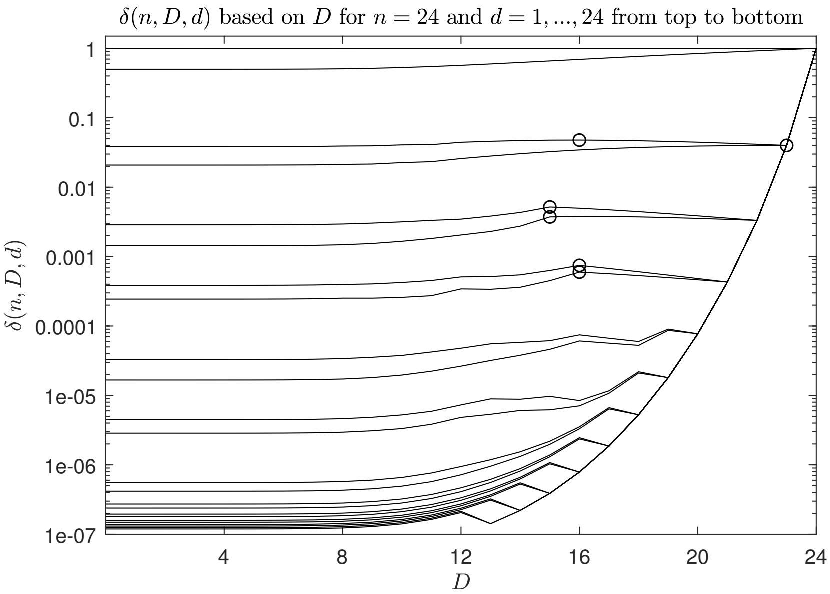
Note that until about $D = \lfloor \frac{n}{2} \rfloor$ the graphs falls off only slightly, because $\Omega_{n/2}$ is by far the largest part of the decomposition for large n. This somewhat visualizes the earlier mentioned result of [2], that for any $D \leq \frac{n}{2}$ the optimal rate of a binary code can still be reached asymptotically by increasing n, more on that later.

The graphs appear to be in pairs $(d_1, d_2) = (2t-1, 2t)$, starting close to each other and converging at the same time at 0. In practice we would always choose a code with odd minimum distance d, the upper of the two graphs, as the amount of errors that can be corrected is $\lfloor \frac{d-1}{2} \rfloor$, which is the same for d_1 and d_2.

Behaviour based on D, density variant

It might be of interest to see which codes have the highest density within $\{0,1\}^n_{\geq D}$, so here are the graphs for

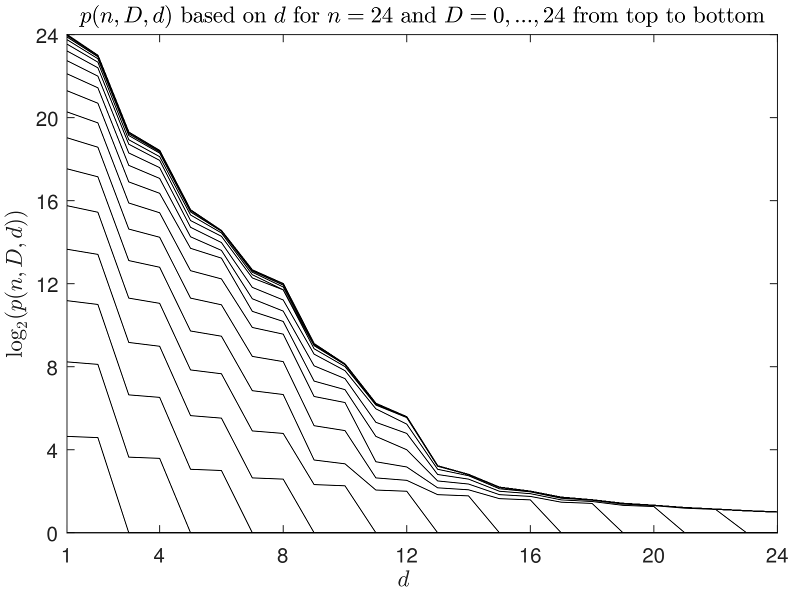
$$\delta(n, D, d) = \frac{p(n, D, d)}{|\sum_{k=D}^{n} \binom{n}{k}|}$$



Obviously we can fill it completely for $D = n - 1$, where there is just one free spot for a single code word. In the graphs all "interesting" local maxima are marked, which seem to lie around $D = \frac{2}{3}n$ for most n and small d it was tested for. For example it might worth looking into the case $A(24, 16, 7)$, the second lowest of the marked points. The graphs for larger d always have a local maximum at the highest D with just enough space for two words in $\{0,1\}^n_{\leq D}$, and all graphs of course join once there is only space for a single word.

Behaviour based on d

If we fix D and instead increase d we get graphs that should look familiar, at least the top one for $D = 0$. Usually the Delsarte bound is calculated only for every second d, since $A(n, d) = A(n+1, d+1)$ if d is odd, resulting in a smoother curve.



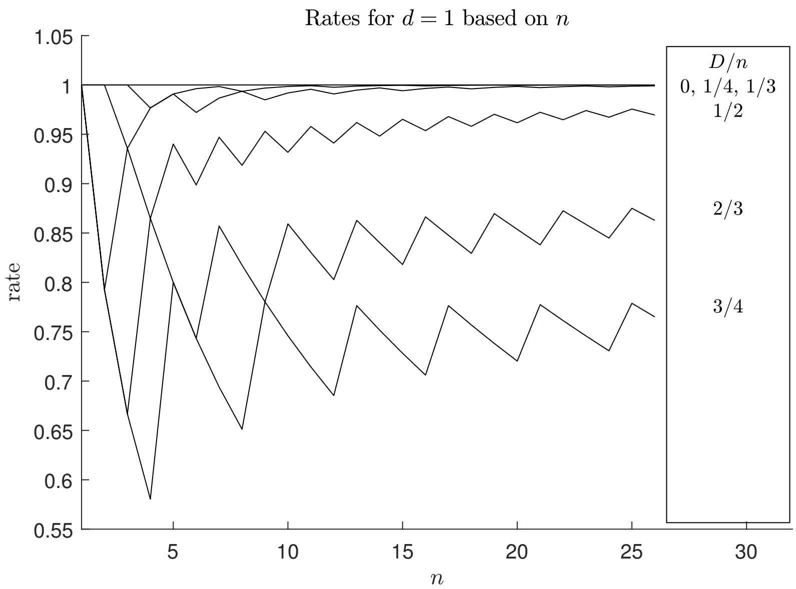
We see again that the graphs closer to the top (small D) behave very similarly, in fact the graphs for $D = 0, \ldots, 12$ are drawn nearly on top of each other here.

Behaviour based on n, for $d = 1$

Our upper bounds for $A(n, D, d)$ do give us upper bounds for the code rates by

$$r = \frac{\log_2(A(n, D, d))}{n}$$

In [2] the authors proved that the optimal asymptotic rate of a sequence of binary codes with fixed $\frac{d}{n}$ can be reached asymptotically by UEP codes with one special message for up to $\frac{D}{n} = \frac{1}{2}$ and any $\frac{d}{n} \in [0, \frac{1}{2}]$. This bound for D is sharp for $\frac{d}{n} = 0$ (e.g. d is constant), which we can visualize:

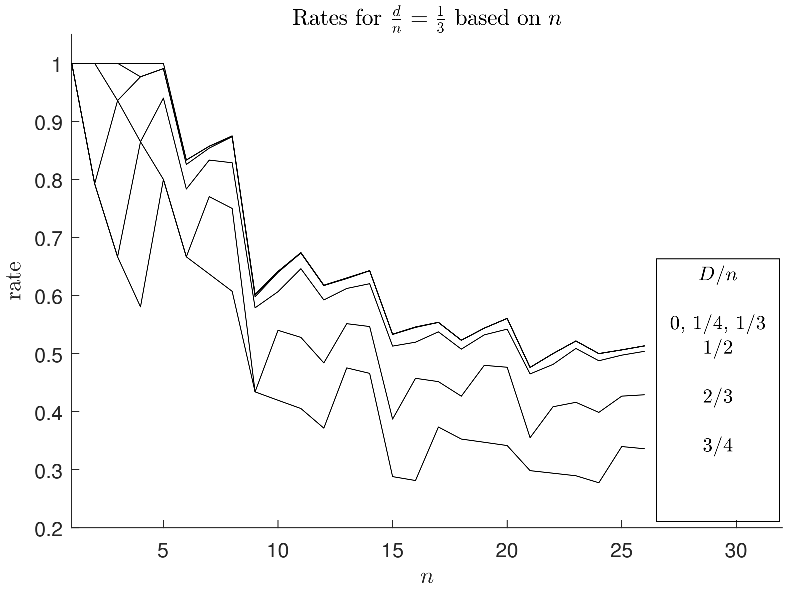


The graphs for $\frac{D}{n} < \frac{1}{2}$ do clearly converge in 1, the optimal rate of codes with $d = 1$. The case $\frac{D}{n} = \lfloor \frac{n}{2} \rfloor$ is close, and looks like it might still converge with 1, but even if it does not its would not be a contradiction, as it is only proven that there is an optimal sequence which approaches $\frac{D}{n} = \frac{1}{2}$ for $n \to \infty$.

The cases with $\frac{D}{n} > \frac{n}{2}$ do not look like they will converge with $D = 0$, as was proven in [2].

Behaviour based on n, for $\frac{d}{n} = \frac{1}{3}$

And finally, the case $\frac{d}{n} = \frac{1}{3}$, which was proven to reach the optimal rate for $\frac{D}{n} \leq \frac{n}{2}$. While proven in the case $d = 1$, the authors of [2] only expected that this bound is sharp for other ratios $\frac{d}{n}$. While we do not have enough data to be sure (and definitely cannot prove it), it does seem to be correct:



In conclusion, we can always set $\frac{D}{n} = \frac{n}{2}$, adding one special, strongly protected word to binary error correcting codes, without lowering the rate of the code by much, even for small n. Here there seems to be a rate-loss of about 5% at $n = 24$, which might still be too much to be practical, but it is definitely lower than one might expect.

Appendix A

Raw data

$n = 1$	Delsarte	$D = 0$	1
$d = 1$	2	2	1

$n = 2$	Delsarte	$D = 0$	1	2
$d = 1$	4	4	3	1
2	2	2	2	1

$n = 3$	Delsarte	$D = 0$	1	2	3
$d = 1$	8	8	7	4	1
2	4	4	4	3	1
3	2	2	2	1	1

$n = 4$	Delsarte	$D = 0$	1	2	3	4
$d = 1$	16	16	15	11	5	1
2	8	8	8	7	4	1
3	2.66667	2.66667	2.6	2.5	1	1
4	2	2	2	2	1	1

$n = 5$	Delsarte	$D = 0$	1	2	3	4	5
$d = 1$	32	32	31	26	16	6	1
2	16	16	16	15	11	5	1
3	4	4	4	4	3	1	1
4	2.66667	2.66667	2.66667	2.6	2.5	1	1
5	2	2	2	2	1	1	1

$n=6$	Delsarte	$D=0$	1	2	3	4	5	6
$d=1$	64	64	63	57	42	22	7	1
2	32	32	32	31	26	16	6	1
3	8	8	8	8	7	3.5	1	1
4	4	4	4	4	4	3	1	1
5	2.4	2.4	2.39529	2.36364	2.33333	1	1	1
6	2	2	2	2	2	1	1	1

$n=7$	Delsarte	$D=0$	1	2	3	4	5	6	7
$d=1$	128	128	127	120	99	64	29	8	1
2	64	64	64	63	57	42	22	7	1
3	16	16	16	15	15	9.33333	4	1	1
4	8	8	8	8	8	7	3.5	1	1
5	3	3	3	3	3	2.66667	1	1	1
6	2.4	2.4	2.4	2.39529	2.36364	2.33333	1	1	1
7	2	2	2	2	2	1	1	1	1

$n=8$	Delsarte	$D=0$	1	2	3	4	5	6	7	8
$d=1$	256	256	255	247	219	163	93	37	9	1
2	128	128	128	127	120	99	64	29	8	1
3	25.6	25.6	25.5808	25.5	23	22	12	4.5	1	1
4	16	16	16	16	15	15	9.33333	4	1	1
5	4	4	4	4	4	3.85714	3	1	1	1
6	3	3	3	3	3	3	2.66667	1	1	1
7	2.28571	2.28571	2.28521	2.2804	2.26316	2.25	1	1	1	1
8	2	2	2	2	2	2	1	1	1	1

$n=9$	Delsarte	$D=0$	1	2	3	4	5	6	7	8	9
$d=1$	512	512	511	502	466	382	256	130	46	10	1
2	256	256	256	255	247	219	163	93	37	9	1
3	42.6667	42.6667	42.6667	42.6667	41.6667	37	31	15	5	1	1
4	25.6	25.6	25.6	25.5808	25.5	23	22	12	4.5	1	1
5	6	6	6	6	6	6	5	3.33333	1	1	1
6	4	4	4	4	4	4	3.85714	3	1	1	1
7	2.66667	2.66667	2.66667	2.66667	2.66667	2.66667	2.5	1	1	1	1
8	2.28571	2.28571	2.28571	2.28521	2.2804	2.26316	2.25	1	1	1	1
9	2	2	2	2	2	2	1	1	1	1	1

$n=10$	Delsarte	$D=0$	1	2	3	4	5	6	7	8	9	10
$d=1$	1024	1024	1023	1013	968	848	638	386	176	56	11	1
2	512	512	512	511	502	466	382	256	130	46	10	1
3	85.3333	85.3333	85.3333	85.3333	84.3333	73.646	67	42.25	18.3333	5.5	1	1
4	42.6667	42.6667	42.6667	42.6667	42.6667	41.6667	37	31	15	5	1	1
5	12	12	12	12	12	12	11	6.6	3.66667	1	1	1
6	6	6	6	6	6	6	6	5	3.33333	1	1	1
7	3.2	3.2	3.2	3.2	3.2	3.19608	3.14286	2.75	1	1	1	1
8	2.66667	2.66667	2.66667	2.66667	2.66667	2.66667	2.66667	2.5	1	1	1	1
9	2.22222	2.22222	2.22215	2.22139	2.21732	2.2069	2.2	1	1	1	1	1
10	2	2	2	2	2	2	2	1	1	1	1	1

$n=11$	Delsarte	$D=0$	1	2	3	4	5	6	7	8	9	10	11
$d=1$	2048	2048	2047	2036	1981	1816	1486	1024	562	232	67	12	1
2	1024	1024	1024	1023	1013	968	848	638	386	176	56	11	1
3	170.667	170.667	170.667	169.667	169.667	152.797	138	100	56	22	6	1	1
4	85.3333	85.3333	85.3333	85.3333	85.3333	84.3333	73.646	67	42.25	18.3333	5.5	1	1
5	24	24	24	24	24	23	23	15.84	9	4	1	1	1
6	12	12	12	12	12	12	12	11	6.6	3.66667	1	1	1
7	4	4	4	4	4	4	4	3.69231	3	1	1	1	1
8	3.2	3.2	3.2	3.2	3.2	3.2	3.19608	3.14286	2.75	1	1	1	1
9	2.5	2.5	2.5	2.5	2.5	2.5	2.5	2.4	1	1	1	1	1
10	2.22222	2.22222	2.22222	2.22215	2.22139	2.21732	2.2069	2.2	1	1	1	1	1
11	2	2	2	2	2	2	2	1	1	1	1	1	1

$n=12$	Delsarte	$D=0$	1	2	3	4	5	6	7	8	9	10	11	12
$d=1$	4096	4096	4095	4083	4017	3797	3302	2510	1586	794	299	79	13	1
2	2048	2048	2048	2047	2036	1981	1816	1486	1024	562	232	67	12	1
3	292.571	292.571	292.562	292.5	289.923	287	248.687	221	144	72.5	26	6.5	1	1
4	170.667	170.667	170.667	170.667	169.667	169.667	152.797	138	100	56	22	6	1	1
5	40	40	39.9996	39.9798	39.8872	39.7825	36.7694	35.32	23.4	13	4.33333	1	1	1
6	24	24	24	24	24	24	23	23	15.84	9	4	1	1	1
7	5.33333	5.33333	5.33333	5.33333	5.33333	5.33333	5.33333	5.2	4.33333	3.25	1	1	1	1
8	4	4	4	4	4	4	4	4	3.69231	3	1	1	1	1
9	2.85714	2.85714	2.85714	2.85714	2.85714	2.85714	2.85185	2.82609	2.6	1	1	1	1	1
10	2.5	2.5	2.5	2.5	2.5	2.5	2.5	2.5	2.4	1	1	1	1	1
11	2.18182	2.18182	2.18181	2.18167	2.18081	2.17751	2.17073	2.16667	1	1	1	1	1	1
12	2	2	2	2	2	2	2	2	1	1	1	1	1	1

$n=13$	Delsarte	$D=0$	1	2	3	4	5	6	7	8	9	10	11	12	13
$d=1$	8192	8192	8191	8178	8100	7814	7099	5812	4096	2380	1093	378	92	14	1
2	4096	4096	4096	4095	4083	4017	3797	3302	2510	1586	794	299	79	13	1
3	512	512	512	512	511	503.888	486.333	411.667	340.667	201.2	92	30.3333	7	1	1
4	292.571	292.571	292.571	292.562	292.5	289.923	287	248.687	221	144	72.5	26	6.5	1	1
5	64	64	64	64	64	64	63	56.9089	51.5556	36.4	15.1667	4.66667	1	1	1
6	40	40	40	39.9996	39.9798	39.8872	39.7825	36.7694	35.32	23.4	13	4.33333	1	1	1
7	8	8	8	8	8	8	8	8	7	5.09091	3.5	1	1	1	1
8	5.33333	5.33333	5.33333	5.33333	5.33333	5.33333	5.33333	5.33333	5.2	4.33333	3.25	1	1	1	1
9	3.33333	3.33333	3.33333	3.33333	3.33333	3.33333	3.33333	3.33333	3.18182	2.8	1	1	1	1	1
10	2.85714	2.85714	2.85714	2.85714	2.85714	2.85714	2.85714	2.85185	2.82609	2.6	1	1	1	1	1
11	2.4	2.4	2.4	2.4	2.4	2.4	2.4	2.4	2.33333	1	1	1	1	1	1
12	2.18182	2.18182	2.18182	2.18181	2.18167	2.18081	2.17751	2.17073	2.16667	1	1	1	1	1	1
13	2	2	2	2	2	2	2	2	1	1	1	1	1	1	1

$n = 14$	Delsarte	$D = 0$	1	2	3	4	5	6	7	8	9	10	11	12	13	14
$d = 1$	16384	16384	16383	16369	16278	15914	14913	12911	9908	6476	3473	1471	470	106	15	1
2	8192	8192	8192	8191	8178	8100	7814	7099	5812	4096	2380	1093	378	92	14	1
3	1024	1024	1024	1024	1023	1002.63	989	849.3	750	508	274	114.75	35	7.5	1	1
4	512	512	512	512	512	511	503.888	486.333	411.667	340.667	201.2	92	30.3333	7	1	1
5	128	128	128	128	128	128	127	113.839	96	74.3871	45.5	17.5	5	1	1	1
6	64	64	64	64	64	64	64	63	56.9089	51.5556	36.4	15.1667	4.66667	1	1	1
7	16	16	16	16	16	16	16	16	15	10	6	3.75	1	1	1	1
8	8	8	8	8	8	8	8	8	8	7	5.09091	3.5	1	1	1	1
9	4	4	4	4	4	4	4	4	3.94737	3.57143	3	1	1	1	1	1
10	3.33333	3.33333	3.33333	3.33333	3.33333	3.33333	3.33333	3.33333	3.33333	3.18182	2.8	1	1	1	1	1
11	2.66667	2.66667	2.66667	2.66667	2.66667	2.66667	2.66667	2.66154	2.64706	2.5	1	1	1	1	1	1
12	2.4	2.4	2.4	2.4	2.4	2.4	2.4	2.4	2.4	2.33333	1	1	1	1	1	1
13	2.15385	2.15385	2.15384	2.15382	2.15363	2.15277	2.15012	2.14545	2.14286	1	1	1	1	1	1	1
14	2	2	2	2	2	2	2	2	2	1	1	1	1	1	1	1

$n = 15$	Delsarte	$D = 0$	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
$d = 1$	32768	32768	32767	32752	32647	32192	30827	27824	22819	16384	9949	4944	1941	576	121	16	1
2	16384	16384	16384	16383	16369	16278	15914	14913	12911	9908	6476	3473	1471	470	106	15	1
3	2048	2048	2048	2047	2047	2012	1984	1767.14	1571	1184	736	365	141	40	8	1	1
4	1024	1024	1024	1024	1024	1023	1002.63	989	849.3	750	508	274	114.75	35	7.5	1	1
5	256	256	256	256	256	255.264	255	228.882	207	157	113	56	20	5.33333	1	1	1
6	128	128	128	128	128	128	128	127	113.839	96	74.3871	45.5	17.5	5	1	1	1
7	32	32	32	32	32	32	32	31	31	22.8571	16	7.11111	4	1	1	1	1
8	16	16	16	16	16	16	16	16	16	15	10	6	3.75	1	1	1	1
9	5	5	5	5	5	5	5	5	5	4.70588	4	3.2	1	1	1	1	1
10	4	4	4	4	4	4	4	4	4	3.94737	3.57143	3	1	1	1	1	1
11	3	3	3	3	3	3	3	3	3	2.90909	2.66667	1	1	1	1	1	1
12	2.66667	2.66667	2.66667	2.66667	2.66667	2.66667	2.66667	2.66667	2.66154	2.64706	2.5	1	1	1	1	1	1
13	2.33333	2.33333	2.33333	2.33333	2.33333	2.33333	2.33333	2.33333	2.33333	2.28571	1	1	1	1	1	1	1
14	2.15385	2.15385	2.15385	2.15384	2.15382	2.15363	2.15277	2.15012	2.14545	2.14286	1	1	1	1	1	1	1
15	2	2	2	2	2	2	2	2	2	1	1	1	1	1	1	1	1

$n = 16$	Delsarte	$D = 0$	1	2	3	4	5	6
$d = 1$	65536	65536	65535	65519	65399	64839	63019	58651
2	32768	32768	32768	32767	32752	32647	32192	30827
3	3640.89	3640.89	3640.88	3640.83	3638.12	3633.33	3558.01	3470.42
4	2048	2048	2048	2048	2047	2047	2012	1984
5	425.558	425.558	425.558	425.556	425.535	425.38	422.007	416.335
6	256	256	256	256	256	256	255.264	255
7	50.717	50.717	50.717	50.717	50.717	50.717	50.7087	50.5674
8	32	32	32	32	32	32	32	32
9	6.66667	6.66667	6.66667	6.66667	6.66667	6.66667	6.66667	6.66667
10	5	5	5	5	5	5	5	5
11	3.42857	3.42857	3.42857	3.42857	3.42857	3.42857	3.42857	3.42857
12	3	3	3	3	3	3	3	3
13	2.54545	2.54545	2.54545	2.54545	2.54545	2.54545	2.54545	2.54539
14	2.33333	2.33333	2.33333	2.33333	2.33333	2.33333	2.33333	2.33333
15	2.13333	2.13333	2.13333	2.13333	2.13328	2.13306	2.13225	2.1301
16	2	2	2	2	2	2	2	2

$n = 16$	7	8	9	10	11	12	13	14	15	16
$d = 1$	50643	39203	26333	14893	6885	2517	697	137	17	1
2	27824	22819	16384	9949	4944	1941	576	121	16	1
3	2981.68	2602	1813.33	1039.83	477	171	45.3333	8.5	1	1
4	1767.14	1571	1184	736	365	141	40	8	1	1
5	363.596	318.293	250.333	159.667	68	22.6667	5.66667	1	1	1
6	228.882	207	157	113	56	20	5.33333	1	1	1
7	47.9082	46.7692	38.8571	20.1481	8.5	4.25	1	1	1	1
8	31	31	22.8571	16	7.11111	4	1	1	1	1
9	6.66667	6.53846	5.66667	4.47368	3.4	1	1	1	1	1
10	5	5	4.70588	4	3.2	1	1	1	1	1
11	3.42775	3.4	3.1875	2.83333	1	1	1	1	1	1
12	3	3	2.90909	2.66667	1	1	1	1	1	1
13	2.54088	2.53191	2.42857	1	1	1	1	1	1	1
14	2.33333	2.33333	2.28571	1	1	1	1	1	1	1
15	2.12676	2.125	1	1	1	1	1	1	1	1
16	2	2	1	1	1	1	1	1	1	1

$n=17$	Delsarte	$D=0$	1	2	3	4	5	6	7
$d=1$	131072	131072	131071	131054	130918	130238	127858	121670	109294
2	65536	65536	65536	65535	65519	65399	64839	63019	58651
3	6553.6	6553.6	6553.6	6553.6	6552.6	6543.14	6508.6	6324.49	6007.89
4	3640.89	3640.89	3640.89	3640.88	3640.83	3638.12	3633.33	3558.01	3470.42
5	682.667	682.667	682.667	682.667	682.667	682.667	681.667	676.312	661.01
6	425.558	425.558	425.558	425.558	425.556	425.535	425.38	422.007	416.335
7	81.4545	81.4545	81.4545	81.4545	81.4545	81.4545	81.4545	81.3818	80.0093
8	50.717	50.717	50.717	50.717	50.717	50.717	50.717	50.7087	50.5674
9	10	10	10	10	10	10	10	10	10
10	6.66667	6.66667	6.66667	6.66667	6.66667	6.66667	6.66667	6.66667	6.66667
11	4	4	4	4	4	4	4	4	4
12	3.42857	3.42857	3.42857	3.42857	3.42857	3.42857	3.42857	3.42857	3.42857
13	2.8	2.8	2.8	2.8	2.8	2.8	2.8	2.8	2.8
14	2.54545	2.54545	2.54545	2.54545	2.54545	2.54545	2.54545	2.54545	2.54539
15	2.28571	2.28571	2.28571	2.28571	2.28571	2.28571	2.28571	2.28571	2.28571
16	2.13333	2.13333	2.13333	2.13333	2.13333	2.13328	2.13306	2.13225	2.1301
17	2	2	2	2	2	2	2	2	2

$n=17$	8	9	10	11	12	13	14	15	16	17
$d=1$	89846	65536	41226	21778	9402	3214	834	154	18	1
2	50643	39203	26333	14893	6885	2517	697	137	17	1
3	5041.12	4183	2703	1437	613	205	51	9	1	1
4	2981.68	2602	1813.33	1039.83	477	171	45.3333	8.5	1	1
5	563.218	481.857	409	205	81.6	25.5	6	1	1	1
6	363.596	318.293	250.333	159.667	68	22.6667	5.66667	1	1	1
7	74.3572	67.9375	51.8095	25.5	10.2857	4.5	1	1	1	1
8	47.9082	46.7692	38.8571	20.1481	8.5	4.25	1	1	1	1
9	10	9	6.92308	5	3.6	1	1	1	1	1
10	6.66667	6.53846	5.66667	4.47368	3.4	1	1	1	1	1
11	4	3.85714	3.48387	3	1	1	1	1	1	1
12	3.42775	3.4	3.1875	2.83333	1	1	1	1	1	1
13	2.8	2.73913	2.57143	1	1	1	1	1	1	1
14	2.54088	2.53191	2.42857	1	1	1	1	1	1	1
15	2.28571	2.25	1	1	1	1	1	1	1	1
16	2.12676	2.125	1	1	1	1	1	1	1	1
17	2	1	1	1	1	1	1	1	1	1

$n=18$	Delsarte	$D=0$	1	2	3	4	5	6	7
$d=1$	262144	262144	262143	262125	261972	261156	258096	249528	230964
2	131072	131072	131072	131071	131054	130918	130238	127858	121670
3	13107.2	13107.2	13107.2	13107.2	13106.2	13070.4	13050.2	12641.1	12243
4	6553.6	6553.6	6553.6	6553.6	6553.6	6552.6	6543.14	6508.6	6324.49
5	1289.48	1289.48	1289.48	1289.48	1289.48	1289.46	1288.46	1280.98	1240.54
6	682.667	682.667	682.667	682.667	682.667	682.667	682.667	681.667	676.312
7	145.297	145.297	145.297	145.297	145.297	145.297	145.297	145.297	144.128
8	81.4545	81.4545	81.4545	81.4545	81.4545	81.4545	81.4545	81.4545	81.3818
9	20	20	20	20	20	20	20	20	20
10	10	10	10	10	10	10	10	10	10
11	4.8	4.8	4.8	4.8	4.8	4.8	4.8	4.8	4.8
12	4	4	4	4	4	4	4	4	4
13	3.11111	3.11111	3.11111	3.11111	3.11111	3.11111	3.11111	3.11111	3.11111
14	2.8	2.8	2.8	2.8	2.8	2.8	2.8	2.8	2.8
15	2.46154	2.46154	2.46154	2.46154	2.46154	2.46154	2.46154	2.46154	2.46136
16	2.28571	2.28571	2.28571	2.28571	2.28571	2.28571	2.28571	2.28571	2.28571
17	2.11765	2.11765	2.11765	2.11765	2.11764	2.11758	2.11734	2.11658	2.11484
18	2	2	2	2	2	2	2	2	2

$n = 18$	8	9	10	11	12	13	14	15	16	17	18
$d = 1$	199140	155382	106762	63004	31180	12616	4048	988	172	19	1
2	109294	89846	65536	41226	21778	9402	3214	834	154	18	1
3	10532.9	9174.2	6541.75	3933	1947.5	776.2	243.25	57	9.5	1	1
4	6007.89	5041.12	4183	2703	1437	613	205	51	9	1	1
5	1107.07	913.369	752.469	554.714	259.4	96.9	28.5	6.33333	1	1	1
6	661.01	563.218	481.857	409	205	81.6	25.5	6	1	1	1
7	137.782	123.216	100.587	69.2143	32.5714	12.6667	4.75	1	1	1	1
8	80.0093	74.3572	67.9375	51.8095	25.5	10.2857	4.5	1	1	1	1
9	20	19	13.5714	8.63636	5.58824	3.8	1	1	1	1	1
10	10	10	9	6.92308	5	3.6	1	1	1	1	1
11	4.8	4.75	4.38462	3.8	3.16667	1	1	1	1	1	1
12	4	4	3.85714	3.48387	3	1	1	1	1	1	1
13	3.10959	3.09302	2.95556	2.71429	1	1	1	1	1	1	1
14	2.8	2.8	2.73913	2.57143	1	1	1	1	1	1	1
15	2.45756	2.45161	2.375	1	1	1	1	1	1	1	1
16	2.28571	2.28571	2.25	1	1	1	1	1	1	1	1
17	2.11236	2.11111	1	1	1	1	1	1	1	1	1
18	2	2	1	1	1	1	1	1	1	1	1

$n = 19$	Delsarte	$D = 0$	1	2	3	4	5	6	7	8
$d = 1$	524288	524288	524287	524268	524097	523128	519252	507624	480492	430104
2	262144	262144	262144	262143	262125	261972	261156	258096	249528	230964
3	26214.4	26214.4	26214.4	26213.4	26213.4	26156.4	26108.4	25386.9	24715.6	21948.2
4	13107.2	13107.2	13107.2	13107.2	13107.2	13106.2	13070.4	13050.2	12641.1	12243
5	2373.08	2373.08	2373.08	2373.08	2373.08	2373.08	2372.08	2351.27	2300.12	2096.33
6	1289.48	1289.48	1289.48	1289.48	1289.48	1289.48	1289.46	1288.46	1280.98	1240.54
7	290.595	290.595	290.595	290.595	290.595	290.595	290.595	290.217	288.079	276.901
8	145.297	145.297	145.297	145.297	145.297	145.297	145.297	145.297	145.297	144.128
9	40	40	40	40	40	40	40	40	39.4629	39
10	20	20	20	20	20	20	20	20	20	20
11	6	6	6	6.00001	6	6	6	6	6	6
12	4.8	4.8	4.8	4.8	4.8	4.8	4.8	4.8	4.8	4.8
13	3.5	3.5	3.5	3.5	3.5	3.5	3.5	3.5	3.5	3.5
14	3.11111	3.11111	3.11111	3.11111	3.11111	3.11111	3.11111	3.11111	3.11111	3.11111
15	2.66667	2.66667	2.66667	2.66667	2.66667	2.66667	2.66667	2.66667	2.66667	2.66667
16	2.46154	2.46154	2.46154	2.46154	2.46154	2.46154	2.46154	2.46154	2.46154	2.46136
17	2.25	2.25	2.25	2.25	2.25	2.25	2.25	2.25	2.25	2.25
18	2.11765	2.11765	2.11765	2.11765	2.11765	2.11764	2.11758	2.11734	2.11658	2.11484
19	2	2	2	2	2	2	2	2	2	2

$n = 19$	9	10	11	12	13	14	15	16	17	18	19
$d = 1$	354522	262144	169766	94184	43796	16664	5036	1160	191	20	1
2	199140	155382	106762	63004	31180	12616	4048	988	172	19	1
3	19390	14966.7	9976	5600.48	2594	970	286	63.3333	10	1	1
4	10532.9	9174.2	6541.75	3933	1947.5	776.2	243.25	57	9.5	1	1
5	1835.55	1519.46	1200.71	739.286	324	114	31.6667	6.66667	1	1	1
6	1107.07	913.369	752.469	554.714	259.4	96.9	28.5	6.33333	1	1	1
7	253.28	222.305	154.462	93.0612	42.2222	16	5	1	1	1	1
8	137.782	123.216	100.587	69.2143	32.5714	12.6667	4.75	1	1	1	1
9	39	30.1587	21.5909	11.1111	6.25	4	1	1	1	1	1
10	20	19	13.5714	8.63636	5.58824	3.8	1	1	1	1	1
11	6	5.71429	5	4.13793	3.33333	1	1	1	1	1	1
12	4.8	4.75	4.38462	3.8	3.16667	1	1	1	1	1	1
13	3.5	3.41463	3.18182	2.85714	1	1	1	1	1	1	1
14	3.10959	3.09302	2.95556	2.71429	1	1	1	1	1	1	1
15	2.66667	2.62295	2.5	1	1	1	1	1	1	1	1
16	2.45756	2.45161	2.375	1	1	1	1	1	1	1	1
17	2.25	2.22222	1	1	1	1	1	1	1	1	1
18	2.11236	2.11111	1	1	1	1	1	1	1	1	1
19	2	1	1	1	1	1	1	1	1	1	1

$n = 20$	Delsarte	$D = 0$	1	2	3	4	5	6
$d = 1$	1.04858e+06	1.04858e+06	1.04858e+06	1.04855e+06	1.04836e+06	1.04723e+06	1.04238e+06	1.02688e+06
2	524288	524288	524288	524287	524268	524098	523128	519253
3	47662.5	47662.6	47662.6	47662.5	47659.7	47653	47523.5	47330
4	26214.4	26214.4	26214.4	26214.4	26213.4	26213.4	26156.4	26108.4
5	4443.12	4443.12	4443.12	4443.12	4443.11	4442.93	4441.86	4428.42
6	2373.08	2373.08	2373.08	2373.08	2373.08	2373.08	2373.08	2372.08
7	571.745	571.535	571.535	571.535	571.535	571.534	571.497	570.532
8	290.595	290.595	290.595	290.595	290.595	290.595	290.595	290.595
9	64	64	64	64	64	64	64	64
10	40	40	40	40	40	40	40	40
11	8	8	8	8	8	8	8	8
12	6	6	6	6	6	6	6	6
13	4	4	4	4	4	4	4	4
14	3.5	3.5	3.5	3.5	3.5	3.5	3.5	3.5
15	2.90909	2.90909	2.90909	2.90909	2.90909	2.90909	2.90909	2.90909
16	2.66667	2.66667	2.66667	2.66667	2.66667	2.66667	2.66667	2.66667
17	2.4	2.4	2.4	2.4	2.4	2.4	2.4	2.4
18	2.25	2.25	2.25	2.25	2.25	2.25	2.25	2.25
19	2.10526	2.10526	2.10526	2.10526	2.10526	2.10525	2.10518	2.10493
20	2	2	2	2	2	2	2	2

$n = 20$	7	8	9	10	11	12	13	14	15	16	17	18	19	20
$d = 1$	988116	910600	784626	616666	431910	263950	137980	60460	21700	6196	1351	211	21	1
2	507624	480493	430104	354522	262144	169766	94184	43796	16664	5036	1160	191	20	1
3	45854.4	44007.6	37933	32795	23808	14868.5	7822	3402	1198	333.5	70	10.5	1	1
4	25386.9	24715.6	21948.2	19390	14966.7	9976	5600.48	2594	970	286	63.3333	10	1	1
5	4373.74	4210.73	3692.64	3063.31	2401.18	1939	970	400	133	35	7	1	1	1
6	2351.27	2300.12	2096.33	1835.55	1519.46	1200.71	739.286	324	114	31.6667	6.66667	1	1	1
7	568.159	555.076	516.852	465.741	359.077	245.286	126.667	56	21	5.25	1	1	1	1
8	290.217	288.079	276.901	253.28	222.305	154.462	93.0612	42.2222	16	5	1	1	1	1
9	63.9785	63.1351	60.8927	59.5092	50.3788	29.1667	15	7	4.2	1	1	1	1	1
10	40	39.4629	39	39	30.1587	21.5909	11.1111	6.25	4	1	1	1	1	1
11	8	8	8	7.875	7	5.72727	4.5	3.5	1	1	1	1	1	1
12	6	6	6	6	5.71429	5	4.13793	3.33333	1	1	1	1	1	1
13	4	4	4	3.97297	3.76923	3.4186	3	1	1	1	1	1	1	1
14	3.5	3.5	3.5	3.5	3.41463	3.18182	2.85714	1	1	1	1	1	1	1
15	2.90909	2.90909	2.90728	2.89655	2.8	2.625	1	1	1	1	1	1	1	1
16	2.66667	2.66667	2.66667	2.66667	2.62295	2.5	1	1	1	1	1	1	1	1
17	2.4	2.39973	2.39655	2.39241	2.33333	1	1	1	1	1	1	1	1	1
18	2.25	2.25	2.25	2.25	2.22222	1	1	1	1	1	1	1	1	1
19	2.10424	2.10281	2.10092	2.1	1	1	1	1	1	1	1	1	1	1
20	2	2	2	2	1	1	1	1	1	1	1	1	1	1

$n = 21$	Delsarte	$D = 0$	1	2	3	4	5	6	7
$d = 1$	2.09715e+06	2.09715e+06	2.09715e+06	2.09713e+06	2.09693e+06	2.09559e+06	2.08962e+06	2.06926e+06	2.01499e+06
2	1.04858e+06	1.04858e+06	1.04857e+06	1.04857e+06	1.04856e+06	1.04836e+06	1.04722e+06	1.04237e+06	1.02687e+06
3	87381.3	87381.4	87381.4	87381.3	87380.6	87368.6	87311.7	86937	86105.1
4	47662.5	47662.5	47662.5	47662.5	47662.5	47659.6	47653	47523.5	47329.9
5	7723.89	7723.88	7723.88	7723.89	7723.88	7723.88	7722.56	7716.73	7681.76
6	4443.12	4443.12	4443.12	4443.11	4443.12	4443.11	4442.94	4441.86	4428.42
7	1024	1024	1024	1024	1024	1024	1024	1024	1023
8	571.535	571.534	571.534	571.535	571.534	571.534	571.533	571.497	570.531
9	95.3191	95.3191	95.3181	95.3191	95.3191	95.3191	95.3191	95.3192	95.3191
10	64	63.9999	64	64	64	63.9999	64	64	64
11	15.9999	12	12	12	12	12	12	12	12
12	8	8	8	8	8	8	8	8	8
13	4.66667	4.66667	4.66667	4.66667	4.66666	4.66667	4.66667	4.66667	4.66667
14	4	4	4	4	4	4	4	4	4
15	3.2	3.2	3.2	3.2	3.2	3.2	3.2	3.2	3.2
16	2.90909	2.90909	2.90909	2.90909	2.90909	2.90909	2.90909	2.90909	2.90909
17	2.57143	2.57143	2.57143	2.57143	2.57143	2.57143	2.57143	2.57143	2.57143
18	2.4	2.4	2.4	2.4	2.4	2.4	2.4	2.4	2.4
19	2.22222	2.22222	2.22222	2.22222	2.22222	2.22222	2.22222	2.22222	2.22222
20	2.10526	2.10526	2.10526	2.10526	2.10526	2.10526	2.10525	2.10518	2.10493
21	2	2	2	2	2	2	2	2	2

$n = 21$	8	9	10	11	12	13	14	15	16	17	18	19	20	21
$d = 1$	1.89871e+06	1.69524e+06	1.40129e+06	1.04858e+06	695860	401930	198440	82160	27896	7547	1562	232	22	1
2	988116	910594	784626	616665	431910	263950	137980	60460	21700	6196	1351	211	21	1
3	82687.3	77132.8	65168.3	54142	36994	21704	10736	4400	1464	386	77	11	1	1
4	45854.4	44007.5	37933	32795	23808	14868.5	7822	3402	1198	333.5	70	10.5	1	1
5	7551.61	7209.98	6146.56	5193.01	3782.4	2665.75	1255	488.667	154	38.5	7.33333	1	1	1
6	4373.74	4210.73	3692.66	3063.31	2401.18	1939	970	400	133	35	7	1	1	1
7	990.716	967.016	870.182	759.592	598.143	349.333	176	77	23.1	5.5	1	1	1	1
8	568.159	555.076	516.852	465.741	359.077	245.286	126.667	56	21	5.25	1	1	1	1
9	95.2211	93.7671	89.4569	83.923	68.9258	41.25	22	7.85714	4.4	1	1	1	1	1
10	63.9786	63.1351	60.8928	59.5092	50.3788	29.1667	15	7	4.2	1	1	1	1	1
11	12	11.9998	12	11	8.8	6.6	4.88889	3.66667	1	1	1	1	1	1
12	8	8	8	7.875	7	5.72727	4.5	3.5	1	1	1	1	1	1
13	4.66667	4.66666	4.66667	4.52941	4.16216	3.66667	3.14286	1	1	1	1	1	1	1
14	4	4	4	3.97297	3.76923	3.4186	3	1	1	1	1	1	1	1
15	3.2	3.2	3.2	3.14286	2.98305	2.75	1	1	1	1	1	1	1	1
16	2.90909	2.90909	2.90728	2.89655	2.8	2.625	1	1	1	1	1	1	1	1
17	2.57143	2.57143	2.57143	2.53846	2.44444	1	1	1	1	1	1	1	1	1
18	2.4	2.39973	2.39655	2.39241	2.33333	1	1	1	1	1	1	1	1	1
19	2.22222	2.22222	2.22222	2.2	1	1	1	1	1	1	1	1	1	1
20	2.10424	2.10281	2.10092	2.1	1	1	1	1	1	1	1	1	1	1
21	2	2	2	1	1	1	1	1	1	1	1	1	1	1

$n = 22$	Delsarte	$D = 0$	1	2	3	4	5	6	7	8
$d = 1$	4.1943e+06	4.19431e+06	4.1944e+06	4.19431e+06	4.19407e+06	4.19267e+06	4.18534e+06	4.15902e+06	4.08442e+06	3.91372e+06
2	2.09715e+06	2.09714e+06	2.0972e+06	2.09715e+06	2.09715e+06	2.0969e+06	2.0956e+06	2.0896e+06	2.06928e+06	2.01502e+06
3	174763	174762	174763	174762	174761	174706	174679	173652	172991	165985
4	87381.3	87381.2	87381.3	87381.3	87381.1	87380.2	87368.6	87312.4	86936.7	86104.8
5	13775.9	13775.9	13775.9	13775.9	13775.9	13775.9	13774.9	13769.8	13738.9	13591.8
6	7723.89	7723.88	7723.87	7723.84	7723.86	7723.86	7723.88	7722.54	7716.67	7681.68
7	2048	2048.01	2048	2048.08	2048	2048	2048	2048	2047	1936.61
8	1024	1024	1024	1024	1024	1024	1024	1023.98	1024	1023
9	151.864	151.864	151.864	151.864	151.864	151.863	151.864	151.864	151.864	151.853
10	95.3191	95.3189	95.3191	95.3191	95.3187	95.3186	95.3195	95.32	95.3191	95.3191
11	24	23.9995	23.9999	24	24	23.9996	23.9989	24	24	24
12	12	12	12	12	12	12	12	12	12	12
13	5.6	5.6	5.6	5.59999	5.59996	5.6	5.6	5.6	5.6	5.60001
14	4.66667	4.66667	4.66667	4.66667	4.66667	4.66667	4.66667	4.66667	4.66667	4.66667
15	3.55556	3.55555	3.55556	3.55556	3.55556	3.55556	3.55555	3.55556	3.55556	3.55556
16	3.2	3.2	3.2	3.2	3.2	3.2	3.2	3.2	3.19999	3.2
17	2.76923	2.76921	2.76923	2.76921	2.76923	2.76923	2.76923	2.76923	2.76923	2.76923
18	2.57143	2.57143	2.57143	2.57143	2.57143	2.57143	2.57143	2.57143	2.57143	2.57143
19	2.35294	2.35294	2.35294	2.35293	2.35294	2.35294	2.35294	2.35294	2.35294	2.35294
20	2.22222	2.22222	2.22222	2.22222	2.22222	2.22222	2.22222	2.22222	2.22222	2.22222
21	2.09524	2.09524	2.09522	2.09523	2.09524	2.09523	2.09521	2.09514	2.09489	2.09427
22	2	2	2	2	2	2	2	2	2	2

$n = 22$	9	10	11	12	13	14	15	16	17	18	19	20	21	22
$d = 1$	3.59394e+06	3.09653e+06	2.44991e+06	1.74445e+06	1.09779e+06	600370	280600	110056	35443	9109	1794	254	23	1
2	1.89872e+06	1.69522e+06	1.40131e+06	1.04857e+06	695859	401930	198440	82160	27896	7547	1562	232	22	1
3	158067	136992	118511	87339.8	56252.1	31088.4	14505.3	5619.67	1772	443.75	84.3333	11.5	1	1
4	82686.8	77133.7	65168.3	54142	36994	21704	10736	4400	1464	386	77	11	1	1
5	13033.4	11942.6	9913.04	7987.07	6016.42	3606.25	1603.33	591.333	177.1	42.1667	7.66667	1	1	1
6	7551.59	7209.97	6146.57	5193.01	3782.4	2665.75	1255	488.667	154	38.5	7.33333	1	1	1
7	1837.25	1799.26	1541	1112.34	831.84	507	253	88.55	25.3	5.75	1	1	1	1
8	990.716	967.016	870.182	759.598	598.143	349.333	176	77	23.1	5.5	1	1	1	1
9	150.864	146.159	136.783	118.507	90.3571	63.25	25.8163	8.84615	4.6	1	1	1	1	1
10	95.2211	93.7668	89.4563	83.9231	68.9258	41.25	22	7.85714	4.4	1	1	1	1	1
11	24	24	23	17.25	11.5	7.66667	5.30769	3.83333	1	1	1	1	1	1
12	12	11.9999	12	11	8.8	6.6	4.88889	3.66667	1	1	1	1	1	1
13	5.6	5.6	5.55172	5.19355	4.6	3.92683	3.28571	1	1	1	1	1	1	1
14	4.66667	4.66667	4.66667	4.52941	4.16216	3.66667	3.14286	1	1	1	1	1	1	1
15	3.55556	3.55528	3.53846	3.40741	3.17241	2.875	1	1	1	1	1	1	1	1
16	3.2	3.2	3.2	3.14286	2.98305	2.75	1	1	1	1	1	1	1	1
17	2.76923	2.76737	2.76	2.68831	2.55556	1	1	1	1	1	1	1	1	1
18	2.57143	2.57143	2.57143	2.53846	2.44444	1	1	1	1	1	1	1	1	1
19	2.3526	2.34995	2.34694	2.3	1	1	1	1	1	1	1	1	1	1
20	2.22222	2.22222	2.22222	2.2	1	1	1	1	1	1	1	1	1	1
21	2.09308	2.0916	2.09091	1	1	1	1	1	1	1	1	1	1	1
22	2	2	2	1	1	1	1	1	1	1	1	1	1	1

$n=23$	Delsarte	$D=0$	1	2	3	4	5	6	7	8
$d=1$	8.38861e+06	8.38857e+06	8.38845e+06	8.38851e+06	8.38843e+06	8.38651e+06	8.37763e+06	8.34465e+06	8.24302e+06	7.99788e+06
2	4.1943e+06	4.19412e+06	4.19424e+06	4.19407e+06	4.19439e+06	4.194e+06	4.19232e+06	4.18498e+06	4.15895e+06	4.08448e+06
3	349525	349527	349525	349523	349523	349444	349374	347667	346316	334128
4	174763	174764	174764	174763	174761	174760	174711	174674	173651	172988
5	24107.9	24107.8	24107.8	24107.8	24107.8	24107.8	24106.8	24103.1	24073.8	23961.8
6	13775.9	13775.8	13776.1	13775.7	13775.9	13775.7	13775.6	13774.9	13769.6	13738.6
7	4096	4095.97	4095.97	4095.98	4095.99	4094.99	4094.96	4095	4095.27	3906.18
8	2048	2047.99	2048.02	2047.93	2047.99	2048.41	2048	2047.86	2048	2046.97
9	280	280	279.994	279.999	279.999	279.984	279.991	279.964	279.987	279.977
10	151.864	151.864	151.861	151.856	151.863	151.862	151.861	151.864	151.858	151.864
11	48	47.9999	47.9987	47.9977	48	47.9986	47.9982	47.9976	47.9971	47.999
12	24	23.9997	23.9998	24	23.9996	23.9987	24.0004	23.9991	23.9991	23.9994
13	7	6.99999	7	6.99999	6.99996	7	7	7	7	7
14	5.6	5.6	5.6	5.6	5.6	5.6	5.6	5.6	5.6	5.6
15	4	3.99997	4	4	3.99998	4	3.99994	4	3.99996	4
16	3.55556	3.55555	3.55555	3.55556	3.55539	3.55555	3.55555	3.55555	3.55556	3.55555
17	3	3	3	3	2.99995	3	3	2.99999	3	3
18	2.76923	2.76923	2.76921	2.76923	2.76923	2.76923	2.76923	2.76923	2.76923	2.76923
19	2.5	2.5	2.5	2.5	2.5	2.5	2.49999	2.5	2.5	2.49998
20	2.35294	2.35293	2.35294	2.35294	2.35294	2.35294	2.35294	2.35294	2.35294	2.35293
21	2.2	2.2	2.19997	2.2	2.2	2.19999	2.2	2.2	2.2	2.2
22	2.09524	2.09524	2.09524	2.09524	2.09522	2.09521	2.09523	2.09521	2.09513	2.09489
23	2	2	2	2	2	2	2	2	2	2

$n=23$	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
$d=1$	7.5076e+06	6.69102e+06	5.54639e+06	4.19447e+06	2.84224e+06	1.69818e+06	880970	390656	145499	44552	10903	2048	277	24	1
2	3.91384e+06	3.59393e+06	3.09645e+06	2.44987e+06	1.74442e+06	1.09781e+06	600370	280600	110056	35443	9109	1794	254	23	1
3	320984	285238	251782	197621	137846	83846.2	43770	19320	7096	2126.2	507	92	12	1	1
4	165991	158067	136991	118511	87337.5	56251.3	31088.4	14505.3	5619.67	1772	443.75	84.3333	11.5	1	1
5	23598.1	22319.8	19953.3	16188.6	12464.1	9622	4808	2025	709.4	202.4	46	8	1	1	1
6	13591.8	13033.3	11943.5	9913.04	7987.06	6016.42	3606.25	1603.33	591.333	177.1	42.1667	7.66667	1	1	1
7	3671.74	3343.98	3336	2378.38	1641.17	1161.59	760	303.6	101.2	27.6	6	1	1	1	1
8	1936.53	1837.12	1799.26	1541	1112.34	831.855	507	253	88.55	25.3	5.75	1	1	1	1
9	278.976	273.331	257.262	224.353	172.042	119.059	77.449	30.3297	10	4.8	1	1	1	1	1
10	151.853	150.864	146.157	136.783	118.507	90.3571	63.25	25.8163	8.84615	4.6	1	1	1	1	1
11	47.2641	47.002	46.999	37.6364	27.6	16	9	5.76	4	1	1	1	1	1	1
12	23.9988	23.9998	23.9997	22.9991	17.25	11.5	7.66666	5.30769	3.83333	1	1	1	1	1	1
13	7	7	6.99999	6.72	6	5.09091	4.2	3.42857	1	1	1	1	1	1	1
14	5.60002	5.6	5.6	5.55172	5.19355	4.6	3.92683	3.28571	1	1	1	1	1	1	1
15	3.99999	3.99999	4	3.91837	3.69231	3.36842	3	1	1	1	1	1	1	1	1
16	3.55555	3.55556	3.55528	3.53846	3.40741	3.17241	2.875	1	1	1	1	1	1	1	1
17	3	3	3	2.9589	2.8421	2.66667	1	1	1	1	1	1	1	1	1
18	2.76922	2.76923	2.76737	2.76	2.68831	2.55556	1	1	1	1	1	1	1	1	1
19	2.5	2.5	2.5	2.47423	2.4	1	1	1	1	1	1	1	1	1	1
20	2.35292	2.3526	2.34995	2.34694	2.3	1	1	1	1	1	1	1	1	1	1
21	2.19999	2.2	2.2	2.18182	1	1	1	1	1	1	1	1	1	1	1
22	2.09427	2.09308	2.0916	2.09091	1	1	1	1	1	1	1	1	1	1	1
23	2	2	2	1	1	1	1	1	1	1	1	1	1	1	1

$n=24$	Delsarte	$D=0$	1	2	3	4	5	6	7	8	9
$d=1$	1.67772e+07	1.6778e+07	1.6777e+07	1.67766e+07	1.67798e+07	1.67741e+07	1.67643e+07	1.67215e+07	1.65853e+07	1.62454e+07	1.55049e+07
2	8.38861e+06	8.38704e+06	8.38788e+06	8.38808e+06	8.38745e+06	8.38745e+06	8.3855e+06	8.37675e+06	8.34293e+06	8.24154e+06	7.99684e+06
3	645278	645268	645261	645256	645263	645245	645058	644678	641035	636390	611068
4	349525	349510	349486	349506	349501	349508	349405	349311	347634	346298	334087
5	48148.9	48147.4	48147.6	48147.8	48148.1	48147.9	48146.9	48138.8	48096.2	47891.9	47101.5
6	24107.9	24105.9	24106.8	24106.6	24105.8	24105.5	24105.4	24104.2	24099.7	24070.8	23958.2
7	6474.52	6474.4	6474.43	6474.44	6474.46	6474.4	6474.23	6473.31	6469.38	6439.81	6264.15
8	4096	4095.21	4095.39	4095.78	4095.7	4095.85	4094.01	4094.24	4094.28	4093.38	3905.57
9	574.002	551.286	551.336	551.345	551.269	551.319	551.331	551.336	551.219	551.059	549.533
10	280	279.966	279.958	279.917	279.941	279.964	279.977	279.951	279.938	279.919	279.932
11	75.1304	75.1249	75.1241	75.1236	75.1203	75.1273	75.1247	75.1264	75.1238	75.1229	75.1073
12	48	47.9912	47.9937	47.995	47.989	47.9913	47.9928	47.9952	47.9975	47.9879	47.9799
13	9.33333	9.33278	9.33319	9.33326	9.33315	9.33326	9.33316	9.33292	9.3331	9.33278	9.33233
14	7	7	7.00001	6.99951	6.9994	6.9999	7.00018	6.99979	6.99994	6.99937	6.99968
15	4.57143	4.57142	4.5714	4.57141	4.57142	4.57142	4.57128	4.57129	4.57131	4.57142	4.57147
16	7.03854	4	3.99946	4	3.99999	3.99999	4	4	4	3.99999	4
17	3.27273	3.27272	3.27273	3.27273	3.27272	3.27272	3.27194	3.27273	3.27273	3.27272	3.27269
18	3	3	3	3	3	2.99999	3	3	2.99998	3	2.99993
19	2.66667	2.66667	2.66667	2.66666	2.66667	2.66665	2.66666	2.66667	2.66666	2.66672	2.66666
20	2.5	2.5	2.5	2.49999	2.5	2.5	2.5	2.50001	2.5	2.5	2.5
21	2.5569	2.31541	2.31579	2.31552	2.31579	2.31578	2.31578	2.31578	2.31578	2.31579	2.31574
22	2.2	2.2	2.2	2.2	2.2	2.19999	2.2	2.2	2.2	2.2	2.2
23	2.09508	2.08695	2.08695	2.08694	2.08691	2.08695	2.08695	2.08692	2.08684	2.0866	2.08604
24	2	2	2	1.9998	1.99998	1.99999	2	2	2	2	2

$n=24$	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
$d=1$	1.41981e+07	1.22368e+07	9.74008e+06	7.03646e+06	4.54035e+06	2.57916e+06	1.27163e+06	536155	190051	55455	12951	2325	301	25	1
2	7.50646e+06	6.68936e+06	5.54597e+06	4.19417e+06	2.8424e+06	1.69818e+06	880968	390656	145499	44552	10903	2048	277	24	1
3	577852	501543	432095	322554	213173	122706	60663.9	25400	8867.5	2531	576	100	12.5	1	1
4	320961	285176	251770	197612	137846	83845.9	43770	19320	7096	2126.2	507	92	12	1	1
5	44967.1	40691.7	33762.2	26961.3	19633.9	13361.1	6326	2531	844.333	230	50	8.33333	1	1	1
6	23592.1	22316.6	19952.8	16188.6	12463.8	9622	4808	2025	709.4	202.4	46	8	1	1	1
7	6013.55	5531.57	4968.89	3626.23	2477.53	1627.42	949.749	361.429	115	30	6.25	1	1	1	1
8	3670.88	3343.38	3335.92	2378.31	1641.15	1161.59	760	303.6	101.2	27.6	6	1	1	1	1
9	534.983	517.05	466.428	389.706	263.857	158.465	94.7801	35.7143	11.3636	5	1	1	1	1	1
10	278.981	273.266	257.226	224.34	172.036	119.059	77.449	30.3297	10	4.8	1	1	1	1	1
11	74.3466	72.2464	71.1206	62.7267	39.9997	25	10.7143	6.25	4.16667	1	1	1	1	1	1
12	47.2564	46.9949	46.9939	37.6359	27.6	16	9	5.76	4	1	1	1	1	1	1
13	9.33286	9.33262	9.21025	8.33329	7	5.64516	4.48718	3.57143	1	1	1	1	1	1	1
14	6.99989	6.99977	6.99953	6.71999	6	5.09091	4.2	3.42857	1	1	1	1	1	1	1
15	4.57143	4.57142	4.54545	4.34783	4	3.57143	3.125	1	1	1	1	1	1	1	1
16	3.99991	3.99998	4	3.91836	3.69231	3.36842	3	1	1	1	1	1	1	1	1
17	3.27273	3.27213	3.26087	3.16901	3	2.77778	1	1	1	1	1	1	1	1	1
18	2.99998	3	3	2.9589	2.84211	2.66667	1	1	1	1	1	1	1	1	1
19	2.66663	2.66484	2.65957	2.60417	2.5	1	1	1	1	1	1	1	1	1	1
20	2.5	2.5	2.5	2.47423	2.4	1	1	1	1	1	1	1	1	1	1
21	2.3154	2.31318	2.31092	2.27268	1	1	1	1	1	1	1	1	1	1	1
22	2.2	2.19996	2.19999	2.18182	1	1	1	1	1	1	1	1	1	1	1
23	2.08504	2.08386	2.08333	1	1	1	1	1	1	1	1	1	1	1	1
24	2	2	2	1	1	1	1	1	1	1	1	1	1	1	1

$n = 25$	Delsarte	$D = 0$	1	2	3	4	5	6
$d = 1$	3.35544e+07	3.35506e+07	3.35514e+07	3.35475e+07	3.35432e+07	3.35437e+07	3.35286e+07	3.34784e+07
2	1.67772e+07	1.67705e+07	1.67661e+07	1.67713e+07	1.67739e+07	1.6764e+07	1.67597e+07	1.67565e+07
3	1.19837e+06	1.19834e+06	1.19819e+06	1.19823e+06	1.1983e+06	1.19833e+06	1.19817e+06	1.19748e+06
4	645278	645050	645032	645066	645067	645168	645015	644737
5	95325.1	93614.2	93612.6	93622.9	93616.9	93618.2	93618	93600.5
6	48148.9	48133.4	48134.5	48138.8	48131.5	48135.6	48137.8	48127.4
7	12125.3	10433.8	10433.6	10434.3	10433.9	10433.8	10434.2	10434.1
8	6474.52	6473.43	6473.35	6472.72	6472.6	6472.82	6473.45	6472.78
9	1040.21	1040.02	1040.04	1039.95	1039.86	1039.95	1039.81	1039.95
10	555.764	551.156	550.972	551.013	551.151	550.844	551.064	551
11	113.311	113.297	113.282	113.287	113.295	113.272	113.284	113.296
12	83.488	75.1059	75.0955	75.1182	75.1081	75.1036	75.1081	75.112
13	14	13.9923	13.999	13.9899	13.9969	13.9993	13.995	13.9964
14	9.33333	9.32988	9.33243	9.3297	9.33307	9.33296	9.3328	9.33232
15	5.33333	5.33328	5.33327	5.33298	5.33283	5.33326	5.33255	5.33324
16	8.93398	4.57137	4.57135	4.57102	4.57106	4.57083	4.57071	4.57115
17	3.6	3.59994	3.59981	3.59985	3.59994	3.59979	3.59992	3.59991
18	5.2647	3.27241	3.27269	3.27269	3.27266	3.2727	3.27269	3.27263
19	2.85714	2.85701	2.85711	2.85714	2.85697	2.85713	2.85711	2.857
20	2.66667	2.66661	2.66659	2.66653	2.66652	2.66648	2.6666	2.66659
21	2.44444	2.44443	2.44299	2.44439	2.44443	2.44442	2.44439	2.44438
22	2.31579	2.31578	2.31576	2.31568	2.31574	2.31568	2.31574	2.31575
23	2.38016	2.1818	2.18178	2.18181	2.18178	2.18178	2.18178	2.18176
24	2.08696	2.08689	2.08671	2.0869	2.08688	2.08692	2.08693	2.08687
25	2	1.99994	2	1.99999	2	1.99999	2	1.99994

$n = 25$	7	8	9	10	11	12	13	14
$d = 1$	3.33085e+07	3.28183e+07	3.17379e+07	2.9701e+07	2.64188e+07	2.1972e+07	1.67776e+07	1.15777e+07
2	1.67079e+07	1.65702e+07	1.62341e+07	1.54984e+07	1.41864e+07	1.2233e+07	9.73873e+06	7.03528e+06
3	1.19556e+06	1.18614e+06	1.16954e+06	1.11266e+06	1.0235e+06	871393	726548	515869
4	644269	640896	636229	610826	577579	501423	432085	322499
5	93523.7	93254.7	92009.2	89802.7	83287.1	70856.7	57430	42339.7
6	48116.2	48078.6	47859.6	47070.9	44940.8	40681.5	33761.4	26961.7
7	10432.2	10424.1	10353.5	10162.4	9715.14	8810.07	7640.75	5575
8	6471.76	6467.22	6436.3	6260.86	6010.29	5529.8	4968.78	3626.52
9	1039.74	1039.56	1037.32	1030.32	986.224	925.387	842.664	622.188
10	550.887	550.793	550.383	549.259	534.654	516.764	466.426	389.621
11	113.296	113.273	113.24	113.096	111.652	107.876	102.725	85.8701
12	75.0803	75.0975	75.0932	75.0742	74.3437	72.2066	71.1167	62.7176
13	13.9896	13.9967	13.9909	13.9953	13.9986	13.9983	12.9993	10.7049
14	9.32943	9.33262	9.33297	9.33224	9.32937	9.3306	9.20885	8.33307
15	5.33318	5.33289	5.33282	5.33295	5.33289	5.33305	5.19986	4.83716
16	4.57128	4.57136	4.5713	4.57123	4.57134	4.57134	4.54542	4.34778
17	3.59997	3.5999	3.59994	3.59982	3.59971	3.59987	3.54543	3.39129
18	3.2726	3.27267	3.27245	3.27251	3.27248	3.27176	3.26079	3.169
19	2.85703	2.85702	2.85698	2.85688	2.85698	2.85712	2.82607	2.73681
20	2.66682	2.66661	2.66663	2.66654	2.66653	2.66467	2.65936	2.60416
21	2.44431	2.44439	2.44436	2.44438	2.44431	2.4444	2.42371	2.36364
22	2.31571	2.31567	2.31572	2.31549	2.31521	2.31297	2.31092	2.27273
23	2.18179	2.18178	2.18178	2.18173	2.18175	2.18174	2.1666	1
24	2.08686	2.0867	2.08644	2.08596	2.08479	2.08385	2.08333	1
25	1.99995	1.99997	1.99997	1.99998	2	1.99998	1	1

$n = 25$	15	16	17	18	19	20	21	22	23	24	25
$d = 1$	7.11996e+06	3.85087e+06	1.80778e+06	726206	245506	68406	15276	2626	326	26	1
2	4.54019e+06	2.57914e+06	1.27162e+06	536155	190051	55455	12951	2325	301	25	1
3	323437	176578	82876.9	32998.3	10976.3	2991	651	108.333	13	1	1
4	213155	122707	60664.4	25400	8867.5	2531	576	100	12.5	1	1
5	31116.6	18280.4	8223.49	3133.38	997.667	260	54.1667	8.66667	1	1	1
6	19633.8	13361.1	6326.17	2531	844.333	230	50	8.33333	1	1	1
7	3735.54	2283.99	1175.64	427.143	130	32.5	6.5	1	1	1	1
8	2477.52	1627.42	949.747	361.429	115	30	6.25	1	1	1	1
9	413.006	214.619	116.071	42.2078	13	5.2	1	1	1	1	1
10	263.853	158.464	94.7802	35.7143	11.3636	5	1	1	1	1	1
11	64.9917	30.9524	13	6.78261	4.33333	1	1	1	1	1	1
12	39.9983	25	10.7143	6.25	4.16667	1	1	1	1	1	1
13	8.27253	6.27577	4.78947	3.71429	1	1	1	1	1	1	1
14	6.99992	5.64516	4.48718	3.57143	1	1	1	1	1	1	1
15	4.33331	3.78176	3.25	1	1	1	1	1	1	1	1
16	4	3.57143	3.125	1	1	1	1	1	1	1	1
17	3.16216	2.88889	1	1	1	1	1	1	1	1	1
18	3	2.77778	1	1	1	1	1	1	1	1	1
19	2.6	1	1	1	1	1	1	1	1	1	1
20	2.5	1	1	1	1	1	1	1	1	1	1
21	1	1	1	1	1	1	1	1	1	1	1
22	1	1	1	1	1	1	1	1	1	1	1
23	1	1	1	1	1	1	1	1	1	1	1
24	1	1	1	1	1	1	1	1	1	1	1
25	1	1	1	1	1	1	1	1	1	1	1

$n = 26$	Delsarte	$D = 0$	1	2	3	4	5	6
$d = 1$	6.71089e+07	6.7055e+07	6.69472e+07	6.68482e+07	6.69253e+07	6.69136e+07	6.70757e+07	6.68709e+07
2	3.35544e+07	3.34986e+07	3.34581e+07	3.34601e+07	3.34908e+07	3.34833e+07	3.34603e+07	3.34741e+07
3	2.39675e+06	2.39563e+06	2.39574e+06	2.39487e+06	2.39587e+06	2.39486e+06	2.39599e+06	2.3926e+06
4	1.19837e+06	1.19688e+06	1.19688e+06	1.19695e+06	1.19701e+06	1.19763e+06	1.1966e+06	1.19697e+06
5	163840	163764	163765	163767	163752	163770	163765	163798
6	95325.1	93554	93510.1	93481.7	93503.8	93507.2	93558.8	93525.6
7	18189.6	18180.5	18176.5	18174.7	18176.1	18183.2	18183.8	18157.7
8	10435	10428.5	10426.5	10424.7	10426.8	10427.7	10424.9	10422.6
9	1765.52	1744.83	1764.36	1763.23	1762.66	1764.17	1763.1	1763.96
10	1040.21	1038.96	1038.65	1038.87	1039.08	1038.46	1038.93	1039.44
11	170.667	170.515	170.437	170.424	170.503	170.516	170.549	170.451
12	113.311	113.228	113.169	113.202	113.147	113.136	113.223	113.144
13	28	27.877	27.9603	27.8661	27.8504	27.9366	27.8816	27.9514
14	16.9836	13.9794	13.9963	13.9898	14.5143	13.9722	13.9913	13.9785
15	6.4	6.39952	6.39631	6.3995	6.39819	6.39955	6.3994	6.39313
16	11.3932	5.33309	5.32938	5.33153	5.33145	5.33278	5.33289	5.33293
17	4	3.99913	3.9996	3.99914	3.99986	3.99864	3.99969	3.99915
18	3.6	3.59985	3.59908	3.59982	3.59988	3.5996	3.59974	3.59973
19	3.07692	3.07611	3.07589	3.07679	3.0762	3.07686	3.07631	3.07677
20	4.75835	2.85705	2.85642	2.85686	2.85697	2.85634	2.85682	2.85701
21	3.76728	2.58792	2.58802	2.58736	2.58777	2.58798	2.58808	2.58782
22	2.44444	2.44406	2.44395	2.44414	2.4441	2.4439	2.44429	2.44426
23	2.28571	2.28547	2.28555	2.28552	2.28556	2.28549	2.28557	2.28553
24	2.18182	2.18147	2.18149	2.18153	2.18168	2.1815	2.18168	2.18154
25	2.08	2.07974	2.07979	2.07982	2.07982	2.07989	2.07964	2.07963
26	2	1.99996	1.99994	1.99985	1.99983	1.99978	1.99978	1.99977

$n = 26$	7	8	9	10	11	12	13	14
$d = 1$	6.66593e+07	6.60149e+07	6.45636e+07	6.14342e+07	5.59878e+07	4.83217e+07	3.87258e+07	2.83448e+07
2	3.33657e+07	3.32728e+07	3.27702e+07	3.16547e+07	2.96601e+07	2.64099e+07	2.19429e+07	1.67544e+07
3	2.39143e+06	2.36751e+06	2.35058e+06	2.2392e+06	2.10737e+06	1.83874e+06	1.58588e+06	1.19768e+06
4	1.19542e+06	1.19422e+06	1.18327e+06	1.16713e+06	1.10985e+06	1.02289e+06	869832	726599
5	163719	163529	162722	159928	152983	140552	116608	94578.9
6	93485.6	93424.4	93134.4	91829.4	89693.8	83192.8	70737.3	57423.6
7	18179.4	18157.2	18068.1	17728.5	17126	16346.8	13969.5	11722.8
8	10424.4	10413.6	10413.2	10342.3	10143.8	9701.19	8801.33	7636.26
9	1762.86	1762.59	1761.97	1753.64	1727.51	1635.93	1485.21	1306.63
10	1037.55	1037.48	1037.5	1035.39	1027.88	983.194	924.409	841.477
11	170.385	170.44	170.386	170.345	169.367	163.314	157.763	141.186
12	113.099	113.26	113.106	113.148	112.902	111.688	107.763	102.702
13	27.994	27.9478	27.8801	27.9161	27.8823	27.9486	26.9932	20.9733
14	13.9706	13.9662	13.9945	13.9613	13.9692	13.9714	13.9735	12.9991
15	6.39815	6.39851	6.39815	6.39766	6.3975	6.39384	6.3526	5.99928
16	5.33079	5.33261	5.33271	5.33244	5.33164	5.33118	5.33285	5.19963
17	3.99983	3.99943	3.99935	3.9994	3.99805	3.99836	3.98218	3.85701
18	3.59964	3.59976	3.5997	3.59942	3.59953	3.59925	3.59942	3.54514
19	3.07658	3.07635	3.07658	3.07636	3.07577	3.0759	3.06792	2.99946
20	2.857	2.85674	2.85684	2.85684	2.85657	2.85697	2.85684	2.82593
21	2.58809	2.58783	2.58753	2.58803	2.58773	2.58642	2.58193	2.53844
22	2.44416	2.44425	2.44379	2.44389	2.44433	2.44418	2.44426	2.42362
23	2.28556	2.28539	2.28524	2.28485	2.28435	2.28306	2.28152	2.24997
24	2.18166	2.18121	2.18134	2.18171	2.18164	2.18127	2.18166	2.16666
25	2.07981	2.07972	2.07951	2.07899	2.0776	2.07726	2.07681	1
26	1.99977	1.9998	1.99981	1.99978	1.99998	1.99999	2	1

$n = 26$	15	16	17	18	19	20	21	22	23	24	25	26
$d = 1$	1.87024e+07	1.09719e+07	5.65872e+06	2.53401e+06	971712	313912	83682	17902	2952	352	27	1
2	1.1575e+07	7.1188e+06	3.8505e+06	1.80777e+06	726206	245506	68406	15276	2626	326	26	1
3	809160	482120	250184	111736	42404.1	13468.5	3511	732.25	117	13.5	1	1
4	515766	323408	176573	82877.4	32998.3	10976.3	2991	651	108.333	13	1	1
5	66353.6	49390.5	24675.6	10572.8	3845.28	1171	292.5	58.5	9	1	1	1
6	42325.8	31113.6	18280.3	8223.44	3133.39	997.667	260	54.1667	8.66667	1	1	1
7	8718.27	5262.34	3204.51	1442.6	501.429	146.25	35.1	6.75	1	1	1	1
8	5576.2	3735.04	2283.98	1175.64	427.143	130	32.5	6.5	1	1	1	1
9	1012.06	580.468	299.339	142.451	50.1429	15	5.4	1	1	1	1	1
10	622.189	412.994	214.619	116.071	42.2078	13	5.2	1	1	1	1	1
11	114.171	83.5589	38.9998	16.1999	7.36364	4.5	1	1	1	1	1	1
12	85.8045	64.9855	30.9521	13	6.78261	4.33333	1	1	1	1	1	1
13	14.5358	9.94714	7	5.10811	3.85714	1	1	1	1	1	1	1
14	10.7054	8.27264	6.27586	4.78947	3.71429	1	1	1	1	1	1	1
15	5.39976	4.69565	4	3.375	1	1	1	1	1	1	1	1
16	4.83717	4.33323	3.78182	3.25	1	1	1	1	1	1	1	1
17	3.6268	3.32876	3	1	1	1	1	1	1	1	1	1
18	3.39113	3.16215	2.88889	1	1	1	1	1	1	1	1	1
19	2.87233	2.69999	1	1	1	1	1	1	1	1	1	1
20	2.73681	2.6	1	1	1	1	1	1	1	1	1	1
21	2.45446	1	1	1	1	1	1	1	1	1	1	1
22	2.36363	1	1	1	1	1	1	1	1	1	1	1
23	1	1	1	1	1	1	1	1	1	1	1	1
24	1	1	1	1	1	1	1	1	1	1	1	1
25	1	1	1	1	1	1	1	1	1	1	1	1
26	1	1	1	1	1	1	1	1	1	1	1	1

Bibliography

[1] Shashi Borade, Baris Nakiboglu, and Lizhong Zheng. "Unequal error protection: An information-theoretic perspective". In: *IEEE Transactions on Information Theory* 55.12 (2009), pp. 5511–5539.

[2] Shashi Borade and Sujay Sanghavi. "Some fundamental coding theoretic limits of unequal error protection." In: *ISIT*. 2009, pp. 2231–2235.

[3] Brian Borchers. *CSDP, A C Library for Semidefinite Programming*. URL: `https://projects.coin-or.org/Csdp/`.

[4] Tullio Ceccherini-Silberstein, Fabio Scarabotti, and Filippo Tolli. *Harmonic analysis on finite groups: representation theory, Gelfand pairs and Markov chains*. Vol. 108. Cambridge University Press, 2008.

[5] Charles F Dunkl. "A Krawtchouk polynomial addition theorem and wreath products of symmetric groups". In: *Indiana University Mathematics Journal* 25.4 (1976), pp. 335–358.

[6] Charles F Dunkl. "Spherical functions on compact groups and applications to special functions". In: *Symposia Mathematica* 22 (1976), pp. 145–161.

[7] S. Karlin and J.L. McGregor. *The Hahn Polynomials, Formulas and an Application*. 1960.

[8] László Lovász. "On the Shannon capacity of a graph". In: *IEEE Transactions on Information theory* 25.1 (1979), pp. 1–7.

[9] Robert McEliece et al. "New upper bounds on the rate of a code via the Delsarte-MacWilliams inequalities". In: *IEEE Transactions on Information Theory* 23.2 (1977), pp. 157–166.

[10] Bruce E Sagan. *The symmetric group: representations, combinatorial algorithms, and symmetric functions*. Vol. 203. Springer Science & Business Media, 2013.

[11] Alexander Schrijver. "A comparison of the Delsarte and Lovász bounds". In: *IEEE Transactions on Information Theory* 25.4 (1979), pp. 425–429.

[12] Alexander Schrijver. "New code upper bounds from the Terwilliger algebra and semidefinite programming". In: *IEEE Transactions on Information Theory* 51.8 (2005), pp. 2859–2866.

[13] Frank Vallentin. "Lecture notes: Semidefinite programs and harmonic analysis". In: *arXiv preprint arXiv:0809.2017* (2008).

[14] Frank Vallentin. "Symmetry in semidefinite programs". In: *Linear Algebra and its Applications* 430.1 (2009), pp. 360–369.

Hilfsmittel

- Brian Borcher's CSDP [3]
- Visual Studio 2017
- IPE
- Ti*k*Z
- Texmaker

Versicherung

Hiermit versichere ich an Eides statt, dass ich die vorliegende Arbeit selbstständig und ohne die Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe. Alle Stellen, die wörtlich oder sinngemäß aus veröffentlichten und nicht veröffentlichten Schriften entnommen wurden, sind als solche kenntlich gemacht. Die Arbeit ist in gleicher oder ähnlicher Form oder auszugsweise im Rahmen einer anderen Prüfung noch nicht vorgelegt worden. Ich versichere, dass die eingereichte elektronische Fassung der eingereichten Druckfassung vollständig entspricht.