Intertwined Results on Linear Codes and Galois Geometries

Peter Vandendriessche

promotor: prof. dr. Leo Storme

Contents

Summary

The curious interweaving of Galois geometry and linear codes has been inspiring researchers for decades to explore the many intersections of these two topics. This interweaving is also the core theme of this PhD thesis, in which I will be presenting results which all in a sense revolve around the intertwining of linear codes and Galois geometries.

Chapter 1 repeats the general notations in finite geometry and coding theory. This chapter introduces the notations used throughout this thesis.

Chapter 2 is about (LDPC) codes derived from finite geometries. In Section 2.1, I discuss the motivation to study LDPC codes from finite geometries, as well as provide some general preliminaries on this topic, which appear in several of the sections afterwards.

In Section 2.2, I discuss the cyclic LDPC codes constructed from affine and projective geometries, as this is the most well-known class of finite geometry LDPC codes. In this section I also present some results which, even though they look like they should have been discovered decades ago, I have not been able to find in the literature, so which I will assume to be new. In particular, we show that the order of this code as a cyclic code is equal to the length of the code, and we study what happens when removing the all-one vector from the code in a canonical way, resulting in an additional application of my earlier paper [94] with J. Limbupasiriporn and L. Storme, which will be discussed in Section 7.2. The results presented in this section are (a small) part of joint work with Y. Fujiwara, which is submitted to IEEE Trans. Inform. Theory [45].

In Section 2.3, I discuss the LDPC codes from points and lines in linear representations $T_2^*(\mathcal{K})$. My first results on this topic date back to before my official research started, as a spin-off of an assignment during my bachelor's studies. Here, I showed that when $\mathcal{K}$ is an arc and the characteristic of the field of the code is different from that of the field of the underlying geometry, and another condition on the finite field is met, then the code is entirely generated by its code words of minimum weight. Moreover, from this I could compute the actual dimension of the code. Earlier work by Pepe et al. [113] demonstrated this behavior only up to a certain small Hamming weight, and needed in some cases two types of small weight code words to generate them, instead of one (hence my result implies that code words of their second type are also a linear combination of code words of their first type). This result was published in Des. Codes Cryptogr. [139]. Unfortunately, the "another condition on the finite field" excluded the important case of binary codes. In the special case that $\mathcal{K}$ is a conic minus one point, P. Sin and Q. Xiang showed the dimension formula for the case of binary

codes [124]. Later, I managed to improve my technique and show that when the characteristic of the field of the code is different from that of the field of the underlying geometry, then the code is entirely generated by its code words of minimum weight, and the same dimension formula remains valid in this more general setting. This new result improves on all of the previous results: it generalized my own result from [140], it embedded the formula from [124] in a large infinite class of geometries and gave a structural geometric explanation for it, and it extends the linear combination property from [113] without weight restriction, as well as sharpening several minimum distance bounds from [113]. These are the results that will be presented in this section; they have been published in Adv. Math. Commun. [140].

In Section 2.4, I discuss the LDPC codes from generators and points in Hermitian varieties $\mathcal{H}(2n+1,q)$ with q sufficiently large. This class of codes was studied before in [114], where for $n=1$ the code words of small weight are classified up to weight roughly $\frac{1}{2}q^{3/2}$ as a linear combination of code words of weight $2(q+1)$, and for $n=2$ it is shown that the only code words c having $0<\mathrm{wt}(c)\leq 2(q^3+q^2)$ have weight $2(q^3+1)$ and $2(q^3+q^2)$ and each comes from one specific construction. We extend the second result to arbitrary n and add a classification similar to the first result: the only code words c having $0<\mathrm{wt}(c)\leq 4q^{2n-2}(q-1)$ arise from n different constructions, the minimum distance is in general $2q^{2n-4}(q^3+1)$ for $n\geq 2$; and for every $\delta>0$ the code words up to weight δq^{2n-1} are a linear combination of these n smallest types, when q is sufficiently large compared to δ. These results are joint work with M. De Boeck and are accepted for publication in Adv. Math. Commun. [30].

In Section 2.5, I discuss two more infinite families of LDPC codes, derived from the partial geometries $S(\mathcal{K})$ and $T_2^*(\mathcal{K})$ with $\mathcal{K}$ a maximal arc. For the first construction, I show that swapping the role of points and lines yields an equivalent code, and I extend an earlier result from [23] from hyperovals to Denniston arcs and Mathon arcs. For the second construction, I pose and discuss a conjecture that relates the minimum distance of this code to the existence of certain $(q+t,t)$-arcs of type $(0,2,t)$, to which also the next chapter is devoted. I provide a partial proof of this conjecture for the case $k=4$. These results were presented at WCC 2011 as ongoing research, but I did not deem them strong enough for journal publication.

Chapter 3 is about $(q+t,t)$-arcs of type $(0,2,t)$, or shortly $\mathrm{KM}_{q,t}$-arcs, in Desarguesian projective planes of even order. In Section 3.1, I discuss the motivations for studying these arcs, as well as the state of the art. In Section 3.2, I discuss an elegant basis for the projective plane code, and I pose a motivated conjecture, supported by computer simulations, on how linear dependency between incidence vectors of lines translates to the existence of certain $\mathrm{KM}_{q,t}$-arcs. In Section 3.3, I prove this conjecture for the case $k=q/2$, and thereby indirectly provide an alternative proof for the classification of the projective triads [122, 131]. In Section 3.4 I present my main result on this topic: despite being unable to prove the conjectures posed in Section 3.2, I used them as inspiration to invent a new construction technique for $\mathrm{KM}_{q,q/4}$-arcs, and subsequently proved this new construction by different means. This resulted in both a new infinite class of $\mathrm{KM}_{q,t}$-arcs and a great support for the plausibility of the conjectures posed in Section 3.2. The results in this chapter have been published in Finite Fields Appl. [141].

Chapter 4 discusses optimal blocking multisets, i.e. blocking multisets which m-fold block the hyperplanes of $\mathrm{PG}(t,q)$ with as few points as possible. Next to this intrinsic geometric motivation, Section 4.1 discusses another known motivation from coding theory, related to

highly divisible Griesmer codes. Section 4.2 discusses a new motivation: when writing the multisets as linear combinations of hyperplanes, it turns out that these are exactly the ones that have all coefficients nonnegative. Armed with this new observation, Section 4.3 improves on almost all known results on this class of multisets. Finally, Section 4.4 demonstrates yet another link with coding theory, namely a link with projective space codes over the ring of integers modulo a prime power. The results in this chapter are joint work with I. Landjev and were published in J. Comb. Theory Ser. A [87].

Chapter 5 deals with small line sets which have few odd-points, i.e. few points that lie on an odd number of these lines. In particular, we are interested in the minimum value of $|\mathcal{B}| + |\operatorname{odd}(\mathcal{B})|$, where $\mathcal{B}$ is the set of lines and $\operatorname{odd}(\mathcal{B})$ is the set of points that are on an odd number of these lines. In Section 5.1, several motivations for studying such sets are discussed. In Section 5.2, the affine case is studied and we classify all sets with $|\mathcal{B}| + |\operatorname{odd}(\mathcal{B})| \leq 2q$ as one of eight given constructions, or one remaining open case where only a characterization is obtained (which is conjectured not to exist). In Section 5.3, the projective case is studied and here we obtain a full classification of all sets with $|\mathcal{B}| + |\operatorname{odd}(\mathcal{B})| \leq 2q + 2$ as one of six given constructions. The results in this chapter were accepted for publication in Des. Codes Cryptogr. [142].

Chapter 6 is devoted to geometries over finite chain rings. Section 6.1 introduces the relevant preliminaries for finite chain rings, codes over these finite chain rings and geometries over these finite chain rings.

In Section 6.2, we present a standard representation for submodules of $\mathfrak{R}^n$, where $\mathfrak{R}$ is a finite chain ring. Moreover, I provide efficient methods to compute the dual module and the span/intersection of such modules. This section is joint work with I. Landjev.

In Section 6.3, we generalize Kantor's theorem to arbitrary free modules by arbitrary modules. While it may be tempting to conjecture that this holds in general for arbitrary modules by arbitrary modules, this turns out to be false; we provide a counterexample for the more general statement. This section is joint work with I. Landjev; the results in this section have been accepted for publication in Des. Codes Cryptogr. [88].

Chapter 7 discusses various other results I have obtained, and published or submitted for publication. In Section 7.1, I present a new inequality theorem that simultaneously generalizes the inequality between the arithmetic and geometric mean of nonnegative numbers, as well as Turkevich's inequality [123]. This result is joint work with G. Kós and H. Lee, and was published in the general mathematics journal Proc. Amer. Math. Soc. [81].

Section 7.2 discusses the possible weights for which large weight code words can appear in the classical projective space code. In the case q even, this reduces to the study of small blocking sets. For q odd, When the base prime of the field is large enough, we show that there are code words of Hamming weight equal to the length of the code, but for smaller primes this turns out not to be the case. In particular, for $p = 3$ we link the problem to the existence of 2 mod 3 sets with respect to the k-spaces, the existence of which is surprisingly still an open problem for most cases. These results are joint work with J. Limbupasiriporn and L. Storme, and were published in Linear Algebra and its Applications [94].

Section 7.3 studies the existence of blocking sets on the Hermitian unital, whose complement is also a blocking set. Here, we show that for $q \geq 4$, such blocking sets exist on the Hermitian unital of $\mathrm{PG}(2, q^2)$. Moreover, we study the possible sizes and related results. These results are joint work with A. Blokhuis, A.E. Brouwer, D. Jungnickel, V. Krčadinac, S. Rottey, L. Storme and T. Szőnyi, and are submitted to Finite Fields Appl. [16].

Next to these results, I have worked on quantum coding theory, in a joint effort with Yuichiro Fujiwara et al. We have obtained results on entanglement-assisted quantum LDPC codes, which were published in Phys. Rev. A [43]; on high-rate quantum LDPC codes assisted by reliable qubits, which was submitted to IEEE Trans. Inform. Theory [44]; and on quantum synchronizable codes from finite geometries, which was submitted to IEEE Trans. Inform. Theory [45]. However, introducing all the machinery and preliminaries for explaining these results on quantum theory in a mathematically correct way, would require too much time to be feasible, therefore I will just refer the reader to the cited papers for further information on my work on these topics.

Additionally, I have also worked on the applications of my theoretical research, in order to close the gap between the mathematical and engineering research on these topics. I have written a (nearly finished) manuscript that shows how to utilize the property that all small weight code words are a linear combination of a known set of code words; and I have made a modification in the LDPC decoding algorithm to reduce the memory requirements by an order of magnitude at the cost of a moderate reduction in performance. I have been recommended not to add these to my PhD thesis to avoid this thesis being cited later instead of the papers that may potentially follow from the results, but those interested in these results can feel free to contact me.

Finally, I have written large computational libraries to efficiently work with various objects in coding theory and finite geometry, including vector spaces, linear codes, Grassmannians of projective subspaces, operations on subspaces, groups acting on collections of subspaces, matrix algebra over finite fields, finite chain rings, etc. Special attention has been given to the LDPC decoding algorithm, which I have implemented in OpenCL to perform decoding highly efficient on GPU. In 2012 our department received a 9000 euro FCWO grant to build a GPU computing machine, which I built from scratch myself. In Appendix A, I will report on the building of this machine, as well as on the OpenCL implementation of the LDPC decoding algorithm.

Acknowledgement

I would like to express a word of thanks to all people and instances that have been of great value to me during my time as a PhD fellow at the FWO. First and foremost, the FWO, for providing me with the necessary funds and supplementary grants.

Secondly, I want to thank my supervisor, for guiding and directing me for many years, for being a neverending source of new problems to investigate, and for being a great networking hub to get involved with other researchers in our field.

Thirdly, I want to thank my colleagues whom I had fruitful discussions and joint work with, and in particular those who have shown me their hospitality while I was visiting them: Ivan Landjev, Qing Xiang, Anton Betten and Yuichiro Fujiwara.

Finally, I would like to thank my family and close friends for being very supportive overall and in particular for their mental and logistical supportiveness over the past years.

–Peter, April 24, 2014

Chapter 1

Preliminaries: Finite Geometry & Coding Theory

1.1 Finite geometry

Definition 1.1.1. An incidence structure $(\mathcal{P}, \mathcal{B}, I)$ consists of a finite set $\mathcal{P}$, the elements of which are called *points*, a finite set $\mathcal{B}$, the elements of which are called *blocks*, and a relation $I \subseteq \mathcal{P} \times \mathcal{B}$, which is called the *incidence relation*. Often, $\mathcal{B}$ is a collection of subsets of $\mathcal{P}$ and the incidence relation is simply $\in$. If additionally, any two distinct points are contained in at most one common block (or equivalently, any two distinct blocks contain at most one common point), the blocks are sometimes called *lines*.

Definition 1.1.2. The *dual* incidence structure to an incidence structure $(\mathcal{P}, \mathcal{B}, I)$ is the incidence structure $(\mathcal{B}, \mathcal{P}, I')$ with $pIb \Leftrightarrow bI'p$ for all $p \in \mathcal{P}$ and all $b \in \mathcal{B}$. For example, the dual of $(\mathcal{P}, \mathcal{B}, \in)$ is $(\mathcal{B}, \mathcal{P}, \ni)$.

Definition 1.1.3. The *incidence matrix* of an incidence structure $(\mathcal{P}, \mathcal{B}, I)$, where we let $\mathcal{P} = \{p_1, \ldots, p_m\}$ and $\mathcal{B} = \{b_1, \ldots, b_n\}$, is the $m \times n$ matrix, in which the rows are labeled by the points and the columns are labeled by the blocks, such that $H_{ij} = 1$ if $p_i I b_j$, and $H_{ij} = 0$ otherwise. The incidence matrix of the dual incidence structure is then simply the transposed of this matrix. Sometimes this second matrix is also referred to as the incidence matrix of the structure, so it needs to be clear from the context which matrix is referred to.

Notation 1.1.4. A finite field $\mathbb{F}_q$ of order q has $q = p^h$ elements, where p is a prime number and h is a positive integer. We denote by $\operatorname{char} F$ the characteristic of the field F, so $\operatorname{char} \mathbb{F}_q = p$.

Notation 1.1.5. Let $V = \mathbb{F}_q^n$.

- By $\mathrm{PG}(n-1, q)$ we denote the geometry with as its $(i-1)$-spaces the i-dimensional subspaces of V, and inclusion as its incidence relation.
- By $\mathrm{AG}(n, q)$ we denote the geometry with as its i-spaces the i-dimensional subspaces of V and their cosets, and inclusion as its incidence relation.

- By $\mathrm{EG}(n,q)$ we denote the geometry with as its i-spaces the cosets of i-dimensional subspaces of V (but not the subspaces themselves), and inclusion as its incidence relation.

Notation 1.1.6. For $n \geq 1$, $\mathrm{AG}(n,q)$ and $\mathrm{PG}(n,q)$ are called the affine and projective line (when $n = 1$), plane (when $n = 2$), space (when $n \geq 3$). Sometimes the name space is used for general n as well, to simplify the wording.

Definition 1.1.7. A *non-singular Hermitian variety* in $\mathrm{PG}(m,q^2)$ is the set of absolute points of a Hermitian polarity, which is defined by a Hermitian matrix and the non-trivial involution $x \mapsto x^q$ of $\mathrm{Aut}(\mathbb{F}_{q^2})$. All non-singular Hermitian varieties are projectively equivalent to the one given by the equation

$$X_0^{q+1} + X_1^{q+1} + \cdots + X_m^{q+1} = 0\,.$$

The projective index of a non-singular Hermitian variety in $\mathrm{PG}(m,q^2)$ equals $\left\lfloor \frac{m-1}{2} \right\rfloor$. The maximal subspaces of a Hermitian variety are called *generators*. From now on, we will denote the standard non-singular Hermitian variety in $\mathrm{PG}(m,q^2)$ by $\mathcal{H}(m,q^2)$. A *singular Hermitian variety* in $\mathrm{PG}(m,q^2)$ is a cone with an i-dimensional subspace as vertex and a non-singular Hermitian variety in an $(m-i-1)$-dimensional subspace, disjoint from the vertex, as base, $-1 \leq i \leq m$. All singular Hermitian varieties in $\mathrm{PG}(m,q^2)$ with an i-dimensional vertex are projectively equivalent to the one given by the equation

$$X_0^{q+1} + X_1^{q+1} + \cdots + X_{m-i-1}^{q+1} = 0\,.$$

Note that a non-singular Hermitian variety is a singular Hermitian variety with vertex dimension equal to -1.

Definition 1.1.8. A *k-arc* (or briefly an *arc*) in $\mathrm{PG}(2,q)$ is a set of k points such that no three lie on the same line. Dually, a *dual k-arc* (or briefly a *dual arc*) is a set of k lines such no three are concurrent.

It is easy to show that in $\mathrm{PG}(2,q)$, q odd, every arc has at most $q+1$ points (and dually, a dual arc in $\mathrm{PG}(2,q)$, q odd, has at most $q+1$ lines). Moreover, the following characterization result on $(q+1)$-arcs is known.

Theorem 1.1.9 (Segre [119])**.** *Every $(q+1)$-arc in $\mathrm{PG}(2,q)$, q odd, is a conic.*

By duality, every dual $(q+1)$-arc in $\mathrm{PG}(2,q)$, q odd, is a dual conic. Easy counting arguments show that every line in a dual $(q+1)$-arc in $\mathrm{PG}(2,q)$, q odd, intersects each of the q other lines in a different point, and hence there is exactly one 1-point on every line of the dual $(q+1)$-arc and the other q^2 points in $\mathrm{PG}(2,q)$, q odd, are 0-points or 2-points. The 1-points form a conic. Dually, the set of tangents to a conic in $\mathrm{PG}(2,q)$, q odd, also forms a dual conic.

Definition 1.1.10. A hyperoval is a set of $q+2$ points in $\mathrm{PG}(2,q)$ such that no three of them are collinear.

Hyperovals exist if and only if q is even, and they are the largest arcs that exist when q is even. More background on (substructures of) projective and affine spaces can be found in [59].

Notation 1.1.11. The symbol $\begin{bmatrix} n \\ k \end{bmatrix}_q$ with $k \leq n$ integers and q a prime power, denotes the classical Gaussian coefficient:

$$\begin{bmatrix} n \\ k \end{bmatrix}_q = \frac{(q^n-1)\dots(q^{n-k+1}-1)}{(q^k-1)\dots(q-1)}.$$

It is well known that $\begin{bmatrix} n+1 \\ k+1 \end{bmatrix}_q$ is the number of subspaces in $\mathrm{PG}(n,q)$.

Definition 1.1.12. A $(q+t,t)$-arc of type $(0,2,t)$ in $\mathrm{PG}(2,q)$ is a set S of $q+t$ points in $\mathrm{PG}(2,q)$ for which every projective line meets S in either 0, 2 or t points.

Definition 1.1.12 was introduced in [80] and it is proven that $(q+t,t)$-arcs of type $(0,2,t)$ with $1 < t < q$ can only exist if q is even. Moreover, they prove that t needs to be a divisor of q, i.e. $t = 2^r$ with $r \leq h$. They also provide a construction of such arcs if $h-r$ divides h. From now on, we will assume that q is even (and hence is a power of 2) and t divides q.

Remark 1.1.13. A hyperoval in $\mathrm{PG}(2,q)$, $q = 2^h$ with $h \geq 1$, can be seen as a $(q+2,2)$-arc of type $(0,2,2)$. One can see $(q+t,t)$-arcs of type $(0,2,t)$ as a generalization of hyperovals. The symmetric difference of two lines of $\mathrm{PG}(2,q)$ can be seen as a $(2q,q)$-arc of type $(0,2,q)$.

Definition 1.1.14. A $(q+t,t)$-arc of type $(0,2,t)$ in $\mathrm{PG}(2,q)$, $q = 2^h$, is said to have a t-nucleus if all the t-secants are concurrent.

In [80] it is proven that all $(q+t,t)$-arcs of type $(0,2,t)$ have a t-nucleus if $h-r+1 \neq \gcd(h,r-1)$, conjecturing that it holds for all r,h. That conjecture was proven in [47].

Theorem 1.1.15 ([47])**.** *Every $(q+t,t)$-arc of type $(0,2,t)$ in $\mathrm{PG}(2,q)$, $q = 2^h$, has a t-nucleus.*

Conjecture 1.1.16 ([80])**.** *If 4 divides t and t divides q, then there exists a $(q+t,t)$-arc of type $(0,2,t)$.*

Conjecture 1.1.16 is open for more than 20 years now. In [80] it is proven that a $(2^h+2^r, 2^r)$-arc exists when $h-r$ is a proper divisor of h. Later, in [47] the authors prove another infinite class of such arcs for which $h-r$ is not a proper divisor of h; more precisely they construct

- a $(2^{hr}+2^{h(r-1)}, 2^{h(r-1)})$-arc of type $(0,2,2^{h(r-1)})$ in $\mathrm{PG}(2,2^{hr})$;
- a $(2^{hr}+2^{h(r-1)+1}, 2^{h(r-1)+1})$-arc of type $(0,2,2^{h(r-1)+1})$ in $\mathrm{PG}(2,2^{hr})$;
- a $(2^{hr}+2^{h(r-1)+s}, 2^{h(r-1)+s})$-arc of type $(0,2,2^{h(r-1)+s})$ in $\mathrm{PG}(2,2^{hr})$ if there exists a $(2^h+2^s, 2^s)$-arc of type $(0,2,2^s)$ in $\mathrm{PG}(2,2^h)$.

Some $(40,8)$-arcs of type $(0,2,8)$ in $\mathrm{PG}(2,32)$ were found in [93] via computer searches. Shortly after, a $(36,4)$-arc of type $(0,2,4)$ in $\mathrm{PG}(2,32)$ was discovered in [76], also via computer searches. Hence, in $\mathrm{PG}(2,32)$, there are $(32+t,t)$-arcs of type $(0,2,t)$ for all divisors t of 32. The next open cases are $(68,4)$-arcs of type $(0,2,4)$ in $\mathrm{PG}(2,64)$, and $(128+t,t)$-arcs of type $(0,2,t)$ for $t = 4,8,16,32$. In Section 3.4 I will construct a new infinite class of $(q+q/4, q/4)$-arcs of type $(0,2,q/4)$, for all $q = 2^h$, $h \geq 3$.

Definition 1.1.17. A dual $(q+t,t)$-arc of type $(0,2,t)$ in $\mathrm{PG}(2,q)$ is a set S of $q+t$ lines in $\mathrm{PG}(2,q)$ for which every projective point lies on either 0, 2 or t lines of S.

Note that the (binary) sum of the incidence vectors of the lines in a dual $(q+t,t)$-arc is equal to the zero word, since t is necessarily even.

It is clear that, since $\mathrm{PG}(2,q)$ is self-dual, arcs are equivalent to dual arcs, and all properties for arcs also hold for dual arcs (and vice versa). In a similar fashion one can use concepts such as *dual t-nucleus*, which is just the dual of the t-nucleus.

1.2 Coding theory

Definition 1.2.1. A *linear $[n,k,d]$-code* C over a finite field $\mathbb{F}_q$ is a k-dimensional subspace of $V(n,q)$, such that every two distinct vectors in C differ in at least d positions.

There are two common ways to define such codes:

- as the null space of an $m \times n$ matrix H of rank $n-k$ over $\mathbb{F}_q$ (which is called the *parity check matrix* H of C);
- as the row space of an $m' \times n$ matrix G of rank k over $\mathbb{F}_q$ (which is called the *generator matrix* G of C).

Often, one deletes dependent rows from these matrices to make them $(n-k) \times n$, resp. $k \times n$.

Linear codes (often with $q=2$) form an important tool in information theory, communication theory, data transmission, combinatorics, finite geometry and many other areas. In data transmission, these codes are used for error correction:

- A message vector $m \in V(k,q)$ is transmitted as $v = m \cdot G \in V(n,q)$;
- during transmission, several errors may occur in the transmitted code word $(v \to v')$;
- upon arrival, if less than d errors occurred, this can always be detected as $Hv' \neq 0$;
- if less than $d/2$ errors occurred, they can even be corrected algorithmically (this is called *decoding*).

Definition 1.2.2. The parameters n,k,d are called the *length*, *dimension* and *minimum distance* of C.

Definition 1.2.3. Given an incidence structure $(\mathcal{P},\mathcal{B},I)$, we define $C_{\mathbb{F}}$ to be the code over $\mathbb{F}$ having its parity check matrix H equal to the incidence matrix of this incidence structure $(\mathcal{P},\mathcal{B},I)$. If the field $\mathbb{F}$ is clear from the context, we will simply write C.

This gives the code a nice interpretation: a code word c of the code corresponds to a map $\varphi : \mathcal{B} \to \mathbb{F}$ such that, for each point $r \in \mathcal{P}$, we have $\sum_{L \ni r} \varphi(L) = 0$ over $\mathbb{F}$. We call $\varphi(L)$ the *coefficient* of the line L in the code word c, and we denote this by c_L. Similarly, when considering the code derived from the dual structure, a code word is a map $\varphi : \mathcal{P} \to \mathbb{F}$ such that, for each block $L \in \mathcal{B}$, we have $\sum_{r \in L} \varphi(r) = 0$ over $\mathbb{F}$.

Remark 1.2.4. A *code word* c of $C_k(\mathcal{S})^\perp$ is an element of the $\mathbb{F}_p$-null space of A, which is equivalent to a mapping from $\mathcal{P}$ to $\mathbb{F}_p$ with the additional property that $\sum_{p \in \pi} c_p = 0$, for all $\pi \in \mathcal{B}$. Hence, code words can be studied as multisets of points such that each k-space on $\mathcal{S}$ contains $0 \pmod p$ of the points in the multiset.

Definition 1.2.5. The *support* of a word c is the set of positions with nonzero entry, i.e. $\operatorname{supp}(c) = \{p : c_p \neq 0\}$, where P is the set of positions. For finite geometry codes, we usually identify this set with the corresponding set of points (or subspaces when using the dual incidence structure).

Definition 1.2.6. The *Hamming weight* $\operatorname{wt}(c)$ of a word c is the number of nonzero symbols in it, i.e. it is $|\operatorname{supp}(c)|$. The *minimum distance* d of $C_k(\mathcal{S})^\perp$ is $\min_{c \in C_k(\mathcal{S})^\perp \setminus \{0\}} \operatorname{wt}(c)$.

Definition 1.2.7. Let C be an LDPC code defined by the parity check matrix H, with as its rows $H_1, \ldots, H_m$. Then we define the following distance functions:

- A *code word* of C is simply a word $c \in C$. The *Hamming distance* of C (denoted by $d(C)$) is the smallest Hamming weight among all nonzero code words.
- A *stopping set* of C is a set $S \subseteq \mathbb{F}^n$ with the property

$$\forall s \in S, \forall r \in \{1, \ldots, m\} : |\operatorname{supp}(s) \cap \operatorname{supp}(H_r)| \neq 1.$$

  The *stopping distance* of C (denoted by $sd(C)$) is the size of the smallest nonempty stopping set.

Note that, while $d(C)$ is an invariant of the code, the stopping distance depends on the parity check matrix used to define the code.

Chapter 2

LDPC codes derived from Galois geometries

2.1 Motivation and preliminaries

Originally introduced by Gallager [48], low density parity check (LDPC) codes are frequently used these days due to their excellent empirical performance under belief-propagation (a.k.a. sum-product) decoding. In some cases, their performance is even close to the Shannon limit [98]. In general, a binary LDPC code C is a linear block code defined by a sparse parity check matrix H, this is a matrix that contains a lot more 0s than 1s.

To exploit structural properties, one usually wants explicit constructions rather than random matrices. Early on, constructions have been proposed based on permutation matrices [41],[135], Ramanujan graphs [100],[116], expander graphs [125], q-regular bipartite graphs [77] or other incidence structures in discrete mathematics.

Lately, many constructions coming from Galois geometries have been investigated, because of their low complexity decoding features [82, 108], such as projective and affine spaces [33, 34], generalized quadrangles [78, 95], linear representations [113, 139, 140] and partial and semipartial geometries [74, 91]. Geometrical LDPC codes have been used in several high-end modern data transmission systems [33, 34, 150] and in entanglement-assisted quantum decoding [43].

Examples of such codes can be found in [70, 71, 72, 73, 97, 147]. Later, simulation results of Liu and Pados [95] showed that several generalized polygon LDPC codes have powerful bit-error-rate performance when decoding is carried out via low-complexity variants of belief propagation. It would be interesting to perform the same simulations for the incidence geometries studied in this thesis, since all handled structures have a girth of at least 6 in their associated Tanner graph. If $\mathcal{K}$ is an arc, then the Tanner graph even has girth at least 8.

The performance of LDPC codes under iterative decoding is determined by several parameters and properties of the code and of the particular parity check matrix used for decoding. The

main characteristics that are of interest are the girth of the associated Tanner graph, the minimum distance, the stopping distance, the pseudodistance and the trapping distance. We will in this section focus on the first three.

Definition 2.1.1. The *Tanner graph* of an incidence structure is a bipartite graph with as its bipartition classes the rows and columns of the incidence matrix of an incidence structure, where two nodes are connected if and only if the corresponding matrix entry is nonzero (i.e. equals 1).

Definition 2.1.2. Let C be a linear code defined by the sparse parity check matrix H. Let G be the bipartite graph associated with H, then G is called the *Tanner graph* of this code. The *girth of the Tanner graph* is the graph theoretic girth of G, i.e. the size of the smallest cycle in G.

Determining the girth of the Tanner graph of a finite geometry is almost always trivial:

- since the Tanner graph is bipartite and simple, the girth is always even and at least 4;
- the fact that the Tanner graph has girth at least 6 can be stated equivalently as the fact that two different blocks have at most one common point or, again equivalently, as the fact that two different points are contained in at most one common block;
- if the girth is at least 6, it is exactly $2t$, where $t \geq 3$ is the smallest integer for which a t-gon exists in the geometry.

Note that a linear code has several parity check matrices defining it, so several Tanner graphs can be associated to a single linear code. Therefore, the notion of a Tanner graph will only be used in circumstances where the parity check matrix is explicitly given.

The following general lower bound on the minimum distance is known.

Theorem 2.1.3 ([7])**.** *Let C be an LDPC code defined by a $\{0,1\}$-parity check matrix H over the finite field $\mathbb{F}_p$. If v is the minimum number of ones in a column of the parity check matrix H, and r is the maximum number of common ones in any two different columns, then*

$$d(C) \geq \frac{2}{p}\left((p-1)\frac{v}{r}+1\right) .$$

For most non-binary codes, this is the strongest bound available.

The motivation for studying the minimum distance and stopping distance of finite geometry codes is three-fold.

- First of all, these codes have been shown by several authors to have excellent performance for their length. Even though the performance of LDPC codes can be brought arbitrarily close to the Shannon limit [98], this is only achieved at very large code lengths, in which case even iterative decoding becomes unfeasible. For shorter code lengths, finite geometry codes are among the top competitors in terms of iterative decoding performance.

- Secondly, theoretical and structural arguments are all we have. The length, dimension and Tanner girth of an LDPC code can be computed easily (in polynomial time). The minimum distance is however not at all easy to determine. In fact, it has been shown [146] that the determination of the minimum distance is in general an NP-complete problem, and even in the special case of LDPC codes, the problem remains unfeasible [66]. Later on, determining the stopping distance of an LDPC code was also proven to be NP-hard [83]. Hence, we can only find these parameters from a theoretical study.

- Thirdly, studying these parameters may also yield interesting theoretical results on the code and on the underlying geometry. For example, in [140], I computed the minimum distance for a code derived from $T_2^*(\mathcal{K})$, finding on the way a general formula for the dimension of the code and an exact basis for the code. In [141], the study of the PG$(2,q)$-code resulted in a new class of $(q+t,t)$-arcs of type $(0,2,t)$, the existence of which was an open problem in finite geometry.

- Finally, when transmission is done over an erasure channel, maximum likelihood (ML) decoding can correct any error which erases less than $d(C)$ positions, while it is proven[1] in [25, Lemma 1.1] that iterative sum-product decoding can correct any error which erases less than $sd(C)$ positions. For this reason, we will focus our study on these two distance functions.

In [74], the authors show that LDPC codes derived from partial and semipartial geometries have excellent empirical performance under LDPC decoding methods based on belief propagation, and they derive bounds on the dimension and the minimum distance of these codes. In this paper, we will improve the known bounds on the minimum distance and we provide several new theoretical and computer results about it. We also attempt to determine the stopping distance, or lower bounds on this stopping distance, whenever possible.

2.2 LDPC codes from projective and affine geometries

The most commonly studied finite geometry is by no doubt the projective plane/space, and most of the applied results in engineering have been based on codes from projective spaces PG(n,q) and their derived spaces AG(n,q) and EG(n,q) [33, 34, 43, 45, 82, 108, 150]. More specifically, one takes the incidence matrix of t-spaces by k-spaces in PG(n,q) (or AG(n,q) or EG(n,q)) for some values of n,t,k, and uses it as a parity check matrix. The most studied case is $n=2,k=1,t=0$, i.e. points and lines in the plane. Since the entries of the incidence matrices are always 0 or 1 (which are in the prime subfield of any finite field), one usually only considers codes over $\mathbb{F}_{p'}$ with p' some prime (usually $p'=2$).

The dimension of the codes created in this way, follows straightforwardly from the p'-rank of the incidence matrices, where often one takes $p'=p$ (with $q=p^h$). A general formula for the p-rank of the incidence matrices of points by t-spaces in PG(n,q), AG(n,q) and EG(n,q) was found by Hamada [54], see below. A general formula for the rank of the incidence matrix of s-spaces by t-spaces is not known.

[1] Note that the stated lemma assumes that the channel is binary, but its proof works for arbitrary channels.

Theorem 2.2.1 ([54]). *The p-rank of the incidence matrix of points by t-spaces in* $\mathrm{PG}(m, p^h)$ *is*

$$R_{m,t,p,h} := \sum_{(s_0,\ldots,s_h)\in S} \prod_{j=0}^{h-1} \sum_{i=0}^{L(s_{j+1},s_j)} (-1)^i \binom{n+1}{i} \binom{m + s_{j+1}p - s_j - ip}{m},$$

where

$$S = \{(s_0, s_1, \ldots, s_h) | s_0 = s_h, \forall j : t+1 \leq s_j \leq m+1, \forall j : 0 \leq s_{j+1}p - s_j \leq (m+1)(p-1)\}$$

and $L(s_{j+1}, s_j) = \left\lfloor \frac{s_{j+1}p - s_j}{p} \right\rfloor$.

The p-rank of the incidence matrix of points by t-spaces in $\mathrm{AG}(m, p^h)$ *is* $R_{m,t,p,h} - R_{m-1,t,p,h}$.

The p-rank of the incidence matrix of points by t-spaces in $\mathrm{EG}(m, p^h)$ *is* $R_{m,t,p,h} - R_{m-1,t,p,h} - 1$.

Over fields with a characteristic different from p, these matrices have full rank or full rank minus one, depending on whether the number of t-spaces through a point equals 0 modulo the characteristic of this field or not.

In 2006, Chandler, Sin and Xiang showed a strong generalization of these results, by decomposing these matrices to their Smith Normal Form over $\mathbb{Z}$ [24]. The rank over fields of arbitrary characteristic follows immediately from these results.

Definition 2.2.2. A *cyclic* $[n, k, d]$-code C is a linear $[n, k, d]$-code in which every cyclic shift of every code word $c = (c_0, \ldots, c_{n-1}) \in C$ is also a code word, that is, for any $c = (c_0, \ldots, c_{n-1}) \in C$, we have $(c_1, \ldots, c_{n-1}, c_0) \in C$. It is known that, by regarding each code word as the coefficient vector of a polynomial in $\mathbb{F}_2[x]$, a cyclic code of length n can be seen as a principal ideal in the ring $\mathbb{F}_2[x]/(x^n - 1)$ generated by the unique monic nonzero polynomial $g(x)$ of minimum degree in the code which divides $x^n - 1$. When a cyclic code is of length n and dimension k, the set of code words can be written as $C = \{i(x)g(x) \mid \deg(i(x)) < k\}$, where the degree $\deg(g(x))$ of the generator polynomial is $n - k$.

Clearly, since the Singer cycle acts transitively on the points of $\mathrm{PG}(m, q)$ and $\mathrm{EG}(m, q)$, the codes derived from these geometries are cyclic. Moreover, one can determine the exact generator polynomial of these codes, as well as derive some other useful properties. First, we will introduce some simplifying notations and lemmata.

The *incidence vector* χ_π of π in $\mathrm{PG}(m, 2^h)$ is the binary $\frac{2^{h(m+1)}-1}{2^h-1}$-dimensional vector such that the coordinates are indexed by the points and such that each entry is 1 if π contains the corresponding point and 0 otherwise. Similarly, one can define incidence vectors of subspaces of $\mathrm{EG}(m, 2^h)$.

Definition 2.2.3. The *complement* $\chi_{\overline{\pi}}$ of an incidence vector χ_π is the vector $\chi_{\overline{\pi}} = \chi_\pi + \mathbf{1}$, where $\mathbf{1}$ is the all-one vector. In other words, $\chi_{\overline{\pi}}$ is obtained by flipping 0's and 1's in χ_π.

Definition 2.2.4. Denote in this section by $\mathcal{B}$ the set of all t-spaces in $\mathrm{PG}(m, 2^h)$. We let $\mathcal{P}_{m,t,2^h} = \langle \chi_\pi \mid \pi \in \mathcal{B} \rangle$, the code spanned by all the incidence vectors of t-spaces in $\mathcal{B}$, and $\mathcal{C}_{m,t,2^h} = \langle \chi_{\overline{\pi}} \mid \pi \in \mathcal{B} \rangle^\perp$ be the dual of the vector space spanned by the set of complements of the incidence vectors of t-dimensional subspaces of $\mathrm{PG}(m, 2^h)$.

The fact that $\mathcal{C}_{m,t,2^h} = \langle \chi_{\overline{\pi}} \mid \pi \in \mathcal{B} \rangle^{\perp}$ is a cyclic code follows directly from the fact that $\mathcal{P}_{m,t,2^h} = \langle \chi_\pi \mid \pi \in \mathcal{B} \rangle$ is cyclic as a linear code.

Proposition 2.2.5. $\mathcal{P}_{m,t,2^h} = \langle \mathcal{C}^{\perp}_{m,t,2^h}, \mathbf{1} \rangle$, where $\mathbf{1} \notin \mathcal{C}^{\perp}_{m,t,2^h}$.

Proof. Because all generators $\chi_{\overline{\pi}}$ of $\mathcal{C}^{\perp}_{m,t,2^h}$ are of even weight, all code words of $\mathcal{C}^{\perp}_{m,t,2^h}$ are also of even weight. Since the length $\frac{2^{h(m+1)}-1}{2^h-1}$ of this cyclic code is odd, the all-one vector is not a code word. Recall that $\mathcal{P}_{m,t,2^h}$ is the vector space spanned by the incidence vectors of t-dimensional subspaces in $\mathrm{PG}(m, 2^h)$. Because the number of t-dimensional subspaces that contain a given point in $\mathrm{PG}(m, 2^h)$ is always odd, we have

$$\sum_{\pi \in \mathcal{B}} \chi_\pi = \mathbf{1},$$

which implies that $\mathbf{1} \in \mathcal{P}_{m,t,2^h}$. Because $\chi_{\overline{\pi}} = \chi_\pi + \mathbf{1}$, it follows that $\mathcal{C}^{\perp}_{m,t,2^h} \subset \mathcal{P}_{m,t,2^h}$. Thus, we have $\langle \mathcal{C}^{\perp}_{m,t,2^h}, \mathbf{1} \rangle \subseteq \mathcal{P}_{m,t,2^h}$. Now the fact that $\chi_{\overline{\pi}} = \chi_\pi + \mathbf{1}$ is equivalent to the relation that $\chi_\pi = \chi_{\overline{\pi}} + \mathbf{1}$, which implies that $\mathcal{P}_{m,t,2^h} \subseteq \langle \mathcal{C}^{\perp}_{m,t,2^h}, \mathbf{1} \rangle$. Thus, $\mathcal{P}_{m,t,2^h} = \langle \mathcal{C}^{\perp}_{m,t,2^h}, \mathbf{1} \rangle$ as desired. □

Proposition 2.2.6. $\mathcal{C}_{m,t,2^h} = \langle \mathcal{P}^{\perp}_{m,t,2^h}, \mathbf{1} \rangle$, where $\mathbf{1} \notin \mathcal{P}^{\perp}_{m,t,2^h}$.

Proof. Because the generators of $\mathcal{P}_{m,t,2^h}$ are all of odd weight, the inner product between $\mathbf{1}$ and any of the generators is nonzero, which implies that $\mathbf{1} \notin \mathcal{P}^{\perp}_{m,t,2^h}$. By the same token, because the generators of $\mathcal{C}^{\perp}_{m,t,2^h}$ are of even weight, we have $\mathbf{1} \in \mathcal{C}_{m,t,2^h}$. By Proposition 2.2.5, $\mathcal{C}^{\perp}_{m,t,2^h} \subset \mathcal{P}_{m,t,2^h}$, which implies that $\mathcal{P}^{\perp}_{m,t,2^h} \subset \mathcal{C}_{m,t,2^h}$. Again by Proposition 2.2.5, the dimensions of $\mathcal{P}_{m,t,2^h}$ and $\mathcal{C}^{\perp}_{m,t,2^h}$ satisfy the equation that $\dim \mathcal{P}_{m,t,2^h} = \dim \mathcal{C}^{\perp}_{m,t,2^h} + 1$. Hence, $\mathcal{C}_{m,t,2^h} = \langle \mathcal{P}^{\perp}_{m,t,2^h}, \mathbf{1} \rangle$ as desired. □

Definition 2.2.7. The *weight* $\mathrm{w}_{2^h}(a)$ of the 2^h-ary expansion of a positive integer a, that is,

$$\mathrm{w}_{2^h}(a) = \sum_i a_i,$$

where addition is performed over $\mathbb{Z}$ and

$$a = \sum_{i \in \mathbb{N} \cup \{0\}} a_i 2^{hi}$$

with $0 \le a_i \le 2^h - 1$.

The following theorem gives the explicit form of the generator polynomial of $\mathcal{C}_{m,t,2^h}$.

Theorem 2.2.8. *Let α be a primitive element in $\mathbb{F}_{2^{h(m+1)}}$ and $\beta = \alpha^{2^h-1}$. The generator polynomial $g(x)$ of $\mathcal{C}_{m,t,2^h}$ is*

$$g(x) = \prod_{j \in I_{m,t,h}} (x - \beta^j),$$

where

$$I_{m,t,h} = \Bigg\{ a \in \mathbb{N} \mid a \leq \frac{2^{h(m+1)}-1}{2^h-1}, \max_{0\leq i\leq h} \mathrm{w}_{2^h}(a(2^h-1)2^i) \leq (m-t)(2^h-1) \Bigg\}.$$

Proof. It is known that $\mathcal{P}_{m,t,2^h}$ is the subfield subcode in $\mathbb{F}_2$ of a punctured generalized Reed-Muller code [6, Chapter 16]. The generator polynomial $h(x)$ of its dual $\mathcal{P}^{\perp}_{m,t,2^h}$ is

$$h(x) = (x-1) \prod_{j\in I_{m,t,h}} (x-\beta^j)$$

(see [13, Theorem 13.9.2] and also [2]). It suffices to show that $h(x) = (x-1)g(x)$. By Lemma 2.2.6, $\mathcal{P}^{\perp}_{m,t,2^h} \subset \mathcal{C}_{m,t,2^h}$ and $\dim \mathcal{C}_{m,t,2^h} = \dim \mathcal{P}^{\perp}_{m,t,2^h} + 1$. Thus, the generator polynomial $g(x)$ of $\mathcal{C}_{m,t,2^h}$ is a divisor of $h(x)$, where the quotient is a polynomial of degree 1 over $\mathbb{F}_2$. Since $h(0) = 1$, the polynomial x is not a factor of $h(x)$. Hence, we have $h(x) = (x-1)g(x)$ as desired. □

Another important parameter for cyclic codes is their *order* (also known as the *period* or *exponent*). This has impact on various aspects of the decoding strength, such as the ability to synchronize misaligned transmissions [45].

Definition 2.2.9. The cardinality $\mathrm{ord}(f(x)) = |\{x^a \pmod{f(x)} \mid a \in \mathbb{N}\}|$ is called the *order* of the polynomial $f(x)$, where $\mathbb{N}$ is the set of positive integers.

Obviously, the order is at most the length of the code, and we will show that for these codes, equality is reached.

Proposition 2.2.10. Let $f(x) = \prod_i f_i(x)$ be a polynomial over $\mathbb{F}_2$, where the polynomials $f_i(x)$ are all nonzero and pairwise relatively prime in $\mathbb{F}_2[x]$. Then

$$\mathrm{ord}(f(x)) = \mathrm{lcm}_i\{\mathrm{ord}(f_i(x))\}.$$

Proposition 2.2.11. Let q be a prime or prime power, and α a nonzero element of the extension field $\mathbb{F}_{q^e}$ of $\mathbb{F}_q$ for a positive integer e. Define $f(x) \in \mathbb{F}_q[x]$ to be the minimal polynomial of α over $\mathbb{F}_q$. The order $\mathrm{ord}(f(x))$ is equal to the order of α in the multiplicative group $\mathbb{F}^*_{q^e}$.

For the proofs of these propositions, we refer the reader to [92, Theorems 3.9 and 3.33].

Theorem 2.2.12. *Let $g(x)$ and $h(x)$ be the generator polynomials of $\mathcal{C}_{m,t,2^h}$ and $\mathcal{C}_{m,t-i,2^h}$ for a positive integer $i \leq t-1$ respectively. Define $f(x)$ to be the quotient of $h(x) = f(x)g(x)$ divided by $g(x)$. Then* $\mathrm{ord}(f(x)) = \frac{2^{h(m+1)}-1}{2^h-1}$.

[2] To avoid confusion in notation, "m" in Theorem 13.9.2 in [13] corresponds to "$m+1$" in this paper while "r" and "s" there are "t" and "h" here respectively. Note also that the current edition of the textbook contains typographical errors in the statement of Theorem 13.9.2, so that "$0 < j \ldots$" and "$0 < \max \ldots$" should read "$0 \leq j \ldots$" and "$0 \leq \max \ldots$" respectively

Proof. Let α be a primitive element in $\mathbb{F}_{2^{h(m+1)}}$ and $\beta = \alpha^{2^h-1}$. By Theorem 2.2.8, we have

$$f(x) = \prod_{j \in I_{m,t-i,h} \setminus I_{m,t,h}} (x - \beta^j).$$

We consider two special factors of $f(x)$. Let $j_0 = \frac{2^{h(m-t)}-1}{2^h-1}$ and $j_1 = \frac{2^{h(m-t+1)}-1}{2^h-1} - 2$. Then

$$\begin{aligned} \max_{0 \le i \le h} \mathrm{w}_{2^h}(j_0(2^h-1)2^i) &= \max_{0 \le i \le h} \mathrm{w}_{2^h}(j_1(2^h-1)2^i) \\ &= (m-t)(2^h-1). \end{aligned}$$

Hence, the minimal polynomials $M_{\beta^{j_0}}(x)$, $M_{\beta^{j_1}}(x)$ of β^{j_0} and β^{j_1} are nonzero factors of $f(x)$. Since $M_{\beta^{j_0}}(x)$ and $M_{\beta^{j_1}}(x)$ are minimal polynomials, the two are relatively prime. So, by Propositions 2.2.10 and 2.2.11 and the fact that j_0 and j_1 are relatively prime, we have

$$\begin{aligned} \mathrm{ord}(f(x)) &\ge \mathrm{lcm}\left(\mathrm{ord}\left(M_{\beta^{j_0}}(x)\right), \mathrm{ord}\left(M_{\beta^{j_1}}(x)\right)\right) \\ &= \mathrm{lcm}\left(\frac{2^{h(m+1)}-1}{\gcd\left(j_0(2^h-1), 2^{h(m+1)}-1\right)}, \right. \\ &\qquad \left. \frac{2^{h(m+1)}-1}{\gcd\left(j_1(2^h-1), 2^{h(m+1)}-1\right)}\right) \\ &= \frac{2^{h(m+1)}-1}{2^h-1}. \end{aligned}$$

Since the order of a factor of the generator polynomial of a cyclic code is at most the length of the code, we have $\mathrm{ord}(f(x)) = \frac{2^{h(m+1)}-1}{2^h-1}$. □

Define $\mathcal{E}_{m,t,2^h} = \langle \chi_\pi \mid \pi \in \mathcal{B} \rangle^\perp$ to be the dual of the vector space spanned by the incidence vectors of t-dimensional subspaces in $\mathrm{EG}(m, 2^h)$. Similar to the case of projective geometry over a finite field, the cyclic group of order $2^{hm}-1$ acts regularly on the points in the case of $\mathrm{EG}(m, 2^h)$, making $\mathcal{E}_{m,t,2^h}$ cyclic. It is also one the oldest efficiently decodable codes and was first discovered in 1960's. Its basic properties in this context can be found in [13, Section 13.8].

Theorem 2.2.13 ([26])**.** *Let α be a primitive element in $\mathbb{F}_{2^{hm}}$. The generator polynomial $g(x)$ of the code of points and t-spaces (with points as positions) is*

$$g(x) = \prod_{j \in I'_{m,t,h}} (x - \alpha^j),$$

where

$$\begin{aligned} I'_{m,t,h} = \Big\{ & a \in \mathbb{N} \mid a \le 2^{hm} - 1, \\ & \max_{0 \le i \le h} \mathrm{w}_{2^h}(a2^i) \le (m-t)(2^h-1) \Big\}. \end{aligned}$$

Theorem 2.2.14. *Let $g(x)$ and $h(x)$ be the generator polynomials of $\mathcal{E}_{m,t,2^h}$ and $\mathcal{E}_{m,t-i,2^h}$ for a positive integer $i \le t-1$ respectively. Define $f(x)$ to be the quotient of $h(x) = f(x)g(x)$ divided by $g(x)$. Then* $\mathrm{ord}(f(x)) = 2^{hm} - 1$.

Proof. By Theorem 2.2.13, we have

$$f(x) = \prod_{j \in I'_{m,t-i,h} \setminus I'_{m,t,h}} (x - \alpha^j).$$

Let $j_0 = 2^{h(m-t)} - 1$ and $j_1 = 2^{h(m-t)} - 2$. It is easy to see that these two relatively prime integers belong to the set $I'_{m,t-i,h} \setminus I'_{m,t,h}$. Write the minimal polynomials of α^{j_0} and α^{j_1} as $M_{\alpha^{j_0}}(x)$ and $M_{\alpha^{j_1}}(x)$ respectively. By Propositions 2.2.10 and 2.2.11, we have

$$\begin{aligned}
\operatorname{ord}(f(x)) &\geq \operatorname{lcm}\left(\operatorname{ord}\left(M_{\alpha^{j_0}}(x)\right), \operatorname{ord}\left(M_{\alpha^{j_1}}(x)\right)\right) \\
&= \operatorname{lcm}\left(\frac{2^{hm}-1}{\gcd\left(j_0, 2^{hm}-1\right)}, \frac{2^{hm}-1}{\gcd\left(j_1, 2^{hm}-1\right)}\right) \\
&= 2^{hm} - 1.
\end{aligned}$$

Since the order of a factor of the generator polynomial of a cyclic code is at most the length of the code, we have $\operatorname{ord}(f(x)) = 2^{hm} - 1$ as desired. □

Finally, on the other major parameter, the minimum distance, the following is known for cyclic codes.

Theorem 2.2.15 (BCH bound for binary codes)**.** *Let $g(x)$ be the generator polynomial of a cyclic code of length n and minimum distance d. Let n' be the smallest integer such that n divides $2^{n'} - 1$ and α a primitive nth root of unity in $\mathbb{F}_{2^{n'}}$. If there exist a nonnegative integer b and positive integer $\delta \geq 2$ such that $g(\alpha^{b+i}) = 0$ for $0 \leq i \leq \delta - 2$ in $\mathbb{F}_{2^{n'}}$, then $d \geq \delta$.*

Proposition 2.2.16. Let C be the binary LDPC code of points and t-spaces in $\mathrm{PG}(n, q)$ or $\mathrm{AG}(n, q)$, $n \geq 2$ where points correspond to the positions of the code. If q is odd, then the code is trivial or almost-trivial (dimension 1) depending on whether or not $j \in C$. If q is even then the minimum distance is $(q + 2)q^{n-t+1}$; an example of a code word of this Hamming weight is the incidence vector a cone with as its vertex an $(n - 2 - t)$-space (not included) and as its base a hyperoval.

Proof. For both geometries, the claim for q odd follows immediately from the matrix being full rank or full rank minus one.

It follows from [21, Theorem 1] that the minimum for q even of the $\mathrm{AG}(n, q)$-code is as claimed. For $\mathrm{PG}(n, q)$, this is lower bound by Theorem 2.2.15, the example in the problem statement shows equality. □

Proposition 2.2.17. Let C be the binary LDPC code of points and t-spaces in $\mathrm{EG}(n, q)$ or $\mathrm{AG}(n, q)$, $n \geq 2$, where points correspond to the positions of the code. If q is odd, then the code is trivial or almost-trivial (dimension 1) depending on whether or not $j \in C$. If q is even then the minimum distance is $(q + 2)q^{n-t+1} - 1$; an example of a code word is a cone with as its vertex an $(n - 2 - t)$-space (not included) and as its base a hyperoval containing the point $(0, 0, \ldots, 0)$.

Proof. The claim for q odd follows again from the matrix being full rank or full rank minus one.

For q odd, let $q = 2^h$, then we can work as follows. It is straightforward to see that the positive integers a smaller than $2^{h(m-t)} + 2^{h(m-t-1)+1} - 1$ all satisfy the condition that

$$\max_{0 \le i \le h} \mathrm{w}_{2^h}(a2^i) \le (m-t)(2^h - 1).$$

By Theorem 2.2.13, for all positive integers $i \le 2^{h(m-t)} + 2^{h(m-t-1)+1} - 2$, the generator polynomial $g(x)$ of $\mathcal{E}_{m,t,2^h}$ has $x - \alpha^i$ as its factors. Hence, by Theorem 2.2.15, we have $d \ge (2^{hm} + 2)2^{m-t-1} - 1$. The code word construction in the problem statement shows that we have equality in this bound. □

For codes over a field with characteristic different from two, the minimum distance is not always known.

Conjecture 2.2.18. *Let A be the incidence matrix of $\mathrm{PG}(2, q)$. For $q \neq p \neq 2$, the minimum distance of the code defined by A as its parity check matrix, is an open problem. It is conjectured to be $d = 2q - \frac{q-p}{p-1}$.*

For more background on these codes, we refer to [5, 43, 89].

2.3 LDPC codes from linear representations

One class of geometries studied for this purpose are linear representations of geometries. One case that received a lot of attention lately is $T_2^*(\mathcal{K})$, with $\mathcal{K}$ a hyperoval [113, 139]. Here the minimum weight is known, the dimension is known when the characteristic of the code field $\mathrm{char}\,\mathbb{K} \neq 2$ and then we also know that the code is generated by its code words of minimum weight. Other structures studied in less detail in [113] include $T_2^*(\mathcal{B})$ with $\mathcal{B}$ a Baer subplane, $T_2^*(\mathcal{U})$ with $\mathcal{U}$ a unital, and $T_2^*(\mathcal{L})$ with $\mathcal{L}$ the pointwise union of two intersecting lines. One linear representation that has received a lot of attention is $LU(3, q)^D$ (and its dual $LU(3, q)$, which is not a linear representation) [78, 77, 124, 139, 140]. In [77], the authors conjecture the binary dimension of the associated code to be

$$\frac{q^3 - 2q^2 + 3q - 2}{2}$$

if q is odd. Here $q = p^h$ with p prime denotes the order of the finite field of the geometry. This conjecture was proven in [124]. Over all code fields with $\mathrm{char}\,\mathbb{K} \neq p$, I proved this in [139]. In this section we present a uniform approach, having both of these as an immediate corollary; when $\mathrm{char}\,\mathbb{K} \neq p$, we compute the minimum weight, the rank and the code rate of the code, we classify the code words of minimum weight and we prove that every code word is a linear combination of the code words of minimum weight. These results were published in Adv. Math. Commun [140].

2.3.1 Preliminaries

Let us begin by introducing some basic notations and definitions.

Definition 2.3.1 ([1])**.** Let $\mathrm{PG}(3,q)$ be the 3-dimensional projective space over the field $\mathbb{F}_q$. Let $\Pi_0 := \mathrm{PG}(2,q)$ be a (hyper)plane in it and let $\mathcal{K}$ be an arbitrary subset of the points of that hyperplane. We define the geometry $T_2^*(\mathcal{K})$ in the following way:

- the points of $T_2^*(\mathcal{K})$ are the affine points, being the points of $\mathrm{PG}(3,q) \setminus \mathrm{PG}(2,q)$,
- the lines are the affine lines of $\mathrm{PG}(3,q)$ which pass through a point of $\mathcal{K}$,
- the incidence relation is inherited from $\mathrm{PG}(3,q)$.

Remark 2.3.2. Note that through every (affine) point we have $|\mathcal{K}|$ lines, one through each point of $\mathcal{K}$, while every line contains q points. In total there are q^3 points and $|\mathcal{K}|q^2$ lines: q^2 through each point of $\mathcal{K}$.

Remark 2.3.3. Let $N = |\mathcal{K}|$ and let H be the $q^3 \times Nq^2$ incidence matrix of $T_2^*(\mathcal{K})$, where points correspond to rows of H and lines correspond to columns of H and to the positions in the code. Let C be the linear code with H as its parity check matrix, over an arbitrary finite field $\mathbb{K}$. One can associate a *coefficient* to each line in a code word w, being its value at the corresponding position. A word $c = (c_1, \ldots, c_{Nq^2}) \in \mathbb{K}^{Nq^2}$ is in C if and only if $w \cdot H^T = \vec{0}$, hence (since $H_{ji} = 1 \Leftrightarrow \ell_i \ni p_j$) if and only if

$$\sum_{\ell_i} c_i H_{ji} = \sum_{\ell_i \ni p_j} c_i = 0$$

as an element of $\mathbb{K}$ for every point p_j. Alternatively formulated: a word is a code word of C if and only if the sum of the coefficients of the lines through every point equals 0 over $\mathbb{K}$.

Definition 2.3.4. Let $r_i, r_j \in \mathcal{K}$ with $i < j$ and let π be a projective two-dimensional plane through r_i, r_j different from Π_0. The *plane word through r_i and r_j in π* is the code word with

- $+1$ in the positions corresponding to the lines of π through r_i,
- -1 in the positions corresponding to the lines of π through r_j,
- 0 in the positions corresponding to all other lines.

Notation 2.3.5. We denote by C' the code generated by all plane words. Given a plane word w through p_i and p_j, define $T(w)$ to be the plane π in Definition 2.3.4 and $L(w)$ to be the line $p_i p_j$ in Definition 2.3.4.

Definition 2.3.6. Let L be a line in Π_0 containing at least two points of $\mathcal{K}$. Let π be a projective (two-dimensional) plane through L different from Π_0, and let $p_0, \ldots, p_{k-1}$ be the points of $L \cap \mathcal{K}$. We define a *generalized plane word in π* to be a code word with

- a_0 in the positions corresponding to the lines of π through p_0,

- a_1 in the positions corresponding to the lines of π through p_1,
- ...
- a_{k-2} in the positions corresponding to the lines of π through p_{k-2},
- $-a_0 - a_1 - \cdots - a_{k-2}$ in the positions corresponding to the lines of π through p_{k-1},
- 0 in the positions corresponding to all other lines,

for some scalars $a_0, a_1, a_2, \ldots, a_{k-2} \in \mathbb{K}$.

Remark 2.3.7. Note that if a line would contain two points of $\mathcal{K}$, then it is not a line of $T_2^*(\mathcal{K})$, because it is contained in the plane at infinty and hence not an affine line.

Remark 2.3.8. Note that a sum of plane words in a fixed plane π is a generalized plane word in π, and a sum of generalized plane words in π is still a generalized plane word in π. Moreover, if $\pi \cap \mathcal{K} = \{p_0, \ldots, p_{k-1}\}$, then the set of generalized plane words in π is spanned by the plane words through $(p_0, p_1), (p_0, p_2), \ldots, (p_0, p_{k-1})$. Hence, C' is also the code spanned by all generalized plane words and we need at most one generalized plane word per plane to obtain any word of C'.

Remark 2.3.9. It is known that $T_2^*(\mathcal{K})$ is a partial geometry if and only if $\mathcal{K}$ is a (maximal) $\{qn - q + n; n\}$-arc for some $n \geq 1$ (see [136]) and $T_2^*(\mathcal{K})$ is a semipartial geometry if and only if $\mathcal{K}$ is a Baer subplane or a unital (see [31]). A good general reference on $T_2^*(\mathcal{K})$ is [29].

2.3.2 Dimension of C'

Notation 2.3.10. Denote by $\mathcal{L}$ the set of projective lines at infinity that contain at least one point of $\mathcal{K}$. Denote by L_N the size of $\mathcal{L}$, i.e.

$$\mathcal{L} = \{\ell_1, \ell_2, \ldots, \ell_{L_N}\}$$

and by L_S the summed size of $\mathcal{L}$, i.e.

$$L_S = \sum_{\ell \in \mathcal{L}} |\ell \cap \mathcal{K}|.$$

Let $\mathbb{K}$ be an arbitrary field with $\operatorname{char} \mathbb{K} \neq p$.

Lemma 2.3.11. *Let $\ell \in \mathcal{L}$ be a line in the plane at infinity, containing exactly k points of $\mathcal{K}$. Then there are exactly $q(k-1)$ linearly independent plane words among all plane words w with $L(w) = \ell$.*

Proof. Number the k points $p_0, p_1, \ldots, p_{k-1}$, then the pairs

$$(p_0, p_1), (p_0, p_2), \ldots, (p_0, p_{k-1})$$

each yield q different plane words, and these are linearly independent. Now for all other pairs, the plane words through (p_i, p_j), with $i < j$, can be written as the difference of the corresponding plane words through (p_0, p_i) and (p_0, p_j). Hence the result follows. □

Lemma 2.3.12. *Fix an arbitrary point $p_0 \in \mathcal{K}$. Let*[3] $\sum_{i=1}^{n} \lambda_i v_i = \vec{0}$ *be a linear combination of generalized plane words yielding the zero word, at most one generalized plane word per plane. If $L(v_i) = L(v_j)$ and this line contains p_0, then $\lambda_i = \lambda_j$.*

Proof. The q affine lines through p_0 in $T(v_i)$ each get a contribution of λ_i from v_i. The q affine lines through p_0 in $T(v_j)$ each get a contribution of λ_j from v_j. All other generalized plane words v_m contribute equally much to the sum of both sets of q lines (namely λ_m if $T(v_m)$ contains p_0 and 0 otherwise). Denote by R the total summed contribution to both sets of q lines.

Since the total sum of all contributions is 0 for every line (since we assumed that this linear combination yields the zero word) we have $q\lambda_i + R = 0 = q\lambda_j + R$, hence $q\lambda_i = -R = q\lambda_j$. Since $q \neq 0$ as an element of $\mathbb{K}$, it follows that $\lambda_i = \lambda_j$. □

Corollary 2.3.13. *We did not assume $\lambda_i \neq 0$, hence if one of the generalized plane words v_i appears in the linear combination with a nonzero λ_i, then all generalized plane words v_j through the same line at infinity should appear with $\lambda_j = \lambda_i$. Hence if we start from an empty code (considered as vector space), and we consider one by one all lines ℓ at infinity and we add the q generalized plane words through ℓ to this vector space, then each line increases the dimension with at least $(|\ell \cap \mathcal{K}| - 1)(q-1)$, since the codimension can be at most $|\ell \cap \mathcal{K}| - 1$.*

Theorem 2.3.14. *The dimension of C' is $(N-1) + (q-1)(L_S - L_N)$.*

Proof. Take any point p_0 and look at the lines $L_0, \ldots, L_q$ through p_0 in the plane at infinity. The plane words through (p_0, p), for $p \in \mathcal{K} \setminus \{p_0\}$, form a basis for the linear combinations of plane words on the lines $L_0, \ldots, L_q$. Starting from an empty vector space V as described in Corollary 2.3.13, these plane words contribute $(N-1)q$ to the dimension, because of Lemma 2.3.11. Now adding every other line $L \in \mathcal{L}$ at infinity, not through p_0, contributes

- at least $(|L \cap \mathcal{K}| - 1)(q-1)$ to $\dim V$, since Lemma 2.3.12 states that in any linear combination of generalized plane words yielding the zero word, all planes through L appear with the same coefficient, hence we miss at most $|L \cap \mathcal{K}| - 1$ degrees of freedom,
- and at most $(|L \cap \mathcal{K}| - 1)(q-1)$ to $\dim V$, since one can write the zero word as a linear combination of plane words through $(p_0, p), (p_0, p'), (p, p')$ for any two points $p, p' \in L \cap \mathcal{K}$ (note that all plane words through lines through p_0 are already in the code at this point).

Therefore, the dimension is exactly

$$(N-1)q + \sum_{p_0 \notin L \in \mathcal{L}} (|L \cap \mathcal{K}| - 1)(q-1).$$

[3] By convention, we choose all generalized plane words containing lines through p_0 in their support, to have coefficient $+1$ on the lines through p_0 (this can always be accomplished by scaling the λ_i).

Note that $\sum_{p_0 \in L \in \mathcal{L}} (|L \cap \mathcal{K}| - 1) = N - 1$ since both represent all points of $\mathcal{K}$ except for p_0. Hence

$$\begin{aligned} \dim(C') &= (N-1)q + \textstyle\sum_{p_0 \notin L \in \mathcal{L}} (|L \cap \mathcal{K}| - 1)(q-1) \\ &= (N-1) + (q-1) \textstyle\sum_{L \in \mathcal{L}} (|L \cap \mathcal{K}| - 1) \\ &= (N-1) + (q-1) \left(\left(\textstyle\sum_{L \in \mathcal{L}} |L \cap \mathcal{K}| \right) - |\mathcal{L}| \right) \\ &= (N-1) + (q-1)(L_S - L_N). \end{aligned}$$

□

Remark 2.3.15. We now know that C' is a linear $[Nq^2, N - 1 + (q-1)(L_S - L_N)]$-code. There is no general expression for $L_S - L_N$ in terms of q and N. However, there is an easy algorithm to compute $L_S - L_N$ for an arbitrary set $\mathcal{K}$:

Let $\mathcal{K}$ be an arbitrary subset of $\mathrm{PG}(2, q)$. Fix any point p_0 (inside $\mathcal{K}$ or outside of $\mathcal{K}$). Call a line through p_0

- a *secant* if it contains two or more points of $\mathcal{K} \setminus \{p_0\}$,
- a *tangent* if it contains exactly one point of $\mathcal{K} \setminus \{p_0\}$, and
- a *passant* if it contains no points of $\mathcal{K} \setminus \{p_0\}$.

When adding/removing a point p_0,

- L_S increases/decreases by $q + 1$, while
- L_N increases/decreases by the number of passants through p_0.

Hence, $L_S - L_N$ increases/decreases by the number of non-passant lines through p_0.

Some examples:

- If $\mathcal{K}$ is a k-arc, then adding the ith point increases $L_S - L_N$ by $i - 1$. Hence,

$$L_S - L_N = \sum_{i=1}^{k} (i-1) = \frac{k(k-1)}{2}.$$

- If $\mathcal{K}$ is the pointwise union of two intersecting lines, then adding the points on the first line increases $L_S - L_N$ by 1 each time (except for the first point), while adding the other q points increases it by $q + 1$ each time. Hence, in this case

$$L_S - L_N = q + q(q+1) = q^2 + 2q.$$

Remark 2.3.16. Since one has in general that $L_S = N(q+1)$, the dimension formula can be rewritten as $q^2 N - q^3 + (q-1)(q^2 + q + 1 - L_N)$. Since the parity check matrix is a $q^3 \times q^2 N$ matrix, this means that the rank deficiency of the parity check matrix is $q - 1$ times the number of lines at infinity, skew to $\mathcal{K}$. It may be interesting to find out if there exists a more direct way to obtain this formula.

2.3.3 Dimension of C

We will compute the dimension of $T_2^*(\mathcal{K})$ as follows. First we will compute $\dim C$ in the case $\mathcal{K} = \mathrm{PG}(2,q)$, and find that it equals $\dim C'$ in that case, hence $C = C'$. Then we will present a technique to keep this property valid while removing arbitrary points from $\mathcal{K}$. Every subset $\mathcal{K}$ of $\mathrm{PG}(2,q)$ can be obtained by removing a finite number of points from $\mathrm{PG}(2,q)$, so the conclusion will follow. As in the previous section, we will always assume that $q \neq 0$ over $\mathbb{K}$, i.e. $\mathrm{char}\,\mathbb{K} \neq p$.

First, we study the case that $\mathcal{K} = \mathrm{PG}(2,q)$, so then we simply have $T_2^*(\mathcal{K}) = \mathrm{AG}(3,q)$. Remark that $\mathrm{AG}(3,q)$ is a 2-$(q^3,q,1)$ block design, using lines as blocks. Denote by A_n the incidence matrix of $\mathrm{AG}(n,q)$, with $n \geq 2$, where points correspond to rows and lines correspond to columns.

It is a classical result in design theory (see [22] for a proof and more general background on designs) that $A_n A_n^T = (q^{n-1} + q^{n-2} + \cdots + q^2 + q)I + J$, where J denotes the $q^n \times q^n$ matrix with all entries equal to 1. This has determinant

$$(q^n + q^{n-1} + \cdots + q^2 + q)(q^{n-1} + q^{n-2} + \cdots + q^2 + q)^{q^n - 1}.$$

In fact, $A_n A_n^T$ has $q^n - 1$ eigenvalues equal to $q^{n-1} + q^{n-2} + \cdots + q^2 + q$ and one eigenvalue equal to $q^n + q^{n-1} + \cdots + q^2 + q$. The determinant is non-zero when $q^{n-1} + q^{n-2} + \cdots + q^2 + q \neq 0$ and $q^n + q^{n-1} + \cdots + q^2 + q \neq 0$. For most choices of the characteristic, this already shows that A_n has full rank, however in some cases further study is required:

- The case $q = 0$ has been excluded by our assumptions. In fact, A_n does not have full rank in this case. We will further assume $q \neq 0$.
- The case $q^{n-1} + q^{n-2} + \cdots + q + 1 = 0$ is easily solved. Note that this implies $q \neq 0$ and $q^{n-2} + q^{n-3} + \cdots + q + 1 \neq 0$, hence in this case only one of the eigenvalues of $A_n A_n^T$ equals zero over $\mathbb{K}$, and we see that $(1, \ldots, 1)^T$ is an eigenvalue corresponding to this eigenvector. Hence $(1, \ldots, 1)^T$ is (up to scalar multiples) the only eigenvector corresponding to this eigenvalue, but one can verify that

  $$(1, \ldots, 1)A_n^T = q^n(1, \ldots, 1) \neq \vec{0}$$

  since $q \neq 0$. Hence, this is not a code word, and hence there is no $v \neq \vec{0}$ such that $A_n v = \vec{0}$, i.e. A_n has full rank in this case.
- The case $q^{n-2} + q^{n-3} + \cdots + q + 1 = 0$ is more difficult. Purely combinatorial approaches seem to fail, but a geometric trick works. We will now develop this technique to find the rank of A_n over $\mathbb{K}$ for the case $q^{n-2} + q^{n-3} + \cdots + q + 1 = 0$.

Lemma 2.3.17. *Let $k \geq 2$. If the incidence matrix of $\mathrm{AG}(k+1,q)$ is rank deficient over $\mathbb{K}$, then the incidence matrix of $\mathrm{AG}(k,q)$ is also rank deficient over $\mathbb{K}$.*

Proof. Assume that the incidence matrix of $\mathrm{AG}(k+1,q)$ is rank deficient. Since $k \geq 2$, there are more lines than points. Hence, rank deficiency means that there exists a linear

combination of points whose corresponding line-incidence vectors yield the zero vector (zero for each line):

$$\sum_{p_i \in \mathrm{AG}(k+1,q)} c_i p_i = \vec{0}$$

with not all $c_i = 0$ over $\mathbb{K}$.

Now consider any hyperplane $\Pi \cong \mathrm{AG}(k,q)$ in our $\mathrm{AG}(k+1,q)$ which contains at least one point with non-zero coefficient in the linear combination. For all lines contained in Π, the linear combination of the point-line incidence vectors has to be zero as well. I.e. in Π we also have $\sum_{p_i \in \Pi} c_i p_i = \vec{0}$. Since Π contains at least one point with non-zero coefficient in the linear combination, this linear combination is nontrivial. Hence, the incidence matrix of $\mathrm{AG}(k,q)$ is rank deficient as well. □

Theorem 2.3.18. *The incidence matrix of* $\mathrm{AG}(n,q)$ *(with* $n \geq 2$*),* $q = p^h$ *with* p *prime, has full rank over all fields with* $\operatorname{char}\mathbb{K} \neq p$.

Proof. We will prove this by induction on n. For $n = 2$ this is clear from the remarks in the beginning of this section, since the problematic case above ($q^{n-2} + \cdots + q + 1 = 0$) reduces to $1 = 0$, hence can be excluded immediately. For each $n \geq 2$ it follows by contraposition of Lemma 2.3.17 that if the statement is true for n, then it is also true for $n+1$. Hence, by induction, it is true for all $n \geq 2$. □

We have proven that if A_n has full rank for all $n \geq 2$ when $\operatorname{char}\mathbb{K} \neq p$. Explicitly, for the general setting of $\mathcal{T}_n^*(\mathcal{K})$ with $\mathcal{K} = \mathrm{PG}(n,q)$ (which yields exactly $\mathrm{AG}(n+1,q)$) the geometry has $q^n(q^n + q^{n-1} + \cdots + q + 1)$ lines and q^{n+1} points, hence Theorem 2.3.18 yields

$$\dim C = q^n(q^n + q^{n-1} + \cdots + q^3 + q^2 + 1).$$

Now compare this to the result of Section 3. In Section 3 we worked with $n = 2$, hence $N = q^2 + q + 1$ and $T_2^*(\mathcal{K}) = \mathrm{AG}(3,q)$. This gives us

$$\begin{aligned} \dim(C') &= (N-1) + (q-1)(L_S - L_N) \\ &= q^2 + q + (q-1)((q^2+q+1)(q+1) - (q^2+q+1)) \\ &= q^2(q^2+1) \\ &= \dim C. \end{aligned}$$

and hence $\dim C = \dim C'$. Hence, the code associated with $\mathrm{AG}(3,q)$ is spanned completely by its plane words. However, for $n > 3$, Theorem 2.3.14 is no longer valid. We finish with a conjecture for the higher dimensions:

Conjecture 2.3.19. *If* $\mathcal{K} = \mathrm{PG}(n,q)$ *with* $n > 2$ *and* $\operatorname{char}\mathbb{K} \neq p$ *then the code associated with* $\mathcal{T}_n^*(\mathcal{K}) = \mathrm{AG}(n+1,q)$ *over* $\mathbb{K}$ *also has* $\dim C' = \dim C$.

Now we are ready to do the general case. The main idea here is the following: if we remove a point from $\mathcal{K}$, we claim that the property that the code is spanned by its plane words remains valid. To distinguish between different point sets, we denote by $C_\mathcal{K}, C'_\mathcal{K}$ respectively the full code and the plane words code associated with $T_n^*(\mathcal{K})$. Similarly, we denote by $L_{N,\mathcal{K}}, L_{S,\mathcal{K}}$ the respective values of L_N and L_S for the set $\mathcal{K}$.

Theorem 2.3.20. *Let $\mathcal{K}$ be a nonempty subset of* $\mathrm{PG}(2,q)$ *and let* $\operatorname{char}\mathbb{K} \neq p$. *We have* $\dim C_{\mathcal{K}} = \dim C'_{\mathcal{K}}$ *(and hence $C_{\mathcal{K}} = C'_{\mathcal{K}}$).*

Proof. According to Theorem 2.3.14, if $T \subseteq \mathrm{PG}(2,q)$, then removing a point p_0 from T decreases $\dim C'$ by

$$\dim C'_T - \dim C'_{T\setminus\{p_0\}} = 1 + (q-1)((L_{S,T} - L_{N,T}) - (L_{S,T\setminus\{p_0\}} - L_{N,T\setminus\{p_0\}}))$$

and hence decreases $\dim C$ by at least this amount.

Fix one point $p_0 \in \mathcal{K}$. Now remove all other points of $\mathrm{PG}(2,q)$ one by one, first the points outside of $\mathcal{K}$ then the points inside $\mathcal{K}$, except for p_0. Denote by T_i the set in the intermediary step with i points:

$$T_{|\mathrm{PG}(2,q)|} = \mathrm{PG}(2,q), \quad T_N = \mathcal{K}, \quad T_1 = \{p_0\},$$

and define $Q = |\mathrm{PG}(2,q)| - 1$. Here, $|\mathrm{PG}(2,q)|$ denotes the number of points in $\mathrm{PG}(2,q)$, which is q^2+q+1. Note that in any code word, every affine point lies on either 0 or at least 2 lines of the support of that code word, hence $C_{\{p_0\}} = \{\vec{0}\}$. Note that $C_{\{p_0\}}$ is simply $C_{\mathcal{K}}$ with $\mathcal{K} = \{p_0\}$. Hence, we have

$$\begin{aligned}
\dim C_{\mathrm{PG}(2,q)} &= \dim C_{\mathrm{PG}(2,q)} - \dim C_{\{p_0\}} \\
&= \sum_{i=1}^{Q} \left(\dim C_{T_{i+1}} - \dim C_{T_i}\right) \\
&\geq \dim C'_{T_2} - \dim C'_{T_1} + \sum_{i=2}^{Q} \left(\dim C_{T_{i+1}} - \dim C_{T_i}\right) \\
&\geq \cdots \\
&\geq \sum_{i=1}^{Q} \left(\dim C'_{T_{i+1}} - \dim C'_{T_i}\right) \\
&= \dim C'_{\mathrm{PG}(2,q)} - \dim C'_{\{p_0\}} \\
&= \dim C'_{\mathrm{PG}(2,q)}.
\end{aligned}$$

It was proven in the previous subsection that $\dim C_{\mathrm{PG}(2,q)} = \dim C'_{\mathrm{PG}(2,q)}$, hence we must have equality in each inequality "$\geq$". This means that

$$\dim C_{T_{i+1}} - \dim C_{T_i} = 1 + (q-1)((L_{S,T_{i+1}} - L_{N,T_{i+1}}) - (L_{S,T_i} - L_{N,T_i}))$$

for each i. A simple induction gives $\dim C_{T_i} = \dim C'_{T_i}$ for all i, in particular for $i = N$ we have $\dim C_{\mathcal{K}} = \dim C'_{\mathcal{K}}$. □

Hence for $T_2^*(\mathcal{K})$ in general it is now proven that $\dim C = \dim C'$ and the code C is generated completely by its plane words.

Remark 2.3.21. If Conjecture 2.3.19 is true, then Theorem 2.3.20 can be extended to arbitrary subsets of $\mathrm{PG}(n,q)$: then we have $\dim C = \dim C'$ for the code associated with $T_n^*(\mathcal{K})$ with arbitrary $\mathcal{K} \subseteq \mathrm{PG}(n,q)$.

Remark 2.3.22. As an immediate consequence, we get that in the binary code associated with $T_2^*(\mathcal{K})$ for q odd, all code words have even weight.

Corollary 2.3.23. *Since $LU(3,q)^D$ is projectively equivalent to $T_2^*(\mathcal{K})$ with $\mathcal{K}$ a conic minus one point [78], and since the rank of a matrix is equal to the rank of its transposed, it follows that the dimension of C in this case is*

$$\frac{q^3 - 2q^2 + 3q - 2}{2}.$$

Hence, Theorem 2.3.20 extends the main result from [124].

2.3.4 The minimum distance of C

Now that the dimension and structure of C are known, we can attack another one of its key properties: the minimum distance. In some sporadic cases the minimum distance is known [78, 113], however in most cases one only has lower bounds from the tree bound [133], the bit-oriented bound and the parity-oriented bound [134].

Theorem 2.3.24. *For any finite field $\mathbb{K}$, all code words c with $w(c) < 2q$ must be contained in a single plane. If $w(c) = 2q$, then either c is a plane word, supp(c) is the set of lines of a hyperbolic quadric with two intersecting lines contained in $\mathcal{K}$, or* $\operatorname{char}\mathbb{K} = p = 2$.

Proof. This follows from [113], Proposition 4. □

Now, we will use the structure of C to sharpen this result and classify the minimum weight code words.

Theorem 2.3.25. *If* $\operatorname{char}\mathbb{K} \neq p$, *there are no code words $c \in C$ with $w(c) < 2q$. If $w(c) = 2q$, then either c is a plane word or supp(c) is the set of lines of a hyperbolic quadric with two intersecting lines contained in $\mathcal{K}$.*

Proof. The second part follows immediately from Theorem 2.3.24. For the first part, assume that there exists a code word with $w(c) < 2q$ and let U be its support. By Theorem 2.3.24, the support of this code word is contained in a plane T. Define $m = |\mathcal{K} \cap T|$ and write $\mathcal{K} \cap T = \{p_1, \ldots, p_m\}$. Since supp$(c) \subset T$, we have that c is a generalized plane word in T. Hence, either each of the q lines through p_i appear in U, or none of them do. Since one needs either 0 or at least 2 lines through a point, it follows that $w(c) \geq 2q$, a contradiction. □

Remark 2.3.26. Note that it may actually happen that there are minimum weight code words other than plane words – Theorem 2.3.24 classfies them. However, it follows from Theorem 2.3.20 that these other minimum weight code words are also a linear combination of plane words. Hence, even in this case, the statements 'C is generated by its (generalized) plane words' and 'C is generated by its code words of minimum weight' are equivalent.

So far we have proven that C is a linear $[Nq^2, N-1+(q-1)(L_S-L_N), 2q]$-code and it is completely generated by its minimum weight code words.

Now, let us see what happens for $T_n^*(\mathcal{K})$ with $n > 2$, assuming Conjecture 2.3.19 is true.

Theorem 2.3.27. *If Conjecture 2.3.19 is true, then $d(C) = 2q$ is true for $T_n^*(\mathcal{K})$ with arbitrary $n \geq 2$ and arbitrary $\mathcal{K} \subseteq \mathrm{PG}(n,q)$ (still assuming $\operatorname{char}\mathbb{K} \neq p$).*

Proof. Assume there exists a non-zero code word c with $w(c) < 2q$ and let U be its support. Let T be any plane and define $t = |U \cap T|$. Since the sum of the coefficients has to be 0 in each point, any point on a line of the support lies on at least one other line of the support. Hence we have

$$t(q-t+1) < 2q-t,$$

meaning $t > q$ or $t < 2$. Since t is an integer, this means $t \geq q+1$ or $t \leq 1$.

Now, if there are at least two planes for which $t \geq q+1$, then $w(c) \geq 2q+1$, contradiction. Hence there is at most one such plane and U is completely contained in this plane. The rest of the proof can be copied from Theorem 2.3.25. □

The following theorem summarizes the results obtained so far:

Theorem 2.3.28. *The code associated with $T_2^*(\mathcal{K})$ over any field $\mathbb{K}$, with $\operatorname{char}\mathbb{K} \neq p$, is a linear $[Nq^2, N-1+(q-1)(L_S-L_N), 2q]$-code and it is completely generated by its minimum weight code words. If Conjecture 2.3.19 is true, then for $\operatorname{char}\mathbb{K} \neq p$ the code associated with $T_n^*(\mathcal{K})$ also has $d(C) = 2q$ and it is also generated completely by its minimum weight code words.*

2.3.5 Some practical considerations and further work

A commonly used approach when constructing good LDPC codes is the maximization of the girth of its Tanner graph [66, 100, 149], since high girth decreases the dependence between passing messages in the belief-propagation sum-product algorithm. The Tanner graph of $T_2^*(\mathcal{K})$ always has a girth of at least 6. If $\mathcal{K}$ is an arc, then the girth is 8, as $T_2^*(\mathcal{K})$ contains no triangles.

Liu and Pados [95] mention an opposing objective: the minimization of the diameter of the Tanner graph, which brings them to generalized polygons. If $\mathcal{K}$ is not contained within a line, the diameter of $T_2^*(\mathcal{K})$ is at most 6. If $\mathcal{K}$ contains no tangents at infinity, the diameter is as low as 4. Examples of such choices of $\mathcal{K}$ include $\mathcal{K}$ a hyperoval and $\mathcal{K}$ a double blocking set.

There is a unique choice for $\mathcal{K}$ that combines both of the preceding objectives: the case where $\mathcal{K}$ is a hyperoval, which only exists for q even. Then $T_2^*(\mathcal{K})$ is a generalized quadrangle with girth 8 and diameter 4. From the above points of view, this is probably the most appealing case, however, the restriction $\operatorname{char}\mathbb{K} \neq p$ excludes the most important field for practical applications: $\mathbb{F}_2$.

Hyperoval	$q = p^h$	$\dim_{\mathbb{F}_2} C$	$\dim_{\mathbb{R}} C = \dim_{\mathbb{R}} C'$	$\dim_{\mathbb{F}_2} C'$
Regular hyperoval	$q = 2$	9	9	8
Regular hyperoval	$q = 4$	50	50	37
Regular hyperoval	$q = 8$	341	324	194
Regular hyperoval	$q = 16$	2670	2312	1105
Lunelli-Sce hyperoval	$q = 16$	2550	2312	1107
Regular hyperoval	$q = 32$	22248	17424	6578
Translation hyperoval	$q = 32$	21258	17424	6608
Cherowitzo hyperoval	$q = 32$	20358	17424	6613
Payne hyperoval	$q = 32$	20388	17424	6613
Segre hyperoval	$q = 32$	20553	17424	6613
O'Keefe-Penttila hyperoval	$q = 32$	20343	17424	6613
Regular hyperoval	$q = 64$	188665	135200	39937
Adelaide hyperoval	$q = 64$	169772	135200	40312
Subiaco I hyperoval	$q = 64$	169254	135200	40312
Subiaco II hyperoval	$q = 64$	169388	135200	40309

Table 2.1: Simulation results for the binary codes associated with $T_2^*(\mathcal{K})$ where $\mathcal{K}$ is a hyperoval.

For the binary code associated with $T_2^*(\mathcal{K})$ when $\mathcal{K}$ is a hyperoval, the results in this section are no longer valid. Lemma 2.3.12 no longer guarantees the lower bound on $\dim C'$ and since the 2-rank is at most the real rank, the dimension of C could be larger than what we have derived in this section. In Table 2.1, we have calculated the dimension (and hence the code rate, which is dimension over length) of the $\mathbb{F}_2$-code and $\mathbb{R}$-codes associated with $T_2^*(\mathcal{K})$ with $\mathcal{K}$ a hyperoval, by computer simulations, for multiple types of hyperovals in $\mathrm{PG}(2, 2^h)$. For $h \leq 5$, these are the only hyperovals (for proofs of these facts, see [52, 110, 112, 118]). For $h = 6$, it is commonly believed that these are the only hyperovals, but a proof has not been found yet. For $h > 6$, a classification is not even conjectured and the computations also become unfeasible.

We see that the results indeed deviate from the numbers in Theorem 2.3.28. In this case we get even better parameters: the dimension increases while no other visible parameters change. However, we lose the structural property that the code is spanned by its code words of minimum weight. For other choices of $\mathcal{K}$ when q is even, even the minimum distance may decrease. This has been investigated more closely in [113]. In general, only a minimum weight of $q + 1$ can be guaranteed.

If one wants to maintain the structure property and still use a binary code, then q must be odd. Some examples that are 'near-optimal' choices for $\mathcal{K}$ in these cases include:

- $\mathcal{K}$ is a $(q + 1)$-arc (and hence a conic by [119]). Compared to the case where $\mathcal{K}$ is a hyperoval, the dimension and rate are slightly smaller, the code is a bit shorter in length and the girth remains 8. The diameter is 6 now, but there are only very few pairs of vertices where this distance is actually reached. It would be interesting to perform practical simulations to find out if this case still guarantees a fast average decoding

speed.

- $\mathcal{K}$ has no tangents. For example, $\mathcal{K}$ is a dual double blocking set. Here the Tanner graph of $T_2^*(\mathcal{K})$ has girth 6 and the diameter is still 4, but the length of the code is necessarily longer compared to the case where $\mathcal{K}$ is a hyperoval, without an increase in minimum distance.

To finish, we take a look ahead on possible further work on this subject. Each of the following could be a significant contribution to the understanding of this class of codes.

- For $T_n^*(\mathcal{K})$, with $n > 2$, little is known. If Conjecture 2.3.19 is true, it could be interesting to find a generalization of the formula in Theorem 2.3.14 and to analyze its geometric interpretation. Regarding the optimal choice of $\mathcal{K}$, there is no n-dimensional equivalent of the hyperoval, so it would be interesting to know which choices for $\mathcal{K}$ yield interesting geometries (if any).
- If $\operatorname{char}\mathbb{K} = p$, few results in this section remain valid. The only trick that works completely when $\operatorname{char}\mathbb{K} = p$ is that if the incidence matrix of $\mathrm{AG}(n+1,q)$ is rank deficient, then the incidence matrix of $\mathrm{AG}(n,q)$ is rank deficient. A suited structural property could potentially be preserved in a way similar to Section 4.2. This suggests that it may be a good help to first find out the structure of the base case $\mathrm{AG}(2,q)$, especially which code words remain valid if we remove certain classes of parallel lines. Another indication that this may be an interesting topic is the minimum distance. From Theorem 2.3.24, code words c with $w(c) < 2q$ are necessarily contained within a plane. If one knows the structure of the LDPC code associated with $\mathrm{AG}(2,q)$, one is likely to find a general result on the minimum weight. Until now, the only known lower bounds are the bounds in [113].
- If $\operatorname{char}\mathbb{K} = p$, it would be useful to find a structure or dimension result even just for special cases. Even for $T_2^*(\mathcal{K})$ with $\mathcal{K}$ a hyperoval this seems a lot harder. From the simulation results in Table 2.1, one can see that the dimensions are different between different types of hyperovals. This may be related to the approach in [113]: $\dim C$ may depend on how many points of a conic are contained in $\mathcal{K}$, while the difference in $\dim C'$ between regular/translation/other hyperovals may be related to Remark 7 in [113], since other hyperovals are not known to have such special points.

2.4 LDPC codes from Hermitian varieties

A major class of geometries that has been used for LDPC decoding is that of the classical generalized quadrangles, and in particular the quadrics and Hermitian varieties. Here, the positions of the code correspond to the points of the geometry, and the parity check rows are the incidence vectors of the generators (i.e. the subspaces of maximal dimension).

Due to the structure of quadrics and Hermitian varieties, there are essentially five different families of geometries to consider: $Q(2n,q)$, $Q^+(2n+1,q)$, $Q^-(2n+1,q)$, $\mathcal{H}(2n,q^2)$ and $\mathcal{H}(2n+1,q^2)$, with q a prime power. The best known results on the minimum distance and

the classification of small weight code words are summarized in [114]. In one particular case, we were able to make a strong improvement to the state of the art. In particular, we improve the earlier results for $n = 2$, we solve the minimum distance problem for general n, we classify the n smallest types of nonzero code words and we characterize all small weight code words as being a linear combination of these n types. This section is joint work with M. De Boeck and was accepted for publication in Adv. Math. Commun. [30].

The following theorems on $C_n(\mathcal{H}(2n+1,q^2))^\perp$ are known.

Theorem 2.4.1 ([78, Proposition 3.7])**.** *Let $n = 1$. Then the supports of all code words c with $0 < \mathrm{wt}(c) < 3q$ are projectively equivalent, and their Hamming weights are $2(q+1)$. If $\mathrm{wt}(c) \leq \frac{\sqrt{q}(q+1)}{2}$, then c is a linear combination of these code words.*

Theorem 2.4.2 ([114, Theorem 43])**.** *Let $n = 2$. If c is a code word with $0 < \mathrm{wt}(c) \leq 2(q^3+q^2)$ and if q is sufficiently large, then there are only two possible projective equivalence classes for $\mathrm{supp}(c)$, and the Hamming weights of the corresponding code words are $2(q^3+1)$ and $2(q^3+q^2)$. These two types of code words are examples of the code words constructed in Theorem 2.4.9.*

In this section, we will discuss the dual code arising from the points and generators of a Hermitian variety. This improves upon earlier work of [114, Section 5]. We determine the minimum Hamming weight for general n, and we show that if q is sufficiently large, a similar statement to the second part of Theorem 2.4.1 holds for general n. Our main result is as follows.

Theorem 2.4.3. *Let n be any positive integer and let $\delta > 0$ be any constant. If c is a code word with $0 < \mathrm{wt}(c) \leq 4q^{2n-2}(q-1)$ and q is sufficiently large, then there are only n possible projective equivalence classes for $\mathrm{supp}(c)$; call S this set of projective equivalence classes. For every δ sufficiently small compared to q, every code word c with $\mathrm{wt}(c) < \delta q^{2n-1}$ is a linear combination of code words in S. The minimum distance of $C_n(\mathcal{H}(2n+1,q^2))^\perp$ is $2q^{2n-4}(q^3+1)$ for $n \geq 2$.*

2.4.1 The code words

In this section we introduce a set of code words of the code $C_n(\mathcal{H}(2n+1,q^2))^\perp$. From now on, we consider the projective space $\mathrm{PG}(2n+1,q^2)$, $n \geq 1$. We begin with a few lemmata.

Lemma 2.4.4. *The number of generators on $\mathcal{H}(2n+1,q^2)$ is $\prod_{i=0}^{n}(q^{2i+1}+1)$.*

Proof. This, and many other results on Hermitian varieties, can be found in [60, Chapter 23]. □

Notation 2.4.5. Throughout this section, we will denote the number of points in $\mathrm{PG}(m,q)$ by $\theta_m(q) = \frac{q^{m+1}-1}{q-1}$ and the number of points on $\mathcal{H}(m,q^2)$ by

$$\mu_m(q^2) = \frac{(q^{m+1}-(-1)^{m+1})(q^m-(-1)^m)}{q^2-1}.$$

Lemma 2.4.6. *Consider a non-singular Hermitian variety $\mathcal{H}(2n+1,q^2)$ in $\mathrm{PG}(2n+1,q^2)$ and let σ be the corresponding polarity. Let π be a k-dimensional subspace in $\mathrm{PG}(2n+1,q^2)$ such that $\pi \cap \mathcal{H}(2n+1,q^2)$ is a cone $\pi_i H_{k-i-1}$ with $H_{k-i-1} \cong \mathcal{H}(k-i-1,q^2)$ and π_i an i-space, $-1 \leq i \leq \min\{k,n\}$. Then $\pi \cap \pi^\sigma = \pi_i$. Conversely, if $\pi \cap \pi^\sigma$ is an i-space π_i, then $\pi \cap \mathcal{H}(2n+1,q^2)$ is a cone $\pi_i H_{k-i-1}$ with $H_{k-i-1} \cong \mathcal{H}(k-i-1,q^2)$.*

Proof. The first statement is [60, Lemma 23.2.8]; the second statement is a corollary of the first. □

We will use this theorem mostly in the case $k = n$. Using the above lemma, we can prove an easy counting result.

Theorem 2.4.7. *The number of generators on $\mathcal{H}(2n+1,q^2)$ through a fixed k-space on $\mathcal{H}(2n+1,q^2)$, $0 \leq k \leq n$, equals $\prod_{i=0}^{n-k-1}(q^{2i+1}+1)$.*

Proof. Let π_k be a k-space on $\mathcal{H}(2n+1,q^2)$ and let σ be the polarity corresponding to $\mathcal{H}(2n+1,q^2)$. Then π_k^σ is a $(2n-k)$-space intersecting $\mathcal{H}(2n+1,q^2)$ in a cone $\pi_k H$ with $H \cong \mathcal{H}(2n-2k-1,q^2)$. Every generator on $\mathcal{H}(2n+1,q^2)$ through π_k corresponds uniquely to a generator on H. Hence, there are $\prod_{i=0}^{n-k-1}(q^{2i+1}+1)$ generators on $\mathcal{H}(2n+1,q^2)$ through π_k. □

In the construction of the code words we need the following lemma.

Lemma 2.4.8. *Let π be an n-space in $\mathrm{PG}(2n+1,q^2)$ and let μ be a generator of $\mathcal{H}(2n+1,q^2)$. Then $\pi \cap \mu$ and $\pi^\sigma \cap \mu$ are subspaces of the same dimension.*

Proof. We denote $\mu \cap \pi = \pi_j$, a j-space, possibly empty ($j = -1$). It follows that $2n - j = \dim((\mu \cap \pi)^\sigma) = \dim(\langle \mu^\sigma, \pi^\sigma \rangle)$. Using the Grassmann identity and $\mu = \mu^\sigma$ (μ is a generator), we find $\dim(\mu \cap \pi^\sigma) = \dim(\mu) + \dim(\pi^\sigma) - \dim(\langle \mu^\sigma, \pi^\sigma \rangle) = j$. □

Now, we can give the construction of small weight code words in the code $C_n(\mathcal{H}(2n+1,q^2))^\perp$.

Theorem 2.4.9. *Consider $\mathcal{H}(2n+1,q^2)$ and its corresponding polarity σ. Let π be an n-space in $\mathrm{PG}(2n+1,q^2)$. Denote the incidence vector of $\pi \cap \mathcal{H}(2n+1,q^2)$ by v_π and the incidence vector of $\pi^\sigma \cap \mathcal{H}(2n+1,q^2)$ by v_{π^σ}. Then $\alpha(v_\pi - v_{\pi^\sigma})$, $\alpha \in \mathbb{F}_p$, is a code word of $C_n(\mathcal{H}(2n+1,q^2))^\perp$.*

Proof. Let μ be a generator of $\mathcal{H}(2n+1,q^2)$ and denote its incidence vector by v_μ. Using Lemma 2.4.8, we find μ intersects both π and π^σ, or neither. In the first case $|\pi \cap \mu| \equiv |\pi^\sigma \cap \mu| \equiv 1 \pmod q$ and in the second case $|\pi \cap \mu| = |\pi^\sigma \cap \mu| = 0$. In both cases $v_\pi \cdot v_\mu = v_{\pi^\sigma} \cdot v_\mu$. The theorem follows. □

Example 2.4.10. We list the different possibilities for $\pi \cap \pi^\sigma$. Hereby, we use Lemma 2.4.6 for $k = n$ and Theorem 2.4.9. We write $\mathcal{H} = \mathcal{H}(2n+1,q^2)$.

- $\pi \cap \pi^\sigma = \emptyset$. We write $\pi \cap \mathcal{H} = H$ and $\pi^\sigma \cap \mathcal{H} = H'$. We know, $H, H' \cong \mathcal{H}(n, q^2)$. The corresponding code words have weight $2\mu_n(q^2)$.
- $\pi \cap \pi^\sigma = \pi_i$, an i-space, $0 \leq i \leq n-2$. We write $\pi \cap \mathcal{H} = \pi_i H$ and $\pi^\sigma \cap \mathcal{H} = \pi_i H'$, which are both cones, with $H, H' \cong \mathcal{H}(n-i-1, q^2)$. The corresponding code words have weight $2q^{2i+2}\mu_{n-i-1}(q^2)$.
- $\pi \cap \pi^\sigma = \pi_{n-1}$, an $(n-1)$-space. Then $\pi \cap \mathcal{H} = \pi^\sigma \cap \mathcal{H} = \pi_{n-1}$ since $\mathcal{H}(0, q^2)$ is empty. The construction gives rise to the zero code word.
- $\pi \cap \pi^\sigma = \pi_n$, an n-space. Then $\pi = \pi^\sigma = \pi_n \subset \mathcal{H}$. Also in this case, the construction gives rise to the zero code word.

It can easily be checked that among these four cases, the code words with smallest weight are the ones corresponding to $i = n-3$.

Remark 2.4.11. Consider the construction from Theorem 2.4.9, with $\pi \cap \mathcal{H} = \pi_i H_{n-i-1}$ and $\pi^\sigma \cap \mathcal{H} = \pi_i H'_{n-i-1}$. Let P be a point of $\pi_i H_{n-i-1}$ and let P' be a point of $\pi_i H'_{n-i-1}$. We know that $P' \in \pi^\sigma \subseteq P^\sigma$ and $P' \in \mathcal{H}$. Hence, the line PP' is a line of $\mathcal{H}$.

2.4.2 Some counting results

Lemma 2.4.12. *Consider the non-singular Hermitian variety $\mathcal{H}(2n+1, q^2) \subset \mathrm{PG}(2n+1, q^2)$ and let σ be the corresponding polarity. Let τ be a j-space such that $\tau \cap \mathcal{H}(2n+1, q^2) = H_j \cong \mathcal{H}(j, q^2)$, $-1 \leq j \leq n$. The number of generators on $\mathcal{H}(2n+1, q^2)$ skew to τ equals*

$$c_{n,j} := q^{\binom{j+1}{2}} \prod_{k=0}^{n-j-1} (q^{2k+1}+1) \prod_{l=2(n-j)+1}^{2n-j+1} (q^l - (-1)^l).$$

Proof. By [60, Theorem 23.4.2 (i)] we know that the number of generators skew to τ only depends on the parameters n and j and not on the choice of τ itself.

We will prove this theorem using induction on j. If $j = -1$, τ is the empty space and hence $c_{n,-1}$ equals the total number of generators. By Lemma 2.4.4 we find $c_{n,-1} = \prod_{k=0}^{n}(q^{2k+1}+1)$. Now, we prove that a relation between $c_{n,j}$ and $c_{n-1,j-1}$ holds.

By Lemma 2.4.6 we know $\tau \cap \tau^\sigma = \emptyset$. Hence, every point $P \in \mathrm{PG}(2n+1, q^2) \setminus (\tau \cup \tau^\sigma)$ can uniquely be written as $P_\tau + \lambda_P P_{\tau^\sigma}$, $P_\tau \in \tau$, $P_{\tau^\sigma} \in \tau^\sigma$, $\lambda_P \in \mathbb{F}_{q^2}^*$. For every point $P \in \mathrm{PG}(2n+1, q^2) \setminus (\tau \cup \tau^\sigma)$, we define $\phi_\tau(P) = P_\tau$. This is the projection of P from τ^σ on τ. We define a correlation $\overline{\sigma} : \tau \to \tau$ that maps the subspace $U \subset \tau$ to $U^\sigma \cap \tau$. It is straightforward to check that $\overline{\sigma}$ defines a polarity on τ. Moreover, it can be seen easily that the points of H_j are the absolute points of $\overline{\sigma}$. Hence, $\overline{\sigma}$ is the polarity of τ corresponding to H_j.

Now, we consider the set

$$S = \{(P, \mu) \mid P \in \mu \setminus \tau^\sigma, \phi_\tau(P) \notin H_j, \mu \text{ a generator}, \mu \cap \tau = \emptyset\}.$$

We count the number of elements of S in two ways. On the one hand, there are $c_{n,j}$ generators skew to τ. Let μ be such a generator. The intersection $\mu \cap \tau^\sigma$ is an $(n-j-1)$-space since $\dim(\mu \cap \tau^\sigma) + \dim(\langle \mu, \tau \rangle) = 2n$. We also know that $\phi_\tau(P) = R$ for every point $P \in \langle R, \mu \cap \tau^\sigma \rangle \setminus (\mu \cap \tau^\sigma)$, $R \in \tau$. Hence, for each generator there are

$$\theta_n(q^2) - \theta_{n-j-1}(q^2) - \mu_j(q^2)(\theta_{n-j}(q^2) - \theta_{n-j-1}(q^2)) = q^{2(n-j)}(\theta_j(q^2) - \mu_j(q^2))$$

points fulfilling the requirements. On the other hand, we count the points $P \in \mathcal{H}(2n+1, q^2) \setminus (\tau \cup \tau^\sigma)$ fulfilling the requirements. There are $\mu_{2n+1}(q^2) - \mu_j(q^2) - \mu_{2n-j}(q^2)$ points in this set. We must assure that $\phi_\tau(P) \notin H_j$. Let R be a point of H_j. Since $\tau^\sigma \subseteq R^\sigma$, a line RQ, $Q \in \tau^\sigma$, is a tangent line (in R) to $\mathcal{H}(2n+1, q^2)$ or a line which is completely contained in $\mathcal{H}(2n+1, q^2)$. Hence, $\phi_\tau(P)$ is a point of H_j iff P lies on a line through $\phi_\tau(P)$ and a point of $\tau^\sigma \cap \mathcal{H}(2n+1, q^2)$. Consequently there are

$$\begin{aligned} &\mu_{2n+1}(q^2) - \mu_j(q^2) - \mu_{2n-j}(q^2) - \mu_j(q^2)\mu_{2n-j}(q^2)(q^2-1) \\ &= q^{2n-j}(\theta_j(q^2) - \mu_j(q^2))(q^{2n-j+1} - (-1)^{2n-j+1}) \end{aligned}$$

points $P \in \mathcal{H}(2n+1, q^2) \setminus (\tau \cup \tau^\sigma)$ fulfilling the requirement $\phi_\tau(P) \notin H_j$. Now, we fix such a point P and we count the number of generators skew to τ, through it. All these generators are contained in P^σ. We know $P^\sigma \cap \mathcal{H}(2n+1, q^2)$ is a cone PH_{2n-1}, with $H_{2n-1} \cong \mathcal{H}(2n-1, q^2)$. There is a one-one correspondence between the generators of $\mathcal{H}(2n+1, q^2)$ through P and the generators of H_{2n-1}. We also find

$$\tau \cap P^\sigma = \tau \cap (\phi_\tau(P) + \lambda P')^\sigma = \tau \cap ((\phi_\tau(P))^\sigma + \lambda^q P'^\sigma) = \tau \cap (\phi_\tau(P))^\sigma = (\phi_\tau(P))^{\overline{\sigma}},$$

with $P' \in \tau^\sigma$ (and thus $\tau \subset P'^\sigma$). Hence, the $(j-1)$-space $\tau \cap P^\sigma$ intersects $\mathcal{H}(2n+1, q^2)$ in $H_{j-1} \cong \mathcal{H}(j-1, q^2)$, since $\phi_\tau(P) \notin H_j$. We can choose the base of the cone PH_{2n-1} such that it contains $\tau \cap P^\sigma$. The generators through P and skew to τ correspond to the generators of H_{2n-1}, skew to $\tau \cap P^\sigma$. There are $c_{n-1,j-1}$ such generators. We conclude

$$\begin{aligned} c_{n,j} q^{2(n-j)} \left[\theta_j(q^2) - \mu_j(q^2)\right] &= c_{n-1,j-1} q^{2n-j} \left[\theta_j(q^2) - \mu_j(q^2)\right] (q^{2n-j+1} - (-1)^{2n-j+1}) \\ \Rightarrow \qquad c_{n,j} &= c_{n-1,j-1} q^j (q^{2n-j+1} - (-1)^{2n-j+1}). \end{aligned}$$

An induction calculation now finishes the proof. □

From now on in this section, we use the following notation: $H \cong \mathcal{H}(2n+1, q^2)$ is a non-singular Hermitian variety and σ is the polarity corresponding to it; π is an n-space in $\mathrm{PG}(2n+1, q^2)$, such that $\pi \cap H$ is a cone $\pi_i H_{n-i-1}$ with $H_{n-i-1} \cong \mathcal{H}(n-i-1, q^2)$ and π_i an i-space, $-1 \leq i \leq n$. By Lemma 2.4.6, for $k = n$, we know $\pi \cap \pi^\sigma = \pi_i$ and consequently $\pi^\sigma \cap H$ is a cone $\pi_i H'_{n-i-1}$ with $H'_{n-i-1} \cong \mathcal{H}(n-i-1, q^2)$.

Definition 2.4.13. The number of generators on H intersecting π in a fixed point $P \in \pi_i H_{n-i-1} \setminus \pi_i$ and no other point of $\pi_i H_{n-i-1}$, and intersecting π^σ in a fixed point $P' \in \pi_i H'_{n-i-1} \setminus \pi_i$ and no other point of $\pi_i H'_{n-i-1}$ is denoted by $N(\pi, P, P')$. The number of generators on H skew to π is denoted by $N'(\pi)$.

By Lemma 2.4.8 we know that the generators skew to π are also skew to π^σ and that the generators intersecting π in precisely one point also intersect π^σ in precisely one point.

Lemma 2.4.14. *The number $N'(\pi)$ only depends on the intersection parameters (n,i) of π.*

Proof. This follows immediately from [60, Theorem 23.4.2 (i)]. □

Notation 2.4.15. Consequently, we can denote $N'(\pi)$ by $N'(n,i)$.

Lemma 2.4.16. *For $n \geq 2$, $-1 \leq i \leq n-2$, $N(\pi, P, P') = N'(n-2,i)$. Consequently, $N(\pi, P, P')$ only depends on the intersection parameters (n,i) of π.*

Proof. Consider the points $P \in (\pi_i H_{n-i-1} \setminus \pi_i) \subseteq \pi$ and $P' \in (\pi_i H'_{n-i-1} \setminus \pi_i) \subseteq \pi^\sigma$. Denote $\ell = \langle P, P' \rangle$. Then ℓ^σ is a $(2n-1)$-space intersecting H in a cone with ℓ as vertex and a non-singular $(2n-3)$-dimensional Hermitian variety H_{2n-3} as base. Since $\dim(\ell \cap \pi) = \dim(\ell \cap \pi^\sigma) = 0$, $\ell^\sigma \cap \pi = V$ is an $(n-1)$-space and $\ell^\sigma \cap \pi^\sigma = V'$ is an $(n-1)$-space. Also, $\ell \subset \langle \pi, \pi^\sigma \rangle = \pi_i^\sigma$, hence $\pi_i \subset l^\sigma$. Let W, resp. W', be an $(n-2)$-space in V, resp. V', containing π_i and not through P, resp. P'. Denote the $(2n-i-4)$-space $\langle W, W' \rangle$ by τ'. It can be seen that on the one hand $\tau' \subset \ell^\sigma$ and on the other hand $\ell \cap \tau' = \emptyset$, so the $(2n-3)$-space τ containing the base H_{2n-3} can be chosen such that $\tau' \subseteq \tau$. Let σ' be the polarity of τ corresponding to H_{2n-3}. Analogously to the proof of Lemma 2.4.12 we can define this polarity as follows: $U^{\sigma'} = \tau \cap U^\sigma$. It now immediately follows that $W^{\sigma'} = W'$ because both are $(n-2)$-spaces contained in W^σ and in τ.

Arguing as in the proof of Lemma 2.4.12, we see there is a one-one correspondence between the generators of H_{2n-3} and the generators of H through ℓ (the generators containing P and P'). If a generator of H through ℓ contains no points of $\pi \cup \pi^\sigma$ but P and P', then its corresponding generator of H_{2n-3} is skew to W and W'. Vice versa, every generator μ of H_{2n-3} skew to W and W', is contained in precisely one generator of H intersecting $\pi \cup \pi^\sigma$ in only the points P and P', namely $\langle \mu, P, P' \rangle$. Since $W^{\sigma'} = W'$, the generators of H_{2n-3} skew to W and W' are the ones skew to W, by Lemma 2.4.8. Hence, $N(\pi, P, P') = N'(n-2,i)$.

The second statement of the lemma follows immediately from the first one. □

Notation 2.4.17. Since $N(\pi, P, P')$ only depends on the intersection parameters (n,i) of π, we can denote it by $N(n,i)$.

The previous theorem now states $N(n,i) = N'(n-2,i)$ for $n \geq 2$, $-1 \leq i \leq n-2$.

Lemma 2.4.18. *For $n \geq 1$ and $-1 \leq i \leq n-2$, the following equality holds:*

$$N(n,i) = q^{(n-1)^2 - \binom{n-i-1}{2}} \prod_{j=1}^{n-i-2} (q^j - (-1)^j).$$

Proof. We prove this theorem by induction. Using Lemma 2.4.16, we know that $N(n,-1)$ equals $N'(n-2,-1)$, the number of generators of a Hermitian variety $H' \cong \mathcal{H}(2n-3, q^2)$ skew to an $(n-2)$-space intersecting H' in a Hermitian variety $\mathcal{H}(n-2, q^2)$, if $n \geq 2$. This number equals $c_{n-2,n-2}$. Hence, by Lemma 2.4.12,

$$N(n,-1) = q^{\binom{n-1}{2}} \prod_{l=1}^{n-1} (q^l - (-1)^l) = q^{(n-1)^2 - \binom{n-(-1)-1}{2}} \prod_{j=1}^{n-(-1)-2} (q^j - (-1)^j),$$

which proves the induction base for $n \geq 2$. If $n = 1$, it is easy to prove that $N(1, -1) = 1$. Hence, the formula holds also in this case.

Now, we will prove that $N(n, i) = q^{2n-3}N(n-1, i-1)$. By Lemma 2.4.16, this is equivalent to proving that $N'(n, i) = q^{2n+1}N'(n-1, i-1)$. Consider the set $S = \{(R, \mu) \mid R \in \mu, \mu$ a generator skew to $\pi, R \notin \langle \pi, \pi^\sigma \rangle = \pi_i^\sigma\}$. Clearly, π_i^σ intersects H in a cone $\pi_i H_{2(n-i)-1}$. We will count $|S|$ in two ways.

On the one hand, there are $N'(n, i)$ generators skew to π. Fix such a generator μ. Then $\dim(\mu \cap \pi_i^\sigma) = n - i - 1$ since $\dim(\mu \cap \pi_i^\sigma) + \dim(\langle \mu, \pi_i \rangle) = 2n$. So, μ contains precisely $\theta_n(q^2) - \theta_{n-i-1}(q^2) = q^{2(n-i)}\theta_i(q^2)$ points of $\mathrm{PG}(2n+1, q^2) \setminus \pi_i^\sigma$. Consequently, $|S| = q^{2(n-i)}\theta_i(q^2)N'(n, i)$.

On the other hand, there are $\mu_{2n+1}(q^2) - \theta_i(q^2) - \mu_{2(n-i)-1}(q^2) - (q^2-1)\theta_i(q^2)\mu_{2(n-i)-1}(q^2) = q^{4n-2i+1}\theta_i(q^2)$ points in $H \setminus \pi_i^\sigma$. Fix such a point P. The hyperplane P^σ intersects π in an $(n-1)$-space V and intersects π_i in an $(i-1)$-space $\pi_{i-1} \subset V$. Hence, the intersection $V \cap H$ has intersection parameters $(n-1, i-1)$. The intersection $P^\sigma \cap H$ is a cone PH_{2n-1}, with $H_{2n-1} \cong \mathcal{H}(2n-1, q^2)$. Let τ be the $(2n-1)$-space containing H_{2n-1}. We can choose τ such that it contains V. Then, there is a one-one correspondence between the generators of $\mathcal{H}(2n+1, q^2)$ through P, skew to π and the generators of H_{2n-1} skew to V. Consequently, there are $N'(n-1, i-1)$ such generators. Thus, $|S| = q^{4n-2i+1}\theta_i(q^2)N'(n-1, i-1)$.

Comparing both expressions for $|S|$, we find the desired relation between $N'(n, i)$ and $N'(n-1, i-1)$. An easy calculation now finishes the proof. □

Lemma 2.4.19. *Assume $n \geq 2$ and $-1 \leq i \leq n-2$. Let P be a point of $H \setminus (\pi \cup \pi^\sigma)$. Let $n_P(n, i)$ be the number of generators through P intersecting both $\pi \setminus \pi_i$ and $\pi^\sigma \setminus \pi_i$ in precisely one point. Then,*

- $n_P(n, i) = N(n-1, i-1)q^{4i}\left(\mu_{n-i-1}(q^2)\right)^2$ *if $P \notin \langle \pi, \pi^\sigma \rangle = \pi_i^\sigma$;*
- $n_P(n, i) = N(n-1, i)q^{4i+4}\left(\mu_{n-i-2}(q^2)\right)^2$ *if $P \in \langle \pi, \pi^\sigma \rangle = \pi_i^\sigma$ but P does not belong to a line of H through a point of $\pi \setminus \pi_i$ and a point of $\pi^\sigma \setminus \pi_i$;*
- $n_P(n, i) = N(n-1, i+1)q^{4i+4}\mu_{n-i-3}(q^2)\left[q^4\mu_{n-i-3}(q^2) + q^2 - 1\right]$ *if $P \in \langle \pi, \pi^\sigma \rangle = \pi_i^\sigma$ and P belongs to a line of H through a point of $\pi \setminus \pi_i$ and a point of $\pi^\sigma \setminus \pi_i$, and $i \leq n-4$.*
- $n_P(n, i) = q^{2i+2}N(n, i)$ *if $P \in \langle \pi, \pi^\sigma \rangle = \pi_i^\sigma$ and P belongs to a line of H through a point of $\pi \setminus \pi_i$ and a point of $\pi^\sigma \setminus \pi_i$, and $i = n-3, n-2$.*

The first case can only occur if $i \geq 0$. The second case can only occur if $i \leq n-3$.

Proof. Since $P \notin \pi \cup \pi^\sigma$, $P^\sigma \cap \pi = V$ is an $(n-1)$-space and $P^\sigma \cap \pi^\sigma = V'$ is an $(n-1)$-space. Furthermore $P^\sigma \cap H$ is a cone PH_{2n-1} with $H_{2n-1} \cong \mathcal{H}(2n-1, q^2)$. Let τ be the $(2n-1)$-space containing H_{2n-1}.

First we consider the case $P \notin \langle \pi, \pi^\sigma \rangle = \pi_i^\sigma$. In this case P^σ intersects π_i in an $(i-1)$-space $\pi_{i-1} = V \cap V'$. Also, τ can be chosen so that it contains V and V'. Hence, the

number of generators through P fulfilling the requirements equals the number of generators of H_{2n-1} intersecting V and V' in a point. Let σ' be the polarity of τ corresponding to H_{2n-1}. Analogously to the argument in the proof of Lemma 2.4.16, it can be seen that $V' = V^{\sigma'}$. Consequently there are $N(n-1, i-1)$ generators of this type through a fixed point of $V \setminus \pi_{i-1}$ and a fixed point of $V' \setminus \pi_{i-1}$. There are $q^{2i}\mu_{n-i-1}(q^2)$ possible choices for each of these points. The first part of the lemma follows. Note that $\langle \pi, \pi^\sigma \rangle = \mathrm{PG}(2n+1, q^2)$ if $i = -1$. Hence, this case cannot occur if $i = -1$.

We fix some notation for the remaining cases. Let $W \subseteq \pi$ and $W' \subseteq \pi^\sigma$ be the $(n-i-1)$-spaces containing H_{n-i-1} and H'_{n-i-1}, respectively. Furthermore, let $\overline{\sigma}$ and $\overline{\sigma}'$ be the polarities of W and W', respectively corresponding to H_{n-i-1} and H'_{n-i-1}. In all three remaining cases, $\pi_i \subset P^\sigma$, hence $P^\sigma \cap W = W_1$ and $P^\sigma \cap W' = W'_1$ are $(n-i-2)$-spaces. Now, the point P can be written in a unique way as $P = \lambda P_{\pi_i} + \lambda_P P_W + P_{W'}$, with $P_W \in W$, $P_{W'} \in W'$, $P_{\pi_i} \in \pi_i$ and $\lambda, \lambda_P \in \mathbb{F}_{q^2}$. Arguing as in the proof of Lemma 2.4.12 we can see that $W_1 = P^\sigma \cap W = P_W^{\overline{\sigma}}$ and that $W'_1 = P^\sigma \cap W' = P_{W'}^{\overline{\sigma}'}$. Moreover, since P and P_{π_i} are contained in P^σ, neither or both of P_W and $P_{W'}$ are contained in P^σ. Hence, we need to distinguish two cases.

- $P_W \in W_1$ and $P_{W'} \in W'_1$ are both contained in P^σ; consequently, $P_W \in P_W^{\overline{\sigma}}$, thus $P_W \in H_{n-i-1} \subset H$ and $P_W^{\overline{\sigma}} \cap H_{n-i-1}$ is a cone $P_W H_{n-i-3}$, with $H_{n-i-3} \cong \mathcal{H}(n-i-3, q^2)$. Let $W_2 \subset W_1$ be the $(n-i-3)$-space containing H_{n-i-3}. Then, the intersection of $V = \langle \pi_i, W_1 \rangle$ and H is the cone with vertex $\langle \pi_i, P_W \rangle$ and base H_{n-i-3}. Analogously we introduce $H'_{n-i-3} \subset W'_2 \subset W'_1$. Then $V' \cap H$ is the cone with vertex $\langle \pi_i, P_{W'} \rangle$ and base H'_{n-i-3}. Furthermore, since $P_W \in V$, $P_{W'} \in V'$, and $P_{\pi_i} \in V, V'$, P is contained in $\langle V, V' \rangle$. Also, the line PP_W is contained in P^σ and is not a 1-secant since $P, P_W \in H$, hence it is a line of H. This line intersects π^σ in a point of $\langle P_{W'}, \pi_i \rangle \setminus \pi_i$.

- $P_W \notin W_1$ and $P_{W'} \notin W'_1$ are both not contained in P^σ; consequently, $P_W \notin P_W^{\overline{\sigma}}$, thus $P_W \notin H_{n-i-1}$, $P_W \notin H$ and $P_W^{\overline{\sigma}} \cap H_{n-i-1}$ is a non-singular Hermitian variety $H_{n-i-2} \cong \mathcal{H}(n-i-2, q^2)$ in W_1. Then, the intersection of $V = \langle \pi_i, W_1 \rangle$ and H is the cone $\pi_i H_{n-2-i}$. Analogously we introduce $H'_{n-i-2} \subset W'_1$. The intersection $V' \cap H$ is the cone $\pi_i H'_{n-i-2}$. Furthermore, $P \notin \langle V, V' \rangle$ since $P_W \notin W_1$ and $P_{W'} \notin W'_1$. Also, all lines in π_i^σ through P intersecting $\pi \setminus \pi_i$ and $\pi^\sigma \setminus \pi_i$, are contained in $\langle P_W, P_{W'}, \pi_i \rangle$, but not in $\langle P, \pi_i \rangle$. Since $P_W, P_{W'} \notin P^\sigma$, none of the lines through P can be contained in H.

These two cases clearly correspond to the three remaining cases of the lemma. We will treat them separately.

First of all, we look at the latter, which is the second case in the statement of the lemma. Since $P \notin \langle V, V' \rangle$, we can choose τ such that it contains $\langle V, V' \rangle$. Hence, every generator through P, intersecting both $\pi \setminus \pi_i$ and $\pi^\sigma \setminus \pi_i$ in a point, corresponds to a generator of H_{2n-1} intersecting both $V \setminus \pi_i$ and $V' \setminus \pi_i$ in a point, and vice versa. For a fixed point in $V \setminus \pi_i$ and a fixed point in $V' \setminus \pi_i$, there are $N(n-1, i)$ such generators. We also know that $|V \setminus \pi_i| = |V' \setminus \pi_i| = q^{2i+2}\mu_{n-i-2}(q^2)$. The second part of the lemma follows. Note that $V \setminus \pi_i$ and $V' \setminus \pi_i$ are empty if $i = n-2$. Hence, this case only occurs if $i \leq n-3$.

Finally, we look at the former case, the third and the fourth case in the statement of the lemma. Let ℓ be a line on H through P, a point of $\pi \setminus \pi_i$ and a point of $\pi^\sigma \setminus \pi_i$. By

changing, if necessary, the choices for W and W', we can assume $\ell = P_W P_{W'}$. We distinguish between two types of generators: the ones that contain ℓ and the ones that do not contain ℓ. First we look at the ones that contain ℓ. We know $\ell^\sigma \cap H$ is a cone with vertex ℓ and base $H_{2n-3} \cong \mathcal{H}(2n-3, q^2)$. Let τ' be the $(2n-3)$-space containing H_{2n-3}. We can choose τ' so that it contains π_i, W_2 and W_2'. As before, one can see that $\langle \pi_i, W_2 \rangle^{\widehat{\sigma}'} = \langle \pi_i, W_2' \rangle$, with $\widehat{\sigma}'$ the polarity of τ' corresponding to H_{2n-3}. The number of generators of the requested type through ℓ then equals the number of generators of H_{2n-3} skew to $\langle \pi_i, W_2 \rangle$. This number equals $N'(n-2, i) = N(n, i)$. Furthermore, since ℓ is a line on H through P intersecting $\pi \setminus \pi_i$ and $\pi^\sigma \setminus \pi_i$, every line through P and a point of $\langle P_W, \pi_i \rangle \setminus \pi_i$ belongs to H and intersects $\langle P_{W'}, \pi_i \rangle \setminus \pi_i \subset \pi^\sigma \setminus \pi_i$. Thus, there are $\theta_{i+1}(q^2) - \theta_i(q^2) = q^{2i+2}$ such lines. Hence, there are $q^{2i+2}N(n, i)$ generators of the first type. Now, we assume no line through P, intersecting π and π^σ, is contained in the generator. Let Q_W and $Q_{W'}$ be the points of the generator in W and W', respectively. By the previous remarks on this case, we know there are $\mu_{n-i-3}(q^2)q^{2i+4}$ possible choices for Q_W and for $Q_{W'}$. Now, we consider the plane $\langle P, Q_W, Q_{W'} \rangle$. Using arguments, similar to the ones in the previous case, we find $N'(n-3, i+1) = N(n-1, i+1)$ generators fulfilling the requirements for every choice of Q_W and $Q_{W'}$. Hence, the total number of generators in this third case equals

$$\begin{aligned} n_P &= q^{2i+2}N(n,i) + \left(\mu_{n-i-3}(q^2)q^{2i+4}\right)^2 N(n-1, i+1) \\ &= \left[q^{2i+2}q^{2i+2}(q^2-1)\mu_{n-i-3}(q^2) + \left(\mu_{n-i-3}(q^2)q^{2i+4}\right)^2\right] N(n-1, i+1) \\ &= q^{4i+4}\mu_{n-i-3}(q^2)\left[q^2 - 1 + q^4\mu_{n-i-3}(q^2)\right] N(n-1, i+1)\,. \end{aligned}$$

Hereby we used the relation between $N(n, i)$ and $N(n-1, i+1)$ which can immediately be derived from Lemma 2.4.18.

Note that $V \setminus \langle \pi_i, P_W \rangle$ and $V' \setminus \langle \pi_i, P_{W'} \rangle$ are empty if $n - 3 \leq i \leq n - 2$. In this case, we cannot consider the points Q_W and $Q_{W'}$. So, there are no generators of the second type. Consequently, all generators are of the first type and there are precisely $q^{2i+2}N(n, i)$ such generators. □

2.4.3 Classifying the small weight code words

Before stating the new classification theorem, we will first state the results about the codes $C_1(\mathcal{H}(3, q^2))^\perp$ and $C_2(\mathcal{H}(5, q^2))^\perp$ to which we referred earlier.

Theorem 2.4.20 ([78, Proposition 3.7])**.** *Let C be the code $C_1(\mathcal{H}(3, q^2))^\perp$. There is only one non-trivial type of code words among the ones described in Example 2.4.10, namely $i = -1$. These are the code words of minimal weight. Let c be a code word of C with $\mathrm{wt}(c) \leq \frac{\sqrt{q}(q+1)}{2}$. Then c is a linear combination of code words of minimal weight.*

Theorem 2.4.21 ([114, Theorem 43])**.** *Let c be a code word of $C_2(\mathcal{H}(5, q^2))^\perp$, $q > 893$, with $\mathrm{wt}(c) \leq 2(q^3 + q)$, then c is a code word of one of the types described in Theorem 2.4.9. Regarding Example 2.4.10, we know that there are precisely two possibilities since $n = 2$, namely $i = -1$ and $i = 0$.*

It is our aim to generalise this result. We start our arguments with two lemmas about n-spaces: the second lemma shows the existence of an n-space containing a non-trivial amount of points of the support of a code word, while the first lemma shows that a generator cannot contain many points of the support of a code word. In the proof of the second lemma we use the following result.

Theorem 2.4.22. *Let $c \in C_n(\mathcal{H}(2n+1,q^2))^\perp$ be a code word and denote $\mathrm{supp}(c) = S$. Let P be a point in S. Then $|P^\sigma \cap S| \geq 2 + q^{2n-1}$.*

Proof. This is a special case of [114, Proposition 9(d)]. □

Throughout the three following lemmas the value $\Sigma_{n,i}$ is used, $-1 \leq i \leq n-2$. It is defined by

$$\Sigma_{n,i} = \begin{cases} 2q^{2i+2}\mu_{n-i-1}(q^2) + 4\frac{\mu_{n-i-2}(q^2)(q^{n-i-1}-1)}{q^{n-3i-5}(q^2-1)} & n-i \text{ odd}, \\ 2q^{2i+2}\left[\mu_{n-i-1}(q^2) + 2\frac{q^4\mu_{n-i-3}(q^2)+q^2-1}{q^2-1}\right] & n-i \text{ even}. \end{cases}.$$

Note that in both cases $\Sigma_{n,i} = 2q^{2n-1} + f$, with $f \in \mathcal{O}(q^{2n-2})$ and $f > 0$ if $q > 0$.

Lemma 2.4.23. *Let $c \in C_n(\mathcal{H}(2n+1,q^2))^\perp$ be a code word with $\mathrm{wt}(c) \leq w = \delta q^{2n-1}$, and denote $\mathrm{supp}(c) = S$. Let π be a generator of $\mathcal{H}(2n+1,q^2)$. Then $|\pi \cap S| \leq \delta\theta_{n-1}(q^2)$.*

Proof. The proof is a generalisation of the proof of [114, Lemma 41].

Denote $x = |\pi \cap S|$ and let P be a point in $\pi \cap S$. Then $P^\sigma \cap \mathcal{H}(2n+1,q^2)$ is a cone with vertex P. Let $H' \cong \mathcal{H}(2n-1,q^2)$ be a base of this vertex and consider the projection from P onto H'. Denote the projection of $S \cap P^\sigma$ by S'. The projection of π is a generator π' of H'. Note that S' is a blocking set of the generators on H'.

By [79, Lemma 10], we know there are q^{n^2} generators in H' that are skew to π', of which $q^{(n-1)^2}$ pass through a fixed point of $H' \setminus \pi'$. Hence, the blocking set S' contains at least q^{2n-1} points not in π'. Counting the tuples (P,Q), $P \in \pi \cap S$, $Q \in S \setminus \pi$, with $PQ \subset \mathcal{H}(2n+1,q^2)$, in two ways we find

$$xq^{2n-1} \leq \delta q^{2n-1}\theta_{n-1}(q^2),$$

where the upper bound follows from the fact that every point $Q \in S \setminus \pi$ is collinear with the points of an $(n-1)$-space in π and not with the other points in π. The theorem follows immediately. □

Note that the size of a blocking set on a Hermitian variety $\mathcal{H}(2n+1,q^2)$ is at least the size of an ovoid, hence at least $q^{2n+1}+1$.

Recall that the symmetric difference $A\Delta B$ of two sets A and B is the set $(A \cup B) \setminus (A \cap B)$.

Lemma 2.4.24. *Let p be a fixed prime and denote $q = p^h$, $h \in \mathbb{N}$. Let $c \in C_n(\mathcal{H}(2n+1,q^2))^\perp$ be a code word with $\mathrm{wt}(c) \leq w = \delta q^{2n-1}$, $\delta > 0$ a constant, and denote $\mathrm{supp}(c) = S$. Denote $\mathcal{H}(2n+1,q^2)$ by H and let σ be the polarity related to H. Then a constant $C_n > 0$, a value $Q > 0$ and an n-space π can be found such that $|(\pi\Delta\pi^\sigma) \cap S| > C_n q^{2n-1}$ and such that*

$\frac{p-1}{p}|(\pi\Delta\pi^\sigma)\cap H| < \Sigma_{n,i} - C_n q^{2n-1}$, *if* $q \geq Q$. *Hereby,* i *is such that* $\pi \cap H$ *is a cone with an* i*-dimensional vertex and* $i \leq n-2$.

Proof. We introduce the notion of a *semi-arc*. A semi-arc $\mathcal{A}$ is a set of $k \geq n$ points in $\mathrm{PG}(2n+1, q^2)$ such that no $n+1$ points of $\mathcal{A}$ are contained in an $(n-1)$-space. We make two remarks about these semi-arcs. First, if $|S| > \binom{k}{n}\theta_{n-1}(q^2)$, then S contains a semi-arc with $k+1$ points, since it is possible to construct the semi-arc point by point: we start with a set of n linearly independent points in S and we extend the semi-arc point by point until we have $k+1$ points, which is possible by the condition on S. Secondly, if we choose K points $\{P_1, \ldots, P_K\}$ in a semi-arc $\mathcal{A} \subseteq S$, then

$$\sum_{\{i\}\in S_{K,1}} |P_i^\sigma \cap S| - \sum_{\{i,j\}\in S_{K,2}} |P_i^\sigma \cap P_j^\sigma \cap S| + \ldots + \sum_{\{i_1,\ldots,i_{2l+1}\}\in S_{K,2l+1}} |P_{i_1}^\sigma \cap P_{i_2}^\sigma \cap \cdots \cap P_{i_{2l+1}}^\sigma \cap S| \geq |(P_1^\sigma \cup P_2^\sigma \cup \cdots \cup P_K^\sigma)\cap S|, \quad (2.1)$$

since every point of $(P_1^\sigma \cup P_2^\sigma \cup \cdots \cup P_K^\sigma)\cap S$ is counted at least once on the left hand side. Also

$$\sum_{\{i\}\in S_{K,1}} |P_i^\sigma \cap S| - \sum_{\{i,j\}\in S_{K,2}} |P_i^\sigma \cap P_j^\sigma \cap S| + \ldots - \sum_{\{i_1,\ldots,i_{2l}\}\in S_{K,2l}} |P_{i_1}^\sigma \cap P_{i_2}^\sigma \cap \cdots \cap P_{i_{2l}}^\sigma \cap S| \leq |(P_1^\sigma \cup P_2^\sigma \cup \cdots \cup P_K^\sigma)\cap S|, \quad (2.2)$$

since every point of $(P_1^\sigma \cup P_2^\sigma \cup \cdots \cup P_K^\sigma)\cap S$ is counted at most once on the left hand side. In both expressions we denoted the set of all subsets of $\{1, \ldots, K\}$ of size j by $S_{K,j}$.

Now, we prove using induction, for every $0 \leq t \leq n$, that we can find for any $(t+1)$-tuple $(c_0, \ldots, c_t)$, $c_j > 0$ a constant (independent of q), a constant $K_t \in \mathbb{N}$ such that

$$\forall K \geq K_t, \forall \{P_1, \ldots, P_K\} \subseteq \mathcal{A} \subseteq S : \sum_{\{i_0,\ldots,i_t\}\in S_{K,t+1}} |P_{i_0}^\sigma \cap P_{i_1}^\sigma \cap \cdots \cap P_{i_t}^\sigma \cap S| \geq c_t q^{2n-1}.$$

We consider the case $t = 0$, the induction base. Let $\{P_1, \ldots, P_K\}$ be a set of points in $\mathcal{A} \subseteq S$ (without restriction on K). By Theorem 2.4.22, we know

$$\sum_{i=1}^{K} |P_i^\sigma \cap S| \geq K q^{2n-1}.$$

Hence, it is sufficient to choose $K_0 = \lceil c_0 \rceil$.

Next, we prove the induction step. We distinguish between two cases: t even and t odd. We look at the former, so we assume the inequality to be proven for $t \leq 2l-1$ and we prove it for $t = 2l$. Let K_m be the constant arising from the $(m+1)$-tuple $(c_0, \ldots, c_m)$, $m < 2l$, and

let $\{P_1, \ldots, P_K\}$ be a set of points in $\mathcal{A} \subseteq S$ with $K \geq K_{2l-1}$. By (2.1), we know that

$$\sum_{\{i\} \in S_{K,1}} |P_i^\sigma \cap S| - \sum_{\{i,j\} \in S_{K,2}} |P_i^\sigma \cap P_j^\sigma \cap S| + \ldots + \sum_{\{i_0,\ldots,i_{2l}\} \in S_{K,2l+1}} |P_{i_0}^\sigma \cap P_{i_1}^\sigma \cap \cdots \cap P_{i_{2l}}^\sigma \cap S| \geq |(P_1^\sigma \cup P_2^\sigma \cup \cdots \cup P_K^\sigma) \cap S|.$$

Using the induction hypothesis and Theorem 2.4.22, we find

$$\begin{aligned}\sum_{\{i_0,\ldots,i_{2l}\} \in S_{K,2l+1}} |P_{i_0}^\sigma \cap P_{i_1}^\sigma \cap \cdots \cap P_{i_{2l}}^\sigma \cap S| \geq{}& \frac{\binom{K}{K_{2l-1}}}{\binom{K-2l}{K_{2l-1}-2l}} c_{2l-1} q^{2n-1} + \frac{\binom{K}{K_{2l-3}}}{\binom{K-2l+2}{K_{2l-3}-2l+2}} c_{2l-3} q^{2n-1} \\ &+ \cdots + \frac{\binom{K}{K_1}}{\binom{K-2}{K_1-2}} c_1 q^{2n-1} \\ &- \left[\binom{K}{2l-1} + \binom{K}{2l-3} + \cdots + K\right] \delta q^{2n-1} \\ &+ q^{2n-1}\end{aligned}$$

and thus

$$\begin{aligned}&\sum_{\{i_0,\ldots,i_{2l}\} \in S_{K,2l+1}} |P_{i_0}^\sigma \cap P_{i_1}^\sigma \cap \cdots \cap P_{i_{2l}}^\sigma \cap S| \\ &\quad\geq \frac{\binom{K}{2l}}{\binom{K_{2l-1}}{2l}} c_{2l-1} q^{2n-1} + \frac{\binom{K}{2l-2}}{\binom{K_{2l-3}}{2l-2}} c_{2l-3} q^{2n-1} + \cdots + \frac{\binom{K}{2}}{\binom{K_1}{2}} c_1 q^{2n-1} \\ &\qquad - \left[\binom{K}{2l-1} + \binom{K}{2l-3} + \cdots + K\right] \delta q^{2n-1} + q^{2n-1} \\ &\qquad = q^{2n-1} f(K, \delta, l, K_1, K_3, \ldots, K_{2l-1}, c_1, c_3, \ldots, c_{2l-1}).\end{aligned}$$

Note that $\frac{\binom{K}{K_{2i-1}}}{\binom{K-2i}{K_{2i-1}-2i}} = \frac{\binom{K}{2i}}{\binom{K_{2i-1}}{2i}}$. We now study the function f, which is clearly independent of q. Considering f as a function of K and comparing the exponents, we see that the term $\frac{\binom{K}{2l}}{\binom{K_{2l-1}}{2l}} c_{2l-1}$ dominates the others. Hence, we can find a value $K_{2l} \geq K_{2l-1}$ such that the right hand side is at least $c_{2l} q^{2n-1}$ for all $K \geq K_{2l}$, with c_{2l} as chosen above. Then the statement follows. Note that K_{2l} depends on the parameters $l, c_1, \ldots, c_{2l}$ chosen before (the values K_i, $0 \leq i < 2l$, depend themselves on $i, c_1, \ldots, c_i$).

For the latter case, t odd, the argument is similar, in this case starting from (2.2).

We will now apply the previous result for $t = n$. In order to do this, we need a semi-arc containing at least K_n points. We argued in the beginning of the proof that $\delta q^{2n-1} = |S| > \binom{K_n - 1}{n} \theta_{n-1}(q^2)$ is a sufficient condition. Since K_n is a constant, independent of q, and $\theta_{n-1}(q^2) = q^{2n-2} + q^{2n-4} + \cdots + q^2 + 1$, we can find $Q_1' > 0$ such that this inequality is true for all $q \geq Q_1'$. Then we know

$$\sum_{\{i_0,\ldots,i_n\} \in S_{K_n,n+1}} |P_{i_0}^\sigma \cap P_{i_1}^\sigma \cap \cdots \cap P_{i_n}^\sigma \cap S| \geq c_n q^{2n-1}$$

for the points $\{P_1, P_2, \ldots, P_{K_n}\}$ defining a semi-arc in S. Hence, we can find $n+1$ points - without loss of generality the points $\{P_1, \ldots, P_{n+1}\}$ - such that

$$|P_1^\sigma \cap P_2^\sigma \cap \cdots \cap P_{n+1}^\sigma \cap S| \geq \frac{c_n}{\binom{K_n}{n+1}} q^{2n-1}.$$

We can find a constant $\overline{K} > 0$ and a value $Q' \geq Q'_1$ such that $\frac{c_n}{\binom{K_n}{n+1}} q^{2n-1} \geq \overline{K} q^{2n-1} + \theta_{n-2}(q^2)$ for $q \geq Q'$. We write $C_n = \overline{K} - \epsilon$, $\max\{0, \overline{K} - \frac{2}{p}\} < \epsilon < \overline{K}$, and we denote the n-space $P_1^\sigma \cap P_2^\sigma \cap \cdots \cap P_{n+1}^\sigma$ by π. Note that π is an n-space since the points $P_1, P_2, \ldots, P_{n+1}$ belong to a semi-arc. Then $|\pi \cap S| > C_n q^{2n-1} + \theta_{n-2}(q^2)$.

We know the intersection $\pi \cap H$ can be written as $\pi_i H_{n-i-1}$, with $H_{n-i-1} \cong \mathcal{H}(n-i-1, q^2)$ and π_i an i-space, $-1 \leq i \leq n$. Let $Q'' \geq Q'$ be such that $C_n q^{2n-1} + \theta_{n-2}(q^2) > \delta\theta_{n-1}(q^2)$ for all $q \geq Q''$. Such a value exists since the first term on the left hand side dominates the right hand side. If $i \geq n-1$, then $\pi \cap H$ is contained in a generator of H. Thus, using Lemma 2.4.23 and the assumption $q \geq Q''$ we find a contradiction. Hence, $i \leq n-2$. We find:

$$|(\pi \Delta \pi^\sigma) \cap S| \geq |(\pi \setminus \pi_i) \cap S| \geq C_n q^{2n-1} + \theta_{n-2}(q^2) - \theta_i(q^2) \geq C_n q^{2n-1}\ .$$

We still need to check the second claim in the statement of the lemma: $\frac{p-1}{p}|(\pi \Delta \pi^\sigma) \cap H| < \Sigma_{n,i} - C_n q^{2n-1}$. Looking at the terms of highest degree in $\Sigma_{n,i} - C_n q^{2n-1} - \frac{p-1}{p}|(\pi \Delta \pi^\sigma) \cap H|$, we find $2 - C_n - 2\frac{p-1}{p} = \epsilon - \frac{c_n}{\binom{K_n}{n+1}} + \frac{2}{p} > 0$. Hence, we can find $Q \geq Q''$ such that the inequality $\frac{p-1}{p}|(\pi \Delta \pi^\sigma) \cap H| < \Sigma_{n,i} - C_n q^{2n-1}$ holds for all $q \geq Q$. □

In this proof $\frac{c_n}{\binom{K_n}{n+1}}$ depends also on the choice of $c_0, \ldots, c_{n-1}$. So, investigating the possible values for $c_0, \ldots, c_n$, we can find many different values for C_n. With each of these values, a value Q corresponds. We pick one of the possible values for C_n. By investigating different possibilities for C_n, we can see there is a trade-off between the choice of C_n and the corresponding value Q.

From now on, we consider C_n and the corresponding value Q to be fixed.

Lemma 2.4.25. *Let $c \in C_n(\mathcal{H}(2n+1, q^2))^\perp$ be a code word with $\mathrm{wt}(c) \leq w = \delta q^{2n-1}$, $\delta > 0$ a constant, and denote $\mathrm{supp}(c) = S$. Consider $H \cong \mathcal{H}(2n+1, q^2)$. Let π be an n-space such that $\pi \cap H$ is a cone $\pi_i H_{n-i-1}$ with $H_{n-i-1} \cong \mathcal{H}(n-i-1, q^2)$. Assume that $|S \cap (\pi \setminus \pi_i)| = x$ and $|S \cap (\pi^\sigma \setminus \pi_i)| = t$. Then there exists a value $Q_{n,i} \geq 0$ such that $x + t \leq C_n q^{2n-1}$ or $x + t \geq \Sigma_{n,i} - C_n q^{2n-1}$ if $q \geq Q_{n,i}$.*

Proof. Let P be a point of $S \cap (\pi \setminus \pi_i)$ and let P' be a point of $((\pi^\sigma \cap H) \setminus \pi_i) \setminus S$ and denote $\ell = PP'$. By Lemma 2.4.18 we know the number $N(n,i)$ of generators through ℓ intersecting π and π^σ in precisely one point, namely P and P'. Each of these generators contains an additional point of S. Let R be a point of $H \setminus (\pi \cup \pi^\sigma)$. By Lemma 2.4.19 we know the number $n_R(n,i)$ of generators through R intersecting both π and π^σ in a point. Hence, $S \setminus (\pi \cup \pi^\sigma)$ contains at least

$$x(|(\pi^\sigma \cap H) \setminus \pi_i| - t)\frac{N(n,i)}{n_{\max}(n,i)} = x(q^{2i+2}\mu_{n-i-1}(q^2) - t)\frac{N(n,i)}{n_{\max}(n,i)}$$

points, whereby $n_{\max}(n,i) = \max_{R \in S \setminus (\pi \cup \pi^\sigma)} n_R(n,i)$. Switching the roles of π and π^σ, and adding these two inequalities, we find after dividing by two

$$x(q^{2i+2}\mu_{n-i-1}(q^2) - t)\frac{N(n,i)}{2n_{\max}(n,i)} + t(q^{2i+2}\mu_{n-i-1}(q^2) - x)\frac{N(n,i)}{2n_{\max}(n,i)} + x + t \leq |S| \leq w \; .$$

Rewriting this inequality yields

$$(x+t)\left(q^{2i+2}\mu_{n-i-1}(q^2)N(n,i) + 2n_{\max}(n,i)\right) - 2xtN(n,i) \leq 2w\, n_{\max}(n,i) \; .$$

Using the inequality $2xt \leq \frac{1}{2}(x+t)^2$ and writing $y = x + t$, we find

$$\frac{1}{2}y^2 N(n,i) - \left[q^{2i+2}\mu_{n-i-1}(q^2)N(n,i) + 2n_{\max}(n,i)\right] y + 2w\, n_{\max}(n,i) \geq 0 \; .$$

We now distinguish between two cases: $n-i$ odd and $n-i$ even. First we look at the former. By detailed analysis one can see that in this case

$$\begin{aligned} N(n-1,i)q^{4i+4}\left(\mu_{n-i-2}(q^2)\right)^2 & \\ &\geq N(n-1,i-1)q^{4i}\left(\mu_{n-i-1}(q^2)\right)^2 \\ &\geq N(n-1,i+1)q^{4i+4}\mu_{n-i-3}(q^2)\left[q^4\mu_{n-i-3}(q^2) + q^2 - 1\right] \end{aligned}$$

if $n - i > 3$ and

$$N(n-1,n-3)q^{4n-8}\left(q+1\right)^2 \geq N(n-1,n-4)q^{4n-12}\left(q^3+1\right)^2 \geq N(n,n-3)q^{2n-4} \, .$$

These inequalities correspond to $i = n-3$. Hence,

$$n_{\max}(n,i) = N(n-1,i)q^{4i+4}\left(\mu_{n-i-2}(q^2)\right)^2 \, .$$

Using the formula for $N(n,i)$ from Lemma 2.4.18, and simplifying, we can rewrite this inequality as

$$\begin{aligned} \frac{1}{2}q^{n-3i-5}y^2 - \left[q^{n-i-3}\mu_{n-i-1}(q^2) + 2\mu_{n-i-2}(q^2)\frac{q^{n-i-1}-1}{q^2-1}\right] y & \\ + 2\delta q^{2n-1}\mu_{n-i-2}(q^2)\frac{q^{n-i-1}-1}{q^2-1} \geq 0 \; . & \quad (2.3) \end{aligned}$$

Let $\alpha_{n,i}(q^2)$ and $\alpha'_{n,i}(q^2)$ be the two solutions of the corresponding equation, with $\alpha_{n,i}(q^2) \leq \alpha'_{n,i}(q^2)$. Then $x + t \leq \alpha_{n,i}(q^2)$ or $x + t \geq \alpha'_{n,i}(q^2)$. Moreover,

$$\alpha_{n,i}(q^2) + \alpha'_{n,i}(q^2) = 2q^{2i+2}\mu_{n-i-1}(q^2) + 4\frac{\mu_{n-i-2}(q^2)(q^{n-i-1}-1)}{q^{n-3i-5}(q^2-1)} = \Sigma_{n,i} \; .$$

For the given δ we calculate

$$\overline{\alpha_{n,i}} = \lim_{q\to\infty} \alpha_{n,i}(q^2) = \lim_{q\to\infty} \frac{B' - \sqrt{B'^2 - 4\delta q^{3n-3i-6}C'}}{q^{n-3i-5}} \; ,$$

with

$$B' = q^{n-i-3}\mu_{n-i-1}(q^2) + 2\mu_{n-i-2}(q^2)\frac{q^{n-i-1}-1}{q^2-1}\,,$$
$$C' = \mu_{n-i-2}(q^2)\frac{q^{n-i-1}-1}{q^2-1}\,.$$

Since $\overline{\alpha_{n,i}} \in O(q^{2n-2})$, we can find $Q_{n,i} > 0$ such that $\alpha_{n,i}(q^2) \leq C_n q^{2n-1}$ for $q \geq Q_{n,i}$.

In the latter case, $n-i$ even, similar arguments can be used. However, in this case we need to distinguish between $n-i > 2$ and $i = n-2$. First, we discuss $n-i > 2$. We can deduce that

$$\begin{aligned} N(n-1,i)q^{4i+4}\left(\mu_{n-i-2}(q^2)\right)^2 & \\ &\leq N(n-1,i-1)q^{4i}\left(\mu_{n-i-1}(q^2)\right)^2 \\ &\leq N(n-1,i+1)q^{4i+4}\mu_{n-i-3}(q^2)\left[q^4\mu_{n-i-3}(q^2)+q^2-1\right]\,, \end{aligned}$$

hence $n_{\max}(n,i) = N(n-1,i+1)q^{4i+4}\mu_{n-i-3}(q^2)\left[q^4\mu_{n-i-3}(q^2)+q^2-1\right]$. We find the inequality

$$\begin{aligned} \frac{q^2-1}{2}y^2 - q^{2i+2}\left[\mu_{n-i-1}(q^2)(q^2-1) + 2(q^4\mu_{n-i-3}(q^2)+q^2-1)\right]y & \\ + 2\delta q^{2n-1}q^{2i+2}(q^4\mu_{n-i-3}(q^2)+q^2-1) \geq 0\,. & \quad (2.4) \end{aligned}$$

Just as in the previous case, we define $\Sigma_{n,i}$, which is the sum of the solutions of the corresponding equation, and $\overline{\alpha_{n,i}}$:

$$\Sigma_{n,i} = 2q^{2i+2}\left[\mu_{n-i-1}(q^2) + 2\frac{q^4\mu_{n-i-3}(q^2)+q^2-1}{q^2-1}\right]\,,$$
$$\overline{\alpha_{n,i}} = \lim_{q\to\infty}\frac{B'' - \sqrt{B''^2 - 4\delta q^{2n-1}(q^2-1)C''}}{q^2-1}\,,$$

with

$$B'' = q^{2i+2}\left[\mu_{n-i-1}(q^2)(q^2-1) + 2(q^4\mu_{n-i-3}(q^2)+q^2-1)\right]\,,$$
$$C'' = q^{2i+2}(q^4\mu_{n-i-3}(q^2)+q^2-1)\,.$$

Since $\overline{\alpha_{n,i}} \in O(q^{2n-2})$ also holds in this case, we again can find $Q_{n,i} > 0$ such that $\alpha_{n,i}(q^2) \leq C_n q^{2n-1}$ for $q \geq Q_{n,i}$.

Finally, we consider the case $i = n-2$. The second possibility in Lemma 2.4.19 can thus not occur. We note that

$$N(n-1,n-3)q^{4(n-2)}(q+1)^2 \leq q^{2n-2}N(n,n-2)\,.$$

The arguments in this case are analogous.

Hence, in all cases we can find $Q_{n,i} > 0$ such that $x+t \leq C_n q^{2n-1}$ or $x+t \geq \Sigma_{n,i} - C_n q^{2n-1}$ for $q \geq Q_{n,i}$. □

Using the three previous lemmas, we can now prove a classification theorem for the small weight code words in $C_n(\mathcal{H}(2n+1,q^2))^\perp$.

Theorem 2.4.26. *Let p be a fixed prime, $\delta > 0$ be a fixed constant and n be a fixed positive integer. Then there is a constant $\overline{Q}$ such that, for any $q = p^h$ with $h \in \mathbb{N}$ and $q \geq \overline{Q}$, and any $c \in C_n(\mathcal{H}(2n+1,q^2))^\perp$ with $\mathrm{wt}(c) \leq w = \delta q^{2n-1}$, c is a linear combination of code words described in Theorem 2.4.9.*

Proof. For the given values p and δ we have found a set of possible C_n-values, of which we have chosen one, in Lemma 2.4.24, with Q, a power of p, corresponding to it. By the proof of this lemma, we know that $C_n q^{2n-1} > \delta\theta_{n-1}(q^2)$ for all $q \geq Q$. Define $\overline{Q} = \max(\{Q\} \cup \{Q_{n,i} \mid -1 \leq i \leq n-2\})$, with $Q_{n,i}$ as in Lemma 2.4.25, corresponding to the chosen value C_n. We assume $q \geq \overline{Q}$.

Denote $\mathrm{supp}(c) = S$. By Lemma 2.4.24, we find an n-space π such that $N := |(\pi\Delta\pi^\sigma) \cap S| > C_n q^{2n-1}$. The intersection $\pi \cap H$ can be written as $\pi_i H_{n-i-1}$, with $H_{n-i-1} \cong \mathcal{H}(n-i-1,q^2)$, $-1 \leq i \leq n-2$.

Since $N > C_n q^{2n-1}$ and $q \geq Q_{n,i}$, we know by Lemma 2.4.25 that $N \geq \Sigma_{n,i} - C_n q^{2n-1}$. For each element $\alpha \in \mathbb{F}_p^*$, we denote by N_α the sum of the number of points $P \in \pi$ such that $c_P = \alpha$ and the number of points $Q \in \pi^\sigma$ such that $c_Q = -\alpha$. We can find $\beta \in \mathbb{F}_p^*$ such that $N_\beta \geq \frac{N}{p-1}$. We now consider the code word $c' = c - \beta(v_\pi - v_{\pi^\sigma})$, with v_π and v_{π^σ} as in Theorem 2.4.9. We know

$$\mathrm{wt}(c') = (N - N_\beta) + (|(\pi\Delta\pi^\sigma) \cap H| - N) = |(\pi\Delta\pi^\sigma) \cap H| - N_\beta \leq |(\pi\Delta\pi^\sigma) \cap H| - \frac{N}{p-1} .$$

We also know that $N \geq \Sigma_{n,i} - C_n q^{2n-1} > \frac{p-1}{p}|(\pi\Delta\pi^\sigma) \cap H|$ by Lemma 2.4.24. It follows that

$$\mathrm{wt}(c') < \frac{p}{p-1}N - \frac{N}{p-1} = N \leq \mathrm{wt}(c) .$$

Hence, the theorem follows using induction on $w = \mathrm{wt}(c)$. □

We now focus on the code words that we described in Section 2.4.1.

Remark 2.4.27. Let c be a small weight code word and q sufficiently large. Following the arguments in the proof of Theorem 2.4.26, we know that $c = c_1 + \cdots + c_m$, with c_i, $1 \leq i \leq m$, a code word that we described in Theorem 2.4.9 and Example 2.4.10, such that $\mathrm{wt}(c_1 + \cdots + c_{m'}) < \mathrm{wt}(c_1 + \cdots + c_{m'+1})$ for all $1 \leq m' \leq m$. From this observation, it immediately follows that the code words that we described in Theorem 2.4.9 and Example 2.4.10 are the code words of smallest weights.

Now we consider small weight code words different from the ones described in Theorem 2.4.9. Let c be a code word of weight at most $4q^{2n-2}(q-1)$, q sufficiently large. Since c is not of the type we described in Theorem 2.4.9, c can be written as a linear combination of at least two of these code words. By the above arguments, we can find a code word c' which is a linear combination of precisely two of these code words, such that $\mathrm{wt}(c') \leq \mathrm{wt}(c)$. In particular, we can find $\alpha, \alpha' \in \mathbb{F}_p^*$ and n-spaces π, π', $\pi \notin \{\pi', \pi'^\sigma\}$, such that $c' = \alpha(v_\pi - v_{\pi^\sigma}) + \alpha'(v_{\pi'} - v_{\pi'^\sigma})$

and $\mathrm{wt}(c') \leq 4q^{2n-2}(q-1)$. Let S be the support of c'. We know $S \subseteq ((\pi\Delta\pi^\sigma) \cup (\pi'\Delta\pi'^\sigma)) \cap \mathcal{H}(2n+1, q^2)$. However, it can be seen that $|(\pi\Delta\pi^\sigma) \cap (\pi'\Delta\pi'^\sigma)| \leq 4q^{2n-2}$. Hence,

$$|S| \geq \mathrm{wt}(\alpha(v_\pi - v_{\pi^\sigma})) + \mathrm{wt}(\alpha'(v_{\pi'} - v_{\pi'^\sigma})) - |(\pi\Delta\pi^\sigma) \cap (\pi'\Delta\pi'^\sigma)| > 4q^{2n-2}(q-1) \ ,$$

a contradiction. It follows that the only code words of weight at most $4q^{2n-2}(q-1)$ are of the type described in Theorem 2.4.9.

Note that Theorem 2.4.26 only proves the second half of Theorem 2.4.3. From Remark 2.4.27 now the first half also follows.

2.5 LDPC codes from partial geometries

In this section we study several high-rate LDPC codes derived from partial geometries. We study in particular the minimum distance and stopping distance the two main infinite classes of these partial geometries, and we improve the known bounds on this minimum distance. In some cases, we can determine the exact minimum distance and/or stopping distance.

We focus on the case where the largest set ($\mathcal{P}$ or $\mathcal{B}$) corresponds to the positions, as this results in the highest code rates, which is important for LDPC transmission. In case anything noteworthy can be said about the other (lower-rate) code, we will add this in a remark.

Definition 2.5.1. An (s, t, α)-*partial geometry* is an incidence structure $(\mathcal{P}, \mathcal{B}, \in)$ for which:

(a) each block contains exactly $s+1$ points and each point is contained in exactly $t+1$ blocks;

(b) any two distinct blocks have at most one point in common; and

(c) for any non-incident point-block pair (p, L) there are exactly α blocks which contain p and which intersect L.

A partial geometry is called *proper* when $1 < \alpha < \min(s, t)$.

Clearly, the dual of an (s, t, α)-partial geometry is a (t, s, α)-partial geometry. In an (s, t, α)-partial geometry $(\mathcal{P}, \mathcal{B}, I)$, one has $|\mathcal{P}| = \frac{(s+1)(st+\alpha)}{\alpha}$ and $|\mathcal{B}| = \frac{(t+1)(st+\alpha)}{\alpha}$. Hence, $|\mathcal{P}| \geq |\mathcal{B}|$ is equivalent to $s \geq t$.

Definition 2.5.2. An (n, k)-*arc* $\mathcal{K}$ in $\mathrm{PG}(2, q)$ is a set of n points of $\mathrm{PG}(2, q)$, such that each line intersects $\mathcal{K}$ in at most k points. Clearly, the size $|\mathcal{K}| = n$ of an (n, k)-arc is at most $1 + (q+1)(k-1) = qk - q + k$, since each of the $q+1$ lines through any one point of $\mathcal{K}$ can contain at most $k-1$ other points of $\mathcal{K}$.

Note that n, k here are new letters specific to this section, which plays entirely in a planar setting. They do not indicate the code length/dimension or the dimension of any (sub)spaces.

Definition 2.5.3. A *maximal* (n,k)*-arc* $\mathcal{K}$ in $\mathrm{PG}(2,q)$ is an (n,k)-arc of size $n = qk - q + k$, with $1 < k < q$. It has been shown [9] that maximal (n,k)-arcs in $\mathrm{PG}(2,q)$ only exist when q is even and k divides q. In that case, each line intersects $\mathcal{K}$ in either 0 (and then the line is called *skew*) or k (and then the line is called *secant*) points. A maximum 2-arc is called a *hyperoval*. When it is not necessary to specify k, or when the parameters are clear from the context, we will simply write *maximal arc*.

Up to dualization, only two infinite classes of proper partial geometries are known, and both of them are related to maximal arcs. We now provide a construction of these geometries below.

In particular, we will focus on these two constructions.

(a) Let $\mathcal{K}$ be a maximal (n,k)-arc in $\mathrm{PG}(2,q)$, with q even. Define $S(\mathcal{K}) = (\mathcal{P}, \mathcal{B}, \in)$ as follows: $\mathcal{P}$ is the set of points outside of $\mathcal{K}$, and $\mathcal{B}$ is the set of lines which contain at least one (and hence exactly k) points of $\mathcal{K}$. Then $S(\mathcal{K})$ is an (s,t,α)-partial geometry, where $s = q - k$, $t = q - \frac{q}{k}$ and $\alpha = q - \frac{q}{k} + 1 - k$.

(b) Let again $\mathcal{K}$ be a maximal (n,k)-arc in $\mathrm{PG}(2,q)$, with q even. Define $T_2^*(\mathcal{K}) = (\mathcal{P}, \mathcal{B}, \in)$ in the following way: embed this plane $\mathrm{PG}(2,q)$ (containing $\mathcal{K}$) as a plane π_0 in $\mathrm{PG}(3,q)$, let $\mathcal{P}$ be the set of points of $\mathrm{PG}(3,q)$ outside of π_0 and let $\mathcal{B}$ be the set of lines of $\mathrm{PG}(3,q)$ which intersect π_0 in a point of $\mathcal{K}$ (and in no points of $\pi_0 \setminus \mathcal{K}$). Then $T_2^*(\mathcal{K})$ is an (s,t,α)-partial geometry, where $s = q - 1$, $t = |\mathcal{K}| - 1 = qk - q + k - 1$ and $\alpha = k - 1$.

Similar to [74], we consider C to be a linear code over a finite field $\mathbb{F}$, having its parity check matrix H equal to the incidence matrix of a partial geometry. Important hereby is that we will always set $q = p^h$, with $\mathbb{F} = \mathbb{F}_p$. Hence, the characteristic of the code's field and of the geometry's field coincide. This is known experimentally to yield the highest code rates. Since maximal (n,k)-arcs only exist when q is even, we will only consider binary codes in this section. We will now further study each of these codes, and we study the minimum distance of these codes. We improve the existing bounds from [74] and we try to determine when our new bounds are sharp.

2.5.1 The codes arising from $S(\mathcal{K})$

In this section we will study the LDPC codes derived from the first infinite class: $S(\mathcal{K})$, with $\mathcal{K}$ a maximal (n,k)-arc in $\mathrm{PG}(2,q)$.

Points as positions

The following conjecture was proven in [23, Theorem 3.10] when $\mathcal{K}$ is a regular hyperoval; we conjecture it to be true for arbitrary maximal arcs.

Conjecture 2.5.4. *Let $\mathcal{K}$ be a maximal arc in $\mathrm{PG}(2,q)$, q even. Then the incidence vector of each line in $\mathrm{PG}(2,q)$ can be written as a linear combination of incidence vectors of secant lines to $\mathcal{K}$.*

Computer simulations show that Conjecture 2.5.4 is true for all maximal arcs in $\mathrm{PG}(2,q)$ with $q \leq 32$, and for all known maximal arcs in $\mathrm{PG}(2,q)$, $q \leq 64$. We conjecture it to be true for arbitrary q; it would be interesting to prove this in general. However, this appears to be a nontrivial problem even for $k = 2$ (when $\mathcal{K}$ is a nonregular hyperoval). An equivalent way of stating this problem is to ask whether the binary rank of the incidence matrix is always equal to $3^h + 1$, the binary rank of the incidence matrix of $\mathrm{PG}(2,q)$, for q even [126].

A large class of maximal arcs in $\mathrm{PG}(2,q)$, q even, was constructed by Denniston [32]. In particular, he constructed an example for all possible values of k and q. However, other examples have been constructed as well, notably by Mathon [101] and others.

Theorem 2.5.5. *If $\mathcal{K}$ is an arc of Denniston or Mathon type, then Conjecture 2.5.4 is true.*

Proof. In [23] it is proven that the incidence vector of every projective line in $\mathrm{PG}(2,q)$, q even, is a linear combination of incidence vectors of secant lines to a regular hyperoval. Since all Denniston and Mathon arcs contain a regular hyperoval, it follows that the set of secant lines to the Denniston arc contains the set of secant lines to this included regular hyperoval, and hence, all projective lines are also a linear combination of secants to this Denniston arc. Thus, Conjecture 2.5.4 holds for all Denniston and Mathon arcs. □

From now on, we assume $\mathcal{K}$ to be a maximal (n,k)-arc for which Conjecture 2.5.4 holds (for example, a maximal arc of Denniston or Mathon type).

Lemma 2.5.6. *Let $C^{\perp}_{\mathrm{PG}(2,q)}$ be the code with the incidence matrix of $\mathrm{PG}(2,q)$ as its parity check matrix. Then we have $C \subseteq C^{\perp}_{\mathrm{PG}(2,q)}$. More precisely, C is exactly the subset of code words of $C^{\perp}_{\mathrm{PG}(2,q)}$ which do not contain any point of $\mathcal{K}$ in their support.*

Proof. Let $c \in C$, hence $c \cdot \ell = 0$ over $\mathbb{F}_2$ for every secant ℓ. Consequently, $c \cdot (\ell_1 + \ell_2 + \cdots + \ell_m) = 0$ for any m and any linear combination $\ell_1 + \ell_2 + \cdots + \ell_m$ of secants. We have chosen $\mathcal{K}$ such that Conjecture 2.5.4 holds, hence we have $c \cdot \ell = 0$ for every projective line. Therefore, one has indeed that $c \in C^{\perp}_{\mathrm{PG}(2,q)}$.

The second part is easy now: on one hand, c does not contain any points of $\mathcal{K}$ in its support; on the other hand, any code word of $C^{\perp}_{\mathrm{PG}(2,q)}$ which does not contain a point of $\mathcal{K}$ in its support, is a code word of C. □

Theorem 2.5.7. *For this code C, $d(C) \geq q + 2$. Equality in this bound is attained if and only if there exists a hyperoval disjoint from $\mathcal{K}$.*

Proof. Let c be any code word in C. From the first part of Lemma 2.5.6 it follows that $c \in C^{\perp}_{\mathrm{PG}(2,q)}$. Therefore, $\mathrm{wt}(c) \geq q+2$, and equality is attained if and only if c is the incidence vector of a hyperoval.

The second part in Lemma 2.5.6 tells us that the incidence vector of a hyperoval lies in C if and only if the hyperoval is disjoint from $\mathcal{K}$. This concludes the proof. □

Remark 2.5.8. Brute-force calculations for $q \leq 16$ have shown that there always exists a hyperoval skew to $\mathcal{K}$ when $1 < k < q$, yielding $d(C) = q + 2$ for every maximal k-arc in $\mathrm{PG}(2, q)$ with $q \leq 16$. It would be interesting to find out if this holds in general.

Theorem 2.5.9. *One has* $sd(C) \geq q - \frac{q}{k} + 2$.

Proof. Let S be any nonempty stopping set and let $r \in S$. Then $r \notin \mathcal{K}$ and hence there are $q - \frac{q}{k} + 1$ secant lines through r. The stopping set property requires each of these lines to contain at least one extra point of S, hence $|S| \geq q - \frac{q}{k} + 2$. □

Lines as positions

For the geometry $S(\mathcal{K})$, it turns out that we do not need to study the two types of codes separately (once for points corresponding to positions, and once for lines corresponding to positions), as we will now show.

Definition 2.5.10. Let $\mathcal{K}$ be a maximal (n, k)-arc. Consider the set of lines of $\mathrm{PG}(2, q)$ skew to $\mathcal{K}$. Clearly, each point lies on either 0 or $\frac{q}{k}$ of these lines, hence it is the dual of a maximal $\frac{q}{k}$-arc. This maximal $\frac{q}{k}$-arc is called the *dual maximal arc* of $\mathcal{K}$.

Theorem 2.5.11. *Let $\mathcal{K}$ be a maximal arc and let $\mathcal{K}'$ be its dual. Then the block code derived from $\mathcal{K}$ is equivalent to the point code derived from $\mathcal{K}'$.*

Proof. Let $\mathcal{P}$ and $\mathcal{B}$ respectively be the point and block set of $S(\mathcal{K})$. Similarly, denote by $\mathcal{P}'$ and $\mathcal{B}'$ the point and block set of $S(\mathcal{K}')$, i.e. $\mathcal{P}'$ is the set of lines skew to $\mathcal{K}$ and $\mathcal{B}'$ is the set of points lying on $\frac{q}{k}$ lines of $\mathcal{P}$.

Clearly, $\mathcal{B}$ is the complement of $\mathcal{P}'$. Less clearly, $\mathcal{B}'$ is the complement of $\mathcal{P}$; in other words: the points lying on $\frac{q}{k}$ lines skew to $\mathcal{K}$ are exactly the points outside of $\mathcal{K}$. We will explain this now. A point of $\mathcal{P}$ is clearly not contained in $\mathcal{B}'$, since any element of $\mathcal{B}'$ lies on a line skew to $\mathcal{K}$. On the other hand, a point outside of $\mathcal{P}$ lies on $\frac{qk-q+k}{k} = q - \frac{q}{k} + 1$ secants to $\mathcal{K}$, hence it lies on $\frac{q}{k}$ lines skew to $\mathcal{K}$, i.e. it is contained in $\frac{q}{k}$ elements of $\mathcal{P}'$, which is exactly the definition of an element of $\mathcal{B}'$.

Hence, the point-by-line incidence matrix of $\mathrm{PG}(2, q)$ can be written in the following way:

$$\left(\begin{array}{c|c} \mathcal{P} \times \mathcal{P}' & \mathcal{B}' \times \mathcal{P}' \\ \hline \mathcal{P} \times \mathcal{B} & \mathcal{B}' \times \mathcal{B} \end{array}\right).$$

Now, a point in $\mathcal{P}$ cannot lie on a line of $\mathcal{P}'$, hence $\mathcal{P} \times \mathcal{P}'$ is the all-zero (sub)matrix.

Since $\mathrm{PG}(2, q)$ is self-dual, every row in the incidence matrix of $\mathrm{PG}(2, q)$ also has to appear as a column of this matrix, and vice versa. Since there are no zero rows in $\mathcal{P} \times \mathcal{B}$, and there are no zero columns in $\mathcal{B}' \times \mathcal{P}'$, it follows that the set of columns of $\mathcal{P} \times \mathcal{B}$ is equal to the set of rows of $\mathcal{B}' \times \mathcal{P}'$. Hence, the block code constructed from $\mathcal{P} \times \mathcal{B}$ is equivalent to the point code constructed from the transpose of $\mathcal{B}' \times \mathcal{P}'$, which was to be proven. □

So, in this case, both are equivalent.

2.5.2 The codes arising from $T_2^*(\mathcal{K})$

Here we consider three cases: $\mathcal{K}$ is a maximal k-arc, $\mathcal{K}$ is a Baer subplane and $\mathcal{K}$ is a Hermitian arc. In each of these cases one has $|\mathcal{K}| > q$, hence there are more blocks than points and the code will be constructed using blocks (here: lines) as the positions of the code.

Definition 2.5.12. A $(q+t,t)$-*arc of type* $(0,2,t)$ in $\mathrm{PG}(2,q)$, with $2 < t < q$, or shortly a $\mathrm{KM}_{(q,t)}$-*arc* or *KM-arc*, is a set S of $q+t$ points in $\mathrm{PG}(2,q)$ for which every projective line ℓ meets S in either 0, 2 or t points. To avoid trivial constructions we will always assume $1 < t < q$.

Definition 2.5.12 was introduced by Korchmáros and Mazzocca in [80] (hence the short name) and it was proven that KM-arcs can only exist if q is even. Moreover, they prove that t needs to be a divisor of q, i.e. $t = 2^r$ with $r \leq h$. Several infinite families of KM-arcs are known, see [47, 80, 141]. A hyperoval in $\mathrm{PG}(2,q)$, $q = 2^h$ with $h \geq 1$, can be seen as a $\mathrm{KM}_{(q,2)}$-arc. One can see $\mathrm{KM}_{(q,t)}$-arcs as a generalization of hyperovals. The symmetric difference of two lines of $\mathrm{PG}(2,q)$ can be seen as a $\mathrm{KM}_{(q,q)}$-arc .

For the geometry $T_2^*(\mathcal{K})$, we have to consider three cases: $\mathcal{K}$ is a maximal (n,k)-arc, $\mathcal{K}$ is a Baer subplane, or $\mathcal{K}$ is a Hermitian arc. In each of these cases, one has $|\mathcal{K}| > q$, hence there are more blocks than points and the code will be constructed using blocks (here: lines) as the positions of the code.

Definition 2.5.13. Let π be any plane different from π_0 such that the line $L = \pi_0 \cap \pi$ contains at least two points of $\mathcal{K}$. Let p_1, p_2 be two distinct points of $\mathcal{K} \cap L$. Define φ in the following way: all the lines in $\pi \setminus \pi_0$ through p_1 map to 1, all the lines in $\pi \setminus \pi_0$ through p_2 map to -1 and all other lines map to 0. Then $\sum_{L \ni r} \varphi(L) = 0$ for any point r, since for $r \in \pi$ we get one line with coefficient 1 and one line with coefficient -1 (and all other lines 0), and for $r \notin \pi$ we only sum up lines with coefficient 0; hence this defines a code word of weight $2q$. Such a code word is called a *plane word*.

Remark 2.5.14. The construction in Definition 2.5.13 was introduced in [113]. Note that this construction yields a code word, no matter over which field $\mathbb{F}$ the code is considered. In the case that $\mathrm{char}\,\mathbb{K} \neq p$, this code has been studied extensively in [139, 140]. In this case, the minimum distance is $2q$ and when $\mathcal{K}$ is a maximal k-arc, a Baer subplane or a Hermitian arc, the code words of minimum weight are exactly the plane words. Moreover, the set of plane words generates the entire code, i.e. every code word of C is a linear combination of plane words. The dimension of C is in this case known to be $|\mathcal{K}| - 1 + (q-1)(|\mathcal{K}|(q+1) - L_N)$ [140, Theorem 5.5], where L_N is the number of lines in π_0 not skew to $\mathcal{K}$. Since any line in π_0 intersects a maximal k-arc in 0 or k lines, the number of non-skew lines is $\frac{|\mathcal{K}|(q+1)}{k}$, and as stated before we have $|\mathcal{K}| = qk - q + k$. Hence, the dimension of the code only depends on k and n. If $\mathcal{K}$ is a Baer subplane or a Hermitian arc, there are no lines in π_0 skew to $\mathcal{K}$, and hence the incidence matrix has full rank in these cases.

In [113, Proposition 5], it is shown[4] that $sd(C) \geq q + \sqrt{q}$ (if $\mathcal{K}$ is a Baer subplane or Hermitian arc) or $sd(C) \geq q + \frac{q}{k-1}$ (if $\mathcal{K}$ is a maximal (n,k)-arc). In the first case, this bound is sharp

[4] Actually, their claim was about $d(C)$, but their proof works for $sd(C)$ as well.

when $p = 2$, since Korchmáros and Mazzocca [80] have shown the existence of a $\mathrm{KM}_{(q,\sqrt{q})}$-arc with the points on a Baer subline as its dual t-secants. For $p \neq 2$ or if $\mathcal{K}$ is a maximal (n, k)-arc with $k > 2$, this bound is no longer sharp, and the exact minimum distance is not known in these cases.

In the second case (when $\mathcal{K}$ is a maximal (n, k)-arc and hence $p = 2$), we can however find partial results. For $k = 2$, one can easily show that $d = 2q$, and the code words of minimum weight correspond to dual $\mathrm{KM}_{(q,q)}$-arcs, which are exactly the plane words from [139, 140]. For $k > 2$, we can however find a geometrical upper bound on the minimum distance d, in the following theorem, which is valid for any maximal arc $\mathcal{K}$.

Theorem 2.5.15. *For this code C, $d(C) \leq q + r$, where r is the smallest integer for which there exists a line ℓ in π_0 and a dual $\mathrm{KM}_{(q,r)}$-arc having its dual r-secants contained in $\ell \cap \mathcal{K}$.*

Proof. Let ℓ be such a line and let S be the line set of such a dual $\mathrm{KM}_{(q,r)}$-arc. Clearly, S is the support of a code word of C, of weight $q + r$. Hence we have indeed $d(C) \leq q + r$. □

Conjecture 2.5.16. *The bound in Theorem 2.5.15 is always sharp, i.e. $d(C) = q + r$, where r is the smallest integer for which there exists a line ℓ in π_0 and a dual $\mathrm{KM}_{(q,r)}$-arc having its dual r-secants contained in $\ell \cap \mathcal{K}$.*

Remark 2.5.17. Computer simulations have shown the bound in Theorem 2.5.15 to be sharp for small values of q and for several constructions with larger q. We conjecture it to be sharp for all q. In all cases we tested, this resulted in a maximum weight of

$$d(C) = q + \frac{q}{2^{\lfloor \log_2(k-1) \rfloor}} = q + \frac{2q}{k} \,. \tag{2.5}$$

It would be interesting to prove this in the general case; even for $k = 4$ we only achieved a partial result (the conjecture being that the case after 'or' in Theorem 2.5.18 can never occur).

Theorem 2.5.18. *When $k = 4$,*

- *either (2.5) holds (i.e. $d(C) = \frac{3q}{2}$),*
- *or $d(C) = 4s$ with $\frac{q}{3} \leq s \leq 3\frac{q}{8}$ and the code words of minimum weight consist of four sets of s lines, each concurrent at a point of a fixed line ℓ, with the additional property that each line contains $\frac{3s-q}{2}$ points on four of these lines and $\frac{3(q-s)}{2}$ points on two of these lines.*

Proof. Since $k = 4$, we can find four points p_1, p_2, p_3, p_4 on $\mathcal{K} \cap L$, where L is the line with equation $X_0 = 0$. Let c be a code word of minimum weight, then $\mathrm{supp}(c)$ corresponds to a set of lines such that each point outside of L lies on either 0, 2 or 4 lines of this set.

Let S, T, U, V be respectively the set of lines in $\mathrm{supp}(c)$ through p_1, p_2, p_3, p_4. Denote $s = |S|$, $t = |T|$, $u = |U|$, $v = |V|$ and $w = s + t + u + v$. Now fix one line ℓ in S. Each of the affine points on ℓ needs to be contained in either one or three others line of T, U, V. In particular,

since there are only q affine points on ℓ, we know that ℓ contains $\frac{(t+u+v)-q}{2} = \frac{w-s-q}{2}$ dual four-secants and $\frac{3q-(t+u+v)}{2} = \frac{3q-w+s}{2}$ dual two-secants.

Let k be the total number of dual four-secants. Summing up the above for all $\ell \in S$, we obtain $\frac{s(t+u+v-q)}{2} = k$, or equivalently, $s^2 - s(w-q) + 2k = 0$. Similarly for T, U and V, we obtain the system of equations

$$\begin{cases} s^2 - s(w-q) + 2k = 0, \\ t^2 - t(w-q) + 2k = 0, \\ u^2 - u(w-q) + 2k = 0, \\ v^2 - v(w-q) + 2k = 0. \end{cases}$$

Since the quadratic equation $x^2 - x(w-q) + 2k = 0$ has at most two roots, there can be at most two different values among s, t, u, v. Now there are two options: either $s = t = u = v$, or not all variables are equal, without loss of generality $s \neq t$. If $s \neq t$, then their sum needs to be $w - q$ because of de Viete's formula, hence $s + t = w - q$ and $u + v = q$. Now, since $u, v \in \{s, t\}$, we may assume without loss of generality that $t = u$ and we split cases between $t = v$ and $s = v$:

- If $t = v$, then $t = u = v$, and since $u + v = q$ this means $t = u = v = \frac{q}{2}$. Since $s + t = w - q$, this implies $w = s + \frac{3}{2}q$. Either $s = 0$, and then $T \cup U \cup V$ is a dual $\mathrm{KM}_{(q,t)}$-arc with $t = \frac{q}{2}$; or $s > 0$ and then $w > \frac{3}{2}q$ and then this is clearly not a code word of minimum weight.
- If $t \neq v$, i.e. $s = v$, then $w - q = s + t = v + u = q$, hence $w = 2q$, which is never a code word of minimum weight.

Hence, if another code word of weight no larger than $\frac{3}{2}$ exists, it must be of the described form with $4s \leq \frac{3}{2}q$, and since $d(C) \geq q + \frac{q}{k-1}$, it must also have $s \geq \frac{q}{3}$. □

Remark 2.5.19. It is unknown whether the second case in Theorem 2.5.18 can actually occur. Experimentally, only the first case seems to occurs, confirming the conjecture in Remark 2.5.17.

We summarize the new results on the minimum distances in Table 2.2. In almost all cases, we obtain improvements to the bound in Theorem 2.1.3.

Table 2.2: Summary of the new bounds obtained for partial and semipartial geometry codes. Here, $\approx$ stands for conjectured equality.

Code	$p = 2$	$p \neq 2$
General (semi)partial geometry *(blocks correspond to positions)*	$d \geq s + 2$	$d \geq \frac{2}{p}\left((p-1)s + p\right)$
General (semi)partial geometry *(points correspond to positions)*	$d \geq t + 2$	$d \geq \frac{2}{p}\left((p-1)t + p\right)$
$S(\mathcal{K})$ with $\mathcal{K}$ maximal (n,k)-arc	$d \approx q + 2$	N/A
$T_2^*(\mathcal{K})$ with $\mathcal{K}$ maximal (n,k)-arc	$d \approx q + \frac{2q}{k}$	N/A

Chapter 3

$(q+t,t)$-arcs of type $(0,2,t)$

In this chapter I will discuss $(q+t,t)$-arcs of type $(0,2,t)$, or shortly $\mathrm{KM}_{q,t}$-arcs, in Desarguesian projective planes of even order. In Section 3.1, I discuss the motivations for studying these arcs, and the state of the art. In Section 3.2, I discuss an elegant basis for the projective plane code, and I pose a motivated conjecture, supported by computer simulations, on how linear dependency between incidence vectors of lines translates to the existence of certain $\mathrm{KM}_{q,t}$-arcs. In Section 3.3, I prove this conjecture for $k = q/2$, indirectly providing an alternative proof for the classification of the projective triads [122, 131]. In Section 3.4, I present the main result: despite being unable to prove the conjectures posed in Section 3.2, I used them as inspiration to invent a new construction technique for $\mathrm{KM}_{q,q/4}$-arcs, and subsequently proved this new construction by different means. This resulted in both a new infinite class of $\mathrm{KM}_{q,t}$-arcs and a great support for the plausibility of the conjectures posed in Section 3.2. The results in this chapter were published in Finite Fields Appl. [141].

3.1 Preliminaries and motivation

The incidence matrix M_q of $\mathrm{PG}(2,q)$, with $q = p^h$ and p prime, has a p-rank of $\binom{p+1}{2}^h + 1$ [126] and is symmetric, because of the self-duality of $\mathrm{PG}(2,q)$. Two linear codes related to this matrix are commonly studied: the p-ary $[q^2+q+1, \binom{p+1}{2}^h + 1]$-code generated by M_q over $\mathbb{F}_p$, which we denote by C_{gen}, and the p-ary $[q^2+q+1, 2\binom{q+1}{2} - \binom{p+1}{2}^h]$-code with M_q as its parity check matrix over $\mathbb{F}_p$, which we denote by C_{pcm}. In this chapter, when studying C_{gen}, we let the points of the geometry correspond to the positions of the code, and when studying C_{pcm}, we let the lines of the geometry correspond to the positions of the code. For example, in the case of a binary code ($p = 2$), a code word of C_{pcm} is a set of lines such that each point is contained in an even number of these lines, and a code word of C_{gen} is the binary sum of any number of incidence vectors of lines.

The reason why we use a different setting for each code is the following. Since we will study the row span of C_{gen}, and in particular the dimension of certain subspaces of it, we are interested in linear combinations of rows of the incidence matrix M of $\mathrm{PG}(2,q)$ which yield

the zero vector. Now in the transposed matrix M^T, where columns correspond to lines and rows correspond to points, this is a linear combination of columns yielding the zero vector, which is well-known to correspond to a code word of the code defined by M^T as its parity check matrix, which is in our case C_{pcm}. A code word of C_{pcm} hence corresponds to a set of lines hitting each point an even number of times.

We recall from Section 2.2 that the minimum distance of C_{gen} is $q+1$ and that the code words of minimum weight are exactly the incidence vectors of the projective lines. The minimum weight of C_{pcm} is not known in general. For $p=2$, the minimum weight of C_{pcm} is $q+2$ and the code words of minimum weight are exactly the dual hyperovals [5].

In 1991, G.E. Moorhouse [105] found and proved an explicit basis for the rows of the incidence matrix, in the case $h=1$ (i.e. $\binom{p+1}{2}+1$ rows which are linearly independent). The construction is as follows: fix one line L and let $S=\{L\}$. Now consider the line L as the line at infinity of the projective plane. Then add to S all p affine lines through one point of L. Then add to S any $p-1$ affine lines through another point of L. Continue in this way, and finally add to S any one affine line through the second last point of L. Do nothing for the last point of L. Then S forms a basis for the p-ary row space of the incidence matrix.

For $q=p^h$, with p prime and $h>1$, the existence of a similar result has been an open problem for nearly 20 years now. The nature of finite fields of non-prime order suggests that any generalization of this result will no longer allow to pick the points/lines in arbitrary order. This is, however, not a though restriction: a general construction, even in one particular order, would already be an interesting result.

In Section 3.2, we provide a detailed conjecture, backed up by computer simulations, of how such a generalized Moorhouse basis for $\mathrm{PG}(2,q)$ can look like for the case $p=2$, i.e. $q=2^h$. We discuss a strong relationship with (dual) $(q+t,t)$-arcs of type $(0,2,t)$, a special type of small code words of C_{pcm}. In Section 3.3, we prove a special case of this conjecture and we derive from it an alternative proof for the classification of the projective triads. In Section 3.4, we construct a new infinite class of such arcs with $t=q/4$, parameters which were previously unknown to exist. We end by listing some possibilities for further work.

From now on, for the rest of this chapter, we will work completely in this dual setting and we will limit ourselves to the case that q is even, i.e. $q=2^h$.

3.2 A basis for $\mathrm{PG}(2,q)$, q even

Notation 3.2.1. We will denote

$$S(h,i) := \sum_{k=i}^{h} \binom{h}{k}.$$

For any projective point $p(0,1,\beta)$ with

$$\beta = a_{h-1}\alpha^{h-1} + a_{h-2}\alpha^{h-2} + \cdots + a_1\alpha + a_0 \in \mathbb{F}_q,$$

where α is a primitive element of $\mathbb{F}_q$ and all $a_i \in \mathbb{F}_2$, we denote $lp(p) = \max\{i : a_i \neq 0\} + 1$ and we call this the *leading position* of the point. The leading position of $(0, 1, 0)$ is defined to be 0 and the leading position of $(0, 0, 1)$ is defined to be $+\infty$. Finally, we denote by $|\beta| = |\{i : a_i \neq 0\}|$, i.e. its number of ones when written as a vector in $\mathbb{F}_2^h$.

A standard way to find a basis of any vector space, is to start from the zero vector space and sequentially add all vectors to it. A basis is then the set of vectors which caused an increase in dimension when they were added.

Using a row-reduced form to store the basis, this can be implemented efficiently in software. Applying this standard technique to the vector space spanned by the rows of the matrix of PG$(2, q)$, with $q = 2^h$, we find that the following pattern holds for all $q \leq 512$. We conjecture it to hold for all q.

Conjecture 3.2.2. *Let L be the projective line with equation $X_0 = 0$, and let A be the $1 \times (q^2 + q + 1)$-matrix containing the point-incidence vector of L. We again consider this line L as the line at infinity of an affine plane. Now, for*

$$p \in [(0, 1, 0), (0, 1, 1), (0, 1, \alpha), (0, 1, \alpha + 1), (0, 1, \alpha^2), \ldots, (0, 1, \alpha^{h-1} + \cdots + \alpha + 1), (0, 0, 1)],$$

in that order, we add the incidence vectors of each of the q affine lines through p to the set of rows of A. Then the rank of A increases by $S(h, i)$ when adding the lines through a point p with $lp(p) = i$, for $i = 0, 1, \ldots, h, +\infty$.

This yields us a more structural rank formula: the rank of the incidence matrix of PG$(2, q)$ is $3^h + 1$, which can be written as

$$1 + S(h, 0) + S(h, 1) + \underbrace{S(h, 2) + S(h, 2)}_{2 \text{ terms}} + \underbrace{S(h, 3) + \cdots + S(h, 3)}_{4 \text{ terms}} + \cdots .$$

When adding the lines in lexicographical order, one can even see a clear pattern in which lines eventually end up in the basis:

Conjecture 3.2.3. *The line $X_0 = 0$ and the set of lines*

$$\{\langle (0, 1, \beta), (1, 0, \gamma) \rangle : |\gamma| + lp(\beta) \leq h\}$$

together form a basis for C_{gen}.

Harder to verify by computer, but structurally more important, is the following Conjecture 3.2.4. Conjecture 3.2.4 provides a structural explanation for Conjecture 3.2.2 and on itself greatly generalizes Conjecture 1.1.16.

Conjecture 3.2.4. *The numbers from Conjecture 3.2.2 can be explained as follows.*

- *The vanishing of the term $\binom{h}{0}$ when adding any point p with $lp(p) > 0$, is explained by the presence of dual $(2q, q)$-arcs of type $(0, 2, q)$ in* PG$(2, q)$ *with as its dual t-secants: p and the points with lp at most 0 (i.e. $(0, 1, 0)$).*

- *The vanishing of the term* $\binom{h}{1}$ *when adding any point* p *with* $lp(p) > 1$*, is explained by the presence of dual* $\left(\frac{3}{2}q, \frac{1}{2}q\right)$*-arcs of type* $\left(0, 2, \frac{1}{2}q\right)$ *in* $\mathrm{PG}(2,q)$ *with as its dual* t*-secants:* p *and the points with* lp *at most* 1*.*

- *The vanishing of the term* $\binom{h}{i}$ *when adding any point* p *with* $lp(p) > i$ *(with* $0 \leq i \leq h-1$*) is explained by the presence of dual* $\left(2^h + 2^{h-i}, 2^{h-i}\right)$*-arcs of type* $(0, 2, 2^{h-i})$ *in* $\mathrm{PG}(2,q)$ *with as its dual* t*-secants:* p *and the points with* lp *at most* i*.*

- *The vanishing of the term* $\binom{h}{h-1}$ *when adding any point* p *with* $lp(p) > h-1$*, is explained by the presence of dual hyperovals that do not contain the line* $X_0 = 0$*, and in which* p *and the points with* lp *at most* $h-1$ *are dual secants. These can be seen as* $\left(2^h + 2, 2\right)$*-arcs of type* $(0, 2, 2)$ *in* $\mathrm{PG}(2,q)$ *with as its dual* t*-secants:* p *and the points with* lp *at most* $h-1$*.*

- *The vanishing of the term* $\binom{h}{h}$ *when adding the point* p *with* $lp(p) = +\infty$ *(which is equivalent to* $lp(p) > h$*), is explained by the presence of a dual hyperoval that does contain the line* $X_0 = 0$ *and in which all points on that line are dual secants. (To some extent, after removing the line at infinity this can be seen as* $\left(2^h + 1, 1\right)$*-arcs of type* $(0, 2, 1)$ *in* $\mathrm{PG}(2,q)$*, with as its dual* t*-secants the whole line* $X_0 = 0$*. Adding the line at infinity yields a code word of* C_{pcm} *as in the cases above.)*

Conjecture 3.2.4 is a strong generalization of Conjecture 1.1.16, which only claims the existence of the code words mentioned in Conjecture 3.2.4. Conjecture 1.1.16 has been open for over 20 years now. We hope that this more structural conjecture can give a new impulse to the problem. In particular, the author believes that one can find $(q+t,t)$-arcs of type $(0,2,t)$ for all parameters in Conjecture 1.1.16, with the additional requirement that these arcs are defined by sets of lines with linear $\mathbb{F}_2$-equations on their coefficients when considering $\mathbb{F}_{2^h}$ as $\mathbb{F}_2^h$, as in the examples constructed in Section 3.4.

To support the plausibility of Conjecture 3.2.4, let us look at some particular cases.

- The last bullet of Conjecture 3.2.4 is clear, since for each affine line ℓ there exist dual regular hyperovals containing both $X_0 = 0$ and ℓ.

- The first bullet is easily shown as follows: let L be a line intersecting $X_0 = 0$ in a point $p \neq (0,1,0)$. Then the incidence vector of L can be written as the sum of all incidence vectors of the other lines through p and the incidence vectors of all lines through $(0,1,0)$.

- The second bullet is not trivial anymore. We will prove this part in Lemma 3.3.3, which fully classifies all $(q + q/2, q/2)$-arcs of type $(0, 2, q/2)$ and gives a more concise construction than the one in [80].

- The third and fourth bullet are still open. Computer results suggest that the weight of each code word in the code generated by all lines through points with $lp \leq i$, is always a multiple of 2^{h-i+1}; but a proof of this is still unknown. However, despite Conjecture 3.2.4 only being a conjecture, an interesting result pops up: our linear dependence search yields code words of C_{pcm} which use only a small number of points on $X_0 = 0$. **If** Conjecture 3.2.4 holds, **then** these code words are likely to be the sum of one or

more $(q+t,t)$-arcs of type $(0,2,t)$. Using this idea, we obtained a new infinite family of $(q+t,t)$-arcs of type $(0,2,t)$, which is an interesting result on its own, and which also greatly improves the plausibility of Conjecture 3.2.4. This new infinite family is presented in Section 3.4.

When considering the points on $X_0 = 0$ in a different order, it seems that the rank of the matrix consisting of the line incidence vectors is never larger than what is claimed in Conjecture 3.2.2. With random ordering, it is also not true that using at most 2^i points of $X_0 = 0$, the weight of the obtained code words of C_{pcm} is always a multiple of 2^{h-i+1}. For example, for $q = 64$ and $i = 2$, one can obtain code words of weight 120 when the points are taken on an $\mathbb{F}_4$-subline.

As for future work, proving Conjecture 1.1.16 or Conjecture 3.2.4 would be an important achievement. However, it seems that this is a difficult problem. Intermediate results, improving our understanding of these structures, would be a good path to follow. In particular, proving Conjecture 3.2.2 by other methods could be a good first step, and finding more infinite classes of arcs could also potentially bring us closer to an answer to the problem.

In Section 3.3, we will prove the second bullet, and we discuss an interesting corollary about projective triads. As said before, if Conjecture 3.2.4 is true, then one should be able to construct dual $(q+t,t)$-arcs of type $(0,2,t)$ by looking at linear dependencies between incidence vectors of lines. In particular, we exploited this idea by studying the linear dependencies between the incidence vectors of lines through the points $(0,0,1)$, $(0,1,0)$, $(0,1,1)$, $(0,1,\alpha)$ and $(0,1,\alpha^2)$. Adding these vectors to a vector space in a well-chosen order and using the standard technique from the start of this section, we found linear dependencies between these lines resulting in $(q+q/4,q/4)$-arcs of type $(0,2,q/4)$ for all $q \geq 512$. For $q = 128$ and $q = 512$, this resulted in arcs of previously unknown parameters. And again, the fact that this technique works again strengthens the plausibility of Conjecture 3.2.4. In Section 3.4, we used the above observation and the arcs derived from it, to obtain a general construction of dual $(q+q/4,q/4)$-arcs of type $(0,2,q/4)$ in $\mathrm{PG}(2,q)$, q even. For $q = 2^h$ with h odd, such arcs were not previously known.

3.3 Projective triads and $(q+t,t)$-arcs of type $(0,2,t)$

Definition 3.3.1. In $\mathrm{PG}(2,q)$, q even, consider three lines L_1, L_2, L_3, concurrent at a point r. A *projective triad* is a set S of $\frac{3}{2}q+1$ points of $\mathrm{PG}(2,q)$, contained in $L_1 \cup L_2 \cup L_3$ and containing r, such that each line L_i contains $\frac{q}{2}+1$ points of S, and each projective line not through r intersects S in 1 or 3 points.

Projective triads are mainly studied in the context of blocking sets.

Remark 3.3.2. Let S be a projective triad and let $S' = (L_1 \cup L_2 \cup L_3) \setminus S$. Then

- each line L_i contains $(q+1) - \left(\frac{q}{2}+1\right) = \frac{q}{2}$ points from S',
- each other line through r contains $1-1=0$ points from S',

- each line not through r contains $3-i$ points from S' with $i \in \{1,3\}$, hence each such line contains 0 or 2 points from S'.

All in all, each line contains 0, 2 or $\frac{q}{2}$ points of S'. Since $|S'| = (3q+1) - \left(\frac{3}{2}q+1\right) = q + \frac{q}{2}$, it follows that S' is a $(q+t,t)$-arc of type $(0,2,t)$, for $t = \frac{q}{2}$.

On the other hand, if S' is a $(q+t,t)$-arc of type $(0,2,t)$ with $t = \frac{q}{2}$, then it has a t-nucleus r, and hence it is contained in three lines L_1, L_2, L_3. In a similar fashion, $S = (L_1 \cup L_2 \cup L_3) \setminus S'$ is now a projective triad.

Hence, a projective triad uniquely corresponds to a $(q+t,t)$-arc of type $(0,2,t)$ with $t = \frac{q}{2}$. We will now classify the (dual) $(q+t,t)$-arcs of type $(0,2,t)$ with $t = \frac{q}{2}$. Without loss of generality we may assume the dual nucleus to be the line $X_0 = 0$ and by a coordinate transformation, we can let the dual secants be $(0,0,1)$, $(0,1,0)$ and $(0,1,1)$.

Lemma 3.3.3. *The subset of C_{pcm} of code words consisting of the lines through $(0,0,1)$, $(0,1,0)$ and $(0,1,1)$, different from the line $X_0 = 0$, is a subcode of dimension $h+2 = 0 + \binom{h}{0} + \left(\binom{h}{0} + \binom{h}{1}\right)$ with weight polynomial $1 + (4q-4)X^{3q/2} + 3X^{2q}$.*

Proof. We will completely classify the code words of this code. Denote our three points by $p_0(0,0,1)$, $p_1(0,1,0)$, $p_2(0,1,1)$. We recall that a code word here corresponds to a set of lines through one of p_0, p_1 or p_2, such that each point outside of $X_0 = 0$ is contained in an even number of lines (and hence in either 0 or 2 lines) of the set.

Let c be any code word. Denote by s,t,u respectively the number of lines in $\mathrm{supp}(c)$ through p_0, p_1, p_2. If at least one of s,t,u is zero, there are only four code words: the empty word and the $\binom{3}{2} = 3$ words formed by the $(2q,q)$-arcs of type $(0,2,q)$. Now consider any other word with $s,t,u > 0$. Any line through p_0 must intersect exactly $t+u$ lines through the other two points, hence $t+u = q$. Similarly, $s+t = q$ and $s+u = q$. Solving this system of equations, we get $s = t = u = q/2$.

Now coordinatize the lines of $\mathrm{supp}(c)$ as follows:

- write the lines through $p_0(0,0,1)$ as $[\mu,1,0]$ with $\mu \in S$, $|S| = q/2$,
- write the lines through $p_1(0,1,0)$ as $[\mu,0,1]$ with $\mu \in T$, $|T| = q/2$,
- write the lines through $p_2(0,1,1)$ as $[\mu,1,1]$ with $\mu \in U$, $|U| = q/2$.

The condition that each point $(1,x,y)$ should be contained in an even number of lines of $\mathrm{supp}(c)$, is equivalent to saying that for each $x,y \in \mathbb{F}_q$, an even number of the statements $x \in S$, $y \in T$, $x+y \in U$ should be fulfilled. In particular, for fixed $x \notin S$ we have $x+T = U$ and $x+U = T$, hence $\forall x, x', x'' \notin S$ we have $x+x'+x''+T = x+T$. This means that, considering $(\mathbb{F}_q, +)$ as a h-dimensional vector space over $\mathbb{F}_2$, the elements of S form an affine subspace. Similarly, T and U also need to be affine subspaces. From their sizes, S, T and U are affine hyperplanes.

For hyperplanes. it follows from $x+T=U$ that T and U need to be equal or parallel. Similarly, S,T,U all belong to the same parallel class. Hence, for some $c,c',c_0,c_1,\ldots,c_{h-1}\in\mathbb{F}_2$,

$$S=\{a_{h-1}\alpha^{h-1}+\cdots+a_1\alpha+a_0 : c_{h-1}a_{h-1}+\cdots+c_1a_1+c_0a_0=c\},$$

$$T=\{a_{h-1}\alpha^{h-1}+\cdots+a_1\alpha+a_0 : c_{h-1}a_{h-1}+\cdots+c_1a_1+c_0a_0=c'\},$$

$$U=\{a_{h-1}\alpha^{h-1}+\cdots+a_1\alpha+a_0 : c_{h-1}a_{h-1}+\cdots+c_1a_1+c_0a_0=c+c'+1\},$$

where we remind that an even number of $c,c',c+c'+1$ are zero, for each $c,c'\in\{0,1\}$.

Clearly, for each binary choice of these $h+2$ parameters, we get a different code word of weight $\frac{3}{2}q$, and the degenerate choice $c_0=c_1=\cdots=c_{h-1}=0$ yields the 4 code words mentioned at the start of the proof, having weight different from $\frac{3}{2}q$. □

As a by-product, we find a complete classification of the projective triads. A classification equivalent to Corollary 3.3.4 was found before in [131], and implicitly in [122].

Corollary 3.3.4. *Let L_1,L_2,L_3 be three concurrent lines in $\mathrm{PG}(2,q)$, q even. Let $A\in\mathrm{PGL}(3,q)$ be any coordinate transformation which maps these lines to the three lines $[0,0,1]$, $[0,1,0]$ and $[0,1,1]$. Let Π_t be any hyperplane in $\mathrm{AG}(h,2)$, with equation $c_{h-1}X_{h-1}+\cdots+c_1X_1+c_0X_0=t$, and let $c,c'\in\{0,1\}$. If we let*

$$\begin{aligned}S=\ & \{(1,0,0)\}\\ & \cup\{(a_{h-1}\alpha^{h-1}+\cdots+a_1\alpha+a_0,0,1)|(a_{h-1},\ldots,a_1,a_0)\in\Pi_c\}\\ & \cup\{(a_{h-1}\alpha^{h-1}+\cdots+a_1\alpha+a_0,1,0)|(a_{h-1},\ldots,a_1,a_0)\in\Pi_{c'}\}\\ & \cup\{(a_{h-1}\alpha^{h-1}+\cdots+a_1\alpha+a_0,1,1)|(a_{h-1},\ldots,a_1,a_0)\in\Pi_{c+c'}\},\end{aligned}$$

then $\{A^{-1}s|s\in S\}$ forms a projective triad on L_1,L_2,L_3. Moreover, if $q>2$, all $4q-4$ projective triads on L_1,L_2,L_3 arise from this construction.

3.4 A new infinite family

For an n-dimensional vector space V and a vector $v\in V$, we can computationally find v as a linear combination of a given basis $\{v_1,\ldots,v_n\}$ of V. Using only incidence vectors of lines as basis, and with v also an incidence vector of a different line, the linear combination $v=\sum_{i\in I}v_i$ shows that the corresponding set of lines forms a code word of C_{pcm}. If Conjecture 3.2.4 is true, every code word is composed of a linear combination of $(q+t,t)$-arcs of type $(0,2,t)$. For $t=q/4$, we found that in some cases, code words obtained in this way can be equal to such an arc. This observation led us to several examples, which we could embed in the following construction.

Let $\mathbb{F}_q$ be a finite field, with $q=2^h$, $h\geq 4$, built up with

$$\alpha^h=a_{h-1}\alpha^{h-1}+a_{h-2}\alpha^{h-2}+\cdots+a_1\alpha+a_0$$

with all $a_i\in\{0,1\}$, as its primitive polynomial. From [37] it follows that we may choose $a_{h-1}=a_{h-2}=0$ for $h\geq 8$. For $h=4,5,6,7$ one can easily verify that respectively $\alpha^4+\alpha+1=$

0, $\alpha^5+\alpha^2+1=0$, $\alpha^6+\alpha+1=0$ and $\alpha^7+\alpha^3+1=0$ are primitive polynomials of degree h with $a_{h-1}=a_{h-2}=0$.

Consider the projective line in $\mathrm{PG}(2,q)$ with equation $X_0=0$, and consider the points $(0,0,1)$, $(0,1,0)$, $(0,1,1)$, $(0,1,\alpha)$ and $(0,1,\alpha^2)$; these points will be the dual t-secants and the line $X_0=0$ will be the dual t-nucleus. Now we write all other lines through $(0,0,1)$ as $\langle(0,0,1),(1,t,0)\rangle$ with $t\in\mathbb{F}_q$ and we write all other lines through $(0,1,x)$ as $\langle(0,1,x),(1,0,t)\rangle$ with $t\in\mathbb{F}_q$.

Any element $z\in\mathbb{F}_q$ can be written uniquely as

$$z=z_{h-1}\alpha^{h-1}+z_{h-2}\alpha^{h-2}+\cdots+z_1\alpha+z_0,$$

with each $z_i\in\{0,1\}$. By $(z)_i$ we will denote z_i. We will now construct two (very similar) classes of examples: let $par\in\{0,1\}$ be a fixed element of $\mathbb{F}_2$; our infinite class will depend on par. Consider the following five sets of lines.

- $A:=\{\langle(0,0,1),(1,t,0)\rangle$ with $t_{h-2}=0,t_{h-3}=1\}$,
- $B:=\{\langle(0,1,0),(1,0,t)\rangle$ with $t_{h-1}=0,t_{h-2}=1\}$,
- $C:=\{\langle(0,1,1),(1,0,t)\rangle$ with $t_{h-2}=0,t_{h-3}+t_{h-4}+\cdots+t_0=par\}$,
- $D:=\{\langle(0,1,\alpha),(1,0,t)\rangle$ with $t_{h-1}+t_{h-2}=1,t_{h-3}+t_{h-4}+\cdots+t_0=par\}$,
- $E:=\{\langle(0,1,\alpha^2),(1,0,t)\rangle$ with $t_{h-1}=0,t_{h-2}+t_{h-3}+t_{h-4}+\cdots+t_0=par\}$,

then we will show that these form a dual $(q+q/4,q/4)$-arc of type $(0,2,q/4)$. That the set $A\cup B\cup C\cup D\cup E$ contains $q+q/4$ lines, is clear. That there are 5 points in which $q/4$ lines meet is also clear. What is not clear, is that each point with coordinates $(1,x,y)$ lies on either 0 or 2 of these lines. This will be proven in what follows.

Notation 3.4.1. Denote by S the set of lines of $A\cup B\cup C\cup D\cup E$.

From now on, we consider $X_0=0$ to be the line at infinity and we consider its complement as the affine plane $\mathrm{AG}(2,q)$.

Lemma 3.4.2. *The union of the affine points contained in any line of a set A, B, C, D or E, is for each of the 5 sets as follows:*

- $p_A:=\{(1,x,y):x_{h-2}=0,x_{h-3}=1\}$,
- $p_B:=\{(1,x,y):y_{h-1}=0,y_{h-2}=1\}$,
- $p_C:=\{(1,x,y):x_{h-2}+y_{h-2}=0,x_{h-3}+x_{h-4}+\cdots+x_0+y_{h-3}+y_{h-4}+\cdots+y_0=par\}$,
- $p_D:=\{(1,x,y):x_{h-2}+x_{h-3}+y_{h-1}+y_{h-2}=1,x_{h-4}+\cdots+x_0+y_{h-3}+y_{h-4}+\cdots+y_0=par\}$,
- $p_E:=\{(1,x,y):x_{h-3}+y_{h-1}=0,x_{h-4}+\cdots+x_0+y_{h-2}+y_{h-3}+y_{h-4}+\cdots+y_0=par\}$.

Proof. For A, B and C, this is obvious. For D, let the primitive polynomial be $\alpha^h = a_{h-3}\alpha^{h-3} + a_{h-4}\alpha^{h-4} + \cdots + a_1\alpha + a_0$ as assumed before. Then

$$\begin{aligned}\alpha x &= x_{h-1}\alpha^h + x_{h-2}\alpha^{h-1} + x_{h-3}\alpha^{h-2} + x_{h-4}\alpha^{h-3} + \cdots + x_0\alpha \\ &= x_{h-2}\alpha^{h-1} + x_{h-3}\alpha^{h-2} + (x_{h-4} + x_{h-1}a_{h-3})\alpha^{h-3} + \cdots \\ &\quad +(x_0 + x_{h-1}a_1)\alpha + x_{h-1}a_0.\end{aligned}$$

Hence, $t_{h-1} + t_{h-2} = (\alpha x + y)_{h-1} + (\alpha x + y)_{h-2} = 1$ reduces to $x_{h-2} + x_{h-3} + y_{h-1} + y_{h-2} = 1$ and

$$t_{h-3} + t_{h-4} + \cdots + t_0 = (\alpha x + y)_{h-3} + (\alpha x + y)_{h-4} + \cdots + (\alpha x + y)_0 = par$$

reduces to

$$x_{h-1}(a_{h-3} + a_{h-4} + \cdots + a_0) + x_{h-4} + \cdots + x_0 + y_{h-3} + y_{h-4} + \cdots + y_0 = par.$$

Since $a_{h-3} + a_{h-4} + \cdots + a_0 = 0$ (otherwise 1 is a root of the primitive polynomial), the latter reduces to

$$x_{h-4} + \cdots + x_0 + y_{h-3} + y_{h-4} + \cdots + y_0 = par$$

as claimed. Finally, for E,

$$\begin{aligned}\alpha^2 x &= \alpha(x_{h-1}\alpha^h + x_{h-2}\alpha^{h-1} + x_{h-3}\alpha^{h-2} + x_{h-4}\alpha^{h-3} + \cdots + x_0\alpha) \\ &= \alpha(x_{h-2}\alpha^{h-1} + x_{h-3}\alpha^{h-2} + (x_{h-4} + x_{h-1}a_{h-3})\alpha^{h-3} \\ &\qquad + \cdots + (x_0 + x_{h-1}a_1)\alpha + x_{h-1}a_0) \\ &= x_{h-2}\alpha^h + x_{h-3}\alpha^{h-1} + (x_{h-4} + x_{h-1}a_{h-3})\alpha^{h-2} + \cdots \\ &\qquad +(x_0 + x_{h-1}a_1)\alpha^2 + x_{h-1}a_0\alpha \\ &= (x_{h-3} + a_{h-1}x_{h-2})\alpha^{h-1} + (x_{h-4} + x_{h-1}a_{h-3} + x_{h-2}a_{h-2})\alpha^{h-2} \\ &\qquad + \cdots + (x_0 + x_{h-1}a_1 + x_{h-2}a_2)\alpha^2 \\ &\qquad +(x_{h-1}a_0 + x_{h-2}a_1)\alpha + x_{h-2}a_0 \\ &= x_{h-3}\alpha^{h-1} + (x_{h-4} + x_{h-1}a_{h-3})\alpha^{h-2} \\ &\qquad +(x_{h-5} + x_{h-1}a_{h-4} + x_{h-2}a_{h-3})\alpha^{h-3} + \cdots \\ &\qquad +(x_0 + x_{h-1}a_1 + x_{h-2}a_2)\alpha^2 + (x_{h-1}a_0 + x_{h-2}a_1)\alpha \\ &\qquad +x_{h-2}a_0.\end{aligned}$$

Hence, $t_{h-1} = (\alpha^2 x + y)_{h-1} = 0$ reduces to $x_{h-3} + y_{h-1} = 0$ and

$$\begin{aligned}&t_{h-2} + t_{h-3} + t_{h-4} + \cdots + t_0 \\ &\quad = (\alpha^2 x + y)_{h-2} + (\alpha^2 x + y)_{h-3} + (\alpha^2 x + y)_{h-4} + \cdots + (\alpha^2 x + y)_0 \\ &\quad = par\end{aligned}$$

reduces to

$$(x_{h-1} + x_{h-2})(a_{h-3} + \cdots + a_0) + x_{h-4} + \cdots + x_0 + y_{h-2} + y_{h-3} + y_{h-4} + \cdots + y_0 = par.$$

Since again $a_{h-3} + \cdots + a_0 = 0$, this reduces to

$$x_{h-4} + \cdots + x_0 + y_{h-2} + y_{h-3} + y_{h-4} + \cdots + y_0 = par$$

as claimed. □

Lemma 3.4.3. *Let L_1, L_2, L_3 be three concurrent lines of S. Then L_1, L_2, L_3 all belong to the same set A, B, C, D or E.*

Proof. It is clear that if two of them belong to different sets, they all belong to a different set. So what we have to verify is that

$$p_A \cap p_B \cap p_C = \emptyset, p_A \cap p_B \cap p_D = \emptyset, \ldots, p_C \cap p_D \cap p_E = \emptyset.$$

If we define $c := x_{h-4} + \cdots + x_0 + y_{h-4} + \cdots + y_0 + par$, then the systems of equations obtained in Lemma 3.4.2 become:

- $p_A := \{(1, x, y) : x_{h-2} = 0, x_{h-3} = 1\}$,
- $p_B := \{(1, x, y) : y_{h-1} = 0, y_{h-2} = 1\}$,
- $p_C := \{(1, x, y) : x_{h-2} + y_{h-2} = 0, x_{h-3} + y_{h-3} + c = 0\}$,
- $p_D := \{(1, x, y) : x_{h-2} + x_{h-3} + y_{h-1} + y_{h-2} = 1, y_{h-3} + c = 0\}$,
- $p_E := \{(1, x, y) : x_{h-3} + y_{h-1} = 0, y_{h-2} + y_{h-3} + c = 0\}$,

and one can easily verify that any three of these yield an inconsistent system of linear equations over $\mathbb{F}_2$. □

Theorem 3.4.4. *The set of lines S is a dual $(q + q/4, q/4)$-arc of type $(0, 2, q/4)$.*

Proof. As we remarked before, all that is left to prove is that each affine point lies on either 0 or 2 lines of S. From Lemma 3.4.3, it follows that an affine point cannot lie on three or more lines. Hence, each point lies on either 0, 1 or 2 lines. Now assume that there exists an affine point p which only lies on one line $L \in S$. The q lines in the 4 sets not containing L, intersect L in one affine point each, different from p. By the pigeonhole principle (q incidences for $q-1$ possible points), there must be two such lines intersecting L in the same point, a contradiction with Lemma 3.4.3. Hence, every affine point lies on 0 or 2 lines of S. □

Hence, we have constructed a new infinite families of $(q+t, t)$-arcs of type $(0, 2, t)$ with $t = q/4$.

Remark 3.4.5. An important step in the construction of this new class of arcs was the result that the coefficient of α^{h-1} and α^{h-2} can be assumed to be zero in the primitive polynomial of the field. Without this assumption, several extra terms would usually be added to the equations and we would no longer be able to cancel out the extra terms $a_0 + \cdots + a_{h-3}$. It would be very interesting to see a general construction without assumptions on the primitive polynomial – in particular, to understand how adding the terms α^{h-1} or α^{h-2} affects the form of the linear $\mathbb{F}_2$-equations imposed on t. A better understanding of this behavior would bring us closer to a general construction.

Remark 3.4.6. A similar trick might be used to obtain an infinite family of $(q + q/8, q/8)$-arcs of type $(0, 2, q/8)$, since [37] allows to choose the first three coordinates zero, where we only used this for the first two coordinates. Examples of such arcs however do not roll easily out of the projective plane matrix, which is why I have not yet constructed such a family. However I believe it can be done with relatively few effort.

Remark 3.4.7. Lemma 3.3.3 shows that all $(q+t,t)$-arcs of type $(0,2,t)$ are linear (i.e. they arise from linear equations considering $\mathbb{F}_q$ as $\mathbb{F}_2^h$, or in other words, their intersection with each t-secant forms an affine linear set) if $t = q/2$. For $t = q$, this is also clearly the case. For $t = 2$, this is trivially fulfilled since every two points in $\mathrm{AG}(2,h)$ form an affine subspace. A natural question would be if there exist $(q+t,t)$-arcs of type $(0,2,t)$ which are not linear, for $4 \leq t \leq q/4$. If not, this would be a major step towards proving Conjecture 3.2.4, and an important result on its own. It would also allow more efficient computer searches for classifying $(q+t,t)$-arcs of type $(0,2,t)$ in small planes.

The author believes that Conjecture 1.1.16 can be sharpened as follows.

Conjecture 3.4.8. *If 4 divides t and t divides q, then there exists a $(q+t,t)$-arc of type $(0,2,t)$ with the additional property that its intersection with each of its t-secants forms an affine linear set (the t-nucleus being the point at infinity).*

Chapter 4

Optimal blocking multisets

In this chapter we investigate (xv_t, xv_{t-1})-minihypers in $\mathrm{PG}(t,q)$, i.e. minihypers with the same parameters as a weighted sum of x hyperplanes. We characterize these minihypers as a nonnegative rational sum of hyperplanes and we use this characterization to extend and improve the main results of several papers which have appeared on the special case $t = 2$. We establish a new link with coding theory and we use this link to construct several new infinite classes of (xv_t, xv_{t-1})-minihypers in $\mathrm{PG}(t,q)$ that cannot be written as an integer sum of hyperplanes. This chapter is joint work with Ivan Landjev and the results have been published in J. Comb. Theory Ser. A [87].

4.1 Preliminaries and motivation

Notation 4.1.1. By $\mathbb{N}_0$, we denote the set of nonnegative integers. By $\mathcal{P}$, we denote the point set of $\mathrm{PG}(t,q)$. By $v_{u+1} = \frac{q^{u+1}-1}{q-1}$, we denote the number of points in any u-dimensional subspace of $\mathrm{PG}(t,q)$. The set of hyperplanes of $\mathrm{PG}(t,q)$ will be denoted by $\mathcal{H}$.

Definition 4.1.2. A *multiset* is a mapping $\mathfrak{K} : \mathcal{P} \to \mathbb{N}_0$. This mapping is extended additively to the power set of $\mathcal{P}$: for any $\mathcal{Q} \subseteq \mathcal{P}$, we put $\mathfrak{K}(\mathcal{Q}) = \sum_{x\in\mathcal{Q}} \mathfrak{K}(x)$. The image of a point or subset under this mapping is called the *multiplicity* of the point or subset. The *cardinality* of the multiset is $\mathfrak{K}(\mathcal{P})$. The *support* $\operatorname{supp}\mathfrak{K}$ of a multiset $\mathfrak{K}$ is defined as the set of all points of positive multiplicity:

$$\operatorname{supp}\mathfrak{K} = \{x \in \mathcal{P} \mid \mathfrak{K}(x) > 0\}.$$

Multisets with $\mathrm{Im}(\mathfrak{K}) = \{0,1\}$ are called *non-weighted*, or *projective*, and can be viewed as sets by identifying them with their supports. A multiset $\mathfrak{K}$ is said to be *proper* if $\operatorname{supp}\mathfrak{K} \neq \mathcal{P}$.

Definition 4.1.3. An $(f,m;t,q)$-*minihyper* is an m-fold blocking f-multiset w.r.t. hyperplanes in $\mathrm{PG}(t,q)$, i.e. a multiset of cardinality f in $\mathrm{PG}(t,q)$ such that each hyperplane has multiplicity at least m. If t and q are clear from the context, we will speak of an (f,m)-minihyper. Similarly, an $(n,w;t,q)$-*arc*, or (n,w)-arc for short, is a multiset of cardinality n in $\mathrm{PG}(t,q)$ such that each hyperplane has multiplicity at most w. A *proper minihyper* is a

minihyper which is proper as a multiset. To avoid trivial cases, we will always assume $t \geq 2$ and $f > 0$.

The set of points of a u-dimensional subspace of $\mathrm{PG}(t, q)$ is an example of a (v_{u+1}, v_u)-minihyper. Note that (xv_t, xv_{t-1})-minihypers in $\mathrm{PG}(t, q)$, with $x \leq q$, are always proper, since their total multiplicity is only $xv_t \leq qv_t < v_{t+1}$.

4.1.1 Motivation 1: from finite geometry

A first motivation to study (xv_t, xv_{t-1})-minihypers comes from the following problem.

Natural Problem 4.1.4. How can we m-block the hyperplanes with as few points f as possible?

One easily finds the following bound on this size.

Theorem 4.1.5. *Let $\mathfrak{F}$ be a proper (f, m)-minihyper in $\mathrm{PG}(t, q)$. Then $\dfrac{f}{m} \geq \dfrac{v_t}{v_{t-1}}$.*

Proof. Since $\mathfrak{F}$ is proper, there is a point $u \in \mathcal{P}$ with $\mathfrak{F}(u) = 0$. Let $\mathfrak{F}' = \sum_{H \ni u} \mathfrak{F} \cap H$.

- $|\mathfrak{F}'| = fv_{t-1}$ since each of the f points lies exactly on v_{t-1} such hyperplanes;
- $|\mathfrak{F}'| \geq mv_t$, since each of the v_t hyperplanes contain at least m points of $\mathfrak{F}$

Hence, $fv_{t-1} \geq mv_t$. □

So naturally, we are interested in finding which minihypers would reach equality in this bound.

Definition 4.1.6. An optimal blocking multiset is a proper minihyper with equality in this bound: $fv_{t-1} = mv_t$.

Corollary 4.1.7. *Since $\gcd(v_{t-1}, v_t) = 1$, an optimal blocking multiset has $f = xv_t$ and $m = xv_{t-1}$ for some positive integer x.*

Hence, the "best" blocking proper multisets in $\mathrm{PG}(t, q)$, are the (xv_t, xv_{t-1})-minihypers, with x any positive integer.

4.1.2 Motivation 2: from coding theory

Linear $[n, k, d]$-codes do not exist for all possible values of n, k, d. Ideally, one would like to optimize the parameters, simultaneously requiring

- n to be low, as this requires less computational complexity in decoding the code;

- k to be large (compared to n), allowing more data to be transmitted using the same amount of signals;
- d to be large, allowing to detect and correct more transmission errors during decoding.

This optimization problem is called the *fundamental problem of linear codes* and can (for a fixed field $\mathbb{F}_q$) be stated in 3 equivalent ways:

Open Problem 4.1.8 (Fundamental Problem).
- Given k, d, what is the smallest n for which an $[n, k, d]$-code exists over $\mathbb{F}_q$?
- Given n, k, what is the largest d for which an $[n, k, d]$-code exists over $\mathbb{F}_q$?
- Given n, d, what is the largest k for which an $[n, k, d]$-code exists over $\mathbb{F}_q$?

For very small n, k, d, these numbers have been computed exactly. In general, one has to rely on bounds.

For small values of n, k, d, the optimal parameters can be computed by computer [50], but in general one needs to rely on bounds. The three most common (and usually strongest) bounds are as follows.

Theorem 4.1.9 (Hamming bound). *For any $[n, k, d]$-code over $\mathbb{F}_q$, one has $q^k \sum_{i=0}^{\lfloor \frac{d-1}{2} \rfloor} \binom{n}{i} (q-1)^i \leq q^n$.*

Theorem 4.1.10 (MDS bound). *For any $[n, k, d]$-code over $\mathbb{F}_q$, one has $n + 1 \geq k + d$.*

Theorem 4.1.11 (Griesmer bound [127, 51]). *For any $[n, k, d]$-code over $\mathbb{F}_q$, one has $n \geq \sum_{i=0}^{k-1} \left\lceil \frac{d}{q^i} \right\rceil$.*

Definition 4.1.12. Codes meeting the Hamming, MDS and Griesmer bounds are called *perfect*, *MDS* and *Griesmer* codes.

For perfect linear codes over $\mathbb{F}_q$, a full classification is known. A series of papers by Van Lint [143, 144, 145], Tietäväinen [137], Zinoviev and Leontiev [152] can be summarized in the following theorem.

Theorem 4.1.13. *Let C be a perfect $[n, k, d]$-code over $\mathbb{F}_q$. Then C is either:*

- *the empty code ($k = 0$) or the entire space ($k = n$);*
- *a binary repetition code of odd length ($k = 1, n = d$);*
- *a Hamming code;*
- *a Golay code.*

For MDS codes, a strong characterization theorem exists.

Theorem 4.1.14. *Let G be a $k \times n$ generator matrix of a code C. Let S be the set of points of $\mathrm{PG}(k-1,q)$ defined by the columns of G. Then C is MDS if and only if S is a set of n points, no $k-1$ of which lie in the same hyperplane.*

Such sets are called *arcs* and the following result and conjecture have been made on their existence.

Theorem 4.1.15 ([20])**.** *If $k \leq q+1$ then such sets exist if and only if $n \leq k+1$.*

Conjecture 4.1.16 (The MDS conjecture [120])**.** *If $k \geq q+2$ then such sets exist if and only if either $n \leq q+1$, or ($n = q+2$ and q is even and $k \in \{3, q-1\}$).*

For Griesmer codes, the characterization theorem is somewhat more complicated.

Theorem 4.1.17 ([53, 55])**.** *There exists a bijective correspondence between the set of all non-equivalent $[n,k,d]_q$-codes meeting the Griesmer bound, and the set of*

$$\left(\sum_{i=0}^{k-2} \mu_i v_{i+1}, \sum_{i=0}^{k-2} \mu_i v_i\right)$$

-minihypers in $\mathrm{PG}(k-1,q)$ with each $\mu_i \leq q-1$.

Griesmer codes however have a property that makes them well worth the effort, called *divisibility*.

Theorem 4.1.18 ([148])**.** *Let p be a prime and let C be an $[n,k,d]_p$ Griesmer code. If $p^e | d$ for some $e \geq 1$, then all code words of C have Hamming weight divisible by p^e.*

Conjecture 4.1.19 ([58])**.** *Let $q = p^h$ and let C be an $[n,k,d]_q$ Griesmer code. If $p^e | d$ for some $e \geq h$, then all code words of C have Hamming weight divisible by p^{e+1-h}.*

Theorem 4.1.20 ([55])**.** *Writing $d = \theta q^{k-1} - \sum_{i=0}^{k-2} \mu_i q^i$ for a Griesmer code, with $\mu_i \in \{0, 1, \ldots, q-1\}$, the corresponding minihyper has parameters*

$$\left(\sum_{i=0}^{k-2} \mu_i v_{i+1}, \sum_{i=0}^{k-2} \mu_i v_i\right).$$

To make a Griesmer code as divisible as possible, one should let $\mu_0 = \mu_1 = \ldots = \mu_{k-3} = 0$ and $\mu_{k-2} = x \neq 0$. Since the characterization of minihypers with the above parameters is equivalent to the characterization of the corresponding Griesmer codes (cf. [85] and the references therein), this means we end up again with (xv_t, xv_{t-1})-minihypers in $\mathrm{PG}(t,q)$.

4.2 A new way of looking: rational sums of hyperplanes

Given that two natural problems in finite geometry and coding theory coincide in the same class of structures, it is of no surprise that this structure has been studied extensively before [10, 57, 58, 84]. However, one essential property of these structures has been long overlooked. In this section, we will take a closer look at the algebraic structure of these (xv_t, xv_{t-1})-minihypers.

Definition 4.2.1. The *characteristic function* of a set $\mathcal{Q} \subseteq \mathcal{P}$ is denoted by

$$\chi_{\mathcal{Q}}(x) = \begin{cases} 1 & \text{for } x \in \mathcal{Q}, \\ 0 & \text{for } x \notin \mathcal{Q}. \end{cases}$$

Remark 4.2.2. Every multiset $\mathfrak{K}$ in $\mathrm{PG}(t,q)$ can be uniquely interpreted as a vector $w \in \mathbb{Q}^{\mathcal{P}}$ as $w = (\mathfrak{K}(u))_{u\in\mathcal{P}}$. There is a natural bijective correspondence between the set of all multisets in $\mathrm{PG}(t,q)$ and the subset $\mathbb{N}_0^{\mathcal{P}} \subset \mathbb{Q}^{\mathcal{P}}$.

Addition (often referred to as *sum* or *weighted sum*) and scalar multiplication of multisets can be defined by

$$(\mathfrak{K}_1 + \mathfrak{K}_2)(x) = \mathfrak{K}_1(x) + \mathfrak{K}_2(x), \quad (c\mathfrak{K})(x) = c\mathfrak{K}(x)$$

which is just the standard addition and multiplication for their corresponding vectors. Clearly, the sum of two minihypers with parameters (f_1, m_1) and (f_2, m_2) is an (f, m)-minihyper with $f = f_1 + f_2$ and $m \geq m_1 + m_2$.

The intersection of a multiset $\mathfrak{K}$ and a set S is defined as follows:

$$(\mathfrak{K} \cap S)(x) = \begin{cases} \mathfrak{K}(x) & \text{if } x \in S. \\ 0 & \text{if } x \notin S. \end{cases}$$

Definition 4.2.3. An (f, m)-minihyper $\mathfrak{F}$ is called *indecomposable* if it cannot be represented as the sum of two nonempty minihypers with parameters (f_1, m_1) and (f_2, m_2), respectively, for which $m = m_1 + m_2$ and $f = f_1 + f_2$.

Clearly, an (f, m)-minihyper which is not proper and which is not the point set of $\mathrm{PG}(t,q)$, is decomposable: it can be represented as the sum of a (v_{t+1}, v_t)-minihyper (namely the entire space $\mathrm{PG}(t,q)$) and an $(f - v_{t+1}, m - v_t)$-minihyper.

Definition 4.2.4. Let X be a finite set of size v (which we call the *points*) and let $\mathcal{B}$ be a family of k-element subsets of X (which we call the *blocks*) in which every unordered pair of elements of X is contained in exactly λ blocks of $\mathcal{B}$. Then $(X, \mathcal{B})$ is called a *balanced incomplete* $2-(v,k,\lambda)$ *block design*. It is easy to see that each point of X is contained in $r = \lambda(v-1)/(k-1)$ blocks of $\mathcal{B}$. Letting $b = |\mathcal{B}|$, an easy double-counting argument yields that $vr = bk$. If $b = v$, the design is called *symmetric*.

Definition 4.2.5. Let $\mathcal{D} = (X, \mathcal{B})$ be a $2-(v,k,\lambda)$-design and fix any ordering of the points and of the blocks. The *incidence matrix* of $\mathcal{D}$ is the $b \times v$ matrix $A = (a_{ij})$ defined by

$$a_{ij} = \begin{cases} 1 & \text{if the } j\text{th point is contained in the } i\text{th block}, \\ 0 & \text{otherwise.} \end{cases}$$

Hence, A can be interpreted as an isomorphism between $\mathbb{Q}^{\mathcal{P}}$ and $\mathbb{Q}^{\mathcal{H}}$.

Remark 4.2.6. It is easily checked that $A^T A = (r-\lambda)I + \lambda J$, where I and J are the unit matrix of order v and the all-one matrix of order v, respectively. Hence $\det A^T A = rk(r-\lambda)^{v-1}$ over $\mathbb{Q}$. Hence, when $r \neq \lambda$ (or, equivalently, when $v \neq k$), $A^T A$ is nonsingular and hence A is nonsingular. In $\mathrm{PG}(t,q)$, $\mathcal{D} = (\mathcal{P}, \mathcal{B})$, with $\mathcal{B}$ the set of hyperplanes, is a symmetric $2-(v_{t+1}, v_t, v_{t-1})$-design. For proofs of these statements and an in-depth introduction to designs (and their links with finite geometry), we refer to [5, 11, 22].

The remainder of this chapter is structured as follows. In the remainder of this section, we present a new characterization of proper (xv_t, xv_{t-1})-minihypers in $\mathrm{PG}(t,q)$ as rational sums of hyperplanes. We thereby generalize a result by Landjev and Storme [84, Theorem 5]. In Section 4.3, we extend and improve several key results that have appeared on the special case $n = 2$ [58, 84]. Most notably, we prove a strong modular result and a useful inequality between x, q and c (c is defined in Theorem 4.2.11). Finally, in Section 4.4, we establish a new connection between the code words of certain geometrically defined codes and indecomposable minihypers. We exploit this new connection to present a new non-trivial construction for (xv_t, xv_{t-1})-minihypers in $\mathrm{PG}(t,q)$.

Lemma 4.2.7. *Let $\mathfrak{K}$ be an arbitrary multiset in $\mathrm{PG}(t,q)$, $q = p^h$. Then its incidence vector w can uniquely be written as a linear combination over $\mathbb{Q}$ of incidence vectors of hyperplanes: $w = \sum_{H\in\mathcal{H}} r_H\chi_H$ with $r_H \in \mathbb{Q}$.*

Proof. Let A be an incidence matrix of the points and hyperplanes of $\mathrm{PG}(t,q)$. By Remark 4.2.6, A is invertible. Hence, the rows of A form a $\mathbb{Q}$-basis for the vector space $\mathbb{Q}^{\mathcal{P}}$ and for any $w \in \mathbb{Q}^{\mathcal{P}}$, one can find a unique collection of rational coefficients $\{r_H\}_{H\in\mathcal{H}}$ such that $w = \sum_{H\in\mathcal{H}} r_H\chi_H$. Note that, with $r = (r_H)_{H\in\mathcal{H}}$, $w = rA$. □

Notation 4.2.8. From now on, if $\mathfrak{F}$ is an (xv_t, xv_{t-1})-minihyper in $\mathrm{PG}(t,q)$, we will denote by $r_H(\mathfrak{F})$ the coefficient r_H associated to the hyperplane H in the rational sum obtained in Theorem 4.2.9. If the minihyper $\mathfrak{F}$ is clear from the context, we will simply write r_H. Since the minihyper can be written as a rational sum in a unique way, this will often be the case.

Theorem 4.2.9. *Let $\mathfrak{K}$ be a multiset in $\mathrm{PG}(t,q)$ and let $w = \sum_{H\in\mathcal{H}} r_H\chi_H$ be its incidence vector. Then $r_H \geq 0$ for each $H \in \mathcal{H}$ if and only if w is an (f,m)-minihyper with $m \geq \frac{v_{t-1}}{v_t} f$. If in addition, $\mathfrak{K}$ is proper, then $r_H \geq 0$ for each $H \in \mathcal{H}$ if and only if $\mathfrak{K}$ is an (xv_t, xv_{t-1})-minihyper, with $x = \sum_{H\in\mathcal{H}} r_H \in \mathbb{N}_0$.*

Proof. Since A is invertible and $JA = rJ$, we may write $A^TA = (r-\lambda)I + \lambda J$ as $(A^T - \frac{\lambda}{r}J)A = (r-\lambda)I$, which yields $A^{-1} = \frac{1}{r-\lambda}(A^T - \frac{\lambda}{r}J)$.

Let now $w \in \mathbb{Q}^{\mathcal{P}}$ be the incidence vector of any multiset $\mathfrak{K}$ (as defined in Remark 4.2.2). Then $w = (wA^{-1})A$, which yields an explicit form for the rational coefficients:

$$w = \sum_{H\in\mathcal{H}} (wA^{-1})_H\chi_H,$$

and this form is unique by Lemma 4.2.7.

Hence, we want to determine when each of the elements of $wA^{-1} \in \mathbb{Q}^{\mathcal{H}}$ is non-negative. From the explicit form derived above, $wA^{-1} = \frac{1}{r-\lambda}\left(wA - \frac{\lambda}{r}Jw\right)$. However, Jw is a vector with each of its entries equal to the total size of the multiset, f. Hence, we need $(wA^T)_H \geq \frac{\lambda}{r}f$ for each $H \in \mathcal{H}$.

Now the element $(wA^T)_H$ represents the total multiplicity of the hyperplane H, $\mathfrak{K}(H)$, and hence this inequality is equivalent to saying that $\mathfrak{K}(H) \geq \frac{\lambda}{r}f$ for each $H \in \mathcal{H}$. In other words,

this is true if and only if w is the incidence vector of an $(f, m; t, q)$-minihyper with $m \geq \frac{\lambda}{r} f$. This proves the first statement.

If the multiset $\mathfrak{K}$ is proper, then there is a point u with $\mathfrak{K}(u) = 0$. We define a new multiset $\mathfrak{K}'$ as follows: $\mathfrak{K}' = \sum_{H \ni u} \mathfrak{K} \cap H$. Then the total multiplicity of this new multiset is $f\lambda$, since for each point of $\mathfrak{K}$ there are λ hyperplanes through u and through this point. On the other hand, this number is at least m times the number r of such hyperplanes, since each hyperplane contains at least m points. Hence, we also have $m \leq \frac{\lambda}{r} f$ and thus $m = \frac{\lambda}{r} f$. However, $\gcd(\lambda, r) = 1$, so r divides f, and thus $f = xr$ for some positive integer x. Hence, $m = x\lambda$ and since $r = \frac{q^t-1}{q-1}$ and $\lambda = \frac{q^{t-1}-1}{q-1}$, we have $f = x\left(\frac{q^t-1}{q-1}\right) = xv_t$ and $m = x\left(\frac{q^{t-1}-1}{q-1}\right) = xv_{t-1}$. □

Remark 4.2.10. The last part of the proof of Theorem 4.2.9 shows that for every proper (f, m)-minihyper in $\mathrm{PG}(t, q)$, one has $\frac{f}{m} \geq \frac{v_t}{v_{t-1}}$. This provides an additional motivation for the study of (xv_t, xv_{t-1})-minihypers in $\mathrm{PG}(t, q)$.

Theorem 4.2.11. *For any proper (xv_t, xv_{t-1})-minihyper $\mathfrak{F} = \sum_{H \in \mathcal{H}} r_H \chi_H$ in $\mathrm{PG}(t, q)$, the smallest positive integer c for which $cr_H \in \mathbb{N}_0$ for all $H \in \mathcal{H}$, is a power of p and a divisor of q^{t-1}.*

Proof. From the proof of Theorem 4.2.9, we know that the coefficients r_H are given by $(wA^{-1})_H = \frac{1}{r-\lambda}\left((wA^T)_H - \frac{\lambda f}{r}\right)$. Since Jw is a vector with all its entries equal to $f = rx$, $\frac{\lambda}{r} Jw$ is an integer vector which only consists of entries λx. Since the entries of wA^T are also integers, $wA^T - \frac{\lambda}{r} Jw$ is an integer vector, and $(r - \lambda)wA^{-1}$ only contains integer entries.

Since $r - \lambda = q^{t-1}$, the smallest positive integer c for which $cr_H \in \mathbb{N}_0$ for all $H \in \mathcal{H}$, is a divisor of q^{t-1}, and hence it is indeed a power of p. □

Note that $c = 1$ corresponds to the minihyper being a weighted sum of hyperplanes.

Notation 4.2.12. Similar to Remark 4.2.8, we will write $c(\mathfrak{F})$ for the integer c from Theorem 4.2.11. If the minihyper $\mathfrak{F}$ is clear from the context, we will simply write c.

Remark 4.2.13. A proper (xv_t, xv_{t-1})-minihyper in $\mathrm{PG}(t, q)$ (with $x > 0$) cannot be decomposed into a hyperplane and an $((x-1)v_t, (x-1)v_{t-1})$-minihyper if and only if $r_\pi < 1$ for each hyperplane π. In this case, we call the minihyper *hyperplane-indecomposable*. For $x \leq q$, we will see in Section 4.3 that hyperplane-indecomposability is equivalent to indecomposability.

4.3 Generalizations of previous results

In this section, we will apply Theorem 4.2.9 to generalize and improve several key results from [58] and [84]. In what follows, we let $q = p^h$ with p prime; this defines p and h.

R. Hill and H.N. Ward [58] proved the following modular result via polynomial techniques for $t = 2$. This was extended to $t > 2$ in [57, Theorem 4.6], using similar techniques.

Theorem 4.3.1. *Let $\mathfrak{F}$ be an (xv_t, xv_{t-1})-minihyper in $\mathrm{PG}(t,q)$, with $x \leq q - p^g$ for some nonnegative integer g. Then $\mathfrak{F}(\pi) \equiv xv_{t-1} \pmod{p^{g+1}q^{t-2}}$ for every hyperplane π in $\mathrm{PG}(t,q)$.*

Using Theorem 4.2.9, we can present a sharper version of this modular result. We begin with an easy counting lemma. We recall that if $\mathfrak{F} = \sum_{H\in\mathcal{H}} r_H \chi_H$ is an (xv_t, xv_{t-1})-minihyper in $\mathrm{PG}(t,q)$, then $\sum_{H\in\mathcal{H}} r_H = x$. We also recall that whenever we write r_H or c, this has to be interpreted as in Remark 4.2.8.

Lemma 4.3.2. *Let $\mathfrak{F}$ be an (xv_t, xv_{t-1})-minihyper in $\mathrm{PG}(t,q)$. Then a hyperplane π with rational coefficient r_π has multiplicity $\mathfrak{F}(\pi) = r_\pi q^{t-1} + xv_{t-1}$.*

Proof. The hyperplane π contributes r_π to the multiplicity of each point in π, and hence contributes $r_\pi v_t$ to the total multiplicity of π. Every other hyperplane π' intersects π in $\lambda = v_{t-1}$ points, hence contributing $r_{\pi'} v_{t-1}$ to $\mathfrak{F}(\pi)$. Since the sum of all rational coefficients is x, this yields a total multiplicity in π of $r_\pi v_t + (x - r_\pi)v_{t-1}$. Since $v_t = q^{t-1} + v_{t-1}$, this proves the statement. □

From this it follows that for any s-dimensional subspace π, one has

$$\mathfrak{F}(\pi) = xv_s + q^s \sum_{H\supseteq\pi, H\in\mathcal{H}} r_H.$$

Moreover, if π contains a point u with multiplicity 0, then all hyperplanes through u (and hence all hyperplanes through π) have their rational coefficient equal to 0. Hence, in this case $\mathfrak{F}(\pi) = xv_s$.

Theorem 4.3.3. *Let $\mathfrak{F}$ be a proper (xv_t, xv_{t-1})-minihyper in $\mathrm{PG}(t,q)$. Then $\mathfrak{F}(\pi) \equiv xv_{t-1} \pmod{\frac{q^{t-1}}{c}}$ for every hyperplane π in $\mathrm{PG}(t,q)$. Moreover, if $x \leq q - p^g$, then p^{g+1} divides $\frac{q}{c}$, making this result stronger than Theorem 4.3.1.*

Proof. Let π be an arbitrary hyperplane and let r_π be its rational coefficient. Then $\mathfrak{F}(\pi) = r_\pi q^{t-1} + xv_{t-1}$ by Lemma 4.3.2. Since the denominator of r_π is a divisor of c, the product $q^{t-1}r_\pi$ is an integer multiple of $\frac{q^{t-1}}{c}$, and hence the first part of the statement follows.

For the second part, it is sufficient to recall that c is the smallest integer such that for all r_H, $cr_H \in \mathbb{N}_0$. By Theorem 4.3.1, $r_\pi q^{t-1}$ is divisible by $p^{g+1}q^{t-2}$ and hence $\left(\frac{q}{p^{g+1}}\right) r_\pi$ is an integer. Since π was arbitrary, and since c is the smallest positive integer for which cr_π is an integer for all π, it follows that $c \leq \frac{q}{p^{g+1}}$. Since c is a power of p by Theorem 4.2.11, it follows that p^{g+1} divides $\frac{q}{c}$. □

In Theorem 4.3.3, we work modulo $\frac{q^{t-1}}{c} = \frac{q}{c}q^{t-2}$. In Theorem 4.3.1, the result is only valid modulo $p^{g+1}q^{t-2}$. Since we just have just proven that p^{g+1} divides $\frac{q}{c}$, Theorem 4.3.3 is a generalization of Theorem 4.3.1.

Corollary 4.3.4. *Let $\mathfrak{F}$ be a nonempty (xv_t, xv_{t-1})-minihyper in $\mathrm{PG}(t,q)$. Then $x > q - \frac{q}{c}$. In other words: if $x \leq q - \frac{q}{c_0}$ for some positive integer c_0, then $c < c_0$.*

Proof. If $x \geq q$, then the statement is trivially fulfilled. Otherwise, let g be the largest nonnegative integer for which $x \leq q - p^g$. By this maximality assumption, $x > q - p^{g+1}$. Since p^{g+1} divides $\frac{q}{c}$, it indeed follows that $x > q - \frac{q}{c}$. □

As a special case of Corollary 4.3.4, we get the following corollary.

Corollary 4.3.5. *For $x \leq q - \frac{q}{p}$ (and hence for $x < q$ when $q = p$), we have $c = 1$. Hence, if $x \leq q - q/p$ then any (xv_t, xv_{t-1})-minihyper in $\mathrm{PG}(t, q)$ is a sum of x hyperplanes.*

This special case was proven earlier for $t = 2$ in [58, Theorem 20] and for general t in [57, Corollary 4.8]. The sharpness of the bound in Corollary 4.3.5 had not yet been demonstrated. In Section 4.4, we will show the sharpness of this bound. This family of examples will show the sharpness of the bound in Corollary 4.3.4 in general when $c = p^e$ with $e|h$ (with $q = p^h$).

Corollary 4.3.6. *If $x \leq 2q - 2\frac{q}{p} + 1$, then a proper (xv_t, xv_{t-1})-minihyper is decomposable if and only if it is hyperplane-decomposable.*

Proof. Assume by contraposition that there exists a proper decomposable, but hyperplane-indecomposable (xv_t, xv_{t-1})-minihyper $\mathfrak{F}$ with $x \leq 2q - 2\frac{q}{p} + 1$. Since it is proper and decomposable, it can be written as $\mathfrak{F} = \mathfrak{F}_1 + \mathfrak{F}_2$, where $\mathfrak{F}_1$ is a nonempty (x_1v_t, x_1v_{t-1})-minihyper and $\mathfrak{F}_2$ is a nonempty (x_2v_t, x_2v_{t-1})-minihyper, and $x_1 + x_2 = x$. Since $x \leq 2q - 2\frac{q}{p} + 1$, it follows that $\min(x_1, x_2) \leq q - \frac{q}{p}$, and, by Corollary 4.3.5, this minihyper is a sum of hyperplanes. Hence, we can subtract any such hyperplane from $\mathfrak{F}$ and end up with an $((x-1)v_t, (x-1)v_{t-1})$-minihyper, contradicting the assumption that $\mathfrak{F}$ is hyperplane-indecomposable. □

Remark 4.3.7. Corollary 4.3.5 and its sharpness determine the smallest x for which there is a (hyperplane-)indecomposable (xv_t, xv_{t-1})-minihyper in $\mathrm{PG}(t, q)$.

An upper bound on the largest integer x for which there exists a proper, hyperplane-indecomposable (xv_t, xv_{t-1})-minihyper, can be derived as follows. Fix a point u with multiplicity 0 in this minihyper. Since we assume that $\mathfrak{F}$ is hyperplane-indecomposable, $r_H < 1$ for all hyperplanes H. Since $cr_H \in \mathbb{N}_0$ and since c is a divisor of q^{t-1}, by Theorem 4.2.11, this yields $r_H \leq 1 - \frac{1}{c} \leq 1 - \frac{1}{q^{t-1}}$. Hence,

$$x = \sum_{H \ni u} r_H + \sum_{H \not\ni u} r_H = 0 + \sum_{H \not\ni u} r_H \leq \sum_{H \not\ni u} \left(1 - \frac{1}{q^{t-1}}\right) = q^t \left(1 - \frac{1}{q^{t-1}}\right) = q^t - q,$$

with equality if and only if all hyperplanes not through u have $r_H = 1 - \frac{1}{q^{t-1}}$. And indeed, this equality can occur; in that case $\mathfrak{F}$ is $q^{t-1} - 1$ times the setwise complement of u in $\mathrm{PG}(t, q)$, since each point different from u lies on q^{t-1} hyperplanes not containing u.

The largest x for which such a proper indecomposable minihyper exists is not known, not even for $t = 2$. A generalization of the result by Landjev and Storme [84] on the case $t = 2$ follows straightforwardly from the techniques in this section; it is presented in Theorem 4.3.8. We however believe that this bound is not sharp at all.

Theorem 4.3.8. *Let $\mathfrak{F}$ be a proper indecomposable (xv_t, xv_{t-1})-minihyper which is not the setwise complement of a point. Then $x \leq q^t - 2q + \frac{q}{p} - 1$ and the multiplicity of any point in $\mathfrak{F}$ is at most $q^{t-1} - 1$.*

Proof. Assume that $\mathfrak{F}$ is a proper indecomposable (xv_t, xv_{t-1})-minihyper in $\mathrm{PG}(t, q)$, and let u be a point of multiplicity 0. Hence, $r_H = 0$ for all hyperplanes H through u. Since we assume that $\mathfrak{F}$ is indecomposable, it is also hyperplane-indecomposable, which means that $r_H < 1$ for all hyperplanes. Since $cr_H \in \mathbb{N}_0$ and c is a divisor of q^{t-1} by Theorem 4.2.11, this yields $r_H \leq 1 - \frac{1}{c} \leq 1 - \frac{1}{q^{t-1}}$.

Let u' be an arbitrary point different from u. From the fact that $r_H \leq 1 - \frac{1}{q^{t-1}} = \frac{q^{t-1}-1}{q^{t-1}}$ and the fact that there are only q^{t-1} hyperplanes through u' and not through u, it follows that the multiplicity of this point u' is at most $q^{t-1} - 1$. Since u' was arbitrary, this yields the second claim.

Now, we revisit the switching construction from [84] with respect to u. In our terminology, it reduces to the natural substitution

$$\psi : \begin{cases} r_H \mapsto r_H (= 0) & \text{if } H \ni u, \\ r_H \mapsto 1 - \frac{1}{q^{t-1}} - r_H & \text{if } H \not\ni u. \end{cases}$$

Clearly, since $0 \leq r_H(\mathfrak{F}) \leq 1 - \frac{1}{q^{t-1}}$, the same holds for $r_H(\psi(\mathfrak{F}))$, and since each point different from u lies on $v_t - v_{t-1} = q^{t-1}$ hyperplanes not through u, the fact that each point has an integer multiplicity is also preserved under ψ. Hence, $\psi(\mathfrak{F})$ is a (yv_t, yv_{t-1})-minihyper in $\mathrm{PG}(t, q)$ with $y = q^t \left(1 - \frac{1}{q^{t-1}}\right) - x$.

Since $\mathfrak{F}$ is not the setwise complement of u, $\psi(\mathfrak{F})$ is nonempty. Moreover, since $r_H(\psi(\mathfrak{F})) < 1$, the minihyper $\psi(\mathfrak{F})$ is hyperplane-indecomposable, which means $c > 1$ and hence $c \geq p$. By Corollary 4.3.4, $y \geq q - \frac{q}{p} + 1$, which means that

$$x = (q^t - q) - y \leq (q^t - q) - (q - \frac{q}{p} + 1) = q^t - 2q + \frac{q}{p} - 1. \qquad \square$$

Corollary 4.3.9. *No hyperplane-indecomposable (xv_t, xv_{t-1})-minihyper in $\mathrm{PG}(t, q)$ exists for any x satisfying $q^t - 2q + \frac{q}{p} - 1 < x < q^t - q$.*

4.4 A surprising new link with coding theory

In this section, we will establish a new correspondence between hyperplane-indecomposable (xv_t, xv_{t-1})-minihypers in $\mathrm{PG}(t, q)$ and the dual projective space code over the ring $\mathbb{Z}_c$, with c the number described in Theorem 4.2.11. Let $\mathbb{Z}_c$ be the ring of integers modulo c, i.e. $\mathbb{Z}_c = (\{0, 1, 2, \ldots, c-1\}, +_c, \cdot_c)$, where $a +_c b$ and $a \cdot_c b$ denote the remainder of respectively $a + b$ and $a \cdot b$ after division by c. Note that the set $\{0, 1, 2, \ldots, c-1\}$ is a set of integers, a subset of $\mathbb{Z}$. If $c = p$, then $\mathbb{Z}_c$ is a field, isomorphic to $\mathbb{F}_p$.

Let H be the hyperplane-by-point incidence matrix of $\mathrm{PG}(t,q)$. Let $C_c^\perp(t,q)$ be the linear $\mathbb{Z}_c$-code defined by H as a parity check matrix, where the positions of the code correspond to the hyperplanes:

$$C_c^\perp(t,q) = \{z = (z_H)_{H\in\mathcal{H}} \in \mathbb{Z}_{\mathcal{H}} : zH = \bar{0}\};$$

hereby, the matrix multiplication is done over $\mathbb{Z}_c$. For this code $C_c^\perp(t,q)$, we define a new weight function $\mathrm{wt}(z) = \sum_{H\in\mathcal{H}} z_H$, where z_H is interpreted as an integer in $\{0,1,\ldots,c-1\}$ and summation is done over $\mathbb{Z}$. In the special case that $c = p$, $C_c^\perp(t,q)$ is equivalent to the commonly studied projective space code of points and hyperplanes.

Geometrically, code words of C correspond to multisets of hyperplanes in $\mathrm{PG}(t,q)$, with hyperplane multiplicities in the set $\{0,1,\ldots,c-1\}$, such that for each point r we have $\sum_{H\ni r} z_H \equiv 0 \pmod c$. We will interpret z_H in the proof of Theorem 4.4.1 as $z_H = c\cdot r_H$, where r_H are (as always) the rational coefficients from Lemma 4.2.7 for the minihyper $\mathfrak{F}$.

Theorem 4.4.1. *There is a natural bijective correspondence between the code words*

$$z = (z_H)_{H\in\mathcal{H}} \in C_{c_0}^\perp(t,q)$$

and the hyperplane-indecomposable (xv_t, xv_{t-1})-minihypers $\sum_{H\in\mathcal{H}} r_H\chi_H$ with $c = c_0$; this correspondence is given by $z_H = c\cdot r_H$.

Proof. First assume that we have a code word $z = (z_H)_{H\in\mathcal{H}} \in C_{c_0}^\perp(t,q)$. By definition of the code $C_{c_0}^\perp(t,q)$, we have $\sum_{H\ni u} z_H \equiv 0 \pmod{c_0}$ for each point u. Hence, for each point u, the multiplicity $\frac{1}{c_0}\sum_{H\ni u} z_H$ of the point u is an integer. Since we also have that each weight is nonnegative (as $z_H \in \{0,1,\ldots,c_0-1\}$), it follows from Theorem 4.2.9 that $\mathfrak{F} := \sum_{H\in\mathcal{H}} \frac{z_H}{c_0}\chi_H$ is an (xv_t, xv_{t-1})-minihyper for $x = \sum_{H\in\mathcal{H}} \frac{z_H}{c_0}$. Since for each $H\in\mathcal{H}$, $z_H \in \{0,1,\ldots,c-1\}$, one has $\frac{z_H}{c_0} < 1$, and hence $\mathfrak{F}$ is a hyperplane-indecomposable (xv_t, xv_{t-1})-minihyper.

For the other direction, assume that we have a hyperplane-indecomposable (xv_t, xv_{t-1})-minihyper $\mathfrak{F}$ in $\mathrm{PG}(t,q)$. By Theorem 4.2.9, $\mathfrak{F} = \frac{1}{c}\sum_{H\in\mathcal{H}} r_H\chi_H$. By Remark 4.2.13, $r_H < 1$ for each $H\in\mathcal{H}$. Let $z_H = cr_H$, then the multiplicity at each point u is $\frac{1}{c}\sum_{H\ni u} z_H \in \mathbb{N}_0$. This implies that $\sum_{H\ni u} z_H \equiv 0 \pmod c$, which means that $z = (z_H)_{H\in\mathcal{H}}$ is a code word of $C_c^\perp(t,q)$. □

Theorem 4.4.1 can be used in the construction of non-trivial (xv_t, xv_{t-1})-minihypers. Ball's construction, mentioned in [84], can be derived as a special case of this construction. The key is to dualize the setting: we start with an arbitrary multiset of points, dualize it to have an arbitrary multiset of hyperplanes, and take a rational sum of them to obtain a minihyper. This yields the following interesting constructions.

Lemma 4.4.2 (Ball's construction)**.** *Let B be a set of points in $\mathrm{PG}(t,q)$ and let e be the largest nonnegative integer such that B meets each hyperplane in 0 modulo p^e points. Then there exists a $\left(\frac{|B|}{p^e}v_t, \frac{|B|}{p^e}v_{t-1}\right)$-minihyper in $\mathrm{PG}(t,q)$ with $c = p^e$.*

Proof. Let B' be the dual set of hyperplanes of the points in B. By the self-duality of $\mathrm{PG}(t,q)$, each point is contained in 0 modulo p^e hyperplanes of B'. Associating a coefficient $r_H = \frac{1}{p^e}$ to

each of these hyperplanes (and 0 to all other hyperplanes) yields a $\left(\frac{|B|}{p^e}v_t, \frac{|B|}{p^e}v_{t-1}\right)$-minihyper. By construction, $c|p^e$, and by the maximality of e, it follows that $c = p^e$. □

More interestingly, we can also utilize 1 modulo p^e sets to construct new examples, as the following lemma demonstrates.

Lemma 4.4.3. *Let A and B be sets of points in* $\mathrm{PG}(t,q)$ *and let e be the largest nonnegative integer such that A and B both meet each hyperplane in 1 modulo p^e points. Then for any $\lambda \in \{1, 2, \ldots, p^e - 1\}$ there exists an (xv_t, xv_{t-1})-minihyper $\mathfrak{F}$ in* $\mathrm{PG}(t,q)$ *with $c = p^e$ and $x = |B \setminus A| + \lambda\frac{|A|-|B|}{p^e}$.*

Proof. Since A and B represent point sets, we can consider their associated dual sets A' and B' of hyperplanes. Since A and B intersect each hyperplane in 1 modulo p^e points, their differences $A \setminus B$ and $B \setminus A$ intersect each hyperplane in 0 modulo p^e points. Therefore if we add λ times the incidence vector of each hyperplane in $A' \setminus B'$ and $p^e - \lambda$ times the incidence vector of each hyperplane in $B' \setminus A'$, the multiplicity of each point will be divisible by p^e. Hence, dividing this by p^e yields a minihyper with c a divisor of p^e. By the maximality of e, it follows that $c = p^e$.

The total weight in the multiset before dividing by p^e, is

$$\lambda|A \setminus B| + (p^e - \lambda)|B \setminus A| = p^e|B \setminus A| + \lambda(|A| - |B|).$$

Dividing out p^e yields $x = |B \setminus A| + \lambda\frac{|A|-|B|}{p^e}$ as claimed. □

Several examples of 1 modulo p^e sets (with $e \geq 1$) are known: i-dimensional subspaces with $i \geq 1$, Baer subgeometries, unitals and Hermitian varieties, linear blocking sets and many, many other commonly studied structures in finite geometries. With Lemma 4.4.3, all of them can be used to obtain structurally new examples. In particular, we were able to construct a minimal nontrivial example, i.e. a minihyper with $x = q - \frac{q}{p} + 1$ which is not a sum of x hyperplanes. This shows the sharpness of Corollary 4.3.5 and can also be used to show the sharpness of Theorem 4.3.8. In some cases, the construction can also be used to show the sharpness of Corollary 4.3.4.

Theorem 4.4.4. *For each divisor e of h (where $q = p^h$), there exists an (xv_t, xv_{t-1})-minihyper in* $\mathrm{PG}(t,q)$ *with $x = q - \frac{q}{p^e} + 1$.*

Proof. Let $q = p^h$ and let e be a divisor of h. Let A be the line in $\mathrm{PG}(2,q)$ having $X_0 = 0$ as its equation, and let B be the set

$$B = \{(1, z, z^{p^e}) | z \in \mathbb{F}_q\} \cup \{(0, z, z^{p^e}) | z \in \mathbb{F}_q^*\}.$$

Then it is shown in [14] that $|B| = q+y$ and $|B \cap A| = y$, with $y = \frac{q-1}{p^e-1}$. Moreover, it is shown there that each line intersects B in 1 modulo p^e points. This set B is called a Rédei-type blocking set.

Applying Lemma 4.4.3 with this A and B and with $\lambda = p^e - 1$, one obtains an (xv_2, xv_1)-minihyper with $x = q - \frac{q}{p^e} + 1$ in $\mathrm{PG}(2, q)$. This proves the statement for $t = 2$.

For $t > 2$, the construction in the plane can easily be extended. Let π be a 2-dimensional subspace of $\mathrm{PG}(t, q)$ and let π' be a $(t-3)$-dimensional subspace skew to π. Let $\mathfrak{F}$ be the constructed example for $t = 2$ in the 2-dimensional space π. Now for each line L in π, let r_L be its rational coefficient in $\mathfrak{F}$ and let H_L be the hyperplane spanned by L and π'. Then $\mathfrak{F}' := \sum_{L \subset \pi} r_L \chi_{H_L}$ is a cone with π' as its vertex and $\mathfrak{F}$ as its base. Moreover, $\mathfrak{F}'$ is an (xv_t, xv_{t-1})-minihyper with $x = q - \frac{q}{p^e} + 1$ in $\mathrm{PG}(t, q)$. □

Remark 4.4.5. Let again $t = 2$ and let $q = p^2$ and $e = 1$. Repeating the construction in the proof of Theorem 4.4.4 with the same choices of A and B, but now varying $\lambda \in \{1, \ldots, p-1\}$, one obtains a spectrum result: a nontrivial (xv_2, xv_1)-minihyper for each $x \in \{q - \frac{q}{p} + 1, \ldots, q-1\}$.

Corollary 4.4.6. *The bound in Corollary 4.3.5 is sharp. When e divides h (with $c = p^e$ and $q = p^h$), the bound in Corollary 4.3.4 is also sharp.*

Proof. Consider the $((q - \frac{q}{p^e} + 1)v_t, (q - \frac{q}{p^e} + 1)v_{t-1})$-minihyper in $\mathrm{PG}(t, q)$ obtained in Theorem 4.4.4. Its rational coefficients are 0, $\frac{1}{p^e}$ and $\frac{p^e - 1}{p^e}$, and hence this minihyper has $c = p^e$. This shows the sharpness of Corollary 4.3.4 when e divides h.

For $e = 1$, this yields a $((q - \frac{q}{p} + 1)v_t, (q - \frac{q}{p} + 1)v_{t-1})$-minihyper in $\mathrm{PG}(t, q)$ which is a rational sum of hyperplanes with rational coefficients 0, $\frac{1}{p}$ and $\frac{p-1}{p}$. This minihyper is not a sum of hyperplanes (since $c = p > 1$) and has $x = q - \frac{q}{p} + 1$, showing the sharpness of Corollary 4.3.5. □

Open Problem 4.4.7. It is not known whether the bound in Corollary 4.3.4 is sharp for all c.

Finally, there is an interesting relation between Conjecture 2.2.18, and the similar problem for the following modified distance function.

Definition 4.4.8. Let $d_S(C) = \min_{c \in C^*} \sum_{H \in \mathcal{H}} c_H$ with $c \in \{0, 1, \ldots, c-1\}^{\mathcal{H}}$.

With this modified distance function, Corollary 4.3.5 immediately yields the following result.

Corollary 4.4.9. *One has $d_S(C_p^\perp(2, q)) = (q - \frac{q}{p} + 1)p$, i.e. every proper line-indecomposable (xv_2, xv_1)-minihyper with $c = p$ in $\mathrm{PG}(2, q)$, is a sum of at least $(q - \frac{q}{p} + 1)p$ lines (with coefficient $\frac{1}{p}$).*

Conjecture 2.2.18, on the other hand, can be rephrased as follows.

Conjecture 4.4.10. *One has $d_H(C_p^\perp(t, q)) = 2q - \frac{q-p}{p-1}$, i.e. every proper line-indecomposable (xv_2, xv_1)-minihyper with $c = p$ in $\mathrm{PG}(2, q)$, is a sum of at least $2q - \frac{q-p}{p-1}$* different *lines (with coefficient a multiple of $\frac{1}{p}$).*

Remark 4.4.11. The construction in the proof of Theorem 4.4.4 was inspired by the construction of the smallest known code words (in terms of Hamming weight) in the dual code $C^{\perp}_{\mathrm{PG}(2,q)}$ associated to the projective plane $\mathrm{PG}(2,q)$ [90]. These code words are conjectured to be the smallest in Hamming weight. So also here we find support for our claim that these problems are similar.

Chapter 5

Small line sets with few odd-points

In this chapter, we study small sets of lines in $\mathrm{PG}(n,q)$ and $\mathrm{AG}(n,q)$, q odd, that have a small number of odd-points. We fix a glitch in the proof of an earlier bound in the affine case, we extend the theorem to the projective case, and we attempt to classify all the sets where equality is reached. For the projective case, we obtain a full classification. For the affine case, we obtain a full classification minus one open case where there is only a characterization. The results in this chapter have been accepted for publication in Des. Codes Cryptogr. [142].

5.1 Motivation and preliminaries

Notation 5.1.1. Denote by $\mathcal{B}$ a set of lines in $\mathrm{PG}(n,q)$ or $\mathrm{AG}(n,q)$, and denote by $\mathrm{odd}(\mathcal{B})$ the set of odd-points of $\mathcal{B}$, i.e. the set of points in $\mathrm{PG}(n,q)$ or $\mathrm{AG}(n,q)$ that lie on an odd number of lines of $\mathcal{B}$.

There are several motivations to study small sets of lines in $\mathrm{PG}(n,q)$ and $\mathrm{AG}(n,q)$ that have a small number of odd-points, and in particular, sets $\mathcal{B}$ with a small value for $|\mathcal{B}| + |\,\mathrm{odd}(\mathcal{B})|$.

Firstly, it is shown in [56] that erasure-resilient codes (ERC) can be very useful in RAID setups (redundant arrays of independent disks) to allow information to survive hardware failures on large arrays of harddisks. It was shown in [106] that good codes for this purpose correspond to large minimal nonzero values of $|\mathcal{B}| + |\,\mathrm{odd}(\mathcal{B})|$ in the corresponding geometry. Even though [106] mentions $\mathrm{AG}(n,q)$ only, this part of their observations is valid for arbitrary point-line geometries.

Secondly, a common problem in geometrical coding theory is to use the incidence matrix of a finite geometry $(\mathcal{P},\mathcal{L})$ as the generator matrix (or parity check matrix) of a binary code, and study the minimum weight of the code and classify the code words of this minimum weight. This has been most commonly done for the classical spaces $\mathrm{PG}(n,q)$ and $\mathrm{AG}(n,q)$; an overview of a large part of this work can be found in [5]. Unfortunately, the binary code generated by points and lines in Desarguesian planes of odd order, is trivial for projective planes and almost trivial (codimension 1) for affine planes. One way to make a nontrivial

code out of this is to add a unit matrix to the generator matrix, i.e. $G = (I|A)$ with I the unit matrix and with A the incidence matrix of the affine or projective plane. Studying the minimum distance of this code and classifying its minimum weight code words, turns out to be equivalent to the problem of determining the minimum nonzero value of $|\mathcal{B}| + |\operatorname{odd}(\mathcal{B})|$ and classifying the sets $\mathcal{B}$ that attain this value.

Thirdly, it was shown in [106, Theorem 4.2] (after fixing a small mistake in the proof) that $|\mathcal{B}| + |\operatorname{odd}(\mathcal{B})| \geq 2q$ in $\mathrm{AG}(n, q)$, q odd, unless $\mathcal{B}$ is empty or $\mathcal{B}$ consists of a single line. It is hence a natural geometrical question which sets of lines can have equality in this bound, especially since there are many, seemingly unrelated examples. In a similar way, one can wonder if under the same condition $|\mathcal{B}| + |\operatorname{odd}(\mathcal{B})| \geq 2q + 2$ is valid in $\mathrm{PG}(n, q)$, and if one can classify the examples where equality is attained.

Finally, it is also just an interesting problem on itself, as is suggested by [8] which explicitly studies the minimum size of $|\operatorname{odd}(\mathcal{B})|$ in function of $|\mathcal{B}|$, and obtains an exact result for $|\mathcal{B}| \leq q + 1$.

When going from $\mathrm{AG}(n, q)$ to $\mathrm{PG}(n, q)$ and back, by adding or removing the hyperplane at infinity, the notation $\operatorname{odd}(\mathcal{B})$ may cause ambiguity. Therefore, we will sometimes specify the scope by adding a subscript, wherever necessary.

Notation 5.1.2. Let $\mathcal{B}$ be any line set in $\mathrm{AG}(n, q)$. By $\operatorname{odd}_{\mathrm{AG}}(\mathcal{B})$ we denote the set of affine points in $\mathrm{AG}(n, q)$ that are contained in an odd number of lines of $\mathcal{B}$.

Notation 5.1.3. Let $\mathcal{B}$ be any line set in $\mathrm{PG}(n, q)$. By $\operatorname{odd}_{\mathrm{PG}}(\mathcal{B})$ we denote the set of points in $\mathrm{PG}(n, q)$ that are contained in an odd number of lines of $\mathcal{B}$.

Note that, if we embed a line set in $\mathrm{AG}(n, q)$ into $\mathrm{PG}(n, q)$, $\operatorname{odd}_{\mathrm{PG}}(\mathcal{B})$ is the union of $\operatorname{odd}_{\mathrm{AG}}(\mathcal{B})$ and the set of all directions with an odd number of $\mathcal{B}$-lines. We will often call these directions the "points at infinity".

The following lemmata can be obtained by classical counting techniques.

Lemma 5.1.4. *Let S be a set of affine lines in $\mathrm{AG}(2, q)$, q odd. Let p be any point on the line at infinity such that an odd number of lines of S do not contain p. Then every affine line through p that is not in S, contains at least one point of $\operatorname{odd}_{\mathrm{PG}}(S) \setminus \{p\}$ and hence at least one point of $\operatorname{odd}_{\mathrm{AG}}(S)$. In particular, $|\operatorname{odd}_{\mathrm{AG}}(S)| \geq |\{L \ni p : L \notin S\}|$.*

Lemma 5.1.5. *Let S be a set of lines in $\mathrm{PG}(2, q)$, q odd. Let p be any point such that an odd number of lines of S do not contain p. Then every line through p that is not in S, contains at least one point of $\operatorname{odd}(S) \setminus \{p\}$. In particular, $|\operatorname{odd}(S)| \geq |\{L \ni p : L \notin S\}|$ if $p \notin \operatorname{odd}(S)$ and $|\operatorname{odd}(S)| \geq |\{L \ni p : L \notin S\}| + 1$ if $p \in \operatorname{odd}(S)$.*

Lemma 5.1.6. *Let $\mathcal{B}$ be any line set in $\mathrm{AG}(n, q)$ or $\mathrm{PG}(n, q)$, q odd. Let π be a two-dimensional subspace of this space, and let $\mathcal{B}_\pi = \{L \in \mathcal{B} : L \subseteq \pi\}$. Then*

$$|\mathcal{B}| + |\operatorname{odd}(\mathcal{B})| \geq |\mathcal{B}_\pi| + |\operatorname{odd}(\mathcal{B}_\pi)|.$$

If there is any line $L \in \mathcal{B} \setminus \mathcal{B}_\pi$ that contains a point of $\pi \setminus \operatorname{odd}(\mathcal{B}_\pi)$, or if there is any point of $\operatorname{odd}(\mathcal{B})$ outside of π, then the inequality is strict.

5.2 The affine case

Example 5.2.1. The following line sets $\mathcal{B}$ all have a small value for $|\mathcal{B}|+|\operatorname{odd}(\mathcal{B})|$ in $\mathrm{AG}(n,q)$, q odd.

(a) If $\mathcal{B}$ is the empty set, then $|\mathcal{B}| = 0$ and $|\operatorname{odd}(\mathcal{B})| = 0$ (sum: 0).

(b) If $\mathcal{B}$ consists of a single line, then $|\mathcal{B}| = 1$ and $|\operatorname{odd}(\mathcal{B})| = q$ (sum: $q+1$).

(c) If $\mathcal{B}$ consists of any two (different) intersecting lines, then $|\mathcal{B}| = 2$ and $|\operatorname{odd}(\mathcal{B})| = 2q-2$ (sum: $2q$).

(d) Let C be a dual conic in a projective plane $\pi \leq \mathrm{PG}(n,q)$, let L be any line of C, and embed π in $\mathrm{AG}(n,q)$ with hyperplane at infinity Π such that $\Pi \cap \pi = L$. If $\mathcal{B}$ is the embedding of $C \setminus \{L\}$, then $|\mathcal{B}| = q$ and $|\operatorname{odd}(\mathcal{B})| = q$ (sum: $2q$).

(e) Let C be a dual conic in a projective plane $\pi \leq \mathrm{PG}(n,q)$, let L be a line that contains two 1-points of C, and embed π in $\mathrm{AG}(n,q)$ with hyperplane at infinity Π such that $\Pi \cap \pi = L$. If $\mathcal{B}$ is the embedding of C, then $|\mathcal{B}| = q+1$ and $|\operatorname{odd}(\mathcal{B})| = q-1$ (sum: $2q$).

(f) Let p_1, p_2 be two points in the hyperplane at infinity. Let π be any affine plane through $\langle p_1, p_2\rangle$. If $\mathcal{B} = \{L \in \pi : p_1 \in L\} \cup \{L \in \pi : p_2 \in L\}$, then $|\mathcal{B}| = 2q$ and $|\operatorname{odd}(\mathcal{B})| = 0$ (sum: $2q$).

(g) Let p_1 be a point in the hyperplane at infinity and let p_2 be an affine point. Let π be any affine plane through $\langle p_1, p_2\rangle$. If $\mathcal{B} = (\{L \in \pi : p_1 \in L\} \cup \{L \in \pi : p_2 \in L\}) \setminus \{\langle p_1, p_2\rangle\}$, then $|\mathcal{B}| = 2q-1$ and $|\operatorname{odd}(\mathcal{B})| = 1$ (namely $\operatorname{odd}(\mathcal{B}) = \{p_2\}$) (sum: $2q$).

(h) Let Q be the line set of a hyperbolic quadric $Q^+(3,q)$ in a 3-dimensional projective subspace $\pi \leq \mathrm{PG}(n,q)$, and let π' be a 2-dimensional subspace of π that intersects Q in two lines L_1 and L_2. Embed π in $\mathrm{AG}(n,q)$ with hyperplane at infinity Π such that $\Pi \cap \pi = \pi'$. If $\mathcal{B}$ is the embedding of $Q \setminus \{L_1, L_2\}$, then $|\mathcal{B}| = 2q$ and $|\operatorname{odd}(\mathcal{B})| = 0$ (sum: $2q$).

(i) Let m be an even positive integer with $4 \leq m \leq q-1$, let k be an odd positive integer. Let S be a set of m points on the line at infinity in $\mathrm{AG}(2,q)$. Let $\mathcal{B}$ be a set consisting of k lines through each point of S, and $q-(m-1)k$ other lines such that each of the $q+1-m$ points at infinity outside of S lie on an even number of $\mathcal{B}$-lines. Then $|\mathcal{B}| = q+k$ and $|\operatorname{odd}(\mathcal{B})| = q-k$ (sum: $2q$).

Theorem 5.2.2. *Let $\mathcal{B}$ be a set of lines in $\mathrm{AG}(n,q)$, $n \geq 2$, q odd. If $|\mathcal{B}| + |\operatorname{odd}(\mathcal{B})| \leq 2q$, then $\mathcal{B}$ is one of the examples in Example 5.2.1.*

Proof. The proof consists of a detailed case study. Let $\mathcal{B}$ be a set of lines in $\mathrm{AG}(n,q)$ with $|\mathcal{B}| + |\operatorname{odd}(\mathcal{B})| \leq 2q$. In particular, this also yields $|\mathcal{B}| \leq 2q$. We distinguish the following cases.

- Assume $\mathcal{B}$ is planar, i.e. $\mathcal{B}$ is completely contained in some 2-dimensional affine subspace π of $\mathrm{AG}(n,q)$. From now on, we only consider this 2-dimensional subspace and hence we act as if $n=2$. We distinguish the following cases.
  - $|\mathcal{B}| \leq 1$. In this case, the classification is trivial, we indeed obtain Example 5.2.1(a) if $|B|=0$ or Example 5.2.1(b) if $|\mathcal{B}|=1$.
  - $|\mathcal{B}| \in [2,q]$. Each of the $|\mathcal{B}|$ lines has q affine points. Each point that is not incident with at least one other line of $\mathcal{B}$, is an odd-point. Consequently, each line of $\mathcal{B}$ contains at least $q-(|\mathcal{B}|-1)$ odd-points. Moreover, all these odd-points are different (since they lie on only one line of $\mathcal{B}$). Hence, the total number of odd-points of $\mathcal{B}$ is at least $|\mathcal{B}|(q-|\mathcal{B}|+1)$. All together, this yields
    $$|\mathcal{B}| + |\operatorname{odd}(\mathcal{B})| \geq |\mathcal{B}| + |\mathcal{B}|(q-|\mathcal{B}|+1) = |\mathcal{B}|(q+2-|\mathcal{B}|). \tag{5.1}$$
    The right hand side is a strictly concave function of $|\mathcal{B}|$, and hence obtains its minimum only on (one or both of) the end points of the considered interval $[2,q]$, so $|\mathcal{B}|=2$ or $|\mathcal{B}|=q$. In both cases, the right hand expression evaluates to $2q$. Consequently, one has $|\mathcal{B}|+|\operatorname{odd}(\mathcal{B})| \geq 2q$. Since we are given that $|\mathcal{B}|+|\operatorname{odd}(\mathcal{B})| \leq 2q$, this can only occur when $|\mathcal{B}|+|\operatorname{odd}(\mathcal{B})|=2q$, which means the right hand side must exactly reach its minimum and hence $|\mathcal{B}|=2$ or $|\mathcal{B}|=q$.
    - If $|\mathcal{B}|=2$, then we clearly have Example 5.2.1(c).
    - If $|\mathcal{B}|=q$, and equality is attained in equation (5.1), then each line of $\mathcal{B}$ needs to contain precisely $q-(|\mathcal{B}|-1)$ odd-points. This implies that $\mathcal{B}$ cannot contain three concurrent lines, otherwise one of these lines would have all of its intersections with $\mathcal{B}$-lines in at most $|\mathcal{B}|-2$ distinct points, which would result in at least $q-(|\mathcal{B}|-2)$ odd-points on that line. Hence, $\mathcal{B}$ is a dual arc of size q.
      Now, embed the affine plane in a projective plane $\mathrm{PG}(2,q)$. Then, in this projective plane, $\mathcal{B}$ is still a dual arc of size q. It is shown in [61] that any (dual) arc of size q is embedded in a (dual) arc of size $q+1$, and it is shown by Segre [119] that any (dual) arc of size $q+1$ in $\mathrm{PG}(2,q)$, q odd, is a (dual) conic by Theorem 1.1.9. This shows that $\mathcal{B}$ consists of q lines of a dual conic, and denote by L the missing line of that dual conic. In our embedding $\mathrm{PG}(2,q)$, there are two odd-points on every line of $\mathcal{B}$: one on L and one outside of L. Since $|\mathcal{B}|=q$ and $|\mathcal{B}|+|\operatorname{odd}(\mathcal{B})|=2q$, we need $|\operatorname{odd}(\mathcal{B})|=q$, and thus the only way to obtain a correct example in this case is when L is the line at infinity of our affine plane, so we end up with Example 5.2.1(d).
  - $|\mathcal{B}| \in [q+1,2q]$, $|\mathcal{B}|$ odd. Hence, $|\mathcal{B}| \in [q+2,2q-1]$. First, we will show that every parallel class of lines in our affine plane contains at least one line of $\mathcal{B}$. Assume by contradiction that this is not the case, i.e. there is a point p on the line at infinity, through which there are no lines in $\mathcal{B}$. By Lemma 5.1.4, $|\operatorname{odd}(\mathcal{B})| \geq q$; together with $|\mathcal{B}| \geq q+1$ this contradicts our assumption that $|\mathcal{B}|+|\operatorname{odd}(\mathcal{B})| \leq 2q$. Hence, there is no such point p and every point of the line at infinity, belongs to at least one line of $\mathcal{B}$.
    Since $|\mathcal{B}|$ is odd, and the number of points on the line at infinity is even (namely $q+1$), there must be some point p at infinity through which there are an even number of lines of $\mathcal{B}$ (as the sum of an even number of odd numbers would be

even, and $|\mathcal{B}|$ is odd). Let k be the number of lines of $\mathcal{B}$ through p, then Lemma 5.1.4 yields that $|\operatorname{odd}(\mathcal{B})| \geq q - k$. On the other hand, there are k lines of $\mathcal{B}$ through p, and there is at least one line of $\mathcal{B}$ through each other point of the line at infinity, resulting in $|\mathcal{B}| \geq q + k$. Since $|\mathcal{B}| + |\operatorname{odd}(\mathcal{B})| \leq 2q$, it follows that $|\mathcal{B}| + |\operatorname{odd}(\mathcal{B})| = 2q$ and there is exactly one line of $\mathcal{B}$ through every other point at infinity.

Now, embed our affine plane in a projective plane $\mathrm{PG}(2, q)$, and denote by L the line at infinity of the affine plane. Then L has q odd-points in $\mathrm{PG}(2, q)$ (namely q points with one line of $\mathcal{B}$ through each) and one k-point; we recall that k is even. Now add L to $\mathcal{B}$ and call the new set $\mathcal{B}^*$. The resulting set has $|\mathcal{B}^*| = q + k + 1$ and $|\operatorname{odd}(\mathcal{B}^*)| = q + 1 - k$: all the $q - k$ odd-points in the affine plane, plus the point p (which is now a $(k + 1)$-point).

For any point p' in $\mathrm{PG}(2, q)$, denote by $\deg p'$ the number of $\mathcal{B}^*$-lines through p'. Now, for every odd-point p' and for every $\mathcal{B}^*$-line L' through it, we can consider L' as the line at infinity of an affine plane and call $\mathcal{B}'$ the embedding of $\mathcal{B}^*$ in that plane. Then, we obtain in a similar way that $|\mathcal{B}'| \geq q + (\deg p' - 1)$ (since there are $\deg p' - 1$ lines in $\mathcal{B}'$, different from L', through p') and Lemma 5.1.4 yields $|\operatorname{odd}(\mathcal{B}')| \geq q - (\deg p' - 1)$ in this affine plane. Since p' was chosen arbitrarily, it follows that $\deg p' = k + 1$ for every odd-point of the projective line set $\mathcal{B}^*$. Since every non-$\mathcal{B}^*$-line through p' contains at least one other point of $\operatorname{odd}(\mathcal{B}^*)$, and there are only $q + 1 - k$ of them in total, it follows that every non-$\mathcal{B}^*$-line through p' contains exactly one other point of $\operatorname{odd}(\mathcal{B}^*)$, and that any $\mathcal{B}^*$-line through p' does not contain other odd-points.

Therefore, every line in $\mathcal{B}^*$ contains exactly one odd-point, and every line not in $\mathcal{B}^*$ contains 0 or 2 points of $\operatorname{odd}(\mathcal{B}^*)$. This implies that we cannot have three collinear odd-points. It also implies that each point $p' \in \operatorname{odd}(\mathcal{B}^*)$ is contained in $|\operatorname{odd}(\mathcal{B}^*)| - 1$ lines not in $\mathcal{B}^*$, and hence in $q+1-(|\operatorname{odd}(\mathcal{B}^*)|-1) = q+2-|\operatorname{odd}(\mathcal{B}^*)|$ lines in $\mathcal{B}^*$. Since

$$2q + 2 = |\mathcal{B}^*| + |\operatorname{odd}(\mathcal{B}^*)| = |\operatorname{odd}(\mathcal{B}^*)|(q + 2 - |\operatorname{odd}(\mathcal{B}^*)|) + |\operatorname{odd}(\mathcal{B}^*)|,$$

solving this quadratic equation yields that $|\operatorname{odd}(\mathcal{B}^*)| = 2$ or $|\operatorname{odd}(\mathcal{B}^*)| = q + 1$. The latter would imply $|\mathcal{B}| = q + 1$, which yields $k = 0$ and is hence excluded by our assumptions, so we must have $|\operatorname{odd}(\mathcal{B}^*)| = 2$ and $k = q - 1$; we recall that $p \in \operatorname{odd}(\mathcal{B}^*)$.

Finally, we return to the original affine case, with L as the line at infinity. Since p is the only odd-point on L, this means that in the affine plane we have $|\mathcal{B}| = 2q - 1$ and $|\operatorname{odd}(\mathcal{B})| = 1$, meaning all lines not through p must pass through this single affine odd-point. Hence, we end up with Example 5.2.1(g).

– $|\mathcal{B}| \in [q + 1, 2q]$, $|\mathcal{B}|$ even. First, we show that if $\mathcal{B}$ has odd-points at infinity, then they all have the same number k of $\mathcal{B}$-lines through them, and $|\mathcal{B}| = q + k$ and $|\operatorname{odd}(\mathcal{B})| = q - k$. Let p, p' be such points (if no two such points exist, there are none because $|\mathcal{B}|$ is even, hence the claim is trivially true) and assume that $\deg p = k$ and $\deg p' = k'$, with $k < k'$ and both are odd. Since $|\mathcal{B}| + |\operatorname{odd}(\mathcal{B})| \leq 2q$ and $|\mathcal{B}| \geq q+1$, it follows that $|\operatorname{odd}(\mathcal{B})| \leq q-1$. By Lemma 5.1.4, each of the $q-k$ distinct non-$\mathcal{B}$-lines through p and each of the $q - k'$ distinct non-$\mathcal{B}$-lines through p' contains at least one point of $\operatorname{odd}(\mathcal{B})$. If there are $q - \varepsilon$ lines in $\mathcal{B}$ not through

p, with $\varepsilon > 0$, then $k > 1$ (otherwise this conflicts with $|\mathcal{B}| \geq q+1$) and there are at least $k\varepsilon$ points of $\text{odd}(\mathcal{B})$ on $\mathcal{B}$-lines through p, resulting in

$$|\mathcal{B}| + |\,\text{odd}(\mathcal{B})| \geq (q - \varepsilon + k) + (q - k + k\varepsilon) > 2q,$$

a contradiction. So, there must be at least q lines in $\mathcal{B}$ not through p, and similarly for p'. Now, if there were at least $q+1$, we would have

$$|\mathcal{B}| + |\,\text{odd}(\mathcal{B})| \geq (q + 1 + k) + (q - k) > 2q,$$

again a contradiction, which means there must be exactly q lines in $\mathcal{B}$ not through p, and similarly for p'. Hence, $|\mathcal{B}| = q + k$ and similarly $|\mathcal{B}| = q + k'$, resulting in $k = k'$. So, if $\mathcal{B}$ has odd-points at infinity, then they all have the same number k of $\mathcal{B}$-lines through them. Since we have $|\,\text{odd}(\mathcal{B})| \geq q - k$, it follows indeed that we have $|\mathcal{B}| = q + k$ and $|\,\text{odd}(\mathcal{B})| = q - k$.

Denote the number of odd-points at infinity (which we have shown to be k-points) by m. Clearly m must be even.

Now, we show that $m > 0$. Assume $m = 0$, i.e. all points of the line at infinity are contained in an even number of lines of $\mathcal{B}$. Embed our affine plane in a projective plane $\text{PG}(2, q)$, then since the line at infinity has no odd-points, $\mathcal{B}$ and $\text{odd}(\mathcal{B})$ remain invariant after this embedding. In this case, there can be no affine 1-points: if there were an affine point that lies on a single $\mathcal{B}$-line, Lemma 5.1.5 would yield $|\,\text{odd}(B)| \geq 1 + q$ and hence $|\mathcal{B}| + |\,\text{odd}(\mathcal{B})| \geq 2q + 2 > 2q$, a contradiction. So there cannot be any affine 1-points in this case. Let L be any line in $\mathcal{B}$. Since there are no 1-points, every point $p \in L$ has at least one other $\mathcal{B}$-line through it. Since L is intersected by an odd number of lines (namely $|\mathcal{B}| - 1$), and there are an even number of points on L, there must be some point $p \in L$ which is intersected by an even number of lines of $\mathcal{B} \setminus \{L\}$ and hence by an odd number of lines of $\mathcal{B}$ in total. Therefore, the number of lines in $\mathcal{B}$ that do not contain p is odd, and Lemma 5.1.5 yields

$$|\,\text{odd}(\mathcal{B})| \geq (q + 1 - |\{L \in \mathcal{B} : p \in L\}|) + 1,$$

and on the other hand, we derived that every point of $L \setminus \{p\}$ is contained in at least one other line of $\mathcal{B}$ (different from L), yielding that $|\mathcal{B}| \geq k + q$, where we denote $k = |\{M \in \mathcal{B} : p \in M\}|$. Summing up, we obtain that

$$|\mathcal{B}| + |\,\text{odd}(\mathcal{B})| \geq (q + 1 - k) + 1 + (k + q) = 2q + 2 > 2q,$$

a contradiction.

Hence, $m \geq 2$ and we have an odd number k such that each of these m points at infinity has degree k, all other points at infinity have even degree, $|\mathcal{B}| = q + k$, $|\,\text{odd}(\mathcal{B})| = q - k$, each of the $q - k$ non-$\mathcal{B}$-lines through each of these m points contains exactly one odd-point each, and consequently there are no odd-points on any $\mathcal{B}$-lines through any of these m points.

Now we distinguish three cases.

- If $m = 2$, let p, p' be the odd-points at infinity, both with k lines of $\mathcal{B}$ passing through them. Embed $\mathcal{B}$ in a projective plane $\text{PG}(2, q)$, then $|\mathcal{B}|$ remains

identical and $|\operatorname{odd}_{\mathrm{PG}}(\mathcal{B})| = |\operatorname{odd}_{\mathrm{AG}}(\mathcal{B})| + 2$ (namely the odd-points p and p' are added), hence we have $|\mathcal{B}| = q + k$ and $|\operatorname{odd}_{\mathrm{PG}}(\mathcal{B})| = q - k + 2$. Let p'' be an arbitrary odd-point of $\mathcal{B}$ and note that the number of lines in $\mathcal{B}$ that do not contain p'', is odd, so Lemma 5.1.5 yields

$$q + 2 - k = |\operatorname{odd}_{\mathrm{PG}}(\mathcal{B})| \geq (q + 1 - \deg p'') + 1, \tag{5.2}$$

which shows that $\deg p'' \geq k$. Now pick any line $L \in \mathcal{B}$ through p''. Every point on L is contained in at least one other line of $\mathcal{B}$, or is another odd-point which was not counted in (5.2), yielding that

$$2q + 2 = |\mathcal{B}| + |\operatorname{odd}_{\mathrm{PG}}(\mathcal{B})| \geq (q + \deg p'') + (q + 2 - \deg p'') = 2q + 2.$$

Equality is met, while p'' was random, so both inequalities must attain equality for any odd-point p'':

- every odd-point has the same degree k;
- every line $L' \in \mathcal{B}$ containing an odd-point p'', has its other q points covered bijectively by the other q lines of $\mathcal{B}$ (and in particular, no two odd-points can lie on the same $\mathcal{B}$-line);
- every line $L' \notin \mathcal{B}$ containing an odd-point p'', contains exactly one other point of $\operatorname{odd}_{\mathrm{PG}}(\mathcal{B})$.

In particular, this implies that we cannot have three collinear odd-points, i.e. $\operatorname{odd}_{\mathrm{PG}}(\mathcal{B})$ is an arc. Hence, each point $p'' \in \operatorname{odd}_{\mathrm{PG}}(\mathcal{B})$ is contained in $|\operatorname{odd}_{\mathrm{PG}}(\mathcal{B})| - 1$ lines not in $\mathcal{B}$, and so in $q + 1 - (|\operatorname{odd}_{\mathrm{PG}}(\mathcal{B})| - 1) = q + 2 - |\operatorname{odd}_{\mathrm{PG}}(\mathcal{B})|$ lines in $\mathcal{B}$. Consequently, $\mathcal{B}$ has at least $|\operatorname{odd}_{\mathrm{PG}}(\mathcal{B})|(q + 2 - |\operatorname{odd}_{\mathrm{PG}}(\mathcal{B})|)$ lines in total, from which it follows that

$$2q + 2 = |\mathcal{B}| + |\operatorname{odd}_{\mathrm{PG}}(\mathcal{B})| \geq |\operatorname{odd}_{\mathrm{PG}}(\mathcal{B})|(q + 2 - |\operatorname{odd}_{\mathrm{PG}}(\mathcal{B})|) + |\operatorname{odd}_{\mathrm{PG}}(\mathcal{B})|,$$

which yields $|\operatorname{odd}_{\mathrm{PG}}(\mathcal{B})| \leq 2$ or $|\operatorname{odd}_{\mathrm{PG}}(\mathcal{B})| \geq q + 1$. The latter would imply $|\operatorname{odd}_{\mathrm{PG}}(\mathcal{B})| = q + 1$, and since it is an arc, it is a conic (Theorem 1.1.9). Since no two odd-points are connected by a line of $\mathcal{B}$, $\mathcal{B}$ must be exactly the set of tangents to $\operatorname{odd}_{\mathrm{PG}}(\mathcal{B})$, which is a dual conic, and hence $m = q + 1 \neq 2$, contradicting our assumptions. It follows that we must have $|\operatorname{odd}_{\mathrm{PG}}(\mathcal{B})| = 2$, which means $k = q$ and p, p' are the only odd-points. This means there are no affine odd-points and we end up with Example 5.2.1(f).

* If $m = q+1$, then $k = 1$ and so $\mathcal{B}$ has only 1-points on the line at infinity. Then, $|\mathcal{B}| = q+1$ and Lemma 5.1.4 on any point at infinity yields $|\operatorname{odd}_{\mathrm{AG}}(\mathcal{B})| \geq q-1$. Since $|\mathcal{B}| + |\operatorname{odd}_{\mathrm{AG}}(\mathcal{B})| \leq 2q$, it follows that in Lemma 5.1.4, equality holds in every point at infinity, hence for any line $L \in \mathcal{B}$, each of the q other lines must hit in exactly one of its q points, which yields that $\mathcal{B}$ is a dual arc and hence (by Theorem 1.1.9) a dual conic. This means we end up with Example 5.2.1(e).
* If $2 < m < q + 1$, then we have Example 5.2.1(i).

- Assume $\mathcal{B}$ is not planar. Assume that $|\mathcal{B}| + |\operatorname{odd}(\mathcal{B})| \leq 2q$, then we will now determine all possible intersections of $\mathcal{B}$ with planes. Let π be an arbitrary plane, and consider the subset $\mathcal{B}_\pi = \{L \in \mathcal{B} : L \subseteq \pi\}$, then Lemma 5.1.6 shows that $|\mathcal{B}_\pi| + |\operatorname{odd}(\mathcal{B}_\pi)| \leq 2q$,

and by the planar case in this proof, it is one of the types of Example 5.2.1(a)-(g) or Example 5.2.1(i).

Let π be an arbitrary plane, and let L be any line of $\mathcal{B}$ outside of π (which must exist for every plane π, since $\mathcal{B}$ is not planar). The set $\mathcal{B}_\pi \cup \{L\}$ has $|\mathcal{B}_\pi|+1$ lines, and its set of odd-points has size $|\operatorname{odd}(\mathcal{B}_\pi)|+q-1$ (if L hits π in a point of $\operatorname{odd}(\mathcal{B}_\pi)$) or $|\operatorname{odd}(\mathcal{B}_\pi)|+q$ (otherwise). Now any line not in $\mathcal{B}_\pi \cup \{L\}$ can cancel at most two odd-points, while adding to the total size of $\mathcal{B}$. It follows that $|\mathcal{B}_\pi|+|\operatorname{odd}(\mathcal{B}_\pi)| \geq |\mathcal{B}_\pi|+1+\frac{|\operatorname{odd}(\mathcal{B}_\pi)|+q-1}{2}$, which means that if $|\mathcal{B}_\pi|+|\operatorname{odd}(\mathcal{B}_\pi)| = 2q$ for some plane π, then $|\operatorname{odd}(\mathcal{B}_\pi)| \geq q+1$.

Examples 5.2.1(d), 5.2.1(e), 5.2.1(f), 5.2.1(g) and 5.2.1(i) all have $|\mathcal{B}|+|\operatorname{odd}(\mathcal{B})| = 2q$, but $|\operatorname{odd}(\mathcal{B})| < q+1$, so they cannot occur as $\mathcal{B}_\pi$. Consequently, $\mathcal{B}_\pi$ is of type Example 5.2.1(a)-(c) for all planes π.

In particular, this means that $\mathcal{B}$ contains no line triangles, i.e. there are no lines $L, L', L'' \in \mathcal{B}$ which intersect in pairwise different points. If all lines of $\mathcal{B}$ were concurrent (and $\mathcal{B}$ is nonplanar so $|\mathcal{B}| \geq 3$), there would be at least $3q$ odd-points, which contradicts the assumption $|\mathcal{B}|+|\operatorname{odd}(\mathcal{B})| \leq 2q$. Therefore, the fact that $\mathcal{B}$ is not planar implies that there are lines $L_1, L_2 \in \mathcal{B}$ which span a 3-dimensional subspace (and hence do not intersect). Let k be the number of $\mathcal{B}$-lines that intersect both L_1 and L_2, denote them by $M_1, M_2, \ldots, M_k$. Since there are no line triangles, $M_1, \ldots, M_k$ intersect L_1 and L_2 both in k distinct points. Clearly, we have $k \geq 2$, since there are $2(q-k)$ points which are either odd or need a line just to cover that point, which together with the $2+k$ lines $L_1, L_2, M_1, \ldots, M_k$ would already yield $|\mathcal{B}|+|\operatorname{odd}(\mathcal{B})| \geq 2(q-k)+2+k = 2q+2-k > 2q$ if $k \leq 1$.

Now, similarly, let m be the number of lines that intersect both M_1 and M_2, denote them by $L_1, L_2, \ldots, L_m$. Since there are no line triangles, $L_1, \ldots, L_m$ intersect M_1 and M_2 both in m distinct points. Now, we already have $m+k$ lines in $\mathcal{B}$, and there are all together $4q-2(l+k)$ points on L_1, L_2, M_1, M_2 which are either odd-points, or which need another $\mathcal{B}$-line through them that cannot contain any other point of these $4q-2(l+k)$ points (again because there are no line triangles). Hence, $|\mathcal{B}|+|\operatorname{odd}(\mathcal{B})| \geq l+k+4q-2(l+k) = 4q-(l+k)$, where $l, k \leq q$ by construction. Since we assumed that $|\mathcal{B}|+|\operatorname{odd}(\mathcal{B})| \leq 2q$, this implies $l = k = q$, $|\mathcal{B}| = 2q$ and $|\operatorname{odd}(\mathcal{B})| = 0$, which means $\mathcal{B}$ forms the union of the reguli of a hyperbolic quadric (see [62, Section 15.3.III] for more background on hyperbolic quadrics) and hence $\mathcal{B}$ must be Example 5.2.1(h). □

Open Problem 5.2.3. It is an open problem whether a set as in Example 5.2.1(i) can actually exist. The author believes it cannot. Proving this remains an important open problem: this is the only case where there is only a characterization. If this could be proven, a full classification of these sets is known.

5.3 The projective case

In $\mathrm{PG}(n,q)$, we can obtain a similar classification. Here we obtain a full classification of all examples.

Example 5.3.1. The following line sets $\mathcal{B}$ all have a small value for $|\mathcal{B}|+|\operatorname{odd}(\mathcal{B})|$ in $\mathrm{PG}(n,q)$, q odd.

(a) If $\mathcal{B}$ is the empty set, then $|\mathcal{B}| = 0$ and $|\operatorname{odd}(\mathcal{B})| = 0$ (sum: 0).

(b) If $\mathcal{B}$ consists of a single line, then $|\mathcal{B}| = 1$ and $|\operatorname{odd}(\mathcal{B})| = q + 1$ (sum: $q + 2$).

(c) If $\mathcal{B}$ consists of any two (different) intersecting lines, then $|\mathcal{B}| = 2$ and $|\operatorname{odd}(\mathcal{B})| = 2q$ (sum: $2q + 2$).

(d) If $\mathcal{B}$ is the line set of a dual (planar) conic, then $|\mathcal{B}| = q+1$ and $|\operatorname{odd}(\mathcal{B})| = q+1$ (sum: $2q + 2$).

(e) If $\mathcal{B}$ is the symmetric difference of two (different) planar pencils in the same plane, then $|\mathcal{B}| = 2q$ and $|\operatorname{odd}(\mathcal{B})| = 2$ (sum: $2q + 2$).

(f) If $\mathcal{B}$ is the line set of a hyperbolic quadric $Q^+(3, q)$ in a 3-dimensional subspace, then $|\mathcal{B}| = 2q + 2$ and $|\operatorname{odd}(\mathcal{B})| = 0$ (sum: $2q + 2$).

Theorem 5.3.2. *Let $\mathcal{B}$ be a set of lines in* $\mathrm{PG}(n, q)$*, $n \geq 2$, q odd. If $|\mathcal{B}| + |\operatorname{odd}(\mathcal{B})| \leq 2q+2$, then $\mathcal{B}$ is one of the examples in Example 5.3.1.*

Proof. We distinguish the following cases.

- Assume $\mathcal{B}$ is planar, i.e. it is completely contained in some 2-dimensional subspace π of $\mathrm{PG}(n, q)$. From now on, we only consider this two-dimensional subspace and we act as if $n = 2$. We distinguish the following cases.
  - Assume $\mathcal{B}$ has no odd-points. So, it is a set of lines which contains every point in $\mathrm{PG}(2, q)$ an even number of times. This implies that $\mathcal{B}$ is a code word of the binary code having the incidence matrix of $\mathrm{PG}(2, q)$ as its parity check matrix. It is well known (e.g. as a special case of [42], which shows that this code has dimension 1) that this code only contains the empty set and the set of all lines; together with $|\mathcal{B}| + |\operatorname{odd}(\mathcal{B})| \leq 2q + 2$ this means that $\mathcal{B}$ must be the empty set, so we have Example 5.3.1(a).
  - Assume $\mathcal{B}$ has at least one odd-point p. Since p is an odd-point, the number of $\mathcal{B}$-lines through p is at least one. Let L be such a line and embed $\mathcal{B}$ in an affine plane by considering L as the line at infinity. Since both p and L are now at infinity, the resulting affine structure now has $|\mathcal{B}| + |\operatorname{odd}(\mathcal{B})| \leq 2q$. Theorem 5.2.2 then yields that the affine part of $\mathcal{B}$ must be one of the planar structures in Theorem 5.2.2 (and not (h) as that is not planar).
    - if the affine part of $\mathcal{B}$ is Example 5.2.1(a), then $\mathcal{B}$ must be Example 5.3.1(b);
    - if the affine part of $\mathcal{B}$ is Example 5.2.1(b), then $\mathcal{B}$ must be Example 5.3.1(c);
    - if the affine part of $\mathcal{B}$ is Example 5.2.1(d), then $\mathcal{B}$ must be Example 5.3.1(d);
    - if the affine part of $\mathcal{B}$ is Example 5.2.1(g), then $\mathcal{B}$ must be Example 5.3.1(e);
    - if the affine part of $\mathcal{B}$ is Example 5.2.1(c), 5.2.1(e) or 5.2.1(f), then there would be at least two odd-points at infinity, which would give $|\mathcal{B}| + |\operatorname{odd}(\mathcal{B})| \geq 2q+3$ projectively.

  - ∗ if the affine part of $\mathcal{B}$ is is Example 5.2.1(i), then there would be at least $\min(m, q+1-m) \geq 2$ odd-points at infinity (whether it is m or $q+1-m$ depends on whether or not the line at infinity is in $\mathcal{B}$ or not), which would give $|\mathcal{B}| + |\operatorname{odd}(\mathcal{B})| \geq 2q+3$ projectively.

- Assume $\mathcal{B}$ is not planar. Assume that $|\mathcal{B}|+|\operatorname{odd}(\mathcal{B})| \leq 2q+2$, then we will now determine all possible intersections of $\mathcal{B}$ with planes. Let π be an arbitrary plane, and consider the subset $\mathcal{B}_\pi = \{L \in \mathcal{B} : L \subseteq \pi\}$, then Lemma 5.1.6 shows that $|\mathcal{B}_\pi| + |\operatorname{odd}(\mathcal{B}_\pi)| \leq 2q+2$, and by the planar case in this proof, it is one of the types of Example 5.3.1(a)-(e).

  First of all, we show that there are no planes π such that $\mathcal{B}_\pi$ is of type Example 5.3.1(d) or Example 5.3.1(e). Assume that a plane π exists such that $\mathcal{B}_\pi$ is of type 5.3.1(d). By Lemma 5.1.6, since we have equality in the inequality there, every line in $\mathcal{B} \setminus \mathcal{B}_\pi$ must intersect π in a point of $\operatorname{odd}(B_\pi)$. Let L, L' be two such lines in $\mathcal{B} \setminus \mathcal{B}_\pi$ (L and L' exist since $\mathcal{B}$ is not planar and with one line we clearly have $|\mathcal{B}| + |\operatorname{odd}(\mathcal{B})| > 2q+2$).

  - If L, L' do not intersect, each of the $2q$ points on them outside of π must either be an odd-point or lie on yet another line of $\mathcal{B}$. Since another line of $\mathcal{B}$ can intersect L, L' in at most one point each, $\operatorname{odd}(\mathcal{B}) \setminus (\pi \cup L \cup L')$ has at least $2(q-k)$ points, with $k = |\mathcal{B} \setminus (\mathcal{B}_\pi \cup \{L, L'\})|$. Since $|\mathcal{B}| = (q+1)+2+k$, it follows that

    $$|\mathcal{B}| + |\operatorname{odd}(\mathcal{B})| \geq (q+1)+2+k+\max(2q-2k, 0) \geq 2q+3,$$

    contradiction.
  - If L, L' intersect, let $k = |\mathcal{B} \setminus \mathcal{B}_\pi|$. Since $\mathcal{B}_\pi$ is a dual arc, no line in $\mathcal{B} \setminus \mathcal{B}_\pi$ can intersect both L and L'. So, there are at least $(2q-1)-(k-2)$ points of $\operatorname{odd}(\mathcal{B})$ on L outside of π. So

    $$|\mathcal{B}| + |\operatorname{odd}(\mathcal{B})| \geq (q+1+k)+(2q-1-(k-2)) = 3q+2 > 2q+2,$$

    a contradiction.

  This shows that there are no planes π such that $\mathcal{B}_\pi$ is of type Example 5.3.1(d). Similarly, assume that a plane π exists such that $\mathcal{B}_\pi$ is of type 5.3.1(e). By Lemma 5.1.6, since we have equality in the inequality there, every line in $\mathcal{B} \setminus \mathcal{B}_\pi$ must intersect π in a point of $\operatorname{odd}(B_\pi)$. Let L be such a line in $\mathcal{B} \setminus \mathcal{B}_\pi$ (L exists since $\mathcal{B}$ is not planar) and let k be the number of such lines. Then there are at least $q-(k-1)$ points of $\operatorname{odd}(\mathcal{B})$ on L outside of π. Hence, we have

  $$|\mathcal{B}| + |\operatorname{odd}(\mathcal{B})| \geq (2q+k)+(q-(k-1)) = 3q+1 > 2q+2,$$

  again a contradiction. Consequently, there are no planes π such that $\mathcal{B}_\pi$ is of type Example 5.3.1(e).

  So, $\mathcal{B}_\pi$ is of type Example 5.3.1(a)-(c) for all planes π. In particular, this means that $\mathcal{B}$ contains no line triangles, i.e. there are no lines $L, L', L'' \in \mathcal{B}$ which intersect in pairwise different points. Since $\mathcal{B}$ is not planar, and since the assumption $|\mathcal{B}|+|\operatorname{odd}(\mathcal{B})| \leq 2q+2$ excludes the possibility that all lines of $\mathcal{B}$ are concurrent, there are $L_1, L_2 \in \mathcal{B}$ which span a 3-dimensional subspace (and hence do not intersect). Let k be the number of lines of $\mathcal{B}$ that intersect both L_1 and L_2, denote them by $M_1, M_2, \ldots, M_k$. Since there

are no line triangles, $M_1, \ldots, M_k$ intersect L_1 and L_2 both in k distinct points. Clearly, we have $k \geq 2$, since there are $2(q+1-k)$ points which are either odd or need a line just to cover that point, which together with the $2+k$ lines $L_1, L_2, M_1, \ldots, M_k$ would already yield $|\mathcal{B}| + |\operatorname{odd}(\mathcal{B})| \geq 2(q+1-k)+2+k = 2q+4-k > 2q+2$ if $k \leq 1$. Now, similarly, let m be the number of lines that intersect both M_1 and M_2, denote them by $L_1, L_2, \ldots, L_m$. Since there are no line triangles, $L_1, \ldots, L_m$ intersect M_1 and M_2 both in m distinct points. Now, we already have $m+k$ lines in $\mathcal{B}$, and there are all together $4(q+1)-2(l+k)$ points on L_1, L_2, M_1, M_2 which are either odd-points, or which need another $\mathcal{B}$-line through it that cannot contain any other point of these $4(q+1)-2(l+k)$ points (again because there are no line triangles). Hence, $|\mathcal{B}| + |\operatorname{odd}(\mathcal{B})| \geq l+k+4(q+1)-2(l+k) = 4(q+1)-(l+k)$, where $l, k \leq q+1$ by construction. Since we assumed that $|\mathcal{B}| + |\operatorname{odd}(\mathcal{B})| \leq 2q+2$, this implies $l = k = q+1$, $|\mathcal{B}| = 2(q+1)$ and $|\operatorname{odd}(\mathcal{B})| = 0$, which means $\mathcal{B}$ must be Example 5.3.1(f). □

This completes the classification.

Chapter 6

Geometries over finite chain rings

6.1 Motivation and preliminaries

6.1.1 Finite chain rings

We start by introducing the notations that will be used about finite chain rings.

Definition 6.1.1. An associative ring with identity is called a left (right) chain ring if the lattice of its left (right) ideals is a chain. The following lemma essentially describes the structure of a finite chain ring.

Lemma 6.1.2. *For a finite ring $\mathfrak{R}$, the following conditions are equivalent:*

1. *$\mathfrak{R}$ is a left chain ring;*
2. *$\mathfrak{R}$ is a right chain ring;*
3. *the principal left ideals of $\mathfrak{R}$ form a chain;*
4. *$\mathfrak{R}$ is a local ring and $\operatorname{Rad}\mathfrak{R} = \mathfrak{R}\theta$ for any $\theta \in \operatorname{Rad}\mathfrak{R}/(\operatorname{Rad}\mathfrak{R})^2$.*

If $\mathfrak{R}$ satisfies the above conditions then every proper ideal of $\mathfrak{R}$ has the form $(\operatorname{Rad}\mathfrak{R})^i = \mathfrak{R}\theta^i = \theta^i\mathfrak{R}$ for some positive integer i.

Notation 6.1.3. By $\mathfrak{R}^{opp}$ we denote the opposite ring of $\mathfrak{R}$, i.e. the ring by switching the left and right multiplication rules. For a commutative ring, $\mathfrak{R} = \mathfrak{R}^{opp}$.

Let $\mathfrak{R}$ be a finite chain ring. It is well known that $|\mathfrak{R}| = q^m$ for some $q = p^h$ with p prime and h a positive integer; and that its radical $\operatorname{Rad}\mathfrak{R}$ can be generated by a single element, say θ. Moreover, it is known that there exists a set $\Gamma = \{\gamma_0 = 0, \gamma_1 = 1, \gamma_2, \ldots, \gamma_{q-1}\} \subseteq \mathfrak{R}$, such that $\gamma_i \not\equiv \gamma_j \pmod{\operatorname{Rad}\mathfrak{R}}$ for $i \neq j$, and such that for each $\beta \in \mathfrak{R}$ there exist unique $b_0, b_1, \ldots, b_{m-1} \in \Gamma$ such that $\beta = \sum_{i=0}^{m-1} b_i\theta^i$. Such a set Γ, together with the generator θ,

is called a *basis* for the finite chain ring $\mathfrak{R}$. Lastly, it can be shown that Γ, together with the $\mathfrak{R}$-addition modulo $\operatorname{Rad}\mathfrak{R}$, and the $\mathfrak{R}$-multiplication modulo $\operatorname{Rad}\mathfrak{R}$, forms a finite field (i.e. $\mathfrak{R}/\operatorname{Rad}\mathfrak{R} \cong \mathbb{F}_q$). For a more detailed study of finite chain rings we refer to [27, 102, 107].

Now, we will define a bijection $\varphi : \mathfrak{R} \to \{0, 1, \ldots, q^m - 1\}$, which at the same time provides a natural ordering of the elements of $\mathfrak{R}$ (with respect to the given basis) and yields a simple computer representation of the element as an integer in $\{0, 1, \ldots, q^m - 1\}$.

Definition 6.1.4. Let $\varphi(0) = 0$ and $\varphi(1) = 1$. For $a_2, a_3, \ldots, a_{q-1}$, we set $\varphi(a_i) = i$. Now, for an arbitrary $\beta = \sum_{i=0}^{m-1} b_i\theta^i$, we let $\varphi(\beta) = \sum_{i=0}^{m-1} \varphi(b_i)q^i$.

Corollary 6.1.5. *Some properties of φ:*

- $\beta \in \Gamma \Leftrightarrow \varphi(\beta) < q$;
- *for each $i \in \mathbb{N}$, the equivalence $\beta \in (\operatorname{Rad}\mathfrak{R})^i \Leftrightarrow \exists r \in \mathfrak{R} : \beta = r\theta^i \Leftrightarrow \varphi(\beta)|q^i$ holds;*
- *if $q^i|\varphi(\beta)$, then $r = \varphi^{-1}\left(\frac{\varphi(\beta)}{q^i}\right)$ has $\beta = r\theta^i$.*

It can be shown that for every positive integer i and for every element $\alpha \in \mathfrak{R}$ with $\varphi(\alpha) < q^{m-i}$, there is a unique element $\operatorname{conj}_i(\alpha) \in \mathfrak{R}$ with $\varphi(\operatorname{conj}_i(\alpha)) < q^{m-i}$ such that $\alpha\theta^i = \theta^i \operatorname{conj}_i(\alpha)$. This is called the i-conjugate of α.

In what follows, we consider a fixed finite chain ring $\mathfrak{R}$, a fixed basis for it, and a fixed function φ. We would like to stress that Corollary 6.1.5 plays an important role in Section 6.2.

6.1.2 Modules and their Grassmannians

Modules are the ring analog of vector spaces. Note that given the computational nature of my research on this topic, we only consider finite modules, which can always be represented as a submodule of $\mathfrak{R}^n$, for some positive integer n. For the rest of this chapter, we will stick to the notations from the previous section, i.e. $|\mathfrak{R}| = q^m$ and $|\mathfrak{R}/\operatorname{Rad}\mathfrak{R}| = q$.

Definition 6.1.6. A left (right) *module* M over a finite chain ring $\mathfrak{R}$ is a subset of $\mathfrak{R}^n$ (the set of all n-tuples over $\mathfrak{R}$), which is closed under $\mathfrak{R}$-addition and left (right) multiplication with an element of $\mathfrak{R}$.

Definition 6.1.7. A *partition* of length ℓ and maximum m of the positive integer N is a multiset of integers $\lambda = \{\lambda_1, \lambda_2, \ldots, \lambda_\ell\}$ with $N = \lambda_1 + \lambda_2 + \cdots + \lambda_\ell$ and $m \geq \lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_\ell \geq 0$. Its *conjugate partition* is $\lambda' = \{\lambda'_1, \lambda'_2, \ldots, \lambda'_m\}$, where $\lambda'_j = |\{i : \lambda_i \geq j\}|$.

The following theorem describes the structure of an arbitrary (finite) $\mathfrak{R}$-module M up to isomorphism.

Theorem 6.1.8. *For every $\mathfrak{R}$-module M there is a unique partition $\lambda = \{\lambda_1, \lambda_2, \ldots, \lambda_n\}$ of length n and maximum m such that*

$$M \cong \mathfrak{R}/(\operatorname{Rad}\mathfrak{R})^{\lambda_1} \oplus \cdots \oplus \mathfrak{R}/(\operatorname{Rad}\mathfrak{R})^{\lambda_n}.$$

Its conjugate partition $\lambda' = (\lambda'_1, \lambda'_2, \ldots, \lambda'_m)$ is given by $\lambda'_i = \dim_{\mathfrak{R}/\operatorname{Rad}\mathfrak{R}}(M[\theta] \cap \theta^{i-1}M)$, the Ulm-Kaplansky invariants.

Definition 6.1.9. The partitions λ and λ' are called the *shape* and *conjugate shape*, respectively.

Theorem 6.1.10 ([63])**.** *Let A be an $n \times n$ matrix over $\mathfrak{R}$. Then the left module generated by the rows of A and the right module generated by the columns of A have the same shape.*

Definition 6.1.11. Let λ be a partition of length n. The set of all modules of shape λ in $\mathfrak{R}^n$ is called the λ-*Grassmannian* and is denoted by $\mathcal{G}(\lambda)$. The set of all submodules of $\mathfrak{R}^n$ is called the *Grassmannian*, and is denoted by $\mathcal{G}$.

Theorem 6.1.12 ([96])**.** *Let M be a $\mathfrak{R}$-module of shape $\lambda = (\lambda_1, \ldots, \lambda_n)$, and let $\mu = (\mu_1, \ldots, \mu_n)$ be a partition with $\mu_i \leq \lambda_i$ for all i. Then there are exactly*

$$\begin{bmatrix} \lambda \\ \mu \end{bmatrix}_q := \prod_{i=1}^{\infty} q^{\mu'_{i+1}(\lambda'_i - \mu'_i)} \cdot \begin{bmatrix} \lambda'_i - \mu'_{i+1} \\ \mu'_i - \mu'_{i+1} \end{bmatrix}_q$$

submodules of shape μ contained in M.

For a given positive integer n and a non-increasing sequence of non-negative integers $\kappa = (\kappa_1, \ldots, \kappa_n)$ we denote by $\mathcal{G}(n, \kappa)$ the set of all submodules of shape κ of ${}_{\mathfrak{R}}\mathfrak{R}^n$. In what follows, we denote the sequence $(\underbrace{m, \ldots, m}_{k}, \underbrace{0, \ldots, 0}_{n-k})$ by $\boldsymbol{m}^k$. For two sequences $\lambda = (\lambda_1, \ldots, \lambda_n)$ and $\mu = (\mu_1, \ldots, \mu_n)$, we write $\lambda \preceq \mu$ iff $\lambda_i \leq \mu_i$ for all $i = 1, \ldots, n$.

By duality, Theorem 6.1.12 allows to find the number of shape λ submodules of ${}_{\mathfrak{R}}\mathfrak{R}^n$ containing a fixed submodule of shape μ. One has to apply Theorem 6.1.12 to the dual submodules that have shapes $\boldsymbol{m}^n - \lambda$ and $\boldsymbol{m}^n - \mu$, respectively. So, this number equals $\begin{bmatrix} \boldsymbol{m}^n - \mu \\ \boldsymbol{m}^n - \lambda \end{bmatrix}_q$.

6.1.3 Hjelmslev geometries

The set of all modules of the Grassmannian $\mathcal{G}$, together with the inclusion relation, defines an incidence geometry which is commonly denoted by $\mathrm{PHG}(\mathfrak{R}^n)$. Its Grassmannian forms a lattice and one can study the geometry of its subspaces just like one can study the projective spaces $\mathrm{PG}(n, q)$ constructed over finite fields (and in fact, there are many similarities).

The (left) projective Hjelmslev geometries $\mathrm{PHG}({}_{\mathfrak{R}}\mathfrak{R}^n)$ are produced from the finite modules ${}_{\mathfrak{R}}\mathfrak{R}^n$ in the same way one produces the classical projective geometries $\mathrm{PG}(n-1, q)$ from the vector spaces $\mathbb{F}_q^n$. The geometry $\mathrm{PHG}({}_{\mathfrak{R}}\mathfrak{R}^n)$ is defined as an incidence structure $(\mathcal{P}, \mathcal{L}, I)$ having as points the free rank 1 submodules of ${}_{\mathfrak{R}}\mathfrak{R}^n$ and as lines the free rank 2 submodules of ${}_{\mathfrak{R}}\mathfrak{R}^n$. Incidence is given by set-theoretical inclusion. The set of points contained in a submodule ${}_{\mathfrak{R}}\mathfrak{R}^n\mathfrak{R}M \subset {}_{\mathfrak{R}}\mathfrak{R}^n$ which is of shape λ is called a *subspace* of shape λ. The subspaces defined by free submodules of ${}_{\mathfrak{R}}\mathfrak{R}^n$ are called *Hjelmslev subspaces*.

Two subspaces $\boldsymbol{L}$ and $\boldsymbol{M}$ of the same shape defined by the modules ${}_{\mathfrak{R}}\mathfrak{R}^n\mathfrak{R}L$ and ${}_{\mathfrak{R}}\mathfrak{R}^n\mathfrak{R}M$, respectively, are called i-neighbors if ${}_{\mathfrak{R}}\mathfrak{R}^n\mathfrak{R}L = {}_{\mathfrak{R}}\mathfrak{R}^n\mathfrak{R}M + \theta^i\mathfrak{R}^n$. This fact is denoted by $\boldsymbol{L} \backsim_i \boldsymbol{M}$. It can be checked that i-neighborhood is an equivalence relation on the set of all

subspaces. The equivalence class of all subspaces that are i-neighbors to $\boldsymbol{L}$ is denoted by $[\boldsymbol{L}]^{(i)}$. Set

$$\mathcal{P}^{(i)} = \{[x]^{(i)} \mid x \in \mathcal{P}\}, \mathcal{L}^{(i)} = \{[L]^{(i)} \mid L \in \mathcal{L}\},$$

and

$$I^{(i)} = \{([x]^{(i)}, [L]^{(i)}) \mid \exists x' \in [x]^{(i)}, \exists L' \in [L]^{(i)} : (x', L') \in I\}.$$

Theorem 6.1.13. *The incidence structure $(\mathcal{P}^{(i)}, \mathcal{L}^{(i)}, I^{(i)})$ is isomorphic to the geometry*

$$\mathrm{PHG}\left({}_{\mathfrak{R}}\mathfrak{R}^n\mathfrak{R}/\theta^i\mathfrak{R}(\mathfrak{R}/\theta^i\mathfrak{R})^n\right).$$

Let us note that $\mathfrak{R}/\theta^i\mathfrak{R}$ is again a chain ring of size q^i, with residue field $\mathbb{F}_q$. In particular, for $i = 1$ we get $(\mathcal{P}^{(1)}, \mathcal{L}^{(1)}, I^{(1)}) \cong \mathrm{PG}(n-1, q)$.

Another family of substructures $\mathrm{PHG}({}_{\mathfrak{R}}\mathfrak{R}^n)$ is given by the next theorem. Fix a $(k-1)$-dimensional Hjelmslev subspace $\boldsymbol{M}$ in $\mathrm{PHG}({}_{\mathfrak{R}}\mathfrak{R}^n)$, and an integer j with $0 < j < m$. Denote by $\mathcal{P}_j(\boldsymbol{M})$ and $\mathcal{L}_j(\boldsymbol{M})$ the set of all points/lines that have a j-th neighbour on $\boldsymbol{M}$. Let $[x]^{(m-j)}$, $x \in \mathcal{P}_j(\boldsymbol{M})$, be the neighbour class of all $(m-j)$-th neighbors to x, and, similarly, let $[L]^{(m-j)}$, $L \in \mathcal{L}_j(\boldsymbol{M})$, be the neighbour class of all $(m-j)$-th neighbors to L, Define a new point set

$$\mathfrak{P} = \{\boldsymbol{L} \cap [L]^{(m-j)} \mid L \in \mathcal{P}_j(\boldsymbol{M}), \boldsymbol{L} \backsim_i \boldsymbol{M}, \boldsymbol{L} \cap [x]^{(m-j)} \neq \varnothing\}.$$

and a new set of lines $\mathfrak{L}$ as

$$\mathfrak{L} = \{\boldsymbol{L} \cap [L]^{(m-j)} \mid L \in \mathcal{L}_j(\boldsymbol{M}), \boldsymbol{L} \backsim_i \boldsymbol{M}, \boldsymbol{L} \cap [L]^{(m-j)} \neq \varnothing\}.$$

The incidence $\mathfrak{J} \subseteq \mathfrak{P} \times \mathfrak{L}$ is given by set-theoretical inclusion.

Theorem 6.1.14. *The incidence structure $(\mathfrak{P}, \mathfrak{L}, \mathfrak{J})$ can be embedded isomorphically into $\mathrm{PHG}({}_{\mathfrak{R}}\mathfrak{R}^n\mathfrak{R}/(\operatorname{Rad}\mathfrak{R})^{m-j}(\mathfrak{R}/(\operatorname{Rad}\mathfrak{R})^{m-j})^n)$. The missing part consists of the points of an $(n-k-1)$-dimensional Hjelmslev subspace $\boldsymbol{H}$ and all the subspaces which have commom points with $\boldsymbol{H}$.*

Note that $\mathfrak{R}/(\operatorname{Rad}\mathfrak{R})^i$ is again a chain ring with q^i elements. In the special case when $j = 1$, the structure $(\mathfrak{P}, \mathfrak{L}, \mathfrak{J})$ is a part of $\mathrm{PHG}({}_{\mathfrak{R}}\mathfrak{R}^n S S^n)$ where S is a chain ring with q^{m-1} elements. In this case, subspaces of shape $\sigma = (\sigma_1, \ldots, \sigma_t)$ (we suppress trailing zeros) in $[\boldsymbol{M}]$ become subspaces of shape $(\sigma_1 - 1, \ldots, \sigma_t - 1)$ in $(\mathfrak{P}, \mathfrak{L}, \mathfrak{J})$.

If $\boldsymbol{M}$ is a point, the missing part is a hyperplane. Thus the j-th neighbour classes of points carry the structure of an affine geometry over $\mathfrak{R}/(\operatorname{Rad}\mathfrak{R})^j$. For a more detailed introduction into projective Hjelmslev geometries we refer to [64, 65].

Most of the work investigating $\mathrm{PHG}(\mathfrak{R}^n)$ is focused on the structure of the Grassmannian and its substructures and properties up to isomorphism. However, if one wants to make specific constructions and create new incidence matrices, e.g. to be used as an LDPC code, or just to verify certain properties computationally, then an explicit form for the subspaces is required. Hence, in this paper we will introduce the required methods and techniques to perform efficient computations with arbitrary modules, without falling back on their isomorphism class.

6.2 Standard form representation of modules

The first thing we need to handle modules computationally is to have a proper way to represent them in a computer. The most obvious way to do this is the same way as subspaces of vector spaces are usually represented: as matrices. While this idea and the techniques to do it are old, and date back to [12], it is only recently in [38] that a well-defined unique standard form for modules over arbitrary finite chain rings was formally introduced.

We will modify the standard form presented in [38] a little to suit our computational needs. In particular, we will fix the number of zero rows to make the total matrix a square matrix. This does not require extra storage space per module as the number of bits needed in a compact-storage format is proportional to the logarithm of the number of possible contents (and $\log(1) = 0$), while in a fast compute-storage format, the rows should be readily there to work on for algorithmic reasons. Next to that, we will also provide a different order on the rows, as one will see that this provides clear benefits for the algorithms we describe, resulting in much simpler and more human-readable methods, as well as speeding up the algorithms by requiring less index lookups. The results in this section are joint work with I. Landjev.

Remark 6.2.1. From now on, we will consider left modules over $\mathfrak{R}$ only. Of course, the machinery can be copied mutatis mutandis for right modules.

Definition 6.2.2. A matrix $A \in \mathfrak{R}^{n\times n}$ is said to be *standard* or *in standard form* if and only if each of the following conditions are met:

- each element A_{ii} is $A_{ii} = \theta^{m-t_i}$ for some $t_i \in \{0, \ldots, m\}$;
- each element A_{ij}, with $j < i$, is a left multiple of θ^{m-t_i+1} (i.e. $A_{ij} = a_{ij}\theta^{m-t_i+1}$);
- each element A_{ij}, with $j > i$, is a left multiple of θ^{m-t_i} (i.e. $A_{ij} = a_{ij}\theta^{m-t_i}$);
- each element A_{ji}, with $j \neq i$, has $\varphi(A_{ji}) < \varphi(A_{ii})$ if $A_{ii} \neq 0$.

Theorem 6.2.3. *Every module $M \leq \mathfrak{R}^n$ has a unique standard matrix generating it. Similarly, every matrix in standard form corresponds to a unique module $M \leq \mathfrak{R}^n$.*

Proof. Since our form differs only from the one in [38] in terms of the order of the rows and the number of zero rows involved, its existence and uniqueness follow readily from the proofs in [38]. □

From now on, we will represent all $\mathfrak{R}$-modules by their corresponding matrices. To compute this standard form from a given set of generating vectors of the module, Algorithm 1 is used.

Theorem 6.2.4. *Algorithm 1 works as described. Given $\mathfrak{R}$ and given constant-time speed for arithmetic operations in $\mathfrak{R}$ and for φ and φ^{-1}, Algorithm 1 works in $\mathcal{O}(mkn \cdot \min(n, k))$ time, where k is the size of the given generating set.*

Proof. First, we have to show that the division on line 5 is well defined. Whenever the algorithm reaches line 5, the vector r has an element that is not a multiple of θ^{m-t+1}. However,

Algorithm 1 Reduction to standard form

Input: any set $S \subseteq M$ with $M = \langle S \rangle$.
Output: a matrix A in standard form with row span $\langle A \rangle = M$.

1: **for** $t = m, \ldots, 1$ **do**
2: **for** $r \in S$ **do**
3: **if** not all elements in r are multiples of θ^{m-t+1} **then**
4: Let i be the smallest (=leftmost) position with $s_i \nmid \theta^{m-t+1}$.
5: Left-multiply all elements in r by $\left(\varphi^{-1}\left(\frac{\varphi(r_i)}{q^{m-t}}\right)\right)^{-1}$.
6: **for** $r' \in (A \cup S) \setminus \{r\}$ **do**
7: Let $c = \varphi^{-1}\left(\left\lfloor \frac{\varphi(r'_i)}{q^{m-t}} \right\rfloor\right)$ and replace r' by $r' - c \cdot r$.
8: If $r' = \bar{0}$ (can only occur for $r' \in S$), remove it from S.
9: **end for**
10: Put r as the i-th row of A, and remove it from S.
11: **end if**
12: **end for**
13: **end for**
14: **return** A

it cannot have elements that are not a multiple of θ^{m-t}, otherwise it would have been removed on line 10 during a previous iteration of the outer loop on t. Hence, $r_i = u \cdot \theta^{m-t}$, where u is a unit in $\mathfrak{R}$. Such a unit u is given by the expression on line 5, as explained in Corollary 6.1.5. Since it is a unit, it has a multiplicative inverse.

Now, we will show the correctness of the algorithm. Since we are only dividing by units and subtracting scalar multiples of rows from other rows, the row space is invariant during the process, which shows that $\langle A \rangle = \langle S \rangle$. Therefore, it is sufficient to verify that the resulting matrix A is in standard form. If a row of A is not declared during the algorithm, all of its elements are 0 and hence the properties are trivially fulfilled. Otherwise, it gets added at line 10, in which case we can easily verify the defining properties of the standard form.

- Before line 5, $r_i = u \cdot \theta^{m-t}$ as explained above, hence dividing by u makes it indeed of the form θ^{m-t}. Since r is added as the i-th row, this is indeed the element A_{ii} of the resulting matrix.
- Line 4 chooses the leftmost element that is not a multiple of θ^{m-t+1}. Hence, all A_{ij} with $j < i$ must indeed be a multiple of θ^{m-t+1}.
- As explained in the first paragraph of this proof, all elements in the vector r must be multiples of θ^{m-t}, hence this is indeed true for all A_{ij} with $j > i$.
- For every other row r', we can write its value r'_i at position i uniquely as $r'_i = \alpha + \beta\theta^{m-t}$, where $\varphi(\alpha) < q^{m-t}$ and $\varphi(\beta) < q^t$. Line 7 computes $c = \beta$, and since $r_i = \theta^{m-t}$, the result of the subtraction is $r'_i = (\alpha + \beta\theta^{m-t}) - \beta\theta^{m-t} = \alpha$, which has indeed $\varphi(A_{ji}) = \varphi(\alpha) < q^{m-t} = \varphi(A_{ii})$. So, at the time it is added, this property is fulfilled. Moreover, it cannot be modified again later: all rows r'' that would modify our row r' later, must have a multiple of θ^{m-t} at position i (otherwise they would have been

processed in an earlier iteration of the t-loop), together with $\varphi(r''_i) < q^{m-t}$ this shows that $r''_i = 0$ for all unprocessed rows r''. Thus, after it has been added to A, the entries at position i will no longer change and so the property will remain valid.

Finally, we will show that its execution speed is bounded as claimed. Lines 4, 5, 7 and 8 of the algorithm all clearly run in $\mathcal{O}(n)$ time, and cannot have a sharper bound. Line 10 runs $\mathcal{O}(1)$. Now, we count the number of times the statements inside the innermost loop are executed. This number is clearly bounded by $k(k-1)$, which is $\mathcal{O}(k^2)$. However, it is also clear that when the inner loop has been executed n times, all remaining rows must be zero rows, and hence this number is also bounded by $\mathcal{O}(kn)$. Together, the claimed upper bound follows. □

Remark 6.2.5. If no value is assigned to a row in A, it is assumed that this is a zero row. It is not computationally necessary to store or initialize these rows, as they contribute nothing to the span. However, considering the matrix as an $n \times n$ matrix with zero rows, greatly simplifies the notations (compared to the more classical approach of removing them completely) and speeds up the algorithms in the following sections by reducing the number of required index lookups.

Remark 6.2.6. If A is the standard matrix of a module M, the shape of M can be found directly as $\lambda = \{m - \log_q \varphi(A_{ii})\}_{i=0,\ldots,n-1}$ where $\log_q \varphi(0) = m$ by definition.

While the shape is commonly used as the main invariant for modules, it is not so useful to work with from a computational point of view. We will replace it by an ordered variant, that is not invariant under isomorphism, but that turns out to be a useful thing to consider from a computational point of view.

Definition 6.2.7. The *type vector* of M is the vector $\{\log_q \varphi(A_{ii})\}_{i=0,\ldots,n-1}$, where A is the standard matrix of M. In other words: its ith entry is the type of the row A_i, and it is a specific ordering on the shape multiset.

The Grassmannian over $\mathfrak{R}$ is a lattice: every two modules have a unique least common upper bound (their span) and a greatest common lower bound (their intersection). On the way of describing an algorithmic way to compute these, we will also give an algorithmic method to compute the dual lattice of a lattice.

Definition 6.2.8. Let $M = \langle S \rangle$ be a left module. Then the dual module is $M^\perp = \{m \in \mathfrak{R}^n | m \cdot s = 0\}$; it is easily verified that $M^\perp$ is a right module of $\mathfrak{R}^{opp}$.

Lemma 6.2.9. *If the diagonal elements of a matrix are all* 1*, and everything below the diagonal is a multiple of* θ*, then the matrix is invertible.*

Proof. Apply the following elementary row operations to the matrix A.

1: **for** $i = 0, \ldots, n-1$ **do**
2: $\quad A_i = A_{ii}^{-1} A_i$
3: $\quad$ **for** $j = 0, \ldots, i-1$ **do**
4: $\quad\quad A_i = A_i - A_{ij} A_j$

5: **end for**
6: **end for**

Since A_{ij} is always a multiple of θ, A_{ii} is still a unit whenever the inversion in line 2 is called. Since we have only applied elementary row operations to A, we have not changed the invertibility of A. Since we end up with a matrix with 1s on the diagonal and 0s below the diagonal, which is invertible, it follows that the original matrix A was also invertible. □

Lemma 6.2.10. *Let A be the standard form matrix of a module $M \leq \mathfrak{R}^n$. Then there exists an invertible $n \times n$ matrix A' such that $A = CA'$, where C has diagonal form and A' only has 1s on its diagonal.*

Proof. Let C be a diagonal matrix with diagonal entries $C_{ii} = A_{ii} = \theta^{m-t_i}$ for some t_i, define the ith row of A' to be $(A'_i)_j = \mathrm{conj}_{t_i}\left(\varphi^{-1}\left(\frac{\varphi(A_{ij})}{\varphi(A_{ii})}\right)\right)$ for each j.

Now, consider the product CA'. Since C is diagonal,

$$\begin{aligned}(CA')_{ij} &= C_{ii}A'_{ij} \\ &= \theta^{m-t_i}\,\mathrm{conj}_{m-t_i}\left(\varphi^{-1}\left(\tfrac{\varphi(A_{ij})}{\varphi(A_{ii})}\right)\right) \\ &= \varphi^{-1}\left(\tfrac{\varphi(A_{ij})}{\varphi(A_{ii})}\right)\theta^{m-t_i} \\ &= A_{ij}\end{aligned}$$

as claimed.

Finally, it follows from Lemma 6.2.9 that A' is invertible, since A is in standard form so all A_{ij} with $j < i$ have a higher power in θ than A_{ii} and hence $\varphi^{-1}\left(\frac{\varphi(A_{ij})}{\varphi(A_{ii})}\right)$ cannot be a unit for these entries. □

Algorithm 2 Dualization of modules over finite chain rings

Input: The standard matrix A of a module M
Output: A generating matrix for the dual module $M^\perp$

1: Initialize $n \times n$ matrices D, A' as all-zero.
2: **for** $i = 0, \ldots, n-1$ **do**
3: **if** $A_{ii} = 0$ **then**
4: $A'_{ii} = 1$
5: **else**
6: $D_{ii} = \varphi^{-1}\left(\frac{q^m}{\varphi(A_{ii})}\right)$
7: $t_i = m - \log_q \varphi(A_{ii})$
8: **for** $j = 0, \ldots, n-1$ **do**
9: $A'_{ij} = \mathrm{conj}_{m-t_i}\left(\varphi^{-1}\left(\frac{\varphi(A_{ij})}{\varphi(A_{ii})}\right)\right)$
10: **end for**
11: **end if**
12: **end for**
13: **return** $(A'^{-1} \cdot D)^T$ // right inverse, so $A \cdot A^{-1} = I_n$

Theorem 6.2.11. *Algorithm 2 works correctly.*

Proof. Write the input matrix A as $A = CA'$ with notations as in Lemma 6.2.9. Since the constructions of the matrix A' in Lemma 6.2.9 and Algorithm 2 are identical, we can write A' for both matrices without any problem. Let B be the matrix returned by Algorithm 2, then

$$A \cdot B^T = (CA')((A'^{-1}D)^T)^T = (CA')(A'^{-1}D) = C(A'A'^{-1})D = CD = 0.$$

Hence, $\langle B \rangle \leq M^\perp$. On the other hand, the type of the dual module matches that of $\langle B \rangle$ (and that of $\langle D \rangle$), and hence they must be equal. □

Corollary 6.2.12. *Thanks to Algorithm 1, one can compute the span of two modules. Thanks to Algorithm 2, one can compute the intersection of two modules, as* $M_1 \cap M_2 = \langle M_1^\perp, M_2^\perp \rangle^\perp$ *for every two modules* M_1 *and* M_2*.*

Testing whether a vector m is contained in a module M is equivalent to testing whether it is orthogonal to all generating rows of $M^\perp$. However, it can also be done more efficiently, without the need to compute $M^\perp$ at all; this is shown in Algorithm 3, which runs in $\mathcal{O}(kn)$ time, with $k \leq n$ the number of nonzero rows in the standard form of M. Hereby we note that the division in line 3 is exact, for otherwise the algorithm would have exited in a previous iteration of the t-loop.

Algorithm 3 Membership test

Input: The standard matrix A for a module $M \leq \mathfrak{R}^n$, and a vector $m \in \mathfrak{R}^n$
Output: Whether or not $m \in M$

```
1: for t = m, ..., 1 do
2:   for all rows M_i of type t do
3:     m = m − φ^{-1}( (m_i · M_ii) / q^{m−t} ) · M_i
4:     if m contains entries that are not multiples of θ^{m−t+1} then
5:       return false
6:     end if
7:   end for
8: end for
9: return true
```

Testing whether a given submodule is contained in another submodule, can be done by testing inclusion for each of its generating vectors. Testing for equality can also be done that way, although it is sufficient to test if their standard forms are equal.

6.3 Extension of Kantor's theorem to finite chain rings

Let $\Omega = \mathrm{PHG}({}_\mathfrak{R}\mathfrak{R}^n)$. Let $\tau = (\tau_1, \ldots, \tau_n)$ be an integer sequence satisfying $m = \tau_1 \geq \tau_2 \geq \cdots \geq \tau_n \geq 0$. We consider the incidence matrix of all shape $\boldsymbol{m}^s = (\underbrace{m, \ldots, m}_{s})$ versus all shape τ subspaces of Ω with $\boldsymbol{m}^s \preceq \tau \preceq \boldsymbol{m}^{n-s}$. We prove that the rank of $M_{\boldsymbol{m}^s,\tau}(\Omega)$ over

$\mathbb{Q}$ is equal to the number of shape σ subspaces. This is a partial analog of Kantor's result about the rank of the incidence matrix of all s-dimensional versus all t-dimensional subspaces of $\mathrm{PG}(n,q)$, $0 \leq s < t \leq n-s-1$. While it may be tempting to claim that this result holds for arbitrary shapes σ, τ, we construct an example for non-free shapes σ and τ for which the rank of $M_{\sigma,\tau}(\Omega)$ is not maximal. This section is joint work with I. Landjev and the results obtained have been accepted for publication in Des. Codes Cryptogr. [88].

In this section we shall be confined to left modules; this is no restriction since every left module can be considered as a right module over the opposite chain ring.

6.3.1 Incidence matrices and the main theorem

Let $\Omega = \mathrm{PHG}({}_{\mathfrak{R}}\mathfrak{R}^n)$. Let further $\sigma = (\sigma_1, \ldots, \sigma_n)$ and $\tau = (\tau_1, \ldots, \tau_n)$ be non-increasing sequences of non-negative integers, i.e. $m \geq \sigma_1 \geq \cdots \geq \sigma_n \geq 0$, $m \geq \tau_1 \geq \cdots \geq \tau_n \geq 0$, with $\sigma \preceq \tau$. We define a $(0,1)$-matrix $M_{\sigma,\tau}$ in which the rows are indexed by the elements of $\mathcal{G}(n,\sigma)$ and columns are indexed by the elements of $\mathcal{G}(n,\tau)$. The element $m(S,T)$ which is in the row indexed by $S \in \mathcal{G}(n,\sigma)$ and the column indexed by $T \in \mathcal{G}(n,\tau)$ is defined by

$$m(S,T) = \begin{cases} 1 & \text{if } S \subset T, \\ 0 & \text{if } S \not\subset T. \end{cases}$$

We denote by $\rho(S)$ the row of $M_{\sigma,\tau}(\Omega)$ indexed by the shape σ subspace S. Our goal is to prove the following theorem which is an analog of Kantor's result [75] about the rank of the incidence matrix of dimension s versus dimension t subspaces in $\mathrm{PG}(n,q)$. Since our proof relies on Kantor's theorem, we state it explicitly below.

Theorem 6.3.1. *[75] Let $0 \leq s < t \leq n-s-1$ and let $M_{s,t}$ be the incidence matrix of all s-spaces by all t-spaces of $\mathrm{PG}(n,q)$ or $\mathrm{AG}(n,q)$. Then the rank of $M_{s,t}$ is the number of s-spaces in the geometry.*

The goal of this paper is to prove the following analog of Kantor's result.

Theorem 6.3.2. *Let $\mathfrak{R}$ be a finite chain ring with $|\mathfrak{R}| = q^m$, $\mathfrak{R}/\operatorname{Rad}\mathfrak{R} \cong \mathbb{F}_q$, and let $\Omega = \mathrm{PHG}({}_{\mathfrak{R}}\mathfrak{R}^n)$ Let $\tau = (\tau_1, \ldots, \tau_n)$ be an integer sequence with*

$$m = \tau_1 \geq \tau_2 \geq \ldots \geq \tau_n \geq 0$$

and with $\boldsymbol{m}^s \preceq \tau \preceq \boldsymbol{m}^{n-s}$. Then the rank of $M_{\boldsymbol{m}^s,\tau}(\Omega)$ is equal to the number of shape $(s-1)$-dimensional Hjelmslev subspaces of Ω, i.e. $\begin{bmatrix} n \\ s \end{bmatrix}_{\boldsymbol{m}}$.

This theorem covers the case where the rows of $M_{\sigma,\tau}(\Omega)$ are indexed by free submodules. In the last subsection, we construct an example of an incidence matrix $M_{\sigma,\tau}(\Omega)$ with $\sigma \neq \boldsymbol{m}^s$ (i.e. the subspaces of shape σ are not Hjelmslev subspaces) which is not of full rank over $\mathbb{Q}$.

6.3.2 A special case

Before we start with the proof of Theorem 6.3.2, we mention briefly the case of incidence matrices with rows indexed by the points and columns indexed by the subspaces of shape

τ. Not only does this have an elegant and elementary proof, I also believe the determinant formula used to prove it can have its use as a standalone result.

Let $\tau = (\tau_1, \ldots, \tau_n)$ be a non-increasing sequence of non-negative integers, i.e.

$$m = \tau_1 \geq \cdots \geq \tau_n \geq 0,$$

with $\tau \preceq \boldsymbol{m}^{n-1}$. Given a linear order on the points and on the subspaces of shape τ, we define $M(\tau) = M_{\boldsymbol{m}^1,\tau} = (m_{ij})$.

The size of $M(\tau)$ is $\begin{bmatrix} \boldsymbol{m}^n \\ \boldsymbol{m}^1 \end{bmatrix}_q \times \begin{bmatrix} \boldsymbol{m}^n \\ \tau \end{bmatrix}_q$. We shall fix a particular ordering on the points of $\mathrm{PHG}({}_{\mathfrak{R}}\mathfrak{R}^n)$. First we order linearly the 1-neighbour classes of points, i.e. the elements of $\mathcal{P}^{(1)}$; further we order linearly the 2-neighbour classes of points within each 1-neighbour class. We continue in the same way until we reach a linear order of the elements of $\mathcal{P}^{(m)}$ (which are single points) within each $(m-1)$-neighbour class of points. If our indices start from 0, i.e. our points are $x_0, x_1, \ldots$, then the points x_i and x_j are k-th neighbours if and only if

$$\lfloor \frac{i}{q^{(m-k)(n-1)}} \rfloor = \lfloor \frac{j}{q^{(m-k)(n-1)}} \rfloor. \tag{6.1}$$

Set

$$A := M(\tau) \cdot M^t(\tau).$$

The matrix $A = (a_{ij})$ is a symmetric matrix of order $\begin{bmatrix} \boldsymbol{m}^n \\ \boldsymbol{m}^1 \end{bmatrix}_q$ and has in position (i,j) the number of shape τ subspaces containing the points x_i and x_j.

Let x and y be two points in $\mathrm{PHG}({}_{\mathfrak{R}}\mathfrak{R}^n)$ with $x \hookdownarrow_k y$. Denote by N_k the number of subspaces of $\mathrm{PHG}({}_{\mathfrak{R}}\mathfrak{R}^n)$ of shape $\tau = (\tau_1, \ldots, \tau_n)$ containing x and y. Since the module $\langle x, y \rangle$ has shape $(m, m-k)$, we have $N_k = 0$ if $\tau_2 < m-k$ and $N_k > 0$ if $\tau_2 \geq m-k$. So, if k is the maximal integer for which (6.1) is satisfied, then $a_{ij} = N_k$. Note that $N_m \neq 0$ and that $N_m > N_{m-1} > \cdots > N_0 > 0$ or $N_m > N_{m-1} > \cdots > N_k = \ldots = N_0 = 0$ for some $k \in \{0, \ldots, m-1\}$. The inequalities above are obtained using the remark after Theorem 6.1.13.

We need the following lemma which I proved in [86].

Lemma 6.3.3. *Let n be a positive integer, let $k_0, k_1, \ldots, k_n$ be positive integers with $k_0 = 1$, $k_1 | k_2, \ldots, k_{n-1} | k_n$. Let $b_0, b_1, \ldots, b_n$ be arbitrary elements of a field F and let C be the $k_n \times k_n$ matrix over F given by $c_{ij} = b_{\min\left\{t : \left\lfloor \frac{i}{k_t} \right\rfloor = \left\lfloor \frac{j}{k_t} \right\rfloor\right\}}$, where the rows and columns are labeled from 0 up to $k_n - 1$. Then*

$$\det(C) = \prod_{i=0}^{n} \left(\sum_{j=0}^{i} k_j (b_j - b_{j+1}) \right)^{\frac{k_n}{k_i} - \frac{k_n}{k_{i+1}}},$$

where by convention $a_{n+1} = 0$ and $k_{n+1} = +\infty$.

Proof. Denote this determinant by $D(n, k_1, k_2, \ldots, k_n, a_0, a_1, \ldots, a_n)$. We will first derive a recursive formula for $D(n, *)$ in terms of $D(n-1, *)$ and then solve the recursion.

For each row i with $0 \leq i < k_n$, we can find unique integers q_i and r_i with $0 \leq r_i < k_1$. Similarly, for each column j with $0 \leq j < k_n$, we can find unique integers q'_j and r'_j with $0 \leq r'_j < k_1$. Now, we apply the following row operations, which do not modify the determinant of A. For each row i with $r_i \neq 0$, we will substract row $i - r_i$ from it. The resulting matrix B has the form

$$B_{ij} = \begin{cases} a_{\min\left\{t:\left\lfloor \frac{i}{k_t} \right\rfloor = \left\lfloor \frac{j}{k_t} \right\rfloor\right\}} & \text{if } r_i = 0, \\ a_0 - a_1 & \text{if } r_i \neq 0 \wedge i = j, \\ a_1 - a_0 & \text{if } r_i \neq 0 \wedge r'_j = 0 \wedge q_i = q'_j, \\ 0 & \text{otherwise.} \end{cases}$$

Now, we apply the following set of column operations to B: we add all columns with $r'_j \neq 0$ to column $j - r'_j$. This still leaves the determinant invariant, and the resulting matrix C has the form

$$C_{ij} = \begin{cases} k_1 a_{\min\left\{t:\left\lfloor \frac{i}{k_t} \right\rfloor = \left\lfloor \frac{j}{k_t} \right\rfloor\right\}} & \text{if } r_i = 0 \wedge r'_j = 0 \wedge i \neq j, \\ a_{\min\left\{t:\left\lfloor \frac{i}{k_t} \right\rfloor = \left\lfloor \frac{j}{k_t} \right\rfloor\right\}} & \text{if } r_i = 0 \wedge r'_j \neq 0, \\ (a_0 - a_1) + k_1 a_1 & \text{if } r_i = 0 \wedge i = j, \\ a_0 - a_1 & \text{if } r_i \neq 0 \wedge i = j, \\ 0 & \text{otherwise.} \end{cases}$$

From this form, it can be seen that all rows i in C with $r_i \neq 0$, have only one non-zero entry, in column j with value $a_0 - a_1$. Hence, we can remove all rows and columns with indices not 0 modulo k_1, at a cost of a factor $(a_0 - a_1)^{k_n - \frac{k_n}{k_1}}$ in the determinant. The resulting matrix has dimensions $\frac{k_m}{k_1} \times \frac{k_m}{k_1}$ and is of the same form as the original matrix A. This results in a recursive formula for the determinant D:

$$\begin{aligned} & D(n, k_1, \ldots, k_n, a_0, \ldots, a_n) \\ = \; & (a_0 - a_1)^{k_n - \frac{k_n}{k_1}} D\left(n-1, \tfrac{k_2}{k_1}, \ldots, \tfrac{k_n}{k_1}, (a_0 - a_1) + k_1 a_1, k_1 a_2, \ldots, k_1 a_n\right). \end{aligned}$$

Now, iteratively repeating the recursion formula for D yields in the following result:

$$\begin{aligned} & D(n, k_1, k_2, \ldots, k_n, a_0, a_1, \ldots, a_n) \\ = \; & (a_0 - a_1)^{k_n - \frac{k_n}{k_1}} \cdot D\left(n-1, \frac{k_2}{k_1}, \frac{k_3}{k_1}, \ldots, \frac{k_n}{k_1}, (a_0 - a_1) + k_1 a_1, k_1 a_2, k_1 a_3, \ldots, k_1 a_n\right) \\ = \; & (a_0 - a_1)^{k_n - \frac{k_n}{k_1}} \cdot \left((a_0 - a_1) + k_1(a_1 - a_2)\right)^{\frac{k_n}{k_1} - \frac{k_n}{k_2}} \\ & \cdot D\left(n-2, \frac{k_3}{k_2}, \frac{k_4}{k_2}, \ldots, \frac{k_n}{k_2}, (a_0 - a_1) + k_1(a_1 - a_2) + k_2 a_2, k_2 a_3, \ldots, k_2 a_n\right) \\ = \; & \cdots \\ = \; & (a_0 - a_1)^{k_n - \frac{k_n}{k_1}} \cdot \left((a_0 - a_1) + k_1(a_1 - a_2)\right)^{\frac{k_n}{k_1} - \frac{k_n}{k_2}} \\ & \cdots \left((a_0 - a_1) + k_1(a_1 - a_2) + \cdots + k_{n-1}(a_{n-1} - a_n)\right) \\ & \cdot D(0, (a_0 - a_1) + k_1(a_1 - a_2) + \cdots + k_{n-1}(a_{n-1} - a_n) + k_n a_n) \\ = \; & \prod_{i=0}^{n} \left(\sum_{j=0}^{i} k_j (a_j - a_{j+1})\right)^{\frac{k_n}{k_i} - \frac{k_n}{k_{i+1}}} \end{aligned}$$

since $a_{n+1} = 0$. This concludes the proof. □

Theorem 6.3.4. *Let $\tau = (\tau_1, \ldots, \tau_n)$ be a non-increasing sequence of non-negative integer with $\boldsymbol{m}^1 \preceq \tau \preceq \boldsymbol{m}^{n-1}$. Then the matrix $M(\tau)$ is of full rank over $\mathbb{Q}$.*

Proof. For the matrix $A = M(\tau) \cdot M(\tau)^t$ one has $k_i = q^i$ for $i = 0, \ldots, m-1$, $k_m = \begin{bmatrix} \boldsymbol{m}^n \\ \boldsymbol{m}^1 \end{bmatrix}_q$ and $b_i = N_{m-i}$. Now by Lemma 6.3.3,

$$\det A = \det \left(M(\tau) M^t(\tau) \right) \neq 0.$$

This implies that A is of rank $\begin{bmatrix} \boldsymbol{m}^n \\ \boldsymbol{m}^1 \end{bmatrix}_q$ which in turn gives that the rank of $M(\tau)$ is equal to the number of its rows, i.e. $\begin{bmatrix} \boldsymbol{m}^n \\ \boldsymbol{m}^1 \end{bmatrix}_q$. □

6.3.3 The proof of the main theorem

We start with the case when the modules that index the rows and the columns are indexed by Hjelmslev subspaces, i.e. $\sigma = \boldsymbol{m}^s$ and $\tau = \boldsymbol{m}^t$.

Theorem 6.3.5. *Let $\mathfrak{R}$ be a chain ring with $|\mathfrak{R}| = q^m$, $\mathfrak{R}/\operatorname{Rad}\mathfrak{R} \cong \mathbb{F}_q$, and let $\Omega = \mathrm{PHG}({}_{\mathfrak{R}}\mathfrak{R}^n)$. Let further s and t be integers with $1 \leq s \leq t \leq n - s$. Then the rank of $M_{\boldsymbol{m}^s, \boldsymbol{m}^t}(\Omega)$ is equal to the number of free Hjelmslev subspaces of Ω of dimension $s - 1$, i.e. the rank is equal to $\begin{bmatrix} \boldsymbol{m}^n \\ \boldsymbol{m}^s \end{bmatrix}_q$.*

Proof. We use induction on m. The case $m = 1$ is Kantor's Theorem, i.e. Theorem 6.3.1. Let us assume that the result is proved for all incidence matrices $M_{\boldsymbol{m}^s, \boldsymbol{m}^t}(\Omega')$ where Ω' is an $(n-1)$-dimensional projective Hjelmslev geometry over a chain ring of nilpotency index at most $m - 1$.

Now let $\mathfrak{R}$ be a chain ring with $|\mathfrak{R}| = q^m$, $q = p^h$, $\mathfrak{R}/\operatorname{Rad}\mathfrak{R} \cong \mathbb{F}_q$, and denote $\Omega = \mathrm{PHG}({}_{\mathfrak{R}}\mathfrak{R}^n)$. Consider two $(m-1)$-neighbor classes of Hjelmslev subspaces of shape $\boldsymbol{m}^s$ and $\boldsymbol{m}^t$, say $[S]^{(m-1)} = \{S_1, \ldots, S_u\}$ and $[T]^{(m-1)} = \{T_1, \ldots, T_v\}$, respectively. If some subspace from $[S]^{(m-1)}$ contains a point which is not incident with a subspace from $[T]^{(m-1)}$ then $S_i \not\subset T_j$ for any $i \in \{1, \ldots, u\}$ and any $j \in \{1, \ldots, v\}$. Hence the $u \times v$ submatrix of $M_{\boldsymbol{m}^s, \boldsymbol{m}^t}(\Omega)$ defined by the rows indexed by $S_1, \ldots, S_u$ and the columns indexed by $T_1, \ldots, T_v$ is the all-zero matrix. Otherwise, each subspace of $[S]^{(m-1)}$ is contained in the same number of subspaces from $[T]^{(m-1)}$ and each subspace from $[T]^{(m-1)}$ contains the same number of subspaces from $S^{(m-1)}$. Hence the submatrix of $M_{\boldsymbol{m}^s, \boldsymbol{m}^t}(\Omega)$ with rows indexed by the subspaces from $[S]^{(m-1)}$ and the columns indexed by the subspaces from $[T]^{(m-1)}$ is a $(0,1)$-matrix, B say, with constant row and column sums. For a suitable ordering of all Hjelmslev subspaces of dimension $s-1$ and $t-1$, the matrix $M_{\boldsymbol{m}^s, \boldsymbol{m}^t}(\Omega)$ can be represented in the following block form:

$$M_{\boldsymbol{m}^s, \boldsymbol{m}^t}(\Omega) = (A_{i,j}),$$

where $i = 1, \ldots, x, j = 1, \ldots, y$. Here x and y are the numbers of the $(m-1)$-st neighbor classes of subspaces of dimension $s-1$ and $t-1$. By Theorem 6.1.13, we get

$$x = q^{s(n-s)(m-2)} \begin{bmatrix} n \\ s \end{bmatrix}_q, \; y = q^{t(n-t)(m-2)} \begin{bmatrix} n \\ t \end{bmatrix}_q.$$

If the i-th $(m-1)$-neighbor class of $(s-1)$-dimensional Hjelmslev subspaces is contained in the j-th $(m-1)$-neighbor class of $(t-1)$-dimensional Hjelmslev subspaces in the factor geometry then $A_{i,j}$ is a $u \times v$ matrix of zeros and ones which has the form PBQ for some suitable permutation matrices P and Q of orders u and v, respectively. Otherwise $A_{i,j}$ is the zero matrix. Moreover, the matrix $A = (a_{i,j})$ of size $x \times y$ defined by

$$a_{i,j} = \begin{cases} 1 & \text{if } A_{i,j} \neq \mathbf{0}_{u\times v}, \\ 0 & \text{if } A_{i,j} = \mathbf{0}_{u\times v}, \end{cases}$$

is equivalent to the incidence matrix of free $(s-1)$-dimensional by free $(t-1)$-dimensional Hjelmslev suspaces in the factor geometry $\mathrm{PHG}({}_{\mathfrak{R}}\mathfrak{R}^n\mathfrak{R}/\mathfrak{R}\theta(\mathfrak{R}/\mathfrak{R}\theta)^n)$. Since $\mathfrak{R}/\mathfrak{R}\theta$ has nilpotency index $m-1$, the rank of A is equal to the number of its rows, by the induction hypothesis.

Assume there exists a non-trivial linear combination of the rows of the matrix $M_{\boldsymbol{m}^s,\boldsymbol{m}^t}(\Omega)$:

$$\sum_S a(S)\rho(S) = \sum_{[S]^{(m-1)}} \sum_{L\in[S]^{(m-1)}} a(L)\rho(L) = \mathbf{0}, \tag{6.2}$$

where $a(L)$ are rational numbers not all zero. Define

$$G = \{I + C\theta^{m-1} \mid C \text{ is an } n \times n \text{ matrix over } \Gamma \text{ with 0's on the main diagonal}\},$$

then G is a commutative group under matrix multiplication, and G fixes all $(m-1)$-neighbor classes of points setwise and acts transitively on the points within these classes. Hence the orbits of G on the set of all Hjelmslev subspaces are the $(m-1)$-neighbor classes of Hjelmslev spaces themselves. In particular, this is true for all $(s-1)$-dimensional Hjelmslev subspaces. Thus for every $(s-1)$-dimensional Hjelmslev subspace S we have

$$|G_S| \cdot |S^G| = |G|.$$

Since all orbits S^G have the same size, the stabilizers G_S have also the same size. For an arbitrary $g \in G$, we get from (6.2):

$$\sum_{[S]^{(m-1)}} \sum_{L\in[S]^{(m-1)}} a(L)\rho(L^g) = 0.$$

Let g run over all elements of G. This implies

$$\sum_{g\in G} \sum_{[S]^{(m-1)}} \sum_{L\in[S]^{(m-1)}} a(L)\rho(L^g) = \sum_{[S]^{(m-1)}} \sum_{L\in S} \sum_{g\in G} a(L)\rho(L^g) = \mathbf{0}.$$

If $[L]^{(m-1)} = [M]^{(m-1)}$, the number of elements $g \in G$ for which $L^g = M$ is equal to the size of the stabilizer of L, i.e. $|G_L| = |G|/|L^G|$, and hence is constant for all Hjelmslev subspaces of the same dimension. Therefore, there exist coefficients $b([S]^{(m-1)})$ such that

$$\sum_{[S]^{(m-1)}} b([S]^{(m-1)}) \left(\sum_{L\in[S]^{(m-1)}} \rho(L) \right) = \mathbf{0}.$$

Let the rows of the incidence matrix of $(s-1)$-dimensional by $(t-1)$-dimensional subspaces of $\mathrm{PHG}({}_{\mathfrak{R}}\mathfrak{R}^n\mathfrak{R}/\mathfrak{R}\theta(\mathfrak{R}/\mathfrak{R}\theta)^n)$ be $\boldsymbol{r}_1,\ldots,\boldsymbol{r}_x$. For a suitable ordering of the $(s-1)$-dimensional Hjelmslev subspaces of Ω and of the $(s-1)$-dimensional subspaces of $\mathrm{PG}({}_{\mathfrak{R}}\mathfrak{R}^n\mathfrak{R}/\mathfrak{R}\theta(\mathfrak{R}/\mathfrak{R}\theta)^n)$ we get

$$\sum_{L\in[S_i]^{(m-1)}}\rho(L)=k(\boldsymbol{r}_i\otimes(\underbrace{1,\ldots,1}_{v})).$$

Here k denotes the number of ones in any column of the block B defined above. This implies that

$$\sum_{[S]^{(m-1)}}b([S]^{(m-1)})\sum_{L\in[S]^{(m-1)}}\rho(L)=\sum_{i=1}^{x}b_i\cdot k(\boldsymbol{r}_i\otimes\underbrace{(1,\ldots,1)}_{v})=0,$$

where $b_i=b([S_i]^{(m-1)})k$. Hence

$$\sum_{i=0}^{x}b_i\boldsymbol{r}_i=0,$$

a contradiction since by the induction hypothesis the rows $\boldsymbol{r}_i$ are linearly independent. □

Now the proof of Theorem 6.3.2 is almost immediate.

Proof. (Theorem 6.3.2) Let t be the rank of the smallest free submodule of $\mathfrak{R}^n$ that contains a submodule of shape τ. By $\sigma\preceq\tau\preceq\boldsymbol{m}^n-\sigma$, we get $s\le t\le n-s$. Now we have

$$M_{\boldsymbol{m}^s,\boldsymbol{m}^t}=\alpha M_{\boldsymbol{m}^s,\tau}M_{\tau,\boldsymbol{m}^t},$$

where α is the number of submodules U of shape τ with $S\subset U\subset T$, where S and T are fixed free submodules of ranks s and t, respectively (hence α is a constant). Since $M_{\boldsymbol{m}^s,\boldsymbol{m}^t}$ is of full rank (by Theorem 6.3.5) then $M_{\boldsymbol{m}^s,\tau}$ is also of full rank by Sylvester's inequality. □

6.3.4 A counterexample and concluding remarks

It might be tempting to conjecture that the matrix $M_{\sigma,\tau}(\Omega)$ is always of full rank, i.e. its rank is the smaller of the numbers $\begin{bmatrix}\boldsymbol{m}^n\\ \sigma\end{bmatrix}_q$ and $\begin{bmatrix}\boldsymbol{m}^n\\ \tau\end{bmatrix}_q$. Below we construct an example which demonstrates that this is not always true.

For the sake of simplicity we construct our example over the ring $\mathbb{Z}_4$, but it can be generalized to any chain ring. Take $\mathfrak{R}=\mathbb{Z}_4$ and consider the 3-dimensional Hjelmslev geometry $\Omega=\mathrm{PHG}({}_{\mathfrak{R}}\mathfrak{R}^n\mathfrak{R}\mathfrak{R}^4)$. Set $\sigma=(2,1,0,0)$ and $\tau=(2,2,0,0)$. The shape σ subspaces are line segments consisting of two points each; the shape τ subspaces are the lines of Ω. Using Theorem 6.1.13 we find that the number of shape σ subspaces is 420 while the number of shape τ subspaces is 560 (Theorem 6.1.13).

Let S be a subspace of shape σ in Ω and let T_1,T_2 be Hjelmslev subspaces of shape τ in Ω with $S\subset T_1$, $S\subset T_2$. Clearly T_1 and T_2 are Hjelmslev subspaces of minimal rank containing S. This implies that T_1 and T_2 are neighbors; otherwise the Hjelmslev subspace $T_1\cap T_2$ would contain S which is a contradiction to the minimality of T_1 and T_2. This implies that there exists such ordering of the shape σ and shape τ subspaces that $M_{\sigma,\tau}$ has diagonal block form

with zero-blocks off the main diagonal. Each block has size 12×16 and there are 35 such blocks that correspond to the 35 lines in the factor geometry which happens to be $\mathrm{PG}(3,2)$.

Now the matrix $M_{\sigma,\tau}$ is of full rank if and only if each block is of full rank. Consider a single block B. It corresponds to a neighbor class of lines in Ω. By Theorem 6.1.14, a block is isomorphic to a part of the point by lines incidence matrix of $\mathrm{PG}(3,2)$. The rows are indexed by the twelve points not incident with a fixed line ℓ and the columns are indexed by the 16 lines skew to ℓ. Let π_0, π_1, π_2 be the planes through ℓ and let the points in π_i off ℓ be $P_1^{(i)}, \ldots, P_4^{(i)}$, $i = 0, 1, 2$. Denote by $\rho(P)$ the row in B indexed by the point P. Now it is easily checked that

$$\sum_{j=1}^{4} \rho(P_j^{(0)}) = \sum_{j=1}^{4} \rho(P_j^{(1)}) = \sum_{j=1}^{4} \rho(P_j^{(2)}) = (\underbrace{1, 1, \ldots, 1}_{16}).$$

This means that B is not of full rank and hence $M_{\sigma,\tau}(\Omega)$ is also not of full rank.

It is clear that the same shapes considered in a higher dimensional space over the same ring will give again a matrix which is not of full rank.

By duality, Theorem 6.3.2 implies that in the case of shapes σ and τ with $\boldsymbol{m}^s \preceq \sigma \preceq \boldsymbol{m}^{n-s} = \tau$, the matrix $M_{\sigma,\tau}(\Omega)$ is of full column rank. Presently there is no reasonable conjecture about the shapes $\sigma \prec \boldsymbol{m}^s \preceq \tau \preceq \boldsymbol{m}^{n-s}$, for which the rank of $M_{\sigma,\tau}(\Omega)$ is maximal.

Chapter 7

Miscellaneous results

In this chapter, I collect three miscellaneous research papers which I have published or submitted for publication.

7.1 Generalizing AM-GM and Turkevich's inequality

In this section, we establish a sharp homogeneous inequality, which extends both the classical weighted AM-GM inequality and Turkevich's inequality. This is joint work with Géza Kós and Hojoo Lee [81]. While this is not directly related to finite geometry and coding theory, I decided to put it in my thesis because it is a nice mathematical result. Moreover, it is published in a general mathematics journal (Proceedings of the American Mathematical Society) [81], which probably makes it my most read publication.

We originally came up with this generalization while creating a problem for the International Mathematics Olympiad (IMO). The most well-known inequality in the olympiads, is AM-GM, the inequality between the arithmetic mean and the geometric mean of nonnegative numbers:

Theorem 7.1.1 (AM-GM)**.** *Let $n \geq 2$ and let $a_1, \ldots, a_n \geq 0$. Then*

$$\frac{a_1 + a_2 + \cdots + a_n}{n} \geq \sqrt[n]{a_1 a_2 \cdots a_n}.$$

Equality occurs if and only if $a_1 = a_2 = \cdots = a_n$.

This is a strong and well-known inequality, of which we will present a new generalization in this section, which at the same time generalizes Turkevich's inequality [123].

Theorem 7.1.2 ([123])**.** *Let $a, b, c, d \geq 0$, then*

$$a^4 + b^4 + c^4 + d^4 + 2abcd \geq a^2b^2 + a^2c^2 + a^2d^2 + b^2c^2 + b^2d^2 + c^2d^2$$

or, equivalently,

$$\left(a^2 - b^2\right)^2 + \left(c^2 - d^2\right)^2 \geq \left(a^2 + b^2\right)\left(c^2 + d^2\right) - (ab + cd)^2.$$

Equality occurs if and only if either $a = b = c = d$ or if three of a, b, c, d are equal and the remaining one is zero.

Several generalizations of Turkevich's inequality are known, for example Shleifer's inequality [123] says that, for $a_1, \ldots, a_n \geq 0$,

$$(n-1)\sum_{i=1}^{n} a_i^4 + n\,(a_1 \cdots a_n)^{\frac{4}{n}} \geq \left(\sum_{i=1}^{n} a_n^2\right)^2 .$$

The second equality case in Turkevich's inequality makes it particularly surprising that a simultaneous generalization of Turkevich's inequality and AM-GM exists. In this section we will present such an inequality, as well as prove its sharpness.

7.1.1 Introduction and main results

In the following, let n be a positive integer with $n \geq 2$ and let $\omega_1, \ldots, \omega_n$ be positive real numbers with $\omega_1 + \cdots + \omega_n = 1$. Define $\omega = \min\{\omega_1, \ldots, \omega_n\} > 0$ and denote $\lambda = (1-\omega)^{-\frac{1-\omega}{\omega}} > 1$.

We now present our two main theorems, which will turn out to be equivalent.

Theorem 7.1.3. *Let $a_1, \ldots, a_n$, $b_1, \ldots, b_n$ be non-negative real numbers ($n \geq 2$) and let $\omega_1, \ldots, \omega_n$ be positive weights with $\omega_1 + \cdots + \omega_n = 1$. We have*

$$\lambda \sum_{k=1}^{n} \omega_k (a_k^2 - b_k^2)^2 + \left(2\sum_{k=1}^{n} \omega_k a_k b_k\right)^2 \geq \left(a_1^2 + b_1^2\right)^{2\omega_1} \cdots \left(a_n^2 + b_n^2\right)^{2\omega_n} . \tag{7.1}$$

Equality in (7.1) occurs if and only if we have either $a_1 = \cdots = a_n = b_1 = \cdots = b_n$, or if we have $|a_k^2 - b_k^2| = \begin{cases} a & \text{if } k = i_0 \\ 0 & \text{if } k \neq i_0 \end{cases}$ and $2a_k b_k = \begin{cases} 0 & \text{if } k = i_0 \\ b & \text{if } k \neq i_0 \end{cases}$ for some integer $i_0 \in \{1, \ldots, n\}$ with $\omega_{i_0} = \omega$ and for some $a, b \geq 0$ for which $\lambda a^2 = b^2(1-\omega)$.

The existence of the equality condition guarantees the minimality of the optimal coefficient λ in inequality (7.1). Theorem 7.1.3 is an n-variable generalization of Turkevich's inequality [123]; the original inequality of Turkevich can be obtained by letting $n = 2$ and $\omega_1 = \omega_2 = \frac{1}{2}$, in which case $\lambda = 2$.

To establish Theorem 7.1.3, we will use the following theorem, which is a non-symmetric equivalent to Theorem 7.1.3.

Theorem 7.1.4. *Let $a_1, \ldots, a_n$, $b_1, \ldots, b_n$ be non-negative real numbers ($n \geq 2$) and let $\omega_1, \ldots, \omega_n$ be positive weights with $\omega_1 + \cdots + \omega_n = 1$. Then we have*

$$\lambda \sum_{k=1}^{n} \omega_k a_k^2 + \left(\sum_{k=1}^{n} \omega_k b_k\right)^2 \geq \left(a_1^2 + b_1^2\right)^{\omega_1} \cdots \left(a_n^2 + b_n^2\right)^{\omega_n} . \tag{7.2}$$

Equality in (7.2) occurs if and only if we have either $a_1 = \cdots = a_n = 0$ *and* $b_1 = \cdots = b_n$*, or if we have* $a_k = \begin{cases} a & \text{if } k = i_0 \\ 0 & \text{if } k \neq i_0 \end{cases}$ *and* $b_k = \begin{cases} 0 & \text{if } k = i_0 \\ b & \text{if } k \neq i_0 \end{cases}$ *for some integer* $i_0 \in \{1, \ldots, n\}$ *with* $\omega_{i_0} = \omega$ *and for some* $a, b \geq 0$ *for which* $\lambda a^2 = b^2(1-\omega)$.

Inequality (7.2) is clearly a generalization of the weighted AM-GM inequality, as can be seen by substituting $a_1 = \cdots = a_n = 0$. That it is a strict generalization, can be seen from the additional equality conditions, where $a_1 = \cdots = a_n = 0$ does not necessarily hold.

Several specific estimations on the optimal coefficient λ in Theorems 7.1.3 and 7.1.4 can be made. First, as the following proposition shows, both inequalities (7.1) and (7.2) still hold when replacing λ with Euler's constant e.

Proposition 7.1.5. Let $n \geq 2$. We have $e > \lambda$ for any positive weights $\omega_1, \ldots, \omega_n$ with $\omega_1 + \cdots + \omega_n = 1$.

Secondly, the following proposition indicates that the resulting inequalities are still sharp, in the sense that e cannot be replaced by a smaller constant.

Proposition 7.1.6. Let $n \geq 2$. Suppose that $\mathcal{C}$ is a positive real constant for which

$$\mathcal{C} \sum_{k=1}^{n} \omega_k a_k^2 + \left(\sum_{k=1}^{n} \omega_k b_k \right)^2 \geq \left(a_1^2 + b_1^2\right)^{\omega_1} \cdots \left(a_n^2 + b_n^2\right)^{\omega_n} \tag{7.3}$$

holds for all positive weights $\omega_1, \ldots, \omega_n$ with $\omega_1 + \cdots + \omega_n = 1$ and for all nonnegative real numbers $a_1, \ldots, a_n$, $b_1, \ldots, b_n$. Then $\mathcal{C} \geq e$.

If $\omega_1 = \cdots = \omega_n = \frac{1}{n}$, we have $\lambda = \left(1 + \frac{1}{n-1}\right)^{n-1}$. This gives our inequalities simple forms for the uniform weight distribution $\omega_1 = \cdots = \omega_n = \frac{1}{n}$, and it is sharper than replacing $\lambda = \left(1 + \frac{1}{n-1}\right)^{n-1}$ by Euler's constant e.

Theorems 7.1.3 and 7.1.4 are the main theorems of this section. In Section 7.1.2, we present a proof of our main theorems, as well as a proof for the propositions above.

7.1.2 Proof of the main theorems and the propositions

In this section we give the proof of our main theorems. First we introduce a useful notation and we present an observation on the minimal optimal coefficient λ. Given a proper subset I of $\{1, \ldots, n\}$, we denote

$$\lambda_I = \left(\sum_{i \notin I} \omega_i \right)^{-\frac{\sum_{i \notin I} \omega_i}{\sum_{i \in I} \omega_i}} = f\left(\sum_{i \in I} \omega_i \right),$$

where we define $f(x) = (1-x)^{-\frac{1-x}{x}}$. We then recall the definitions from the precious subsection:

$$\omega = \min\{\omega_1, \ldots, \omega_n\} > 0 \text{ and } \lambda = f(\omega) = (1-\omega)^{-\frac{1-\omega}{\omega}} > 1.$$

Since the function f is decreasing on $]0,1[$, we have that $\lambda_I \leq \lambda$ for each proper subset $I \subset \{1, \ldots, n\}$. In particular, because the function f is decreasing,

$$\lambda = \max\{\lambda_I \,|\, I \text{ is a proper subset of } \{1, \ldots, n\}\}$$

and this maximum is attained when $\sum_{i \in I} \omega_i$ is minimal, i.e. when $I = \{i_0\}$, where i_0 is any index for which $\omega_{i_0} = \omega$. This *maximality* of the *minimal* optimal coefficient $\lambda = f(\omega)$ is crucial to the proof of Theorem 7.1.4. We start by proving Theorem 7.1.4.

Proof of Theorem 7.1.4. Let $p_i = \sqrt{a_i^2 + b_i^2}$ for all integers i, with $1 \leq i \leq n$. If there is any integer i, with $1 \leq i \leq n$, for which $p_i = 0$, then the right hand side equals 0 and the inequality holds trivially. In this case equality occurs if and only if $a_1 = \cdots = a_n = b_1 = \cdots = b_n = 0$.

Hence we may assume that $p_i > 0$ for all integers i, $1 \leq i \leq n$. We can re-write the claimed estimation as

$$\lambda \sum_{k=1}^{n} \omega_k (p_k^2 - b_k^2) + \left(\sum_{k=1}^{n} \omega_k b_k \right)^2 \geq p_1^{2\omega_1} \cdots p_n^{2\omega_n}.$$

If we now fix the variables $p_1, \ldots, p_n$, $b_1, \ldots, b_{i-1}$ and $b_{i+1}, \ldots, b_n$, for some integer i, with $1 \leq i \leq n$, then we find that the right hand side is a constant, while the left hand side is a quadratic function of b_i with leading coefficient $\omega_i(\omega_i - \lambda)$. Since $\lambda > 1 > \omega_i > 0$, this leading coefficient is negative, thus the left hand side is a concave function in the variable b_i. Therefore, the smallest value of the left hand side is attained either when $b_i = 0$ or $b_i = p_i$. Since this holds for any integer i, with $1 \leq i \leq n$, we may assume that $b_i \in \{0, p_i\}$ for each integer i, with $1 \leq i \leq n$.

Let m be the number of integers i, with $1 \leq i \leq n$, for which $b_i = 0$. We may permute the indices such that $b_1 = b_2 = \cdots = b_m = 0$ and $b_{m+1} = p_{m+1} > 0, \ldots, b_n = p_n > 0$; we denote this permutation by σ. With these observations, it is sufficient to prove the following inequality, for arbitrary positive weights $\omega_1, \ldots, \omega_n$ with $\omega_1 + \cdots + \omega_n = 1$ and arbitrary positive reals $p_1, \ldots, p_n$:

$$\lambda \sum_{k=1}^{m} \omega_k p_k^2 + \left(\sum_{k=m+1}^{n} \omega_k p_k \right)^2 \geq p_1^{2\omega_1} \cdots p_n^{2\omega_n}. \tag{7.4}$$

Now there are three cases: either $m = 0$, $m = n$, or $1 \leq m \leq n-1$. If $m = 0$, then (7.4) is simply the AM-GM inequality for $p_1, \ldots, p_n$. Equality hence occurs if and only if $p_1 = \cdots = p_n$, which in the original problem can be written as $a_1 = \cdots = a_n = 0$ and $b_1 = \cdots = b_n$.

If $m = n$, then

$$\lambda \sum_{k=1}^{n} \omega_k p_k^2 > \sum_{k=1}^{n} \omega_k p_k^2 \geq p_1^{2\omega_1} \cdots p_n^{2\omega_n},$$

by the AM-GM inequality for $p_1^2, \ldots, p_n^2$. Equality cannot be attained in this case.

Hence, we are left with the case $1 \leq m \leq n-1$. Define

$$U = \omega_1 + \cdots + \omega_m, \quad V = \omega_{m+1} + \cdots + \omega_n,$$

$$A = (p_1^{\omega_1} \cdots p_m^{\omega_m})^{1/U} \text{ and } B = \left(p_{m+1}^{\omega_{m+1}} \cdots p_n^{\omega_n}\right)^{1/V}.$$

Applying the weighted AM-GM inequality twice to the left hand side then yields

$$\lambda \sum_{k=1}^{m} \omega_k p_k^2 + \left(\sum_{k=m+1}^{n} \omega_k p_k \right)^2 \geq \lambda \cdot UA^2 + (VB)^2.$$

On the other hand, using the same notations, the right hand side of (7.4) can be written as $p_1^{2\omega_1} \cdots p_n^{2\omega_n} = A^{2U} B^{2V}$ and hence we are left to prove that

$$\lambda \cdot UA^2 + \left(VB\right)^2 \geq A^{2U} B^{2V}.$$

Now, let $I = \{\sigma^{-1}(1), \ldots, \sigma^{-1}(m)\}$ in the original definition of λ_I, then at this point in the proof (after rearranging our indices) we have $\sigma(I) = \{1, 2, \ldots, m\}$. Hence, $\lambda_{\sigma(I)} = (1-U)^{-\frac{1-U}{U}} = f(U)$. Then, the maximality of $\lambda = f(\omega)$ implies

$$\lambda \geq \lambda_{\sigma(I)} = (1-U)^{-\frac{1-U}{U}} = \left(\frac{1}{V}\right)^{\frac{V}{U}}.$$

Finally, we can combine this with the weighted AM-GM inequality to deduce

$$\begin{aligned} \lambda \cdot UA^2 + (VB)^2 &\geq \left(\tfrac{1}{V}\right)^{V/U} \cdot UA^2 + (VB)^2 \\ &= U \cdot \left(\tfrac{A^2}{V^{V/U}}\right) + V \cdot (VB^2) \\ &\geq \left(\tfrac{A^2}{V^{V/U}}\right)^U \cdot (VB^2)^V \\ &= A^{2U} B^{2V} \end{aligned}$$

as claimed. This proves inequality (7.2).

Equality in the above occurs only if $\lambda = \lambda_{\sigma(I)} = \left(\frac{1}{V}\right)^{\frac{V}{U}}$ and $\lambda A^2 = VB^2$. Filling in the definitions of U and V, we see that $\lambda = \lambda_{\sigma(I)}$ implies that $\sigma(I) = \{i_0\}$ with $\omega_{i_0} = \omega$. Hence, this is exactly the claimed equality condition; this proves the 'only if' part. For the 'if' part, let $I = \{i_0\}$ and let a, b be nonnegative real numbers satisfying the given conditions. Denoting $u = \sum_{k \in I} \omega_k = \omega$ and $v = 1 - u = \sum_{k \notin I} \omega_k = 1 - \omega$, we have $\lambda = \lambda_I = v^{-v/u}$ and we have to show that $v^{-v/u} u a^2 + v^2 b^2 = a^{2u} b^{2v}$, which is equivalent to $u\left(\frac{a^2}{v^{v/u}}\right) + v(vb^2) = a^{2u} b^{2v}$. Since we are given that $\lambda_I a^2 = b^2 \sum_{k \notin I} \omega_k$, we know that $\frac{a^2}{v^{v/u}} = b^2 v$, yielding

$$u\left(\frac{a^2}{v^{v/u}}\right) + v(vb^2) = vb^2 = (vb^2)^u \cdot (vb^2)^v = \left(\frac{a^2}{v^{v/u}}\right)^u \cdot (vb^2)^v = a^{2u} b^{2v}.$$

Hence the statement about the equality condition follows. □

We have proven Theorem 7.1.4. Theorem 7.1.3 is a straightforward corollary now.

Proof of Theorem 7.1.3. For each integer i, with $1 \leq i \leq n$, we substitute (a_i, b_i) by $(|a_i^2 - b_i^2|, 2a_ib_i)$ in inequality (7.2). Then inequality (7.2) in Theorem 7.1.4 reduces to inequality (7.1) in Theorem 7.1.3. □

Now we prove the propositions from Section 7.1.1.

Proof of Proposition 7.1.5. We use the inequality $e^t > 1 + t$ for $t > 0$ to deduce

$$\lambda = (1-\omega)^{-\frac{1-\omega}{\omega}} = \left(\frac{1}{1-\omega}\right)^{\frac{1-\omega}{\omega}} = \left(1 + \frac{\omega}{1-\omega}\right)^{\frac{1-\omega}{\omega}} < \left(e^{\frac{\omega}{1-\omega}}\right)^{\frac{1-\omega}{\omega}} = e,$$

as claimed. □

Proof of Proposition 7.1.6. Substituting $\omega_1 = \cdots = \omega_n = \frac{1}{n}$, $b_1 = a_2 = \cdots = a_n = 0$, $a_1 = \left(1 - \frac{1}{n}\right)^{\frac{n}{2}}$ and $b_2 = \cdots = b_n = 1$ in inequality (7.3), yields

$$\mathcal{C}\left(1 - \frac{1}{n}\right)^n + \left(\frac{n-1}{n}\right)^2 \geq 1 - \frac{1}{n},$$

or equivalently,

$$\mathcal{C} \geq \left(1 + \frac{1}{n-1}\right)^{n-1}.$$

Taking the limit for $n \to +\infty$, we meet the desired estimation $\mathcal{C} \geq e$. □

7.2 Large weight code words for $\mathrm{PG}(n, q)$

In this section, we will focus on the large weights of the linear codes and dual linear codes arising from finite projective spaces. This subsection is joint work with Jira Limbupasiriporn and Leo Storme, and the results were published in [94].

7.2.1 Introduction and preliminaries

For several applications, it can be useful to not only study the code words of smallest weight, but also those of largest weight. It turns out that major differences between the results for q even and for q odd arise.

- For q even, the study of large weight code words in $C_k(n, q)^\perp$ reduces to the theory of minimal blocking sets with respect to the k-spaces of $\mathrm{PG}(n, q)$, odd-blocking the k-spaces. This shows that the maximum weight is equal to $q^n + \cdots + q^{n-k+1}$.

- For q odd, in a lot of cases, the maximum weight of the code $C_k(n,q)^\perp$ is equal to $q^n + \cdots + q + 1$, but some exceptions arise to this result. In particular, the maximum weight of the code $C_1(n,3)^\perp$ is equal to $3^n + 3^{n-1}$. In general, the problem of whether the maximum weight of the code $C_k(n,3)^\perp$ is equal to $3^n + \cdots + 3 + 1$ reduces to the problem of the existence of sets in PG$(n,3)$ intersecting every k-space in 2 (mod 3) points. For $k > n/2$, such sets intersecting every k-space in 2 (mod 3) points trivially exist as the union of two disjoint $(n-k)$-spaces intersects every k-space in 2 (mod 3) points. For $k = 1$, such sets do not exist and for $2 \leq k \leq n/2$, the existence of such sets is an open problem.

Notation 7.2.1. We denote by PG(n,q) the n-dimensional projective space over the finite field $\mathbb{F}_q$. For $n = 2$, we call this a *projective plane* and write PG$(2,q)$. The point set of PG(n,q) is denoted by $\mathcal{P}$.

Definition 7.2.2. A set $S \subseteq \mathcal{P}$ in PG(n,q) is called a *blocking set with respect to the k-spaces* if every k-space contains at least one point of S. If it is clear from the context what k is, we will simply call S a *blocking set*. If there is no $s \in S$ such that $S \setminus \{s\}$ is also a blocking set, then S is called *minimal*. If every k-space contains an odd number of points of S, then we say that S is *odd-blocking* the k-spaces.

Definition 7.2.3. An element of the vector space $\mathbb{F}_p^{\mathcal{P}}$, which consists of the mappings $\mathcal{P} \to \mathbb{F}_p$, can be seen as a vector of length $|\mathcal{P}|$ consisting of elements of $\mathbb{F}_p$. For a given subset $\pi \subseteq \mathcal{P}$, let v^π be its *characteristic function*; this is a $\{0,1\}$ mapping which is 1 for points in π and 0 for points outside of π. This vector v^π is called the *incidence vector* of π. Often, we will identify π with its incidence vector and write π instead of v^π. The support supp(c) of an element $c \in \mathbb{F}_p^{\mathcal{P}}$ is the set of points which is mapped to a nonzero element of $\mathbb{F}_p$.

Notation 7.2.4. The code $C_k(n,q)$, $q = p^h$ with p prime and $h \geq 1$, is the linear code over $\mathbb{F}_p$ generated by the incidence vectors of the k-dimensional subspaces of PG(n,q). Its dual, the code $C_k(n,q)^\perp$, is then the set of vectors $c \in \mathbb{F}_p^{\mathcal{P}}$ with $c \cdot v^\pi = 0$ (over $\mathbb{F}_p$) for each k-space π, where $\cdot$ denotes the standard inner product. In other words, a vector $c \in \mathbb{F}_p^{\mathcal{P}}$ belongs to $C_k(n,q)^\perp$ if and only if $\sum_{r \in \pi} c_r = 0$ for every k-space π of PG(n,q).

Definition 7.2.5. A t (mod p) *set* with respect to the k-spaces of PG(n,q), with $q = p^h$ and p prime, is a set S which intersects every k-space of PG(n,q) in t (mod p) points. By convention, we let $0 \leq t \leq p-1$.

7.2.2 The case q even

In this section, we will study the code words of large weight in $C_k(n,q)^\perp$, when q is even. We are studying a binary code, hence a code word is uniquely identified by its support. In particular, the support supp(c) of a code word $c \in C_k(n,q)^\perp$ of large weight corresponds to a large set of points, intersecting every k-space in an even number of points. Since every k-space contains an odd number of points, the complement S of this set is a small set which intersects every k-space in an odd number of points. In particular, S contains at least one point of every k-space, hence it is a blocking set with respect to the k-spaces.

Theorem 7.2.6. *The maximum weight of $C_k(n,q)^\perp$, q even, is $q^n + \cdots + q^{n-k+1}$, and all the code words of this weight are the incidence vector of the complement of an $(n-k)$-space of* $\mathrm{PG}(n,q)$.

Proof. The incidence vector of the complement of any $(n-k)$-space π is a code word of weight $q^n + \cdots + q^{n-k+1}$ of $C_k(n,q)^\perp$: since each projective k-space intersects this $(n-k)$-space in a nonempty projective subspace, this intersection contains 1 (mod q) points, and hence its complement in π contains 0 (mod q) points. Therefore, the maximum weight of $C_k(n,q)^\perp$ is at least $q^n + \cdots + q^{n-k+1}$.

Since the complement S of the support of a code word of $C_k(n,q)^\perp$ is a blocking set with respect to the k-spaces, we have $|S| \geq q^{n-k} + \cdots + q + 1$ by the Bose-Burton theorem [19], and if equality occurs, then S is an $(n-k)$-space. This shows that the bound is sharp, and it characterizes the code words of weight $q^n + \cdots + q^{n-k+1}$. □

The Bose-Burton result on blocking sets is crucial in the proof of Theorem 7.2.6, and this is not the only place where we will run into a connection with blocking sets. The following theorem improves [130, Theorem 3.1] for $p = 2$, for the special case of odd-blocking sets.

Theorem 7.2.7. *Let S be a set of projective points, odd-blocking the k-spaces of* $\mathrm{PG}(n,q)$, *q even. If $|S| \leq 2(q^{n-k} + q^{n-k-1} + \cdots + q + 1)$, then S is a minimal blocking set.*

Proof. Assume by contraposition that S is not minimal, i.e. there is a point $p \in S$ such that $S \setminus \{p\}$ still blocks the k-spaces. Hence, every k-space through p is blocked by S in at least 2 points. But it is also blocked by an odd number of points of S, so every k-space through p contains at least 3 points of S.

Now, there are two cases:

- either there exists a $(k-1)$-space π through p which contains no other points of S. Every k-space through π contains by assumption at least two other points of S, hence each of the $q^{n-k} + q^{n-k-1} + \cdots + q^2 + q + 1$ different k-spaces through π contains at least 2 points of S outside of π. Since two such k-spaces only intersect in π, this means that $|S| \geq 1 + 2(q^{n-k} + q^{n-k-1} + \cdots + q^2 + q + 1)$, a contradiction.

- either every $(k-1)$-space through p contains at least one other point of S. Let now i be the largest integer (with necessarily $i < k-1$) for which there exists an i-space through p which contains no other points of S. Such an integer i must exist, since $i = 0$ clearly has this property and $i = k-1$ does not. Let now π be such an i-space through p, containing no other points of S. Because of the maximality of i, each of the $q^{n-i-1} + q^{n-i-2} + \cdots + q^2 + q + 1$ different $(i+1)$-spaces through π must again contain at least one other point of S. Since two such $(i+1)$-spaces only intersect in π, this means that $|S| \geq 1 + (q^{n-i-1} + q^{n-i-2} + \cdots + q^2 + q + 1)$. Since $i \leq k-2$ and $q \geq 2$,

this implies that

$$\begin{aligned} |S| &\geq 1+(q^{n-i-1}+q^{n-i-2}+\cdots+q^2+q+1) \\ &\geq 1+(q^{n-k+1}+q^{n-k}+\cdots+q^2+q+1) \\ &\geq 1+(2q^{n-k}+2q^{n-k-1}+\cdots+2q+2+1) \\ &> 2(q^{n-k}+q^{n-k-1}+\cdots+q+1), \end{aligned}$$

a contradiction.

Both cases lead to a contradiction and hence S must be minimal. □

The preceding theorem implies that the study of the code words in $C_k(n,q)^\perp$, q even, of weight larger than or equal to $q^n+\cdots+q^{n-k+1}-q^{n-k}-\cdots-q-1$ is reduced to the study of the minimal blocking sets with respect to the k-spaces of PG(n,q), odd-blocking the k-spaces. Some important results on minimal blocking sets with respect to the k-spaces of PG(n,q) were obtained by Szőnyi [132], Szőnyi and Weiner [130], and Sziklai [129].

Let $S(q)$ be the set of possible sizes of minimal blocking sets in PG$(2,q)$ with cardinality smaller than $\frac{3}{2}(q+1)$, then [129, Corollary 5.1 and 5.2] yield the following summarizing theorem for q even.

Theorem 7.2.8. *Let c be a code word of the code $C_k(n,q)^\perp$, q even, of weight larger than $q^n+\cdots+q+1-\sqrt{2}q^{n-k}$. Then the weight of c equals $q^n+\cdots+q+1-x$, with $x\in S(q^{n-k})$. Moreover, c is the incidence vector of the complement of a small minimal blocking set, odd-blocking the k-spaces.*

Regarding larger minimal blocking sets with respect to the k-spaces of PG(n,q), not many results are known. Here, there are still many open problems, including results on the cardinalities of these minimal blocking sets.

7.2.3 Large weight constructions

From now on, we will assume that q is odd. We consider the p-ary linear code of points and k-spaces of PG(n,q), with $q=p^h$ and with $p>2$ prime. A code word of the code $C_k(n,q)^\perp$ corresponds to a map φ from the set of projective points to $\mathbb{F}_p$, such that for each k-space Π we have $\sum_{p\in\Pi}\varphi(p)=0$ as an element of $\mathbb{F}_p$. The image of a point under φ is called the coefficient of that point.

In this section, we try to determine when the maximum possible Hamming weight of this code is attained, i.e. when there exist code words of weight $q^n+\cdots+q+1$. In case this does not work, we provide constructions to attain sharp lower bounds on the maximum weight of these codes. Surprisingly, we will again find several strong links with small minimal blocking sets. We begin with a useful lemma.

Lemma 7.2.9. *Let $\{B_i\}_{i\in I}$ be a family of $1 \pmod p$ sets with respect to the k-spaces of PG(n,q), such that no point is contained in more than $p-1$ of these sets. Then the maximum*

weight of $C_k(n,q)^\perp$ *is at least*

$$q^n + q^{n-1} + \cdots + q + 1 - \left|\bigcap_{i\in I} B_i\right|.$$

Proof. For each $i \in I$, define $c^{(i)}$ to be the incidence vector of the complement of B_i. Since B_i intersects every k-space in 1 (mod p) points, and every k-space has 1 (mod p) points itself, the complement of B_i intersects every k-space in 0 (mod p) points and hence $c^{(i)}$ is a code word of $C_k(n,q)^\perp$.

Now, $c := \sum_{i=0}^{p-2} c^{(i)}$ is a code word of $C_k(n,q)^\perp$, of which we will now determine its weight. The coefficient in c of each point consists of a sum of $p-1$ elements, and each element is either 0 or 1. Hence, a zero coefficient in the sum c cannot be obtained by summing up ones. Therefore, if a point has zero coefficient in the sum c, it has to be zero in each $c^{(i)}$, which means that it should lie in each of the sets B_i. Therefore, the weight of c is exactly $q^n + q^{n-1} + \cdots + q + 1 - \left|\bigcap_{i\in I} B_i\right|$, as claimed. □

The easiest example of a small minimal blocking set with respect to the k-spaces, is an m-space with $m \geq n-k$. This yields us the following lower bounds on the maximum weight.

Theorem 7.2.10. *The maximum weight of* $C_k(n,q)^\perp$, $q = p^h$, p *prime,* $h \geq 1$, *is*

- *exactly* $q^n + q^{n-1} + \cdots + q + 1$ *if* $(n+1)/k \leq p-1$,
- *at least* $q^n + q^{n-1} + \cdots + q^{n-k(p-1)+1}$ *if* $(n+1)/k > p-1$.

Proof. Let $m := \left\lceil \frac{n+1}{k} \right\rceil$. Define as follows subspaces $H_0, \ldots, H_{m-1}$ of $\mathrm{PG}(n,q)$. For $i = 0, 1, \ldots, m-2$, let H_i be the $(n-k)$-space with equations $X_{ik} = X_{ik+1} = \cdots = X_{(i+1)k-1} = 0$. Let H_{m-1} be the $k(m-1)$-space with equations $X_{k(m-1)} = X_{k(m-1)+1} = \cdots = X_n = 0$.

If $(n+1)/k \leq p-1$, then $S := \{H_0, \ldots, H_{m-1}\}$ is a set of 1 (mod p) sets with respect to the k-spaces. The intersection of all sets in S is trivial, because the coordinates $(X_0, \ldots, X_n)$ of any point in $\bigcap_{i=0}^{m-1} H_i$ must have $X_0 = \cdots = X_{k-1} = 0$, $X_k = \cdots = X_{2k-1} = 0$, …, $X_{k(m-1)} = X_{k(m-1)+1} = \cdots = X_n = 0$ and hence it is the zero vector, which is not a point of $\mathrm{PG}(n,q)$. Since there are only $\lceil (n+1)/k \rceil \leq p-1$ sets in S, each point is indeed contained in at most $p-1$ sets of S. Lemma 7.2.9 yields the desired result.

If $(n+1)/k > p-1$, then $S := \{H_0, \ldots, H_{p-2}\}$ is a set of 1 (mod p) sets with respect to the k-spaces. Since S only contains $p-1$ sets, each point is contained in at most $p-1$ sets of S. The intersection of all sets in S consists of all points $(X_0, \ldots, X_n)$ for which $X_0 = \cdots = X_{k-1} = 0$, $X_k = \cdots = X_{2k-1} = 0$, …, $X_{k(p-2)} = \cdots = X_{k(p-1)-1} = 0$. This is a projective subspace of dimension $n - k(p-1)$ in $\mathrm{PG}(n,q)$, which has $q^{n-k(p-1)} + q^{n-k(p-1)-1} + \cdots + q + 1$ points. Lemma 7.2.9 yields the desired result. □

If $(n+1)/k \leq p-1$, then a maximum weight of $q^n + q^{n-1} + \cdots + q + 1$ is reached. If $(n+1)/k > p-1$, the contrary is not necessarily true. For example, we have the following

sufficient condition for the maximum weight $q^n + q^{n-1} + \cdots + q + 1$ to appear, based on $t \pmod p$ sets.

Theorem 7.2.11. *If a $t \pmod p$ set exists with respect to the k-spaces of* PG(n, q)*, with $t \not\equiv 0, 1 \pmod p$, then the maximum weight of $C_k(n, q)^\perp$ is $q^n + q^{n-1} + \cdots + q + 1$.*

Proof. Let S be such a set and let T be its complement. Assign coefficient 1 to all points in T and assign coefficient $1 - t^{-1}$ to all points in S, where the inversion of t is done over $\mathbb{F}_p$. We will show that this defines a code word of $C_k(n, q)^\perp$. Since we are given that every k-space intersects S in $t \pmod p$ points and T in $p + 1 - t \pmod p$ points, the sum of all coefficients in every k-space is $t \cdot (1 - t^{-1}) + (p + 1 - t) \cdot 1 \equiv 0 \pmod p$, so c is a code word of $C_k(n, q)^\perp$. Since 1 and $1 - t^{-1}$ are nonzero elements of $\mathbb{F}_p$, c has full weight, as claimed. □

Sometimes the existence of $t \pmod p$ sets is trivial, for example when $k \geq \frac{n+1}{2}$.

Corollary 7.2.12. *If $k \geq \frac{n+1}{2}$, two skew $(n - k)$-spaces exist in* PG(n, q) *and hence both Theorem 7.2.10 and Theorem 7.2.11 show that a maximum weight of $q^n + q^{n-1} + \cdots + q + 1$ is attained for $C_k(n, q)^\perp$.*

In other cases it is however not at all obvious. Even in the planar case (where $n = 2$ and $k = 1$), this is not trivial. Since $\frac{n+1}{k} = 3$, the maximum weight is attained for $p \geq 5$ by Theorem 7.2.10, but for $p = 3$, no such easy construction is known.

Lemma 7.2.13. *If $q = 3^h$, where $h > 1$, then there exists a non-square element in $\mathbb{F}_q \setminus \{x^2 - x | x \in \mathbb{F}_q\}$.*

Proof. Let f be the mapping of $\mathbb{F}_q$ into itself defined by $f(x) = x^2 - x$ for all $x \in \mathbb{F}_q$. Then $f(1 - x) = (1 - x)^2 - (1 - x) = x^2 - x$ for all $x \in \mathbb{F}_q$, so we have $f(x) = f(1 - x)$ for all $x \in \mathbb{F}_q$. Observe that for any $x \in \mathbb{F}_q$, $1 - x = x$ if and only if $2x = 1$, i.e. if and only if $x = 2$. Thus the cardinality of $Im(f)$ is $\frac{q-1}{2} + 1 = \frac{q+1}{2}$.

We will show that there exists an element in $\mathbb{F}_q \setminus Im(f)$ which is non-square. Suppose, to the contrary, that every non-square element of $\mathbb{F}_q$ belongs to $Im(f)$. Then $Im(f)$ is the set of zero and all non-square elements of $\mathbb{F}_q$. Let $x \in \mathbb{F}_q \setminus \mathbb{F}_3$ and $y = 2 - x$. Then

$$\begin{aligned} f(x)f(y) &= (x^2 - x)(y^2 - y) = (xy)^2 - xy(y + x) + xy \\ &= (xy)^2 - 2xy + xy = (xy)^2 - xy \\ &= f(xy). \end{aligned}$$

Since both $f(x)$ and $f(y)$ are non-square, it follows that $f(x) = \omega^i$ and $f(y) = \omega^j$ for some odd integers i and j, where ω is a primitive element for $\mathbb{F}_q$. Hence, $i + j$ is even and $f(xy) = f(x)f(y) = \omega^{i+j}$ is square, contradiction. □

Lemma 7.2.14 ([59, Lemma 13.8])**.** *In* PG$(2, q)$*, where $q = 3^h$, the set $\{(1, x, x^3) \mid x \in \mathbb{F}_q\} \cup \{(0, x, x^3) \mid x \in \mathbb{F}_q \setminus \{0\}\}$ is a minimal blocking set which intersects every line in 1 $\pmod 3$ points.*

Theorem 7.2.15. *If $q = 3^h$, where $h > 1$, then* $\mathrm{PG}(2, q)$ *contains a* 2 (mod 3) *set of size* $3q - 1$.

Proof. By Lemma 7.2.13, there exists a non-square element b in $\mathbb{F}_q \setminus \{x^2 - x \mid x \in \mathbb{F}_q\}$. Consider the mapping $\varphi : (x, y, z) \mapsto (z, y + bx, x)$ from $\mathrm{PG}(2, q)$ into itself. This is a collineation of $\mathrm{PG}(2, q)$.

Now let

$$S = \{(1, x, x^3) \mid x \in \mathbb{F}_q\} \cup \{(0, x, x^3) \mid x \in \mathbb{F}_q \setminus \{0\}\}.$$

Clearly, all points in the first set are distinct and disjoint from the second set, hence this part contains q points of S. For the second part, points may coincide since projective points are only defined up to a nonzero scalar multiple. In particular, one has $(0, x, x^3) = (0, y, y^3)$ if and only if $\frac{x^3}{x} = \frac{y^3}{y}$, hence if and only if $\left(\frac{x}{y}\right)^2 = 1$. Therefore, each point appears twice in this second set, making the total cardinality of S equal to $q + \frac{q-1}{2}$.

By Theorem 7.2.14, S is a blocking set intersecting every line in 1 (mod 3) points, and hence, so is $T = \varphi(S)$. Note that

$$T = \{(x^3, x + b, 1) \mid x \in \mathbb{F}_q\} \cup \{(x^3, x, 0) \mid x \in \mathbb{F}_q \setminus \{0\}\}.$$

Now we look at the union of S and T. If S and T are disjoint, then the union of these sets gives a 2 (mod 3) set of cardinality $2\left(q + \frac{q-1}{2}\right) = 3q - 1$. We will show that S and T are disjoint. Suppose by contradiction that there exists a point P in the intersection of S and T. Clearly, this is impossible in all but the following cases.

- If P belongs to the first sets of S and T, i.e. $P = (1, x, x^3) = (y^3, y + b, 1)$ for some $x, y \in \mathbb{F}_q \setminus \{0\}$, then since $(1, x, x^3) = (y^3, xy^3, (xy)^3)$, we obtain the equation $xy^3 = y + b$ and $(xy)^3 = 1$, and the latter implies that $xy = 1$, which gives $y^2 = y + b$ or $b = y^2 - y$, a contradiction.
- If P belongs to the second set of S and to the first set of T with zero element in $\mathbb{F}_q$, i.e. $P = (0, x, x^3) = (0, b, 1)$ for some $x \in \mathbb{F}_q \setminus \{0\}$, then since $(0, x, x^3) = (0, x^{-2}, 1)$, it follows that $b = x^{-2}$ which contradicts the fact that b is non-square.

Hence, S and T are disjoint, and the result follows. □

So, the plane code $C_1(2, q)^\perp$, with $q > 3$ odd, indeed has maximum weight $q^2 + q + 1$.

7.2.4 Upper bounds on the maximum weight

In this section, we will provide some upper bounds on the maximum weight. From the preceding section, one might get the feeling that the study for q odd is not really interesting, as one always attains the maximum weight, or gets at least very close to the maximum

weight. However, this is not correct, as we will now reveal upper bounds which show quite a gap relative to $q^n + q^{n-1} + \cdots + q + 1$.

First we show that if the characteristic of the field is 3, then the converse of Theorem 7.2.11 holds as well.

Theorem 7.2.16. *If $p = 3$, the maximum weight $q^n + q^{n-1} + \cdots + q + 1$ is attained in $C_k(n, q)^\perp$ if and only if there exists a 2 (mod 3) set with respect to the k-spaces of* PG(n, q).

Proof. The 'if' part follows from Theorem 7.2.11. For the 'only if' part, let c be a code word of weight $q^n + q^{n-1} + \cdots + q + 1$ in $C_k(n, q)^\perp$. Let S be the set of points with coefficient 1 in c and let T be its complement, i.e. the set of points with coefficient 2 in c. Now fix an arbitrary k-space π. Let s and t be respectively the number of points of S and T in π. Clearly, $s + t \equiv 1 \pmod p$. Moreover, since c is a code word, $s + 2t \equiv 0 \pmod p$. Solving this, we get $s \equiv t \equiv 2 \pmod p$, i.e., S and T are 2 (mod 3) sets with respect to the k-spaces of PG(n, q). □

For $q = 3$, this yields a negative result.

Lemma 7.2.17. *The projective plane* PG$(2, 3)$ *does not have a 2 (mod 3) set with respect to the lines.*

Proof. Let $q = 3$. Clearly, a 2 (mod 3) set S has two points on every line. In particular, let $r \notin S$, then each of the 4 lines through r contains two points of S, i.e. $|S| = 8$. However, the complement T of S is also a 2 (mod 3) set, i.e. $|T| = 8$. But there are only 13 points in this plane, a contradiction. □

Corollary 7.2.18. *The linear code $C_1(2, 3)^\perp$ does not have code words of weight $q^2 + q + 1 = 13$. Hence, the maximum weight of $C_1(2, 3)^\perp$ is $q^2 + q$. In other words, the second bound from Theorem 7.2.10 is sharp for $q = 3, n = 2, k = 1$.*

Now we prove a reduction lemma. Again, it reveals a link with blocking sets and makes use of the Bose-Burton Theorem [19]. It will greatly extend the gap between the actual maximum weight and $q^n + q^{n-1} + \cdots + q + 1$ in some cases.

Lemma 7.2.19. *If there exists an integer m with $k \leq m \leq n$, for which $C_k(m, q)^\perp$ does not attain full weight, then $C_k(n, q)^\perp$ has maximum weight at most $q^n + \cdots + q^{n-m+1}$.*

Proof. Let c be a code word of maximum weight in $C_k(n, q)^\perp$. Let S be the set of points on which c is zero, i.e. S is the complement of supp(c). If there exists an m-space Π disjoint from S, then all points in Π correspond to nonzero positions in the code word. Since $\sum_{r \in \pi} c_r = 0$ for every k-space π of PG(n, q), this also holds for all k-spaces $\pi \subseteq \Pi$. Since the positions corresponding to points outside of Π are not relevant for these equations, they still hold when replacing them by 0, hence the restriction of c to the positions in Π is a code word of $C_k(m, q)^\perp$. But $C_k(m, q)^\perp$ does not attain full weight; this contradicts our assumption.

Hence, each m-space contains at least one point of S, which means that S is a blocking set with respect to the m-spaces of $\mathrm{PG}(n,q)$, and so, by the Bose-Burton Theorem [19], $|S|$ has at least the size of an $(n-m)$-space, i.e. $|S| \geq q^{n-m} + \cdots + q + 1$. Hence, the maximum weight of $C_k(n,q)^\perp$ is at most $q^n + \cdots + q^{n-m+1}$. □

Combining Corollary 7.2.18 and Lemma 7.2.19, with $q = 3$, $m = 2$ and $k = 1$, we get the following result.

Theorem 7.2.20. *The maximum weight in* $C_1(n,3)^\perp$ *is* $3^n + 3^{n-1}$.

This is far below the expected value $3^n + 3^{n-1} + \cdots + 3 + 1$. The maximum weight of $C_k(n,3)^\perp$ is still an open problem for $1 < k < \frac{n+1}{2}$.

Remark 7.2.21. The preceding results show that the study of 2 (mod 3) sets in $\mathrm{PG}(n,q)$, $q = 3^h$, plays a crucial role for the investigation of the large weight code words of the code $C_k(n,q)^\perp$. We therefore propose to investigate the existence problem of these 2 (mod 3) sets in the cases not discussed in this section.

Another interesting problem is to determine the exact maximum weight of the codes $C_k(n,q)^\perp$, q odd, not yet discussed in Theorem 7.2.10 and in the remaining theorems of this section. A way to prove that the maximum weight of $C_k(n,q)^\perp$, q odd, is equal to $q^n + \cdots + q + 1$ is to prove the existence of $t \pmod p$ sets with respect to the k-spaces of $\mathrm{PG}(n,q)$, with $t \not\equiv 0, 1 \pmod p$, as indicated in Theorem 7.2.11. It is unknown whether one can ever obtain a larger weight than with the construction in Lemma 7.2.9.

7.3 Blocking sets of the Hermitian unital

It is known that the classical unital arising from the Hermitian curve in $\mathrm{PG}(2,9)$ does not have a 2-coloring without monochromatic lines. In this section, we show that for $q \geq 4$, the Hermitian curve in $\mathrm{PG}(2,q^2)$ does possess 2-colorings without monochromatic lines. We present general constructions and also prove a lower bound on the size of blocking sets in the classical unital. This section is joint work with A. Blokhuis, A.E. Brouwer, D. Jungnickel, V. Krčadinac, S. Rottey, L. Storme and T. Szőnyi and is submitted to Finite Fields Appl. [16].

7.3.1 Introduction

In any point-line geometry (or, much more generally, any hypergraph) a *blocking set* is a subset B of the point set that has nonempty intersection with each line (or each edge).

Blocking sets in the finite projective planes $\mathrm{PG}(2,q)$ have been investigated in great detail [129, 132]. Since in a projective plane any two lines meet, every set containing a line is a blocking set. A blocking set of a projective plane is called *non-trivial* or *proper* when it does not contain a line. We shall also call blocking sets in other point-line geometries *proper* when they do not contain a line. By definition the complement of a proper blocking set is again a

proper blocking set, and every 2-coloring (vertex coloring with two colors such that no line is monochromatic) provides a complementary pair of proper blocking sets.

A blocking set is *minimal* when no proper subset is a blocking set. A blocking set in $\mathrm{PG}(2,q)$ is *small* when its size is smaller than $3(q+1)/2$.

This latter definition was motivated by the important results of Sziklai and Szőnyi, who proved a 1 (mod p) result for small minimal blocking sets B in $\mathrm{PG}(2,q)$.

Theorem 7.3.1 (Sziklai and Szőnyi [129, 132])**.** *Let B be a small minimal blocking set in $PG(2,q)$, $q=p^h$, p prime, $h\geq 1$. Then B intersects every line in* 1 (mod p) *points.*

If e is the largest integer such that B intersects every line in 1 (mod p^e) *points, then e is a divisor of h, and every line of $PG(2,q)$ that intersects B in exactly $1+p^e$ points intersects B in a subline $PG(1,p^e)$.*

In this section, we investigate blocking sets in the classical unital $\mathcal{U}$ arising from the Hermitian curve $\mathcal{H}(2,q^2)$ of $\mathrm{PG}(2,q^2)$. The lines of the unital are the intersections with $\mathcal{U}$ of projective lines that meet $\mathcal{U}$ in at least 2 (and then precisely $q+1$) points.

This research is in part motivated by [2], where an exhaustive search for the unitals of order 3 containing proper blocking sets was performed. That search showed that there are 68806 distinct 2-$(28,4,1)$ unital designs containing a proper blocking set. The classical unital, arising from the Hermitian curve in $\mathrm{PG}(2,9)$, does not contain a proper blocking set. This poses the question of blocking sets in the Hermitian curves $\mathcal{H}(2,q^2)$ of $\mathrm{PG}(2,q^2)$ for general q.

A second motivation is given by the Shift-Blocking Set Problem discussed in Subsection 7.3.1 below.

We show that for $q\geq 4$, the Hermitian curves $\mathcal{H}(2,q^2)$ contain proper blocking sets. We present general constructions of (proper) blocking sets and also prove a lower bound on the size. The lower bound is obtained via the polynomial method, and makes use of a 1 (mod p) result which arises from the applied techniques.

Green-black colorings

Let a proper green-black coloring of the plane $\mathrm{PG}(2,n)$ be a coloring of the points with the colors green and black such that every point P is on a line L that is completely green, with the possible exception of the point P itself. At least how many green points must there be, or, equivalently, at most how many black points? This question is related to the Flat-Containing and Shift-Blocking Set Problem [18].

By definition, every black point is on a tangent, that is, a line containing no further black point. This immediately gives the upper bound $n^{3/2}+1$ for the number of black points [67].

In order to find examples close to this bound, let $n=q^2$, and let $\mathcal{U}$ be the set of points (of size q^3+1) of a classical unital in $\mathrm{PG}(2,n)$, and let B be a blocking set in $\mathcal{U}$. Then we can take $\mathcal{U}\setminus B$ as the set of black points, while the points of B, and all the points outside of $\mathcal{U}$,

q	$\min_g(q)$	$\min_b(q)$	$\min_{pb}(q)$
2	3	5	-
3	10	13	-
4	15	25	26

Table 7.1: the smallest sizes, for small q

are green. Indeed, for a point P of the unital, we can take for L the tangent to $\mathcal{U}$ at P. For a point P outside of $\mathcal{U}$, the line $M = P^\perp$ meets $\mathcal{U}$ in a line of $\mathcal{U}$ that is blocked by B in a (green) point Q, and we can take for L the (entirely green) tangent line at Q.

This motivates the search for small blocking sets in $\mathcal{U}$. In fact what is needed here is something slightly more general. Let us call a subset S of $\mathcal{U}$ *green* when $\mathcal{U} \setminus S$ can be taken as the set of black points in a proper green-black coloring. Then blocking sets of the unital are green. As we shall see, there are also other green sets.

Small q

Let $\min_g(q)$, $\min_b(q)$ and $\min_{pb}(q)$ be the sizes of the smallest green set, blocking set and proper blocking set, respectively, in the classical unital $\mathcal{U}$ of $\mathrm{PG}(2, q^2)$. Clearly, $\min_g(q) \leq \min_b(q) \leq \min_{pb}(q)$. For small q, these values can be found in Table 7.1.

That is, the classical unital does not have a proper blocking set for $q = 2, 3$, and for $q = 4$, there are proper blocking sets, but the smallest blocking sets contain a line. A green set that does not contain a (unital) line is a blocking set. The smallest green sets contain lines.

We describe the green examples. Note that a subset S of $\mathcal{U}$ is green precisely when for each non-tangent line L disjoint from S, the nonisotropic point $L^\perp$ lies on a non-tangent line M, where $M \cap \mathcal{U} \subseteq S$.

For $q = 2$, the unital is an affine plane $\mathrm{AG}(2, 3)$. Pick for S an affine line. The two parallel lines have perps that lie on this line.

For $q = 3$, let P be a point of the unital, and let K, L, M be three unital lines on P without transversal. Then $S = K \cup L \cup M$ has size 10 and is green.

For $q = 4$, let P, Q, R be an orthogonal basis: three mutually orthogonal nonisotropic points. The three lines PQ, PR and QR meet $\mathcal{U}$ in $5 + 5 + 5 = 15$ points, and one checks that this 15-set is green.

Let $\min_{ip}(q)$ be the size of the smallest blocking set of the Miquelian inversive plane of order q (the $S(3, q + 1, q^2 + 1)$ formed by the points and circles on an elliptic quadric in $\mathrm{PG}(3, q)$). Below, in Subsection 7.3.3, we shall see that $\min_b(q) \leq q(\min_{ip}(q) - 1) + 1$. For small q, the values of $\min_{ip}(q)$ can be found in Table 7.2.

q	2	3	4	5	7	8
$\min_{ip}(q)$	3	5	8	10	17	20

Table 7.2: $\min_{ip}(q)$, for small q

7.3.2 A lower bound on the size of a blocking set of the Hermitian curve

Consider $\mathrm{PG}(2, q^2)$. We denote the points by $(x : y : z)$ and the lines by $[t : u : v]$, where the point $(x : y : z)$ and the line $[t : u : v]$ are incident when $tx + uy + vz = 0$.

The map $(x : y : z) \mapsto [z^q : y^q : x^q]$ defines a unitary polarity. Points of the associated unital $\mathcal{U}$ are the points $(x : y : z)$ satisfying $(x : y : z)I[z^q : y^q : x^q]$, so $xz^q + y^{q+1} + zx^q = 0$. The tangents of $\mathcal{U}$ are the lines $[t : u : v]$ satisfying the same equation, so $tv^q + u^{q+1} + vt^q = 0$.

The 'infinite horizontal' point $\infty := (1 : 0 : 0)$ belongs to $\mathcal{U}$. Its pole $\infty^\perp$, the tangent to $\mathcal{U}$ in ∞, is the line $[0 : 0 : 1]$, i.e., the line 'at infinity' $Z = 0$.

We wish to block the lines of the unital, i.e., the subsets of size $q + 1$ of $\mathcal{U}$ that are of the form $\ell \cap \mathcal{U}$ for some line ℓ of $\mathrm{PG}(2, q^2)$. The main result of this section is a lower bound for the size of a blocking set.

Theorem 7.3.2. *Let S be a blocking set of a Hermitian unital $\mathcal{U}$ in $PG(2, q^2)$, then $|S| \geq (3q^2 - 2q - 1)/2$.*

If a subset of $\mathcal{U}$ blocks all the projective lines, then also the tangents to $\mathcal{U}$, and hence the subset must be all of $\mathcal{U}$ (and have size $q^3 + 1$). Also, $\mathcal{U} \cap \infty^\perp = \{\infty\}$. Therefore our result follows immediately from the following theorem.

Theorem 7.3.3. *Let S be a minimal set of points of $PG(2, q^2)$ that blocks all projective lines that are not tangent to $\mathcal{U}$, but not all projective lines. If $S \cap \infty^\perp = \{\infty\}$, then $|S| \geq (3q^2 - 2q - 1)/2$.*

For example, let L be a secant line to $\mathcal{U}$ containing ∞. Let P be a nonisotropic point of L. One may take for S the set of all points of L except P, together with some point on each of the $q^2 - q - 1$ other secant lines on P. Now $|S| = 2q^2 - q - 1$.

Proof: Since a unital point outside of S lies on q^2 unital lines, $|S| \geq q^2$, and it is easy to see that equality cannot hold. Put $B := \{(a, b) \mid (a : b : 1) \in S\}$, so that $|S| = |B| + 1$, and let $|B| = q^2 - q + k$.

Part 1: Polynomial reformulation.

The set S is a blocking set of $\mathcal{U}$ if and only if the polynomial $H(U, V)$ defined by

$$H(U, V) = C(U, V)R(U, V) = (V^q + V + U^{q+1}) \prod_{(a,b)\in B} (V + a + bU)$$

(with $C(U,V) = V^q + V + U^{q+1}$) vanishes identically in $\mathbb{F}_{q^2} \times \mathbb{F}_{q^2}$.

Indeed, a line is non-horizontal (does not pass through ∞) precisely when it is of the form $[1 : u : v]$. Such a line is a tangent to $\mathcal{U}$ when $C(u,v) = 0$ and passes through the point (a,b) when $a + bu + v = 0$. So if S is a blocking set, then $H(u,v) = 0$ for all $u, v \in \mathbb{F}_{q^2}$. Conversely, if $H(u,v) = 0$ for all $u, v \in \mathbb{F}_{q^2}$ and $[1 : u : v]$ is not a tangent, so that $C(u,v) \neq 0$, then $v + a + bu = 0$ for some $(a,b) \in B$, so that this line is blocked by B. We shall use later that the number of points of S on the non-horizontal line $[1 : u : v]$ (plus 1 if it is a tangent) equals the multiplicity of v as a zero of $H(u,V)$.

Since $H(U,V)$ vanishes identically, it belongs to the ideal generated by $U^{q^2} - U$ and $V^{q^2} - V$, so

$$H(U,V) = C(U,V)R(U,V) = (V^{q^2} - V)f(U,V) + (U^{q^2} - U)g(U,V).$$

We may suppose that $|S| < 2q^2 - q$ (the lower bound we are proving is smaller), so that H has degree smaller than $2q^2$. All terms involving U^{q^2} in f can be moved over to g. Then no cancellation occurs, and f and g have total degree at most $k+1$. Since H has a term $U^{q+1}V^{q^2-q+k}$ that must be from $(V^{q^2} - V)f$, it follows that f has degree precisely $k+1$. Since $\deg_V H = q^2 + k$, it follows that $\deg_V f = k$.

If f and g have a common factor $r(U,V)$, then the polynomial H/r vanishes identically. If r is linear, this means that we can delete a point from S and find a smaller blocking set. If r is not linear, then it must equal C (up to a constant factor) since C is irreducible. This would mean that S is a blocking set of the entire plane $\mathrm{PG}(2,q^2)$, contrary to our hypothesis. So f and g are coprime.

Part 2: *Let $u, v \in \mathbb{F}_{q^2}$. If $f(u,v) = 0$, then also $g(u,v) = 0$.*

For fixed $u \in \mathbb{F}_{q^2}$,

$$H(u,V) = C(u,V)R(u,V) = (V^{q^2} - V)f(u,V),$$

since $u^{q^2} - u = 0$. It follows that v is (at least) a double root of $H(u,V)$. Since $C(u,V) = V^q + V + u^{q+1}$ has derivative 1, v is at most a single zero of $C(u,V)$. For each factor $r(U,V)$ of $H(U,V)$, if v is a zero of $r(u,V)$, then u is a zero of $r(U,v)$. It follows that u is (at least) a double root of $H(U,v) = C(U,v)R(U,v) = (U^{q^2} - U)g(U,v)$, and hence $g(u,v) = 0$.

Part 3:

Observe that the nonzero polynomial $f(u,V)$ is fully reducible (factors into linear factors) over $\mathbb{F}_{q^2}$, for any $u \in \mathbb{F}_{q^2}$. Indeed, $(V^{q^2} - V)f(u,V) = C(u,V)R(u,V)$ and both $C(u,V)$ and $R(u,V)$ are fully reducible.

We apply the following lemma.

Lemma 7.3.4. ([17, p. 145]) *Let $h = h(X,Y)$ be a polynomial of total degree d over $\mathbb{F}_q$ without nontrivial common factor with $\partial_Y h$. Let M be the number of zeros of h in $\mathbb{F}_q^2$, where each zero (x,y) is counted with the multiplicity that y has as zero of $h(x,Y)$. Then the total number of zeros of h (each counted once) is at least $M - d(d-1)$.*

Let $f = f_0 \cdots f_m$ be the factorization of f into irreducible components. Let $d_i = \deg(f_i)$ and $d_i' = \deg_V(f_i)$. Then $d_i' \leq d_i$, $d_0' + \cdots + d_m' = k$ and $d_0 + \cdots + d_m = k+1$. Hence, $d_i' = d_i - 1$ for a single component f_i, and $d_j' = d_j$ for $j \neq i$.

Suppose that f has an irreducible factor f_0 with $\partial_V f_0 \not\equiv 0$. Put $m := \deg f_0$ so that $1 \leq m \leq \deg f = k+1$, then $\deg_V(f_0) = m - \epsilon$, with $\epsilon \in \{0, 1\}$, and $\epsilon = 0$ if $m = 1$.

Let N be the number of zeros of f_0 in $\mathbb{F}_{q^2}^2$. On the one hand, since f and g have no common factor, and all zeros of f are also zeros of g, Bézout's theorem gives $N \leq \deg f_0 \deg g \leq m(k+1)$. On the other hand, for any fixed $u \in \mathbb{F}_{q^2}$ the polynomial $f_0(u, V)$ of degree $\deg_V f_0 = m - \epsilon$ has $m - \epsilon$ zeros, counted with multiplicity, altogether $q^2(m - \epsilon)$.

Lemma 7.3.4 now yields the lower bound $N \geq q^2(m-\epsilon) - m(m-1)$, and combining upper and lower bound yields

$$q^2(m-\epsilon) - m(m-1) \leq m(k+1).$$

If $\epsilon = 0$, this gives $k \geq \frac{1}{2}(q^2 - 1)$. If $\epsilon = 1$ and $m > 2$, this gives $k \geq \frac{1}{2}(q^2 - 3)$. If $\epsilon = 1$ and $m = 2$, then no point was counted with multiplicity > 1, and $q^2(m-\epsilon) \leq m(k+1)$ gives $k \geq \frac{1}{2}(q^2 - 2)$. Hence $|S| = q^2 - q + 1 + k \geq \frac{1}{2}(3q^2 - 2q - 1)$ in these cases, as desired.

If $\partial_V f_i = 0$ for all i, then $\partial_V f = 0$, so that $f(u, V)$ is a p-th power, and the multiplicity of v as a root of $H(u, V) = (V^{q^2} - V)f(u, V)$ is 1 (mod p). By an earlier remark, this means that all non-horizontal lines intersect the set S in 1 (mod p) points if they are non-tangent, and in 0 (mod p) points if they are tangent.

For each affine point P, let the horizontal line on P contain $e_P + 1$ points of S (including ∞). Summing the contributions of all lines on P to $|S|$, we find from the tangents 0, and from the ($q^2 - q - 1$ or $q^2 - 1$) non-horizontal secants -1, and from the horizontal secant $e_P + 1$ (all mod p), so that $|S| \equiv e_P$ (mod p) for all P. Summing the contributions of the horizontal lines we see $|S| \equiv 1$ (mod p). It follows that $e_P \equiv 1$ (mod p) and the point ∞ was not required to block the horizontal lines. □

7.3.3 Small blocking sets

In this section, we construct small blocking sets of Hermitian curves, not necessarily proper. In the next section, proper examples will be constructed.

Fractional covers

For blocking sets in general we can apply a bound of Lovász relating the minimum size of a blocking set (cover) τ with that of a fractional cover τ^* of a hypergraph with maximum degree D:

$$\tau \leq (1 + \log D)\tau^*$$

(see [46, Corollary 6.29]). For the unital $\mathcal{U}$, taking every point with weight $1/(q+1)$ gives $\tau^* = q^2 - q + 1$, $D = q^2$, so $\tau \leq (q^2 - q + 1)(1 + 2\log q)$.

Geometric construction

Let $\mathcal{U}$ be the classical unital in $\mathrm{PG}(2,q^2)$, and consider a blocking set B of $\mathcal{U}$ that is the union of a number of lines on a fixed point p of $\mathcal{U}$. The line pencil $\mathcal{L}_p$ of the lines on p in $\mathrm{PG}(2,q^2)$ has the structure of a projective line with distinguished element L_∞, the tangent to $\mathcal{U}$ at p. For each unital line M not on p, the set $M_p = \{L \in \mathcal{L}_p \mid L \cap M \neq \emptyset\}$ is a Baer subline of $\mathcal{L}_p$, and each Baer subline of $\mathcal{L}_p$ not containing L_∞ arises in this way for q pairwise disjoint lines M. We find $|B| = 1 + qm$, where m is the size of a blocking set of the Baer sublines not on L_∞ of the line $\mathcal{L}_p$.

The set $\mathcal{L}_p \setminus \{L_\infty\}$ carries the structure of an affine plane $\mathrm{AG}(2,q)$ of which the lines are the Baer sublines of $\mathcal{L}_p$ on L_∞. The remaining Baer sublines form a system of circles. Any three noncollinear points determine a unique circle. Here we have $q^2(q-1)$ circles, each of size $q+1$, in a set of size q^2, and $D = q^2 - 1$, so Lovász' bound gives $m < q(1 + 2\log q)$. We did not lose anything (in the estimate) by taking B of special shape.

Consider a blocking set C of this collection of circles that is the union of a number of parallel lines. Then $|C| = qn$, where n is the size of a blocking set for the collection of projections of the circles on a fixed line. We have $q(q-1)$ projections, each of size more than $q/2$, in a set of size q.

In order to block N subsets of a q-set, each of size larger than $q/2$, one needs not more than $1 + \log_2 N$ points: if one picks the points of the blocking set greedily, each new point blocks at least half of the sets that were not blocked yet. So, we find a blocking set of size less than $1 + 2\log_2 q \sim 2.89 \log q$ and lost a factor 1.44 in the estimate.

7.3.4 Proper blocking sets of Hermitian curves

We now construct proper blocking sets of Hermitian curves.

Probabilistic constructions

Radhakrishnan and Srinivasan [115, Theorem 2.1] show using probabilistic methods that any n-uniform hypergraph with at most $0.1\sqrt{n/\log n}\,2^n$ edges is 2-colorable, so contains a proper blocking set. (Their constant 0.1 can be improved to 0.7 for sufficiently large n.) In our case $n = q+1$ and the number of edges is $q^4 - q^3 + q^2$, so a unital has a proper blocking set when $q > 17$.

An older bound by Erdős [35] gives the same conclusion when the number of edges is not more than 2^{n-1}, and this applies when $q \geq 16$.

A result by Erdős and Lovász [36, Theorem 2] says that any n-uniform hypergraph in which each point belongs to at most $2^{n-1}/4n$ edges, is 2-colorable. In our case $n = q+1$ and each point belongs to q^2 edges, so this suffices for $q > 13$.

If we choose points for our blocking set at random with probability $p = 5(\log q)/q$, then the expected number of monochromatic edges is roughly $1/q < 1/2$, and now we can assume (just using Chebyshev's inequality) that in addition the size will be close to the expectation, so $5q^2 \log q$.

We now present two different geometric constructions.

A geometric construction

In this section we construct a proper blocking set in the classical unital $\mathcal{H}(2, q^2)$ in $\mathrm{PG}(2, q^2)$ for $q \geq 7$ and for $q = 4$.

We use the model of the unital from [15], [39], and [121]. A detailed description of this approach is also given in the survey paper [49].

Identify the points of the plane $\mathrm{PG}(2, q^2)$ with the elements of the cyclic group G of order $q^4 + q^2 + 1$, where the lines are given by $D + a$, with D a planar difference set, chosen in such a way that D is fixed by every multiplier.

Then $G = A \times B$, where A is the unique subgroup of G of order $q^2 - q + 1$ and where B is the unique subgroup of order $q^2 + q + 1$. We may now write elements of G as pairs $g \equiv (i, j)$, $0 \leq g \leq q^4 + q^2$, $0 \leq i \leq q^2 - q$, $0 \leq j \leq q^2 + q$, $i \equiv g \pmod{q^2 - q + 1}$, and $j \equiv g \pmod{q^2 + q + 1}$. The subgroup A and its cosets are arcs, while the subgroup B and its cosets are Baer subplanes. The map $g \mapsto \mu g$, where $\mu = q^3$, maps the point (i, j) onto the point $(-i, j)$. The map $g \mapsto D - \mu g$ defines a Hermitian polarity, with absolute points given by the Hermitian curve $\mathcal{U} = \{a + \beta \mid a \in A, 2\beta \in B \cap D\}$. So $\mathcal{U}$ is the union of $q + 1$ cosets of the subgroup A.

We will show that if q is odd and $q \geq 7$, then it is possible to partition this collection of $q + 1$ cosets of A into two sets of size $(q + 1)/2$ such that the union of each is a (proper) blocking set of the Hermitian unital $\mathcal{U}$.

Let $\ell \subset G$ be a line of the plane $\mathrm{PG}(2, q^2)$. Then ℓ intersects each coset of A in 0, 1, or 2 points, since cosets of A are $(q^2 - q + 1)$-arcs. The $q^2 - q + 1$ translates of ℓ by an element of A all determine the same intersection pattern. The cosets of B form a partition of the plane $\mathrm{PG}(2, q^2)$ into Baer subplanes $\mathrm{PG}(2, q)$, and ℓ intersects exactly one of these Baer subplanes in a Baer subline. By taking a suitable translate of ℓ, we may assume that this Baer subplane is B itself.

Since multiplication by μ sends the point (i, j) to the point $(-i, j)$, this map fixes cosets of A (setwise), and fixes B pointwise. It follows that also the line ℓ is fixed (setwise) by multiplication by μ. Consequently, ℓ intersects the cosets of A containing a point of the subline $B \cap \ell$ in exactly one point, and the other cosets in 0 or 2 points.

The unital $\mathcal{U}$ is of the form $\mathcal{U} = A + \frac{1}{2}(B \cap D)$, and if q is odd, then $\frac{1}{2}(B \cap D)$ is an oval in the Baer subplane B [15, p. 65]. This means that the intersection pattern of ℓ with the $q + 1$ cosets of A that partition the unital $\mathcal{U}$ (let us call them $\mathcal{U}$-cosets of A) can be of three types.

If $\ell \cap B$ is a tangent of the oval $\frac{1}{2}(B \cap D)$, then ℓ is a tangent of the unital $\mathcal{U}$ as well, and so of no interest from the blocking set point of view. If $\ell \cap B$ is a secant line of the oval $\frac{1}{2}(B \cap D)$, then this means that ℓ intersects two $\mathcal{U}$-cosets of A in a single point, and the remaining ones in 0 or 2 points, where both possibilities happen precisely $(q-1)/2$ times. Finally if $\ell \cap B$ is an external line of the oval $\frac{1}{2}(B \cap D)$, then ℓ intersects all $\mathcal{U}$-cosets of A in 0 or 2 points, and both possibilities happen precisely $(q+1)/2$ times. There are $(q^2-q)/2$ external lines, and hence $(q^2-q)/2$ partitions of the set of $\mathcal{U}$-cosets of A into two sets of size $(q+1)/2$ that do not lead to proper blocking sets of $\mathcal{U}$. If $\frac{1}{2}\binom{q+1}{(q+1)/2} > \frac{1}{2}(q^2-q)$, then there is a partition of $\mathcal{U}$ into two unions of $(q+1)/2$ cosets of the subgroup A, that are both blocking sets. This happens for $q \geq 7$.

If $q = 5$, then the 10 external lines determine 10 distinct triples of $\mathcal{U}$-cosets of A, no two disjoint, so we find blocking sets (of size 63) but no proper blocking sets in this way.

If q is even, the situation is slightly different: in this case 2 is a multiplier that fixes both B and D, and $\frac{1}{2}(B \cap D) = B \cap D$ is a line in B. Now for a line ℓ in the plane $\mathrm{PG}(2, q^2)$, such that $\ell \cap B$ is a line in the Baer subplane B, we have three possibilities: either $\ell = D$, with intersection pattern 1^{q+1}, or ℓ is a tangent of $\mathcal{U}$, or ℓ has intersection pattern 1^1, $0^{q/2}$, $2^{q/2}$. We now want to partition the unital $\mathcal{U}$ into collections of $q/2$ and $q/2+1$ cosets of A to construct proper blocking sets of $\mathcal{U}$, and the only thing to avoid is to take a $q/2$-set corresponding to the 0's in the intersection pattern of a line ℓ, so there are at most q^2-1 such $q/2$-sets, but $q^2 - 1 < \binom{q+1}{q/2}$ for $q \geq 8$.

If $q = 4$, then multiplication by 2 has two orbits on the $\mathcal{U}$-cosets of A, of sizes 2 and 3, and their unions form a complementary pair of proper blocking sets (of sizes 26 and 39).

So far we constructed proper blocking sets for $q > 3$, $q \neq 5$. For $q = 5$ the above method fails, but a random greedy computer search shows that $\mathcal{H}(2, 25)$ does contain disjoint blocking sets of sizes 45 and 51, so that there exist proper blocking sets of all sizes from 45 to 81.

We summarize the above discussion in the main theorem of this section.

Theorem 7.3.5. *The Hermitian curve $\mathcal{H}(2, q^2)$ contains a proper blocking set if and only if $q > 3$.*

Remark 7.3.6. The above arguments can also be used to show the existence of smaller proper blocking sets. We try to find a blocking set consisting of r cosets of A, with $2r \leq q$ as small as possible (the complement will then automatically also be a blocking set). We have q^2 intersection patterns, each with at most $(q+1)/2$ zero's, implying that at most $q^2\binom{(q+1)/2}{r}$ r-tuples are bad, so if $\binom{q+1}{r} > q^2\binom{(q+1)/2}{r}$ then we are fine, and this is certainly the case if $2^r \geq q^2$. This yields proper blocking sets of size $\frac{2\log q}{\log 2}(q^2-q+1)$, a little larger than the blocking sets we got from Lovász' bound.

Explicit examples

We now present a construction that yields explicit examples of proper blocking sets on the Hermitian curve.

Theorem 7.3.7. *Let $r|(q-1)$, where $r > 1$ and $4r^2+1 < q$. Then, for some value k satisfying $1 \leq k \leq q^2 - q + 1$, the Hermitian curve $\mathcal{U}$ in $PG(2, q^2)$ contains a proper blocking set B of size $k + q(q-1)^2/r$.*

Remark 7.3.8. For $r \sim \sqrt{q}/2$, this construction leads to proper blocking sets on the Hermitian curve $\mathcal{U}$ of $\mathrm{PG}(2, q^2)$ of size approximately $2q^2\sqrt{q}$. One may compare this explicit construction to the result obtained using the probabilistic method. As we saw, the probabilistic method leads to blocking sets of cardinality $Cq^2 \log q$, for some small constant $C(\leq 5)$.

The setting. The Hermitian curve is $\mathcal{U} : X^q + X + Y^{q+1} = 0$ in the affine plane $\mathrm{AG}(2, q^2)$. This Hermitian curve intersects the line at infinity $Z = 0$ in the unique point $(x : y : z) = (1 : 0 : 0)$.

We first consider the case that q is odd. The case q even is similar, but slightly more complicated. Fix r, where $r|(q-1)$. Let k be a fixed non-square in $\mathbb{F}_q$. Let $i^2 = k$, with $i \in \mathbb{F}_{q^2} \setminus \mathbb{F}_q$. Then $i^q = -i$, and $i^{q+1} = -k$. We describe the elements x of $\mathbb{F}_{q^2}$ by $x = x_1 + ix_2$, with $x_1, x_2 \in \mathbb{F}_q$.

Step 1. First of all we construct a blocking set B of $\mathcal{U}$, defined by

$$B = \{(x, y) \in \mathcal{U} \mid y = u^r + iv, \text{ with } u, v \in \mathbb{F}_q\} \cup \{(1 : 0 : 0)\}.$$

So B contains the point $(1 : 0 : 0)$ and the points of $\mathcal{U}$ on the horizontal lines $Y = u^r + iv$, $u, v \in \mathbb{F}_q$. Afterwards in Step 2, a modification will be made to the blocking set B to make it proper.

In order to prove that B is a blocking set, we have to show that it meets all non-horizontal lines, since the horizontal lines are blocked by $(1 : 0 : 0)$. Consider the intersection of a non-horizontal line $X = nY + c$, where $n = n_1 + in_2$ and $c = c_1 + ic_2$ where $n_1, n_2, c_1, c_2 \in \mathbb{F}_q$, with B. Substituting $X = nY + c = n(u^r + iv) + c$ in the equation $X^q + X + Y^{q+1} = 0$ of $\mathcal{U}$, and using $i^q = -i$ and $i^{q+1} = -k$, leads to the equation

$$2n_1u^r + 2kn_2v + 2c_1 + u^{2r} - kv^2 = 0.$$

We make the equation homogeneous and denote the algebraic curve in $\mathrm{PG}(2, q)$ defined by this equation by $\Gamma : 2n_1U^rW^r + 2kn_2VW^{2r-1} + 2c_1W^{2r} + U^{2r} - kV^2W^{2r-2} = 0$.

Lemma 7.3.9. *The point $(0 : 1 : 0)$ is a point of multiplicity $2r - 2$ of the algebraic curve Γ and the algebraic curve Γ is absolutely irreducible of genus $r - 1$.*

Proof: If we put $V = 1$, the minimal degree becomes $2r - 2$, so $(0 : 1 : 0)$ is a point of multiplicity $2r - 2$. Next, put $W = 1$. The equation of Γ becomes $2n_1U^r + 2kn_2V + 2c_1 + U^{2r} - kV^2 = 0$. This is the hyperelliptic curve $k(V - n_2)^2 = U^{2r} + 2n_1U^r + 2c_1 + kn_2^2$. The only way for this curve to be reducible is that the right hand side is the square $(U^r + n_1)^2$, which implies $n_1^2 = 2c_1 + kn_2^2$, but this means that the line $X = nY + c$ with coordinates $[1 : -n : -c]$ satisfies $-c^q + n^{q+1} - c = 0$, and therefore is a tangent to the unital. So the right

hand side factors as $(U^r - \alpha)(U^r - \beta)$ (in $\mathbb{F}_q^2$) where α and β are different. Since $r|(q-1)$, it has no multiple roots, so we have a hyperelliptic curve of genus $g = r - 1$ (see for instance [128, p. 113]). □

Using the Hasse-Weil bound, we see that Γ contains between $q + 1 - (2r - 2)\sqrt{q}$ and $q + 1 + (2r - 2)\sqrt{q}$ points. For small r, the lower bound on the cardinality of Γ is larger than zero.

We need to convert these bounds on the cardinality of Γ into bounds on the number of points of the set B on the non-horizontal line $X = nY + c$. We first determine the number of points of Γ on the line $U = 0$. Since Γ is absolutely irreducible, we have apart from $(0 : 1 : 0)$ at most two other affine points since $(0 : 1 : 0)$ is a point of multiplicity $2r - 2$ of Γ. We decrease the lower bound on the cardinality of Γ by three, which gives the interval $q-2-(2r-2)\sqrt{q} \leq |\Gamma \setminus (U = 0)| \leq q+1+(2r-2)\sqrt{q}$. Now if $(u, v) \in \Gamma$, with $u \neq 0$, then also every point $(u\xi^i, v)$, ξ a primitive r-th root of unity, $i = 0, \ldots, r - 1$, belongs to Γ. But the points (u, v) and $(u\xi^i, v)$, $i = 0, \ldots, r - 1$, define the same affine points $(x, y) = (x, u^r + iv)$ of the set B. Hence, a non-horizontal line $X = nY + c$ contains z points of B, where $(q - 2 - (2r - 2)\sqrt{q})/r \leq z \leq (q + 1 + (2r - 2)\sqrt{q})/r$.

This then implies for small values of r that every non-horizontal line $X = nY + c$ contains at least one point of B, so that B is indeed a blocking set. Of course B contains some horizontal blocks. To turn B into a proper blocking set we proceed as follows.

Step 2. Consider a cyclic $(q^2 - q + 1)$-arc A, contained in $\mathcal{U}$ and passing through $(1 : 0 : 0)$. Then exactly $q + 1$ lines of $\mathrm{PG}(2, q^2)$ through $(1 : 0 : 0)$ are tangent lines to the arc A. These $q+1$ lines through $(1 : 0 : 0)$ tangent to A form a dual Baer subline at $(1 : 0 : 0)$ [39, Theorem 3.4]. One of these $q+1$ lines through $(1 : 0 : 0)$ tangent to the arc A is the tangent line $Z = 0$ to $\mathcal{U}$ in $(1 : 0 : 0)$, and the remaining q are secant lines to $\mathcal{U}$.

We now delete from the blocking set B all points of the arc $A \cap B$, different from $(1 : 0 : 0)$, and all points of B lying on these q lines through $(1 : 0 : 0)$ secant to $\mathcal{U}$ and tangent to A, but different from $(1 : 0 : 0)$. We show that for small values of r, the set $\tilde{B}$ that remains is a proper blocking set of $\mathcal{U}$. Every horizontal line still is blocked by $(1 : 0 : 0)$, but since we delete a point of B on every horizontal line $Y = u^r + iv$, no horizontal block of $\mathcal{U}$ is contained in $\tilde{B}$. Every non-horizontal line $X = nY + c$ contains at most two points of the arc A. Similarly, every non-horizontal line $X = nY + c$ contains at most two points of $\mathcal{U}$ on lines of the dual Baer subline of tangents through $(1 : 0 : 0)$ to A. For, suppose that such a line contains at least three points of $\mathcal{U}$ on lines of this dual Baer subline. Since a Baer subline is uniquely defined by three of its points, this would imply that the line $X = nY + c$ shares $q + 1$ points with $\mathcal{U}$ on the lines of this dual Baer subline. But this is impossible, since the line $Z = 0$ is one of the lines of this dual Baer subline and this line $Z = 0$ is a tangent line to $\mathcal{U}$ only intersecting $\mathcal{U}$ in $(1 : 0 : 0)$. So we subtract four from the lower bound on the intersection size of the non-horizontal line $X = nY + c$ with B. This leads to the new lower bound $(q - 2 - (2r - 2)\sqrt{q})/r - 4$.

Our assumption $4r^2 + 1 < q$ guarantees that this lower bound is still positive, so that the newly obtained set $\tilde{B}$ still blocks all the non-horizontal secant lines to $\mathcal{U}$.

To be sure that the non-horizontal lines do not contain a block, we look at the upper bound on the intersection sizes of these lines with the set $\tilde{B}$. This is $(q+1+(2r-2)\sqrt{q})/r$, which is less than $q+1$, so also the non-horizontal lines do not contain a block of $\mathcal{U}$.

Cardinality. Now that we are sure that the constructed set $\tilde{B}$ is a proper blocking set, we investigate its cardinality.

In the first step of the construction, B consists of the point $(1:0:0)$ and of the points of $\mathcal{U}$ on the horizontal lines $Y = u^r + iv$, with $u, v \in \mathbb{F}_q$. There are $q + (q-1)\cdot q/r$ such horizontal lines, leading to $|B| = 1 + q\cdot(q + \frac{q^2-q}{r})$.

Now in the second step, the points of B, different from $(1:0:0)$, lying on a cyclic (q^2-q+1)-arc A of $\mathcal{U}$ through $(1:0:0)$ and on the q secants through $(1:0:0)$ to $\mathcal{U}$, tangent to A, are deleted from B.

We first determine the maximal number of points that can be deleted from the blocking set B in this way. The maximum can only occur when all q secants of $\mathcal{U}$ on $(1:0:0)$ tangent to A contain q points of B, different from $(1:0:0)$. This leads to the loss of $q\cdot q = q^2$ points of B. Then still $q + (q-1)q/r - q = (q-1)q/r$ horizontal lines remain which still lose one point on the cyclic (q^2-q+1)-arc A. So the smallest size for the blocking set $\tilde{B}$, is

$$1 + q^2 + \frac{q^3-q^2}{r} - q^2 - \frac{(q-1)q}{r} = 1 + \frac{q^3-2q^2+q}{r}.$$

We now determine the minimal number of points that can be deleted from the blocking set B in this way. The minimum can only occur when all q secants of $\mathcal{U}$ on $(1:0:0)$ tangent to A contain zero points of B, different from $(1:0:0)$. Then the $q + (q-1)q/r$ horizontal lines $Y = u^r + iv$ still lose one point on the cyclic (q^2-q+1)-arc A. So the largest possible size for the blocking set $\tilde{B}$, is

$$1 + q^2 + \frac{q^3-q^2}{r} - q - \frac{(q-1)q}{r} = 1 + q^2 - q + \frac{q^3-2q^2+q}{r}.$$

Even q. The preceding results are also valid for q even, but the description of the algebraic curve Γ is different. Namely, for q even, let $k \in \mathbb{F}_q$ with $\mathrm{Tr}_{\mathbb{F}_q/\mathbb{F}_2}(k) = 1$. Let $i^2 + i + k = 0$, then $i^q + i = 1, i^2 = i + k$, and $i^{q+1} = k$. Let $\mathcal{U} : X^q + X + Y^{q+1} = 0$. Let r again be a divisor of $q-1$ and denote every non-horizontal line by $X = nY + c$, with $n = n_1 + in_2$ and $c = c_1 + ic_2$, $n_1, n_2, c_1, c_2 \in \mathbb{F}_q$. Then the corresponding algebraic curve Γ is

$$\Gamma : (n_1+n_2)VW^{2r-1} + n_2U^rW^r + c_2W^{2r} + U^{2r} + U^rVW^{r-1} + kV^2W^{2r-2} = 0.$$

By putting $V = 1$, it is again observed that the point $(0:1:0)$ is a singular point of Γ with multiplicity $2r-2$. Next we put $W = 1$ and obtain the (hyperelliptic) curve $kV^2 + (U^r + n_1 + n_2)V + U^{2r} + n_2U^r + c_2 = 0$. As before we can show that this curve is irreducible unless the line $X = nY + c$ is a tangent. The genus of this curve is again $g = r-1$ [4, p. 317]. This implies that the arguments for q odd also are valid for q even.

This completes the proof of Theorem 7.3.7.

Appendix A

Building a low-cost GPU based supercomputer

A.1 The hardware

GPUs (Graphics Processing Units) are highly performant computing devices with a large number of processing units, which are specifically optimized for heavy data-parallel computing. In their early years these were mainly used to execute the the graphical computations in three-dimensional video games in real time, but nowadays these cards can be used for general purpose programming in a format that can be embedded in all common programming languages: Java, C/C++, Python...

This allows researchers to implement algorithms on these cards, which for particular algorithms (namely those with a small computational kernel that can be executed in a fully data-parallel way) enables speedups with over a factor 100 in comparison with classical processing units (CPUs). Several of the algorithms used during my PhD research, belong to this category. Moreover, due to their data-parallel architecture and high number of low-complexity cores, these algorithms run not only faster but also consume orders of magnitude less power. Additionally, GPUs are much cheaper than CPUs for the same computing power.

For example, a €265 Haswell i7-3771 CPU has roughly 100 GFLOPS (which is 10^{11} Floating Point Operations Per Second) and consumes up to 84 watt, while a Radeon HD 7970 GPU costed €360 about 1.5 years ago and has roughly 4000 GFLOPS theoretical computing power on single precision and consumes up to 250 watt. This trend holds in general: for the same computing power, GPU devices cost about 20 times less to purchase and consume about 10 times less power (which in turn reduces the cost needed for heat dissipation). Moreover, certain functionalities such as interthread synchronization are inherently faster on GPU, as they are provisioned in the hardware rather than in the software.

Unfortunately, not all algorithms can be executed in a data-parallel way. In the remainder of this section, I will give a description of the hardware used in my GPU machine, how I built the machine, and what difficulties I encountered in doing so, as well as how I resolved

them. In Section A.2, I will go more into detail on how parallel computing works and how to understand what is data-parallelism. In Section A.3, I will explain how the LDPC decoding algorithm works, and in Section A.4, I will explain how I implemented this in OpenCL.

My main goal of my setup was to put 8 distinct GPUs on one machine, to allow a maximum number of computing power in a single machine (current motherboards are limited to 8 GPUs because of memory addressing reasons, making this choice obvious). After all, GPU computing is at the time of writing still a rather small niche.

The first decision to make then, is what GPUs to use. There are two main players in the field of GPU computing chips: AMD and nVidia. In general, AMD builds the best computing hardware, while nVidia offers a more mature software suite to work. Since my research does not use any floating-point computations, the large libraries nVidia's CUDA suite has to offer were of little interest to me, so I decided to go with AMD. The programming language used on this card is OpenCL, an open computing language that is aimed at GPU computing but can be run on virtually any computing devices (including CPUs). At the time I was looking into, AMD was about to release their new Radeon HD 7000 series which offered a new architecture (GCN) that was more specifically aimed at computing, and at the same time significantly reduced power consumption and heat production by scaling down the production process from 40nm to to 28nm. The fastest card in this series is the HD 7970, which is the GPU that I have used in this machine.

Connecting these GPUs to a motherboard is the second big challenge. While there are highly specialized motherboards on the market that have 8 ready PCI-express at sufficient distance, they are really expensive due to the small number of people that want to buy them. Mainstream motherboards however don't have 8 PCI-express slots, except for one that surprisingly did: the MSI Big Bang Marshall B3. Being the only motherboard with 8 PCI-express slots, this was also an obvious choice. A next challenge was that the slots only had a single slot of space between them, while the cards are two slots high and it is advisable (although not required with some card models) to have an extra slot of space for sufficient air flow (else the cards get too hot).

The first option I considered was to replace the air cooling with water cooling. Water cooling blocks are thinner and require no extra space for heat dissipation, so spacewise, this would work. However, considering that these 8 GPUs could produce up to 2000 watt of heat, getting them all in one cooling circuit could be dangerous, as it is not impossible that the water would start boiling before it has passed all the cards. Afterall, these solutions are primarily made for gamers, who only use up to four cards (the current limit supported for gaming), so this hardware is untested for such an amount of heat. Since boiling would make the circuit break and potentially destroy the computer by the resulting water leaks, I did not consider this a feasible option. On top of that, it also turned out a lot more expensive than air cooling.

The deus ex machina that I needed here came from cablesaurus.com, a small webshop in the United States who produces powered PCI-express extender cables. Using these cables, it is possible to position the GPUs up to 15cm away from the PCI-express slot on the motherboard that it's attached to. This allows more flexibility in the positioning of the card, but alas the construction can no longer fit in a regular computer case. This is the approach that I finally took, and I made a custom steel frame to support the cards, a picture of which is shown in

Figure A.1: Picture of the machine

Figure A.1. This way, all GPUs can be mounted in such a way that they have sufficient room for air flow and heat dissipation.

Finally, the last challenge in the hardware configuration was to get sufficient power to the cards. A single power supply (PSU) providing over 2000 watt with sufficient connections was nonexistent at the time I built the machine, so I had to find another way. I ended up using two 1200 watt PSUs, and make each of them provide power to 4 GPUs. However, in order not to damage the power supplies by letting them indirectly feed all 8 cards at some point, I had to link the PSUs so that they would be turned on an off together, even in rare events like after a power outage. Conveniently, cablesaurus.com also sold cables for that, but the instance I received turned out to be broken. Fortunately, I managed to work around it in an even simpler way: I linked the two power-on wires with a simple paperclip, taped firmly in place and surrounded with an isolation coating.

The rest of the hardware part was rather classical, so I will not discuss those parts here. Instead, I will finish this section with a short guide to get the software part working, since even though the end result is simple, it took me a lot of time to find the right steps to get there. The worst thing that can go wrong, is that the operating system (in my case Ubuntu Linux) suggests to automatically download and install the appropriate drivers. If one ever says yes to that, the installation is unrepairably damaged: uninstalling the drivers is not sufficient, a full reinstallation of the operating system was for me the only way to get out after that. Instead, I downloaded an installed the latest version of the drivers from the AMD website, and this worked flawlessly, but even after reboot, OpenCL only recognized one GPU. I had to execute

```
aticonfig -f --initial --adapter=all
```

as root, and reboot again, to get all cards working in OpenCL.

A.2 Efficient parallel computing

A.2.1 Memory access and caching

Modern CPUs can process up to 16 floating point operations per cycle per core, resulting in a processing capability of up to $48 \cdot 10^9$ floating points per second on a 3 GHz CPU. However, in order to do meaningful computations with this, we need to be able to supply the CPU with relevant work to perform. And here lies a problem: these numbers have become so high, that access speeds to the computer's RAM memory cannot follow. Not only does it have insufficient bandwidth to feed this much work to the CPU (it could at most utilize somewhere between 5% and 10% of that), it also has a latency of approximately 100 clock cycles which would reduce the performance below 1% of the maximum if data was directly read from and written to the computer's RAM memory.

For this reason, caching was invented. A cache is a storage place on the CPU chip itself, where recently accessed information from the RAM memory can be read from and written to,

removing the requirement to wait 100 cycles between read/write operations. Modern multi-core CPUs typically have three levels of cache: a very small but very fast L1 cache (around 32KB, but only needing 4-6 cycles latency per read/write and sufficient bandwidth for full peak) on each core, a slower but larger L2 cache on each core, and an even larger and slower L3 cache which is shared among all CPU cores (but which is still faster and much smaller than the RAM, typically several megabytes).

When a memory location needs to be read, the CPU first checks if it is cached in the L1 cache. If it is present there, it is read; otherwise we fall back to the L2 cache. If it is present there we move it to the L1 cache and read it; otherwise we fall back to the L3 cache and repeat the same there. If it is also missing there, we fall back to the RAM and repeat the same there. For writing, similar mechanisms are in action to keep all levels consistent, but discussing these is beyond the scope of this section.

This cache is structured as a number of cache lines, each consisting of a fixed length of e.g. 64 bytes. In the case of a 32KB L1 cache, this means 512 cache lines of 64 bytes each. In other words: when we fetch something from any cache level, we fetch 64 bytes at a time, which corresponds to 8 `double`s, 8 `long`s, 16 `float`s, 16 `int`s, 32 `short`s or 32 `boolean`s.

One could rightfully wonder how having only 512 cache lines can ever cause consistent speedups in large programs, especially given the fact that on this low level we cannot use complex logic to determine which entries to store in the cache and which ones not to store, so we are down to just storing the last accessed X entries (where, due to hardware implementation reasons and a phenomenon called *cache thrashing*, only a very small value of X is mathematically assured, typically $X = 8$). The answer to this is that experimentally, most inner loops of algorithms appear to feature some sort of locality:

- temporal locality: they use certain entries over and over again, and because they are so frequently accessed, they never or almost never need to be fetched from a higher level cache than L1;
- spatial locality: if a certain memory location is accessed, it is likely that the memory locations around it will also be accessed, therefore we benefit from loading the entire cache line at once.

Example A.2.1. Consider the following simple problem: given a 1024-element memory

$$D = \begin{bmatrix} A_0 & A_1 & A_2 & \cdots & A_{1023} \end{bmatrix}$$

filled with integers, we want to compute the sum of these numbers. A simple way to do this is as follows:

for $i = 1, \ldots, 1023$ **do**
  $A_0 = A_0 + A_i$
end for

Here A_0 is accessed all the time, so it will be in the L_1 cache the entire time, completely removing the need for it to be loaded from higher level caches or RAM (except for maybe the first time). This is an example of temporal locality. On the other hand, assuming 4-byte integers, any cache line contains 16 of these integers, meaning that for every time we load

a cache line, we can read 16 numbers from it, meaning that thanks to this caching we only need to access the higher caches or RAM only $\frac{1024}{16} = 64$ times instead of 1024 times. This is an example of spatial locality.

Protocols are in place to keep the non-shared cache coherent when present in several CPU's L1 and L2 cache, such as the MESI protocol [111], but this falls beyond the scope of this section. Also, it is clear that sharing the L3 cache can have both positive and negative impact on performance: one core could fetch a cache line that another thread needs later (hence speeding everything up), or it could read other cache lines which indirectly wipe out a cache line that another thread reads later (called *false sharing of cache*). Further discussion of this behavior is also beyond the scope of this section.

A.2.2 An essential shift: parallel computing

Over the past decades, we've been used to computers becoming faster and faster every year. Since 1975, the number of transistors on integrated circuits has been doubling approximately every two years. This is commonly referred to as "Moore's Law", named after Intel co-founder Gordon E. Moore, who described the trend in his 1965 paper [103] (although it should be noted that his original prediction was a yearly doubling, only updated to the current estimate in 1975 [104]).

The actual computation speed of a single processing unit, has more or less grown linearly with that number; until the dawn of the 21st century. With top-range CPUs clock speeds being roughly at the same in 2013 as in 2003, this part of the fairytale seems to be permanently over. Some improvements have still been made since then, e.g.:

- adding several pipelines which allow several instructions to be carried out simultaneously if the compiler can assure that they don't need the result of any unfinished instructions to start;
- adding vector instructions which can perform addition, multiplication, etc. on several numbers at the same time;
- out-of-order execution, register renaming, branch prediction, speculative execution and other nifty hardware tricks;

but they are not comparable to the exponential growth that Moore's law predicts. This doesn't mean that Moore's law is over, though. Performance improvement keeps coming at the same exponential rate, but all current improvements come from the ability of the chips to process several workloads simultaneously (called *concurrency* or *parallelism*), rather than executing the same instructions faster.

Therefore, to execute mathematical algorithms at high speed, it is no longer sufficient to only use 20th century metrics such as the number of required operations. An equally important metric has become: how many of those operations can be executed in parallel. At the time of writing, desktop home computers have 2-8 independent CPU cores, gaming GPUs have

64-2048 shader cores, and high performance clusters (HPCs) have several tens of thousands low-power CPUs at their disposal.

For previous performance improvements, no thorough redesign of mathematical algorithms was needed: the same algorithms worked faster, and even if part of the algorithm needed to be rewritten to make more efficient usage of new hardware (such as low-latency caches), doing it in a sloppy way could at most negate part of the performance gain. With parallelization, the mathematical correctness of the outcome itself is at stake when not dealing with concurrency issues properly. In Section A.2.3, we discuss concurrency issues and how even simple algorithms like summing up an array need to be completely rewritten to make proper use of multiple CPUs or other parallel hardware. In Section A.2.4 we discuss the difference between task-parallelism and data-parallelism.

A.2.3 Principles of parallel computing

Allowing multiple cores to assist simultaneously in a computational process without endangering the correctness of its outcome and without compromising the speed advantage, has certain restrictions. To understand these restrictions, we need to consider the notion of a thread.

Definition A.2.2. A *thread* is a pair (I, D) of an instruction stream and a data set, attached to a physical computation unit, which executes the instructions from I one by one on the data in D. Every computation unit can only execute one instruction from one thread at the same time.

Clearly, all simple, non-parallel algorithms can be seen as consisting of a single thread: I are just the steps to be executed in the algorithm and D is the RAM-memory of the computer accessed by the algorithm.

Let $T_1, \dots, T_n$ be a collection of threads, where $T_i = (I_i, D_i)$ for all i. As long as $D_i \cap D_j = \emptyset$ for all $i \neq j$, these threads can run in parallel on n different computing cores without any correctness problems, hence utilizing the full parallel potential of the n processing units. However, in most cases, the requirements that all the D_is are disjoint, is not a realistic one.

Example A.2.3. One could think to be smart and let 1023 threads $T_1, T_2, \dots, T_{1023}$ execute $A_0 := A_0 + A_i$ on T_i. This way, we can execute all additions in parallel and hence complete the job with only the time of one addition instead of 1023 additions. However, that's not what the hardware will do. If we execute $T_1, T_2, \dots, T_{2013}$ simultaneously, instead the following will happen:

1. All threads read A_0 and A_i from D and store them into their local workspace.
2. All threads compute $A_0 + A_i$ (so T_1 knows $A_0 + A_1$, T_2 knows $A_0 + A_2$, and so on).
3. Some thread T_i writes $A_0 + A_i$ back to the first entry of D, another thread T_j immediately overwrites it by $A_0 + A_j$, and all other threads also overwrite eachother's results in this way.

The order in which the threads will (over)write is completely unpredictable, so the output of this method is worthless.

To prevent this behavior, and to ensure mathematical correctness of our computations, we impose the following two restrictions on any multi-threaded program:

- no two threads may write to the same memory location simultaneously;
- a given memory location may not be read from and written to simultaneously.

If any of these rules is not respected, the outcome of the algorithm becomes undefined. In such case, we say that the algorithm *is not thread-safe*, or that it *has concurrency issues*. This does not necessarily mean that the output *will* be incorrect. In fact, most concurrency issues only cause occasional sporadic errors which makes them rather difficult to debug, and which makes their impact only larger when occuring. The idiom "Think before you write!" is hence even more important when it comes to parallel programming.

To realize these rules, we need to provide *synchronization* methods to threads: ways in which they can communicate with eachother when a certain operation is done. For this purpose, all modern programming environments allow at least two thread synchronization mechanisms: atomic instructions and barriers.

Definition A.2.4. An *atomic (instruction)* in a thread is an instruction (or a series of instructions) that this thread can perform undisturbedly: any other thread trying to access the data this series of instructions acts upon, will be halted and has to wait for the atomic instruction to complete before resuming their own work.

This causes no major slowdown if no other threads try to access the data being operated on, but can completely undo the benefit of parallelization (and even cause large overhead penalties compared to single-thread execution) if all threads have to wait for eachother all the time because of this.

Definition A.2.5. A *barrier* for a collection $\mathcal{T}$ of threads is an instruction that halts any thread that executes it, until all threads in $\mathcal{T}$ have reached this instruction.

An easy way to implement this is to have a variable starting at 0, do an atomic +1 in each thread, and wait for the variable to be $|\mathcal{T}|$ to continue. In some hardware, and in particular in all modern GPUs, this functionality is implemented in the hardware to improve its efficiency. Some architectures can have barriers at no extra cost, because they have a shared instruction stream and the clock cycles are inherently synchronized, so each thread can determine where in the execution the other threads are. This is why we mention barriers as a separate mechanism.

With these two synchronization mechanisms, we can fix the problems that we had with Example A.2.3. We will list three methods for it.

Method 1 ($\mathcal{O}(n)$ atomic ops, 0 barriers). A first possibility would be to keep the method from Example A.2.3, but let each thread execute $A_0 := A_0 + A_i$ as an atomic. This makes the output mathematically correct, but because all threads are writing to A_0, the atomicness

completely serializes the algorithm, requiring 2013 subsequent steps to compute the sum (i.e. just as much as without any parallelism).

Method 2 ($\mathcal{O}(\sqrt{n})$ ops, 1 barrier). A better option is as follows, utilizing 32 processing units: we create threads $T_0, \ldots, T_{31}$, and let thread T_i execute

$$A_{32i} := A_{32i} + A_{32i+1} + \cdots + A_{32i+31}.$$

Then, after all threads have completed this (i.e. a barrier on 32 threads), we let T_0 compute

$$A_0 := A_0 + A_{32} + A_{64} + \cdots + A_{992}.$$

This way, A_0 contains the desired sum, with only $2(\sqrt{1024} - 1) = 62$ addition steps and 1 barrier.

Method 3 ($\mathcal{O}(\log n)$ ops, $\mathcal{O}(\log n)$ barriers). When many processing units are available and their barrier cost is very cheap, the fastest solution is as follows. Create a set of 1024 threads $\mathcal{T} = \{T_0, T_1, \ldots, T_{1023}\}$ on 1023 different processing units, and proceed as follows. In thread T_i, we execute:

for $k = 0, \ldots, 9$ **do**
  if $i \equiv 0 \pmod{2^{k+1}}$ **then**
    $A_i := A_i + A_{i+2^k}$
  end if
  barrier (=all T_i must finish this k-iteration before entering the next)
end for

This way, A_0 contains the desired sum, with only $\log_2(1024) = 10$ addition steps and 9 barriers (since the last barrier technically doesn't need to be executed).

Method 4 (generalization). In the second method, we split our array as $1024 = 32^2$ and then performed $2 \cdot (32 - 1)$ additions and $2 - 1 = 1$ barrier. In the third method, we split our array as $1024 = 2^{10}$ and then performed $10 \cdot (2 - 1) = 10$ additions and $10 - 1 = 9$ barriers. More generally, to process n elements, we can take any integer $N \geq n$ and write it as $N = n_1 \cdot n_2 \cdots n_m$. Then we run the following algorithm in each thread T_i:

for $k = 1, \ldots, m$ **do**
  if $i \equiv 0 \pmod{n_1 n_2 \cdots n_k}$ **then**
    for $j = 1, \ldots, n_k - 1$ **do**
      $A_i := A_i + A_{i+j \cdot n_1 n_2 \cdots n_{k-1}}$
    end for
  end if
  barrier
end for

This finishes the job in $\sum_{k=1}^{m}(n_k - 1)$ additions and $m - 1$ barriers (since also here the last barrier is unnecessary).

Remark A.2.6. While we used addition of elements in an array, it is clear that any commutative and associative operator can be used instead.

Remark A.2.7. Which method is fastest depends on how long it takes for addition (or whatever operation one is applying) versus the cost of a barrier. It also depends on the number of processing elements available (e.g. if there are only 32 processing units available, there is no point in using Method 3 as it will never be faster than Method 2).

A.2.4 Task-parallel vs Data-parallel computing

Let $\mathcal{T}$ be a collection of threads. Flynn's Taxonomy [40] distinguishes three possible ways the instruction streams and data sets of these threads can be connected (the paper contains four, but the MISD architecture is rare and not relevant for us).

- SISD (single-input-single-data). Each processing unit involved necessarily operates the same instruction, on the same data element(s). Obviously, this voids any use of multiple processing elements, so here there is no parallelism.
- SIMD (single-input-multiple-data). Each processing unit involved necessarily operates the same instruction, but on different data element(s). This is the structure used in compute units of GPUs, and superscalar CPUs, and it is perfect to apply the same operation to large arrays of elements.
- MIMD (multiple-input-multiple-data). Each processing unit involved can operate a different instruction, on different data element(s). This is the case in multi-core CPUs, or when involving several devices or machines in the computations: each CPU can function completely independent of the others.

The increased flexibility for MIMD over SIMD (and SIMD over MIMD) comes at a price, the main one being a greater complexity cost (and hence price, power consumption, etc.) in building the hardware. For this reason, today's SIMD hardware attains a much larger total performance than equally complicated or costly MIMD hardware, and similarly, SISD hardware delivers much greater single-core performance than equally complicated or costly SIMD hardware. Therefore, it is important to use the right architecture for the right task: some algorithms can be implemented in a way that they can potentially run orders of magnitude faster on SIMD architectures such as GPU; others cannot be implemented efficiently in this way, but can be parallelized on MIMD devices; others cannot be parallelized at all.

To maximize performance, hardware manufacturers perpetually try to embed parallelism deeper into hardware (e.g. through vector processing or pipelining), blurring the distinction between SISD, SIMD and MIMD. Even single-core processors have some level of data-parallelism (e.g. performing addition/multiplication/... on several numbers at the same time) and task-parallelism (e.g. by having multiple pipelines which can process certain instructions in parallel). Also, inherently data-parallel hardware like GPUs features multiple SIMD blocks in each device, making them technically (as a device) MIMD. For this purpose, we will no longer use SISD/SIMD/MIMD as a classification method for algorithms and devices, but instead we use the following definitions.

Definition A.2.8. A program or algorithm is said to be

- *data-parallel* if it benefits from having several processing elements executing the program, with the additional restriction that each processing unit executes the same line of the program at the same time;
- *task-parallel* if it does not benefit from the above, but does benefit from it without the additional restriction;
- *non-parallel* if it does not benefit from having several processing units executing the program at all.

In the above terminology, we disregard parallelism within the same processing element, such as pipelining and vector instructions. In general, we will execute non-parallel algorithms on fast high-frequency CPUs, we will execute data-parallel algorithms on GPUs, and we will execute task-parallel algorithms on large clusters of low-power consumption CPUs. In this thesis, we will generally refer to SIMD, MIMD, SISD devices as the devices used to perform data-parallel, task-parallel and non-parallel algorithms respectively.

We will provide some examples to demonstrate where which paradigms can be appropriate. For this purpose, let $A = [a_0, a_1, \ldots, a_{1999}]$ be an array of 2000 integers and let $f : \mathbb{Z} \times \mathbb{Z} \to \mathbb{Z}$ be a function (taking some time to compute) whose exact functionality is not known at the time you need to write the algorithm. Assume that you want to compute $f(a_0, f(a_1, f(a_2, f(\cdots f(a_{1998}, a_{1999}) \cdots))))$, and you can use at most 16 processing units.

Example A.2.9. Assume that f is non-associative. Then there is no way to properly parallelize this computation, because the evaluation of any $f()$ requires the result of the evaluations of the inner $f()$ evaluations. Hence, the best way for this is to make a single thread and run the following algorithm on it:

$c = a_{1999}$
for $i = 1998, 1997, \ldots, 0$ **do**
  $c = f(a_i, c)$
end for
return c

Example A.2.10. Assume that f is associative, has a constant runtime T and has only a single execution path (all threads execute the same sequence of instructions). Then we can use the methods described earlier in this section to parallelize this algorithm and run it in parallel on 16 threads with Method 4 in the time of only $124 + 1 + 1 + 1 + 1 = 128$ f-evaluations and 4 barriers (since $2000 = 125 \cdot 16 = 125 \cdot 2^4$), hence running in time $128T$. In particular, we create threads $T_0, \ldots, T_{15}$ and we run the following algorithm on each T_i.

for $j = 1, \ldots, 124$ **do**
  $a_{125i} = f(a_{125i}, a_{125i+j})$
end for
for $k = 0, 1, 2, 3$ **do**
  if $i \equiv 0 \pmod{2^{k+1}}$ **then**
    $a_{125i} = f(a_{125i}, a_{125(i+2^k)})$
  end if
end for

return a_0

which the reader can verify is indeed equivalent to Method 4 with $n_5 = 125$ and $n_1 = n_2 = n_3 = n_4 = 2$, with the modification that the work of $T_{125i}, \ldots, T_{125i+124}$ there, is now executed by T_i here. Since $f()$ follows a single execution path, this is a perfect data-parallel implementation.

Example A.2.11. Assume now that f is still associative and has constant runtime, but it no longer follows a single execution path. Instead, f is a piecewise function with 8 pieces, implemented by an `if`-tree, and $g_0, \ldots, g_7$ are functions that do follow a single execution path. So, f looks as follows.

(determine x and $\gamma_0 < \gamma_1 < \cdots < \gamma_6$ based on input)
if $x < \gamma_3$ **then**
  if $x < \gamma_1$ **then**
    if $x < \gamma_0$ **then**
      return $g_0(x)$
    else
      return $g_1(x)$
    end if
  else
    if $x < \gamma_2$ **then**
      return $g_2(x)$
    else
      return $g_3(x)$
    end if
  end if
else
  if $x < \gamma_5$ **then**
    if $x < \gamma_4$ **then**
      return $g_4(x)$
    else
      return $g_5(x)$
    end if
  else
    if $x < \gamma_6$ **then**
      return $g_6(x)$
    else
      return $g_7(x)$
    end if
  end if
end if

While horrible to read for humans, a CPU finds this very easy code: it only needs to evaluate 3 if-statements to find out which piece needs to be called. In general, a piecewise function with n parts needs only $\lceil \log_2(n) \rceil$ if-statements. This is a very useful way to approximate functions that would otherwise be complex to compute.

On a MIMD architecture, the execution time would still be $128T$. On a SIMD architecture however, part of the parallelization will be gone: we will suddenly need (at most) $8 \cdot 128T = 1024T$ time to execute the program. The reason for this is as follows: if for at least one of the threads, $x < \gamma_0$, then the instructions for $g_0()$ are loaded on the instruction stream. Since all threads share this instruction stream, the threads with $x \geq \gamma_0$ have to wait for the threads with $x < \gamma_0$ to finish $g_0()$ before the instructions for $g_1()$ can be loaded. In general, when evaluating performance of conditional statements on SIMD architectures, one should keep in mind that different branches cannot be executed simultaneously as they require different instructions. As a rule of thumb, it can be said that the execution time of a function allowing more than one execution path on a SIMD device, will have as its execution time that of the union of all its execution paths. For this reason, algorithms that heavily rely on conditional logic will usually be better off on a MIMD architecture.

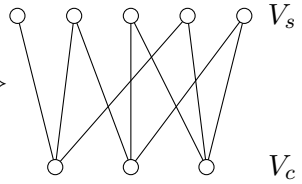
A.3 The LDPC decoding algorithm

In this section, we will explain how the belief propagation/sum-product LDPC decoding algorithm works.

Definition A.3.1 ([133])**.** Let $H \in \mathbb{F}_q^{m \times n}$ be any matrix. Let V_s be the set of columns of H and let V_c be the set of rows of H. Then the *Tanner graph* of H is the bipartite graph on $V_s \cup V_c$ with edge set $E = \{(i,j) \in V_s \times V_c : H_{ji} \neq 0\}$. Even though the graph is undirected, we will consider edges as pairs in $V_s \times V_c$ for notational simplicity.

Usually, Tanner graphs are only considered of $\{0,1\}$-matrices, making the condition $H_{ji} \neq 0$ equivalent to $H_{ji} = 1$.

$$\begin{bmatrix} 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 \end{bmatrix} \Rightarrow$$



When H is used as the parity check matrix of an LDPC code, then V_s can be interpreted as the set of symbols of C, and V_c can be interpreted as the set of parity check equations that a code word in C has to fulfill.

Notation A.3.2. The edge (i,j) will be denoted by e. Hence, whenever this section mentions $e \in E$, this should be read as $(i,j) \in E \subseteq V_s \times V_c$.

Definition A.3.3. To simplify the notations in the coming section, we introduce the following additional symbols.

- We denote $V_i = \{j \in V_c | (i,j) \in E\}$ and $V_j = \{i \in V_s | (i,j) \in E\}$.
- By $F_{i,j}$ we denote the condition "$\forall j' \in V_i \setminus \{j\} : \sum_{i' \in V_{j'}} c_{i'} = 0$", i.e. the condition that all check equations adjacent to i, with the exception of c_j, are fulfilled.

The common model to study coding theory, is to assume that a certain corrupted word $w = (w_i)_{i \in V_s}$ is received and that the statistical distribution of the noise is known, so that we can correctly determine the symbolwise probability distribution $\mathfrak{p}$. That is, for each symbol $i \in V_s$ and for each $a \in \mathcal{A}$, the probability

$$\mathfrak{p}_{a,i} = P(c_i = a | w \text{ is received})$$

is assumed to be known. Obviously, $\sum_{a \in \mathcal{A}} \mathfrak{p}_{a,i} = 1$ for each $i \in V_s$. This matrix $\mathfrak{p} = (\mathfrak{p}_{a,i})_{\substack{a \in \mathcal{A} \\ i \in V_s}}$ is the input of any decoder.

In a practical setting, one nearly always works in the *AWGN model*, i.e. the Additive Gaussian White Noise model. Other channel models exist as well, but they fall beyond the scope of this text. In the AWGN model, the two symbols of the binary alphabet are encoded as ± 1, and for each position one takes an independent ('white') sample from an $N(0, \sigma^2)$ distribution ('Gaussian'), which is then added to the ± 1 ('additive').

Definition A.3.4. The *performance* of an AWGN decoder is the probability distribution which maps σ to the probability that an arbitrary message $c \in C$ can be retrieved successfully by the decoder after adding this $N(0, \sigma^2)$ term to it.

Several upper bounds on the performance of a decoder exist, the most well-known one being the Shannon limit. The most common way to measure the performance $perf$, given that we are mainly interested in approximations of $\log(1 - perf)$ and only when the performance is very high ($p_{err} > 0.99999$), is to put noise on random code words and decode them, until 100 wrong decodings have been made. If N code words were required to get there, then the estimated performance is $1 - \frac{100}{N}$.

The key idea behind belief propagation is to farm out the computations to the symbol and check nodes, rather than looking for a global solution (such as maximum likelihood decoding). The symbol nodes and check nodes will iteratively exchange information over the network, and all of the computations will happen in the symbol nodes and check nodes, using only the information they have available at that point. In each iteration, each symbol node passes information to each of its adjacent check nodes, after which each check node passes information to each of its adjacent symbol nodes. Hence, the information is always passed along the edges of the Tanner graph. First we will study how exactly this information is exchanged over the network, and then we will take a look at which information has to be exchanged.

A naive method would be to make each symbol node $i \in V_s$ pass all the information it has to each $j \in V_i$; and after this, make each check node $j \in V_c$ pass all the information it receives, plus its own information, to each $i \in V_j$. Then, after t iterations, the information from node i will have reached every vertex at graph distance at most $2t$ from i. However, the same information will arrive at the same vertex multiple times, since every two adjacent vertices will perpetually send their message back and forth over the same edge.

There is an easy remedy for this, allowing the same information to be transferred with less overhead: instead of the naive approach above, we let each node pass all of the information it received, to all of its adjacent nodes *except for the node from which it received the information.* From an information theoretic point of view, this alternative is completely equivalent: each

node will get each piece of information for the first time in the same iteration as in the naive approach; getting the same information again at a later point in time has no intrinsic contribution anymore. However, the amount of required messages lowers significantly: the amount of messages is multiplied by $\deg(v)-1$ instead of $\deg(v)$ at vertex $v \in V$. Since LDPC codes have a very sparse Tanner graph, the vertex degrees are usually very low, making this a significant improvement.

Definition A.3.5. This way of transmitting information along the edges of a graph, sending information to all adjacent nodes except for the one from which the information was received, is called *propagation*.

It is easy to see that, after t iterations, a vertex $v \in V$ has received the information from $v' \in V$ if and only if there exists a propagation walk of length $d(v, v') \leq t$ in the graph. We will now have a look at which messages are being propagated. In a practical setting, a node cannot pass all of the information that it received, so it will only propagate part of it. In particular, the messages passed will be estimates of certain probabilities, computed in the nodes. Hence, each node propagates its *beliefs* on certain probabilities over the network. This justifies the name of the algorithm: what we do is indeed *belief propagation*. Another, less common name for this algorithm, is the *message passing algorithm* (MPA).

Denote by $P_{a,i}$ the current estimate by symbol node i of the probability

$$P\left(c_i = a \middle| w, \forall j \in V_i : \sum_{i' \in V_j} c_{i'} = 0\right),$$

i.e. the probability that the ith symbol is a, given that all parity check equations are fulfilled and the word w is received. The best estimate for the received code word is then the word $(\arg\max_{a \in \mathcal{A}} P_{a,i})_{i \in V_s}$. We will iterate this propagation from symbol nodes to check nodes and back, until

$$\left(\arg\max_{a \in \mathcal{A}} P_{a,i}\right)_{i \in V_s} \in C.$$

To prevent the algorithm from getting stuck in an infinite loop, we will also break the algorithm when a certain maximum number of iterations is reached (in which case decoding fails and no valid code word is returned).

Remark A.3.6. Note that in some rare cases, decoding a highly corrupted word may result in a valid code word, different from the code word transmitted. However, the number of errors caused in this way, is usually neglectable compared to the number of errors caused by the algorithm not returning a code word (by reaching its maximum number of iterations).

Remark A.3.7. Sometimes, two (or more) elements $a_1, a_2 \in \mathcal{A}$ have

$$P_{a_1,i} = P_{a_2,i} = \max_{a \in \mathcal{A}} P_{a,i}.$$

In such an event, $\arg\max$ returns E, an *erasure*. On erasure channels, this is exactly the definition of an erasure, and on non-erasure channels the probability of such a tie is zero anyway; hence, this will cause no harm.

Initially, $P_{a,i} = \mathfrak{p}_{a,i}$ for each $i \in V_s$ and each $a \in \mathcal{A}$, since the condition $\forall j \in V_i : \sum_{i' \in V_j} c_{i'} = 0$ is meaningless if i has not yet received any information from any of the check nodes. If $(\arg\max_{a \in \mathcal{A}} \mathfrak{p}_{a,i})_{i \in V_s} \in C$, then no decoding is necessary to reconstruct the transmitted code word. If decoding is necessary, the symbol nodes $i \in V_s$ and check nodes $j \in V_c$ will alternatingly propagate over $e = (i, j)$ the message vector $q_e = (q_{a,e})_{a \in \mathcal{A}}$ with

$$q_{a,e} = \hat{P}\left(c_i = a | w, F_{i,j}\right)$$

from i to j, and $r_e = (r_{a,e})_{a \in \mathcal{A}}$ with

$$r_{a,e} = \hat{P}\left(\sum_{i' \in V_j} c_{i'} = 0 \middle| (c_{i'} = a)_{\substack{a \in \mathcal{A} \\ i' \in V_j}}\right)$$

from j to i. We hereby remind that $F_{i,j}$ was defined in Definition A.3.3 and that w stands for the word that was received.

Unfortunately, correctly determining the best estimate for $\hat{P}$ in q_e turns out to be difficult, perhaps even more difficult than the original maximum likelihood decoding problem that we were trying to avoid. The way in which this is commonly done for LDPC codes, is to make an *independence assumption*: since we assume the exclusion of the information received over an edge e when determining the message to send back over e, the probabilities for each check node $j \in V_i$ to be satisfied, are statistically independent until the information has went through an entire cycle of the graph. To simplify the computations, the decoding algorithm assumes that the messages will be independent throughout the entire duration of the decoding.

While this is clearly incorrect, and voids the mathematical correctness of the probabilities obtained, the loss of performance suffered in this way is experimentally observed to be rather small. Since the Tanner graphs used for LDPC decoding usually have large girth, the assumption is correct during the first few iterations; and even after that the amount of information received in this non-independent way is only responsible for a small fraction of the total amount of information received, causing only slight distortions. Therefore, even when the graph does have cycles, this independence assumption turns out to be approximately correct, probably due to the sparseness and high girth of the Tanner graph.

Theorem A.3.8. *Under the independence assumption, the estimate sent by $i \in V_s$ along edge $e = (i, j)$ is $q_e = (q_{a,e})_{a \in \mathcal{A}}$ with*

$$q_{a,e} = \frac{\mathfrak{p}_{a,i} \cdot \prod_{j' \in V_i \setminus \{j\}} r_{a,(i,j')}}{\sum_{a' \in \mathcal{A}} \mathfrak{p}_{a',i} \cdot \prod_{j' \in V_i \setminus \{j\}} r_{a',(i,j')}}.$$

In the first iteration, when no prior $r_{a,e}$ was received, this reduces to $q_{a,e} = \mathfrak{p}_{a,i}$.

Proof. By the general probability formula

$$P(A|B,C) = \frac{P(A \cap B|C)}{P(B|C)} = P(B|A,C)\frac{P(A|C)}{P(B|C)},$$

one has

$$P\left(c_i = a|w, F_{i,j}\right) = P\left(F_{i,j}|w, c_i = a\right) \cdot \frac{P\left(c_i = a|w\right)}{P\left(F_{i,j}|w\right)}.$$

Moreover, one has $P\left(c_i = a|w\right) = \mathfrak{p}_{a,i}$ and, by the approximate independence assumption,

$$P\left(F_{i,j}|w, c_i = a\right) = \prod_{j' \in V_i \setminus \{j\}} P\left(\sum_{i' \in V_{j'}} c_{i'} = 0 \middle| w, c_i = a\right).$$

Hence,

$$q_{a,e} = \frac{\mathfrak{p}_{a,i} \cdot \prod_{j' \in V_i \setminus \{j\}} P\left(\sum_{i' \in V_{j'}} c_{i'} = 0 \middle| w, c_i = a\right)}{P\left(F_{i,j}|w\right)} = \frac{\mathfrak{p}_{a,i} \cdot \prod_{j' \in V_i \setminus \{j\}} r_{a,(i,j')}}{P\left(F_{i,j}|w\right)}.$$

Since $P\left(F_{i,j}|w\right)$ does not depend on a and since $\sum_{a \in \mathcal{A}} q_{a,e} = 1$ for each $e \in E$, we may eliminate $P(F_{i,j}|w)$ by dividing $q_{a,e}$ by $\sum_{a \in \mathcal{A}} q_{a,e} = 1$:

$$q_{a,e} = \frac{\mathfrak{p}_{a,i} \cdot \prod_{j' \in V_i \setminus \{j\}} r_{a,(i,j')}}{\sum_{a' \in \mathcal{A}} \mathfrak{p}_{a',i} \cdot \prod_{j' \in V_i \setminus \{j\}} r_{a',(i,j')}}.$$

This proves our claim. □

The computation of r_e depends on the structure of the alphabet $\mathcal{A}$. Since we only consider codes over fields, we will assume that $\mathcal{A} \cong \mathbb{F}_p^h$, where $\mathbb{F}_p$ is the prime field of $\mathcal{A}$.

Theorem A.3.9. *Let $\mathcal{A} \cong \mathbb{F}_p^h$, where $\mathbb{F}_p$ is the prime field of $\mathcal{A}$, and write each $a \in \mathcal{A}$ as $(a_1, \ldots, a_h) \in \mathbb{F}_p^h$. The estimate sent by $j \in V_c$ along edge $e = (i, j)$ is $r_e = (r_{a,e})_{a \in \mathcal{A}}$ with*

$$r_{a,e} = \left(\left(\prod_{i' \in V_j \setminus \{i\}} \left(\sum_{a \in \mathcal{A}} q_{a,(i',j)} t_1^{a_1} t_2^{a_2} \cdots t_h^{a_h} \right) \right) \mod t_1^{p-1} - 1, \ldots, t_h^{p-1} - 1 \right)_{t_1 = \cdots = t_h = 0},$$

where $f(t_1, \ldots, t_h)_{t_1 = \cdots = t_h = 0}$ denotes the evaluation of $f(0, \ldots, 0)$.

Proof. Let K be any index set and let $(c_k)_{k \in K} \in A^K \cong (\mathbb{F}_p^h)^K$. Since two elements are equal if and only if their $\mathbb{F}_p$-components are equal, the generating function associated to the probability distribution

$$P\left(\sum_{k \in K} c_k = (n_1, \ldots, n_h) \middle| \left(P(c_k = a)\right)_{\substack{a \in \mathcal{A} \\ k \in K}} \right),$$

is

$$\prod_{k \in K} \sum_{a \in \mathcal{A}} P(c_k = a) \cdot t_1^{a_1} t_2^{a_2} \cdots t_h^{a_h},$$

and the stated probability is given by the coefficient $t_1^{n_1} \cdots t_h^{n_h}$ in the generating function.

In this case, we are interested in the sum $\mod p$. Hence, we will take all powers of $t_1, \ldots, t_h$ $\mod p$, which is algebraically equivalent to reducing the polynomial $\mod t^p - 1$ for each variable $t \in \{t_1, \ldots, t_h\}$. Moreover, the case $a = 0$ is given by the coefficient of $t_1^0 t_2^0 \cdots t_h^0$, which is easily obtained by substituting $t_1 = \cdots = t_h = 0$ in the result. Plugging in $K = \{i' : V_j \setminus i\}$ yields the predicted message to send over $e = (i, j)$. □

Algorithm 4 The original probability-domain belief propagation algorithm

Input: the probability distribution $\mathfrak{p} = (\mathfrak{p}_{a,i})_{\substack{a \in \mathcal{A} \\ i \in V_s}}$ and the maximum number of iterations M

Output: the retrieved code word (or FAIL)

$(P_{a,i})_{\substack{a \in \mathcal{A} \\ i \in V_s}} \leftarrow (\mathfrak{p}_{a,i})_{\substack{a \in \mathcal{A} \\ i \in V_s}}$
if $(\arg\max_{a \in \mathcal{A}} \mathfrak{p}_{a,i})_{i \in V_s} \in C$ **then**
  return $(\arg\max_{a \in \mathcal{A}} \mathfrak{p}_{a,i})_{i \in V_s}$
end if
for $e = (i, j) \in E$ **do**
  $q_e = (\mathfrak{p}_{a,i})_{a \in \mathcal{A}}$
end for
for $m = 1, \ldots, M$ **do**
  for $e \in E$ **do**
    for $a \in \mathcal{A}$ **do**
      $r_{a,e}$ = the long expression from Theorem A.3.9 (or Theorem A.3.10 if $\mathcal{A} = \mathbb{F}_2$)
    end for
  end for
  for $e \in E$ **do**
    $S_e = \sum_{a \in \mathcal{A}} \mathfrak{p}_{a,i} \cdot \prod_{j' \in V_i \setminus \{j\}} r_{a,(i,j')}$
    $q_e = \frac{1}{S_e} \left(\mathfrak{p}_{a,i} \cdot \prod_{j' \in V_i \setminus \{j\}} r_{a,(i,j')} \right)_{a \in \mathcal{A}}$
  end for
  if $\left(\arg\max_{a \in \mathcal{A}} \left(\mathfrak{p}_{a,i} \cdot \prod_{j \in V_i} r_{a,e} \right) \right)_{i \in V_s} \in C$ **then**
    return $\left(\arg\max_{a \in \mathcal{A}} \left(\mathfrak{p}_{a,i} \cdot \prod_{j \in V_i} r_{a,e} \right) \right)_{i \in V_s}$
  end if
end for
return FAIL

The computation of $P_{a,i}$ is completely similar to the computation of $q_{a,e}$, except that no $j \in V_i$ is included. Hence, we put

$$P_{a,i} = \frac{\mathfrak{p}_{a,i} \cdot \prod_{j \in V_i} r_{a,e}}{\sum_{a' \in \mathcal{A}} \mathfrak{p}_{a',i} \cdot \prod_{j \in V_i} r_{a',e}}.$$

In practice, one can omit the denominator, since we're only interested in $\arg\max_{a \in \mathcal{A}} P_{a,i}$ and not in the actual values of $P_{a,i}$.

All together, this motivates Algorithm 4. In the binary case, this can be simplified further to Algorithm 5, using the fact that $q_{0,e} + q_{1,e} = 1$ and $r_{0,e} + r_{1,e} = 1$ for each $e \in E$, and using the following simplification of Theorem A.3.9.

Theorem A.3.10. *In the binary case, the update rule for $r_{a,e}$ reduces to*

$$r_{a,e} = \frac{1}{2} + (-1)^a \frac{1}{2} \prod_{i' \in V_j \setminus \{i\}} \left(1 - 2q_{a,(i',j)}\right).$$

Proof. Denote by $p_k = P(c_k = 1)$, then the generating function from Theorem A.3.9 simplifies to

$$\prod_{k \in K} ((1 - p_k) + p_k t).$$

By induction, we now prove that the sum of the even-powered coefficients (which is the evaluation at $t = 0$ after reducing modulo $t^2 - 1$) equals $\frac{1}{2} + \frac{1}{2} \prod_{k \in K} (1 - 2p_k)$.

- For $|K| = 0$, this is clear, since an empty product equals the polynomial $1 = \frac{1}{2} + \frac{1}{2}$.
- Assume the statement is true for all K with $|K| = n - 1$ for some positive integer n. Let $k \in K$, then $\sum_{k' \in K \setminus \{k\}} c_{k'} = 1$ is equivalent to:

  $$(c_k = 1 \wedge \sum_{k' \in K \setminus \{k\}} c_{k'} = 1) \vee (c_k = 0 \wedge \sum_{k' \in K \setminus \{k\}} c_{k'} = 0),$$

  where both sides of the $\vee$ are mutually disjoint. Hence, the coefficient of t^0 in

  $$((1 - p_k) + p_k t) \prod_{k' \in K \setminus \{k\}} ((1 - p_{k'}) + p_{k'} t) \mod t^2 - 1$$

  equals (by the induction hypothesis)

  $$p_k \left(\frac{1}{2} - \frac{1}{2} \prod_{k' \in K \setminus \{k\}} (1 - 2p_{k'}) \right) + (1 - p_k) \left(\frac{1}{2} + \frac{1}{2} \prod_{k' \in K \setminus \{k\}} (1 - 2p_{k'}) \right),$$

  which is

  $$\frac{1}{2} + \frac{1}{2} \prod_{k' \in K \setminus \{k\}} (1 - 2p_{k'})$$

  as claimed.

Hence, plugging in $K = \{i' : V_j \setminus \{i\}\}$ yields the predicted message to send over $e = (i, j)$. □

Remark A.3.11. Despite the way it is presented (which is optimized for readability), Algorithm 4 runs in $\mathcal{O}(|E|)$ time and $\mathcal{O}(|E|)$ memory, for any fixed alphabet $\mathcal{A}$. This is because we can simplify the computations of $\prod_{i' \in V_j \setminus \{i\}}$ and $\prod_{j' \in V_i \setminus \{j\}}$ by first precomputing $\prod_{i \in V_j}$ and $\prod_{j \in V_i}$ (which can be done in $\mathcal{O}(|E|)$ time since each edge appears in exactly one product of each of both types), and then dividing out the factor for the edge to exclude.

Much effort has been put in finding ways to decrease the complexity of Algorithm 4. When applying Algorithm 4 to binary codes, Theorem A.3.10 already takes away the computational burden of working with polynomials instead of just real numbers, but further improvements can be made. In particular, applying a simple transformation on the values that are passed along the network, one can convert both products into sums, which not only decreases the complexity, but also increases the numerical stability of the algorithm. In particular, we can state the following result.

Theorem A.3.12. *Algorithm 5 is equivalent to Algorithm 4 for binary codes.*

Proof. Since $r_{0,e} + r_{1,e} = 1$ and $q_{0,e} + q_{1,e} = 1$, it is not necessary to transmit both. We will replace $r_{0,e}$ and $q_{0,e}$ by $1 - r_{1,e}$ and $1 - q_{1,e}$ everywhere. Moreover, we will shorten $q_{1,e}$ and $r_{1,e}$ to q_e and r_e. Denote $Lr_{ij} = \ln\left(\frac{1-r_{1,e}}{r_{1,e}}\right)$ and $Lq_{ij} = \ln\left(\frac{1-q_{1,e}}{q_{1,e}}\right)$ for each $e = (i, j) \in E$. Denote $L\mathfrak{p}_i = \ln\left(\frac{\mathfrak{p}_{0,i}}{\mathfrak{p}_{1,i}}\right)$ and $LP_i = \ln\left(\frac{P_{0,i}}{P_{1,i}}\right)$.

The update rule

$$r_e = \frac{1}{2} - \frac{1}{2} \prod_{i' \in V_j \setminus \{i\}} \left(1 - 2q_{(i',j)}\right)$$

can be rewritten as

$$1 - 2r_e = \prod_{i' \in V_j \setminus \{i\}} \left(1 - 2q_{(i',j)}\right).$$

Observe that $1 - 2x = (1 - x) - (x) = \tanh\left(\frac{1}{2}\ln\left(\frac{1-x}{x}\right)\right)$, which rewrites the update rule as

$$\tanh\left(\frac{1}{2}Lr_{ij}\right) = \prod_{i' \in V_j \setminus \{i\}} \tanh\left(\frac{1}{2}Lq_{i'j}\right).$$

Now, split of the signs: denote $\alpha_{ij} = \begin{cases} -1 & \text{if } LQ_{ij} < 0, \\ 1 & \text{if } LQ_{ij} > 0 \end{cases}$ and $\beta_{ij} = |LQ_{ij}|$, such that $LQ_{ij} = \alpha_{ij}\beta_{ij}$. Since $\tanh(-x) = -\tanh(x)$ for all $x \in \mathbb{R}$ and tanh is invertible on $]-1, 1[$, this allows the update rule to be rewritten as

$$Lr_{ij} = \prod_{i' \in V_j \setminus \{i\}} \alpha_{i'j} 2 \tanh^{-1} \prod_{i' \in V_j \setminus \{i\}} \tanh\left(\frac{1}{2}\beta_{i'j}\right).$$

Since

$$\prod_{i' \in V_j \setminus \{i\}} \tanh\left(\frac{1}{2}\beta_{i'j}\right) = \ln^{-1} \sum_{i' \in V_j \setminus \{i\}} \ln\tanh\left(\frac{1}{2}\beta_{i'j}\right),$$

the update rule can be rewritten further as

$$Lr_{ij} = \prod_{i' \in V_j \setminus \{i\}} \alpha_{i'j} \phi^{-1}\left(\sum_{i' \in V_j \setminus \{i\}} \phi(\beta_{i'j})\right)$$

Algorithm 5 The standard LLR-BP algorithm

Input: the probability distribution $\mathfrak{p} = (\mathfrak{p}_{a,i})_{\substack{a\in\mathbb{F}_2\\ i\in V_s}}$ and the maximum number of iterations M

Output: the retrieved code word (or FAIL)

for $i \in V_s$ **do**
  $L\mathfrak{p}_i = \ln\left(\frac{\mathfrak{p}_{0,i}}{\mathfrak{p}_{1,i}}\right)$
  $LP_i = L\mathfrak{p}_i$
  $c_i \leftarrow \begin{cases} 1 & \text{if } LP_i < 0, \\ 0 & \text{if } LP_i > 0 \end{cases}$
end for
if $(c_i)_{i\in V_s} \in C$ **then**
  return $(c_i)_{i\in V_s}$
end if
for $(i,j) \in E$ **do**
  $LQ_{ij} \leftarrow LP_i$
end for
for $m = 1, \ldots, M$ **do**
  for $(i,j) \in E$ **do**
    $\alpha_{ij} \leftarrow \begin{cases} -1 & \text{if } LQ_{ij} < 0, \\ 1 & \text{if } LQ_{ij} > 0 \end{cases}$
    $\beta_{ij} \leftarrow |LQ_{ij}|$
  end for
  for $j \in V_c$ **do**
    $B_j \leftarrow \sum_{i\in V_j} \phi(\beta_{ij})$
    $A_j \leftarrow \prod_{i\in V_j} \alpha_{ij}$
  end for
  for $(i,j) \in E$ **do**
    $LR_{ji} \leftarrow A_j\alpha_{ij} \cdot \phi\left(B_j - \phi(\beta_{ij})\right)$
  end for
  for $i \in V_s$ **do**
    $LP_i \leftarrow L\mathfrak{p}_i + \sum_{j\sim i} LR_{j'i}$
    $c_i \leftarrow \begin{cases} 1 & \text{if } LP_i < 0, \\ 0 & \text{if } LP_i > 0 \end{cases}$
  end for
  if $(c_i)_{i\in V_s} \in C$ **then**
    return $(c_i)_{i\in V_s}$
  end if
  for $(i,j) \in E$ **do**
    $LQ_{ij} \leftarrow LP_i - LR_{ji}$
  end for
end for

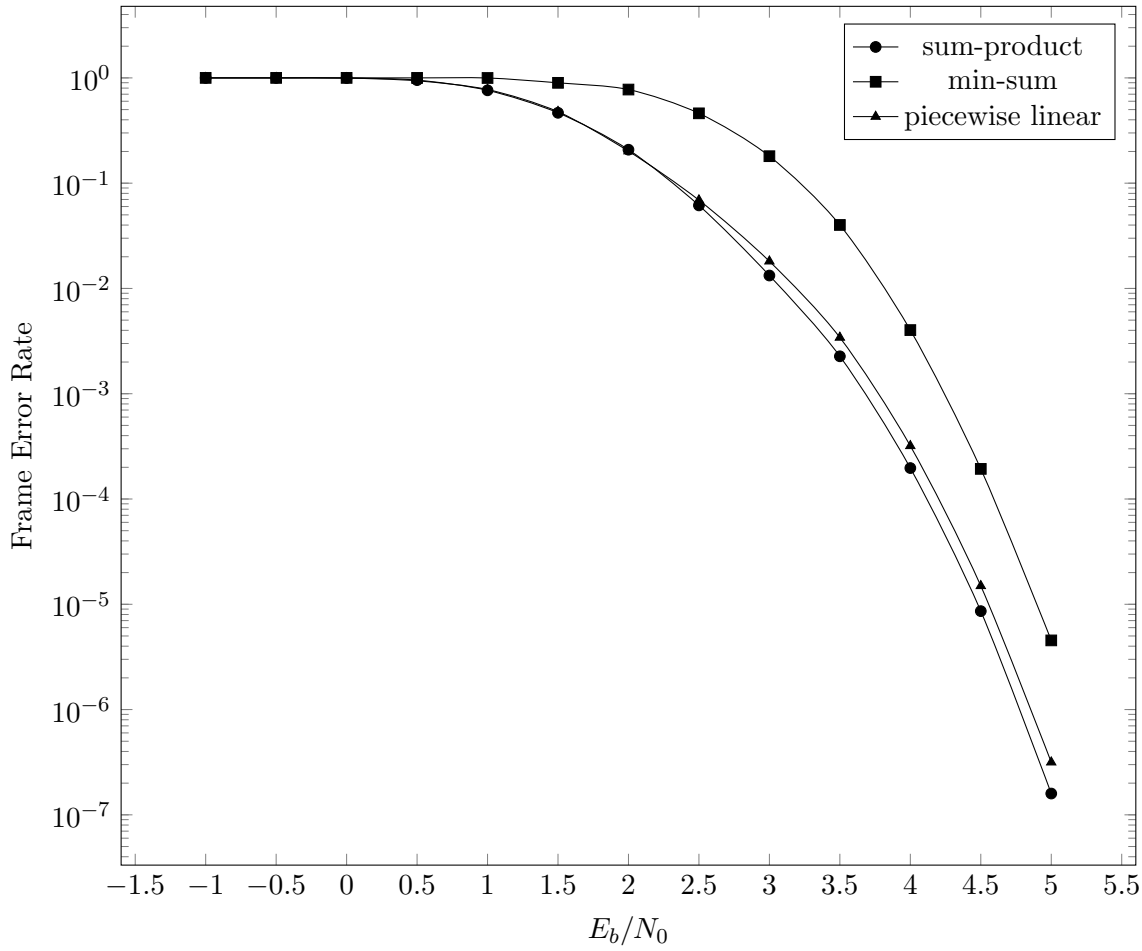


Figure A.2: Performance of the sum-product algorithm, its piecewise linear approximation, and the min-sum algorithm over an AWGN channel, using the PG(2, 16) LDPC code

with $\phi : \mathbb{R}^+ \to \mathbb{R}^+ : x \mapsto -\ln\left(\tanh\left(\frac{x}{2}\right)\right)$ (which is its own inverse). The other modifications should be obvious, being merely log-transforming both sides of an equation, and precomputing the sums (which used to be products) as in Remark A.3.11. Note that $\prod_{i' \in V_j \setminus \{i\}} \alpha_{i'j}$ is a XOR of booleans, which is faster than summation and does not suffer from numerical instability; this also explains why we write the more elegant $A_j \alpha_{ij}$ instead of $\frac{A_j}{\alpha_{ij}}$. □

It is clear that Algorithm 5 runs in $\mathcal{O}(|E|)$ time and $\mathcal{O}(|E|)$ RAM & ROM memory. In the literature, Algorithm 5 is commonly referred to as the *sum-product algorithm* (SPA), *belief propagation* or *message passing algorithm* (MPA). Occasionally *MAP* is also used.

Remark A.3.13. The main computational burden in Algorithm 5 is the computation of ϕ. We present the two common low-complexity approximations below. Many hardware simplifications have also been studied, but this falls beyond the scope of this text.

- The min-sum algorithm (occasional alternative names: *sig-min algorithm* or *max-log-MAP decoding* or *UMP BP-based decoding*). Since $\lim_{x \to 0^+} \phi(x) = +\infty$, the value of

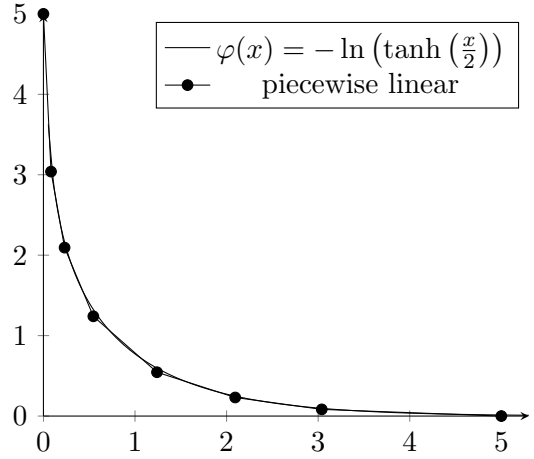


Figure A.3: φ versus its piecewise linear approximation

$\sum_{x\in X} \phi(x)$ can be approximated by $\phi\left(\min_{x\in X} x\right)$. Hence, with this approximation, the update rule reduces to

$$Lr_{ij} = \prod_{i'\in V_j\setminus\{i\}} \alpha_{i'j} \min_{i'\in V_j\setminus\{i\}} \beta_{i'j}.$$

Since this new approximation is always slightly larger (in absolute value) than the original Lr_{ij}, one sometimes divides each Lr_{ij} by a correction factor $\alpha > 1$.

- Piecewise linear approximation. Using only three if-statements, ϕ can be approximated by eight line segments. This turns out to be a simple but effective remedy, which manages to approximate φ quite well, as can be seen in Figure A.3. Similarly, φ can be implemented via a lookup table (i.e. a piecewise constant function), which requires no floating point computations.

A comparative chart, displaying the performance of these approximations and of the original algorithm on the PG$(2, 16)$ code over an AWGN channel (the most common model for modern coding theory, where binary signals are modulated as $\{-1, 1\} \to \mathbb{R}$ and the noise has a normal distribution), is given in Figure A.2.

Another important issue is the memory complexity of Algorithm 5. The RAM memory requirements are responsible for the majority of the physical size, power consumption and production price of the decoder [3, 69, 68, 99], despite serious attempts the memory usage and physical properties have been reduced by a constant factor only [28, 109, 117, 138, 151].

A.4 An OpenCL implementation

In this section I will explain how I implemented Algorithm 5 in OpenCL, as an example of how to implement nontrivial algorithms on GPU. While I have implemented the entire algorithm on GPU, as well as all the tools to generate random data, I will focus this section on the inner loop starting at line 12, since this is the computationally relevant part. Meanwhile, I

will explain the relevant parts of how a GPU differs computationally from a CPU in terms of latency, thread scheduling and vector computing.

What lines 18, 19 and 22 do is essentially computing

$$LR_{ji} = \left(\prod_{i' \in V_j \setminus \{i\}} \alpha_{i'j} \right) \left(\sum_{i' \in V_j \setminus \{i\}} \phi(\beta_{i'j}) \right).$$

Of course, recomputing this partial sum for every index is a terrible waste of resources, so therefore we first compute the total sum and product in lines 18 and 25, and then substract the relevant term and factor in line 22. A similar trick is written in lines 25 and 32. While in theory this works perfect, a straightforward implementation trashes the performance of the decoding pretty quickly: once we are very sure about a certain position, the quantity to ignore in the sum becomes very large, and gets rounded by the hardware to $+\infty$. Then substracting it again yields $\infty - \infty$ which is undetermined ('NaN' value in OpenCL, for 'Not a Number'). This NaN propagates over the Tanner graph (adding or substracting NaN to/from any number yields again NaN) and the decoding process fails. Therefore, we will need to modify the algorithm so that it not just sums up the numbers, but explicitly counts the positive and negative infinities: we will represent every number as $a\infty + b$ with a an integer and b a floating point number, with the additional conventions that $(a\infty+b)+c = a\infty+(b+c)$ for $c \in \mathbb{R}$, $(a\infty + b) + \infty = (a+1)\infty + b$ and $(a\infty + b) - \infty) = (a-1)\infty + b$, where obviously $(a\infty+b) > 0$ holds if and only if either $a > 0$ or ($a = 0$ and $b > 0$). This representation is still only an approximation (since $b + c$ may be so large that it is again rounded to infinity), but then again, so is every numerical rounding. These are unavoidable in numerical computing, and with this extra convention no NaN values are created and propagated.

Next, we need to decide what each thread should be executing. OpenCL is designed to distribute a large amount of work (in this case: decoding a large number N of received vectors) over a large number of small local workgroups within which synchronization methods are available to optimally use an entire compute unit (64 cores) to complete this job as soon as possible. Since different received words are unrelated, but every position of the code word could potentially matter for every other position in the final decoded word, the natural choice here is to choose a workgroup to cover exactly one received noisy code word. As is explained in the previous section, the main idea behind belief propagation decoding is to farm out the computational work to the vertices of the graph; hence, it is natural to give each vertex of the graph its own logical core, and to somehow map these $|V_s| + |V_c|$ logical cores to the 64 physical cores in a way that allows optimum parallelism. Since V_s and V_c are only active in turns, one natural way to do this is to let core t (with $t \in \{0, \ldots, 63\}$) do the computations for node $t \left\lfloor \frac{|V_s|}{64} \right\rfloor$ up to $(t+1) \left\lfloor \frac{|V_s|}{64} \right\rfloor - 1$, of course paying attention that these numbers don't go beyond $|V_s|$. Similar for V_c.

The if-condition in line 28 is done by computing for all $j \in V_c$ the value of $\left(\sum_{i \in V_j} LP_i\right)$ mod 2. Unfortunately, there is no easy way to check in parallel if all elements in that array are zero. Here, we need utilize a trick similar to the one in Example A.2.3 to sum up the remainders modulo 2 and then have one single core return the output if that sum is zero.

To make optimal use of the computational capabilities of the GPU on 128-bit registers, one

needs to fill the entire register to perform the numerical computations. Using 64-bit double-precision floats to represent numbers, this means we can do two $\phi()$-evaluations at a time per core, or two additions/multiplications/... which yields a factor 2 speedup compared to the naive implementation. The most efficient way to do this is to just make groups of two code words and let the local workgroup act on two code words at once. This way we fully utilize the potential of the computing cores on the card.

To avoid branching as discussed at the end of Section A.2, we need to write the if-statements in such a way that they do not require branching. For this purpose, OpenCL provides vector functions that write the output of certain logical boolean operators (like whether a number is finite or larger than another number) to an integer vector with values either 0 (which is stored by all zeroes bitwise) or -1 (which is stored by all ones bitwise). This way, we can avoid branching: for example, when adding the value of `temp` to `sum` only when they're finite, one could use

```
sum += as_double2(isfinite(temp) & as_long2(temp));
```

to replace the branching if-statement by a non-branching logical 'and'.

Next to optimizing the kernel, it is also important to optimize the way the kernel is invoked. The drivers manageing the GPU computations have a *watchdog timer* which monitors applications to see if they're still responsive. If a single kernel is running for more than five seconds, it will get interrupted and the application running it crashes. On the other hand, stopping and launching a new kernel has very little overhead. Therefore, we will just let one kernel execute one loop of the algorithm, and run the iteration loop in the host program. To make sure the kernel runs about one second (which is more than enough to make kernel execution overhead neglectable) we can flexibly adjust the number of code words processed in one iteration of the kernel. Processing more code words in the same kernel has the additional benefit that the scheduler can better utilize parallelism. On one hand GPU memory access has a high latency and having many different code words to process per kernel helps the scheduler to make the cores do computations for other code words while waiting for memory access requests to resolve; on the other hand more code words make the distribution of the work over the compute units (i.e. the different groups of 64 cores in the CPU) more evenly, which also improves optimum usage of the available computing resources.

To finish, I give the reader the raw output of my OpenCL kernel on the next pages, to provide an idea of how an implemented OpenCL kernel looks like. Feel free to copy and use, since at this point there are no such general purpose OpenCL implementations of the LDPC decoding algorithm publicly available.

```
inline double2 phi (double2 x) {
    return -log(tanh(x/2));
}

__kernel void oneIteration(
    __global const int *cumulRowDegree, //int[height+1]
    __global const int *cumulColumnDegree, //int[length+1]
    __global const int *edgeRow,
    __global const int *edgeColumn,
    __global const int *edgesInRow, //stores for each row the edge indices in it
    __global const int *edgesInColumn, //stores for each column the edge indices in it
    __global const int *positionsOfOnes,
    const int length,
    const int height,
    const int numberOfOnes,
    __global int2 *iter,
    __global double2 *aij,
    __global double2 *phibetaij,
    __global double2 *Lqij,
    __global double2 *Lrji,
    __global double2 *Lci,
    __global short2 *LQihard,
    const double sigma,
    const double p,
    __global short2 *detectedErrorInWord,
    __global int2 *hiddenErrorInWord,
    __global const long2 *inputcw,
    __global short2 *checkbits
){
  int wordId = get_global_id(1);
  if (detectedErrorInWord[wordId].x || detectedErrorInWord[wordId].y) {
    int eOffset = get_global_id(1)*numberOfOnes;
    int iOffset = get_global_id(1)*length;
    int jOffset = get_global_id(1)*height;

    //initialize the start and stop values for this work item
    // when iterating over the edges or vertices
    int eNumberPerThread = (int)ceil((double)numberOfOnes/(double)get_global_size(0));
    int eStart = get_global_id(0)*eNumberPerThread;
    int eStop = eStart + eNumberPerThread;
    if (eStop>=numberOfOnes) eStop = numberOfOnes;

    int iNumberPerThread = (int)ceil((double)length/(double)get_global_size(0));
    int iStart = get_global_id(0)*iNumberPerThread;
    int iStop = iStart + iNumberPerThread;
    if (iStop>=length) iStop = length;
```

```
int jNumberPerThread = (int)ceil((double)height/(double)get_global_size(0));
int jStart = get_global_id(0)*jNumberPerThread;
int jStop = jStart + jNumberPerThread;
if (jStop>=height) jStop = height;

int threadID = get_global_id(1)*get_global_size(0)+get_global_id(0);
barrier(CLK_GLOBAL_MEM_FENCE);

//if we're in the first iteration, initialize the edges
if (iter[wordId].x==0) {
  for (int e = eStart; e<eStop; e++) {
    Lqij[eOffset+e].x = Lci[iOffset+edgeColumn[e]].x;
  }
  barrier(CLK_GLOBAL_MEM_FENCE);
}
if (iter[wordId].y==0) {
  for (int e = eStart; e<eStop; e++) {
    Lqij[eOffset+e].y = Lci[iOffset+edgeColumn[e]].y;
  }
  barrier(CLK_GLOBAL_MEM_FENCE);
}
if (get_global_id(0)==0 && detectedErrorInWord[wordId].x) iter[wordId].x++;
if (get_global_id(0)==0 && detectedErrorInWord[wordId].y) iter[wordId].y++;

//compute the values of \alpha_{ij} and \phi(\beta_{ij})
for (int e = eStart; e<eStop; e++) {
  long2 temp = signbit(Lqij[eOffset+e]);
  aij[eOffset+e]=1+2*(double2)(temp.x,temp.y);
  phibetaij[eOffset+e]=phi(fabs(Lqij[eOffset+e]));
}
barrier(CLK_GLOBAL_MEM_FENCE);

for (int j=jStart; j<jStop; j++) {
  //compute A_j and B_j for this j
  double2 sum = 0;
  double2 prod = 1;
  long2 inf = 0;
  for (int idx = cumulRowDegree[j]; idx<cumulRowDegree[j+1]; idx++) {
    int e = edgesInRow[idx];
    double2 temp = phibetaij[eOffset+e];
    sum+=as_double2(isfinite(temp) & as_long2(temp));
    prod*=aij[eOffset+e];
    inf -= isinf(temp);
  }
  //compute LR_{ji} for all i adjacent to j
  for (int idx = cumulRowDegree[j]; idx<cumulRowDegree[j+1]; idx++) {
    int e = edgesInRow[idx];
```

```
      double2 temp = phibetaij[eOffset+e];
      long2 localinf = inf + isinf(temp);
      Lrji[eOffset+e] = phi(fabs(sum-as_double2(isfinite(temp)&as_long2(temp))));
      if (localinf.x) Lrji[eOffset+e].x=0;
      if (localinf.y) Lrji[eOffset+e].y=0;
      double2 myprod = prod;
      myprod *= aij[eOffset+e];
      Lrji[eOffset+e]=myprod*Lrji[eOffset+e];
    }
  }
  barrier(CLK_GLOBAL_MEM_FENCE);

  for(int i=iStart; i<iStop; i++) {
    //compute LP_i for this i
    double2 sum = Lci[iOffset+i];
    long2 inf = 0;
    for (int idx = cumulColumnDegree[i]; idx<cumulColumnDegree[i+1]; idx++) {
      int e = edgesInColumn[idx];
      double2 temp = Lrji[eOffset+e];
      sum += as_double2(isfinite(temp) & as_long2(temp));
      inf += isinf(temp)*isgreater(temp,0);
      inf -= isinf(temp)*isless(temp,0);
    }
    double2 LQi = sum;
    if (inf.x>0) LQi.x = INFINITY;
    if (inf.x<0) LQi.x = -INFINITY;
    if (inf.y>0) LQi.y = INFINITY;
    if (inf.y<0) LQi.y = -INFINITY;

    //compute c_i
    LQihard[iOffset+i].x=((LQi.x<0)!=(inputcw[i].x==1))
                      || ((LQi.x>0)!=(inputcw[i].x!=1));
    LQihard[iOffset+i].y=((LQi.y<0)!=(inputcw[i].y==1))
                      || ((LQi.y>0)!=(inputcw[i].y!=1));

    //compute LQ_{ij} for all j adjacent to i
    for (int idx = cumulColumnDegree[i]; idx<cumulColumnDegree[i+1]; idx++) {
      int e = edgesInColumn[idx];
      double2 temp = Lrji[eOffset+e];
      double2 sumpart = sum-as_double2(isfinite(temp)&as_long2(temp));
      long2 infpart = inf-isinf(temp)*isgreater(temp,0)+isinf(temp)*isless(temp,0);
      if (infpart.x>0) sumpart.x = INFINITY;
      if (infpart.x<0) sumpart.x = -INFINITY;
      if (infpart.y>0) sumpart.y = INFINITY;
      if (infpart.y<0) sumpart.y = -INFINITY;
      Lqij[eOffset+e] = sumpart;
    }
```

```
    }
    barrier(CLK_GLOBAL_MEM_FENCE);

    //check if (c_i)_{i\in V_s} \in C
    for (int j=jStart; j<jStop; j++) {
      short2 value = (short2)(1,1);
      for (int idx=cumulRowDegree[j]; idx<cumulRowDegree[j+1]; idx++) {
        int pos = positionsOfOnes[idx];
        value *= (short2)(1,1)-(short2)(2,2)*LQihard[iOffset+pos];
      }
      checkbits[jOffset+j] = ((short2)(1,1)-value)/(short2)(2,2);
    }
    for (int step=1, div=1; step<length; step<<=1, div++) {
      for (int i=iStart; i<iStop; i++) if (i<length-step && i==i>>div<<div) {
        LQihard[iOffset+i]+=LQihard[iOffset+i+step];
      }
      barrier(CLK_GLOBAL_MEM_FENCE);
    }
    for (int step=1, div=1; step<height; step<<=1, div++) {
      for (int j=jStart; j<jStop; j++) if (j<height-step && j==j>>div<<div) {
        checkbits[jOffset+j]|=checkbits[jOffset+j+step];
      }
      barrier(CLK_GLOBAL_MEM_FENCE);
    }

    //prepare the output to be read
    if (get_global_id(0)==0) {
      hiddenErrorInWord[wordId].x=LQihard[iOffset+0].x;
      hiddenErrorInWord[wordId].y=LQihard[iOffset+0].y;
      if (checkbits[jOffset+0].x!=0) {
        detectedErrorInWord[wordId].x=1;
      } else {
        detectedErrorInWord[wordId].x=0;
      }
      if (checkbits[jOffset+0].y!=0) {
        detectedErrorInWord[wordId].y=1;
      } else {
        detectedErrorInWord[wordId].y=0;
      }
    }
  }
}
```

Nederlandstalige samenvatting

In deze Nederlandstalige samenvatting geef ik een kort overzicht van de resultaten die in deze Engelstalige doctoraatsthesis gepresenteerd worden.

Hoofdstuk 1 herhaalt de algemene notaties in eindige meetkunde en codeertheorie; dit hoofdstuk legt de notaties vast die doorheen de thesis gebruikt worden.

Hoofdstuk 2 handelt over (LDPC) codes die geconstrueerd worden uit eindige meetkundes. In Sectie 2.1 bespreek ik de algemene motivering om dergelijke codes te bestuderen, evenzo vermeld ik de basisresultaten over dit onderwerp.

In Sectie 2.2 bespreek ik de cyclische LDPC codes van affiene en projectieve meetkundes, gezien dit de bekendste klasse LDPC codes zijn die uit eindige meetkundes geconstrueerd worden. In deze sectie presenteer ik ook enkele resultaten die, hoewel ze er uitzien alsof ze al decennia geleden bekend zouden moeten zijn, ik niet in de literatuur heb kunnen terugvinden dus waarvan ik veronderstel dat ze toch nieuw zijn. In het bijzonder toon ik aan dat de orde van deze code (als cyclische code) gelijk is aan de lengte van de code, en bekijk ik ook wat er gebeurt als we op een canonische manier de vector $(1, 1, \ldots, 1)$ verwijderen uit de code. Hierin blijkt een nieuwe toepassing te zitten van een resultaat uit mijn eerdere paper [94] met J. Limbupasiriporn en L. Storme, die ik bespreek in Sectie 7.2. De resultaten in deze sectie zijn (een klein) deel van gezamelijk onderzoek met Y. Fujiwara, en zijn ingediend bij het tijdschrift IEEE Trans. Inform. Theory [45].

In Sectie 2.3 bespreek ik de LDPC codes van punten en rechten in lineaire representaties $T_2^*(\mathcal{K})$. Mijn eerste resultaten op dit onderwerp dateren terug tot voor mijn onderzoek officieel startte, als een spin-off van mijn bachelorproject. Hierin toonde ik aan dat wanneer $\mathcal{K}$ een boog is en de karakteristiek van het veld van de code verschillend is aan die van de onderliggende meetkunde en er is nog een extra conditie voldaan, dan wordt de code volledig voortgebracht door de codewoorden van het kleinste Hamminggewicht. Hierop steunend kon ik ook een algemene dimensieformule voor de code bewijzen. Eerder werk door Pepe et al. [113] toonde dit gedrag enkel tot een zeker klein gewicht, en had in een aantal gevallen ook een tweede type codewoord nodig (waardoor dus uit mijn resultaat volgt dat een codewoord van dit tweede type zelf een lineaire combinatie van codewoorden van het eerste type is). Dit resultaat is gepubliceerd in het tijdschrift Des. Codes Cryptogr. [139]. Helaas bleef het resultaat onbevredigend. De 'extra conditie' sloot het binaire veld uit, wat voor de praktijk een sterke beperking was. In het speciale geval dat $\mathcal{K}$ een kegelsnede min één punt is, bewezen P. Sin and Q. Xiang dezelfde dimensieformule voor binaire codes [124]. Later slaagde ik erin

mijn techniek te verbeteren en zo te tonen dat zodra de karakteristiek van de het veld van de code verschilt van die van de meetkunde, de code voortgebracht is door de codewoorden van kleinste gewicht, evenals dat de dimensieformule ook in deze algemenere setting geldig blijft. Dit nieuwe resultaat verbetert alle voorgaande resultaten: het veralgemeent mijn eigen resultaat uit [140], het bedt de dimensieformule uit [124] in in een grote oneindige klasse van meetkundes en geeft een meetkundige verklaring voor die formule, én het breidt de eigenschap dat de codewoorden van kleinste gewicht de hele code voortbrengen (zoals in [113]) uit zonder restrictie op het gewicht; bovendien verscherpt het verschillende grensen op de minimumafstand uit [113]. Deze resultaten zijn gepubliceerd in Adv. Math. Commun. [140].

In Sectie 2.4 bespreek ik LDPC codes van punten en generatoren in Hermitische variëteit $\mathcal{H}(2n+1,q)$ met q voldoende groot. Deze klasse codes werden eerder al bestudeerd in [114], waar voor $n=1$ de codewoorden van klein gewicht tot ruwweg $\frac{1}{2}q^{3/2}$ geclassificeerd werden als lineaire combinatie van codewoorden van gewicht $2(q+1)$, en voor $n=2$ werd er aangetoond dat de enige codewoorden c met $0<\mathrm{wt}(c)\leq 2(q^3+q^2)$ gewicht $2(q^3+1)$ of $2(q^3+q^2)$ hebben en hun support tot twee specifieke projectieve equivalentieklassen behoort. We breiden het tweede resultaat uit tot willekeurige n en bewijzen een gelijkaardig classificatieresultaat: de enige codewoorden c met $0<\mathrm{wt}(c)\leq 4q^{2n-2}(q-1)$ behoren tot n verschillende projectieve equivalentieklassen, de minimumafstand is in het algemeen $2q^{2n-4}(q^3+1)$ for $n\geq 2$ en, voor elke $\delta>0$, zijn de codewoorden tot gewicht δq^{2n-1} een lineaire combinatie van deze n kleinste types, voor q voldoende groot t.o.v. δ. Deze resultaten zijn gezamelijk onderzoek met M. De Boeck en zijn aanvaard voor publicatie bij Adv. Math. Commun. [30].

In Sectie 2.5 bespreek ik nog twee oneindige klasses LDPC codes, afgeleid uit de partiële meetkundes $S(\mathcal{K})$ en $T_2^*(\mathcal{K})$, met $\mathcal{K}$ een maximale boog. Voor de eerste constructie toon ik aan dat omwisselen van punten en rechten een equivalente code geeft en breid ik een eerder resultaat van [23] uit van hyperovalen naar algemene Denniston- en Mathonbogen. Voor de tweede klasse poneer en bespreek ik een conjectuur die de minimumafstand van deze code linkt aan het bestaan van $(q+t,t)$-bogen van type $(0,2,t)$, waaraan het volgende hoofdstuk is gewijd. Ik geef een partieel bewijs voor het geval $k=4$. Een ingekorte versie van deze resultaten is verschenen in de proceedings van WCC 2011, maar een uiteindelijke publicatie is er nog niet van gekomen.

Hoofdstuk 3 gaat over $(q+t,t)$-bogen type $(0,2,t)$, of kortweg $\mathrm{KM}_{q,t}$-bogen, in Desarguesiaanse projectieve vlakken van even orde. In Sectie 3.1 bespreek ik de motivering om deze bogen te bestuderen, evenals de huidige state of the art. In sectie 3.2 bespreek ik een elegante basis voor de code van dergelijke projectieve vlakken en poneer ik een gemotiveerde conjectuur, ondersteund door computerresultaten, over hoe lineaire onafhankelijkheid tussen incidentievectoren gerelateerd is aan het bestaan van zekere KM-bogen. In Sectie 3.3 bewijs ik deze conjectuur voor $k=q/2$, evenals geef ik indirect een alternatief bewijs voor de classificatie van de projective triads [122, 131]. In Sectie 3.4 formuleer en bewijs ik het hoofdresultaat in dit hoofdstuk: hoewel ik de conjecturen uit het de vorige sectie niet kan bewijzen, heb ik ze toch gebruikt als inspiratie om een nieuwe constructiemethode voor $\mathrm{KM}_{q,q/4}$-bogen te bedenken en heb ik vervolgens deze constructie bewezen via andere technieken. Dit resulteert in zowel een nieuwe oneindige klasse KM-bogen als een grote verhoging van de plausibiliteit van de gemaakte conjectuur. Deze resultaten werden gepubliceerd in Finite Fields Appl. [141].

Hoofdstuk 4 gaat over optimaal blokkerende multiverzamelingen, i.e. blokkerende multiverza-

melingen die de hypervlakken in $\mathrm{PG}(t,q)$ m-voudig blokkeren met zo weinig mogelijk punten. Naast deze intrinsieke meetkundige motivering legt Sectie 4.1 ook nog een tweede gekende motivering uit, vanuit de codeertheorie, namelijk over hoog deelbare Griesmercodes. Sectie 4.2 bespreekt een geheel nieuwe motivering: wanneer we multisets schrijven als lineaire combinatie van hypervlakken, dan blijkt dat deze optimale parameters precies voorkomen wanneer alle coefficienten in die lineaire combinatie niet-negatief zijn. Gewapend met deze nieuwe observatie verbetert Sectie 4.3 zowat alle gekende resultaten over deze klasse multiverzamelingen. Section 4.4 ten slotte toont nog een andere link aan met codeertheorie, namelijk een lijk met de codes van de projectieve ruimte over de ring van gehele getallen modulo een priemmacht. De resultaten in dit hoofdstuk zijn gezamelijk onderzoek met I. Landjev en zijn gepubliceerd in J. Comb. Theory Ser. A [87].

Hoofdstuk 5 gaat over kleine verzamelingen rechten die weinig oneven punten hebben, i.e. weinig punten die op een oneven aantal van deze rechten liggen. In het bijzonder zijn we geïnteresseerd in de kleinste waardes voor $|\mathcal{B}| + |\operatorname{odd}(\mathcal{B})|$, met $\mathcal{B}$ die verzameling rechten en $\operatorname{odd}(\mathcal{B})$ de verzameling van alle oneven punten van die verzameling rechten. In Sectie 5.1 worden verschillende motiveringen gegeven om deze verzamelingen te bestuderen. In Sectie 5.2 wordt het affiene geval besproken; hier classificeer ik alle verzamelingen $\mathcal{B}$ met $|\mathcal{B}| + |\operatorname{odd}(\mathcal{B})| \leq 2q$ as één van acht constructiemethodes óf één resterend geval dat ik niet kon uitsluiten (maar waarvan ik vermoed dat het niet kan optreden). In Sectie 5.3 bespreek ik het projectieve geval, hier kan ik een volledige classificatie voorleggen van alle $\mathcal{B}$ met $|\mathcal{B}| + |\operatorname{odd}(\mathcal{B})| \leq 2q + 2$ als één van zes constructiemethodes. Deze resultaten zijn gepubliceerd in Des. Codes Cryptogr. [142].

Hoofdstuk 6 gaat over meetkundes over eindige kettingringen $\mathfrak{R}$. Sectie 6.1 introduceert de nodige begrippen hiervoor. In Sectie 6.2 presenteer ik een standaardvoorstelling voor deelmodules over $\mathfrak{R}^n$ en voorzie ik efficiënte methodes om de duale module en de span/doornede van dergelijke modules te berekenen. In Sectie 6.3 veralgemeen ik Kantor's stelling over de rang van incidentiematrices van k-spaces bij t-spaces van $\mathrm{PG}(n,q)$, tot willekeurige modules bij vrije modules. Het is verleidelijk te conjectureren dat dit resultaat algemeen zou gelden voor willekeurige modules bij willekeurige modules, maar dat blijkt niet het geval: hiervoor geven we een tegenvoorbeeld. Dit hoofdstuk is gezamelijk onderzoek met I. Landjev; de resultaten van Sectie 6.3 zijn aanvaard voor publicatie inDes. Codes Cryptogr. [88].

In hoofdstuk 7 bespreek ik verscheidene andere resultaten die ik behaald heb en gepubliceerd of ingestuurd heb bij A1-tijdschriften. In Sectie 7.1 presenteer ik een nieuwe ongelijkheid die tegelijk de ongelijkheid tussen het rekenkundig gemiddelde én de ongelijkheid van Turkevich [123] veralgemeent. Dit resultaat is gezamelijk onderzoek met G. Kós en H. Lee en is gepubliceerd in het algemeen wiskundig tijdschrift Proc. Amer. Math. Soc. [81].

In Sectie 7.2 bespreek ik de mogelijke gewichten van codewoorden van groot gewicht die kunnen optreden in de klassieke projectieve ruimtecode. Voor q even reduceert deze studie tot deze van de kleine blokkerende verzamelingen. Voor q oneven, als de orde van het priemveld groot genoeg is, tonen we aan dat er codewoorden zijn van volle gewicht, maar voor kleinere basispriemen is dat niet het geval. In het bijzonder linken we voor $p = 3$ het bestaan van codewoorden van klein gewicht aan het bestaan van 2 mod 3 verzamelingen m.b.t. k-spaces; of deze bestaan is nog steeds een open probleem voor de meeste gevallen. Deze resultaten zijn gezamelijk onderzoek met J. Limbupasiriporn and L. Storme en zijn published in Linear

Algebra and its Applications [94].

In Sectie 7.3 bestudeer ik op de Hermitische unitaal het bestaan van blokkerende verzamelingen waarvan het complement ook een blokkerende verzameling is. Ik toon aan dat deze voor $q \geq 4$ altijd bestaan op de Hermitische unitaal in $\mathrm{PG}(2, q^2)$ en ik bespreek de mogelijke groottes en gerelateerde resultaten. Deze resultaten zijn gezamelijk werk met A. Blokhuis, A.E. Brouwer, D. Jungnickel, V. Krčadinac, S. Rottey, L. Storme and T. Szőnyi en zijn ingestuurd naar Finite Fields Appl. [16].

Naast deze resultaten heb ik ook gewerkt op quantumcodeertheorie, samen met Yuichiro Fujiwara et al. In deze onderzoeken behaalden we resultaten over entanglement-assisted quantum LDPC codes, deze zijn gepubliceerd in Phys. Rev. A [43]; over high-rate quantum LDPC codes assisted by reliable qubits, deze zijn ingestuurd naar IEEE Trans. Inform. Theory [44]; en over quantum synchronizable codes afkomstig uit eindige meetkundes, deze zijn ingestuurd naar IEEE Trans. Inform. Theory [45]. Echter, de nodige machinerie introduceren om deze quantumtheoretische resultaten op een wiskundig verantwoorde manier uit te leggen, zou teveel tijd vergen, dus ik verwijs de geïnteresseerde lezer hiervoor naar de geciteerde papers.

Ook heb ik gewerkt op de toepassingen van mijn theoretisch onderzoek, om de kloof tussen het theoretisch wiskundig onderzoek en toegepast ingenieursonderzoek te dichten. Ik heb een (bijna afgewerkt) manuscript geschreven dat de eigenschap dat codewoorden van klein gewicht voortgebracht zijn door een kleine verzameling codewoorden, gebruikt om gekende informatie over de doorgestuurde gegevens om te zetten in een betere decodeerprestatie; daarnaast heb ik ook een variant op het LDPC decodeeralgoritme ontworpen die een grootte-orde minder geheugen gebruikt, ten koste van slechts een klein verlies in performantie. Deze resultaten zijn echter nog niet ingediend voor publicatie en worden bijgevolg weggelaten uit de thesis, om te vermijden dat deze thesis gerefereerd wordt in plaats van de papers die er potentieel uit voortvloeien, maar wie geïnteresseerd is mag mij deze resultaten gerust vragen.

Ten slotte heb ik ook grote computationele bibliotheken aangelegd om efficiënt met verscheidene objecten uit codeertheorie en eindige meetkunde te werken, zoals vectorruimtes, lineaire codes, Grassmannvariëteiten van projectieve deelruimten, operaties op deelruimten, groepsacties op deelruimten, matrixalgebra over eindige velden, eindige kettingringen, etc. Speciale aandacht heb ik daarbij besteed aan het LDPC decoderingsalgoritme, dat ik in OpenCL geschreven heb om het zo op GPU te kunnen uitvoeren, wat veel sneller en zuiniger is dan dit op CPU uitrekenen. In 2012 kreeg onze vakgroep een krediet van 9000 euro van het FCWO om GPU hardware te kopen, waarmee ik zelfstandig een GPU-rekencomputer gebouwd heb. In Appendix A vermeld ik de uiteindelijke configuratie, de problemen die ik ermee tegengekomen ben en hoe ik ze opgelost heb, en leg ik ook het volledige LDPC decoderingsalgoritme uit, zowel theoretisch als hoe ik het in OpenCL geïmplementeerd heb.

Bibliography

[1] R. W. Ahrens and G. Szekeres, On a combinatorial generalization of 27 lines associated with a cubic surface, *J. Austral. Math. Soc.* **10** (1969), 485–492.

[2] A. Al-Azemi, A. Betten and D. Betten, Unital Designs with Blocking Set. (Preprint).

[3] E. Amador, R. Pacalet and V. Rezard, Optimum LDPC decoder: a memory architecture problem, Proceedings of the 46th ACM/IEEE Design Automation Conference (2009), 891–896.

[4] E. Artin, Algebraic numbers and algebraic functions, London: Gordon and Breach, 1967.

[5] E.F. Assmus, Jr. and J.D. Key, Designs and their codes. *Cambridge University Press*, 1992.

[6] E.F. Assmus Jr., J.D. Key, Handbook of Coding Theory, Vol. II (edited by V. S. Pless and W. C. Huffman, North Holland, Amsterdam, 1998).

[7] B. Bagchi and S.P. Inamdar, Projective Geometric Codes, *J. Combin. Theory, Ser. A* **99** (2002), 128–142.

[8] P. Balister, B. Bollobás, Z. Füredi and J. Thompson, Minimal symmetric differences of lines in projective planes, *unpublished, available as arXiv:1303.4117 [math.CO].*

[9] S. Ball, A. Blokhuis and F. Mazzocca, Maximal arcs in Desarguesian planes of odd order do not exist, *Combinatorica* **17** (1997), 31–41.

[10] S. Ball, R. Hill, I. Landjev and H.N. Ward, On $(q^2+q+2, q+2)$-arcs in the projective plane PG$(2, q)$, *Des. Codes Cryptogr.* **24** (2001), 205–224.

[11] Th. Beth, D. Jungnickel and H. Lenz, Design theory, Second edition, Encyclopedia of Mathematics and its Applications, Cambridge University Press, Cambridge, 1999.

[12] G. Birkhoff, Subgroups of abelian groups, *Proc. of The London Math. Society* **38** (1934), 385–401.

[13] R.E. Blahut, Algebraic codes for Data Transmission, Cambridge Univ. Press (New York, 2003).

[14] A. Blokhuis, A.E. Brouwer and T. Szőnyi, The number of directions determined by a function f on a finite field, *J. Combin. Theory, Ser. A* **70** (1995), 349–353.

[15] A. Blokhuis, A.E. Brouwer and H.A. Wilbrink, Hermitian unitals are codewords. *Discrete Math.* **97** (1991), 63–68.

[16] A. Blokhuis, A. Brouwer, D. Jungnickel, V. Krčadinac, S. Rottey, L. Storme, T. Szőnyi and P. Vandendriessche, Blocking sets of the Hermitian unital, *submitted to Finite Fields Appl.*

[17] A. Blokhuis, R. Pellikaan and T. Szőnyi, Blocking sets of almost Rédei type. *J. Combin. Theory, Ser. A* **78** (1997), 141–150.

[18] A. Blokhuis and V. Lev, Flat-Containing and Shift-Blocking sets in $\mathbb{F}_q^r$. *Mosc. J. Comb. Number Th.*, to appear.

[19] R.C. Bose and R.C. Burton, A characterization of flat spaces in a finite geometry and the uniqueness of the Hamming and the McDonald codes. *J. Combin. Theory* **1** (1966), 96–104.

[20] K.A. Bush, Orthogonal arrays of index unity, *Ann. Math. Stat.* **23** (1952), 426–434.

[21] N.J. Calkin, J.D. Key and M.J. De Resmini, Minimum Weight and Dimension Formulas for Some Geometric Codes, *Designs, Codes and Cryptography* **17** (1999), 105–120.

[22] P. J. Cameron and J. H. Van Lint, "Designs, Graphs, Codes and their Links," Cambridge University Press, 1991.

[23] C. Castleberry, K. Hunsberger and K.E. Mellinger, LDPC codes arising from hyperovals, *Bull. Inst. Comb. Appl.* **58** (2010), 59–72.

[24] B. Chandler, P. Sin, Q. Xiang, The invariant factors of the incidence matrices of points and subspaces in PG(n,q) and AG(n,q), *Trans. Amer. Math. Soc.* **358** (2006), 4935–4957.

[25] D. Changyan, D. Proietti, I.E. Telatar, T.J. Richardson and R.L. Urbanke, Finite-length analysis of low-density parity-check codes on the binary erasure channel, *IEEE Trans. Inform. Theory* **48** (2002), 1570–1579.

[26] D.K. Chow, A geometric approach to coding theory with application to information retrieval, Tech. Rep. Report R-368 (Coordinated Science Laboratory, University of Illinois, 1967.

[27] W. E. Clark, D. A. Drake, Finite chain rings, *Abh. Math. Sem. der Univ. Hamburg* **39**(1974), 147–153.

[28] Y. Dai, N. Chen and Z. Yan, Memory Efficient Decoder Architectures for Quasi-Cyclic LDPC Codes, *IEEE Trans. Circ. Syst. I* **55** (2008), 2898–2911.

[29] F. De Clerck and H. Van Maldeghem, *On linear representations of* (α, β)*-geometries*, *European J. Combin.* **15** (1994), 3–11.

[30] M. De Boeck and P. Vandendriessche, On the dual code of points and generators of the Hermitian variety $H(2n+1, q^2)$, accepted at *Adv. Math. Commun.*

[31] I. Debroey and J. A. Thas, *Semi partial geometries in $AG(2, q)$ and $AG(3, q)$*, *Simon Stevin* **51** (1978), 195–209.

[32] R.H.F. Denniston, Some maximal arcs in finite projective planes, *J. Combin. Theory* **6** (1969), 317–319.

[33] I.B. Djordjevic and B.V. Vasic, Projective geometry LDPC codes for ultralong-haul WDM high-speed transmission, *IEEE Photonics Technology Letters* **15** (2003), 784–786.

[34] I.B. Djordjevic, S. Sankaranarayanan and B.V. Vasic, Projective-Plane Iteratively Decodable Block Codes for WDM High-Speed Long-Haul Transmission Systems. *J. Lightwave Technol.* **22** (2004), 695–702.

[35] P. Erdős, On a combinatorial problem. *Nordisk Mat. Tidskr.* **11** (1963), 5–10.

[36] P. Erdős and L. Lovász, Problems and results on 3-chromatic hypergraphs and some related questions, pp. 609–627, in: *Infinite and Finite Sets*, Proc. Keszthely 1973, Colloq. Math. Soc. János Bolyai 10, Budapest, 1975.

[37] S. Fan and W. Han, Character sums over Galois rings and primitive polynomials over finite fields, *Fin. Fields Appl.* **10** (2004), 36–52.

[38] C. Feng, R.W. N'obrega, F.R. Kschischang and D. Silva, Communication over Finite-Chain-Ring Matrix Channels, Submitted to IEEE Transactions on Information Theory.

[39] J.C. Fisher, J.W.P. Hirschfeld and J.A. Thas, Complete arcs in planes of square order. *Ann. Discrete Math.* **30** (1986), 243–250.

[40] M.J. Flynn, Some Computer Organizations and Their Effectiveness, *IEEE Trans. Comput.* **C-21** (1972), 948–960.

[41] M. P. C. Fossorier, Quasicyclic low-density parity check codes from circulant permutation matrices, *IEEE Trans. Inform. Theory* **50** (2004), 1788–1793.

[42] A. Frumkin and A. Yakir, Rank of inclusion matrices and modular representation theory, *Israel J. Math.* **71** (1990), 309–320.

[43] Y. Fujiwara, D. Clark, P. Vandendriessche, M. De Boeck and V.D. Tonchev. Entanglement-assisted quantum low-density parity-check codes. *Phys. Rev. A* **82** (2010), id 042338.

[44] Y. Fujiwara, A. Gruner and P. Vandendriessche, High-Rate Quantum Low-Density Parity-Check Codes Assisted by Reliable Qubits, submitted to IEEE Trans. Inform. Theory.

[45] Y. Fujiwara and P. Vandendriessche, Quantum Synchronizable Codes from Finite Geometries, submitted to IEEE Trans. Inform. Theory.

[46] Z. Füredi, Matchings and covers in hypergraphs. *Graphs and Combinatorics* **4** (1988), 115–206.

[47] A. Gács and Zs. Weiner, On $(q+t,t)$-arcs of type $(0,2,t)$, *Des. Codes Cryptogr.* **29** (2003), 131–139.

[48] R.G. Gallager, Low density parity check codes, *IRE Trans. Inform. Theory* **8** (1962), 21–28.

[49] D. Ghinelli and D. Jungnickel, Some geometric aspects of finite abelian groups. *Rend. Mat., Ser. VII* **26** (2006), 29–68.

[50] M. Grassl, Code Tables: Bounds on the parameters of various types of codes, www.codetables.de

[51] J.H. Griesmer, A bound for error-correcting codes, *IBM J. Res. Develop.* **4** (1960), 532–542.

[52] M. Hall, Ovals in the Desarguesian plane of order 16, *Ann. Mat. Pura Appl.*, **102** (1975), 159–176.

[53] N. Hamada, Characterization of minihypers in a finite projective geometry and its applications to error-correcting codes, *Bull. Osaka Women's Univ.* **24** (1987), 1–24.

[54] N. Hamada, On the p-rank of the incidence matrix of a balanced or partially balanced incomplete block design and its applications to error-correcting codes, *Hiroshima Math. J.* **3** (1973), 153–226.

[55] N. Hamada and T. Helleseth, Codes and minihypers. Optimal codes and related topics, Proceedings of the EuroWorkshop on Optimal codes and related topics (Sunny Beach, Bulgaria, June 10-16, 2001), 79–84.

[56] L. Hellerstein, G. Gibson, R. Karp, R. Katz and D. Patterson, Coding techniques for handling failures in large disk arrays, *Algorithmica* **12** (1994), 18–208.

[57] P. Herdt, $[n,k,d]_q$-Codes mit $k \geq 3$, $d = rq^{k-2}$ und $n = \lceil r/q \rceil + r + rq + \cdots + rq^{k-2}$, $r \in \mathbb{N}$, Msc. Thesis at Justus-Liebig-Universität Gießen, Germany (2008).

[58] R. Hill and H.N. Ward, A geometric approach to classifying Griesmer codes, *Des. Codes Cryptogr.* **44** (2007), 169–196.

[59] J. W. P. Hirschfeld, "Projective Geometries over Finite Fields," 2^{nd} edition, Oxford University Press, 1998.

[60] J.W.P. Hirschfeld and J.A. Thas. General Galois Geometries. *Oxford Mathematical Monographs*, Oxford University Press, Oxford, 1991.

[61] J.W.P. Hirschfeld and L. Storme, The packing problem in statistics, coding theory and finite projective spaces: update 2001, Developments in Mathematics Vol. 3, Kluwer Academic Publishers, Finite Geometries, Proceedings of the Fourth Isle of Thorns Conference (Chelwood Gate, July 16-21, 2000) (Eds. A. Blokhuis, J.W.P. Hirschfeld, D. Jungnickel and J.A. Thas), 201–246.

[62] J.W.P. Hirschfeld, Finite Projective Spaces of Three Dimensions. Oxford: Oxford University Press 1985.

[63] T. Honold, I. Landjev, Linearly representable codes over chain rings, *Abh. Math. Sem. der Univ. Hamburg*, **69**, 1999, 187–203.

[64] T. Honold, I. Landjev, Linear Codes over Finite Chain Rings and Projective Hjelmslev Geometries, in: Codes over Rings (ed. P. Solé), World Scientific, 2009, 60–123.

[65] T. Honold and I. Landjev, Codes over rings and ring geometries, in: ''Current research topics in Galois geometries" (eds. L.Storme and Jan De Beule) NOVA Publishers, 161–186 (2012).

[66] X.-Y. Hu, M.P.C. Fossorier and E. Eleftheriou, On the computation of the minimum distance of low-density parity-check codes, *Proc. IEEE Intl. Conf. Commun.* (2004), 767–771.

[67] T. Illés, T. Szőnyi and F. Wettl, Blocking sets and maximal strong representative systems in finite projective planes, *Mitt. Math. Sem. Giessen* **201** (1991), 97–107.

[68] J. Jin, C. Tsui, An Energy Efficient Layered Decoding Architecture for LDPC Decoder, IEEE Trans. VLSI Syst. **18** (2010), 1185–1195.

[69] J. Jin, C. Tsui, A low power layered decoding architecture for LDPC decoder implementation for IEEE 802.11n LDPC codes, 2008 ACM/IEEE International Symposium on Low Power Electronics and Design (2008), 253–258.

[70] S. J. Johnson and S. R. Weller, Construction of low-density parity-check codes from combinatorial designs, in "Proceedings of the IEEE Information Theory Workshop," Cairns, (2001), 90–92.

[71] S. J. Johnson and S. R. Weller, Construction of low-density parity-check codes from Kirkman triple systems, in "Proceedings of the IEEE Globecom Conference," San Antonio, 2001.

[72] S. J. Johnson and S. R. Weller, Codes for iterative decoding from partial geometries, in "Proceedings of the IEEE International Symposium on Information Theory," Switzerland, 2002.

[73] S. J. Johnson and S. R. Weller,Regular low-density parity-check codes from oval designs, *Eur. Trans. Telecommun.* **14** (2003), 399–409.

[74] S.J. Johnson and S.R. Weller, Codes for Iterative Decoding From Partial Geometries, *IEEE Trans. Commun.* **52** (2004), 236–243.

[75] W.M. Kantor, On Incidence Matrices of Finite Projective and Affine Spaces, Math. Z. 124,315–318 (1972).

[76] J.D. Key, T.P. McDonough and V.C. Mavron, An upper bound for the minimum weight of the dual codes of Desarguesian planes, *European J. Combin.* **30** (2009), 220–229.

[77] J. L. Kim, U. Peled, I. Perepelitsa, V. Pless and S. Friedland, Explicit construction of families of LDPC codes with no 4-cycles, *IEEE Trans. Inform. Theory* **50** (2004), 2378–2388.

[78] J.-L. Kim, K.E. Mellinger and L. Storme, Small weight code words in LDPC codes defined by (dual) classical generalized quadrangles, *Des. Codes Cryptogr.* **42** (2007), 73–92.

[79] A. Klein, K. Metsch and L. Storme. Small maximal partial spreads in classical finite polar spaces. *Adv. Geom.* **10** (2010), 379–402.

[80] G. Korchmáros and F. Mazzocca, On $(q+t,t)$-arcs of type $(0,2,t)$ in a Desarguesian plane of order q, *Math. Proc. Camb. Phil. Soc.* **108** (1990), 445–459.

[81] G. Kós, H. Lee and P. Vandendriessche, Simultaneous extensions of Turkevich's inequality and the weighted AM-GM inequality, *Proc. Amer. Math. Soc.* **140** (2012), 971–975.

[82] Y. Kou, S. Lin and M.P.C. Fossorier, Low-density parity-check codes based on finite geometries: a rediscovery and new results, *IEEE Trans. Inform. Theory* **47** (2001), 2711–2736.

[83] K.M. Krishnan and P. Shankar, Computing the Stopping Distance of a Tanner Graph Is NP-Hard, *IEEE Trans. Inform. Theory* **53** (2007), 2278–2280.

[84] I. Landjev and L. Storme, A study of $(x(q+1),x;2,q)$-minihypers, *Des. Codes Cryptogr.* **54** (2010), 135–147.

[85] I. Landjev and L. Storme, Galois geometries and coding theory, in: Current Research Topics in Galois Geometry (J. De Beule, L. Storme, eds.), Nova Science Publishers, 2011, 185–212.

[86] I. Landjev and P. Vandendriessche, On the Point-by-Subspace Incidence Matrices of Projective Hjelmslev Spaces, *Compt. Rend. Acad. Bulg. des Sci.*, 2014, to appear.

[87] I. Landjev and P. Vandendriessche, A study of (xv_t,xv_{t-1})-minihypers in $\mathrm{PG}(t,q)$, *J. Comb. Theory Ser. A* **119** (2012), 1123–1131.

[88] I. Landjev and P. Vandendriessche, On the Rank of Incidence Matrices in Projective Hjelmslev Spaces, accepted at Des. Codes Cryptogr (after minor revisions).

[89] M. Lavrauw, L. Storme and G. Van de Voorde. Linear codes from projective spaces. *Error-Correcting Codes, Finite Geometries, and Cryptography.* A.A. Bruen and D.L. Wehlau, editors. *AMS Contemporary Mathematics (CONM) book series* **523** (2010), 185–202.

[90] M. Lavrauw, L. Storme and G. Van de Voorde, On the code generated by the incidence matrix of points and hyperplanes in $\mathrm{PG}(n,q)$ and its dual. *Des. Codes Cryptogr.* **48** (2008), 231–245.

[91] X. Li, C. Zhang and J. Shen, Regular LDPC codes from semipartial geometries, *Acta Appl. Math.* **102** (2008), 25–35.

[92] R. Lidl and H. Niederreiter, Finite Fields, 2nd edition (Cambridge Univ. Press, 1997).

[93] J. Limbupasiriporn, Partial Permutation Decoding for Codes from Designs and Finite Geometries, PhD Thesis, Clemson University (2005).

[94] J. Limbupasiriporn, L. Storme and P. Vandendriessche, Large weight code words in projective space codes, *Linear Algebra and its Applications* **437** (2012), 809–816.

[95] Z. Liu and D.A. Pados, LDPC codes from generalized polygons, *IEEE Trans. Inform. Theory* **51** (2005), 3890–3898.

[96] I. G. MacDonald, Symmetric Functions and Hall Polynomilas, Oxford University Press, 2nd edition, 1995.

[97] D. J. C. MacKay and M. C. Davey, *Evaluation of Gallager codes for short block length and high rate applications*, in "Codes, Systems and Graphical Models" (eds. B. Marcus and J. Rosenthal), Springer-Verlag, New York, (2000), 113–130.

[98] D.J.C. MacKay and R.M. Neal, Near Shannon limit performance of low density parity check codes. *Electron. Lett.* **32** (1996), 1645–1646.

[99] M.M. Mansour and N.R. Shanbhag, High-Throughput LDPC Decoders, *IEEE Trans. VLSI Syst.* **11** (2003), 976–996.

[100] G. A. Margulis, *Explicit constructions of graphs without short cycles and low density codes*, *Combinatorica* **2** (1982), 71–78.

[101] R. Mathon, New maximal arcs in Desarguesian planes, *J. Combin. Theory, Ser. A* **97** (2002), 353–368.

[102] B. R. McDonald, Finite rings with Identity, Marcel Dekker, New York, 1974.

[103] G.E. Moore, Cramming more components onto integrated circuits, *Electronics* **38** (1965), 114–117.

[104] G.E. Moore, Progress in digital integrated electronics, *International Electron Devices Meeting* **21** (1975), 11–13.

[105] G.E. Moorhouse, Bruck nets, codes, and characters of loops, *Des. Codes Cryptogr.* **1** (1991), 7–29.

[106] M. Müller and M. Jimbo, Erasure-resilient codes from affine spaces, *Discrete Appl. Math.* **143** (2004), 292–297.

[107] A. A. Nechaev, Finite principal ideal rings, Russian Acad. of Sciences, *Sbornik Mathematics* **209** (1973), 364–382.

[108] T.M.N. Ngatched, F. Takawira and M. Bossert, An improved decoding algorithm for finite-geometry LDPC codes, *IEEE Trans. Commun.* **57** (2009), 302–306.

[109] D. Oh and K.K. Parhi, Low Complexity Decoder Architecture for Low-Density Parity-Check Codes, *Journal of Signal Processing Systems* **56** (2006), 217–228.

[110] C. M. O'Keefe and T. Penttila, Hyperovals in PG$(2, 16)$, *European J. Combin.*, **12** (1991), 51–59.

[111] M.S. Papamarcos and J.H. Patel, A low-overhead coherence solution for multiprocessors with private cache memories. Proceedings of the 11th annual International Symposium on Computer Architecture (1984), 348–354.

[112] T. Penttila and G. F. Royle, Classification of hyperovals in PG$(2, 32)$, *J. Geom.* **50** (1994), 151–158.

[113] V. Pepe, L. Storme and G. Van de Voorde, Small weight code words in the LDPC codes arising from linear representations of geometries, *J. Combin. Des.* **17** (2009), 1–24.

[114] V. Pepe, L. Storme and G. Van de Voorde. On codewords in the dual code of classical generalised quadrangles and classical polar spaces. *Discrete Math.* **310** (2010), 3132–3148.

[115] J. Radhakrishnan and A. Srinivasan, Improved bounds and algorithms for hypergraph 2-coloring. *Random Structures Algorithms* **16** (2000), no. 1,4–32.

[116] J. Rosenthal and P. O. Vontobel, Construction of LDPC codes using Ramanujan graphs and ideas from Margulis, in "Proceedings of the 38th Allerton Conference on Communications, Control and Computing" (eds. P.G. Voulgaris and R. Srikant), Monticello, (2000), 248–257.

[117] P. Saunders and A.D. Fagan, A Low Memory FPGA Based LDPC Decoder Architecture for Quasi-Cyclic LDPC codes, 2006 Irish Signals and Systems Conference (2006), 223–228.

[118] B. Segre, Sui k-archi nei piani finiti di caratteristica due (in Italian), *Rev. Math. Pures Appl.* **2** (1957), 289–300.

[119] B. Segre, Ovals in a finite projective plane, *Canad. J. Math.* **7** (1955), 414–416.

[120] B. Segre, Curve razionali normali ek-archi negli spazi finite, *Ann. Mat. Pura Appl. IV.* **39** (1955), 357–379.

[121] M. Seib, Unitäre Polaritäten endlicher projectiver Ebenen. *Arch. Math.* **21** (1970), 103–112.

[122] D. Senato, Blocking sets di indice tre, *Rend. Accad. Sci. Fis. Mat. Napoli* **19** (1982), 89–95.

[123] V. Senderov and E. Turkevich, Problem M506, *Kvant* **10** (1979), 35–35.

[124] P. Sin and Q. Xiang, On the dimension of certain LDPC codes based on q-regular bipartite graphs, *IEEE Trans. Inform. Theory* **52** (2006), 3735–3737.

[125] M. Sipser and D. A. Spielman, Expander codes, *IEEE Trans. Inform. Theory* **42** (1996), 1710–1722.

[126] K.J.C. Smith, On the p-rank of the incidence matrix of points and hyperplanes in a finite projective geometry, *J. Combin. Theory* **7** (1969), 122–129.

[127] G. Solomon and J.J. Stiffler, Algebraically punctured cyclic codes, *Inform. and Control* **8** (1965), 170–179.

[128] H. Stichtenoth, Algebraic Function Fields and Codes. Springer Verlag 1993.

[129] P. Sziklai, On small blocking sets and their linearity. *J. Combin. Theory, Ser. A* **115** (2008), 1167–1182.

[130] T. Szőnyi and Zs. Weiner, Small blocking sets in higher dimensions. *J. Combin. Theory, Ser. A* **95** (2001), 88–101.

[131] T. Szőnyi, Combinatorial Problems for Abelian Groups Arising from Geometry, *Periodica Polytechnica* **19** (1991), 91–100.

[132] T. Szőnyi, Blocking sets in Desarguesian affine and projective planes. *Finite Fields Appl.* **3** (1997), 187–202.

[133] R. M. Tanner, A recursive approach to low complexity codes, *IEEE Trans. Inform. Theory* **27** (1981), 533–547.

[134] R. M. Tanner, Minimum distance bounds by graph analysis, *IEEE Trans. Inform. Theory* **47** (2001), 808–821.

[135] R. M. Tanner, D. Sridhara, A. Sridharan, T. E. Fuja and J. D. Costello Jr, LDPC block codes and convolutional codes based on circulant matrices, *IEEE Trans. Inform. Theory* **50** (2004), 2966–2984.

[136] J. A. Thas, Partial geometries in finite affine spaces, *Math. Z.* **158** (1978), 1–13.

[137] A. Tietäväinen, On the nonexistence of perfect codes over finite fields, *SIAM J. Appl. Math.* **24** (1973), 88–96.

[138] Y.-L. Ueng, C.-J. Yang and C.-J. Chen, A shuffled message-passing decoding method for memory-based LDPC decoders, 2009 IEEE International Symposium on Circuits and Systems (2009), 892–895.

[139] P. Vandendriessche, Some low-density parity-check codes derived from finite geometries, *Des. Codes Cryptogr.* **54** (2010), 287–297.

[140] P. Vandendriessche, LDPC codes associated with linear representations of geometries, *Adv. Math. Commun.* **4** (2010), 405–417.

[141] P. Vandendriessche, Codes of Desarguesian projective planes of even order, projective triads and $(q+t,t)$-arcs of type $(0,2,t)$, *Finite Fields Appl.* **17** (2011), 521–531.

[142] P. Vandendriessche, On small line sets with few odd-points, accepted at Des. Codes Cryptogr., doi 10.1007/s10623-014-9920-1

[143] J.H. Van Lint, On the nonexistence of perfect 2- and 3-Hamming-error-correcting codes over GF (q), *Information and Control* **16** (1970), 396–401.

[144] J.H. Van Lint, A survey of perfect codes, *Rocky Mountain J. Math.* **5** (1975), 199–224.

[145] J.H. Van Lint, Introduction to Coding Theory, Springer-Verlag, Berlin, 1982.

[146] A. Vardy, The intractability of computing the minimum distance of a code, *IEEE Trans. Inform. Theory* **43** (1997), 1757–1766.

[147] P. O. Vontobel and R. M. Tanner, Construction of codes based on finite generalized quadrangles for iterative decoding, in "Proceedings of 2001 IEEE International Symposium Information Theory," Washington, (2001), 233–233.

[148] H.N. Ward, Divisibility of codes meeting the Griesmer bound, *J. Comb. Theory Ser. A* **83** (1998), 79–93.

[149] H. Xiao and A. H. Banihashemi, Improved progressive-edge-growth (PEG) construction of irregular LDPC codes, *IEEE Commun. Lett.* **8**, 715–717.

[150] J. Zhang, J.S. Yedidia and M.P.C. Fossorier, Low-Latency Decoding of EG LDPC Codes, *Journal of Lightwave Technology* **25** (2007), 2879–2886.

[151] X. Zhang and F. Cai, Reduced-Complexity Decoder Architecture for Non-Binary LDPC Codes, *IEEE Trans. Commun.* **53** (2005), 1288–1299.

[152] A. Zinoviev and V.K. Leontiev, The nonexistence of perfect codes over Galois fields, *Problems of Control and Information* **2** (1973), 123-132.