A Complete Linear Programming Hierarchy for Linear Codes

Leonardo Nagami Coregliano* Fernando Granha Jeronimo† Chris Jones ‡

February 13, 2022

Abstract

A longstanding open problem in coding theory is to determine the best (asymptotic) rate $R_2(\delta)$ of binary codes with minimum constant (relative) distance δ. An existential lower bound was given by Gilbert and Varshamov in the 1950s. On the impossibility side, in the 1970s McEliece, Rodemich, Rumsey and Welch (MRRW) proved an upper bound by analyzing Delsarte's linear programs. To date these results remain the best known lower and upper bounds on $R_2(\delta)$ with no improvement even for the important class of *linear* codes. Asymptotically, these bounds differ by an exponential factor in the blocklength.

In this work, we introduce a new hierarchy of linear programs (LPs) that converges to the true size $A_2^{\mathrm{Lin}}(n, d)$ of an optimum *linear* binary code (in fact, over any finite field) of a given blocklength n and distance d. This hierarchy has several notable features:

i. It is a natural generalization of the Delsarte LPs used in the first MRRW bound.

ii. It is a hierarchy of linear programs rather than semi-definite programs potentially making it more amenable to theoretical analysis.

iii. It is *complete* in the sense that the optimum code size can be retrieved from level $O(n^2)$.

iv. It provides an answer in the form of a hierarchy (in larger dimensional spaces) to the question of how to cut Delsarte's LP polytopes to approximate the true size of *linear* codes.

We obtain our hierarchy by generalizing the Krawtchouk polynomials and MacWilliams inequalities to a suitable "higher-order" version taking into account interactions of ℓ words. Our method also generalizes to translation schemes under mild assumptions.

*IAS. lenacore@ias.edu. This material is based upon work supported by the National Science Foundation, and by the IAS School of Mathematics.

†IAS. granha@ias.edu. This material is based upon work supported by the National Science Foundation under Grant No. CCF-1900460.

‡UChicago. csj@uchicago.edu. This material is based upon work supported by the National Science Foundation under Grant No. CCF-2008920.

Contents

1 Introduction

A fundamental question in coding theory is the maximum size of a binary code given a blocklength parameter n and a minimum distance parameter d_n. This value is typically denoted by $A_2(n, d_n)$. A particularly important regime occurs when $\lim_{n\to\infty} d_n/n = \delta$ for some absolute constant $\delta \in (0, 1/2)$. In this regime, $A_2(n, d_n)$ is known to grow exponentially in n. However, the precise rate of this exponential growth remains an elusive major open problem. It is often convenient to denote the asymptotic basis of this growth as $2^{R_2(\delta)}$, where the rate $R_2(\delta)$ is defined as

$$R_2(\delta) \coloneqq \limsup_{n\to\infty} \frac{1}{n} \log_2 \left(A_2(n, \lfloor \delta n \rfloor)\right).$$

An equivalent way of defining $A_2(n, d)$ is as the independence number of the graph $H_{n,d}$ whose vertex set is $V(H_{n,d}) \coloneqq \mathbb{F}_2^n$ and two vertices $x, y \in V(H_{n,d})$ are adjacent if and only if their Hamming distance $\Delta(x, y)$ lies in $\{1, \ldots, d-1\}$. Note that there is a one-to-one correspondence between independent sets in this graph and binary codes of blocklength n and minimum distance d. Most of the literature about $A_2(n, d)$ takes advantage of this graph-theoretic interpretation.

A lower bound on $A_2(n, d)$ follows from the trivial degree bound on the independence number of a graph, namely, $\alpha(H_{n,d}) \geq |V(H_{n,d})|/(\deg(H_{n,d}) + 1)$, which gives $\alpha(H_{n,d}) \geq 2^{(1-h_2(d/n)+o(1))n}$ where h_2 is the binary entropy function. First discovered by Gilbert [Gil52] and later generalized to linear codes by Varshamov [Var57], this existential bound is now known as the Gilbert–Varshamov (GV) bound. Observe that the GV bound readily implies that $R_2(\delta) \geq 1 - h_2(\delta)$. Despite its simplicity, this bound remains the best (existential) lower bound on $R_2(\delta)$.

The techniques to upper bound $A_2(n, d)$ are oftentimes more involved, with the most prominent being the Delsarte linear programming method that we now describe. A binary code $C \subseteq \mathbb{F}_2^n$ is *linear* if it is a subspace of $\mathbb{F}_2^n$ and its weight distribution is the tuple $(a_0, a_1, \ldots, a_n) \in \mathbb{N}^{n+1}$, where a_i is the number of codewords of C of Hamming weight i. MacWilliams [Mac63] showed that the weight distribution $(b_0, b_1, \ldots, b_n)$ of the dual code $C^\perp$ can be obtained by applying a linear transformation to $(a_0, a_1, \ldots, a_n)$. More precisely, the MacWilliams identities establish that

$$b_j = \frac{1}{|C|} \sum_{i=0}^{n} K_j(i) \cdot a_i,$$

where the coefficients $K_j(i)$ are evaluations of the so-called Krawtchouk (or Kravchuk) polynomial of degree j. The Krawtchouk polynomials form a family of orthogonal polynomials under the measure $\mu_n(i) = \binom{n}{i}/2^n$ and they play an important role in coding theory [vL99, Chapter 1]. Since the weight distribution of the dual $C^\perp$ is non-negative, the MacWilliams identities can be relaxed to inequalities

$$\sum_{i=0}^{n} K_j(i) \cdot a_i \geq 0$$

for $j = 0, \ldots, n$. This naturally leads to the following linear program (LP) relaxation for $A_2(n, d)$ when C is a linear code (recall that for a linear code, having distance at least d is equivalent to having no words of Hamming weight 1 through $d-1$).

$$
\begin{aligned}
\max \quad & \sum_{i=0}^{n} a_i & & \\
\text{s.t.} \quad & a_0 = 1 & & \text{(Normalization)} \\
& a_i = 0 & \text{for } i = 1, \ldots, d-1 & \text{(Distance constraints)} \\
& \sum_{j=1}^{n} K_i(j) \cdot a_j \geqslant 0 & \text{for } i = 0, \ldots, n & \text{(MacWilliams inequalities)} \\
& a_i \geqslant 0 & \text{for } i = 0, \ldots, n & \text{(Non-negativity).}
\end{aligned}
$$

Figure 1: Delsarte's linear program for $A_2(n, d)$.

The a_i can be suitably generalized to codes which are not necessarily linear (by setting $a_i := \left|\{(x, y) \in C^2 | \Delta(x, y) = i\}\right| / |C|$). The MacWilliams inequalities hold for these generalized a_i's as proven by MacWilliams, Sloane and Goethals [MSG72]. Therefore, the same linear program above also bounds $A_2(n, d)$ for general codes. This family of linear programs was first introduced by Delsarte in [Del73], where it was obtained in greater generality from the theory of association schemes. We refer to the above linear program as Delsarte's linear program, or, more formally, as DelsarteLP(n, d).

The best known upper bound on $R_2(\delta)$ for distances $\delta \in (0.273, 1/2)$ is obtained by constructing solutions to the dual program of Delsarte's linear program, as first done by McEliece, Rodemich, Rumsey and Welch (MRRW) [MRRW77] in their first linear programming bound. In the same work, McEliece et al. also gave the best known bound for $\delta \in (0, 0.273)$ via a second family of linear programs. Since our techniques are more similar to their first linear programming bound, we restrict our attention to it in this discussion. In the first linear programming bound, they showed that $R_2(\delta) \leq h_2(1/2 - \sqrt{\delta(1-\delta)})$ with a reasonably sophisticated argument using properties of general orthogonal polynomials and also particular properties of Krawtchouk polynomials. Simpler perspectives on the first LP bound analysis were found by Navon and Samorodnitsky [NS05] and by Samorodnitsky [Sam21].

Instead of linear programming, one can use more powerful techniques based on semi-definite programming (SDP) to upper bound $A_2(n, d)$. For instance, the Sum-of-Squares/Lasserre SDP hierarchy was suggested for this problem by Laurent [Lau09]. The value of the program equals $\alpha(H_{n,d})$ for a sufficiently high level of the hierarchy, so in principle analyzing these programs could give $A_2(n, d)$ exactly. Analyzing SDP methods to improve $R_2(\delta)$ seems challenging and we do not even know how to analyze the simplest of them [Sch05], which is weaker than degree-4 Sum-of-Squares (see related work below for more details on SDP methods).

On the one hand, we have reasonably simple linear programs of Delsarte already requiring a non-trivial theoretical analysis for proving upper bounds on $R_2(\delta)$. On the other hand we have more sophisticated SDP methods which are provably stronger than the Delsarte LP, but for which no theoretical analyses are known.

1.1 Our Contribution

In this work, we refine the Delsarte linear programming method used in the first LP bound for $A_2(n,d)$ by designing a *hierarchy* of linear programs. For a parameter $\ell \in \mathbb{N}_+$, the hierarchy is based on specific higher-order versions of Krawtchouk polynomials and MacWilliams inequalities that take advantage of ℓ-point interactions of words. We denote by $\text{KrawtchoukLP}(n,d,\ell)$ the linear programming relaxation for $A_2(n,d)$ at level ℓ of our hierarchy.

We define $A_2^{\text{Lin}}(n,d)$ analogously to $A_2(n,d)$ as the maximum size of a *linear* binary code of blocklength n and minimum distance d. For linear codes, we impose additional "semantic" constraints on the programs in the hierarchy taking advantage of the linear structure of the code. We denote by $\text{KrawtchoukLP}_{\text{Lin}}(n,d,\ell)$ the linear program relaxation for $A_2^{\text{Lin}}(n,d)$ with these additional constraints. Both $\text{KrawtchoukLP}_{\text{Lin}}(n,d,1)$ and $\text{KrawtchoukLP}(n,d,1)$ coincide with $\text{DelsarteLP}(n,d)$ at the first level of our hierarchy.

There is a known gap between the value of Delsarte's linear programs and the GV bound. In particular when $\delta = 1/2 - \varepsilon$, Delsarte's linear programs do not yield an upper bound tighter than $R_2(1/2-\varepsilon) \leq \Theta(\varepsilon^2 \log(1/\varepsilon))$, as shown by Navon and Samorodnitsky [NS05], whereas the GV bound establishes a lower bound of $R_2(1/2-\varepsilon) \geq \Omega(\varepsilon^2)$. There are no known improvements to these bounds even for the important class of *linear* codes. If the GV bound is indeed tight, then analyzing DelsarteLP is not sufficient to prove it. The goal of our hierarchy is to give tighter and tighter upper bounds on $A_2(n,d)$ as the level of the hierarchy increases.

We show that for *linear* codes the hierarchy $\text{KrawtchoukLP}_{\text{Lin}}(n,d,\ell)$ is *complete*, meaning that the value of the hierarchy converges to $A_2^{\text{Lin}}(n,d)$ as ℓ grows larger. We prove that level roughly $\ell = O(n^2)$ is enough to retrieve the correct value of $A_2^{\text{Lin}}(n,d)$. More generally, for linear codes over $\mathbb{F}_q$, we have the following completeness theorem for $A_q^{\text{Lin}}(n,d)$.

Theorem 1.1 (Completeness - Informal version of Theorem 6.1)**.** *For $\ell \geq \Omega_{\varepsilon,q}(n^2)$, we have*

$$A_q^{\text{Lin}}(n,d) \;\leqslant\; \mathsf{val}(\text{KrawtchoukLP}_{\text{Lin}}^{\mathbb{F}_q}(n,d,\ell))^{1/\ell} \;\leqslant\; (1+\varepsilon)\cdot A_q^{\text{Lin}}(n,d).$$

We think that the $\text{KrawtchoukLP}_{\text{Lin}}(n,d,\ell)$ hierarchy is an extremely interesting object for the following reasons.

i. It is takes advantage of higher-order interactions of codewords by naturally computing Hamming weight statistics of subspaces spanned by ℓ codewords (see Definition 3.1).

ii. It is a generalization of the Delsarte LP used in the first MRRW bound and the two share strong structural similarities (see Section 3).

iii. It is a hierarchy of linear programs rather than semi-definite programs (see Definition 3.11 and Section 4.2).

iv. It is a *complete* hierarchy (see Theorem 6.1).

v. It provides an answer in the form of a hierarchy (in larger dimensional spaces) to the question of how to cut Delsarte's LP polytopes [NS05] to approximate the true size of *linear* codes.

We hope this hierarchy will fill an important gap in the coding theory literature between Delsarte's LP, for which theoretical analyses are known, and more powerful SDP methods, for which we seem to have no clue how to perform asymptotic analysis.

Not unexpectedly, the hierarchy KrawtchoukLP(n, d, ℓ) corresponding to general (not necessarily linear) codes does not improve on Delsarte's linear program. Without the extra structure of linearity, the number of constraints we can add to our LP hierarchy is limited. We prove that solutions of DelsarteLP(n, d) (easily) lift to solutions of KrawtchoukLP(n, d, ℓ) with the same value as follows.

Proposition 1.2 (Hierarchy Collapse - Informal version of Proposition 6.5)**.** *For $\ell \in \mathbb{N}$, we have*

$$\mathsf{val}(\text{KrawtchoukLP}(n, d, \ell))^{1/\ell} \;=\; \text{DelsarteLP}(n, d).$$

This contrast between the hierarchies KrawtchoukLP$_{\text{Lin}}(n, d, \ell)$ and KrawtchoukLP(n, d, ℓ) reinvigorates the question of whether the maximum sizes of general and linear codes are substantially different or not.

Though we give special attention to the binary case since it may be the most important one, we prove completeness and lifting results more generally in the language of association schemes. For example, a suitable modification of the linear programming hierarchy also converges to the maximum size of a D-code in the Hamming scheme over any finite field (see Remark 6.2); this in particular covers other types of codes that may be of interest such as ε-balanced codes.

More on Related Work

Although quantitatively the McEliece et al. [MRRW77] upper bound on $R_2(\delta)$ has not improved, our qualitative understanding of this upper bound is now substantially better. Friedman and Tillich [FT05] designed generalized Alon–Boppana theorems in order to bound the size of linear binary codes. Inspired by [FT05], Navon and Samorodnitsky [NS05, NS09] rederived the McEliece et al. bound on $R_2(\delta)$ for general codes using a more intuitive proof based on Fourier analysis. Despite a seemingly different language, the proof in [NS05] also yields feasible solutions to the dual of Delsarte's LP as in MRRW. More recently, Samorodnitsky [Sam21] gave yet a new interpretation of the McEliece et al. upper bound and conjectured interesting hypercontractivity inequalities towards improving the upper bound on $R_2(\delta)$.

Schrijver [Sch79] showed that the seemingly artificial Delsarte LP has the same value[1] as the Lovász ϑ' relaxation for $\alpha(H_{n,d})$, which is also essentially the degree-2 Sum-of-Squares/Lasserre relaxation of $\alpha(H_{n,d})$ (with additional non-negativity constraints on the entries of the matrix). Schrijver showed that this holds generally for commutative association schemes, a connection that allows us to also express KrawtchoukLP as ϑ' of a certain graph.

A line of work (similar in motivation to the current work) is to strengthen a convex relaxation of $A_2(n, d)$. In Delsarte's approach, only the distance between pairs of points is taken into account in the optimization. For this reason, Delsarte's approach is classified as a 2-point bound [Val19]. Nonetheless, there is no reason to restrict oneself to just 2-point interactions. Schrijver [Sch05] constructed a family of semi-definite programs (SDPs) for $A_2(n, d)$ designed to take into account

[1] In fact, by a symmetrization of the ϑ' SDP on $H_{n,d}$ using its graph automorphisms, one obtains DelsarteLP(n, d) exactly, see Section 4.

the 3-point interactions. Extending Schrijver's result to a 4-point interaction bound, Gijswijt, Mittelmann and Schrijver [GMS12] gave another tighter family of SDPs for $A_2(n,d)$ (they also give a description of their hierarchy for arbitrary ℓ). A complete SDP hierarchy for $\alpha(H_{n,d})$ is the Sum-of-Squares/Lasserre hierarchy, which was proposed for code upper bounds by Laurent [Lau07], building on de Klerk et al. [dKPS07].

Since the Sum-of-Squares hierarchy is guaranteed to find the correct value of $\alpha(H_{n,d})$ when the level is sufficiently high (precisely, level $2\alpha(H_{n,d})$), in principle it would be enough to analyze this SDP to compute $A_2(n,d)$. Unfortunately, studying the performance of SDPs on a fixed instance is a notoriously difficult task. In particular, the global positive semi-definiteness constraint is nontrivial. Unfortunately, no theoretical analysis is known for "genuine" SDP methods even for the simplest of them, the 3-point bound of Schrijver [Sch05] mentioned above.

In summary, the state of affairs on upper bounding $A_2(n,d)$ or $A_2^{\mathrm{Lin}}(n,d)$ is as follows. On one hand, we have a thorough theoretical understanding of techniques based on Delsarte's LP, but if the true value of $A_2(n,d)$ or $A_2^{\mathrm{Lin}}(n,d)$ is closer to the GV bound, then these techniques fall short of providing tight bounds. On the other hand, we have ℓ-point bounds from SDP techniques capable of yielding the correct value of $A_2(n,d)$, but (apparently) no clue how to theoretically analyze them to bound $R_2(\delta)$ for general codes or linear codes. We hope that our hierarchy will open a new angle of attack on this elusive problem for the important class of *linear* binary codes.

1.2 Outline of the Paper

Section 2 contains some notation and basic facts.

Section 3 shows the construction of the LP hierarchy for the binary code case. In this section, we introduce the notion of an *ℓ-configuration*, which roughly capture the Hamming weights of all words in the subspace spanned by the ℓ points. In analogy with the usual the Delsarte LP, we then analyze statistics of codes called *ℓ-configuration profiles*, which capture the number of ℓ-tuples in each possible ℓ-configuration. In the rest of the section we construct higher-order Krawtchouk polynomials, show MacWilliams identities, define the LP hierarchy and prove that its restrictions can be computed in $O(n^{2^{\ell+1}-2})$ time (for $\ell \in \mathbb{N}_+$ fixed).

Section 4 shows how the LP hierarchy admits several other interpretations. The LP hierarchy is a *symmetrization* of an exponential-size hierarchy, which has a natural interpretation either as checking non-negativity of Fourier coefficients of the code, or as $\vartheta'(G)$ for a large graph G.

In Section 5, we study our construction in more generality through the theory of *association schemes*. Our construction can be seen as adding extra constraints to the ℓ-fold tensor product of the Delsarte LP. More specifically, the underlying association scheme is a *refinement* of the ℓ-fold tensor product scheme in which "semantic" constraints can be added due to linearity of the code in the original translation scheme. We study this type of refinement, giving conditions under which it is still a bona fide translation scheme. The other sections may be read mostly independently of this section.

In Section 6, we show the main results: that the LP hierarchy is complete for linear codes, and no better than the Delsarte LP in the general (not necessarily linear) case.

We conclude in Section 7 with some open problems.

2 Preliminaries

A binary code C of block length n is a subset of $\mathbb{F}_2^n$. For a word $x \in \mathbb{F}_2^n$, we denote by $|x| := |\{i \in [n] \mid x_i \neq 0\}|$ its *Hamming weight*. Given two words $x, y \in \mathbb{F}_2^n$, we denote by $\Delta(x,y) := |x-y|$ their *Hamming distance*. The *(minimum) distance* of C is defined by $\Delta(C) := \min\{\Delta(x,y) \mid x,y \in C \wedge x \neq y\}$. The *rate* of C is defined by $r(C) := \log_2(|C|)/n$. The maximum size of a code of blocklength n and minimum distance at least d is defined as

$$A_2(n,d) := \max\{|C| \mid C \subseteq \mathbb{F}_2^n, \Delta(C) \geq d\}.$$

We denote the *asymptotic rate* of codes of relative distance at least δ and alphabet size q as

$$R_2(\delta) := \limsup_{n\to\infty} \frac{1}{n} \log_2 \left(A_2(n, \lfloor \delta n \rfloor)\right).$$

We define $A_2^{\mathrm{Lin}}(n,d)$ and $R_2^{\mathrm{Lin}}(\delta)$ for linear codes in an analogous way, by further requiring the code C to be *linear* (i.e., an $\mathbb{F}_2$-linear subspace of $\mathbb{F}_2^n$).

Note that a code of distance at least d can alternatively be viewed as an independent set in the *Hamming cube graph of distance less than d*, $H_{n,d}$, whose vertex set is $V(H_{n,d}) := \mathbb{F}_2^n$ and whose edge set is $E(H_{n,d}) := \{\{x,y\} \in \binom{\mathbb{F}_2^n}{2} \mid \Delta(x,y) \leqslant d-1\}$.

Let $f, g : \mathbb{F}_2^n \to \mathbb{R}$. We denote by $\langle f, g\rangle := \mathbb{E}_{x\in_R \mathbb{F}_2^n}[f(x)g(x)]$ the *inner product* of f and g under the uniform measure and we denote by $f * g$ their convolution given by $(f*g)(x) := \mathbb{E}_{y\in_R\mathbb{F}_2^n}[f(y)\cdot g(x-y)]$ $(x \in \mathbb{F}_2^n)$. The *Fourier transform* $\widehat{f}$ of f is given by $\widehat{f}(x) := \langle f, \chi_x\rangle = \mathbb{E}_{y\in_R\mathbb{F}_2^n}[f(y)\cdot\chi_x(y)]$, where $\chi_x(y) := (-1)^{\langle x,y\rangle}$. The (simple) Plancherel identity will be used in our computations.

Fact 2.1 (Plancherel)**.** *Let $f, g\colon \mathbb{F}_2^n \to \mathbb{R}$. Then $\langle f, g\rangle = \sum_{x\in\mathbb{F}_2^n} \widehat{f}(x)\cdot\widehat{g}(x)$.*

Given a linear code $C \subseteq \mathbb{F}_2^n$, the *dual code* of C is defined as $C^\perp := \{x \in \mathbb{F}_2^n \mid \forall y \in C, \chi_x(y) = 1\}$. The Fourier transform of the indicator of a linear code maps it to a multiple of the indicator of its dual code in the following way.

Fact 2.2. *If $C \subseteq \mathbb{F}_2^n$ is a linear code and $\mathbb{1}_C$ is its indicator function, then $\widehat{\mathbb{1}_C} = |C|\cdot\mathbb{1}_{C^\perp}/2^n = \mathbb{1}_{C^\perp}/|C^\perp|$.*

3 Krawtchouk Hierarchies for Binary Codes

In this section, we describe the LP hierarchy for the standard case of binary codes. We opt for an ad hoc derivation from boolean Fourier analysis to show how the higher-order Krawtchouk polynomials nicely parallel the usual Krawtchouk polynomials. Any omitted proofs in this section can be found in Appendix A. In Section 5, we will generalize the construction using the language of association schemes.

3.1 Higher-order Krawtchouk polynomials

As we alluded to previously, we want to incorporate ℓ-point interactions in our optimization problem for $A_2(n,d)$ similar in spirit to the Sum-of-Squares semi-definite programming hierarchy

for the independence number of a graph but in the simpler setting of *linear* programming. To accomplish this goal, we measure the profile of "configurations" of ℓ-tuples of codewords from the code.

We start with the definition of symmetric difference configurations. In plain English, the symmetric difference configuration of an ℓ-tuple $(z_1, \ldots, z_\ell) \in (\mathbb{F}_2^n)^\ell$ of words captures all information of $(z_1, \ldots, z_\ell)$ corresponding to Hamming weights of linear combinations of the words.

Definition 3.1. *The symmetric difference configuration of the ℓ-tuple $(z_1, \ldots, z_\ell) \in (\mathbb{F}_2^n)^\ell$ is the function $\mathsf{Config}^{\Delta}_{n,\ell}(z_1, \ldots, z_\ell) \colon 2^{[\ell]} \to \mathbb{R}$ defined by*

$$\mathsf{Config}^{\Delta}_{n,\ell}(z_1, \ldots, z_\ell)(J) := \left| \sum_{j \in J} z_j \right|,$$

for every $J \subseteq [\ell]$, that is, the value of the function at $J \subseteq [\ell]$ is the Hamming weight of the linear combination $\sum_{j \in J} z_j$.

By viewing $\mathsf{Config}^{\Delta}_{n,\ell}$ as a function $(\mathbb{F}_2^n)^\ell \to \mathbb{R}^{2^{[\ell]}}$ (i.e., a function from the space of ℓ-tuples of words to the space of functions $2^{[\ell]} \to \mathbb{R}$), the set of (valid) symmetric difference configurations of ℓ-tuples of elements of $\mathbb{F}_2^n$ is captured by its image $\mathrm{im}(\mathsf{Config}^{\Delta}_{n,\ell})$.

Given a symmetric difference configuration $g \in \mathrm{im}(\mathsf{Config}^{\Delta}_{n,\ell})$, we will also abuse notation and write $(z_1, \ldots, z_\ell) \in g$ to mean that $(z_1, \ldots, z_\ell) \in (\mathbb{F}_2^n)^\ell$ has configuration $\mathsf{Config}^{\Delta}_{n,\ell}(z_1, \ldots, z_\ell) = g$. In other words, this abuse of notation consists of thinking of a configuration as the set of all ℓ-tuples of words that have this configuration (see also Lemma 3.4 below). We also let $|g|$ be the size of this set, i.e., the number of ℓ-tuples whose configuration is g.

The trivial symmetric difference configuration is the constant 0 function (denoted by 0), which is the symmetric configuration of the tuple $(0, \ldots, 0) \in (\mathbb{F}_2^n)^\ell$.

Remark 3.2. *A configuration measures the Hamming weights of vectors in the subspace of $\mathbb{F}_2^n$ spanned by $z_1, \ldots, z_\ell$. However, Definition 3.1 depends on the choice of basis for the subspace. With more technical difficulty one can define configurations in a basis-independent way; see the discussion at the end of Section 4.1.*

Even though the space $(\mathbb{F}_2^n)^\ell$ has exponential size in n (for a fixed ℓ), the next lemma says that the number of configurations is polynomial in n (for a fixed ℓ).

Lemma 3.3. *We have*

$$\left| \mathrm{im}(\mathsf{Config}^{\Delta}_{n,\ell}) \right| = \binom{n + 2^\ell - 1}{2^\ell - 1}.$$

The next lemma provides an alternative way of viewing configurations: for each symmetric difference configuration $g \in \mathrm{im}(\mathsf{Config}^{\Delta}_{n,\ell})$, the set of ℓ-tuples with a certain configuration g is precisely one of the orbits of the natural (diagonal) right action of the symmetric group S_n on n points on $(\mathbb{F}_2^n)^\ell$.

Lemma 3.4. *Let $n, \ell \in \mathbb{N}_+$ and consider the natural (diagonal) right action of S_n on $(\mathbb{F}_2^n)^\ell$ given by $(x_1, \ldots, x_\ell) \cdot \sigma := (y_1, \ldots, y_\ell)$, where $(y_j)_i := (x_j)_{\sigma(i)}$ ($(x_1, \ldots, x_\ell), (y_1, \ldots, y_\ell) \in (\mathbb{F}_2^n)^\ell$, $\sigma \in S_n$, $j \in [\ell]$, $i \in [n]$).*

The following are equivalent for $(x_1, \ldots, x_\ell), (y_1, \ldots, y_\ell) \in (\mathbb{F}_2^n)^\ell$.

i. $(x_1, \dots, x_\ell)$ *and* $(y_1, \dots, y_\ell)$ *are in the same* S_n*-orbit.*

ii. $\mathsf{Config}^{\Delta}_{n,\ell}(x_1, \dots, x_\ell) = \mathsf{Config}^{\Delta}_{n,\ell}(y_1, \dots, y_\ell)$.

Similarly to the weight profile of a code, we can define a higher-order configuration profile.

Definition 3.5. *The* ℓ*-configuration profile of a code* $C \subseteq \mathbb{F}_2^n$ *is the sequence* $(a_g^C)_{g \in \mathrm{im}(\mathsf{Config}^{\Delta}_{n,\ell})}$ *defined by*

$$a_g^C := \frac{1}{|C|^\ell} \left| \left\{ ((x_1, \dots, x_\ell), (y_1, \dots, y_\ell)) \in C^\ell \times C^\ell \;\middle|\; \mathsf{Config}^{\Delta}_{n,\ell}(x_1 - y_1, \dots, x_\ell - y_\ell) = g \right\} \right|.$$

Remark 3.6. *Note that if* C *is linear,* a_g^C *can alternatively be computed as*

$$a_g^C = |\{(z_1, \dots, z_\ell) \in C^\ell \mid \mathsf{Config}^{\Delta}_{n,\ell}(z_1, \dots, z_\ell) = g\}|.$$

Recall that the (usual) Krawtchouk polynomial K_i of degree i is defined by

$$\begin{aligned} K_i(t) &:= 2^n \mathop{\mathbb{E}}_{x \in \mathbb{F}_2^n} [\mathbf{1}_{W_i}(x) \cdot \chi_y(x)] \\ &= \sum_{x \in W_i} \chi_y(x), \end{aligned}$$

where $W_i \subseteq \mathbb{F}_2^n$ is the set of all words of Hamming weight i, $\mathbf{1}_{W_i} \colon \mathbb{F}_2^n \to \{0,1\}$ is its indicator function and $y \in W_t$ is any element with of Hamming weight t.

Definition 3.7 (Higher-order Krawtchouk)**.** *Let* $h \in \mathrm{im}(\mathsf{Config}^{\Delta}_{n,\ell})$ *be a symmetric difference configuration. The higher-order Krawtchouk polynomial indexed by* h *is the function* $K_h \colon \mathrm{im}(\mathsf{Config}^{\Delta}_{n,\ell}) \to \mathbb{R}$ *defined by*

$$\begin{aligned} K_h(g) &:= 2^{\ell n} \mathop{\mathbb{E}}_{(y_1,\dots,y_\ell) \in (\mathbb{F}_2^n)^\ell} \left[\mathbb{1}_h(y_1, \dots, y_\ell) \cdot \prod_{j=1}^{\ell} \chi_{x_j}(y_j) \right] \\ &= \sum_{(y_1,\dots,y_\ell) \in h} \prod_{j=1}^{\ell} \chi_{x_j}(y_j), \end{aligned} \tag{1}$$

for every symmetric difference configuration $g \in \mathrm{im}(\mathsf{Config}^{\Delta}_{n,\ell})$*, where* $(x_1, \dots, x_\ell) \in g$ *is any* ℓ*-tuple of words with symmetric difference configuration* g *and* $\mathbb{1}_h$ *is the indicator function of the set of* ℓ*-tuples whose symmetric difference configuration is* h *(Lemma 3.19 shows this is well-defined).*

Remark 3.8. *Another way to see* K_h *above is as the unique function (see Lemma 3.19 below) such that*

$$\widehat{\mathbb{1}_h} = \frac{K_h \circ \mathsf{Config}^{\Delta}_{n,\ell}}{2^{n\ell}}.$$

Note that when $\ell = 1$, a symmetric difference configuration $\mathsf{Config}^{\Delta}_{n,1}(x)$ of a word $x \in \mathbb{F}_2^n$ only tracks the Hamming weight $\mathsf{Config}^{\Delta}_{n,1}(x)(\{1\}) = |x|$ of x (as $\mathsf{Config}^{\Delta}_{n,1}(x)(\varnothing)$ is always equal to 0) thus we recover the univariate Krawtchouk polynomials.

For explicit computations of the higher-order Krawtchouk polynomials, the formula (1) is quite inconvenient as it involves a sum of $2^{n\ell}$ terms. We will provide an alternative formula in Section 3.4.

3.2 Higher-order MacWilliams Identities and Inequalities

In this section, we show a higher-order analogue of MacWilliams identities and inequalities using only basic Fourier analysis. Later we are going to define a suitable family of association schemes from which MacWilliams identities and inequalities follow from the general theory of association schemes of Delsarte [Del73, DL98].

The MacWilliams identities show a surprising combinatorial fact: the weight profile of the dual $C^\perp$ of a linear code $C \subseteq \mathbb{F}_2^n$ is completely determined by the weight profile of C. The higher-order MacWilliams identities generalize this fact to ℓ-configuration profiles.

Lemma 3.9 (Higher-order MacWilliams identities)**.** *Let $C \in \mathbb{F}_2^n$ be a linear code and let $h \in \mathrm{im}(\mathsf{Config}^\Delta_{n,\ell})$ be a symmetric difference configuration. Then*

$$a_h^{C^\perp} = \frac{1}{|C|^\ell} \sum_{g \in \mathrm{im}(\mathsf{Config}^\Delta_{n,\ell})} K_h(g) \cdot a_g^C.$$

Proof. By Remark 3.6, we have

$$\begin{aligned} a_h^{C^\perp} &= 2^{n\ell} \left\langle \mathbb{1}_h, \mathbb{1}_{(C^\perp)^\ell} \right\rangle = 2^{n\ell} \sum_{x \in (\mathbb{F}_2^n)^\ell} \widehat{\mathbb{1}_h}(x) \widehat{\mathbb{1}_{(C^\perp)^\ell}}(x) \\ &= \frac{1}{|C|^\ell} \sum_{x \in (\mathbb{F}_2^n)^\ell} K_h(\mathsf{Config}^\Delta_{n,\ell}(x)) \cdot \mathbb{1}_{C^\ell}(x) = \frac{1}{|C|^\ell} \sum_{g \in \mathrm{im}(\mathsf{Config}^\Delta_{n,\ell})} K_h(g) \cdot a_g^C, \end{aligned}$$

where the second equality follows from Fact 2.1 and the third equality follows from Remark 3.8 and Fact 2.2. ∎

Just as the usual MacWilliams inequalities hold for arbitrary codes, we can prove that the same transformation at least yields non-negative numbers.

Lemma 3.10 (Higher-order MacWilliams inequalities)**.** *Let $C \in \mathbb{F}_2^n$ be an arbitrary code. For $h \in \mathrm{im}(\mathsf{Config}^\Delta_{n,\ell})$, we have*

$$\sum_{g \in \mathrm{im}(\mathsf{Config}^\Delta_{n,\ell})} K_h(g) \cdot a_g^C \geqslant 0.$$

Proof. Note that Remark 3.8 implies that the Fourier transform of $K_h \circ \mathsf{Config}^\Delta_{n,\ell}$ is $\mathbb{1}_h$, so we have

$$\begin{aligned} \sum_{g \in \mathrm{im}(\mathsf{Config}^\Delta_{n,\ell})} K_h(g) \cdot a_g^C &= \sum_{x,y \in (\mathbb{F}_2^n)^\ell} K_h(\mathsf{Config}^\Delta_{n,\ell}(x-y)) \cdot \mathbb{1}_{C^\ell}(x) \mathbb{1}_{C^\ell}(y) \\ &= 2^{2n\ell} \left\langle K_h \circ \mathsf{Config}^\Delta_{n,\ell}, \mathbb{1}_{C^\ell} * \mathbb{1}_{C^\ell} \right\rangle \\ &= 2^{2n\ell} \sum_{x \in (\mathbb{F}_2^n)^\ell} \mathbb{1}_h(x) \widehat{\mathbb{1}_{C^\ell}}^2 \geqslant 0, \end{aligned}$$

where the third equality follows from Fact 2.1. ∎

3.3 Higher-order Delsarte's Linear Programs

Now we have all the elements to define a hierarchy of linear programs for $A_2(n,d)$ parameterized by the size of the interactions $\ell \in \mathbb{N}_+$ in analogy to DelsarteLP(n,d).

Definition 3.11. *For $n, \ell \in \mathbb{N}_+$ and $d \in \{0,1,\dots,n\}$, we let* KrawtchoukLP(n,d,ℓ) *be the following linear program.*

$$\begin{aligned}
\max \quad & \sum_{g \in \mathsf{im}(\mathsf{Config}^{\Delta}_{n,\ell})} a_g & & & \\
\text{s.t.} \quad & a_0 = 1 & & & (\textit{Normalization}) \\
& a_g = 0 & & \forall g \in \mathsf{ForbConfig}(n,d,\ell) & (\textit{Distance constraints}) \\
& \sum_{g \in \mathsf{im}(\mathsf{Config}^{\Delta}_{n,\ell})} K_h(g) \cdot a_g \geqslant 0 & & \forall h \in \mathsf{im}(\mathsf{Config}^{\Delta}_{n,\ell}) & (\textit{MacWilliams inequalities}) \\
& a_g \geqslant 0 & & \forall g \in \mathsf{im}(\mathsf{Config}^{\Delta}_{n,\ell}) & (\textit{Non-negativity}),
\end{aligned}$$

where the variables are $(a_g)_{g \in \mathsf{im}(\mathsf{Config}^{\Delta}_{n,\ell})}$ and

$$\mathsf{ForbConfig}(n,d,\ell) \coloneqq \{g \in \mathsf{im}(\mathsf{Config}^{\Delta}_{n,\ell}) \mid \exists j \in [\ell], g(\{j\}) \in \{1,\dots,d-1\}\}.$$

We also define KrawtchoukLP$_{\mathrm{Lin}}(n,d,\ell)$ *as the linear program obtained by replacing the set* $\mathsf{ForbConfig}(n,d,\ell)$ *with*

$$\mathsf{ForbConfig}_{\mathrm{Lin}}(n,d,\ell) \coloneqq \{g \in \mathsf{im}(\mathsf{Config}^{\Delta}_{n,\ell}) \mid \exists J \subseteq [\ell], g(J) \in \{1,\dots,d-1\}\}.$$

Proposition 3.12. *The linear programs* KrawtchoukLP(n,d,ℓ) *and* KrawtchoukLP$_{\mathrm{Lin}}(n,d,\ell)$ *are sound, that is, we have*

$$\mathsf{val}(\mathrm{KrawtchoukLP}(n,d,\ell))^{1/\ell} \geqslant A_2(n,d),$$
$$\mathsf{val}(\mathrm{KrawtchoukLP}_{\mathrm{Lin}}(n,d,\ell))^{1/\ell} \geqslant A_2^{\mathrm{Lin}}(n,d).$$

Proof. Recall that for $C \subseteq \mathbb{F}_2^n$, we have

$$a_g^C \coloneqq \frac{1}{|C|^\ell} |\{(x_1,\dots,x_\ell),(y_1,\dots,y_\ell) \in C^\ell \times C^\ell \mid \mathsf{Config}^{\Delta}_{n,\ell}(x_1-y_1,\dots,x_\ell-y_\ell) = g\}|.$$

If C is an arbitrary code of distance at least d, then Lemma 3.10 implies that the ℓ-configuration profile a^C satisfies the MacWilliams inequalities. On the other hand, if $g \in \mathsf{ForbConfig}(n,d,\ell)$, that is, we have $g(\{j\}) \in \{1,\dots,d-1\}$ for some $j \in [\ell]$, then clearly no pair of ℓ-tuples of codewords $(x_1,\dots,x_\ell),(y_1,\dots,y_\ell) \in C^\ell$ can satisfy $\mathsf{Config}^{\Delta}_{n,\ell}(x_1-y_1,\dots,x_\ell-y_\ell) = g$ as it would imply $|x_j - y_j| = g(\{j\}) \in \{1,\dots,d-1\}$, thus the distance constraints are also satisfied.

All other restrictions follow trivially from the definition of a^C, thus a^C is a feasible solution of KrawtchoukLP(n,d,ℓ). Since the objective value of a^C is $\sum_{g \in \mathsf{im}(\mathsf{Config}^{\Delta}_{n,\ell})} a_g^C = |C|^\ell$, it follows that $\mathsf{val}(\mathrm{KrawtchoukLP}(n,d,\ell))^{1/\ell} \geqslant A_2(n,d)$.

If we further assume that C is linear and $g \in \mathsf{ForbConfig}_{\mathrm{Lin}}(n,d,\ell)$ is such that $g(J) \in [d-1]$ for some $J \subseteq [\ell]$, then no tuple $(z_1,\dots,z_\ell) \in C^\ell$ can satisfy $\mathsf{Config}^{\Delta}_{n,\ell}(z_1,\dots,z_\ell) = g$ as it would imply $|\sum_{j \in J} z_j| = g(J) \in \{1,\dots,d-1\}$. By Remark 3.6, we get $a_g^C = 0$, so a^C is also a feasible solution of KrawtchoukLP$_{\mathrm{Lin}}(n,d,\ell)$ and thus $\mathsf{val}(\mathrm{KrawtchoukLP}_{\mathrm{Lin}}(n,d,\ell))^{1/\ell} \geqslant A_2^{\mathrm{Lin}}(n,d)$. ∎

3.4 Properties of higher-order Krawtchouk polynomials

In this section, we explore more properties of the higher-order Krawtchouk polynomials in order to show that the objective and restrictions of the linear programs $\mathrm{KrawtchoukLP}(n,d,\ell)$ and $\mathrm{KrawtchoukLP}_{\mathrm{Lin}}(n,d,\ell)$ can be algorithmically computed in $O(n^{2^{\ell+1}-2})$ time for a fixed $\ell \in \mathbb{N}_+$ (see Proposition 3.21).

Even though symmetric difference configurations are more natural from the point of view of linear codes, for computations and properties with the higher-order Krawtchouk polynomials, it is more convenient to work with Venn diagram configurations defined below. In plain English, each word $z \in \mathbb{F}_2^n$ induces a partition of $[n]$ into its support $\mathrm{supp}(z) \coloneqq \{i \in [n] \mid z_i \neq 0\}$ and its complement $[n] \setminus \mathrm{supp}(z)$; the Venn diagram configuration of a tuple $(z_1,\dots,z_\ell) \in (\mathbb{F}_2^n)^\ell$ then encodes the information about the sizes of each of the cells of the Venn diagram of the coarsest common refinement of the partitions induced by the z_i.

Definition 3.13. *The Venn diagram configuration of the ℓ-tuple $(z_1,\dots,z_\ell) \in (\mathbb{F}_2^n)^\ell$ is the function $\mathsf{Config}^V_{n,\ell}(z_1,\dots,z_\ell)\colon 2^{[\ell]} \to \mathbb{R}$ defined by*

$$\begin{aligned}\mathsf{Config}^V_{n,\ell}(z_1,\dots,z_\ell)(J) &\coloneqq \left|\bigcap_{j\in J}\mathrm{supp}(z_i) \cap \bigcap_{j\in[\ell]\setminus J}([n]\setminus\mathrm{supp}(z_j))\right| \\ &= \left|\left\{i \in [n] \;\middle|\; \{j\in[\ell] \mid (z_j)_i = 1\} = J\right\}\right|,\end{aligned}$$

for every $J \subseteq [\ell]$.

By viewing $\mathsf{Config}^V_{n,\ell}$ as a function $(\mathbb{F}_2^n)^\ell \to \mathbb{R}^{2^{[\ell]}}$, the set of (valid) Venn diagram configurations of ℓ-tuples of elements of $\mathbb{F}_2^n$ is $\mathrm{im}(\mathsf{Config}^V_{n,\ell})$.

The next lemma gives an easy description of the set of Venn diagram configurations of ℓ-tuples of elements of $\mathbb{F}_2^n$ as the set of all functions $2^{[\ell]} \to \mathbb{R}$ whose values are non-negative integers that add up to n. Combining it with Lemma 3.15 below gives an explicit description of the set of symmetric difference configurations.

Lemma 3.14. *For every $n,\ell \in \mathbb{N}_+$, we have*

$$\mathrm{im}(\mathsf{Config}^V_{n,\ell}) = \left\{g\colon 2^{[\ell]} \to \mathbb{R} \;\middle|\; \sum_{J\subseteq[\ell]} g(J) = n \wedge \forall J \subseteq [\ell], g(J) \in \mathbb{N}\right\}. \tag{2}$$

The next lemma provides a pair of linear transformations that transform a symmetric difference configuration into a Venn diagram configuration and vice-versa.

Lemma 3.15. *Let $n,\ell \in \mathbb{N}_+$, let*

$$S_{n,\ell} \stackrel{\text{def}}{=} \left\{g \in \mathbb{R}^{2^{[\ell]}} \;\middle|\; \sum_{J\subseteq[\ell]} g(J) = n\right\}, \qquad Z_{n,\ell} \stackrel{\text{def}}{=} \{g \in \mathbb{R}^{2^{[\ell]}} \mid g(\varnothing) = 0\}$$

and let $V_{n,\ell}\colon Z_{n,\ell} \to S_{n,\ell}$ *and* $D_{n,\ell}\colon S_{n,\ell} \to Z_{n,\ell}$ *be given by*

$$D_{n,\ell}(g)(J) \stackrel{\text{def}}{=} \sum_{\substack{T\subseteq[\ell]\\ |T\cap J| \text{ odd}}} g(T), \tag{3}$$

$$V_{n,\ell}(g)(J) \stackrel{\text{def}}{=} n\cdot \mathbb{1}[J=\varnothing] + 2^{1-\ell}\sum_{T\subseteq[\ell]} (-1)^{|T\cap J|-1} g(T), \tag{4}$$

for every $J\subseteq[\ell]$.

Then $V_{n,\ell}$ *and* $D_{n,\ell}$ *are inverses of each other and* $\mathsf{Config}^{\Delta}_{n,\ell} = D_{n,\ell}\circ \mathsf{Config}^{V}_{n,\ell}$ *and* $\mathsf{Config}^{V}_{n,\ell} = V_{n,\ell}\circ \mathsf{Config}^{\Delta}_{n,\ell}$.

Making use of Venn diagram configurations, we can also easily compute the number of ℓ-tuples with a given configuration as a multinomial.

Lemma 3.16. *For a symmetric difference configuration* $g\in \mathrm{im}(\mathsf{Config}^{\Delta}_{n,\ell})$*, we have*

$$|g| = K_g(0) = \binom{n}{V_{n,\ell}(g)} = \frac{n!}{\prod_{J\subseteq[\ell]} V_{n,\ell}(g)(J)!},$$

where $V_{n,\ell}$ *is given by* (4).

The following lemma says that, similarly to the univariate case, the higher-order Krawtchouk polynomials are orthogonal with respect to the natural discrete measure on symmetric configurations in which each $g\in\mathrm{im}(\mathsf{Config}^{\Delta}_{n,\ell})$ has measure $|g| = \binom{n}{V_{n,\ell}(g)}$ (see Lemma 3.16), i.e., the number of ℓ-tuples with configuration g.

Lemma 3.17. *[Orthogonality] For* $n,\ell\in\mathbb{N}_+$ *and* $h,h'\in\mathrm{im}(\mathsf{Config}^{\Delta}_{n,\ell})$*, we have*

$$\sum_{g\in\mathrm{im}(\mathsf{Config}^{\Delta}_{n,\ell})} |g|\cdot K_h(g)\cdot K_{h'}(g) = 2^{\ell n}\cdot |h|\cdot \mathbb{1}[h=h'].$$

Also similarly to the univariate case, the higher-order Krawtchouk polynomials satisfy the following reflection property.

Lemma 3.18. *[Reflection] For* $n,\ell\in\mathbb{N}_+$ *and* $g,h\in\mathrm{im}(\mathsf{Config}^{\Delta}_{n,\ell})$*, we have*

$$\frac{K_h(g)}{|h|} = \frac{K_g(h)}{|g|}.$$

The next lemma provides an alternative formula for the higher-order Krawtchouk polynomial in which the sum involves only $O(n^{2^{2\ell}})$ terms (as opposed to the $2^{\ell n}$ terms in (1)).

Lemma 3.19. *For every* $n,\ell\in\mathbb{N}_+$ *and every* $g,h\in\mathrm{im}(\mathsf{Config}^{\Delta}_{n,\ell})$*, we have*

$$K_h(g) = \sum_{F\in\mathcal{F}}\prod_{J\subseteq[\ell]}\frac{V_{n,\ell}(g)(J)!}{\prod_{K\subseteq[\ell]}F(J,K)!}\cdot\prod_{j=1}^{\ell}\prod_{\substack{J,K\subseteq[\ell]\\ j\in J\cap K}}(-1)^{F(J,K)},$$

where $\mathcal{F}$ is the set of functions $F\colon 2^{[\ell]} \times 2^{[\ell]} \to \{0,1,\dots,n\}$ such that

$$\forall J \subseteq [\ell],\ \sum_{K\subseteq[\ell]} F(J,K) = V_{n,\ell}(g)(J),$$
$$\forall K \subseteq [\ell],\ \sum_{J\subseteq[\ell]} F(J,K) = V_{n,\ell}(h)(K),$$

and $V_{n,\ell}$ is given by (4).

The next lemma allows the computation of the Krawtchouk polynomials even faster via dynamic programming.

Lemma 3.20. *Let $n, \ell \in \mathbb{N}_+$ with $n \geqslant 2$, let $g, h \in \operatorname{im}(\mathsf{Config}^{\Delta}_{n,\ell})$ be symmetric difference configurations and let $J_0 \subseteq [\ell]$ be such that $V_{n,\ell}(g)(J_0) > 0$ for $V_{n,\ell}$ given by (4). Then*

$$K_h(g) = \sum_{\substack{K_0\subseteq[\ell]\\ V_{n,\ell}(h)(K_0)>0}} (-1)^{|J_0\cap K_0|} \cdot K_{h\ominus K_0}(g \ominus J_0), \tag{5}$$

$$K_h(g) = -\sum_{\substack{K_0\subseteq[\ell]\\ V(h)(K_0)>0\\ K_0\neq\varnothing}} K_{h\oplus\varnothing\ominus K_0}(g) + \sum_{\substack{K_0\subseteq[\ell]\\ V(h)(K_0)>0}} (-1)^{|J_0\cap K_0|} \cdot K_{h\oplus\varnothing\ominus K_0}(g\oplus\varnothing\ominus J_0), \tag{6}$$

where

$$h \ominus K_0 \coloneqq D_{n-1,\ell}(V_{n,\ell}(h) - \mathbb{1}_{\{K_0\}}), \qquad g \ominus J_0 \coloneqq D_{n-1,\ell}(V_{n,\ell}(g) - \mathbb{1}_{\{J_0\}}),$$
$$h \oplus \varnothing \coloneqq D_{n+1,\ell}(V_{n,\ell}(h) + \mathbb{1}_{\{\varnothing\}}), \qquad g \oplus \varnothing \coloneqq D_{n+1,\ell}(V_{n,\ell}(g) + \mathbb{1}_{\{\varnothing\}}),$$

and $D_{n-1,\ell}$ and $D_{n+1,\ell}$ are given by (3).

Proposition 3.21. *The objective and restrictions of the linear programs* $\mathrm{KrawtchoukLP}(n,d,\ell)$ *and* $\mathrm{KrawtchoukLP}_{\mathrm{Lin}}(n,d,\ell)$ *can be algorithmically computed in $O(n^{2^{\ell+1}-2})$ time for a fixed $\ell \in \mathbb{N}_+$.*

Proof. The number of variables and restrictions of these linear programs is the number of configurations at level ℓ, which is $O(n^{2^\ell-1})$ by Lemma 3.3. Furthermore, converting between symmetric difference configurations and Venn diagram configurations using Lemma 3.15 can be done in time $O(2^\ell) = O(1)$ and using Lemma 3.19 and (6) in Lemma 3.20, we can compute all values of all Krawtchouk polynomials of level ℓ in time $O((n^{2^\ell-1})^2) = O(n^{2^{\ell+1}-2})$. ∎

4 Unsymmetrized Formulations of the Krawtchouk Hierarchies

In this section we give other formulations for KrawtchoukLP. These formulations are *unsymmetrized* versions of the same hierarchy. Working with the unsymmetrized hierarchy can be easier, since it avoids the technical definitions of the Krawtchouk polynomials $K_h(g)$, but computationally the number of variables and constraints of these hierarchies is huge.

4.1 The Hierarchy as Checking Non-negativity of Fourier Coefficients

The LP hierarchy for linear codes can be simply described as checking non-negativity of products of Fourier coefficients. Define the linear programming hierarchy $\mathrm{FourierLP}_{\mathrm{Lin}}(n,d,\ell)$ with the variables a_x $(x \in (\mathbb{F}_2^n)^\ell)$:

$$\begin{aligned}
\max \quad & \sum_{x\in(\mathbb{F}_2^n)^\ell} a_x \\
\text{s.t.} \quad & a_0 = 1 && && \text{(Normalization)}\\
& a_{(x_1,\dots,x_\ell)} = 0 && \exists w \in \mathrm{span}(x_1,\dots,x_\ell), |w| \in \{1,\dots,d-1\} && \text{(Distance constraints)}\\
& \sum_{x\in(\mathbb{F}_2^n)^\ell} a_x \chi_\alpha(x) \geqslant 0 && \forall \alpha \in (\mathbb{F}_2^n)^\ell && \text{(Fourier coefficients)}\\
& a_x \geqslant 0 && \forall x \in (\mathbb{F}_2^n)^\ell && \text{(Non-negativity).}
\end{aligned}$$

Proposition 4.1. *For every $n,\ell \in \mathbb{N}_+$ and $d \in \{0,1,\dots,n\}$, $\mathsf{val}(\mathrm{FourierLP}_{\mathrm{Lin}}(n,d,\ell)) \geqslant A_2^{\mathrm{Lin}}(n,d)^\ell$.*

Proof. Given a linear code C with distance d, a feasible solution with value $|C|^\ell$ is $a_{(x_1,\dots,x_\ell)} := \prod_{i=1}^\ell \mathbb{1}[x_i \in C]$. The Fourier coefficient constraints are satisfied because

$$\sum_{x\in(\mathbb{F}_2^n)^\ell} \prod_{i=1}^\ell \mathbb{1}[x_i \in C]\chi_{\alpha_i}(x_i) = 2^{n\ell} \prod_{i=1}^\ell \widehat{\mathbf{1}_C}(\alpha_i),$$

which are nonnegative by Fact 2.2. ∎

The corresponding hierarchy for non-linear codes $\mathrm{FourierLP}(n,d,\ell)$ is defined over the variables a_x $(x \in (\mathbb{F}_2^n)^\ell)$ as:

$$\begin{aligned}
\max \quad & \sum_{x\in(\mathbb{F}_2^n)^\ell} a_x \\
\text{s.t.} \quad & a_0 = 1 && && \text{(Normalization)}\\
& a_{(x_1,\dots,x_\ell)} = 0 && \exists i \in [\ell], |x_i| \in \{1,\dots,d-1\} && \text{(Distance constraints)}\\
& \sum_{x\in(\mathbb{F}_2^n)^\ell} a_x \chi_\alpha(x) \geqslant 0 && \forall \alpha \in (\mathbb{F}_2^n)^\ell && \text{(Fourier coefficients)}\\
& a_x \geqslant 0 && \forall x \in (\mathbb{F}_2^n)^\ell && \text{(Non-negativity).}
\end{aligned}$$

It turns out that KrawtchoukLP is a symmetrization of FourierLP (and likewise for the programs $\mathrm{KrawtchoukLP}_{\mathrm{Lin}}$ and $\mathrm{FourierLP}_{\mathrm{Lin}}$). We will briefly describe the technique of symmetrizing convex programs, which is also described in the survey article by Vallentin [Val19]. The proof that KrawtchoukLP and FourierLP are equivalent continues at Proposition 4.5.

The technique exploits the fact that convex relaxations for the independence number $\alpha(H_{n,d})$ of the Hamming cube graph $H_{n,d}$ of distance less than d are highly symmetric, that is, programs that are invariant under large permutation groups as defined below.

Definition 4.2 (Program invariance)**.** *Let $\mathcal{P}$ be a linear program with variables $(a_x)_{x\in X}$ for some set X. We say that $\mathcal{P}$ is invariant under a permutation σ of X if for all feasible solutions (a_x), the point $a \cdot \sigma$ defined by $(a \cdot \sigma)_x := a_{\sigma(x)}$ is also feasible, and the objective value is the same.*

Similarly, a semi-definite program $\mathcal{P}$ with variable $M \in \mathbb{R}^{X\times X}$ is invariant under σ if for all feasible M, the matrix $M \cdot \sigma$ defined by $(M \cdot \sigma)[x, y] := M[\sigma(x), \sigma(y)]$ is also feasible, and the objective value is the same.

The group of permutations of X under which $\mathcal{P}$ is invariant is called the automorphism group of $\mathcal{P}$ and is denoted $\mathrm{Aut}(\mathcal{P})$.

If the input of a program $\mathcal{P}$ is a graph G and the program only depends on the isomorphism class of G, then the program is invariant under the automorphism group $\mathrm{Aut}(G)$ of the graph G. For convex relaxations such as the Lovász ϑ-function or the Sum-of-Squares hierarchy, the variables of the program are indexed by tuples of vertices from G, and thus a case of interest is when $\mathrm{Aut}(G)$ acts diagonally on tuples of vertices.

By symmetrizing solutions, i.e., by averaging the values of the variables over the automorphism group $\mathrm{Aut}(\mathcal{P})$, we may assume that the solution has the same symmetry:

Fact 4.3. *For any $H \subseteq \mathrm{Aut}(\mathcal{P})$, the value $\mathsf{val}(\mathcal{P})$ equals the value of $\mathcal{P}$ with the additional constraints $\forall \sigma \in H, \forall x \in X, a_x = a_{\sigma(x)}$ (or $\forall \sigma \in H, \forall x, y \in X, M[x, y] = M[\sigma(x), \sigma(y)]$ for an SDP).*

A symmetrized solution is constant on each orbit of the group action on X or X^2. Therefore, the "effective" number of variables in the convex program is only the number of orbits, which may be significantly smaller than even $|V(G)|$.

For example, the graph $H_{n,d}$ has a large symmetry group:

Fact 4.4. *For $1 < d < n$, $\mathrm{Aut}(H_{n,d})$ is the hyperoctahedral group, which is the semidirect product $\mathbb{F}_2^n \rtimes S_n$ in which S_n permutes the coordinates and $\mathbb{F}_2^n$ applies a bit flip.*

Even though the hypercube has size 2^n and thus $|V(H_{n,\ell})^\ell| = 2^{n\ell}$, the number of orbits of the diagonal action of $\mathrm{Aut}(H_{n,d})$ on ℓ-tuples is only $\mathrm{poly}(n)$ for constant ℓ. For example, for $\ell = 4$, viewing the hypercube momentarily as $\{-1, +1\}^n$, the orbit of (x_1, x_2, x_3, x_4) essentially only depends on the angles between the vectors: it is determined by the seven numbers

$$\langle x_1, x_2\rangle, \quad \langle x_1, x_3\rangle, \quad \langle x_1, x_4\rangle, \quad \langle x_2, x_3\rangle, \quad \langle x_2, x_4\rangle, \quad \langle x_3, x_4\rangle, \quad \sum_{i=1}^{n} x_{1,i}x_{2,i}x_{3,i}x_{4,i}. \tag{7}$$

Equivalently, it is determined by $\mathsf{Config}^{\Delta}_{n,\ell}(x_2 - x_1, x_3 - x_1, x_4 - x_1)$ (see Lemma 3.4).

Since each of the numbers in (7) takes at most $n + 1$ values, the effective number of variables in the degree-4 Sum-of-Squares relaxation for $\alpha(H_{n,d})$ is at most $O(n^7)$. Thus, the search for an upper bound on an exponential-size object is reduced to a polynomial-size convex program! Of course, to actually run this in polynomial time, one also needs to show that this polynomial-size convex program can be computed in polynomial time (which rules out explicitly computing the original program then taking a quotient).

We use the symmetrization technique to show that KrawtchoukLP and FourierLP are equivalent.

Proposition 4.5. *For every $n, \ell \in \mathbb{N}_+$ and every $d \in \{0, 1 \ldots, n\}$, we have*

$$\mathsf{val}(\mathrm{FourierLP}(n,d,\ell)) = \mathsf{val}(\mathrm{KrawtchoukLP}(n,d,\ell)),$$
$$\mathsf{val}(\mathrm{FourierLP}_{\mathrm{Lin}}(n,d,\ell)) = \mathsf{val}(\mathrm{KrawtchoukLP}_{\mathrm{Lin}}(n,d,\ell)).$$

Proof. Recall that the natural right action of S_n on $\mathbb{F}_2^n$ is given by $(x \cdot \sigma)_i := x_{\sigma(i)}$ ($x \in \mathbb{F}_2^n, \sigma \in S_n$, $i \in [n]$) and consider the diagonal action of S_n on $(\mathbb{F}_2^n)^\ell$ given by

$$(x_1, \ldots, x_\ell) \cdot \sigma := (x_1 \cdot \sigma, \ldots, x_\ell \cdot \sigma) \quad ((x_1, \ldots, x_\ell) \in (\mathbb{F}_2^n)^\ell, \sigma \in S_n).$$

It is straightforward to check that $\mathrm{FourierLP}(n,d,\ell)$ is invariant under this diagonal action.

By Fact 4.3 we may consider only solution to the LP that are symmetrized over S_n, that is, we have $a_x = a_y$ for each $x, y \in (\mathbb{F}_2^n)^\ell$ in the same orbit of the S_n-action.

Recall from Lemma 3.4 that ℓ-tuples of words are in the same S_n-orbit if and only if they have the same symmetric difference configuration. We claim that the correspondence between the program KrawtchoukLP with variables $(a'_g)_{g \in \mathrm{im}(\mathsf{Config}^\Delta_{n,\ell})}$ and FourierLP is

$$a'_g = |g| \cdot a_{x_1,\ldots,x_\ell} \qquad \text{for any } (x_1, \ldots, x_\ell) \in g.$$

It is straightforward to check that the objective function, normalization, distance, and non-negativity constraints for $\mathrm{FourierLP}(n,d,\ell)$ (under the assumption of an S_n-invariant solution) match exactly those of $\mathrm{KrawtchoukLP}(n,d,\ell)$. For the MacWilliams inequalities, note that for every $h \in \mathrm{im}(\mathsf{Config}^\Delta_{n,\ell})$ and every $\alpha \in h$, we have

$$\begin{aligned}
&\sum_{g \in \mathsf{Config}(n,\ell)} a'_g K_h(g) \geqslant 0 \\
&\Leftrightarrow \sum_{g \in \mathsf{Config}(n,\ell)} a'_g \frac{|h|}{|g|} K_g(c) \geqslant 0 && \text{(Reflection, Lemma 3.18)} \\
&\Leftrightarrow \sum_{g \in \mathsf{Config}(n,\ell)} \frac{a'_g}{|g|} K_g(h) \geqslant 0 \\
&\Leftrightarrow \sum_{x \in (\mathbb{F}_2^n)^\ell} a_x \chi_\alpha(x) \geqslant 0 && \text{(Definition of } K_g),
\end{aligned}$$

where the third equivalence follows since $a_g/|g| = a_x$ for every $x \in g$.

The same proof goes through for $\mathrm{FourierLP}_{\mathrm{Lin}}$ and $\mathrm{KrawtchoukLP}_{\mathrm{Lin}}$. ∎

Remark 4.6. *The linear programs* FourierLP *and* $\mathrm{FourierLP}_{\mathrm{Lin}}$ *are not invariant under the other automorphisms of the hypercube of the form $x \mapsto x + z$ ($z \in \mathbb{F}_2^n$), because of the normalization constraint and the distance constraints. It makes more sense to view the underlying space as $\mathbb{F}_2^n$ instead of the hypercube, which does not have the $\mathbb{F}_2^n$ automorphism because the origin is treated specially.*

There is actually more symmetry in the programs than just S_n. In the case of the program for non-linear codes, there is a symmetry under the right action of S_ℓ on $(\mathbb{F}_2^n)^\ell$ that permutes the *words* $x_1, \ldots, x_\ell$, that is, we have $(x_1, \ldots, x_\ell) \cdot \tau := (x_{\tau(1)}, \ldots, x_{\tau(\ell)})$ ($(x_1, \ldots, x_\ell) \in (\mathbb{F}_2^n)^\ell, \tau \in S_\ell$). In the

case of the program for linear codes, we have symmetry under the action of $\mathrm{GL}_\ell(\mathbb{F}_2)$ that applies a basis change to $(x_1, \ldots, x_\ell)$, that is, it is given by

$$(A \cdot x)_i := \sum_{j \in [\ell]} A[i,j] \cdot x_j \in \mathbb{F}_2^n$$

for every $A \in \mathrm{GL}_\ell(\mathbb{F}_2)$, every $x \in (\mathbb{F}_2^n)^\ell$ and every $i \in [\ell]$. The distance constraints are evidently invariant under this action as it does not change the linear subspace spanned by $(x_1, \ldots, x_\ell)$. The Fourier constraints are invariant since

$$\chi_\alpha(A \cdot x) = \chi_{A^\top \cdot \alpha}(x)$$

for every $x, \alpha \in \mathbb{F}_2^\ell$.

Note that the actions of $\mathrm{GL}_\ell(\mathbb{F})$ and S_n commute with each other and thus induce an action of the direct product $\mathrm{GL}_\ell(\mathbb{F}) \times S_n$. Another reasonable definition of the higher-order Krawtchouk polynomials and linear program symmetrizes under this larger group action of $\mathrm{GL}_\ell(\mathbb{F}) \times S_n$. There is one Krawtchouk polynomial and one free variable for each orbit of this action.

Definition 4.7 (Fully symmetrized higher-order Krawtchouks)**.** *Let $O := (\mathbb{F}_2^n)^\ell / (\mathrm{GL}_\ell(\mathbb{F}_2) \times S_n)$ be the set of orbits of the $(\mathrm{GL}_\ell(\mathbb{F}_2) \times S_n)$-action as above. For each $h \in O$ we define the higher-order Krawtchouk polynomial $K_h \colon O \to \mathbb{R}$ by*

$$K_h(g) := \sum_{(\alpha_1, \ldots, \alpha_\ell) \in h} \prod_{j=1}^{\ell} \chi_{\alpha_j}(x_j),$$

where $(x_1, \ldots, x_\ell)$ is any element in the orbit $g \in O$.

Since the symmetry group is larger and the number of orbits is smaller, the size of the resulting LP is smaller. However, since $|\mathrm{GL}_\ell(\mathbb{F}_2)| = \prod_{t=0}^{\ell-1}(2^\ell - 2^t) = O_\ell(1)$, for a constant ℓ, this would only decrease the size of KrawtchoukLP by a constant factor. For practical computations, constant factors make a difference and this symmetrization should likely be performed. We chose our definition of Krawtchouks in Section 3 because the orbits are simpler to describe (being captured by explicit combinatorial objects, configuration functions) and we can compute the set of orbits and the Krawtchouk polynomials efficiently (see Proposition 3.21).

There is an equivalent interpretation of $(\mathrm{GL}_\ell(\mathbb{F}_2) \times S_n)$-orbits as "subspace weight profiles" as follows. The right action of S_n naturally induces an action over linear subspaces of $\mathbb{F}_2^n$ given by

$$W \cdot \sigma := \{w \cdot \sigma \mid w \in W\} \quad (W \leqslant \mathbb{F}_2^n, \sigma \in S_n).$$

It is straightforward to see that two ℓ-tuples $(x_1, \ldots, x_\ell), (y_1, \ldots, y_\ell) \in (\mathbb{F}_2^n)^\ell$ are in the same $(\mathrm{GL}_\ell(\mathbb{F}_2) \times S_n)$-orbit if and only if $\mathrm{span}\{x_1, \ldots, x_\ell\}$ and $\mathrm{span}\{y_1, \ldots, y_\ell\}$ are in the same S_n-orbit, which in turn is equivalent to saying that both spaces have the same dimension, say k, and there are ordered bases $b^x = (b_1^x, \ldots, b_k^x)$ and $b^y = (b_1^y, \ldots, b_k^y)$ of these spaces respectively such that $\mathsf{Config}^\Delta_{n,k}(b^x) = \mathsf{Config}^\Delta_{n,k}(b^y)$. Thus, the hierarchy corresponding to the $(\mathrm{GL}_\ell(\mathbb{F}_2) \times S_n)$-action has an interesting interpretation as measuring weight statistics of linear subspaces of the linear code of dimension at most ℓ.

4.2 The Hierarchy as an SDP

The LP hierarchy is also equivalent to an SDP relaxation with the harsh constraint that the SDP matrix must be *translation invariant*.

Define the semi-definite program TranslationSDP(n, d, ℓ) as

$$\begin{aligned}
\max \quad & \sum_{x \in (\mathbb{F}_2^n)^\ell} M[0, x] \\
\text{s.t.} \quad & M[0,0] = 1 && && \text{(Normalization)} \\
& M[0, (x_1, \ldots, x_\ell)] = 0 && \exists i \in [\ell], |x_i| \in \{1, \ldots, d-1\} && \text{(Distance constraints)} \\
& M[x, y] = M[0, y - x] && \forall x, y \in (\mathbb{F}_2^n)^\ell && \text{(Translation symmetry)} \\
& M \succcurlyeq 0 && && \text{(PSD-ness)} \\
& M[x, y] \geqslant 0 && \forall x, y \in (\mathbb{F}_2^n)^\ell && \text{(Non-negativity)},
\end{aligned}$$

where the variable is $M \in \mathbb{R}^{(\mathbb{F}_2^n)^\ell \times (\mathbb{F}_2^n)^\ell}$.

To form TranslationSDP$_{\text{Lin}}(n, d, \ell)$, replace the distance constraints by

$$M[0, (x_1, \ldots, x_\ell)] = 0 \quad \exists w \in \operatorname{span}(x_1, \ldots, x_\ell), |w| \in \{1, \ldots, d-1\}.$$

The crucial translation symmetry property of TranslationSDP ensures M lies in the *commutative* matrix algebra $\operatorname{span}\{D_z \mid z \in (\mathbb{F}_2^n)^\ell\}$, where

$$D_z[x, y] \coloneqq \mathbb{1}[y - x = z].$$

The coefficient of M on D_z is $M[0, z]$.

Since the matrices D_z commute, they are simultaneously diagonalizable. More specifically, their common eigenvectors are the Fourier characters.

Fact 4.8. *The matrices D_z are simultaneously diagonalized by $(\chi_\alpha \mid \alpha \in (\mathbb{F}_2^n)^\ell)$ with the eigenvalue of D_z on χ_α being $\chi_\alpha(z)$.*

Therefore, the PSD-ness constraint in TranslationSDP is particularly simple: to check that $\lambda_z D_z \succcurlyeq 0$, it is equivalent to check $\sum_{z \in (\mathbb{F}_2^n)^\ell} \lambda_z \chi_\alpha(z) \geqslant 0$ for all $\alpha \in (\mathbb{F}_2^n)^\ell$. This is a linear constraint on the λ_z, and hence we can express the SDP as an LP, giving yet another formulation of the hierarchy.

Proposition 4.9. *For every $n, \ell \in \mathbb{N}_+$ and every $d \in \{0, 1, \ldots, n\}$, we have*

$$\begin{aligned}
\mathsf{val}(\text{FourierLP}(n, d, \ell)) &= \mathsf{val}(\text{TranslationSDP}(n, d, \ell)), \\
\mathsf{val}(\text{FourierLP}_{\text{Lin}}(n, d, \ell)) &= \mathsf{val}(\text{TranslationSDP}_{\text{Lin}}(n, d, \ell)).
\end{aligned}$$

Proof. The formal correspondence of the variables is $M[0, x] = a_x$. The Fourier coefficient constraints in FourierLP are equivalent to PSD-ness as described above, and the other constraints also match up. ■

Along with Proposition 4.5, the above implies that TranslationSDP also has the same value as KrawtchoukLP.

Remark 4.10. *In previous convex relaxations for $A_2(n,d)$, in order to implement the program efficiently, a key technical step has been finding an explicit block diagonalization of the SDP matrix (which reduces the program size). This step requires significant technical work [Sch05, GMS12, Gij09]. An advantage of the LP hierarchy is that complete diagonalization is trivial.*

4.3 The Hierarchy as ϑ'

The hierarchy can also be seen as computing the (modified) Lovász ϑ' function on progressively larger graphs, whose definition is recalled below. In fact, this formulation of the hierarchy holds for any *association scheme* (see Theorem 5.22 below).

Definition 4.11 (ϑ' Program)**.** *The (modified) Lovász ϑ' function is defined as follows. For a graph G, $\vartheta'(G)$ is the optimum value of the semi-definite program $\mathcal{S}(G)$ given by*

$$\begin{array}{lll} \max & \langle J, M\rangle & \\ s.t. & \operatorname{tr} M = 1 & (\textit{Normalization}) \\ & M[u,v] = 0 \qquad \forall \{u,v\} \in E(G) & (\textit{Independent set}) \\ & M \succcurlyeq 0 & (\textit{PSD-ness}) \\ & M[u,v] \geq 0 \qquad \forall u,v \in V(G) & (\textit{Non-negativity}), \end{array}$$

where the variable is $M \in \mathbb{R}^{V\times V}$ symmetric, J is the all ones matrix and $\langle A, B\rangle := \operatorname{tr}(A^\top B)$.

By strong duality $\vartheta'(G)$ is also the optimum value of the dual semi-definite program $\mathcal{S}'(G)$ given by

$$\begin{array}{lll} \min & \beta & \\ s.t. & \beta I - N \succcurlyeq 0 & (\textit{PSD-ness}) \\ & N[u,v] \geqslant 1 \qquad \forall u,v \in V(G) \text{ with } \{u,v\} \notin E & (\textit{Independent set}), \end{array}$$

where the variables are $N \in \mathbb{R}^{V\times V}$ symmetric and $\beta \in \mathbb{R}$.

It is straightforward to see that $\vartheta'(G)$ is an upper bound for the independence number of the graph G since if $A \subseteq V(G)$ is an independent set, then $\mathbb{1}_A \mathbb{1}_A^\top / |A|$ is a feasible solution of $\mathcal{S}(G)$ with value $|A|$.

In the same way that a code $C \subseteq \mathbb{F}_2^n$ of distance at least d can be seen as an independent set in the graph $H_{n,d}$, we can see C^ℓ as an independent set in exclusion graphs defined below based on the sets $\mathsf{ForbConfig}(n,d,\ell)$ and $\mathsf{ForbConfig}_{\mathrm{Lin}}(n,d,\ell)$ of Definition 3.11.

Definition 4.12 (Exclusion Graph)**.** *We define the exclusion graph $H_{n,d,\ell}$ to have vertex set $(\mathbb{F}_2^n)^\ell$ and edge set*

$$E(H_{n,d,\ell}) := \left\{ (x,y) \in \binom{(\mathbb{F}_2^n)^\ell}{2} \;\middle|\; \mathsf{Config}^{\Delta}_{n,\ell}(x-y) \in \mathsf{ForbConfig}(n,d,\ell) \right\}.$$

We define $H^{\mathrm{Lin}}_{n,d,\ell}$ analogously replacing $\mathsf{ForbConfig}(n,d,\ell)$ with $\mathsf{ForbConfig}_{\mathrm{Lin}}(n,d,\ell)$.

Lemma 4.13. *For every $n, \ell \in \mathbb{N}_+$ and every $d \in \{0,1,\dots,n\}$, we have*

$$\begin{aligned} \mathsf{val}(\mathrm{TranslationSDP}(n,d,\ell)) &= \mathsf{val}(\vartheta'(H_{n,d,\ell})), \\ \mathsf{val}(\mathrm{TranslationSDP}_{\mathrm{Lin}}(n,d,\ell)) &= \mathsf{val}(\vartheta'(H^{\mathrm{Lin}}_{n,d,\ell})). \end{aligned}$$

Proof. The program $\mathcal{S}(H_{n,d,\ell})$ corresponding to $\vartheta'(H_{n,d,\ell})$ is invariant under $\mathrm{Aut}(G)$, so it is invariant in particular under the translation action of $\mathbb{F}_2^n$ on itself.

Therefore, by Fact 4.3, we may consider only solutions of $\mathcal{S}(H_{n,d,\ell})$ that are translation invariant. Now there is a correspondence between solutions M for TranslationSDP and translation invariant solutions M' for $\mathcal{S}(H_{n,d,\ell})$ given by $M = 2^{n\ell} \cdot M'$. The proof goes through similarly for the linear case. ■

5 Generalized Krawtchouk Hierarchies from Association Schemes

In this section, we recall some of the basic definitions and results of association scheme theory and show that our construction generalizes nicely to translation schemes with an underlying left module structure over some ring, producing a translation scheme "refining" a tensor power of the original scheme; once this is shown, MacWilliams identities and inequalities follow from the theory of translation schemes. The general theory of association schemes will also be used to show our completeness and lifting results of Section 6.1 and Section 6.2.

Further background on association scheme theory can be found in the survey article by Martin and Tanaka [MT09].

5.1 Association Scheme Theory Review

Definition 5.1 (Association schemes)**.** *An association scheme is a pair (X, R) where X is a finite set and $R \subseteq 2^{X \times X}$ is a collection of non-empty subsets of $X \times X$, called relations, satisfying the following properties.*

i. R is a partition of $X \times X$ into non-empty subsets.

ii. The diagonal relation $\mathcal{D}_X := \{(x, x) \mid x \in X\}$ is an element of R.

iii. For every $r \in R$, the transposed relation $r^\top := \{(y, x) \mid (x, y) \in r\}$ is an element of R.

iv. For every $r, s, t \in R$, there exists an intersection number $p^t_{rs} \in \mathbb{N}$ such that for every $(x, y) \in t$, we have

$$p^t_{rs} = |\{z \in X \mid (x, z) \in r \wedge (z, y) \in s\}|.$$

Furthermore, the association scheme S is called:

1. Commutative, if $p^t_{rs} = p^t_{sr}$ for every $r, s, t \in R$.

2. Symmetric, if $r^\top = r$ for every $r \in R$.

Fact 5.2. *A symmetric association scheme is also commutative.*

As the next definition describes, association schemes can be viewed as certain matrix algebras.

Definition 5.3 (Bose–Mesner algebra)**.** *Given an association scheme $S = (X, R)$, for each $r \in R$, let $D_r \in \mathbb{C}^{X\times X}$ be given by $D_r[x, y] := \mathbb{1}[(x, y) \in r]$. The Bose–Mesner algebra of S is the $\mathbb{C}$-algebra $\mathcal{A}_S$ generated by $\{D_r \mid r \in R\}$.*

The key observation underlying the above definition is that the intersection numbers p^t_{rs} in the definition of an association scheme guarantee that the linear span of $\{D_r \mid r \in R\}$ is closed under matrix multiplication and adjoints. Since

$$D_r D_s = \sum_{t\in R} p^t_{rs} D_t,$$

it follows that $(D_r)_{r\in R}$ is also a $\mathbb{C}$-vector space basis of $\mathcal{A}_S$. Furthermore, the above also implies that S is commutative if and only if its Bose–Mesner algebra is commutative. Moreover, S is symmetric if and only if every matrix in $\mathcal{A}_S$ is symmetric.

Fact 5.4. *The Bose–Mesner algebra $\mathcal{A}_S$ of a commutative association scheme S has a unique (up to permutation of its elements) $\mathbb{C}$-vector space basis of idempotent orthogonal matrices $(E_s)_{s\in R'}$, where $|R'| = |R|$. That is, we have $E_{s_1} E_{s_2} = \mathbb{1}[s_1 = s_2] E_{s_1}$ for every $s_1, s_2 \in R'$; namely, each E_s is the projection onto a maximal common eigenspace of the matrices $\{D_r \mid r \in R\}$.*

Since both $(D_r \mid r \in R)$ and $(E_s \mid s \in R')$ are bases of $\mathcal{A}_S$, each of their elements can be written as a linear combination of the elements of the other basis using the p and q-functions defined below.

Definition 5.5 (p-functions and q-functions)**.** *The p-functions $p_r \colon R' \to \mathbb{C}$ $(r \in R)$ and q-functions $q_s \colon R \to \mathbb{C}$ $(s \in R')$ of an association scheme S are the unique functions such that*

$$D_r = \sum_{s\in R'} p_r(s) E_s, \qquad\qquad E_s = \sum_{r\in R} q_s(r) D_r.$$

An important subclass of association schemes is that of Schurian schemes defined below, which arise by considering orbits of the diagonal action induced from a group action on the base set. The Bose–Mesner algebra of Schurian schemes is then precisely the algebra of matrices that are invariant under the natural conjugation action (see Fact 5.7 below). This makes Schurian schemes particularly useful in the study of semi-definite programs as for one such program P (see Fact 4.3).

Definition 5.6 (Schurian scheme)**.** *Let G be a group acting transitively on a finite set X. The Schurian scheme associated with this action is defined as $S := (X, (X \times X)/G)$, where $(X \times X)/G$ is the set of orbits of the natural diagonal action of G on $X \times X$ given by $\sigma \cdot (x, y) := (\sigma(x), \sigma(y))$ $(x, y \in X, \sigma \in G)$.*

Fact 5.7. *Let G be a group acting transitively on a finite set X. The Schurian scheme is an association scheme and its Bose–Mesner algebra is precisely the algebra of G-invariant matrices under the natural conjugation action of G on $\mathbb{C}^{X\times X}$ given by $A \cdot \sigma = P_\sigma^{-1} A P_\sigma$ $(A \in \mathbb{C}^{X\times X}, \sigma \in G)$, where $P_\sigma \in \mathbb{C}^{X\times X}$ is the permutation matrix given by $P_\sigma[x, y] := \mathbb{1}[x = \sigma(y)]$ $(x, y \in X)$.*

Definition 5.8 (Codes in an association scheme)**.** *A code in an association scheme $S = (X, R)$ is a non-empty subset $C \subseteq X$.*

The inner distribution of the code C is the function $a^C \colon R \to \mathbb{R}$ given by $a^C_r := |C^2 \cap r|/|C|$.

For a set $D \subseteq R$, we say that C is a D-code if $a^C_r = 0$ for every $r \in R \setminus D$.

Given an association scheme $S = (X, R)$, by letting $S' := (X, R')$, where $R' := \{r \cup r^\top \mid r \in R\}$, it is straightforward to check that S' is a symmetric association scheme and if C is a D-code in S, then it is also a D'-code in S', where $D' := \{r \cup r^\top \mid r \in D\}$. For this reason, when working with codes, we may suppose without loss of generality that the underlying association scheme is symmetric.

Definition 5.9 (Delsarte linear program)**.** *Given a set $D \subseteq R$, the Delsarte linear program associated with (S, D) is the program $\mathcal{L}_S(D)$ given by*

$$\begin{aligned}
\max \quad & \sum_{r \in R} a_r \\
\text{s.t.} \quad & a_{\mathcal{D}_X} = 1 && && (\textit{Normalization}) \\
& a_r = 0 && \forall r \in R \setminus D && (\textit{D-code constraints}) \\
& \sum_{r \in R} q_s(r) \cdot a_r \in \mathbb{R}_+ && \forall s \in R' && (\textit{MacWilliams inequalities}) \\
& a_r \in \mathbb{R}_+ && \forall r \in R && (\textit{Non-negativity}),
\end{aligned}$$

where the variables are $(a_r)_{r \in R}$.

It is clear that the inner distribution a^C of a D-code is a feasible solution of $\mathcal{L}_S(D)$, so the optimum value of $\mathcal{L}_S(D)$ is an upper bound on the size of D-codes (since $\sum_{r \in R} a_r^C = |C|$).

When the underlying scheme S is symmetric, one can use instead the *real Bose–Mesner algebra of S*, which is the $\mathbb{R}$-algebra generated by $\{D_r \mid r \in R\}$. All facts above remain true with $\mathbb{C}$ replaced with $\mathbb{R}$.

Definition 5.10 (Dual)**.** *Two association schemes $S = (X, R)$ and $\widehat{S} = (X, \widehat{R})$ over the same set X are said to be dual to each other if there exist bijections $f \colon R \to \widehat{R}'$ and $g \colon R' \to \widehat{R}$ such that for every $r \in R$ and every $s \in R'$, we have*

$$p_r(s) = q_{f(r)}(g(s)), \qquad\qquad q_s(r) = p_{g(s)}(f(r)).$$

In this case it is typical to identify R and R' with $\widehat{R}'$ and $\widehat{R}$, respectively through these bijections. An association scheme $S = (X, R)$ is self-dual when it is its own dual.

Definition 5.11 (Translation schemes)**.** *A translation scheme is an association scheme $S = (X, R)$ in which X is further equipped with an Abelian group structure and each relation $r \in R$ is an X-invariant set, i.e., for every $x, y, z \in X$, we have $(x, y) \in r \Leftrightarrow (z + x, z + y) \in r$.*

Equivalently, an association scheme $S = (X, R)$ where X has Abelian group structure is a translation scheme if and only if there exists a function $f_S \colon X \to R$ such that $(x, y) \in f_S(x - y)$ for every $x, y \in X$. This means that we can also alternatively view R as a partition of X rather than $X \times X$; the relations of the association scheme are defined by the "first row" of the matrix.

Fact 5.12. *Every translation scheme is commutative.*

Remark 5.13. *It is easy to see that the function f_S satisfies $f_S(-x) = f_S(x)^\top$ for every $x \in X$, hence S is symmetric if and only if the function f_S is even (i.e., $f_S(-x) = f_S(x)$ for every $x \in R$).*

Remark 5.14. *For translation schemes, the p and q-functions can be computed via Fourier analysis as follows. Let us fix an indexing of the characters $\chi_x \colon X \to \mathbb{C}$ of X by X so that $\chi_x(y) = \chi_y(x)$.*

Define the functions $\varphi_r \colon X \to \mathbb{C}$ ($r \in R$) by

$$\varphi_r(x) \coloneqq \sum_{y \in f_S^{-1}(r)} \overline{\chi_y(x)}.$$

The level sets of each φ_r induce a partition of X, so we let $R' \subseteq 2^X$ be the coarsest common refinement of them and let $f'_S \colon X \to R'$ be the unique function such that $x \in f'_S(x)$ for every $x \in X$. Define then the functions $\psi_s \colon X \to \mathbb{C}$ ($s \in R'$) by

$$\psi_s(y) \coloneqq \sum_{x \in s} \chi_x(y) = \sum_{x \in (f'_S)^{-1}(s)} \chi_x(y).$$

It is a standard fact of association scheme theory that for every $x \in X$, every $r \in R$ and every $s \in R'$, we have

$$\varphi_r(x) = p_r(f'_S(x)), \qquad \psi_s(x) = q_s(f_S(x)).$$

It is also straightforward to see that f'_S induces a translation scheme structure $\widehat{S} = (X, \widehat{R})$ on X where $\widehat{R} \coloneqq \{r_s \mid s \in R'\}$ for the relations $r_s \coloneqq \{(x, y) \in X \times X \mid f'_S(x - y) = s\}$ and S and $\widehat{S}$ are dual of each other, and as such we typically identify R' with $\widehat{R}$ with $s \mapsto r_s$.

Definition 5.15 (Additive codes and annihilator codes)**.** *A code C in a translation scheme S is called additive if it is a subgroup of X. Trivially, an additive code C is a D-code if and only if $f_S(C) \subseteq D$.*

Given an additive code C in S, the annihilator code of C is

$$C^\circ \coloneqq \{y \in X \mid \chi_y(x) = 1\}.$$

It is straightforward to see that C° is additive and $(C^\circ)^\circ = C$ (as long as C is additive). It is more natural to see the annihilator code as a code in the dual scheme $\widehat{S}$, as the (generalized) MacWilliams identities say that the inner distribution of C° in $\widehat{S}$ can be retrieved from the inner distribution of C in S as follows.

Theorem 5.16 (Generalized MacWilliams identities)**.** *For a translation scheme $S = (X, R)$ with dual scheme $\widehat{S} = (X, \widehat{R})$ and an additive code C in S, we have*

$$a_s^{C^\circ} = \frac{1}{|C|} \sum_{r \in R} q_s(r) a_r^C,$$

for all $s \in \widehat{R}$,

Definition 5.17 (Tensor product schemes)**.** *Given two schemes $S_1 = (X_1, R_1)$ and $S_2 = (X_2, R_2)$, their tensor product is the association scheme $S_1 \otimes S_2 \coloneqq (X_1 \times X_2, R_1 \otimes R_2)$, where $R_1 \otimes R_2 \coloneqq \{r_1 \otimes r_2 \mid r_1 \in R_1 \wedge r_2 \in R_2\}$ for the relations*

$$r_1 \otimes r_2 \coloneqq \{((x_1, x_2), (y_1, y_2)) \in (X_1 \times X_2) \times (X_1 \times X_2) \mid (x_1, y_1) \in r_1 \wedge (x_2, y_2) \in r_2\}.$$

For $\ell \in \mathbb{N}_+$, the ℓth tensor power of the association scheme S is defined as

$$S^\ell \coloneqq \underbrace{S \otimes \cdots \otimes S}_{\ell \text{ times}}.$$

It is straightforward to check that $S_1 \otimes S_2$ is an association scheme that inherits the properties of S_1 and S_2 in the sense that if both S_1 and S_2 are commutative (resp., symmetric, translation), then $S_1 \otimes S_2$ is so (in the case of translation scheme, the group structure in $X_1 \times X_2$ is the direct product group). Furthermore, if C_i is a D_i-code in S_i ($i \in [2]$), then $C_1 \times C_2$ is a $D_1 \otimes D_2$-code in $S_1 \otimes S_2$, where

$$D_1 \otimes D_2 \coloneqq \{r_1 \otimes r_2 \mid r_1 \in D_1 \wedge r_2 \in D_2\}.$$

Definition 5.18 (Refinement of a scheme)**.** *A refinement of an association scheme $S = (X, R)$ is an association scheme $S_2 = (X, R_2)$ over the same underlying set X such that each $r \in R$ is a union of elements of R_2.*

Trivially, a D-code in S is a D'-code in S', where

$$D' = \{r' \in R_2 \mid \exists r \in D, r' \subseteq r\}.$$

Example 5.19 (Weak Hamming scheme)**.** *Given a non-trivial finite Abelian group G and $n \in \mathbb{N}_+$, the (weak) Hamming scheme of order n over G is the translation scheme $\mathbb{H}_n(G) \coloneqq (G^n, R)$, where $R \coloneqq \{r_i \mid i \in \{0, \dots, n\}\}$ for*

$$r_i \coloneqq \{(x, y) \mid \Delta(x, y) = i\},$$

where $\Delta(x, y) \coloneqq |\{j \in [n] \mid x_j \neq y_j\}|$ is the Hamming distance between x and y. It is easy to see that $\mathbb{H}_n(G)$ is a translation scheme over the direct product group G^n in which $f_{\mathbb{H}_n(G)}(x) = r_{\Delta(x,0)}$ for every $x \in G^n$. In fact, $\mathbb{H}_n(G)$ is self-dual and its p and q functions are the Krawtchouk polynomials:

$$p_{r_i}(r_j) = q_{r_i}(r_j) = \sum_{t=0}^{j} (-1)^t (|G| - 1)^{i-t} \binom{j}{t} \binom{n-j}{i-t}.$$

We use the notation $\mathbb{H}_n \coloneqq \mathbb{H}_n(\mathbb{F}_2)$, when the underlying group is the field with two elements. Under this notation, a binary code of blocklength n and distance d is simply a D_d-code in $\mathbb{H}_n$, where $D_d \coloneqq \{r_0, r_d, r_{d+1}, \dots, r_n\}$.

Alternatively, the weak Hamming scheme can be seen as a Schurian scheme as follows. Consider the natural right action of the symmetric group S_n on n letters on G^n given by $(x \cdot \sigma)_i \coloneqq x_{\sigma(i)}$ ($x \in G^n$, $\sigma \in S_n$, $i \in [n]$) and the natural left action of the symmetric group S_{G^n} on G^n. These actions together induce an action of a semidirect product $S_{G^n} \rtimes S_n$ on G^n whose associated Schurian scheme is precisely $\mathbb{H}_n(G)$.

Example 5.20 (Strong Hamming scheme)**.** *Given a non-trivial finite Abelian group G and $n \in \mathbb{N}_+$, the strong Hamming scheme of order n over G is the translation scheme $\mathbb{H}_n^*(G) \coloneqq (G^n, R)$, where $R \coloneqq \{r_h \mid h \in \{0, 1, \dots, n\}^G\} \setminus \{\varnothing\}$, for*

$$r_h \coloneqq \{(x, y) \mid \forall g \in G, |x - y|_g = h(g)\},$$

where

$$|z|_g \coloneqq |\{i \in [n] \mid z_i = g\}|.$$

It is easy to see that $\mathbb{H}_n^*(G)$ *is a translation scheme that is a refinement of* $\mathbb{H}_n(G)$ *and for every* $x \in G^n$*, we have* $f_{\mathbb{H}_n^*(G)}(x) = r_{h_x}$ *for* $h_x(g) := |x|_g$*. In fact,* $\mathbb{H}_n^*(G)$ *is self-dual and its* p *and* q *functions are given by*

$$p_{r_{h_1}}(r_{h_2}) = q_{r_{h_1}}(r_{h_2}) = \sum_{F \in \mathcal{F}} \left(\prod_{g_1 \in G} \frac{h_1(g_1)!}{\prod_{g_2 \in G} F(g_1, g_2)!} \right) \prod_{g_1, g_2 \in G} \chi_{g_1}(g_2)^{F(g_1, g_2)},$$

where $\mathcal{F}$ *is the set of all functions* $F \colon G \times G \to \{0, \dots, n\}$ *such that*

$$\sum_{g' \in G} F(g, g') = h_1(g), \qquad \sum_{g' \in G} F(g', g) = h_2(g),$$

for every $g \in G$*.*

Alternatively, the strong Hamming scheme can be seen as a Schurian scheme as follows. Consider the natural right action of the symmetric group S_n *on* n *letters on* G^n *given by* $(x \cdot \sigma)_i := x_{\sigma(i)}$ *(*$x \in G^n$*,* $\sigma \in S_n$*,* $i \in [n]$*) and the natural translation action of the product group* G^n *on itself. These actions together induce an action of a semidirect product* $G^n \rtimes S_n$ *on* G^n *whose associated Schurian scheme is precisely* $\mathbb{H}_n^*(G)$*.*

While for binary alphabets, the strong and weak Hamming scheme obviously coincide (i.e., $\mathbb{H}_n^*(\mathbb{F}_2) = \mathbb{H}_n(\mathbb{F}_2)$), for larger alphabets this is not the case.

We finish this section recalling the connection of the Delsarte linear program with the modified Lovász ϑ'-function from graph theory (see Definition 4.11).

Definition 5.21. *Given a commutative association scheme* $S = (X, R)$ *and* $D \subseteq R$ *with* $\mathcal{D}_X \in D$*, the graph* $G_S(D)$ *is defined by*

$$V(G_S(D)) := X,$$
$$E(G_S(D)) := \left\{ \{x, y\} \in \binom{X}{2} \;\middle|\; \exists r \in R \setminus D, (x, y) \in r \right\}.$$

Under this definition, a D-code on S is simply an independent set in the graph $G_S(D)$. The next theorem by Schrijver connects the Delsarte linear program $\mathcal{L}_S(D)$ to the semi-definite program $\mathcal{S}(G_S(D))$.

Theorem 5.22 (Schrijver [Sch79])**.** *Let* $S = (X, R)$ *be a commutative association scheme and let* $D \subseteq R$ *with* $\mathcal{D}_X \in D$*. Then* $\vartheta'(G_S(D))$ *is equal to the optimum value of the Delsarte linear program* $\mathcal{L}_S(D)$*.*

5.2 Natural Refinements of Translation Schemes

In this section, we show how our construction generalizes nicely to translation schemes with an underlying left module structure over some ring. This can be applied to any translation scheme by recalling that any Abelian group is naturally a $\mathbb{Z}$-module, but sometimes it is more interesting to use a different module structure (e.g., a vector space over a finite field).

Definition 5.23 (Association scheme automorphism)**.** *An automorphism of an association scheme* $S = (X, R)$ *is a bijection* $f \colon X \to X$ *that fixes each* $r \in R$ *as a set, that is, we have* $\{(f(x), f(y)) \mid (x, y) \in r\} = r$*. The group of automorphisms of* S *is denoted* $\mathrm{Aut}(S)$*.*

Definition 5.24. *Let $S = (X, R)$ be a translation scheme and let $f_S \colon X \to R$ be the unique function such that $(x, y) \in f_S(x - y)$ for every $(x, y) \in X \times X$. Given a ring K, let us further assume that X is equipped with a left K-module structure extending the Abelian group structure and let $\mathrm{Aut}_K(S) \leqslant \mathrm{Aut}(S)$ be the subgroup of automorphisms of the association scheme S that are also left K-module automorphisms of X.*

A code C in S is called K-linear if it is both additive and K-invariant in the sense that $kx \in C$ for every $k \in K$ and every $x \in C$.

Let $\ell \in \mathbb{N}_+$ and let $T \subseteq K^\ell$ be a collection of ℓ-tuples of K. Define the function $f_{S,T} \colon X^\ell \to R^T$ by

$$f_{S,T}(x)(k) \coloneqq f_S\left(\sum_{i=1}^{\ell} k_i x_i\right) \qquad (x \in X^\ell, k \in T).$$

We say that $f_{S,T}$ factors through types of S if for every $x, y \in X^\ell$, we have $f_{S,T}(x) = f_{S,T}(y)$ if and only if there exists $\sigma \in \mathrm{Aut}_K(S)$ such that $\sigma(x_i) = y_i$ for every $i \in [\ell]$.

Similarly to symmetric difference configurations of Definition 3.1, the function $f_{S,T}$ captures information of the value of f_S in K-linear combinations of ℓ-tuples of elements of X using coefficients in $T \subseteq K^\ell$. The definition of factoring through types then requires that this information is enough to determine the orbit[2] of a tuple $x \in X^\ell$ under the natural diagonal action of $\mathrm{Aut}_K(S)$.

Remark 5.25. *It is straightforward to see that f_S is $\mathrm{Aut}_K(S)$-invariant in the sense that $f_S \circ \sigma = f_S$ for every $\sigma \in \mathrm{Aut}_K(S)$.*

This in particular implies that in the definition of $f_{S,T}$ factoring through types, the backward implication always holds: if $x, y \in X^\ell$ and $\sigma \in \mathrm{Aut}_K(S)$ are such that $\sigma(x_i) = y_i$ for every $i \in [\ell]$, then for every $k \in T$ we have

$$f_{S,T}(y)(k) = f_S\left(\sum_{i=1}^{\ell} k_i \sigma(x_i)\right) = f_S\left(\sigma\left(\sum_{i=1}^{\ell} k_i x_i\right)\right) = f_S\left(\sum_{i=1}^{\ell} k_i x_i\right) = f_{S,T}(x)(k).$$

The next definition generalizes the construction of Section 3.

Definition 5.26. *Let $S = (X, R)$ be a translation scheme over a left K-module X, let $\ell \in \mathbb{N}_+$, let $T \subseteq K^\ell$ be such that $e_i \in T$ for every $i \in [\ell]$, where $(e_i)_j \coloneqq \mathbb{1}[i = j]$ and suppose $f_{S,T}$ factors through types.*

The T-refined ℓth tensor power of S is the translation scheme $S^{\ell,T} = (X^\ell, R^{\ell,T})$ is defined by letting

$$R^{\ell,T} \coloneqq \{r_h \mid h \in R^T\} \setminus \{\varnothing\},$$

where

$$r_h \coloneqq \{(x, y) \in X^\ell \times X^\ell \mid f_{S,T}(x - y) = h\}$$

for each function $h \colon T \to R$ and X^ℓ is equipped with the direct product left K-module structure. Theorem 5.27 below shows that $S^{\ell,T}$ is indeed a translation scheme.

[2]This is also the reason behind the name "factors through types": in model theory, two elements of a *finite* model have the same type if and only if they are in the same orbit under the action of the automorphism group.

Before we show that $S^{\ell,T}$ is indeed a translation scheme, two particular choices of (K,T) deserve special attention.

i. When $K=\mathbb{Z}$ and $T=\{e_i \mid i\in[\ell]\}$, then $S^{\ell,T}$ is just the ℓth tensor power S^ℓ.

ii. When $T=K^\ell$, then $f_{S,T}$ encodes the *complete K-linear configuration* of x as it is able to determine the value of f_S in any K-linear combination of $x_1,\dots,x_\ell$. This will be particularly useful when K is a (finite) field (and thus X is a K-vector space).

Theorem 5.27. *Let $S=(X,R)$ be a translation scheme over a left K-module X, let $\ell\in\mathbb{N}_+$, let $T\subseteq K^\ell$ be such that $e_i\in T$ for every $i\in[\ell]$, where $(e_i)_j:=\mathbb{1}[i=j]$ and suppose $f_{S,T}$ factors through types.*

Then the following hold.

1. *$S^{\ell,T}:=(X^\ell,R^{\ell,T})$ is a translation scheme over the direct product group X^ℓ that refines the tensor power S^ℓ.*
2. *If S is symmetric, then so is $S^{\ell,T}$.*
3. *If C is a K-linear D-code in S, then C^ℓ is a K-linear $D^{\ell,T}$-code in $S^{\ell,T}$, where*

$$D^{\ell,T}:=\{r_h \mid h\in R^T\wedge \operatorname{im}(h)\subseteq D\}\setminus\{\varnothing\}.$$

Proof. We start proving item (1).

It is obvious that $R^{\ell,T}$ forms a partition of $X^\ell\times X^\ell$ into non-empty subsets.

Note also that if $(x,y)\in X^\ell\times X^\ell$ are such that $f_{S,T}(x-y)(k)=\mathcal{D}_X$ for every $k\in T$, then since $e_i\in T$, we get $x_i=y_i$, thus $x=y$. Since we also have $f_{S,T}(0)(k)=f_S(0)=\mathcal{D}_X$ for every $k\in T$, it follows that for the function $T\to R$ that is constant equal to $\mathcal{D}_X$ we have $r_{\mathcal{D}_X}=\mathcal{D}_{X^\ell}$.

It is also easy to see that for $h\colon T\to R$, by letting $h^\top\colon T\to R$ be given by $h^\top(k):=h(k)^\top$, we have $r_h^\top=r_{h^\top}$.

Note further that for each $r_1,r_2,\dots,r_\ell\in R$, we have

$$r_1\otimes\cdots\otimes r_\ell=\bigcup\{r_h \mid h\in R^T\wedge\forall i\in[\ell],h(e_i)=r_i\}.$$

It remains only to show that the existence of the intersection numbers for $S^{\ell,T}$.

For every $h_1,h_2\colon T\to R$ and every $x,y\in X^\ell$, let

$$N_{h_1,h_2}(x,y):=|\{z\in X^\ell \mid f_{S,T}(x-z)=h_1\wedge f_{S,T}(z-y)=h_2\}|.$$

It is sufficient to show that if $x,y,x',y'\in X^\ell$ are such that $f_{S,T}(x-y)=f_{S,T}(x'-y')$, then $N_{h_1,h_2}(x,y)=N_{h_1,h_2}(x',y')$.

Since $f_{S,T}$ factors through types of S, there exists $\sigma \in \mathrm{Aut}_K(S)$ such that $\sigma(x_i - y_i) = x'_i - y'_i$ for every $i \in [\ell]$. Then we have

$$\begin{aligned}
N_{h_1,h_2}(x',y') &= |\{z \in X^\ell \mid f_{S,T}(x'-z) = h_1 \wedge f_{S,T}(z-y') = h_2\}| \\
&= \left|\left\{u \in X^\ell \;\middle|\; \forall k \in T, \left(f_S\left(\sum_{i=1}^{\ell} k_i(x'_i - y'_i - u_i)\right) = h_1(k) \wedge f_S\left(\sum_{i=1}^{\ell} k_i u_i\right) = h_2(k)\right)\right\}\right| \\
&= \left|\left\{u \in X^\ell \;\middle|\; \forall k \in T, \left(f_S\left(\sum_{i=1}^{\ell} k_i(\sigma(x_i - y_i) - u_i)\right) = h_1(k) \wedge f_S\left(\sum_{i=1}^{\ell} k_i u_i\right) = h_2(k)\right)\right\}\right| \\
&= \left|\left\{u \in X^\ell \;\middle|\; \forall k \in T, \left(f_S\left(\sum_{i=1}^{\ell} k_i(x_i - y_i - \sigma^{-1}(u_i))\right) = h_1(k) \wedge f_S\left(\sum_{i=1}^{\ell} k_i \cdot \sigma^{-1}(u_i)\right) = h_2(k)\right)\right\}\right| \\
&= \left|\left\{w \in X^\ell \;\middle|\; \forall k \in T, \left(f_S\left(\sum_{i=1}^{\ell} k_i(x_i - w_i)\right) = h_1(k) \wedge f_S\left(\sum_{i=1}^{\ell} k_i(w_i - y_i)\right) = h_2(k)\right)\right\}\right| \\
&= N_{h_1,h_2}(x,y),
\end{aligned}$$

where the second equality follows from the substitution $u_i := z_i - y'_i$, the fourth equality follows since σ is a left K-module automorphism of X and f_S is σ-invariant (see Remark 5.25) and the fifth equality follows from the substitution $w_i := \sigma^{-1}(u_i) + y_i$.

For item (2), since a translation scheme S is symmetric if and only if the function f_S is even (see Remark 5.13), the fact that S is symmetric implies f_S is even, hence $f_{S,T}$ is also even and thus $S^{\ell,T}$ is symmetric (as $f_{S^{\ell,T}}(x) = r_{f_{S,T}(x)}$).

For item (3), it is obvious that C^ℓ is both a subgroup of X^ℓ and K-invariant. Let $x \in C^\ell$ and note that since C is K-linear, for every $k \in T$, we have $\sum_{i=1}^{\ell} k_i x_i \in C$, so $f_{S,T}(x)(k) = f_S(\sum_{i=1}^{\ell} k_i x_i) \in D$ and thus

$$f_{S^{\ell,T}}(C^\ell) = \{r_{f_{S,T}(x)} \mid x \in C^\ell\} \subseteq D,$$

hence C^ℓ is a K-linear $D^{\ell,T}$-code. ■

Note that if the underlying translation scheme $S = (X, R)$ is a Schurian scheme associated to a group action of a semidirect product $X \rtimes G$ in X such as in the (weak or strong) Hamming scheme (see Examples 5.19 and 5.20), then we could easily produce a Schurian translation scheme refining the ℓth tensor power by considering the action of a semidirect product $X^\ell \rtimes G$ on X^ℓ obtained by considering the product action of X^ℓ and the diagonal action of G. However, even in the Schurian case, the true value of Theorem 5.27 above lies in two facts:

i. The relation of $(x, y) \in X^\ell \times X^\ell$ is determined by the value of $f_{S,T}(x - y)$, that is, the value of f_S in K-linear combinations of $(x - y)$ using tuples in T.

ii. If C is a K-linear D-code in S, we can deduce "extra" restrictions of the code C^ℓ in $S^{\ell,T}$ besides the ones that follow from the tensor power. More specifically, it is trivial that C^ℓ is a $D^{\otimes\ell}$-code in the tensor power S^ℓ, which in turn implies that it is a $\widehat{D}$-code in $S^{\ell,T}$, where

$$\begin{aligned}
\widehat{D} &:= \{r_h \in R^{\ell,T} \mid \exists r \in R^{\otimes\ell}, r_h \subseteq r\} \\
&= \{r_h \mid h \in R^T \wedge \forall i \in [\ell], h(e_i) \in D\}.
\end{aligned}$$

However, Theorem 5.27 says that we further have $h(k) \in D$ for *every* $k \in T$ (not only for the e_i).

Our next objective is to show that under mild assumptions on the structure of the left K-module G, for the weak and strong Hamming schemes $\mathbb{H}_n(G)$ and $\mathbb{H}_n^*(G)$ of Examples 5.19 and 5.20, the functions $f_{\mathbb{H}_n(G),K^\ell}$ and $f_{\mathbb{H}_n^*(G),K^\ell}$ factor through types when G^n is equipped with the direct product left K-module structure. A particular case when all such mild assumptions hold is when $G = K = \mathbb{F}$ for some finite field $\mathbb{F}$.

Remark 5.28. *Once we prove that the function $f_{\mathbb{H}_n(\mathbb{F}_2),\mathbb{F}_2^\ell}$ factors through types of the weak Hamming scheme $\mathbb{H}_n(\mathbb{F}_2)$ (Corollary 5.46 below), the hierarchy of linear programs presented in Section 3 can be retrieved as $\mathrm{KrawtchoukLP}_{\mathrm{Lin}}(n,d,\ell) = \mathcal{L}_{S^{\ell,T}}(D_d^{\ell,T})$ for $S = \mathbb{H}_n(\mathbb{F}_2)$, $T = \mathbb{F}_2^\ell$ and $D_d \coloneqq \{r_0, r_d, r_{d+1}, \dots, r_n\}$.*

Analogously, the hierarchy for non-linear codes can be retrieved as $\mathrm{KrawtchoukLP}(n,d,\ell) = \mathcal{L}_{S^{\ell,T}}(\widehat{D}_d^\ell)$ using instead the weaker restriction set

$$\widehat{D}_d^\ell \coloneqq \{r \in R^{\ell,T} \mid \exists r' \in D_d^{\otimes \ell}, r \subseteq r'\}.$$

These can also be retrieved using the strong Hamming scheme $\mathbb{H}_n^*(\mathbb{F}_2)$ instead (see Corollary 5.35 below) as for the binary case we have $\mathbb{H}_n^*(\mathbb{F}_2) = \mathbb{H}_n(\mathbb{F}_2)$.

However, the same corollaries apply for the more general case of a (not necessarily binary) finite field $\mathbb{F}$, in which the weak and strong Hamming schemes are different. In this case, we define

$$\mathrm{KrawtchoukLP}_{\mathrm{Lin}}^{\mathbb{F}}(n,d,\ell) \coloneqq \mathcal{L}_{S^{\ell,T}}(D_d^{\ell,T})$$

using $S = \mathbb{H}_n(\mathbb{F})$, $T = \mathbb{F}^\ell$ and $D_d \coloneqq \{r_0, r_d, r_{d+1}, \dots, r_n\}$.

We can also define $\mathrm{KrawtchoukLP}^{\mathbb{F}}(n,d,\ell)$ analogously for arbitrary finite fields, but as we will see in Proposition 6.5, these hierarchies for non-linear codes collapse and yield the same bound as the usual Delsarte linear program.

Before we start with the case of the strong Hamming scheme $\mathbb{H}_n^*(G)$, let us prove a few lemmas.

Lemma 5.29. *Let K be a finite ring, let G be a finite simple left K-module and let $\chi\colon G \to \mathbb{C}$ be a non-trivial character of G. Then*

$$\sum_{k \in K} \chi(kg) = |K| \cdot \mathbb{1}[g = 0]$$

for every $g \in G$.

Proof. If $g = 0$, then $\chi(kg) = 1$ for every $k \in K$, thus $\sum_{k\in K} \chi(kg) = |K|$.

On the other hand, if $g \neq 0$, then we must have $Kg = G$ as Kg is a non-trivial left K-submodule of G and G is simple. Note also that for $k_1, k_2 \in K$, we have $\chi(k_1 g) = \chi(k_2 g)$ if and only if $k_1 - k_2 \in H$, where

$$H \coloneqq \{k \in K \mid \chi(kg) = 1\}.$$

Since H is a subgroup of K, each level set of $k \mapsto \chi(kg)$ is a coset of H, so they must all have the same size, namely $|K|/|H|$, so we get

$$\sum_{k \in K} \chi(kg) = \frac{|K|}{|H|} \cdot \frac{|\mathrm{im}(\chi)|}{|G|} \sum_{g' \in G} \chi(g') = 0,$$

where the last equality follows since χ is a non-trivial character (so it is orthogonal to the trivial character). ■

The next lemma says that the joint distribution of ℓ random variables with values in a finite simple left K-module (for a finite ring K) can be recovered from the (individual) distributions of all K-linear combinations of them.

Lemma 5.30. *Let K be a finite ring, let G be a finite simple left K-module and let $\chi\colon G \to \mathbb{C}$ be a non-trivial character of G. Suppose further that $\boldsymbol{X}$ is a random variable with values in G^ℓ for some $\ell \in \mathbb{N}_+$ and for every $k \in K^\ell$, let $\boldsymbol{Y}_k \coloneqq \sum_{i=1}^{\ell} k_i \boldsymbol{X}_i$. Then*

$$\mathbb{P}[\boldsymbol{X} = x] = \frac{1}{|K^\ell|} \sum_{\substack{k \in K^\ell \\ y \in G}} \chi\left(\sum_{i=1}^{\ell} k_i x_i - y\right) \cdot \mathbb{P}[\boldsymbol{Y}_k = y]$$

for every $x \in G^\ell$.

Proof. First note that for every $k \in K^\ell$ and every $y \in G$, we have

$$\mathbb{P}[\boldsymbol{Y}_k = y] = \sum_{\substack{z \in G^\ell \\ \sum_{i=1}^{\ell} k_i z_i = y}} \mathbb{P}[\boldsymbol{X} = z],$$

which implies that

$$\begin{aligned}
\frac{1}{|K^\ell|} \sum_{\substack{k \in K^\ell \\ y \in G}} \chi\left(\sum_{i=1}^{\ell} k_i x_i - y\right) \cdot \mathbb{P}[\boldsymbol{Y}_k = y] &= \frac{1}{|K^\ell|} \sum_{z \in G^\ell} \mathbb{P}[\boldsymbol{X} = z] \cdot \sum_{\substack{k \in K^\ell \\ y \in G \\ \sum_{i=1}^{\ell} k_i z_i = y}} \chi\left(\sum_{i=1}^{\ell} k_i x_i - y\right) \\
&= \frac{1}{|K^\ell|} \sum_{z \in G^\ell} \mathbb{P}[\boldsymbol{X} = z] \cdot \sum_{k \in K^\ell} \chi\left(\sum_{i=1}^{\ell} k_i (x_i - z_i)\right) \\
&= \mathbb{P}[\boldsymbol{X} = x] + \frac{1}{|K^\ell|} \sum_{\substack{z \in G^\ell \\ z \neq x}} \mathbb{P}[\boldsymbol{X} = z] \cdot \sum_{k \in K^\ell} \chi\left(\sum_{i=1}^{\ell} k_i (x_i - z_i)\right).
\end{aligned} \tag{8}$$

To complete the proof, it is sufficient to show that the inner sum of the second term in (8) is zero. But note that

$$\sum_{k \in K^\ell} \chi\left(\sum_{i=1}^{\ell} k_i (x_i - z_i)\right) = \prod_{i=1}^{\ell} \sum_{k \in K} \chi(k(x_i - z_i))$$

and since $x_i \neq z_i$ for at least one $i \in [\ell]$, from Lemma 5.29, the above is zero as desired. ■

Let us also recall one standard fact from algebra.

Lemma 5.31. *Let K be a ring and G be a left K-module. If G is both finite and faithful, that is, the annihilator*

$$\mathrm{Ann}_K(G) \coloneqq \{k \in K \mid \forall g \in G, kg = 0\}$$

is trivial (i.e., $\mathrm{Ann}_K(G) = \{0\}$). Then K is finite.

Proof. Each element $k \in K$ induces a left K-module endomorphism $f_k \colon G \to G$ of G given by $f_k(g) = k \cdot g$. Note that for $k_1, k_2 \in K$, we have $f_{k_1} = f_{k_2}$ if and only if $k_1 - k_2 \in \mathrm{Ann}_K(G)$, so since G is faithful, all f_k must be different. If G is finite, it has only finitely many endomorphisms, so K must also be finite. ■

We can now show that for the strong Hamming scheme $\mathbb{H}_n^*(G)$ over a finite simple left K-module G, the function $f_{\mathbb{H}_n^*(G),K^\ell}$ factors through types. We recall that when K is commutative, the simplicity condition reduces to saying that there is a maximal ideal I of K such that $G \cong K/I$ as a K-module, that is, it is a 1-dimensional vector space over the (necessarily finite) field $\mathbb{F} \coloneqq K/I$ (note that K itself does not need to be a field, e.g., $K = \mathbb{Z}$ and $G = \mathbb{Z}_p$ for some prime p); in other words, all cases when K is commutative and G is simple are indirectly captured by the usual case $K = G = \mathbb{F}$ for some finite field $\mathbb{F}$.

Proposition 5.32. *Let K be a ring, let G be a finite simple left K-module and let $n, \ell \in \mathbb{N}_+$. Consider the strong Hamming scheme $\mathbb{H}_n^*(G)$ of order n over G equipped with the direct product left K-module structure on G^n. Then $f_{\mathbb{H}_n^*(G),K^\ell}$ factors through types of $\mathbb{H}_n^*(G)$.*

Proof. First, recall that the strong Hamming scheme relations are given by

$$r_h \coloneqq \{(x, y) \mid \forall g \in G, |x - y|_g = h(g)\},$$

for $h \in \{0, 1, \ldots, n\}^G\}$ such that r_h is non-empty, where $|z|_g$ is the number of positions $i \in [n]$ such that $z_i = g$. In this proof we will abuse notation and write $f_{\mathbb{H}_n^*(G)}(x) = h$ in place of $f_{\mathbb{H}_n^*(G)}(x) = r_h$, that is, we will view $f_{\mathbb{H}_n^*(G)}$ as a function with values in $\{0, 1, \ldots, n\}^G$ rather than in $R \coloneqq \{r_h \mid h \in \{0, 1, \ldots, n\}^G\} \setminus \{\varnothing\}$. Accordingly, we will also view $f_{\mathbb{H}_n^*(G),K^\ell}$ as a function with values in $(\{0, 1, \ldots, n\}^G)^{K^\ell}$ rather than in R^{K^ℓ}.

First, we claim that it is enough to prove the case when K is finite. Indeed, recall that the annihilator $\mathrm{Ann}_K(G)$ of G in K is a two-sided ideal of K and the left K-module structure on G induces a natural left $K/\mathrm{Ann}_K(G)$-module structure given by $(k + \mathrm{Ann}_K(G)) \cdot g \coloneqq k \cdot g + \mathrm{Ann}_K(G)$ ($k \in K$, $g \in G$). Note also that for every $x \in (G^n)^\ell$ and every $k \in K^\ell$, we have

$$f_{\mathbb{H}_n^*(G),(K/\mathrm{Ann}_K(G))^\ell}(x)\big((k_i + \mathrm{Ann}_K(G) \mid i \in [\ell])\big) = f_{\mathbb{H}_n^*(G)}\left(\sum_{i=1}^n k_i \cdot x_i\right) = f_{\mathbb{H}_n^*(G),K^\ell}(x)(k),$$

so if $f_{\mathbb{H}_n^*(G),(K/\mathrm{Ann}_K(G))^\ell}$ factors through types, then $f_{\mathbb{H}_n^*(G),K^\ell}$ also does so. From Lemma 5.31, $K/\mathrm{Ann}_K(G)$ must be finite as G is a finite faithful left $K/\mathrm{Ann}_K(G)$-module, completing our reduction.

Let us now prove the case when K is finite.

First, note that $\mathrm{Aut}_K(\mathbb{H}_n(G))$ contains a subgroup isomorphic to the symmetric group S_n on n letters[3]; namely, the natural right action of S_n on G^n given by $(x \cdot \sigma)_i := x_{\sigma(i)}$ ($x \in G^n$, $\sigma \in S_n$, $i \in [n]$) is free and preserves the left K-module structure of G^n and every relation $r_h \in R$ of $\mathbb{H}_n^*(G)$ is invariant under this action and thus this action induces a subgroup of $\mathrm{Aut}_K(\mathbb{H}_n^*(G))$ isomorphic to S_n where $\sigma \in S_n$ corresponds to the automorphism $F_\sigma \colon x \mapsto x \cdot \sigma$.

Given a point $z \in (G^n)^\ell$, let $\boldsymbol{X}^z$ be the random variable with values in G^ℓ defined by

$$\boldsymbol{X}^z_j := (z_j)_{\boldsymbol{i}} \qquad (j \in [\ell]),$$

where $\boldsymbol{i}$ is picked uniformly at random in $[n]$, that is, $\boldsymbol{X}^z_j$ is the value of the jth word z_j at the (uniformly at random) position $\boldsymbol{i}$. Note that we use the same $\boldsymbol{i}$ for all values of $j \in [\ell]$, so the coordinates of $\boldsymbol{X}$ are not necessarily independent.

For every $k \in K^\ell$, let also $\boldsymbol{Y}^z_k := \sum_{j=1}^\ell k_j \boldsymbol{X}^z_j$ and note that for every $g \in G$, we have

$$\mathbb{P}[\boldsymbol{Y}^z_k = g] = \frac{\left|\left\{ i \in [n] \;\middle|\; \sum_{j=1}^\ell k_j (z_j)_i = g \right\}\right|}{n} = \frac{f_{\mathbb{H}_n^*(G),K^n}(z)(k)}{n}.$$

By Lemma 5.30, the distribution of $\boldsymbol{Y}^z$ completely determines the distribution of $\boldsymbol{X}^z$. This means that if $x, y \in (G^n)^\ell$ are such that $f_{\mathbb{H}_n^*(G),K^\ell}(x) = f_{\mathbb{H}_n^*(G),K^\ell}(y)$, then $\boldsymbol{X}^x$ has the same distribution as $\boldsymbol{X}^y$ and thus there exists a permutation $\sigma \in S_n$ such that for every $i \in [n]$ and every $j \in [\ell]$, we have $(x_j)_{\sigma(i)} = (y_j)_i$, that is, for the automorphism $F_\sigma \in \mathrm{Aut}_K(\mathbb{H}_n^*(G))$, we have $F_\sigma(x_j) = y_j$ for every $j \in [\ell]$, so $f_{\mathbb{H}_n^*(G),K^\ell}$ factors through types. ∎

The next example shows that the simplicity assumption in Proposition 5.32 is necessary.

Example 5.33. *Consider the finite left $\mathbb{F}_2$-module $\mathbb{F}_2^2$, let $n := 4$ and $\ell := 2$ and consider the elements $x, y \in ((\mathbb{F}_2^2)^n)^\ell$ defined as*

$$\begin{aligned} y_1 := x_1 &:= \big((0,0),(0,1),(1,0),(1,1)\big), \\ x_2 &:= \big((0,1),(1,0),(0,0),(1,1)\big), \\ y_2 &:= \big((1,0),(0,1),(1,1),(0,0)\big) \end{aligned}$$

and note that

$$\begin{aligned} x_1 + x_2 &= \big((0,1),(1,1),(1,0),(0,0)\big), \\ y_1 + y_2 &= \big((1,0),(0,0),(0,1),(1,1)\big), \end{aligned}$$

If we think of the function $f_{\mathbb{H}_4^*(\mathbb{F}_2^2)}$ as taking values in $\{0,1,2,3,4\}^{\mathbb{F}_2^2}$ as we did in the proof of Proposition 5.32 and the function $f_{\mathbb{H}_4^*(\mathbb{F}_2^2),\mathbb{F}_2^2}$ as taking values in $(\{0,1,2,3,4\}^{\mathbb{F}_2^2})^{\mathbb{F}_2^2}$, then for every $g \in \mathbb{F}_2^2$, we have

$$\begin{aligned} f_{\mathbb{H}_4^*(\mathbb{F}_2^2),\mathbb{F}_2^2}(x)(0,0)(g) &= f_{\mathbb{H}_4^*(\mathbb{F}_2^2),\mathbb{F}_2^2}(y)(0,0)(g) = 4 \cdot \mathbb{1}[g = 0], \\ f_{\mathbb{H}_4^*(\mathbb{F}_2^2),\mathbb{F}_2^2}(x)(1,0)(g) &= f_{\mathbb{H}_4^*(\mathbb{F}_2^2),\mathbb{F}_2^2}(y)(1,0)(g) = 1, \\ f_{\mathbb{H}_4^*(\mathbb{F}_2^2),\mathbb{F}_2^2}(x)(0,1)(g) &= f_{\mathbb{H}_4^*(\mathbb{F}_2^2),\mathbb{F}_2^2}(y)(1,0)(g) = 1, \\ f_{\mathbb{H}_4^*(\mathbb{F}_2^2),\mathbb{F}_2^2}(x)(1,1)(g) &= f_{\mathbb{H}_4^*(\mathbb{F}_2^2),\mathbb{F}_2^2}(y)(1,0)(g) = 1. \end{aligned}$$

[3]In fact, $\mathrm{Aut}_K(\mathbb{H}_n(G))$ is precisely equal to this subgroup, but we will not need this fact.

However, no $\sigma \in \mathrm{Aut}_{\mathbb{F}_2}(\mathbb{H}_4^*(\mathbb{F}_2^2))$ *satisfies* $\sigma(x_1) = y_1$ *and* $\sigma(x_2) = y_2$*. This is because* $\sigma(x_1) = y_1$ *implies that* $\sigma = \mathrm{id}_{(\mathbb{F}_2^2)^4}$ *(as* $\mathrm{Aut}_{\mathbb{F}_2}(\mathbb{H}_4^*(\mathbb{F}_2^2))$ *is precisely given by the natural right action of the symmetric group* S_4 *on* $(\mathbb{F}_2^2)^4$ *by* $(g \cdot \sigma)_i \coloneqq g_{\sigma(i)}$ *(*$g \in (\mathbb{F}_2^2)^4$*,* $\sigma \in S_4$ *and* $i \in [4]$*)).*

The next example shows that it is not enough to consider the set $\{0,1\}^\ell \subseteq K^\ell$ that captures information only about subset sums of the tuples of words.

Example 5.34. *Consider the finite simple left* $\mathbb{F}_3$*-module* $\mathbb{F}_3$*, let* $n \coloneqq 3$ *and* $\ell \coloneqq 2$ *and consider the elements* $x, y \in (\mathbb{F}_3^n)^\ell$ *defined as*

$$\begin{aligned} y_1 \coloneqq x_1 \coloneqq x_2 &\coloneqq (0,1,2), \\ y_2 &\coloneqq (2,0,1), \end{aligned}$$

and note that

$$\begin{aligned} x_1 + x_2 &= (0,2,1), \\ y_1 + y_2 &= (2,1,0). \end{aligned}$$

Again, thinking of the function $f_{\mathbb{H}_3^*(\mathbb{F}_3)}$ *as taking values in* $\{0,1,2,3\}^{\mathbb{F}_3}$ *as we did in the proof of Proposition 5.32 and the function* $f_{\mathbb{H}_3^*(\mathbb{F}_3),\{0,1\}^2}$ *as taking values in* $(\{0,1,2,3\}^{\mathbb{F}_3})^{\{0,1\}^2}$*, then for every* $g \in \mathbb{F}_3$*, we have*

$$\begin{aligned} f_{\mathbb{H}_3^*(\mathbb{F}_3),\{0,1\}^2}(x)(0,0)(g) &= f_{\mathbb{H}_3^*(\mathbb{F}_3),\{0,1\}^2}(y)(0,0)(g) = 3 \cdot \mathbb{1}[g = 0], \\ f_{\mathbb{H}_3^*(\mathbb{F}_3),\{0,1\}^2}(x)(0,1)(g) &= f_{\mathbb{H}_3^*(\mathbb{F}_3),\{0,1\}^2}(y)(0,1)(g) = 1, \\ f_{\mathbb{H}_3^*(\mathbb{F}_3),\{0,1\}^2}(x)(1,0)(g) &= f_{\mathbb{H}_3^*(\mathbb{F}_3),\{0,1\}^2}(y)(1,0)(g) = 1, \\ f_{\mathbb{H}_3^*(\mathbb{F}_3),\{0,1\}^2}(x)(1,1)(g) &= f_{\mathbb{H}_3^*(\mathbb{F}_3),\{0,1\}^2}(y)(1,1)(g) = 1. \end{aligned}$$

However, no $\sigma \in \mathrm{Aut}_{\mathbb{F}_3}(\mathbb{H}_3^*(\mathbb{F}_3))$ *satisfies* $\sigma(x_1) = y_1$ *and* $\sigma(x_2) = y_2$ *because the former implies* $\sigma = \mathrm{id}_{\mathbb{F}_3^3}$*.*

Corollary 5.35. *Let* $\mathbb{F}$ *be a finite field and let* C *be an* $\mathbb{F}$*-linear* D*-code in the strong Hamming scheme* $\mathbb{H}_n^*(\mathbb{F})$*. Then for every* $\ell \in \mathbb{N}_+$*, we have*

$$|C| \leqslant \mathsf{val}(\mathcal{L}_{\mathbb{H}_n^*(\mathbb{F})^{\ell,\mathbb{F}^\ell}}(D^{\ell,\mathbb{F}^\ell}))^{1/\ell}.$$

Proof. Since $\mathbb{F}$ is a simple $\mathbb{F}$-module, by Theorem 5.27 and Proposition 5.32, C^ℓ is a K-linear $D^{\ell,\mathbb{F}^\ell}$-code in $\mathbb{H}_n^*(\mathbb{F})^{\ell,\mathbb{F}^\ell}$ and thus we have the bound

$$|C^\ell| \leqslant \mathsf{val}(\mathcal{L}_{\mathbb{H}_n^*(\mathbb{F})^{\ell,\mathbb{F}^\ell}}(D^{\ell,\mathbb{F}^\ell}))$$

provided by the Delsarte linear program for $\mathbb{H}_n^*(\mathbb{F})^{\ell,\mathbb{F}^\ell}$. ■

For the case of the weak Hamming scheme, annihilators of single elements will play an important role and it will be more convenient to work with annihilator Hamming schemes defined below. We will show that the annihilator Hamming scheme is indeed a symmetric translation scheme in Proposition 5.39 and the connection to the weak Hamming scheme will be established in Lemma 5.37.

Definition 5.36. *Given a ring K, a finite simple left K-module G and $n \in \mathbb{N}_+$, the annihilator Hamming scheme of order n over G is the symmetric translation scheme $\mathbb{H}_n^{\mathrm{Ann}_K}(G) \coloneqq (G^n, R)$ (with G^n equipped with the direct product left K-module structure), where*

$$\begin{aligned} R &\coloneqq \{r_h \mid h\colon 2^K \to \{0,1,\dots,n\}\} \setminus \{\varnothing\}, \\ r_h &\coloneqq \{(x,y) \in G^n \times G^n \mid \forall A \subseteq K, |x-y|_A = h(A)\} \qquad (h\colon 2^K \to \{0,1,\dots,n\}), \\ |z|_A &\coloneqq |\{i \in [n] \mid \mathrm{Ann}_K(z) = A\}|, \end{aligned}$$

that is, (x,y) is in the relation r_h if and only if for each set $A \subseteq K$, the number of positions $i \in [n]$ such that $\mathrm{Ann}_K(x_i - y_i) = A$ is exactly $h(A)$.

Before we actually prove that $\mathbb{H}_n^{\mathrm{Ann}_K}(G)$ is indeed a symmetric translation scheme, let us prove the following small lemma that says that when K is commutative, then the annihilator and the weak Hamming schemes coincide.

Lemma 5.37. *Let K be a ring, let G be a finite simple left K-module and let $n \in \mathbb{N}_+$. If K is commutative, then $\mathbb{H}_n(G) = \mathbb{H}_n^{\mathrm{Ann}_K}(G)$.*

Proof. Since K is commutative, for every $g \in K \setminus \{0\}$ we have

$$\mathrm{Ann}_K(g) = \mathrm{Ann}_K(Kg) = \mathrm{Ann}_K(G) \neq K.$$

Since the relations of $\mathbb{H}_n^{\mathrm{Ann}_K}(G)$ are based on the values of

$$|z|_A \coloneqq |\{i \in [n] \mid \mathrm{Ann}_K(z) = A\}|$$

we get

$$|z|_A = \begin{cases} |\{i \in [n] \mid z_i \neq 0\}|, & \text{if } A = \mathrm{Ann}_K(G), \\ |\{i \in [n] \mid z_i = 0\}|, & \text{if } A = K, \\ 0, & \text{otherwise.} \end{cases}$$

Thus it follows trivially that $\mathbb{H}_n^{\mathrm{Ann}_K}(G) = \mathbb{H}_n(G)$. ∎

Our next order of business is to show that $\mathbb{H}_n^{\mathrm{Ann}_K}(G)$ is indeed a symmetric translation scheme (even when K is not necessarily commutative). To do so, we need one lemma that says that in a finite simple left K-module G, the orbits of G under the natural action of the group $\mathrm{Aut}_K(G)$ of K-module automorphisms of G are completely determined by the annihilators of its elements. We in fact will prove a more general version over G^ℓ that will be needed later.

Lemma 5.38. *Let K be a ring, let G be a simple left K-module and let $\ell \in \mathbb{N}_+$. Let us also equip G^ℓ with the natural left K^ℓ-module structure over the direct product ring K^ℓ given by $(kg)_i \coloneqq k_i g_i$ ($k \in K^\ell$, $g \in G^\ell$ and $i \in [\ell]$). The following are equivalent for $x, y \in G^\ell$.*

i. We have $\mathrm{Ann}_{K^\ell}(x) = \mathrm{Ann}_{K^\ell}(y)$.

ii. There exists a left K-module automorphism $\sigma \in \mathrm{Aut}_K(G)$ of G such that $\sigma(x_i) = y_i$ for every $i \in [\ell]$.

Proof. For the implication (ii)⇒(i), note that for $k \in K^\ell$, we have the equivalences

$$ky = 0 \Leftrightarrow (\forall i \in [\ell], k_i y_i = 0) \Leftrightarrow (\forall i \in [\ell], k_i \sigma(x_i) = 0) \Leftrightarrow (\forall i \in [\ell], k_i x_i = 0) \Leftrightarrow kx = 0,$$

where the third equivalence follows since $\sigma \in \mathrm{Aut}_K(G)$ is a left K-module automorphism of G. Thus $\mathrm{Ann}_{K^\ell}(x) = \mathrm{Ann}_{K^\ell}(y)$.

Let us now prove the implication (i)⇒(ii). If $x = 0$, then $K^\ell = \mathrm{Ann}_{K^\ell}(x) = \mathrm{Ann}_{K^\ell}(y)$. Since G is simple, we must have $y = 0$ (as $\{z \in G \mid \mathrm{Ann}_K(z) = K\}$ is a proper left K-submodule of G, so it must be trivial) and any left K-module automorphism $\sigma \in \mathrm{Aut}_K(G)$ of G satisfies $\sigma(x_i) = y_i$ for every $i \in [\ell]$. Suppose then that $x \neq 0$ and without loss of generality, suppose that $x_1 \neq 0$ and thus $\mathrm{Ann}_K(x_1) \neq K$. Since

$$\mathrm{Ann}_K(x_1) = \{k \in K \mid (k, 0, \dots, 0) \in \mathrm{Ann}_{K^\ell}(x)\} = \{k \in K \mid (k, 0, \dots, 0) \in \mathrm{Ann}_{K^\ell}(y)\} = \mathrm{Ann}_K(y_1),$$

it follows that $\mathrm{Ann}_K(y_1) \neq K$ so $y_1 \neq 0$.

Since G is simple, we have $Kx_1 = G$, so we can define a function $\sigma\colon G \to G$ indirectly by $\sigma(kx_1) \coloneqq ky_1$ for every $k \in K$. To check that σ is well-defined, note that if $k_1 x_1 = k_2 x_1$, then $k_1 - k_2 \in \mathrm{Ann}_K(x_1) = \mathrm{Ann}_K(y_1)$ and thus $k_1 y_1 = k_2 y_2$. It is straightforward to check that σ is a left K-module endomorphism of G. Since the kernel of σ is a left K-submodule of G that does not contain x_1 and G is simple, it follows that the kernel of σ must be trivial, so σ is injective. On the other hand, the image of σ is a left K-submodule of G that contains $y_1 \neq 0$, so simplicity of G implies that σ is surjective and thus σ is a left K-module automorphism of G.

Let us now show that $\sigma(x_i) = y_i$ for every $i \in [\ell]$. For $i = 1$, this is obvious. For $i \geqslant 2$, let $k' \in K$ be such that $k'x_1 = x_i$ so that $\sigma(x_i) = k'y_1$. We now let $k \in K^\ell$ be given by $k_1 \coloneqq k'$, $k_i \coloneqq -1$ and $k_j \coloneqq 0$ for every $j \in [\ell] \setminus \{1, i\}$. Since $k'x_1 - x_i = 0$, we have $k \in \mathrm{Ann}_{K^\ell}(x) = \mathrm{Ann}_{K^\ell}(y)$, so we get $k'y_1 - y_i = 0$, and thus $\sigma(x_i) = k'y_1 = y_i$ as desired. ∎

Let us now prove that $\mathbb{H}_n^{\mathrm{Ann}_K}(G)$ is indeed a symmetric translation scheme. The proof uses similar ideas to that of Theorem 5.27.

Proposition 5.39. *Let K be a ring, let G be a finite simple left K-module and let $n \in \mathbb{N}_+$. Then the annihilator Hamming scheme $\mathbb{H}_n^{\mathrm{Ann}_K}(G)$ of order n over G is a symmetric translation scheme over the direct group G^n.*

Proof. Recall that the relation set of $\mathbb{H}_n^{\mathrm{Ann}_K(G)}$ is given by

$$R \coloneqq \{r_h \mid h\colon 2^K \to \{0, 1, \dots, n\}\} \setminus \{\varnothing\},$$

where

$$r_h \coloneqq \{(x, y) \in G^n \times G^n \mid \forall A \subseteq K, |x - y|_A = h(A)\} \qquad (h\colon 2^K \to \{0, 1, \dots, n\}),$$
$$|z|_A \coloneqq |\{i \in [n] \mid \mathrm{Ann}_K(z) = A\}|.$$

The fact that R forms a partition of $G^n \times G^n$ into non-empty subsets is obvious.

Since G is simple, the only element $g \in G$ with $\mathrm{Ann}_K(g) = K$ is $g = 0$ (as the set of such elements is a proper left K-submodule of G, so it must be trivial), thus for the function $h\colon 2^K \to \{0, 1, \dots, n\}$ given by $h(A) \coloneqq n\mathbb{1}[A = K]$, we have $r_h = \mathcal{D}_{G^n}$.

Note further that $\mathrm{Ann}_K(z) = \mathrm{Ann}_K(-z)$ for every $z \in G$, which immediately implies that $r_h^\top = r_h$ for every $h\colon 2^K \to \{0,1,\dots,n\}$.

It is also obvious that each r_h is invariant under the group action of G^n.

It remains only to show the existence of the intersection numbers for $\mathbb{H}_n^{\mathrm{Ann}_K(G)}$.

For every $A_1, A_2 \subseteq K$ and every $g_1, g_2 \in G$, let

$$N_{A_1,A_2}(g_1,g_2) \coloneqq |\{z \in G \mid \mathrm{Ann}_K(g_1 - z) = A_1 \wedge \mathrm{Ann}_K(z - g_2) = A_2\}|.$$

Claim 5.40. *If* $\mathrm{Ann}_K(g_1 - g_2) = \mathrm{Ann}_K(g_1' - g_2')$, *then* $N_{A_1,A_2}(g_1,g_2) = N_{A_1,A_2}(g_1',g_2')$.

Proof. By Lemma 5.38, there exists a left K-module automorphism $\sigma \in \mathrm{Aut}_K(G)$ of G such that $\sigma(g_1 - g_2) = g_1' - g_2'$. Then we have

$$\begin{aligned}
N_{A_1,A_2}(g_1' - g_2') &= |\{z \in G \mid \mathrm{Ann}_K(g_1' - z) = A_1 \wedge \mathrm{Ann}_K(z - g_2') = A_2\}| \\
&= |\{u \in G \mid \mathrm{Ann}_K(g_1' - g_2' - u) = A_1 \wedge \mathrm{Ann}_K(u) = A_2\}| \\
&= |\{u \in G \mid \mathrm{Ann}_K(\sigma(g_1 - g_2) - u) = A_1 \wedge \mathrm{Ann}_K(u) = A_2\}| \\
&= |\{u \in G \mid \mathrm{Ann}_K(g_1 - g_2 - \sigma^{-1}(u)) = A_1 \wedge \mathrm{Ann}_K(\sigma^{-1}(u)) = A_2\}| \\
&= |\{w \in G \mid \mathrm{Ann}_K(g_1 - w) = A_1 \wedge \mathrm{Ann}_K(w - g_2) = A_2\}|,
\end{aligned}$$

where the second equality follows from the substitution $u \coloneqq z - g_2'$, the fourth equality follows since $\sigma \in \mathrm{Aut}_K(G)$ is a left K-module automorphism and the fifth equality follows from the substitution $w \coloneqq \sigma^{-1}(u) + g_2$. ■

Claim 5.40 implies that for $A_1, A_2, B \subseteq K$ we can define $N^B_{A_1,A_2} \in \mathbb{N}$ such that $N_{A_1,A_2}(g_1,g_2) = N^B_{A_1,A_2}$ whenever $\mathrm{Ann}_K(g_1 - g_2) = B$.

Note now that if $(x,y) \in r_h$ for some $h\colon 2^K \to \{0,1,\dots,n\}$ and $h_1, h_2\colon 2^K \to \{0,1,\dots,n\}$, then we have

$$\begin{aligned}
&|\{z \in G^n \mid (x,z) \in r_{h_1} \wedge (z,y) \in r_{h_2}\}| \\
&\quad = \sum_{F \in \mathcal{F}} \prod_{i=1}^{n} |\{w \in G \mid \mathrm{Ann}_K(x_i - w) = F(i)_1 \wedge \mathrm{Ann}_K(w - y_i) = F(i)_2\}|,
\end{aligned}$$

where $\mathcal{F}$ is the set of functions $F\colon [n] \to 2^K \times 2^K$ such that

$$|\{i \in [n] \mid F(i)_j = A\}| = h_j(A) \qquad (j \in [2], A \subseteq K).$$

Using the definition of the numbers $N^B_{A_1,A_2}$, we get

$$\sum_{F \in \mathcal{F}} \prod_{i=1}^{n} |\{w \in G \mid \mathrm{Ann}_K(x_i - w) = F(i)_1 \wedge \mathrm{Ann}_K(w - y_i) = F(i)_2\}| = \sum_{F \in \mathcal{F}} \prod_{i=1}^{n} N^{\mathrm{Ann}_K(x_i - y_i)}_{F(i)_1,F(i)_2}.$$

Finally, note that if $(x',y') \in r_h$, then there exists a permutation $\sigma \in S_n$ such that $\mathrm{Ann}_K(x_i' - y_i') = \mathrm{Ann}_K(x_{\sigma(i)} - y_{\sigma(i)})$ for every $i \in [n]$, which implies that

$$\sum_{F \in \mathcal{F}} \prod_{i=1}^{n} N^{\mathrm{Ann}_K(x_i' - y_i')}_{F(i)_1,F(i)_2} = \sum_{F \in \mathcal{F}} \prod_{i=1}^{n} N^{\mathrm{Ann}_K(x_i - y_i)}_{F(\sigma^{-1}(i))_1,F(\sigma^{-1}(i))_2} = \sum_{F' \in \mathcal{F}} \prod_{i=1}^{n} N^{\mathrm{Ann}_K(x_i - y_i)}_{F'(i)_1,F'(i)_2},$$

where the last equality follows from the substitution $F' := F \circ \sigma^{-1}$. Thus the existence of the intersection numbers is proved. ■

It will be very convenient to work with the following equivalence relation that can be seen as the equivalence relation of the "projective space of G^ℓ with origin".

Definition 5.41. *Let K be a ring, let G be a finite simple left K-module and $\ell \in \mathbb{N}_+$. The equivalence relation $\sim_\ell$ over G^ℓ is defined by*

$$x \sim_\ell y \Leftrightarrow \exists \sigma \in \mathrm{Aut}_K(G), \forall i \in [\ell], \sigma(x_i) = y_i,$$

that is, the equivalence classes of $\sim_\ell$ are the orbits of the natural diagonal action of $\mathrm{Aut}_K(G)$ on G^ℓ given by $(\sigma \cdot x)_i := \sigma(x_i)$ ($\sigma \in \mathrm{Aut}_K(G)$, $x \in G^\ell$, $i \in [\ell]$).

In the definition above, if $K = \mathbb{F}$ for a finite field $\mathbb{F}$ (hence G is a 1-dimensional $\mathbb{F}$-vector space), then $x \sim_\ell y$ if and only if there exist $k_1, k_2 \in \mathbb{F} \setminus \{0\}$ such that $k_1 x = y$ and $x = k_2 y$, that is, $\sim_\ell$ is the equivalence relation defining the $(\ell - 1)$-dimensional projective space $P(\mathbb{F}^\ell) \cup \{0\}$ with origin.

Remark 5.42. *Under the definition of $\sim_\ell$, we can reinterpret Lemma 5.38, as saying that $x \sim_\ell y$ if and only if $\mathrm{Ann}_{K^\ell}(x) = \mathrm{Ann}_{K^\ell}(y)$.*

The next lemma is an analogue of Lemma 5.30 that says that the distribution of the $\sim_\ell$-equivalence class of an ℓ-tuple of random variables with values in a finite simple left K-module (for a finite ring K) can be recovered from the (individual) distributions of the $\sim_1$-equivalence classes of all K-linear combinations of them.

Lemma 5.43. *Let K be a finite ring, let G be a finite simple left K-module, let $\boldsymbol{X}$ be a random variable with values in G^ℓ for some $\ell \in \mathbb{N}_+$ and for every $k \in K^\ell$, let $\boldsymbol{Y}_k := \sum_{i=1}^\ell k_i \boldsymbol{X}_i$. Then*

$$\mathbb{P}[\boldsymbol{X} \sim_\ell x] = \left(\frac{|G|}{|\mathrm{Stab}_{\mathrm{Aut}_K(G)}(x)| \cdot |K^\ell| \cdot (|G| - 1)} \cdot \sum_{k \in K^\ell} \left| \mathrm{Stab}_{\mathrm{Aut}_K(G)} \left(\sum_{i=1}^\ell k_i x_i \right) \right| \cdot \mathbb{P}\left[\boldsymbol{Y}_k \sim_1 \sum_{i=1}^\ell k_i x_i \right] \right) - \frac{|O_{\mathrm{Aut}_K(G)}(x)|}{|G| - 1}$$

for every $x \in G^\ell$, where $\mathrm{Stab}_{\mathrm{Aut}_K(G)}(z)$ is the stabilizer group of z under the action of $\mathrm{Aut}_K(G)$ and $O_{\mathrm{Aut}_K(G)}(z)$ is the orbit of z under the action of $\mathrm{Aut}_K(G)$ (the actions of $\mathrm{Aut}_K(G)$ on G and G^ℓ are respectively the natural action and the diagonal action).

Proof. By Lemma 5.38 (see also Remark 5.42), we know that the $\sim_\ell$-equivalence class of x is precisely the orbit $O_{\mathrm{Aut}_K(G)}(x)$, so we have

$$\mathbb{P}[\boldsymbol{X} \sim_\ell x] = \frac{1}{|\mathrm{Stab}_{\mathrm{Aut}_K(G)}(x)|} \sum_{\sigma \in \mathrm{Aut}_K(G)} \mathbb{P}[\boldsymbol{X} = \sigma(x)].$$

Letting χ be a non-trivial character of G, by Lemma 5.30, we get

$$\mathbb{P}[\boldsymbol{X} \sim_\ell x] = \frac{1}{|\mathrm{Stab}_{\mathrm{Aut}_K(G)}(x)| \cdot |K^\ell|} \cdot \sum_{\substack{\sigma \in \mathrm{Aut}_K(G) \\ k \in K^\ell \\ y \in G}} \chi\left(\sum_{i=1}^\ell k_i \sigma(x_i) - y \right) \cdot \mathbb{P}[\boldsymbol{Y}_k = y].$$

Recall that for every $g \in G$, if we average the value $\chi(g)$ over all non-trivial characters of G, then we get $(|G|\mathbb{1}[g=0]-1)/(|G|-1)$, thus by performing such averaging operation in the above, we get

$$\mathbb{P}[\boldsymbol{X} \sim_\ell x] = \frac{1}{|\mathrm{Stab}_{\mathrm{Aut}_K(G)}(x)| \cdot |K^\ell| \cdot (|G|-1)} \cdot \sum_{\substack{\sigma \in \mathrm{Aut}_K(G) \\ k \in K^\ell \\ y \in G}} \left(|G| \mathbb{1}\left[\sum_{i=1}^{\ell} k_i \sigma(x_i) = y\right] - 1 \right) \cdot \mathbb{P}[\boldsymbol{Y}_k = y]. \tag{9}$$

Note now that

$$\begin{aligned} \sum_{\substack{\sigma \in \mathrm{Aut}_K(G) \\ k \in K^\ell \\ y \in G}} |G| \mathbb{1}\left[\sum_{i=1}^{\ell} k_i \sigma(x_i) = y\right] \cdot \mathbb{P}[\boldsymbol{Y}_k = y] &= \sum_{\substack{\sigma \in \mathrm{Aut}_K(G) \\ k \in K^\ell}} |G| \cdot \mathbb{P}\left[\boldsymbol{Y}_k = \sigma^{-1}\left(\sum_{i=1}^{\ell} k_i x_i\right)\right] \\ &= |G| \sum_{k \in K^\ell} \left|\mathrm{Stab}_{\mathrm{Aut}_K(G)}\left(\sum_{i=1}^{\ell} k_i x_i\right)\right| \cdot \mathbb{P}\left[\boldsymbol{Y}_k \sim_1 \sum_{i=1}^{\ell} k_i x_i\right], \end{aligned} \tag{10}$$

where the last equality follows since Lemma 5.38 and Remark 5.42 imply that the $\sim_1$-equivalence class of $\sum_{i=1}^{\ell} k_i x_i$ is precisely its $\mathrm{Aut}_K(G)$-orbit.

On the other hand, we have

$$\sum_{\substack{\sigma \in \mathrm{Aut}_K(G) \\ k \in K^\ell \\ y \in G}} \mathbb{P}[\boldsymbol{Y}_k = y] = \sum_{\substack{\sigma \in \mathrm{Aut}_K(G) \\ k \in K^\ell}} 1 = |\mathrm{Aut}_K(G)| \cdot |K^\ell| = |O_{\mathrm{Aut}_K(G)}(x)| \cdot |\mathrm{Stab}_{\mathrm{Aut}_K(G)}(x)| \cdot |K^\ell|. \tag{11}$$

The result now follows by putting together (9), (10) and (11). ■

We can finally prove that for the annihilator Hamming scheme $\mathbb{H}_n^{\mathrm{Ann}_K}(G)$, the associated function $f_{\mathbb{H}_n^{\mathrm{Ann}_K}(G),K^\ell}$ factors through types.

Proposition 5.44. *Let K be a ring, let G be a finite simple left K-module and let $n, \ell \in \mathbb{N}_+$. Consider the annihilator Hamming scheme $\mathbb{H}_n^{\mathrm{Ann}_K}(G)$ of order n over G. Then $f_{\mathbb{H}_n^{\mathrm{Ann}_K}(G),K^\ell}$ factors through types of $\mathbb{H}_n^{\mathrm{Ann}_K}(G)$.*

Proof. Recall that the relation set of $\mathbb{H}_n^{\mathrm{Ann}_K(G)}$ is given by

$$R := \{r_h \mid h \colon 2^K \to \{0,1,\dots,n\}\} \setminus \{\varnothing\},$$

where

$$\begin{aligned} r_h &:= \{(x,y) \in G^n \times G^n \mid \forall A \subseteq K, |x-y|_A = h(A)\} \qquad (h \colon 2^K \to \{0,1,\dots,n\}), \\ |z|_A &:= |\{i \in [n] \mid \mathrm{Ann}_K(z) = A\}|. \end{aligned}$$

Just as in the proof of Proposition 5.32, we will abuse notation and write $f_{\mathbb{H}_n^{\mathrm{Ann}_K}(G)}(x) = h$ in place of $f_{\mathbb{H}_n^{\mathrm{Ann}_K}(G)}(x) = r_h$, thus viewing $f_{\mathbb{H}_n^{\mathrm{Ann}_K}}$ as a $\{0,1,\dots,n\}^{2^K}$-valued function. Accordingly, we will also view $f_{\mathbb{H}_n^{\mathrm{Ann}_K}(G),K^\ell}$ as a function with values in $(\{0,1,\dots,n\}^{2^K})^{K^\ell}$ rather than in R^{K^ℓ}.

By the same argument as in the proof of Proposition 5.32, it is enough to prove the case when K is finite: the arbitrary case can be reduced to the finite case by letting $K' \coloneqq K/\operatorname{Ann}_K(G)$, considering the natural K'-module structure on G and noting that $f_{\mathbb{H}_n^{\mathrm{Ann}_K}(G),K^\ell}$ factors through types of $\mathbb{H}_n^{\mathrm{Ann}_K}(G)$ if and only if $f_{\mathbb{H}_n^{\mathrm{Ann}_{K'}}(G),(K')^\ell}$ factors through types of $\mathbb{H}_n^{\mathrm{Ann}_{K'}}(G)$ and K' is finite by Lemma 5.31.

Let us then prove the case when K is finite.

First, we claim that $\mathrm{Aut}_K(\mathbb{H}_n^{\mathrm{Ann}_K}(G))$ contains a subgroup isomorphic to a semidirect product[4] $\mathrm{Aut}_K(G)^n \rtimes S_n$, where S_n is the symmetric group on $[n]$. Indeed, consider the natural actions of $\mathrm{Aut}_K(G)$ and S_n on G^n given by

$$(F \cdot g)_i \coloneqq F_i(g_i), \qquad\qquad (g \cdot \sigma)_i \coloneqq g_{\sigma(i)}$$

for $F \in \mathrm{Aut}_K(G)^n$, $g \in G^n$, $\sigma \in S_n$ and $i \in [n]$. It is obvious that these actions are free and preserve the K-module structure of G^n, and thus induce subgroups of the K-module automorphism group of G^n isomorphic to $\mathrm{Aut}_K(G)^n$ and S_n, respectively. Let H be the product of these subgroups. It is straightforward to check that if $F \cdot g = g \cdot \sigma$ holds for every $g \in G^n$, then $F_i = \mathrm{id}_G^n$ for every $i \in [n]$ and $\sigma = \mathrm{id}_n$, so these subgroups have trivial intersection. Since

$$\Big(\big(F \cdot (g \cdot \sigma)\big) \cdot \sigma^{-1}\Big)_i = \big(F \cdot (g \cdot \sigma)\big)_{\sigma^{-1}(i)} = F_{\sigma^{-1}(i)}\big((g \cdot \sigma)_{\sigma^{-1}(i)}\big) = F_{\sigma^{-1}(i)}(g_i),$$

it follows that the subgroup isomorphic to $\mathrm{Aut}_K(G)^n$ is normal in H and thus $H \cong \mathrm{Aut}_K(G)^n \rtimes S_n$. It remains to show that H also preserves the association scheme structure. Indeed, note that for $F \in \mathrm{Aut}_K(G)^n$, $g \in G^n$, $\sigma \in S_n$ and $A \subseteq K$, we have

$$\begin{aligned} |F(g)|_A &= |\{i \in [n] \mid \operatorname{Ann}_K(F_i(g_i)) = A\}| = |\{i \in [n] \mid \operatorname{Ann}_K(g_i) = A\}| = |g|_A \\ |g \cdot \sigma|_A &= |\{i \in [n] \mid \operatorname{Ann}_K(g_{\sigma(i)}) = A\}| = |\{i \in [n] \mid \operatorname{Ann}_K(g_i) = A\}| = |g|_A. \end{aligned}$$

Thus $f_{\mathbb{H}_n^{\mathrm{Ann}_K}(G)}$ is invariant under both actions of the groups $\mathrm{Aut}_K(G)^n$ and S_n, so H is a subgroup of $\mathrm{Aut}_K(\mathbb{H}_n^{\mathrm{Ann}_K}(G))$.

We now define the same random variables as in Proposition 5.32: given a point $z \in (G^n)^\ell$, let $\boldsymbol{X}^z$ be the random variable with values in G^ℓ defined by

$$\boldsymbol{X}_j^z \coloneqq (z_j)_{\boldsymbol{i}} \qquad (j \in [\ell]),$$

where $\boldsymbol{i}$ is picked uniformly at random in $[n]$ and for every $k \in K^\ell$, let $\boldsymbol{Y}^z \coloneqq \sum_{j=1}^\ell k_j \boldsymbol{X}_j^z$.

[4] In fact, $\mathrm{Aut}_K(\mathbb{H}_n^{\mathrm{Ann}_K}(G))$ is exactly equal to this semidirect product, but we will not need this result.

Note that for every $g \in G$, we have

$$\begin{aligned}\mathbb{P}[\boldsymbol{Y}_k^z \sim_1 g] &= \mathbb{P}[\mathrm{Ann}_K(\boldsymbol{Y}_k^z) = \mathrm{Ann}_K(g)] \\ &= \frac{\left|\left\{i \in [n] \;\middle|\; \mathrm{Ann}_K\left(\sum_{j=1}^{\ell} k_j(z_j)_i\right) = \mathrm{Ann}_K(g)\right\}\right|}{n} \\ &= \frac{f_{\mathbb{H}_n^{\mathrm{Ann}_K}(G),K^\ell}(z)(k)(\mathrm{Ann}_K(g))}{n},\end{aligned}$$

where the first equality follows from Lemma 5.38 and Remark 5.42.

By Lemma 5.43, the individual distributions of the $\sim_1$-equivalence classes of the $\boldsymbol{Y}_k^z$ ($k \in K^\ell$) completely determine the distribution of the $\sim_\ell$-equivalence class of $\boldsymbol{X}^z$. This means that if $x, y \in (G^n)^\ell$ are such that $f_{\mathbb{H}_n^{\mathrm{Ann}_K}(G),K^\ell}(x) = f_{\mathbb{H}_n^{\mathrm{Ann}_K}(G),K^\ell}(y)$, then for every $w \in G^\ell$, we have

$$\mathbb{P}[\boldsymbol{X}^x \sim_\ell w] = \mathbb{P}[\boldsymbol{X}^y \sim_\ell w].$$

Thus there exists a permutation $\sigma \in S_n$ such that for every $i \in [n]$, we have

$$\left((x_1)_{\sigma(i)}, (x_2)_{\sigma(i)}, \ldots, (x_\ell)_{\sigma(i)}\right) \sim_\ell \left((y_1)_i, (y_2)_i, \ldots, (y_\ell)_i\right),$$

so by the definition of $\sim_\ell$, for each $i \in [n]$, there exists a left K-module automorphism $F_i \in \mathrm{Aut}_K(G)$ such that

$$F_i((x_j)_{\sigma(i)}) = (y_j)_i$$

for every $j \in [\ell]$ and thus for $F = (F_1, \ldots, F_n) \in \mathrm{Aut}_K(G)^n$, we get

$$F \cdot (x_j \cdot \sigma) = y_j$$

for every $j \in [\ell]$, so $f_{\mathbb{H}_n^{\mathrm{Ann}_K}(G),K^\ell}$ factors through types. ■

From Lemma 5.37 and Proposition 5.44 above, we can finally show that under mild assumptions $f_{\mathbb{H}_n(G),K^\ell}$ also factors through types for the weak Hamming scheme $\mathbb{H}_n(G)$.

Proposition 5.45. *Let K be a ring and G be a finite simple left K-module and $n, \ell \in \mathbb{N}_+$. Consider the weak Hamming scheme $\mathbb{H}_n(G)$ of order n over G equipped with the product left K-module structure on G^n.*

If K is commutative, then $f_{\mathbb{H}_n(G),K^\ell}$ factors through types of $\mathbb{H}_n(G)$.

Proof. Follows directly from Proposition 5.44 as Lemma 5.37 implies that the weak Hamming scheme $\mathbb{H}_n(G)$ coincides with the annihilator Hamming scheme $\mathbb{H}_n^{\mathrm{Ann}_K}(G)$. ■

Corollary 5.46. *Let $\mathbb{F}$ be a finite field and let C be an $\mathbb{F}$-linear D-code in the weak Hamming scheme $\mathbb{H}_n(\mathbb{F})$. Then for every $\ell \in \mathbb{N}_+$, we have*

$$|C| \leqslant \mathsf{val}(\mathcal{L}_{\mathbb{H}_n(\mathbb{F})^{\ell,\mathbb{F}^\ell}}(D^{\ell,\mathbb{F}^\ell}))^{1/\ell}.$$

Proof. By Theorem 5.27 and Proposition 5.45, C^ℓ is a K-linear $D^{\ell,\mathbb{F}^\ell}$-code in $\mathbb{H}_n(\mathbb{F})^{\ell,\mathbb{F}^\ell}$ and thus we have the bound

$$|C^\ell| \leqslant \mathsf{val}(\mathcal{L}_{\mathbb{H}_n(\mathbb{F})^{\ell,\mathbb{F}^\ell}}(D^{\ell,\mathbb{F}^\ell}))$$

provided by the Delsarte linear program for $\mathbb{H}_n(\mathbb{F})^{\ell,\mathbb{F}^\ell}$. ■

6 Main Properties of the Krawtchouk Hierarchies

This section presents our main results on the linear programming hierarchy. The first result is the completeness of the higher-order linear programming hierarchies for *linear* codes. The second result is the collapse of the hierarchies for *general* codes.

6.1 Completeness for Linear Codes

In this section, we show the (approximate) completeness of our linear programming hierarchy for linear codes over a finite field $\mathbb{F}$.

We will show completeness at level $O(n^2)$ via a counting argument. The intuition is that the hierarchy is likely already complete at level n (and we conjecture this to be the case). At level n, the feasible region of the LP already encodes $A_q^{\mathrm{Lin}}(n,d)$. That is, since at level n there is a variable for each possible basis of a subspace of $\mathbb{F}_q^n$, just writing down the distance constraints of $\mathrm{KrawtchoukLP}_{\mathrm{Lin}}^{\mathbb{F}}(n,d,n)$ allows one to deduce the true value of $A_q^{\mathrm{Lin}}(n,d)$. Of course this property is not sufficient to imply that the *value* of $\mathrm{KrawtchoukLP}_{\mathrm{Lin}}^{\mathbb{F}}(n,d,n)$ is correct. At an intuitive level, the below proof shows that at level $O(n^2)$ the large-dimensional subspaces outweigh the small-dimensional subspaces enough to deduce the correct value of $A_q^{\mathrm{Lin}}(n,d)$.

Theorem 6.1 (Completeness)**.** *Let $\mathbb{F}$ be a finite field, let $q := |\mathbb{F}|$, let $\varepsilon \in (0,1)$ and let $\ell \geqslant 9(n^2\ln(q)+1)/(\ln(1+\varepsilon))^2$. Then for every $d \in \{0,1,\dots,n\}$, we have*

$$\mathsf{val}(\mathrm{KrawtchoukLP}_{\mathrm{Lin}}^{\mathbb{F}}(n,d,\ell))^{1/\ell} \leqslant (1+\varepsilon)\cdot A_q^{\mathrm{Lin}}(n,d).$$

Before proving this theorem, note that since $\mathbb{F}$-linear codes must necessarily have size of the form q^k for some $k \in \mathbb{N}$, by taking $\varepsilon < q-1$, we get

$$A_q^{\mathrm{Lin}}(n,d) = q^{\left\lfloor (\log_q \mathsf{val}(\mathrm{KrawtchoukLP}_{\mathrm{Lin}}^{\mathbb{F}}(n,d,\ell)))/\ell \right\rfloor}$$

whenever $\ell > 9(n^2\ln(q)+1)/(\ln(q))^2$.

Proof. By Remark 5.28 that $\mathrm{KrawtchoukLP}_{\mathrm{Lin}}^{\mathbb{F}}(n,d,\ell)$ is the Delsarte linear program $\mathcal{L}_{S^{\ell,T}}(D_d^{\ell,T})$ for $S = \mathbb{H}_n(\mathbb{F}_2)$, $T = \mathbb{F}_2^\ell$ and $D_d := \{r_0, r_d, r_{d+1}, \dots, r_n\}$. By Theorem 5.22, the value of this linear program coincides with the value of ϑ' in the associated graph $G := G_{S^{\ell,T}}(D_d^{\ell,T})$, i.e., the optimum value of the semi-definite program

$$\begin{array}{lll}
\max & \langle J, M\rangle & \\
\text{s.t.} & \operatorname{tr} M = 1 & \text{(Normalization)} \\
& M[u,v] = 0 \qquad \forall \{u,v\} \in E(G) & \text{(Independent set)} \\
& M \succcurlyeq 0 & \text{(PSD-ness)} \\
& M[u,v] \geq 0 \qquad \forall u,v \in V(G) & \text{(Non-negativity)},
\end{array}$$

where the variables is $M \in \mathbb{R}^{V\times V}$ symmetric and

$$E(G) := \left\{ \{x,y\} \in \binom{(\mathbb{F}^n)^\ell}{2} \;\middle|\; \forall k \in \mathbb{F}^\ell, \left| \sum_{i=1}^{\ell} k_i(x_i - y_i) \right| \notin [d-1] \right\},$$

where $|z| \coloneqq |\{j \in [n] \mid z_j \neq 0\}|$ is the Hamming weight of z.

Let k_0 be the maximum dimension of an $\mathbb{F}$-linear code of distance d in $\mathbb{F}^n$ (that is, let $k_0 \coloneqq \log_q A_q^{\mathrm{Lin}}(n,d)$), let M be a feasible solution of the program above and let us provide an upper bound for the objective value $\langle J, M\rangle$. Note that symmetrizing M under the automorphism group $\mathrm{Aut}(G)$ of the Cayley graph G does not change the objective value $\langle J, M\rangle$ (and preserves all restrictions), so we may suppose that M is $\mathrm{Aut}(G)$-invariant, which in particular implies that all diagonal entries of M are equal and since the trace of M is 1, it follows that all diagonal entries of M are equal to $q^{-n\ell}$. On the other hand, since M is positive semi-definite, any 2×2 principal minor of M is non-negative and thus all off-diagonal entries of M have absolute value at most $q^{-n\ell}$, that is, we have $\|M\|_\infty = q^{-n\ell}$.

Since the objective value $\langle J, M\rangle$ is simply the sum of all entries of M, we can provide an upper bound for it by simply giving an upper bound on how many entries of M are allowed to be non-zero.

Note that for an entry M_{xy} indexed by $(x,y) \in (\mathbb{F}^n)^\ell \times (\mathbb{F}^n)^\ell$ to be non-zero, the difference vectors $z_1, \ldots, z_\ell \in \mathbb{F}^n$ given by $z_i \coloneqq x_i - y_i$ ($i \in [\ell]$) must span an $\mathbb{F}$-linear subspace of dimension at most k_0 (as any subspace of larger dimension necessarily has distance smaller than d and thus some $k \in \mathbb{F}^\ell$ will have $|\sum_{i=1}^\ell k_i(x_i - y_i)| \in [d-1]$).

By letting $\gamma_{n,\ell,k}$ be the number of tuples $(z_1, \ldots, z_\ell)$ that span a subspace of dimension $k \in \{0, 1, \ldots, n\}$, since each difference $(z_1, \ldots, z_\ell)$ is realized as $z_i = x_i - y_i$ for exactly $q^{n\ell}$ pairs $(x,y) \in (\mathbb{F}^n)^\ell \times (\mathbb{F}^n)^\ell$, we get

$$\langle J, M\rangle \leqslant \sum_{k=0}^{k_0} \gamma_{n,\ell,k} \cdot q^{n\ell} \cdot \|M\|_\infty \leqslant \sum_{k=0}^{k_0} \gamma_{n,\ell,k}.$$

We claim that

$$\gamma_{n,\ell,k} \leqslant \binom{\ell}{k} \cdot \beta_{n,k} \cdot (q^k)^{\ell-k}, \tag{12}$$

where

$$\beta_{n,k} \coloneqq \prod_{j=0}^{k-1} (q^n - q^j)$$

is the number of linearly independent ordered k-tuples in $\mathbb{F}^n$. Indeed, the upper bound in (12) follows by picking k out of the ℓ vectors to have a linearly independent ordered k-tuple, then picking each of the other $\ell - k$ positions to be a linear combination of these k vectors.

Using this bound along with $\binom{\ell}{k} \leqslant \ell^k$ and $\beta_{n,k} \leqslant q^{nk}$, we get

$$\begin{aligned}\langle J, M\rangle &\leqslant \sum_{k=0}^{k_0} \gamma_{n,\ell,k} \leqslant \sum_{k=0}^{k_0} \ell^k q^{nk} q^{k\ell} \leqslant \ell^{k_0} q^{nk_0} \left(q^{k_0\ell} + \sum_{k=0}^{k_0-1} q^{k\ell}\right) \\ &= \ell^{k_0} q^{nk_0} \left(q^{k_0\ell} + \frac{q^{k_0\ell} - 1}{q^\ell - 1}\right) \leqslant 2\ell^n q^{n^2} q^{k_0\ell}.\end{aligned}$$

Taking the ℓth root and recalling that $q^{k_0} = A_q^{\mathrm{Lin}}(n,d)$ we conclude that

$$\mathsf{val}(\mathrm{KrawtchoukLP}^{\mathbb{F}}_{\mathrm{Lin}}(n,d,\ell))^{1/\ell} \leqslant (2\ell^n q^{n^2})^{1/\ell} A_q^{\mathrm{Lin}}(n,d).$$

Finally, the hypothesis $\ell \geqslant 9(n^2\ln(q)+1)/(\ln(1+\varepsilon))^2$ implies that $(2\ell^n q^{n^2})^{1/\ell} \leqslant 1+\varepsilon$, which concludes the proof (a detailed computation is included in Lemma B.1 in Appendix B). ■

Remark 6.2. *The same proof of Theorem 6.1 also works for D-codes over the weak Hamming scheme* $\mathbb{H}_n(\mathbb{F})$ *yielding*

$$\mathsf{val}(\mathcal{L}^{\ell,\mathbb{F}^\ell}_{\mathbb{H}_n(\mathbb{F})}(D^{\ell,\mathbb{F}^\ell}))^{1/\ell} \leqslant (1+\varepsilon)|C^*|,$$

where C^* *is a D-code in* $\mathbb{H}_n(\mathbb{F})$ *of maximum size. The same proof also applies to D-codes over the strong Hamming scheme* $\mathbb{H}^*_n(\mathbb{F})$.

6.2 Hierarchy Collapse for General Codes

In this section, we show that without the additional semantic linearity constraints imposed by $\mathrm{KrawtchoukLP}_{\mathrm{Lin}}(n,d,\ell)$, the associated hierarchy $\mathrm{KrawtchoukLP}(n,d,\ell)$ does not give any improvement over the original Delsarte linear programming approach. The proof is in two steps: first, we show that just tensoring the program does not change the relative value (Lemma 6.3). Second, we show that refining the scheme and adding only natural (non-semantic) constraints does not change the value of the associated Delsarte linear program (Lemma 6.4).

Recall that all of our linear programming hierarchies can be interpreted as a Delsarte LP of some association scheme and some code constraints (see Remark 5.28). For these proofs, we then heavily rely on the connection to the unsymmetrized program ϑ' in Theorem 5.22.

Lemma 6.3. *Let* $S_1 = (X_1, R_1)$ *and* $S_2 = (X_2, R_2)$ *be commutative association schemes and let* $D_i \subseteq R_i$ *with* $\mathcal{D}_{X_i} \in D_i$ *(*$i \in [2]$*). Then*

$$\mathsf{val}(\mathcal{L}_{S_1\otimes S_2}(D_1 \otimes D_2)) = \mathsf{val}(\mathcal{L}_{S_1}(D_1)) \cdot \mathsf{val}(\mathcal{L}_{S_2}(D_2)),$$

where

$$D_1 \otimes D_2 \coloneqq \{r_1 \otimes r_2 \mid r_1 \in D_1 \wedge r_2 \in D_2\}.$$

Proof. By Theorem 5.22, for $i \in [2]$, we have

$$\mathsf{val}(\mathcal{L}_{S_i}(D_i)) = \vartheta'(G_{S_i}(D_i)), \qquad \mathsf{val}(\mathcal{L}_{S_1\otimes S_2}(D_1 \otimes D_2)) = \vartheta'(G_{S_1\otimes S_2}(D_1 \otimes D_2)).$$

Given solutions M_1 and M_2 of the primal semi-definite programs $\mathcal{S}(G_{S_1}(D_1))$ and $\mathcal{S}(G_{S_2}(D_2))$ associated with $\vartheta'(G_{S_1}(D_1))$ and $\vartheta'(G_{S_2}(D_2))$, respectively, note that the tensor product $M \coloneqq M_1 \otimes M_2$ is a feasible solution of $\mathcal{S}(G_{S_1\otimes S_2}(D_1 \otimes D_2))$ since for if $((x_1,x_2),(y_1,y_2)) \in r_1 \otimes r_2$ for some $r_1 \otimes r_2 \in (R_1 \otimes R_2) \setminus (D_1 \otimes D_2)$, then $(x_1,y_1) \in r_1$ or $(x_2,y_2) \in r_2$, which implies that $M_{(x_1,x_2),(y_1,y_2)} = 0$. Since we also have

$$\langle J, M\rangle = \langle J, M_1\rangle \cdot \langle J, M_2\rangle,$$

it follows that

$$\mathsf{val}(\mathcal{L}_{S_1\otimes S_2}(D_1\otimes D_2)) \geqslant \mathsf{val}(\mathcal{L}_{S_1}(D_1))\cdot\mathsf{val}(\mathcal{L}_{S_2}(D_2)).$$

For the other inequality, given solutions (β_1, N^1) and (β_2, N^2) of the dual semi-definite programs $\mathcal{S}'(G_{S_1}(D_1))$ and $\mathcal{S}'(G_{S_2}(D_2))$, respectively, let $\beta := \beta_1\cdot\beta_2$ and $N := N^1\otimes N^2$. Since $N^i \preccurlyeq \beta_i I$ ($i\in[2]$), it follows that $N\preccurlyeq \beta I$. Note also that if $((x_1,x_2),(y_1,y_2))\in r_1\otimes r_2$ for some $r_1\otimes r_2\in D_1\otimes D_2$, then since $r_i\in D_i$, we must have

$$N_{((x_1,x_2),(y_1,y_2))} = N^1_{x_1,y_1}\cdot N^2_{x_2,y_2}\geqslant 1,$$

thus (β, N) is a feasible solution of $\mathcal{S}'(G_{S_1\otimes S_2}(D_1\otimes D_2))$ which implies

$$\mathsf{val}(\mathcal{L}_{S_1\otimes S_2}(D_1\otimes D_2)) \leqslant \mathsf{val}(\mathcal{L}_{S_1}(D_1))\cdot\mathsf{val}(\mathcal{L}_{S_2}(D_2)),$$

as desired. ■

We now proceed to refinements.

Lemma 6.4. *Let $S' = (X, R')$ be a commutative refinement of a commutative association scheme $S = (X, R)$, let $D\subseteq R$ be such that $\mathcal{D}_X\in D$ and let*

$$D' := \{r\in R' \mid \exists \widehat{r}\in R, r\subseteq\widehat{r}\}.$$

Then

$$\mathsf{val}(\mathcal{L}_S(D)) = \mathsf{val}(\mathcal{L}_{S'}(D')).$$

Proof. By Theorem 5.22, we have

$$\mathsf{val}(\mathcal{L}_S(D)) = \vartheta'(G_S(D)), \qquad \mathsf{val}(\mathcal{L}_{S'}(D')) = \vartheta'(G_{S'}(D')).$$

But note that the semi-definite programs $\mathcal{S}(G_S(D))$ and $\mathcal{S}(G_{S'}(D'))$ corresponding to $\vartheta'(G_S(D))$ and $\vartheta'(G_{S'}(D'))$ are identical (i.e., have exactly the same restrictions and objective value), so we get $\vartheta'(G_S(D)) = \vartheta'(G_{S'}(D'))$ trivially. ■

Composing Lemmas 6.3 and 6.4, we conclude that if we do not add any extra restrictions other than the natural ones, the value of the Delsarte linear program remains unchanged.

Proposition 6.5 (Lifting). *For every finite field $\mathbb{F}$ and every $\ell\in\mathbb{N}_+$, we have*

$$\mathsf{val}(\mathrm{KrawtchoukLP}^{\mathbb{F}}(n,d,\ell))^{1/\ell} = \mathsf{val}(\mathrm{DelsarteLP}^{\mathbb{F}}(n,d)).$$

Proof. By Remark 5.28, the program $\mathrm{KrawtchoukLP}^{\mathbb{F}}(n,d,\ell)$ can be seen as the Delsarte linear program $\mathcal{L}_{\mathbb{H}_n(\mathbb{F})^{\ell,\mathbb{F}^\ell}}(\widehat{D}^\ell_d)$ of the refinement $\mathbb{H}_n(\mathbb{F})^{\ell,\mathbb{F}^\ell}$ of the tensor power $\mathbb{H}_n(\mathbb{F})^\ell$ using the natural restrictions

$$\widehat{D}^\ell_d := \{r\in R^{\ell,\mathbb{F}_2^\ell} \mid \exists r'\in R^{\otimes\ell}, r\subseteq r'\},$$

so the result follows from Lemmas 6.3 and 6.4. ■

As a secondary corollary, we can also show that in the linear case, the logarithm of the value of the hierarchy is subadditive. Let us note that this is also true of the hierarchy for non-linear codes for trivial reasons.

Corollary 6.6. *For every finite field $\mathbb{F}$ and every $\ell_1, \ell_2 \in \mathbb{N}_+$, we have*

$$\begin{aligned}\mathsf{val}(\mathrm{KrawtchoukLP}^{\mathbb{F}}_{\mathrm{Lin}}(n,d,\ell_1+\ell_2)) \\ \leqslant \mathsf{val}(\mathrm{KrawtchoukLP}^{\mathbb{F}}_{\mathrm{Lin}}(n,d,\ell_1)) \cdot \mathsf{val}(\mathrm{KrawtchoukLP}^{\mathbb{F}}_{\mathrm{Lin}}(n,d,\ell_2)),\end{aligned}$$

Proof. By Remark 5.28, the program $\mathrm{KrawtchoukLP}^{\mathbb{F}}_{\mathrm{Lin}}(n,d,\ell)$ can be seen as the Delsarte linear program $\mathcal{L}_{S_\ell}(D_d^{\ell,\mathbb{F}^\ell})$ of the translation scheme $S_\ell := \mathbb{H}_n(\mathbb{F})^{\ell,\mathbb{F}^\ell}$.

Note also that for $\ell_1, \ell_2 \in \mathbb{N}_+$, if

$$\widehat{D}_d^{\ell_1,\ell_2} := \{r \in R^{\ell_1+\ell_2,\mathbb{F}^{\ell_1+\ell_2}} \mid \exists r_1 \in D_d^{\ell_1,\mathbb{F}^{\ell_1}}, \exists r_2 \in D_d^{\ell_2,\mathbb{F}^{\ell_2}}, r \subseteq r_1 \otimes r_2\},$$

then we have

$$D_d^{\ell_1+\ell_2,\mathbb{F}^{\ell_1+\ell_2}} \subseteq \widehat{D}_d^{\ell_1,\ell_2},$$

and thus we get

$$\begin{aligned}&\mathsf{val}(\mathrm{KrawtchoukLP}^{\mathbb{F}}_{\mathrm{Lin}}(n,d,\ell_1+\ell_2)) \\ &= \mathsf{val}(\mathcal{L}_{S_{\ell_1+\ell_2}}(D_d^{\ell_1+\ell_2,\mathbb{F}^{\ell_1+\ell_2}})) \\ &\leqslant \mathsf{val}(\mathcal{L}_{S_{\ell_1+\ell_2}}(\widehat{D}_d^{\ell_1,\ell_2})) \\ &= \mathsf{val}(\mathcal{L}_{S_{\ell_1}}(D_d^{\ell_1,\mathbb{F}^{\ell_1}})) \cdot \mathsf{val}(\mathcal{L}_{S_{\ell_2}}(D_d^{\ell_2,\mathbb{F}^{\ell_2}})) \\ &= \mathsf{val}(\mathrm{KrawtchoukLP}^{\mathbb{F}}_{\mathrm{Lin}}(n,d,\ell_1)) \cdot \mathsf{val}(\mathrm{KrawtchoukLP}^{\mathbb{F}}_{\mathrm{Lin}}(n,d,\ell_2)),\end{aligned}$$

where the second equality follows from Lemmas 6.3 and 6.4. ∎

7 Conclusion

In this paper, we presented a pair of hierarchies of linear programs $\mathrm{KrawtchoukLP}^{\mathbb{F}}(n,d,\ell)$ and $\mathrm{KrawtchoukLP}^{\mathbb{F}}_{\mathrm{Lin}}(n,d,\ell)$ that provide upper bounds for the maximum size of codes and linear codes, respectively, of distance d in the weak Hamming scheme $\mathbb{H}_n(\mathbb{F})$ over a finite field $\mathbb{F}$. We showed that while the first hierarchy $\mathrm{KrawtchoukLP}^{\mathbb{F}}(n,d,\ell)$ collapses, the second hierarchy obtains the true value of the maximum code up to rounding by level $\ell = O(n^2)$. Finally, we also showed how to extend these hierarchy constructions to translation schemes under the mild assumption of factoring through types.

As we mentioned in the introduction, we view the main contribution of $\mathrm{KrawtchoukLP}_{\mathrm{Lin}}$ as being a hierarchy that is sufficiently powerful to ensure completeness while still being sufficiently simple to remain a hierarchy of linear programs (as opposed to SDPs), and bearing enough similarities with the original Delsarte's LP to be amenable to theoretical analysis. Thus the main open problem is to provide better upper or lower bounds to the optimum value of $\mathrm{KrawtchoukLP}_{\mathrm{Lin}}$.

The contrast between completeness of $\mathrm{KrawtchoukLP}_{\mathrm{Lin}}$ and collapse of KrawtchoukLP also surfaces a very natural question: are optimum codes very far from being linear? Along these lines, note that at level ℓ, $\mathrm{KrawtchoukLP}_{\mathrm{Lin}}(n,d,\ell)$ does not require full linearity of a code; namely, if $C \subseteq \mathbb{F}_2^n$ satisfies

$$\Delta\left(\sum_{j=1}^{t} x_j, \sum_{j=1}^{t} y_j\right) \notin [d-1], \tag{13}$$

for every $t \leqslant \ell$ and every $x_1,\dots,x_t,y_1,\dots,y_t \in C$, then a^C is a feasible solution of the program $\mathrm{KrawtchoukLP}_{\mathrm{Lin}}(n,d,\ell)$. For constant ℓ, the condition (13) is extremely mild and much weaker than C being a linear (or even affine) code. For example, if $0 \in C$, then (13) boils down to requiring sums of at most 2ℓ codewords from C to not have Hamming weight in $[d-1]$. This makes studying $\mathrm{KrawtchoukLP}_{\mathrm{Lin}}(n,d,\ell)$ at constant levels ℓ quite interesting.

In Theorem 6.1, we showed the (approximate) completeness of $\mathrm{KrawtchoukLP}^{\mathbb{F}}_{\mathrm{Lin}}(n,d,\ell)$ at level $O(n^2)$, via an unusual counting argument. The hierarchy does not have the same conceptual structure as Sum-of-Squares or Sherali–Adams, so completeness does not follow in the same way. In an earlier version of this manuscript, we conjectured that level n would have exact completeness, and we believe we now have a proof of this result[5]. It is plausible that exact completeness of $\mathrm{KrawtchoukLP}^{\mathbb{F}}_{\mathrm{Lin}}(n,d,\ell)$ can be attained at level $O(k_0)$, where k_0 is the dimension of an optimum linear code over $\mathbb{F}$ of distance d and blocklength n.

As we mentioned in the introduction, our techniques provide a higher-order version of the linear program responsible for the first linear programming bound in [MRRW77]. The second linear programming bound in [MRRW77] also consists of analyzing a Delsarte LP but for the Johnson scheme instead of the Hamming scheme. However, since the Johnson scheme is not a translation scheme, one cannot apply the theory developed in Section 5.2 directly. It is then natural to ask if there is a suitable generalization of this construction that would apply to non-translation schemes such as the Johnson scheme.

In Section 5.2, we showed how to generalize the hierarchy constructions to translation schemes under the assumption of factoring through types. However, in the general case it is not clear that the p and q-functions of $S^{\ell,T}$ can be computed efficiently even if those of S can be computed efficiently. For the particular case of the binary Hamming scheme, we obtained efficient formulas in Lemmas 3.19 and 3.20 (see also Proposition 3.21), but one can also compute the higher-order Krawtchouk polynomials efficiently from the usual Krawtchouk polynomials. This raises the natural question: for a translation scheme S in which $f_{S,T}$ factors through types, can the p and q-functions of $S^{\ell,T}$ be efficiently computed from the p and q-functions of S?

For the particular case of the strong Hamming scheme $\mathbb{H}^*_n(\mathbb{F})$ over an arbitrary finite field $\mathbb{F}$, an efficient formula for the higher-order $\mathbb{F}$-Krawtchouk polynomials can be obtained by generalizing Lemma 3.19: first, one generalizes the notion of Venn diagram configuration by saying that $(x_1,\dots,x_\ell) \in (\mathbb{F}^n)^\ell$ has $\mathbb{F}$-Venn diagram configuration $g\colon \mathbb{F}^\ell \to \{0,1,\dots,n\}$ if

$$g(t) = |\{i \in [n] \mid \forall j \in [\ell], (x_j)_i = t_j\}|$$

for every $t \in \mathbb{F}^\ell$. Lemma 5.30 implies that $f_{\mathbb{H}^*_n(\mathbb{F}),\mathbb{F}^\ell}(x) = f_{\mathbb{H}^*_n(\mathbb{F}),\mathbb{F}^\ell}(y)$ if and only if x and y have the same $\mathbb{F}$-Venn diagram configuration. By indexing the $\mathbb{F}$-Krawtchouk polynomials of order ℓ

[5]To give appropriate time for the verification of this proof, we leave it to a future work. We are including this note here to alert the interested reader that a proof might now be known.

by $\mathbb{F}$-Venn diagram configurations, a proof analogous to that of Lemma 3.19 gives

$$K_h(g) = \sum_{F \in \mathcal{F}} \prod_{t \in \mathbb{F}^\ell} \frac{g(t)!}{\prod_{u \in \mathbb{F}^\ell} F(t,u)!} \prod_{t,u \in \mathbb{F}^\ell} \chi_t(u)^{F(t,u)},$$

for all $\mathbb{F}$-Venn diagram configurations $h, g \colon \mathbb{F}^\ell \to \{0, 1, \ldots, n\}$, where $\mathcal{F}$ is the set of functions $F \colon \mathbb{F}^\ell \times \mathbb{F}^\ell \to \{0, 1, \ldots, n\}$ such that

$$\forall t \in \mathbb{F}^\ell, \sum_{u \in \mathbb{F}^\ell} F(t,u) = g(t),$$
$$\forall u \in \mathbb{F}^\ell, \sum_{t \in \mathbb{F}^\ell} F(t,u) = h(u).$$

One can also obtain efficient formulas for the $\mathbb{F}$-Krawtchouk polynomials of order ℓ in the weak Hamming scheme $\mathbb{H}_n(\mathbb{F})$ with similar methods (but the formulas are considerably more complicated).

References

[Del73] P. Delsarte. *An Algebraic Approach to the Association Schemes of Coding Theory*. Philips Journal of Research / Supplement. N.V. Philips' Gloeilampenfabrieken, 1973. 2, 9

[dKPS07] Etienne de Klerk, Dmitrii V. Pasechnik, and Alexander Schrijver. Reduction of symmetric semidefinite programs using the regular $*$-representation. *Math. Program.*, 109(2-3, Ser. B):613–624, 2007. doi:10.1007/s10107-006-0039-7. 5

[DL98] P. Delsarte and V. I. Levenshtein. Association schemes and coding theory. *IEEE Transactions on Information Theory*, 44(6):2477–2504, 1998. 9

[FT05] Joel Friedman and Jean-Pierre Tillich. Generalized Alon–Boppana theorems and error-correcting codes. *SIAM J. Discret. Math.*, 19(3):700–718, July 2005. 4

[Gij09] Dion Gijswijt. Block diagonalization for algebra's associated with block codes, 2009. arXiv:0910.4515. 19

[Gil52] E.N. Gilbert. A comparison of signalling alphabets. *Bell System Technical Journal*, 31:504–522, 1952. 1

[GMS12] D. C. Gijswijt, H. D. Mittelmann, and A. Schrijver. Semidefinite code bounds based on quadruple distances. *IEEE Transactions on Information Theory*, 58(5):2697–2705, 2012. 5, 19

[Lau07] Monique Laurent. Strengthened semidefinite programming bounds for codes. *Mathematical Programming*, 109:1436–4646, 2007. 5

[Lau09] Monique Laurent. Sums of squares, moment matrices and optimization over polynomials. In *Emerging Applications of Algebraic Geometry (of IMA Volumes in Mathematics and its Applications)*. Springer, 2009. 2

[Mac63] Jessie MacWilliams. A theorem on the distribution of weights in a systematic code†. *Bell System Technical Journal*, 42(1):79–94, 1963. 1

[MRRW77] R. McEliece, E. Rodemich, H. Rumsey, and L. Welch. New upper bounds on the rate of a code via the Delsarte-MacWilliams inequalities. *IEEE Transactions on Information Theory*, 23(2):157–166, 1977. 2, 4, 46

[MSG72] Mrs. F. J. MacWilliams, N. J. A. Sloane, and J.M. Goethals. The MacWilliams identities for nonlinear codes. *The Bell System Technical Journal*, 51(4):803–819, 1972. 2

[MT09] William J. Martin and Hajime Tanaka. Commutative association schemes. *European Journal of Combinatorics*, 30(6):1497–1525, 2009. 20

[NS05] M. Navon and A. Samorodnitsky. On Delsarte's linear programming bounds for binary codes. In *46th Annual IEEE Symposium on Foundations of Computer Science (FOCS'05)*, pages 327–336, 2005. 2, 3, 4

[NS09] Michael Navon and Alex Samorodnitsky. Linear programming bounds for codes via a covering argument. *Discrete Comput. Geom.*, 41(2):199–207, March 2009. 4

[Sam21] Alex Samorodnitsky. One more proof of the first linear programming bound for binary codes and two conjectures, 2021. arXiv:2104.14587. 2, 4

[Sch79] A. Schrijver. A comparison of the Delsarte and Lovász bounds. *IEEE Transactions on Information Theory*, 25(4):425–429, 1979. 4, 25

[Sch05] A. Schrijver. New code upper bounds from the Terwilliger algebra and semidefinite programming. *IEEE Transactions on Information Theory*, 51(8):2859–2866, 2005. 2, 4, 5, 19

[Val19] Frank Vallentin. Semidefinite programming bounds for error-correcting codes. *CoRR*, abs/1902.01253, 2019. URL: http://arxiv.org/abs/1902.01253, arXiv:1902.01253. 4, 14

[Var57] R.R. Varshamov. Estimate of the number of signals in error correcting codes. *Doklady Akademii Nauk SSSR*, 117:739–741, 1957. 1

[vL99] Jacobus H. van Lint. *Introduction to Coding Theory*. Springer-Verlag, 1999. 1

A Deferred Binary Case Proofs

Since the proofs of Lemmas 3.3 and 3.4 use Lemma 3.15, we postpone them until after the proof of the latter.

Lemma 3.14. *For every $n, \ell \in \mathbb{N}_+$, we have*

$$\operatorname{im}(\mathsf{Config}^V_{n,\ell}) = \left\{ g \colon 2^{[\ell]} \to \mathbb{R} \;\middle|\; \sum_{J \subseteq [\ell]} g(J) = n \wedge \forall J \subseteq [\ell], g(J) \in \mathbb{N} \right\}. \tag{2}$$

Proof. It is obvious that every Venn diagram configuration g is in the set in right-hand side of (2). On the other hand, if g is in the set in the right-hand side of (2), then the hypotheses imply that we can find a partition $(X_J)_{J\subseteq[\ell]}$ of $[n]$ into 2^ℓ parts such that $|X_J| = g(J)$. It is easy to see that the words $z_1, \dots, z_\ell \in \mathbb{F}_2^n$ defined by

$$(z_j)_i \stackrel{\text{def}}{=} \mathbb{1}[\exists J \subseteq [\ell], (j \in J \wedge i \in X_J)] \qquad (i \in [n], j \in [\ell])$$

have Venn diagram configuration g. ∎

Lemma 3.15. *Let $n, \ell \in \mathbb{N}_+$, let*

$$S_{n,\ell} \stackrel{\text{def}}{=} \left\{ g \in \mathbb{R}^{2^{[\ell]}} \;\middle|\; \sum_{J\subseteq[\ell]} g(J) = n \right\}, \qquad Z_{n,\ell} \stackrel{\text{def}}{=} \{g \in \mathbb{R}^{2^{[\ell]}} \mid g(\varnothing) = 0\}$$

and let $V_{n,\ell}\colon Z_{n,\ell} \to S_{n,\ell}$ and $D_{n,\ell}\colon S_{n,\ell} \to Z_{n,\ell}$ be given by

$$D_{n,\ell}(g)(J) \stackrel{\text{def}}{=} \sum_{\substack{T\subseteq[\ell] \\ |T\cap J| \text{ odd}}} g(T), \tag{3}$$

$$V_{n,\ell}(g)(J) \stackrel{\text{def}}{=} n \cdot \mathbb{1}[J = \varnothing] + 2^{1-\ell} \sum_{T\subseteq[\ell]} (-1)^{|T\cap J|-1} g(T), \tag{4}$$

for every $J \subseteq [\ell]$.

Then $V_{n,\ell}$ and $D_{n,\ell}$ are inverses of each other and $\mathsf{Config}^{\Delta}_{n,\ell} = D_{n,\ell} \circ \mathsf{Config}^{V}_{n,\ell}$ and $\mathsf{Config}^{V}_{n,\ell} = V_{n,\ell} \circ \mathsf{Config}^{\Delta}_{n,\ell}$.

Proof. First note that for $g \in Z_{n,\ell}$, we have

$$\sum_{J\subseteq[\ell]} V_{n,\ell}(g)(J) = n + 2^{1-\ell} \sum_{T\subseteq[\ell]} g(T) \sum_{J\subseteq[\ell]} (-1)^{|T\cap J|-1} = n - 2^{1-\ell} \sum_{T\subseteq[\ell]} g(\varnothing) 2^\ell = n,$$

so $V_{n,\ell}$ is well-defined. Since for $g \in S_{n,\ell}$, we clearly have $D_{n,\ell}(g)(\varnothing) = 0$, it follows that $D_{n,\ell}$ is also well-defined.

Let now $g \in S_{n,\ell}$ and note that

$$\begin{aligned} V_{n,\ell}(D_{n,\ell}(g))(J) &= n \cdot \mathbb{1}[J = \varnothing] + 2^{1-\ell} \sum_{T\subseteq[\ell]} (-1)^{|T\cap J|-1} \sum_{\substack{K\subseteq[\ell] \\ |K\cap T| \text{ odd}}} g(K) \\ &= n \cdot \mathbb{1}[J = \varnothing] + 2^{1-\ell} \sum_{K\subseteq[\ell]} g(K) \sum_{\substack{T\subseteq[\ell] \\ |K\cap T| \text{ odd}}} (-1)^{|T\cap J|-1}. \end{aligned} \tag{14}$$

But note that

$$\sum_{\substack{T\subseteq[\ell]\\ |K\cap T| \text{ odd}}} (-1)^{|T\cap J|-1} = \sum_{T\subseteq[\ell]} (-1)^{|T\cap J|-1}\frac{1-(-1)^{|K\cap T|}}{2}$$
$$= \frac{1}{2}\left(-\sum_{T\subseteq[\ell]}(-1)^{|T\cap J|} + \sum_{T\subseteq[\ell]}(-1)^{|T\cap(J\triangle K)|}\right)$$
$$= 2^{\ell-1}(-\mathbb{1}[J=\varnothing]+\mathbb{1}[J=K]),$$

so plugging this in (14), we get

$$V_{n,\ell}(D_{n,\ell}(g))(J) = n\cdot\mathbb{1}[J=\varnothing] + \sum_{K\subseteq[\ell]} g(K)(-\mathbb{1}[J=\varnothing]+\mathbb{1}[J=K]) = g(J),$$

where the second equality follows since $\sum_{K\subseteq[\ell]} g(K) = n$ as $g \in S_{n,\ell}$.

Therefore $V_{n,\ell}$ is a left-inverse of $D_{n,\ell}$. But since both $S_{n,\ell}$ and $Z_{n,\ell}$ are $\mathbb{R}$-linear subspaces of dimension $2^\ell - 1$ and $V_{n,\ell}$ and $D_{n,\ell}$ are $\mathbb{R}$-linear, it follows that $V_{n,\ell}$ and $D_{n,\ell}$ are inverses of each other.

By Lemma 3.14, we know that $\operatorname{im}(\mathsf{Config}^V_{n,\ell}) \subseteq S_{n,\ell}$. On the other hand, if $g \in \operatorname{im}(\mathsf{Config}^V_{n,\ell})$ and $\mathsf{Config}^V_{n,\ell}(z_1,\dots,z_\ell) = g$, then it is straightforward to check that $\mathsf{Config}^\Delta_{n,\ell}(z_1,\dots,z_\ell) = D_{n,\ell}(g)$, thus $\mathsf{Config}^\Delta_{n,\ell} = D_{n,\ell}\circ\mathsf{Config}^V_{n,\ell}$. Applying $V_{n,\ell}$ to both sides, we get $V_{n,\ell}\circ\mathsf{Config}^\Delta_{n,\ell} = \mathsf{Config}^V_{n,\ell}$. ■

Lemma 3.3. *We have*

$$\left|\operatorname{im}(\mathsf{Config}^\Delta_{n,\ell})\right| = \binom{n+2^\ell-1}{2^\ell-1}.$$

Proof. By Lemma 3.15, it is sufficient to prove that $|\operatorname{im}(\mathsf{Config}^V_{n,\ell})| = \binom{n+2^\ell-1}{2^\ell-1}$. But the number of valid Venn diagram configurations is easy to count using Lemma 3.14: it is exactly the number of partitions of n indistinguishable objects into 2^ℓ distinguishable parts, which is $\binom{n+2^\ell-1}{2^\ell-1}$. ■

Lemma 3.4. *Let $n,\ell \in \mathbb{N}_+$ and consider the natural (diagonal) right action of S_n on $(\mathbb{F}_2^n)^\ell$ given by $(x_1,\dots,x_\ell)\cdot\sigma := (y_1,\dots,y_\ell)$, where $(y_j)_i := (x_j)_{\sigma(i)}$ ($(x_1,\dots,x_\ell),(y_1,\dots,y_\ell)\in(\mathbb{F}_2^n)^\ell$, $\sigma\in S_n$, $j\in[\ell]$, $i\in[n]$).*

The following are equivalent for $(x_1,\dots,x_\ell),(y_1,\dots,y_\ell)\in(\mathbb{F}_2^n)^\ell$.

i. $(x_1,\dots,x_\ell)$ and $(y_1,\dots,y_\ell)$ are in the same S_n-orbit.

ii. $\mathsf{Config}^\Delta_{n,\ell}(x_1,\dots,x_\ell) = \mathsf{Config}^\Delta_{n,\ell}(y_1,\dots,y_\ell)$.

Proof. By Lemma 3.15, item (ii) is equivalent to:

iii. $\mathsf{Config}^V_{n,\ell}(x_1,\dots,x_\ell) = \mathsf{Config}^V_{n,\ell}(y_1,\dots,y_\ell)$.

Let us prove that (i)⇒(iii), if $(y_1, \ldots, y_\ell) = (x_1, \ldots, x_\ell) \cdot \sigma$ for some $\sigma \in S_n$, then for every $J \subseteq [\ell]$, we have

$$\begin{aligned}\mathsf{Config}^V_{n,\ell}(y_1, \ldots, y_\ell) &= |\{i \in [n] \mid \{j \in [\ell] \mid (y_j)_i = 1\} = J\}| \\ &= |\{i \in [n] \mid \{j \in [\ell] \mid (x_j)_{\sigma(i)} = 1\} = J\}| = \mathsf{Config}^V_{n,\ell}(x_1, \ldots, x_\ell).\end{aligned}$$

To show (iii)⇒(i), let $(X_J)_{J \subseteq [\ell]}$ and $(Y_J)_{J \subseteq [\ell]}$ be the partitions corresponding to $(x_1, \ldots, x_\ell)$ and $(y_1, \ldots, y_\ell)$, respectively, given by

$$\begin{aligned}X_J &\stackrel{\text{def}}{=} \bigcap_{j \in J} \mathrm{supp}(x_j) \cap \bigcap_{j \in [\ell] \setminus J} ([n] \setminus \mathrm{supp}(x_j)), \\ Y_J &\stackrel{\text{def}}{=} \bigcap_{j \in J} \mathrm{supp}(y_j) \cap \bigcap_{j \in [\ell] \setminus J} ([n] \setminus \mathrm{supp}(y_j)).\end{aligned}$$

Since $\mathsf{Config}^V_{n,\ell}(x_1, \ldots, x_\ell) = \mathsf{Config}^V_{n,\ell}(y_1, \ldots, y_\ell)$, it follows that $|X_J| = |Y_J|$ for every $J \subseteq [\ell]$, so there exists a permutation $\sigma \in S_n$ such that $\sigma(X_J) = Y_J$ for every $J \subseteq [\ell]$. Since

$$\begin{aligned}(x_j)_i &\stackrel{\text{def}}{=} \mathbb{1}[\exists J \subseteq [\ell], j \in J \wedge i \in X_J], \\ (y_j)_i &\stackrel{\text{def}}{=} \mathbb{1}[\exists J \subseteq [\ell], j \in J \wedge i \in Y_J],\end{aligned}$$

for every $j \in [\ell]$ and every $i \in [n]$, it follows that $(x_1, \ldots, x_\ell) \cdot \sigma = (y_1, \ldots, y_\ell)$. ■

Lemma 3.16. *For a symmetric difference configuration $g \in \mathrm{im}(\mathsf{Config}^\Delta_{n,\ell})$, we have*

$$|g| = K_g(0) = \binom{n}{V_{n,\ell}(g)} = \frac{n!}{\prod_{J \subseteq [\ell]} V_{n,\ell}(g)(J)!},$$

where $V_{n,\ell}$ is given by (4).

Proof. By Lemma 3.15, $|g|$ is precisely the number of $(z_1, \ldots, z_\ell) \in \mathbb{F}_2^n$ whose Venn diagram configuration is $V_{n,\ell}(g)$. But the set of such $(z_1, \ldots, z_\ell)$ is naturally in bijection with the set of partitions $(X_J)_{J \subseteq [\ell]}$ of $[n]$ such that $|X_J| = V_{n,\ell}(g)(J)$ ($J \subseteq [\ell]$) and the number of the latter is clearly the multinomial

$$\binom{n}{V_{n,\ell}(g)} = \frac{n!}{\prod_{J \subseteq [\ell]} V_{n,\ell}(g)(J)!}.$$

Finally, from (1), we also have

$$K_g(0) = \sum_{(y_1, \ldots, y_\ell) \in g} \prod_{j=1}^{\ell} \chi_{y_j}(0) = |g|.$$ ■

Lemma 3.17. *[Orthogonality] For $n, \ell \in \mathbb{N}_+$ and $h, h' \in \mathrm{im}(\mathsf{Config}^\Delta_{n,\ell})$, we have*

$$\sum_{g \in \mathrm{im}(\mathsf{Config}^\Delta_{n,\ell})} |g| \cdot K_h(g) \cdot K_{h'}(g) = 2^{\ell n} \cdot |h| \cdot \mathbb{1}[h = h'].$$

Proof. By Remark 3.8, we have

$$K_h(g) = 2^{\ell n} \cdot \widehat{\mathbb{1}_h}(x), \qquad K_{h'}(g) = 2^{\ell n} \cdot \widehat{\mathbb{1}_{h'}}(x),$$

and thus we have

$$\begin{aligned}\sum_{g \in \mathrm{im}(\mathsf{Config}^{\Delta}_{n,\ell})} |g| \cdot K_h(g) \cdot K_{h'}(g) &= \sum_{x \in (\mathbb{F}_2^n)^\ell} 2^{2\ell n} \cdot \widehat{\mathbb{1}_h}(x) \cdot \widehat{\mathbb{1}_{h'}}(x) \\ &= 2^{2\ell n} \cdot \langle \mathbb{1}_h, \mathbb{1}_{h'} \rangle \\ &= 2^{\ell n} \cdot |h| \cdot \mathbb{1}[h = h'],\end{aligned}$$

as desired. ∎

Since the proof of Lemma 3.18 uses Lemma 3.19, we prove the latter first.

Lemma 3.19. *For every $n, \ell \in \mathbb{N}_+$ and every $g, h \in \mathrm{im}(\mathsf{Config}^{\Delta}_{n,\ell})$, we have*

$$K_h(g) = \sum_{F \in \mathcal{F}} \prod_{J \subseteq [\ell]} \frac{V_{n,\ell}(g)(J)!}{\prod_{K \subseteq [\ell]} F(J,K)!} \cdot \prod_{j=1}^{\ell} \prod_{\substack{J,K \subseteq [\ell] \\ j \in J \cap K}} (-1)^{F(J,K)},$$

where $\mathcal{F}$ is the set of functions $F \colon 2^{[\ell]} \times 2^{[\ell]} \to \{0, 1, \ldots, n\}$ such that

$$\begin{aligned}\forall J \subseteq [\ell], \sum_{K \subseteq [\ell]} F(J,K) &= V_{n,\ell}(g)(J), \\ \forall K \subseteq [\ell], \sum_{J \subseteq [\ell]} F(J,K) &= V_{n,\ell}(h)(K),\end{aligned}$$

and $V_{n,\ell}$ is given by (4).

Proof. For an ℓ-tuple $z = (z_1, \ldots, z_\ell) \in (\mathbb{F}_2^n)^\ell$, let $P^z = (P^z_J)_{J \subseteq [\ell]}$ be the natural partition of $[n]$ associated with z given by

$$P^z_J := \bigcap_{j \in J} \mathrm{supp}(z_j) \cap \bigcap_{j \in [\ell] \setminus J} ([n] \setminus \mathrm{supp}(z_j)).$$

Note that by Lemma 3.15, if the symmetric difference configuration of z is some function f, then it has Venn diagram configuration $V_{n,\ell}(f)$ and thus $|P^z_J| = V_{n,\ell}(f)(J)$ for every $J \subseteq [\ell]$.

Fix an ℓ-tuple $x = (x_1, \ldots, x_\ell) \in g$ whose symmetric difference configuration is g. We now classify the ℓ-tuples $y = (y_1, \ldots, y_\ell) \in h$ of symmetric difference configuration h based on how the partitions P^x and P^y interact; namely, to each such y we associate the function $F_y \colon 2^{[\ell]} \times 2^{[\ell]} \to \{0, 1, \ldots, n\}$ given by

$$F_y(J,K) := |P^x_J \cap P^y_K|$$

for every $J, K \subseteq [\ell]$.

By our previous observations, we know that for every $J \subseteq [\ell]$, we have

$$\sum_{K \subseteq [\ell]} F_y(J,K) = \sum_{K \subseteq [\ell]} |P_J^x \cap P_K^y| = |P_J^x| = V_{n,\ell}(g)(J).$$

Similarly, we know that for every $K \subseteq [\ell]$, we have

$$\sum_{J \subseteq [\ell]} F_y(J,K) = \sum_{J \subseteq [\ell]} |P_J^x \cap P_K^y| = |P_K^y| = V_{n,\ell}(h)(K).$$

Therefore $F_y \in \mathcal{F}$.

Note further that for each $j \in [\ell]$ we have

$$\operatorname{supp}(x_j) \cap \operatorname{supp}(y_j) = \bigcup_{\substack{J,K \subseteq [\ell] \\ j \in J \cap K}} P_J^x \cap P_K^y,$$

so in the formula (1), the summand of $y \in h$ is given by

$$\prod_{j=1}^{\ell} \chi_{y_j}(x_j) = \prod_{j=1}^{\ell} (-1)^{\operatorname{supp}(x_j) \cap \operatorname{supp}(y_j)} = \prod_{j=1}^{\ell} \prod_{\substack{J,K \subseteq [\ell] \\ j \in J \cap K}} (-1)^{F(J,K)}.$$

For each $F \in \mathcal{F}$, let n_F be the number of $y \in h$ such that $F_y = F$. It is easy to compute n_F from the definition of F_y: since $F_y = F$ if and only if the partition $(P_J^x \cap P_K^y)_{J,K \subseteq [\ell]}$ satisfies $|P_J^x \cap P_K^y| = F(J,K)$, it follows that to get $F = F_y$, each part P_J^x (whose size is $V_{n,\ell}(g)$) has to be partitioned into 2^ℓ parts of sizes $(F(J,K))_{K \subseteq [\ell]}$ and thus n_F is given by the following product of multinomials

$$n_F = \prod_{J \subseteq [\ell]} \binom{V_{n,\ell}(g)(J)}{F(J,\cdot)} = \prod_{J \subseteq [\ell]} \frac{V_{n,\ell}(g)(J)!}{\prod_{K \subseteq [\ell]} F(J,K)!}.$$

Putting everything together, we get

$$\begin{aligned} K_h(g) &= \sum_{F \in \mathcal{F}} n_F \cdot \prod_{j=1}^{\ell} \prod_{\substack{J,K \subseteq [\ell] \\ j \in J \cap K}} (-1)^{F(J,K)} \\ &= \sum_{F \in \mathcal{F}} \prod_{J \subseteq [\ell]} \frac{V_{n,\ell}(g)(J)!}{\prod_{K \subseteq [\ell]} F(J,K)!} \cdot \prod_{j=1}^{\ell} \prod_{\substack{J,K \subseteq [\ell] \\ j \in J \cap K}} (-1)^{F(J,K)}, \end{aligned}$$

as desired. ∎

Lemma 3.18. *[Reflection] For $n, \ell \in \mathbb{N}_+$ and $g, h \in \operatorname{im}(\mathsf{Config}_{n,\ell}^{\Delta})$, we have*

$$\frac{K_h(g)}{|h|} = \frac{K_g(h)}{|g|}.$$

Proof. Let $V_{n,\ell}$ the function of Lemma 3.15 given by (4). By Lemma 3.16, we have

$$\frac{|g|}{|h|} = \frac{\binom{n}{V_{n,\ell}(g)}}{\binom{n}{V_{n,\ell}(h)}} = \prod_{J\subseteq[\ell]} \frac{V_{n,\ell}(h)(J)!}{V_{n,\ell}(g)(J)!}$$

and thus by using the formula of Lemma 3.19, we get

$$\frac{|g|}{|h|} \cdot K_h(g) = K_g(h),$$

which gives the result. ∎

Lemma 3.20. *Let $n, \ell \in \mathbb{N}_+$ with $n \geqslant 2$, let $g, h \in \mathrm{im}(\mathsf{Config}^{\Delta}_{n,\ell})$ be symmetric difference configurations and let $J_0 \subseteq [\ell]$ be such that $V_{n,\ell}(g)(J_0) > 0$ for $V_{n,\ell}$ given by (4). Then*

$$K_h(g) = \sum_{\substack{K_0\subseteq[\ell]\\ V_{n,\ell}(h)(K_0)>0}} (-1)^{|J_0\cap K_0|} \cdot K_{h\ominus K_0}(g \ominus J_0), \tag{5}$$

$$K_h(g) = -\sum_{\substack{K_0\subseteq[\ell]\\ V(h)(K_0)>0\\ K_0\neq\varnothing}} K_{h\oplus\varnothing\ominus K_0}(g) + \sum_{\substack{K_0\subseteq[\ell]\\ V(h)(K_0)>0}} (-1)^{|J_0\cap K_0|} \cdot K_{h\oplus\varnothing\ominus K_0}(g\oplus\varnothing\ominus J_0), \tag{6}$$

where

$$h \ominus K_0 \coloneqq D_{n-1,\ell}(V_{n,\ell}(h) - \mathbb{1}_{\{K_0\}}), \qquad g \ominus J_0 \coloneqq D_{n-1,\ell}(V_{n,\ell}(g) - \mathbb{1}_{\{J_0\}}),$$
$$h \oplus \varnothing \coloneqq D_{n+1,\ell}(V_{n,\ell}(h) + \mathbb{1}_{\{\varnothing\}}), \qquad g \oplus \varnothing \coloneqq D_{n+1,\ell}(V_{n,\ell}(g) + \mathbb{1}_{\{\varnothing\}}),$$

and $D_{n-1,\ell}$ and $D_{n+1,\ell}$ are given by (3).

Proof. We start by proving (5).

First note that if $K_0 \subseteq [\ell]$ is such that $V_{n,\ell}(h)(K_0) > 0$ for some symmetric difference configuration $h \in \mathrm{im}(\mathsf{Config}^{\Delta}_{n,\ell})$, then Lemmas 3.14 and 3.15 imply that $V_{n,\ell}(h) - \mathbb{1}_{\{K_0\}}$ is a Venn diagram configuration in the space $\mathbb{F}_2^{n-1}$ (and level ℓ) and thus $h \ominus K_0 = D_{n-1,\ell}(V_{n,\ell}(h) - \mathbb{1}_{\{K_0\}})$ is a symmetric difference configuration in the space $\mathbb{F}_2^{n-1}$. This also shows that $g \ominus J_0$ is a symmetric difference configuration in the space $\mathbb{F}_2^{n-1}$.

Let us denote by $\mathcal{F}_{g,h}$ the set of functions $F\colon 2^{[\ell]} \times 2^{[\ell]} \to \{0, 1, \ldots, n\}$ such that

$$\forall J \subseteq [\ell], \sum_{K\subseteq[\ell]} F(J,K) = V_{n,\ell}(g)(J),$$
$$\forall K \subseteq [\ell], \sum_{J\subseteq[\ell]} F(J,K) = V_{n,\ell}(h)(K).$$

We define $\mathcal{F}_{g\ominus J_0, h\ominus K_0}$ analogously (replacing n with $n-1$).

By Lemma 3.19, we have

$$K_h(g) = \sum_{F\in\mathcal{F}_{g,h}} \prod_{J\subseteq[\ell]} \frac{V_{n,\ell}(g)(J)!}{\prod_{K\subseteq[\ell]} F(J,K)!} \cdot \prod_{j=1}^{\ell} \prod_{\substack{J,K\subseteq[\ell]\\ j\in J\cap K}} (-1)^{F(J,K)}.$$

Using the multinomial identity

$$\frac{V_{n,\ell}(g)(J_0)!}{\prod_{K\subseteq[\ell]} F(J_0,K)!} = \binom{V_{n,\ell}(g)(J_0)}{F(J_0,\,\cdot\,)} = \sum_{\substack{K_0\subseteq[\ell]\\ F(J_0,K_0)>0}} \frac{(V_{n,\ell}(g)(J_0)-1)!}{(F(J_0,K_0)-1)!\prod_{\substack{K\subseteq[\ell]\\K\neq K_0}} F(J_0,K)!}$$

and noting that $F(J_0,K_0)>0$ implies $V(h)(K_0)>0$, we obtain

$$\begin{aligned} K_h(g) &= \sum_{\substack{K_0\subseteq[\ell]\\ V(h)(K_0)>0}} \sum_{\substack{F\in\mathcal{F}_{g,h}\\ F(J_0,K_0)>0}} \frac{(V_{n,\ell}(g)(J_0)-1)!}{(F(J_0,K_0)-1)!\prod_{\substack{K\subseteq[\ell]\\K\neq K_0}} F(J_0,K)!}, \cdot \prod_{j=1}^{\ell} \prod_{\substack{J,K\subseteq[\ell]\\ j\in J\cap K}} (-1)^{F(J,K)} \\ &= \sum_{K_0\subseteq[\ell]} \sum_{F'\in\mathcal{F}_{g\ominus J_0,h\ominus K_0}} \prod_{J\subseteq[\ell]} \frac{V_{n-1,\ell}(g\ominus J_0)(J)!}{\prod_{K\subseteq[\ell]} F'(J,K)!} \cdot \prod_{j=1}^{\ell} \prod_{\substack{J,K\subseteq[\ell]\\ j\in J\cap K}} (-1)^{F'(J,K)} \cdot (-1)^{|J_0\cap K_0|} \end{aligned}$$

where the second equality follows from the substitution corresponding to the bijection

$$\{F\in\mathcal{F}_{g,h} \mid F(J_0,K_0)>0\} \to \mathcal{F}_{g\ominus J_0,h\ominus K_0}$$

that maps F to $F' := F - \mathbb{1}_{\{(J_0,K_0)\}}$.

Equation (5) now follows by applying Lemma 3.19 again.

Note now that since $h\oplus\varnothing\ominus\varnothing = h$, equation (6) is equivalent to

$$\sum_{\substack{K_0\subseteq[\ell]\\ V(h)(K_0)>0}} K_{h\oplus\varnothing\ominus K_0}(g) = \sum_{\substack{K_0\subseteq[\ell]\\ V(h)(K_0)>0}} (-1)^{|J_0\cap K_0|}\cdot K_{h\oplus\varnothing\ominus K_0}(g\oplus\varnothing\ominus J_0).$$

By (5), both sides of the above are equal to $K_{h\oplus\varnothing}(g\oplus\varnothing)$: the left-hand side using (5) with $J_0=\varnothing$ and the right-hand side using $J_0 = J_0$. ■

B Deferred Computations

Lemma B.1. *Let $\varepsilon\in(0,1)$ and $n,q\in\mathbb{N}_+$ be positive integers with $q\geq 2$. For $\ell \geqslant 9(n^2\ln(q)+1)/(\ln(1+\varepsilon))^2$, we have*

$$(2\ell^n q^{n^2})^{1/\ell} \leqslant 1+\varepsilon.$$

Proof. The statement is equivalent to

$$\ln 2 + n\ln\ell + n^2\ln q \leqslant \ell\ln(1+\varepsilon),$$

which in turn is equivalent to

$$\frac{n^2\ln q+\ln 2}{\ln(1+\varepsilon)} \leqslant \ell\left(1-\frac{\ln\ell}{\ell}\cdot\frac{n}{\ln(1+\varepsilon)}\right). \tag{15}$$

We claim that it is sufficient to show that

$$\frac{\ln \ell}{\ell} \leqslant \frac{8}{9} \cdot \frac{\ln(1+\varepsilon)}{n}. \tag{16}$$

Indeed, if this is the case, then the right-hand side of (15) is at least $\ell/9$, which in turn is at least $(n^2 \ln(q)+1)/(\ln(1+\varepsilon))^2$ and thus (15) follows from $\ln(1+\varepsilon) \leqslant \ln 2 \leqslant 1$.

To show (16), first note that the function $f(t) \stackrel{\text{def}}{=} \ln(t)/t$ is decreasing when $t \geqslant e$ and since

$$\ell \geqslant \frac{9(n^2 \ln(q)+1)}{(\ln(1+\varepsilon))^2} \geqslant \frac{9n^2 \ln(q)}{(\ln(1+\varepsilon))^2} \geqslant e,$$

it is sufficient to prove that

$$f\left(\frac{9n^2 \ln(q)}{(\ln(1+\varepsilon))^2}\right) \leqslant \frac{8}{9} \cdot \frac{\ln(1+\varepsilon)}{n}. \tag{17}$$

But since

$$f\left(\frac{9n^2 \ln(q)}{(\ln(1+\varepsilon))^2}\right) = \frac{\ln 9 + 2\ln n + \ln\ln q + 2\ln\dfrac{1}{\ln(1+\varepsilon)}}{\dfrac{9n^2 \ln(q)}{\ln(1+\varepsilon)^2}},$$

(17) is equivalent to

$$\ln 9 + 2\ln n + \ln\ln q + 2\ln\frac{1}{\ln(1+\varepsilon)} \leqslant \frac{8n\ln(q)}{\ln(1+\varepsilon)}.$$

This is clearly true by recalling that $\ln(1+\varepsilon) \leqslant 1$ and upper bounding the terms on the left-hand side of the above respectively by

$$3, \qquad 2n, \qquad \ln(q), \qquad \frac{2}{\ln(1+\varepsilon)}.$$

The last three bounds follow from $\ln x \leqslant x$ for $x > 0$. ■